Title
Mathematical Modeling and Computational Methods for Structured Populations

Author
Xia, Mingtao

Publication Date
2023

Peer reviewed|Thesis/dissertation

University of California
Los Angeles

Mathematical Modeling and Computational Methods for Structured Populations

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Mathematics

by

Mingtao Xia

2023

Abstract of the Dissertation

Mathematical Modeling and Computational Methods for Structured Populations

by

Mingtao Xia

Doctor of Philosophy in Mathematics

University of California, Los Angeles, 2023

Professor Tom Chou, Chair

Structured population models are fundamental in the fields of biology, ecology, and social sciences, as they provide both theoretical insights and practical applications. Different structured population models range from modeling cellular population proliferation and population dynamics to simulating disease spread on social networks. However, there has been little work on modeling populations across different scales that could link individual behavior to population dynamics. Additionally, for existing mathematical models on structured populations, several computational challenges arise as how to develop efficient numerical solvers to simulate those models and to control the dynamics of those models.

Overall, my dissertation covers three related topics: modeling structured populations, developing efficient numerical solvers to simulate these models, and developing control algorithms to control population dynamics. Specifically, my dissertation focuses on modeling and devising algorithms for two types of structured populations: i) age, size, or added size-structured cell population for describing cellular proliferation and ii) the structured infected-time- or number-of-contact-based human population for describing disease spread.

Regarding the structured cellular population, we derive mathematical models at both the macroscopic population dynamics level and microscopic individual behavior level, leading to structured partial differential equation (PDE) models for cellular proliferation with different

structure variables such as cellular age, size, or added size.

Next, we develop an efficient adaptive spectral method for numerically solving spatiotemporal PDEs, which was inspired by simulating the blowup behavior in the unbounded-domain PDE model for cellular populations. In addition to the structured population models, the adaptive spectral method proves efficient and accurate in solving a wide range of spatiotemporal PDEs in unbounded domains such as the Schrödinger equations in quantum mechanics.

Regarding the structured human population, we introduce an infected-time-structured PDE model and a number-of-contact-structured ODE model for simulating disease spread, e.g., COVID-19, in the population. Then, for the number-of-contact-structured ODE model, we develop classic Pontryagin-maximum-principle-based and reinforcement-learning-based optimal control algorithms. These two algorithms can effectively mitigate the spread of disease by appropriately allocating limited test kits or vaccination resources.

The dissertation of Mingtao Xia is approved.

Shenshen Wang

Marcus Roper

Chris Anderson

Tom Chou, Committee Chair

University of California, Los Angeles

2023

Table of Contents

List of Figures

List of Tables

Acknowledgments

Completing a PhD during the COVID-19 pandemic has been a unique experience. With most academic activities taking place online and limited in-person contact, it was the support and encouragement of those around me that helped me through this difficult time and enabled me to finish my studies at UCLA. As I prepare to graduate, I would like to express my sincere thanks to the many people who have supported me along the way.

First and foremost, I would like to thank my advisor, Professor Tom Chou. I first worked with Professor Tom Chou when I was a third-year undergraduate student at Peking University when I participated in the UCLA CSST Summer Program. Since then, I began my research journey in the unknown field of biomathematical modeling and computation with Professor Tom Chou's help to understand the physical mechanisms that are behind mathematical modeling. Professor Tom Chou always supports me in exploring all different topics and fields that I am interested in, and therefore, I have the chance to carry out research across different fields and establish collaboration with different scholars in modeling, numerical analysis, as well as control theories. Most importantly, Professor Tom Chou reveals to me how to rigorously carry out research, including how to discover new problems and evaluate them, how to tune mathematical models and devise numerical experiments, as well as how to do careful scientific writing. Finishing a PhD in applied mathematics seems difficult for most people, but with the help of Professor Tom Chou, I quite enjoyed my four-year PhD studies in applied mathematics at UCLA. In the future, I wish I could continue my research in academia and become an advisor like my advisor, Professor Tom Chou who gives students the freedom to explore different research directions but can always give support to students when they need support and help.

Next, I wish to thank my family, my parents, and my fiancee Qijing. Due to COVID-19, I haven't got a chance to return to China for about four years, and I also haven't met my fiancee for about two years. Yet, they support me both mentally and physically in the era of COVID isolation when I need company and help. It is their unselfish love that supports

me through the darkest time when I cannot go home to meet my family. When I feel lonely and need help, they are always making phone calls to support me. Four years could greatly change a person, and the prevalence of COVID-19 for three years change every one of us in many aspects. I did not get frustrated, disappointed, or isolated because my family is always caring for me and ready to help me in every aspect even though I cannot go to see them. Also, Qijing gave me much support in my research, and I greatly acknowledged that. When I have a chance to return home and reunite with my parents, I cannot wait to share with them my gratefulness for their support, and when I can meet Qijing soon, I wish we could get married soon.

I also would like to thank my collaborators and members in and outside of Professor Tom Chou's group. I collaborated with Professor Sihong Shao for more than three years in unbounded-domain spectral methods. Professor Sihong Shao provided me with many useful suggestions for developing adaptive spectral methods. Chapter 5 is a version of [XSC21a]; Chapter 6 is a version of [XSC21b]; Chapter 7 is a version of [CSX23], all of which are based on collaborating with Professor Sihong Shao. This experience opens a new door for my future research and greatly motivates me in doing further research in unbounded-domain problems which is an interesting yet underexplored field. Also, Professor Lucas Böttcher, a former postdoc in our group and now a professor at Frankfurt School of Finance and Management helps me in learning machine learning and applying machine learning to numerical analysis as well as control theories. He introduces many useful machine-learning techniques and some really fascinating ideas to me so that I combine modern machine-learning tools with spectral methods as well as traditional control algorithms. In this dissertation, Chapter 8 is a version of [XBC23]; Chapter 9 is a version of [BXC20]; Chapter 10 is a version of [XBC22]. Those papers are based on collaboration with Professor Lucas Böttcher who introduces modern control theories and physics-informed neural networks to me. I would like to thank my co-author Professor Chris D. Greenman for his participation and suggestions on Chapter 2, which is a version of [XGC20].

My group mate Xiangting collaborates with me on various different projects, and his help

and suggestions play an important role in finishing and polishing those projects. Also, our senior postdoc groupmate Yue, who is also a PKU alumnus, gave me much advice on job applications, and I greatly acknowledge his help.

I would also like to thank my friends who helped and supported me. My peers in the mathematics department at UCLA often gather together and they helped me a lot to settle down especially when I first arrived in LA. Also, I lived with my former roommate Yuxuan for three years, and he helped me a lot during the COVID-19 era. I had a very happy time playing badminton at UCLA with some friends here, which is one of the quite limited entertainment in the COVID-19 era. Moreover, at the beginning of the COVID-19 era when I was unable to return home, many friends in China gave me their support and talked to me to help me get through that stage. Here, I wish to thank my friends who accompanied me through the four years.

Also, I would like to express my gratitude to the instructors whom I worked with as a TA and the students in my class. Your support and understanding encouraged me to continue working as a teacher to share more applied mathematics. I am also grateful to my fans on social media who took the time to watch my videos on applied mathematics talks and course videos and wrote encouraging comments to support me.

Finally, I acknowledge IOP Publishing for permitting me to reproduce my work [BXC20] and [XC21] in this dissertation. I also acknowledge SIAM for permitting me to reproduce my work [XSC21b] and [XSC21a] in this dissertation.

Thank you all for making my PhD journey a memorable and fulfilling experience.

Vita

1997	Born, Chengdu, Sichuan, China.
2015-2019	Bachelor of Science, School of Mathematical Science, Peking University, China.
2019-2020	Ph.D. student, Mathematics Department, UCLA, USA.
2020-present	Ph.D. candidate, Mathematics Department, UCLA, USA.

Publications

Mingtao Xia, Lucas Böttcher, Tom Chou, *Spectrally adapted physics-informed neural networks for solving unbounded domain problems*, **4**, 025024, Machine Learning: Science and Technology, (2023).

Tom Chou, Sihong Shao, Mingtao Xia, *Adaptive Hermite Spectral methods in unbounded domians*, Applied Numerical Mathematics, 183, 201-220, (2023)

Renaud Dessalles, Yunbei Pan, Mingtao Xia, Davide Maestrini, Maria R. D'Orsogna, Tom Chou, *How heterogeneous thymic output and homeostatic proliferation shape naive T cell receptor clone abundance distributions*, **12**, 735135, Frontiers in Immunology, (2022).

Mingtao Xia, Lucas Böttcher, Tom Chou, *Controlling epidemics through optimal allocation of test kits and vaccine doses across networks*, IEEE Transactions on Network Science and Engineering, **9**, 1422-1436, (2022).

Haitong Sun, Youngsub Matthew Shin, Mingtao Xia, Shengxian Ke, Michelle Wan, Le Yuan, Yuming Guo, Alexander T. Archibald, *Spatial resolved surface ozone with urban and rural differentiation during 1990–2019: a space–time Bayesian neural network downscaler*, Environmental Science & Technology , **56**, 7337-7349, (2021).

Mingtao Xia, Sihong Shao, Tom Chou, *A frequency-dependent p-adaptive technique for spectral methods, Journal of Computational Physics*, 446, 110627, (2021).

Mingtao Xia, Tom Chou, *Kinetic theory for structured populations*, Journal of Physics: A, 54, 385601, (2021).

Mingtao Xia, Sihong Shao, Tom Chou, *Efficient scaling and moving techniques for spectral methods in unbounded domains*, SIAM Journal on Scientific Computing, **43**, A3244-A3268 , (2021).

Lucas Böttcher, Mingtao Xia, Tom Chou, *Why case fatality ratios can be misleading: individual- and population-based mortality estimates and factors influencing them*, Physical Biology, **17**, 065003, (2020)

Mingtao Xia, Chris D. Greenman, Tom Chou, *PDE Models of adder mechanisms in cellular proliferation*, SIAM Journal on Applied Mathematics, **80**, 1307-1335, (2020)

CHAPTER 1

Introduction

My dissertation covers three topics: i) modeling different structured populations (Chapters 2, 3, 4, and 9), ii) devising an efficient adaptive spectral method to solve spatiotemporal PDEs in unbounded domains (Chapter 5, 6, 7, and 8), and iii) developing efficient control algorithms to control disease spread in a structured human population (Chapter 10).

Chapter 2 first introduces modeling the structured cellular population whose division and growth rates depend on cellular size and added size from the macroscopic aspect. Stuctured population models have been of wide applications and research interests from both mathematical and biological fields. Among all mathematical models, partial differential equation(PDE) models have been very often applied, and many related mathematical problems in this field are to be explored. In this chapter, we will introduce a PDE model for modeling cellular population proliferation on the macroscopic level, which also inspired later research in modeling cellular population from the microscopic level and devising numerical algorithms to solve those models.

In Chapter 3, following Chapter 2, we derive the full kinetic equations describing the evolution of the probability density distribution for a structured population such as cells distributed according to their ages and sizes. The kinetic equations for such a "sizer-timer" model incorporates both demographic and individual cell growth rate stochasticities. Averages taken over the densities obeying the kinetic equations can be used to generate a second order PDE that incorporates the growth rate stochasticity. On the other hand, marginalizing over the densities yields a modified birth-death process that shows how age and size influence demographic stochasticity. Our kinetic framework in this chapter is thus a more

complete model that subsumes both the deterministic PDE and birth-death master equation representations for structured populations.

In Chapter 4, we will further introduce modeling the structured cellular population from the microscopic aspect, which links individual cellular division with cellular population proliferation across different generations. In this chapter, we formulate a kinetic theory for describing the evolution of cellular population that tracks both an individual cell's internal states and cells in different generations (the number of times a cell has divided). Specifically, noise in the evolution of the cell's internal state as well as randomness in cell's division time are incorporated in our model. Based on the kinetic theory, we shall derive equations that describe the dynamics of macroscopic quantities of interest such as cellular population density or the total .amount of a certain kind of protein or mRNA that could be applied to study various biophysical processes such as how cells regulate their sizes over generations and cell differentiation.

In Chapter 5, in order to solve the numerical difficulty of simulating the structured population in Chapter 2, we devise efficient scaling and moving techniques for spectral methods so that spatiotemporal PDEs in unbounded domains could be efficiently and accurately solved. When using Laguerre and Hermite spectral methods to numerically solve PDEs in unbounded domains, the number of collocation points assigned inside the region of interest is often insufficient, particularly when the region is expanded or translated to safely capture the unknown solution. Simply increasing the number of collocation points cannot ensure a fast convergence to spectral accuracy. We propose a scaling technique and a moving technique to adaptively cluster enough collocation points in a region of interest in order to achieve a fast spectral convergence. Our scaling algorithm employs an indicator in the frequency domain that is used to determine when scaling is needed and informs the tuning of a scaling factor to redistribute collocation points to adapt to the diffusive behavior of the solution. Our moving technique adopts an exterior-error indicator and moves the collocation points to capture the translation. Both frequency and exterior-error indicators are defined using only the numerical solutions. We apply our methods to a number of different models, including diffusive

and moving Fermi-Dirac distributions and nonlinear Dirac solitary waves, and demonstrate recovery of spectral convergence for time-dependent simulations. Performance comparison in solving a linear parabolic problem shows that our frequency scaling algorithm outperforms the existing scaling approaches. Finally, we show our frequency scaling technique is able to track the blowup of average cell sizes in a model for cell proliferation.

Chapter 6 further develops a p-adaptive technique that adaptively adjusts the expansion order of spectral methods. When using spectral methods, a question arises as how to determine the expansion order, especially for time-dependent problems in which emerging oscillations may require adjusting the expansion order. Therefore, we propose a frequency-dependent p-adaptive technique that adaptively adjusts the expansion order based on a frequency indicator. Using this p-adaptive technique, combined with the scaling and moving techniques in Chapter 5, we are able to devise an adaptive spectral method in unbounded domains that can capture and handle diffusion, advection, and oscillations. As an application, we use this adaptive spectral method to numerically solve the Schrödinger equation in the whole domain and successfully capture the solution's oscillatory behavior at infinity.

In Chapter 7, we perform the first numerical analysis of the adaptive spectral method using generalized Hermite functions defined on the whole line. There have been few analyses of numerical methods for unbounded domain problems. Specifically, there is no analysis of adaptive spectral methods to provide insight into how to increase efficiency and accuracy through dynamical adjustment of parameters. Therefore, we investigate how the implementation of the adaptive spectral methods affects numerical results, thereby providing guidelines for the proper tuning of parameters. Also, we further improve performance by extending the adaptive methods to allow bidirectional basis function translation.

Next, Chapter 8 combines physics-informed neural networks with adaptive spectral methods to develop a highly efficient machine-learning-based adaptive spectral method that could solve both forward- and inverse-type problems for spatiotemporal PDEs in unbounded domains. Solving analytically intractable partial differential equations (PDEs) that involve at least one variable defined on an unbounded domain arises in numerous physical appli-

cations. Accurately solving unbounded domain PDEs requires efficient numerical methods that can resolve the dependence of the PDE on the unbounded variable over at least several orders of magnitude. In this chapter, we propose a solution to such problems by combining two classes of numerical methods: (i) adaptive spectral methods and (ii) physics-informed neural networks (PINNs). The numerical approach that we develop takes advantage of the ability of PINNs to easily implement high-order numerical schemes to efficiently solve PDEs and extrapolate numerical solutions at any point in space and time. We then show how recently introduced adaptive techniques for spectral methods can be integrated into PINN-based PDE solvers to obtain numerical solutions of unbounded domain problems that cannot be efficiently approximated by standard PINNs. Through a number of examples, we demonstrate the advantages of the proposed spectrally adapted PINNs in solving PDEs and estimating model parameters from noisy observations in unbounded domains.

In Chapter 9, we introduce modeling spread of disease from an infected-time-based structured PDE model for human populations. Different ways of calculating mortality during epidemics have yielded very different results, particularly during the current COVID-19 pandemic. For example, the "CFR" has been interchangeably called the case fatality ratio, case fatality rate, and case fatality risk, often without standard mathematical definitions. The most commonly used CFR is the *case fatality ratio*, typically constructed using the estimated number of deaths to date divided by the estimated total number of confirmed infected cases to date. How does this CFR relate to an infected individual's probability of death? To explore such issues, we formulate both a survival probability model and an associated infection duration-dependent SIR model to define individual- and population-based estimates of dynamic mortality measures to show that neither of these are directly represented by the case fatality ratio. The key parameters that affect the dynamics of different mortality estimates are the incubation period and the time individuals were infected before confirmation of infection. Using data on the recent SARS-CoV-2 outbreaks, we estimate and compare the different dynamic mortality estimates and highlight their differences. Informed by our modeling, we propose more systematic methods to determine mortality during epi-

demic outbreaks and discuss sensitivity to confounding effects and uncertainties in the data arising from, *e.g.*, undertesting and heterogeneous populations.

Chapter 10 introduces number-of-contact-based ODE models for describing disease spread within the human population that is derived from social networks. Additionally, control algorithms that reduce the infections for this ODE model are developed. Efficient testing and vaccination protocols are critical aspects of epidemic management. To study the optimal allocation of limited testing and vaccination resources in a heterogeneous contact network of interacting susceptible, infected, and recovered individuals, we present a degree-based testing and vaccination model for which we derive optimal policies using control-theoretic methods. Within our framework, we find that optimal intervention policies first target high-degree nodes before shifting to lower-degree nodes in a time-dependent manner. Using such optimal policies, it is possible to delay outbreaks and reduce incidence rates to a greater extent than uniform and reinforcement-learning-based interventions, particularly on certain scale-free networks.

CHAPTER 2

PDE models of adder mechanisms in cellular proliferation

Part of this chapter is modified from the paper that was originally published in *SIAM Journal on Applied Mathematics*, **80**, (2020), pp.1307-1335. It is reproduced here with permission of the publisher. SIAM is not responsible for any errors or omissions in this version of the manuscript or any version derived from it. The Version of Record is available online at [10.1137/19M1246754]

2.1 Constructing PDE models for structured population describing cell population proliferation

How cells regulate and maintain their size, as well as sizes of appendages is a longstanding topic in cell biology. Besides the growth of an individual cell, the size distributions of a population of cells are also a quantity of interest. When considering proliferating cell populations, individual cell growth is interrupted by cell division events that generate smaller daughter cells. The biological mechanisms that control when and how a cell divides is a fundamental topic of research. While complex and involving many processes such as metabolism, gene expression, protein production, DNA replication, chromosome separation (for eukaryotic cells), and fission or cell wall formation [SM73, GJJ96, CSW17, CMG18, DWH17]. These processes are regulated and may involve intricate biochemical signaling.

Despite the complexity of cell growth and the cell cycle, three hypotheses for cell division control have arisen. Cell division is often assumed to be governed by cell age a, cell volume x, or added volume since birth y [VKF93, TBS15, MVG17]. Volume growth of an individual cell can be straightforwardly measured and can be modeled by an effective empirical law such as $\dot{x} = g(a, x, y, t)$. A commonly used approximation that is supported by observations is the exponential growth law $g(x) = \lambda x$ [SMK58].

The division mechanism employed by a type of cell is probably most directly classified by tracking the volumes x, added volumes y, and ages a of all division events. The distribution of the event coordinates in (a, x, y)-space, accumulated over time, may provide data that favors a mechanistic interpretation. For example, if the division events are concentrated within a narrow range of volumes x, one might infer a sizer mechanism. However, comparison among the variabilities of the volumes, added volumes, and ages across all division events is difficult. Moreover, in addition to the intrinsic variability in the mechanism of division, the variability in division sizes and times is sensitive to stochasticity arising in the growth and in the sizes of the new daughter cells. Therefore, it can be difficult to precisely classify the mechanism division.

Much like a general growth law $g(a, x, y, t)$ that can depend on age, size, added size, and time, the three distinct mechanisms of cell division need not be mutually exclusive. The birth rate or probability can be an explicit function of any combination of time since birth (age), size, or volume added since birth. For example, cell division may occur through a cell cycle that is started only after the cell exceeds a certain volume, rendering the division rate a function of both size *and* age.

To model cell size control, stochastic maps that relate daughter cell sizes to mother cell sizes have been developed [KB18, MVG17, LA17]. These models describe how cell sizes evolve with generation and can interpolate between timer, sizer, and adder mechanisms. Kessler and Burov [KB18] assumed stochastic growth which lead to a stochastic map with multiplicative noise. They found that an adder mechanism can admit "blow-up" in which the expected cell sizes can increase without bound with increasing generation. Modi *et al.* [MVG17] assume additive noise and do not find blow-up in an adder model. Stochastic maps of generational cell size do not describe population-level distributions in size or age.

To describe population-level distributions, PDE approaches have been developed. For example, the timer model, in which the cell division rate depends only on age of the cell is described by the well-known McKendrick equation for $n(a, t)$ the expected density of cells with age a at time t [Foe59, GC16, CG16]. The McKendrick "transport" equation for the cell density takes the form $\partial_t n(a,t) + \partial_a n(a,t) = -(\mu(a) + \beta(a))n(a,t)$ with the boundary condition $n(t,0) = 2\int_0^a \beta(s)n(s,t)\mathrm{d}s$ describing birth of zero-age cells with age-dependent division rate $\beta(s)$. Note that this timer model does not explicitly track cell sizes. PDE models incorporating sizer mechanisms have also been developed [Per08, DPZ09, RHK14]. In these studies, it was shown that depending on the form of the size-dependent birth rate $\beta(x)$, cells can diverge in size x in the absence of death [DG10]. Existence and uniqueness of weak solutions have been proved for certain boundary and initial conditions. These types of models can be partially solved using the method of characteristics but the boundary condition can only be reduced to a Volterra-type integral equation [Per08, CG16].

Apart from the sizer and the timer models, the adder mechanism has been recently shown

to be consistent with E. coli division [SM73, TBS15, VKF93]. The adder model is motivated by an initiator accumulation mechanism distinct from those used to justify sizers or timers [TBS15, CSW17]. Therefore, we formulate a model that assumes a division rate that is a function of both cell volume and added volume. Terms including cell death can be easily be included afterwards.

2.1.1 Adder-sizer model

Here, we introduce adder-sizer PDE models and generalize them to describe recently observed characteristics of population-level bacterial cell division. An adder-sizer model is one that incorporates a cell division rate $\beta(x, y, t)$ and a single-cell growth rate $g(x, y, t)$ that, instead of depending on a cell's age a, are functions of cell size x and a cell's volume *added since birth* y. Such an adder-sizer PDE model can be developed by defining $n(x, y, t)\mathrm{d}x\mathrm{d}y$ as the mean number of cells with size in $[x, x + \mathrm{d}x]$ and added volume in $[y, y + \mathrm{d}y]$. As cells have finite size and their added volume must be less than total size, $n(x \leq 0, y, t) = n(x, y \geq x, t) = 0$. A derivation similar to that given in [MD86] for the sizer model yields a transport equation of the form

$$\frac{\partial n(x,y,t)}{\partial t} + \frac{\partial [g(x,y,t)n(x,y,t)]}{\partial x} + \frac{\partial [g(x,y,t)n(x,y,t)]}{\partial y} = -\beta(x,y,t)n(x,y,t) \qquad (2.1.1)$$

for the adder-sizer PDE. Here, we have neglected the effects of death, which can be simply added to the right-hand-side of Eq. (2.1.1).

To explicitly outline our general derivation, consider the total population flux into and out of the size and added size domain Ω shown in Fig. 2.1(a) and define $\tilde{\beta}(x', y', z', t)\mathrm{d}z'$ as the rate of fission of cells of size x' and added size y' to divide into two cells, one with size in $[z', z' + \mathrm{d}z']$ and the other with size within $[x' - z', x' - (z' + \mathrm{d}z')]$. For binary fission, the conservation of daughter cell volumes requires $\tilde{\beta}(x', y', z', t) \equiv \tilde{\beta}(x', y', x' - z', t)$. This differential division function allows mother cells to divide into two daughter cells of

differing sizes (asymmetric division), a process that has been observed in numerous contexts [HH92, GJJ96, BK04]. We also assume that daughter cells must have positive sizes so $\tilde{\beta}(x', y', z' = 0, t) = \tilde{\beta}(x', y', z' = x', t) = 0$.

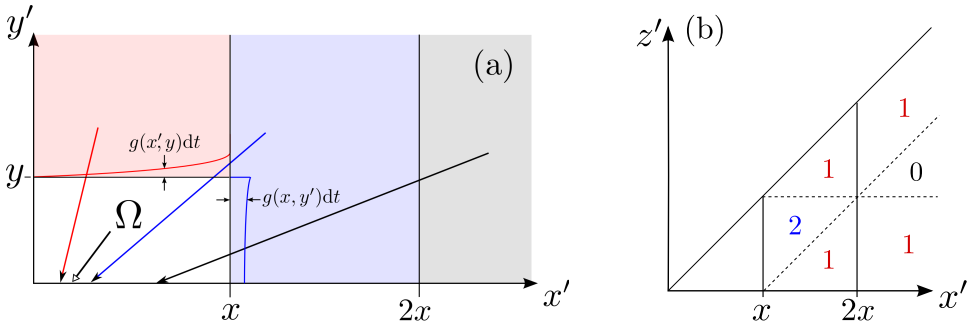

Figure 2.1: The size and added-size state space for cell populations. The expected total number of cells at time t with added size within $[0, y]$ and volume (or "size") within $[0, x]$ is defined as $N(x, y, t)$. Over an increment in time $\mathrm{d}t$, the domain $\Omega = [0, y] \times [0, x]$ infinitesimally distorts $\Omega \to \Omega + \mathrm{d}\Omega$ through the growth increment $g\mathrm{d}t$. The total population within this distorted domain changes only due to birth and death. Cells within Ω that divide always give rise to two daughters within Ω, leading to a net change of $+1$ cell. (b) The z' and x' domains of the differential birth rate function $\tilde{\beta}(x', y', z', t)$. Cells outside of Ω can contribute a net $+1$ or $+2$ cells in Ω depending on the division patterns defined in the depicted regions.

The change in the number of cells in Ω due to fission can arise in a number of ways. First, if a cell in Ω divides, it can only produce two cells with size less than x. Thus, such fission events lead to a net change of $+1$ in the number of cells with $y = 0$ and size in $[0, x]$. If a cell with size within $[0, x]$ but with added size $> y$ divides, it creates two cells with added size $y = 0$ and size within $[0, x]$, leading to a net change of $+2$ cells.

For cells with any added size $y' > 0$ but with size $x' > x$, we have two subcases. If the dividing cell has size $x < x' < 2x$, it will produce one daughter cell in Ω if a daughter cell has size $0 < z' < x' - x$ or $x < z' < x'$ as shown in Fig. 2.1(b). If $x' - x < z' < x$, both daughter cells have size $< x$. Finally, if the dividing cell has size $x' > 2x$, at most one daughter will have size $x' < x$ (see Fig. 2.1(b)). Upon simplifying the above birth terms by using $\int_0^{x'} \mathrm{d}z' = \int_0^x \mathrm{d}z' + \int_x^{x'} \mathrm{d}z'$ for $x' > x$ and the symmetry $\tilde{\beta}(x', y', z', t) = \tilde{\beta}(x', y', x' - z', t)$, we combine terms to balance proliferation with transport and find

$$\int_0^x \mathrm{d}x' \int_0^y \mathrm{d}y' \, \frac{\partial n(x',y',t)}{\partial t} + \int_0^x \mathrm{d}x' \, g(x',y,t) n(x',y,t) + \int_0^y \mathrm{d}y' \, g(x,y',t) n(x,y',t)$$
$$= \int_0^\infty \mathrm{d}y' \int_0^x \mathrm{d}x' \int_0^{x'} \mathrm{d}z' \, \tilde{\beta}(x',y'z',t) n(x',y't)$$
$$+ \int_y^\infty \mathrm{d}y' \int_0^x \mathrm{d}x' \int_0^{x'} \mathrm{d}z' \, \tilde{\beta}(x',y,z',t) n(x',y',t)$$
$$+ 2 \int_0^\infty \mathrm{d}y' \int_x^\infty \mathrm{d}x' \int_0^x \mathrm{d}z' \, \tilde{\beta}(x',y',z',t) n(x',y',t). \tag{2.1.2}$$

Upon taking the derivatives $\frac{\partial^2}{\partial x \partial y}$, we find the PDE given in Eq. (2.1.1) where the total division rate is defined by $\beta(x,y,t) \coloneqq \int_0^x \tilde{\beta}(x,y,z,t)\mathrm{d}z$. For the boundary condition at $y = 0$, we take the derivative $\partial/\partial x$ and set $y \to 0^+$ to find

$$g(x, y=0, t) n(x, y=0, t) = 2 \int_x^\infty \mathrm{d}x' \int_0^{x'} \mathrm{d}y' \, \tilde{\beta}(x', y', z=x, t) n(x', y', t). \tag{2.1.3}$$

The other boundary condition defined by construction is $n(x, x, t) = 0$.

In the special restricted case of symmetric cell division, $\tilde{\beta}(x,y,z,t) = \beta(x,y,t)\delta(z - x/2)$, and boundary condition of the adder-sizer model reduces to

$$g(x, y=0, t) n(x, y=0, t) = 4 \int_0^{2x} \beta(2x, y', t) n(2x, y', t) \mathrm{d}y'. \tag{2.1.4}$$

The above derivation provides an explicit boundary condition representing newly born cells that may be asymmetric in birth size. Quantities such as the total cell population $N(t)$ and the mean total biomass $M(t)$ (the total volume over all cells) can be easily constructed from the density $n(x,y,t)$:

$$N(t) = \int_0^\infty \mathrm{d}x \int_0^x \mathrm{d}y \, n(x,y,t), \quad M(t) = \int_0^\infty \mathrm{d}x \int_0^x \mathrm{d}y \, x n(x,y,t). \tag{2.1.5}$$

Higher moments of the total volume can also be analogously defined. By applying these operations to Eq. (2.1.1) and using the boundary condition (Eq. (2.1.3)), we find the dynamics of the total population and biomass

$$\frac{\mathrm{d}N(t)}{\mathrm{d}t} = \int_0^\infty \mathrm{d}x \int_0^x \mathrm{d}y\, \beta(x,y,t) n(x,y,t), \ \frac{\mathrm{d}M(t)}{\mathrm{d}t} = \int_0^\infty \mathrm{d}x \int_0^x \mathrm{d}y\, g(x,y,t) n(x,y,t). \quad (2.1.6)$$

Finally, we also define the distribution of division events over the size and added size variables, accumulated over a time T:

$$\rho_{\mathrm{d}}(x,y,T) = \frac{\displaystyle\int_0^T \beta(x,y,t) n(x,y,t) \mathrm{d}t}{\displaystyle\int_0^T \mathrm{d}t \int_0^\infty \mathrm{d}x' \int_0^x \mathrm{d}y'\, \beta(x',y',t) n(x',y',t)}. \quad (2.1.7)$$

2.1.2 Sizer-timer model

A PDE model of cell division that combines both size- and age-control division mechanisms can be formulated by defining $n(a,x,t)\mathrm{d}a\mathrm{d}x$ as the expected number of cells at time t with size in $[x, x+\mathrm{d}x]$ and age $[a, a+\mathrm{d}a]$. The PDE can be derived in the same way as introduced in Subsection 2.1.1 which is

$$\begin{gathered} \frac{\partial n}{\partial t}(a,x,t) + \frac{\partial n}{\partial a}(a,x,t) + \frac{\partial (gn)}{\partial x}(a,x,t) = -(\beta n)(a,x,t), \\ n(a,0,t) = 0, \quad n(0,x,t) = 2\int_x^\infty \mathrm{d}z \int_0^\infty \tilde{\beta}(z,x,a,t) n(a,z,t). \end{gathered} \quad (2.1.8)$$

where g is the growth rate, β is the division rate and $\tilde{\beta}$ is the differential splitting rate describing the rate that cells of age a and size z giving birth to newborn cells of size $x \le z$.

On the other hand, a full kinetic theory can be constructed for the structured cell population model in which we track each individual cell and Eq. (2.1.8) can be derived by studying the mean-field behavior of the cell population. The kinetic theory will also enable

us to consider stochasticity in a cell's growth rate which is often the case in experiments when fluctuations in a cell's growth rate are often observed. Furthermore, we can study the interdependence of cells' growth rates on each other.

2.1.3 Division probability and splitting rate

In general, the birth rate functions $\tilde{\beta}(x, y, z, t)$ and $\beta(x, y, t)$ associated with adder-sizer models can take many forms that make biological sense. However, some classes of $\beta(x, y, t)$ may allow the adder-sizer model to be transformed into the well-known "sizer-timer" structured population model [SS67]. To illustrate the relationship, we consider a division rate function β which depends explicitly only on age a and see how it could be converted to a function of size and added size.

For a cell born at time t_0, the probability that the cell splits within time $[a, a + \mathrm{d}a]$ is defined by $\gamma(a; \bar{a})\mathrm{d}a$. In the absence of death, to ensure that any single cell will eventually split, $\int_0^\infty \gamma(a; \bar{a})\mathrm{d}a = 1$. Reasonable choices for $\gamma(a; \bar{a})$ are Gamma, lognormal, or normal distributions. Without loss of generality, we propose a simple gamma distribution for $\gamma(a; \bar{a})$:

$$\gamma(a; \bar{a}) = \frac{1}{a\Gamma((\bar{a}/\sigma_a)^2)} \exp\left[-\frac{a\bar{a}}{\sigma_a^2} + \left(\frac{\bar{a}}{\sigma_a}\right)^2 \ln\left(\frac{a\bar{a}}{\sigma_a^2}\right)\right], \tag{2.1.9}$$

where $\bar{a}$ is the mean division age and σ_a^2 is the variance. This type of distribution can be derived from the sum of independent, exponentially distributed ages.

For deterministic exponential growth $g = \lambda x$, age a and the parameter $\bar{a}$ can be explicitly expressed in terms of x, y and possibly other fixed parameters:

$$a(x, y) = \frac{1}{\lambda} \ln\left(\frac{x}{x - y}\right), \quad \bar{a}(x, y) = \frac{1}{\lambda} \ln\left(\frac{x - y + \Delta}{x - y}\right), \tag{2.1.10}$$

in which Δ is the fixed added size parameter that represents the adder mechanism.

With $a(x, y)$ and $\bar{a}(x, y)$ defined in Eqs. (2.1.10), the division rate function $\beta(x, y)$ can be expressed in terms of x and y by using the splitting probability $\gamma(a(x, y); \bar{a}(x, y))$:

$$\beta(x,y,t) = \frac{\gamma(a(x,y);\bar{a}(x,y))}{1-\int_0^{a(x,y)} \mathrm{d}a'\gamma(a';\bar{a}(x,y))}. \tag{2.1.11}$$

Assuming this "hazard function" form of a growth law, cells born at small initial size $x(0) = x_0 = x - y$ take longer time to divide, while cells born with large size split sooner. Using the gamma distribution, we find a division rate of the form

$$\beta(x,y) = \frac{\Gamma\left(\frac{\bar{a}^2(x,y)}{\sigma_a^2}\right)\gamma(a(x,y);\bar{a}(x,y))}{\Gamma\left(\frac{\bar{a}^2(x,y)}{\sigma_a^2}, \frac{a(x,y)\bar{a}(x,y)}{\sigma_a^2}\right)}, \tag{2.1.12}$$

where $\Gamma(\cdot,\cdot)$ is the upper incomplete gamma function. We plot two examples of the time-independent rate $\beta(x,y)$ in Fig. 2.2.

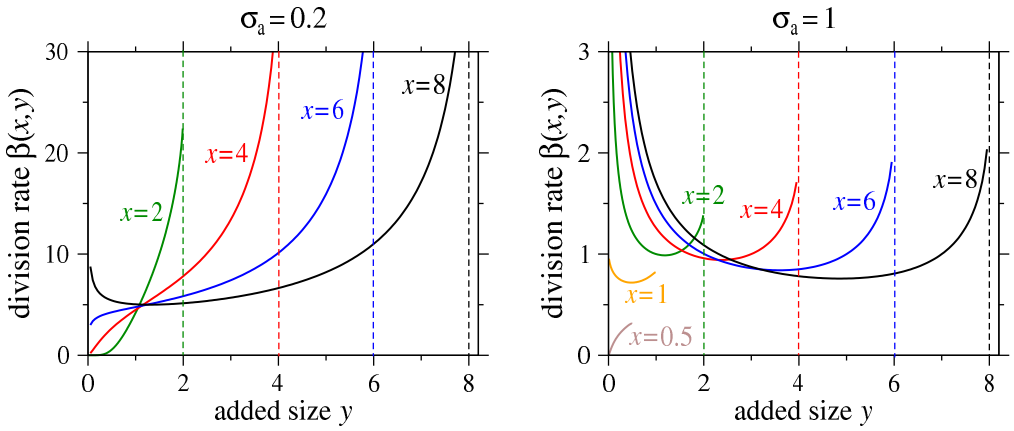

Figure 2.2: The size and added-size dependent rate $\beta(x,y)$ constructed using a gamma distribution for the splitting probability γ (Eq. (2.1.9)) and Eq. (2.1.11). We show projections at fixed values of x. In (a) the parameters are $\sigma_a = 0.2$, while in (b) $\sigma_a = 1$. Note the difference in scale and that $\gamma(a)$ with a higher standard deviation leads to a lower overall cell division rate β. When x is large, $\bar{a}$ defined in Eq. (2.1.10) is small, a nonzero division rate $\beta(x, y \to 0) > 0$ arises indicating that large newborn cells divide quickly to control size across the population. This particular feature arises from our construction of β as a hazard function. Modifying birth rate at small values of y so that $\beta(x, y = 0) \to 0$ will not qualitatively change the predicted densities as long as the birth rate peak persists at small y.

With $\beta(x,y,t)$ defined, we still need to construct the full fission rate $\tilde{\beta}$, which we will assume is a product of the overall division rate $\beta(x,y,t)$ and a differential division probability.

The simplest model is to assume that the differential division probability $h(r)$ is a function of only the ratio r between the size of the daughter cell and that of the mother cell, and independent of the cell size just before division. Thus,

$$\tilde{\beta}(x, y, z, t) = \beta(x, y, t)h(z/x)/x, \tag{2.1.13}$$

where $r \equiv z/x \in [0, 1]$. The boundary condition (Eq. (2.1.3)) can thus be written in the form

$$g(x, 0, t)n(x, 0, t) = 2\int_x^\infty \mathrm{d}x' \int_0^1 \mathrm{d}s\, \beta(x', sx', t)h(x/x')n(x', sx', t). \tag{2.1.14}$$

A reasonable model for $h(r = x/x')$ is a lognormal form that is symmetric about $r = 1/2$:

$$h(r) = \frac{h_0(r) + h_0(1 - r)}{Z(\sigma_r, \delta)}, \quad h_0(r) = e^{-\frac{(-\delta+\ln r)^2}{2\sigma_r^2}} e^{-\frac{\ln^2(1-r)}{2\sigma_r^2}}, \tag{2.1.15}$$

where the parameters δ and σ_r determine the bias and spread of the daughter cell size distribution, and the normalization constant is $Z(\sigma_r, \delta) = \int_0^1 (h_0(r) + h_0(1 - r))\mathrm{d}r$.

2.1.4 Numerical implementation and simulations

With the differential birth rate function $\tilde{\beta}$ defined, we can now consider the implementation of numerical solutions to Eqs. (2.1.1) and (2.1.3) as well as event-based simulations of the underlying corresponding stochastic process.

The numerical approximation to the weak solution will be based on an upwind finite difference scheme in which both x and y are discretized with step size h. We define locally averaged functions by

$$f_{i+\frac{1}{2},j+\frac{1}{2}} \coloneqq \frac{1}{h^2}\int_{ih}^{(i+1)h} \mathrm{d}x \int_{jh}^{(j+1)h} \mathrm{d}y\, f(x, y, t), \tag{2.1.16}$$

where $f(x, y, t)$ can represent $n(x, y, t)$, $g(x, y, t)$, or $\beta(x, y, t)$. Similarly,

$$\tilde{\beta}_{i+\frac{1}{2},j+\frac{1}{2}}((s+\frac{1}{2})h,t) = h^{-3}\int_{ih}^{(i+1)h} \mathrm{d}x \int_{jh}^{(j+1)h} \mathrm{d}y \int_{kh}^{(k+1)h} \mathrm{d}z\, \tilde{\beta}(x,y,z,t) \tag{2.1.17}$$

in the domain $i,j \geq 0$ and $j,k < i$. The discretization of the transport equation can be expressed as

$$\begin{aligned} &\frac{n_{i+\frac{1}{2},j+\frac{1}{2}}(t+\Delta t)-n_{i+\frac{1}{2},j+\frac{1}{2}}(t)}{\Delta t} + \frac{g_{i+1,j+\frac{1}{2}}\tilde{n}_{i+1,j+\frac{1}{2}} - g_{i,j}\tilde{n}_{i,j+\frac{1}{2}}}{h} + \frac{g_{i+\frac{1}{2},j+1}\tilde{n}_{i+\frac{1}{2},j+1} - g_{i+\frac{1}{2},j}\tilde{n}_{i+\frac{1}{2},j}}{h} \\ &\qquad = -\beta_{i+\frac{1}{2},j+\frac{1}{2}} n_{i+\frac{1}{2},j+\frac{1}{2}}(t), \end{aligned} \tag{2.1.18}$$

for $1 \leq i,j \leq L$, where Lh is the maximum size which we take sufficiently large such that $n_{i,j>K}(t=0) = 0, n_{i\leq j} = 0$. We also set $g_{i+\frac{1}{2},i} = 0$ to prevent density flux out of the $y < x$ domain. In Eq. (2.1.18), $g_{i+1,j+\frac{1}{2}}(t)$ can be taken as $g((i+1)h,(j+\frac{1}{2})h,t)$ while $\tilde{n}_{i+1,j+\frac{1}{2}}(t) = \int_{jh}^{(j+1)h}\mathrm{d}y\, n((i+\frac{1}{2})h,y,t)$ is a finite-volume numerical approximation to $\int_{jh}^{(j+1)h}\mathrm{d}y\, n((i+1)h,y,t)$. The discretized version of the boundary condition (Eq. (2.1.3)) can be expressed as

$$g_{i+\frac{1}{2},0} n_{i+\frac{1}{2},0}(t) = 2h^2 \sum_{k=i+1}^{L}\sum_{j=0}^{k-1} \tilde{\beta}_{k+\frac{1}{2},j+\frac{1}{2}}((i+\frac{1}{2})h,t) n_{k+\frac{1}{2},j+\frac{1}{2}}(t). \tag{2.1.19}$$

Direct Monte-Carlo simulations of the birth process are also performed and compared with our numerically computed deterministic distributions. We construct a list of cells and their associated sizes and their sizes at birth. This list is updated at every time step Δt. The cell sizes grow according to $g(x,y,t)$. If a cell divides, the initial sizes of the daughter cells are randomly chosen according to the distribution $h(z/x)$. The daughter cells then replace the mother cell in the list. Simulations of the underlying stochastic process results in, at any given time, a collection of cells, each with a specific size and added size. This collection of cells represents a realization of the population that should be approximated by the distributions that are solutions to Eqs. (2.1.1) and (2.1.3).

2.2 Results and discussion

In this section, we numerically investigate the adder-sizer model and plot various cell population densities and birth event distributions under different parameter regimes. We also show the consistency of numerical solutions of the adder-sizer PDE with results from direct Monte-Carlo simulations of the corresponding stochastic process, which demonstrates that numerical solutions of the linear PDE model for cell the population is in agreement with single-cell level stochastic models. After investigating birth rate parameters that can lead to the blow-up of population-averaged cell sizes, we extend the basic adder model to include mother-daughter growth rate correlations and processes that measure the added size from different points in the cell cycle, *i.e.*, an initation-adder model.

2.2.1 Cell and division event densities

We evaluated our adder-sizer PDE model by using the division rate given in Eq. (2.1.11) and first assuming the simple and well-accepted growth function $g(x, y, t) = \lambda x$. Fig. 2.3 shows the numerical results for the density $\bar{n}(x, y, t) = n(x, y, t)/N(t)$ at successive times $t = 1, 4, 12$, respectively.

Stochastic simulations of the underlying process yield cell populations consistent with the deterministic densities derived from the PDE model. In Fig. 2.4, we compare the cell densities $\bar{n}(x, y, t)$ the division event densities $\rho_{\mathrm{d}}(x, y, T)$ for two different differential division functions $h(r)$. As before, the more asymmetric the division the broader the cell and event densities.

2.2.2 Cell volume explosion

At the single-cell level, a stochastic map model by Kessler and Burov assumed a multiplicative noise and predicted that cell sizes can eventually grow without bound, in agreement with what was experimentally observed for filamentous bacteria [KB18]. However, stochastic maps of generational cell size do not capture population-level distributions in size or age.

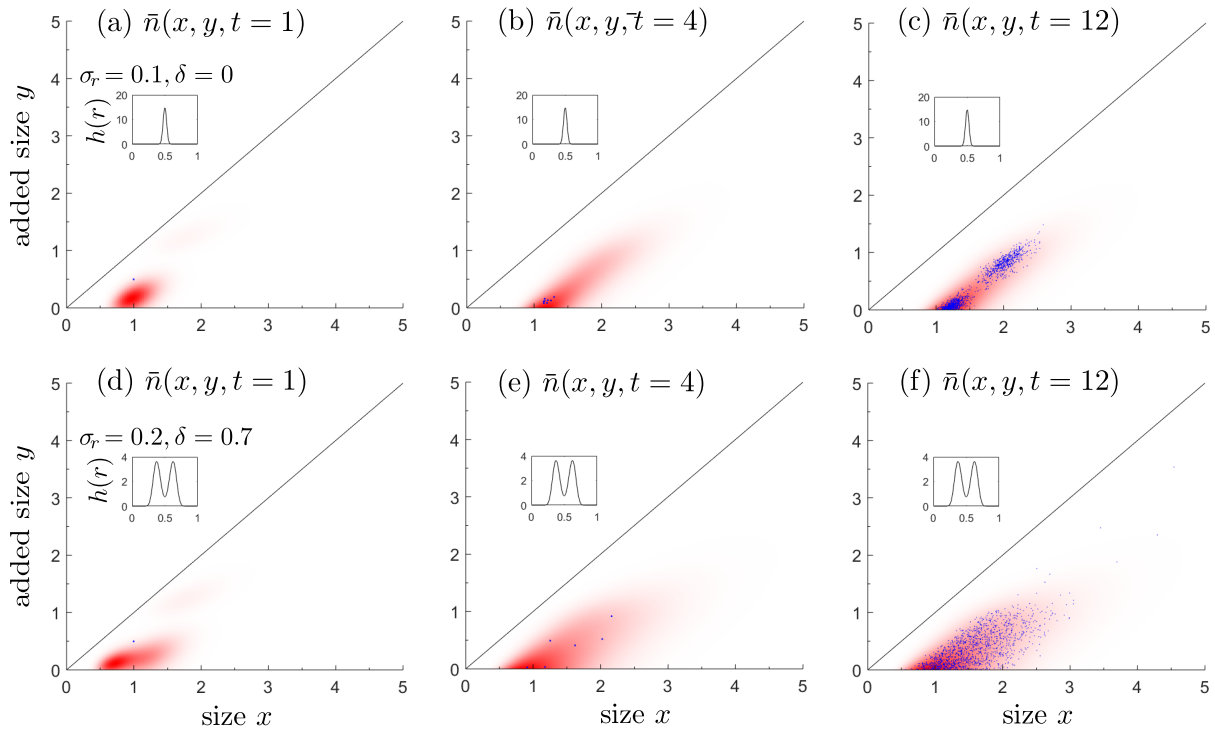

Figure 2.3: Numerically computed densities $\bar{n}(x, y, t) = n(x, y, t)/N(t)$ using $g(x, y, t) = \lambda x$ and $\tilde{\beta}(x, y, z, t)$ defined by Eqs. (2.1.11), (2.1.9), and (2.1.15). For all plots, we use $\sigma_a = 0.1$ in $\gamma(a)$ (Eq. (2.1.9)) and rescale size in units of Δ. In (a-c), we use the sharp, single-peaked differential division function $h(r)$ shown in the inset ($\sigma_r = 0.1, \delta = 0$) and plot $\bar{n}(x, y, 1), \bar{n}(x, y, 4)$, and $\bar{n}(x, y, 12)$, respectively. In (d-f), we plot the densities using a broad (in fact, double-peaked) differential division function $h(r)$ with parameters $\sigma_r = 0.2, \delta = 0.7$. In all calculations, we assumed an initial condition corresponding to a single newly born ($y = 0$) cell with size $x = 1$. For more asymmetric cell division in (d-f), the density spreads faster. In these cases, the densities closely approach a steady-state distribution by about $t = 12$. Also shown in each plot are realizations of Monte-Carlo simulations of the discrete process. Individual cells are represented by blue dots which accurately sample the normalized continuous densities $\bar{n}(x, y, t)$.

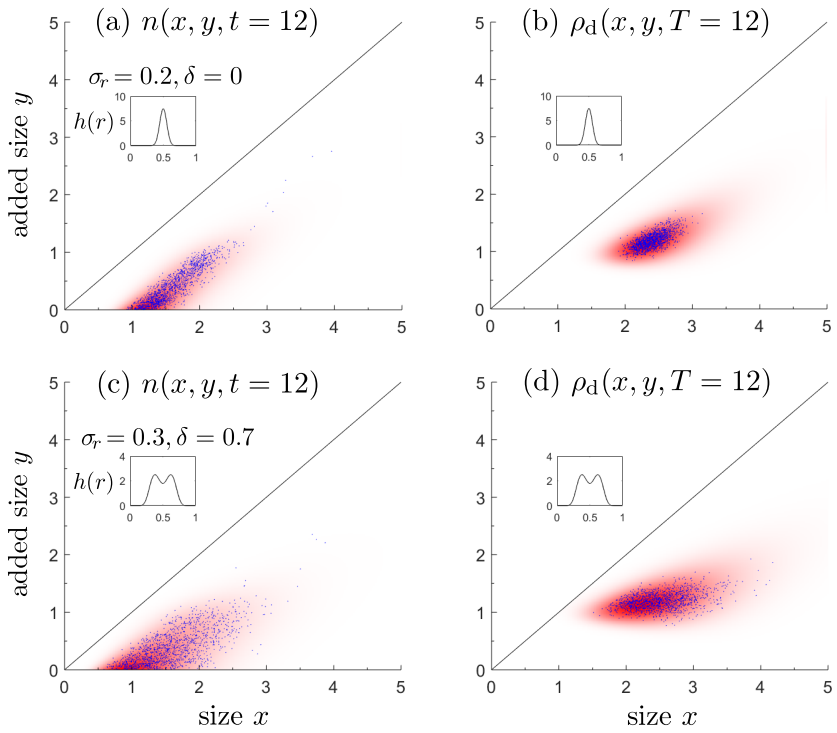

Figure 2.4: Comparison of cell densities $\bar{n}(x, y, t)$ and cell division event densities $\rho_{\mathrm{d}}(x, y, T)$ (Eq. (2.1.7)). The standard deviation $\sigma_a = 0.1$ is used in all calculations. In (a) and (b) we plot $\bar{n}(x, y, t = 12)$ and $\rho_{\mathrm{d}}(x, y, T)$ using $\sigma_r = 0.2, \delta = 0$ while in (c) and (d) we used a broader differential division function in which $\sigma_r = 0.3, \delta = 0.7$. Realizations from Monte-Carlo simulations are overlayed. In (b) and (d), divisions are accumulated up to time $T = 12$.

In this subsection, we will numerically explore how a possible "blowup" in the population-averaged cell volumes. Within PDE models that describe population distributions, timer and sizer mechanisms have been shown to exhibit blow-up depending on the properties of the birth rate $\beta(a, x)$ [BDG19, DHK15, DG10]. Analysis of the conditions on full differential division rate $\tilde{\beta}(x, y, z, t)$ that would result in blowup in the "adder-sizer" PDE model is more involved. Here, we provide only a heuristic argument for sufficient conditions for blowup.

First, we characterize the shape of the densities in the adder-sizer model. In the analogous McKendrick equation [Ian95] one can investigate the age profile defined by dividing the number density by the total population size. The long-term age profile may be stable even when the total population size continuously increases. We take a similar approach here by analyzing $\bar{n}(x, y, t) = n(x, y, t)/N(t)$ where $N(t)$ is given by Eq. (2.1.5). Writing the adder-sizer PDE in terms of $\bar{n}$, we find

$$\frac{\partial \bar{n}}{\partial t} + \frac{\bar{n}}{N}\frac{\mathrm{d}N}{\mathrm{d}t} + \frac{\partial (g\bar{n})}{\partial x} + \frac{\partial (g\bar{n})}{\partial y} = -\beta \bar{n}. \tag{2.2.1}$$

Integrating this equation over x, y leads to $\dot{N}/N = \int_0^\infty \mathrm{d}x \int_0^x \mathrm{d}y\, \beta \bar{n}$, which can be substituted into the first term in Eq. (2.2.1) to yield the nonlinear PDE

$$\frac{\partial \bar{n}}{\partial t} + \frac{\partial (g\bar{n})}{\partial x} + \frac{\partial (g\bar{n})}{\partial y} = -\left(\beta + \int_\Omega \beta \bar{n}\right)\bar{n}. \tag{2.2.2}$$

A number of standard approaches may be applied to analyze Eq. (2.2.2). For example, in [Ian95], solutions are attempted by controlling the analogous non-linear integral term. In the adder-sizer problem, we can define $\langle \beta(t) \rangle = \int_\Omega \beta \bar{n}$ in the above expression to find a self-consistent condition on $\langle \beta(t) \rangle$. One can also assess the steady-state $\bar{n}_{ss}$ by setting $\frac{\partial \bar{n}_{ss}}{\partial t} = 0$ and establishing convergence.

One indication of blow-up is a diverging mean cell size $\langle x(t) \rangle = M(t)/N(t)$. By multiplying the Eq. (2.2.1) by x and integrating (using the boundary condition and symmetry of

the $\tilde{\beta}$ distribution) we find

$$\frac{\mathrm{d}\langle x(t)\rangle}{\mathrm{d}t} + \langle\beta(t)\rangle\langle x(t)\rangle = q(t), \tag{2.2.3}$$

in which $q(t) \coloneqq \int_\Omega g\bar{n}$. If β, g, and $\bar{n} = \bar{n}_{ss}$ are time-independent and a steady state mean cell size exists, we expect it to obey $\langle x(\infty)\rangle = q(\infty)/\langle\beta(\infty)\rangle$. For the special case of deterministic exponential growth $g(x) = \lambda x$, we can write the time evolution of the mean size as

$$\frac{\mathrm{d}\langle x(t)\rangle}{\mathrm{d}t} = \left[\lambda - \langle\beta(t)\rangle\right]\langle x(t)\rangle, \quad \langle\beta(t)\rangle \equiv \int_0^\infty \mathrm{d}x \int_0^x \mathrm{d}y\, \beta(x,y,t)\bar{n}(x,y,t). \tag{2.2.4}$$

If $\beta(\infty)$ is bounded above by λ, then we expect a blowup. For $\beta(\infty)$ that is not bounded, as in our example (Eq. (2.1.11)), one cannot determine if a blowup occurs without a more detailed and difficult analysis. Since the precise conditions on β leading to cell volume explosion are difficult to find, we will explore these possible phenomena using numerical experiments. We numerically examine the density $n(x,y,t\to\infty)$ and the mean cell size $\langle x(t)\rangle$ using the $\beta, \tilde{\beta}$ defined in Eqs. (2.1.11), (2.1.9), and (2.1.15).

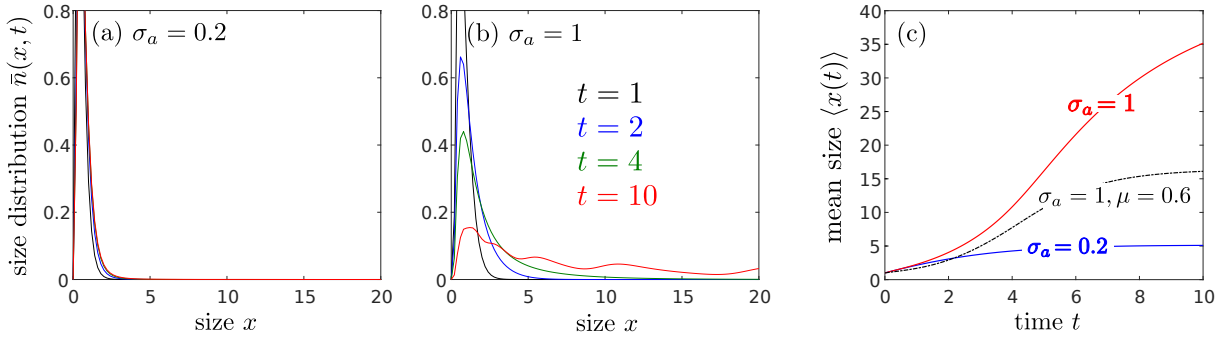

Figure 2.5: (a) Size distributions $\bar{n}(x,t)$ for $\sigma_a = 0.2$ at times $t = 1, 2, 4, 10$. (b) $\bar{n}(x, t = 1, 2, 4, 10)$ for $\sigma_a = 1$, $\sigma_r = 0.1$, and $\delta = 0$. (c) The corresponding mean cell sizes $\langle x(t)\rangle$. The curve associated with the $\sigma_a = 0.2$ saturates while the one corresponding to $\sigma_a = 1$ exhibits blow-up. However, the blowup is suppressed if a death term ($\mu = \ln 2$) is included.

In Figs. 2.5 (a) and (b) we plot the marginal distribution $\bar{n}(x,t) \coloneqq \frac{\int_x^\infty \mathrm{d}y\, n(x,y,t)}{\int_0^\infty \mathrm{d}x \int_x^\infty \mathrm{d}y\, n(x,y,t)} s$ for different values of the division rate variability σ_a at different times. The associated division rates correspond to those plotted in Figs. 2.2(a) and (b). In Fig. 2.5(c) we plot the mean cell sizes $\langle x(t)\rangle = M(t)/N(t)$ corresponding to the distributions in (a) and (b). For sufficiently broad division probabilities $\gamma(a)$ (large σ_a), the division rates β are small, and $\langle x(t)\rangle$ fails to saturate and diverges.

With the presence of the possible "blowup" phenomenon in which the average volume $\langle x(t)\rangle \to \infty$ as t goes to infinity, normal finite volume difference method will be less reliable at long time. The reason is that we really need to investigate the numerical solution's behavior for $x \in (0,\infty)$. Therefore, the finite volume method can only stay valid for some finite time, as it truncates the domain. Spectral methods, however, provide a possible way to track the long-time blowup behavior, as the Laguerre function basis is defined in $(0,\infty)$ and no domain truncation is needed. Yet proper scaling is required to maintain accuracy. In Chapter 5, we develop an adaptive spectral method in unbounded domains that can perform scaling and moving for the basis functions which can successfully capture the diffusive and translative behavior of the solution.

2.2.3 Mother-daughter growth rate correlation

Recent experiments indicate that the growth rate of a mother cell is "remembered" by its daughter cells. For growth rates of the form $g(x,y,t) = \lambda x$, the exponential growth parameter λ between successive generations $i, i+1$ have been proposed to evolve [LA17, DHK15]. In [LA17], fluctuations in λ have been discussed at the single-cell level to explore their effects on the population-averaged growth rate while in [DHK15], changes in growth rates across two consecutive generations are modeled as a Markov process in order to estimate a division rate function β. In this subsection, we first introduce a generalized adder-sizer PDE incorporating variability in λ and then explore the mother-daughter growth rate correlation affects the population dynamics.

A mother-daughter growth rate correlation between two consecutive generations can be described by

$$\lambda_{i+1} = (\lambda_i - \bar{\lambda})R + \bar{\lambda} + \xi, \tag{2.2.5}$$

where ξ is a random variable, $0 \leq R < 1$ is the successive-generation growth rate correlation, and $\bar{\lambda}$ is the mean long-term, or preferred growth rate. Given a growth rate λ_i of a mother cell, Eq. (2.2.5) describes the predicted growth rate λ_{i+1} of its daughter cells. We assume that the random variable has a mean zero and is distributed according to some probability density $P(\xi)$, which vanishes for $\xi \leq (1-R)\bar{\lambda}$ to ensure that the growth rates remain positive.

To incorporate the memory of growth rates between successive generations in the adder-sizer PDE model, we extend the cell density in the growth rate variable λ. Thus, $n(x, y, t, \lambda)$ is the density of cells with volume x, added volume y, *and* growth rate λ. The growth function $g(x, y, t, \lambda)$ is now explicitly a function of the growth rate λ. We propose the extended PDE model

$$\begin{cases} \dfrac{\partial n(x,y,t,\lambda)}{\partial t} + \dfrac{\partial (gn)}{\partial x} + \dfrac{\partial (gn)}{\partial y} = -\beta(x,y,t)n(x,y,t,\lambda), \\ g(x,0,t,\lambda)n(x,0,t,\lambda) = 2\displaystyle\int_0^\infty \mathrm{d}\lambda' \int_x^\infty \mathrm{d}x' \int_0^{x'} \mathrm{d}y \, (\tilde{\beta}(x',y,x,t)n(x',y,t,\lambda') \cdot \\ \qquad\qquad P(\xi = \lambda - R\lambda' - (1-R)\bar{\lambda}), \\ \tilde{\beta}(x,y,x',t) = \tilde{\beta}(x,y,x-x',t)), \\ n(x,y,0,\lambda) = n_0(x,y,\lambda), \end{cases} \tag{2.2.6}$$

A possible symmetric mean zero distribution that vanishes at $-(1-R)\bar{\lambda}$ takes on a log-normal form:

$$P(\xi) \propto \exp\left[-\frac{\ln^2(\xi + (1-R)\bar{\lambda})}{2\sigma_\xi^2} - \frac{\ln^2((1-R)\bar{\lambda} - \xi)}{2\sigma_\xi^2} \right]. \tag{2.2.7}$$

If we start with one newly born daughter cell at size x_0 and growth rate λ_0, the initial condition in our PDE model would be $n_0(x, y, \lambda) = \delta(x - x_0)\delta(y)\delta(\lambda - \lambda_0)$.

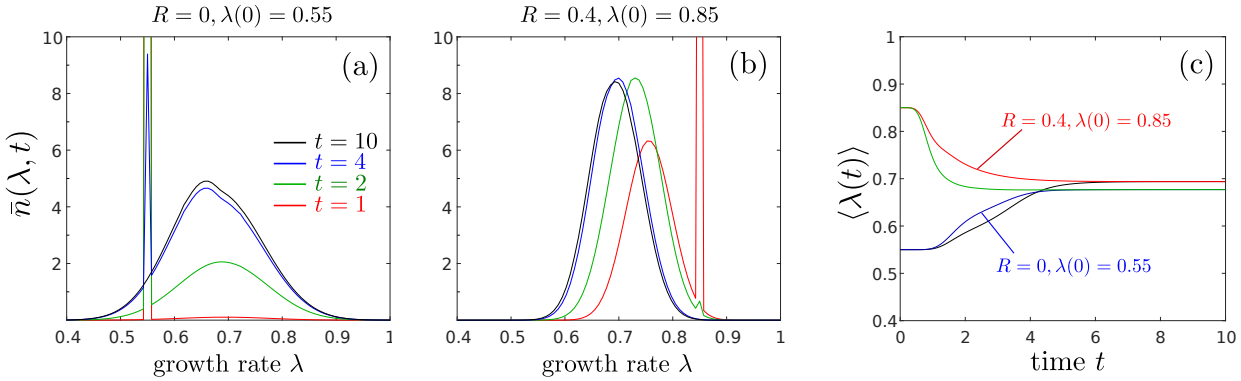

Figure 2.6: Population-level evolution of cellular growth rate. Parameters used are $\bar{\lambda} = \ln 2, \sigma_a = 0.2, \sigma_r = 0.1, \delta = 0$. (a-b) The marginalized density $\bar{n}(\lambda, t)$ as a function of growth rate λ for no correlation ($R = 0$) and initial growth rate $\lambda = 0.55$. The peak in the distribution broadens as the mean evolves toward the preferred mean value $\bar{\lambda} = \ln 2$. (c) The evolution of the mean $\langle \lambda(t) \rangle$ for different values of correlation R. Note that the steady-state values $\langle \lambda(\infty) \rangle$ depend on the correlation R.

Numerical solutions of Eqs. (2.2.6) shown in Fig. 2.6 indicate that although $\bar{\lambda}$ is the same for two different cases, $R = 0$ and $R = 0.4$, their corresponding mean growth rates $\langle \lambda(t) \rangle$ converge to different values. For larger correlation R, the daughter cells' growth rates do not deviate much from those of their mothers' growth rates. This means that the offspring of faster-growing cells tend to grow faster and the offspring of slower-growing cells tend to grow slower. Because it takes a shorter time for faster cells to divide, they will produce more generations of faster-growing cells, leading to a larger average growth rate defined as

$$\langle \lambda(t) \rangle = \frac{\int_0^\infty \mathrm{d}x \int_0^x \mathrm{d}y \int_0^\infty \mathrm{d}\lambda \ \lambda n(t, x, y, \lambda)}{\int_0^\infty \mathrm{d}x \int_0^x \mathrm{d}y \int_0^\infty \mathrm{d}\lambda n(t, x, y, \lambda)}. \tag{2.2.8}$$

On the other hand, for a fixed mother growth rate λ_i, smaller correlations R lead to mean daughter cell growth rates $\langle \lambda_{i+1} \rangle$ that are closer to $\bar{\lambda}$. Since cells with growth rates less than $\bar{\lambda}$ will live longer before division, these cells persist in the population longer than those with larger λ, pushing the average growth rate $\langle \lambda(t) \rangle$ to values smaller than $\bar{\lambda}$. Fig. 2.6(c)

explicitly shows that when $R = 0$, the mean growth rate approaches a value smaller than $\bar{\lambda} = \ln 2$.

2.2.4 Initiation-adder model

Recent experiments suggest a new type of adder mechanism for bacterial cell size control [SLS19]. Rather than a fixed volume added between birth and division as the primary control parameter, new experimental evidence suggests that the control parameter in *E. coli* is the added volume between successive initiations of DNA replication. Initiation occurs when the *ori* sites in a cell's genome are separated, leading to DNA replication and segregation. The number of *ori* sites depend on cell type and species, typically one in prokaryotic cells and more than one in eukaryotic cells. The initiation-adder model assumes that a cell's volume per initiation site (the *ori* site in the genome) tends to add a fixed volume between two consecutive initiations.

If the number of *ori* sites in a cell is q, initiation increases the number to $2q$. Immediately after division and DNA separation, the number of *ori*s decreases back to q in each daughter cell.

In this subsection, we generalize the adder PDE model to describe this new initiation-adder mechanism. We classify all cells into two subpopulations: cells that have not yet undergone initiation and cells that have initiated DNA replication but that have not yet divided. We define $n_1(x, y, t)\mathrm{d}x\mathrm{d}y$ as the expected number of pre-initiation cells in with volume in $[x, x + \mathrm{d}x]$ and with added volume $y < x$ in $[y, y + \mathrm{d}y]$. Mean post-initiation cell numbers with volume in $[x, x + \mathrm{d}x]$ and added volume in $[y, y + \mathrm{d}y]$ are described by $n_2(x, y, t)\mathrm{d}x\mathrm{d}y$. In the general initiation-adder process, when a pre-initiation cell commences DNA replication (initiates) can depend on the volume or added volume. Thus, we describe transitions from a pre-initiation cell transitions into a post-initiation cell by the rate $k_{\mathrm{i}}(x, y, t)$. After initiation, the number of *ori* sites doubles and the added volume is reset to zero in the newly formed post-initiation cell. In analogy with the differential division rate in Eq. (2.1.1), we

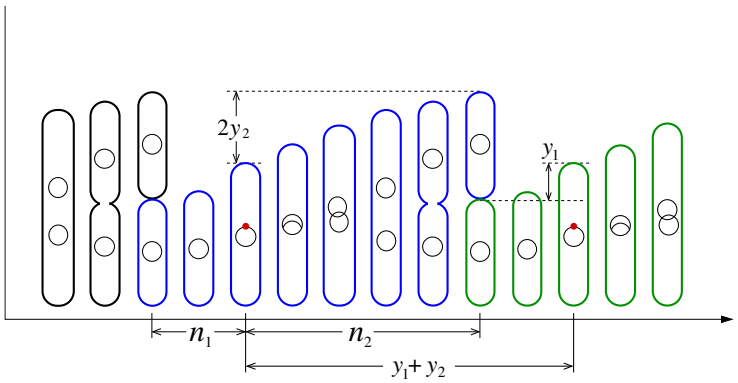

Figure 2.7: Schematic for the initiation adder process. DNA replication is initiated (indicated by the red dot) before copied DNA is segregated and cell division. In this example, $q = 1$ and y_2 is the added volume per origination site for two origination sites. The density of cells with $q = 1$ copy of DNA (before DNA replication initiation) is denoted $n_1(x, y, t)$ while the density of cells post-initiation is denoted $n_2(x, y, t)$, where y denotes the volume added after initiation. The factor that controls $y_1 + y_2$ in the initiation-adder model is the volume Δ added between successive initiation events, rather than between successive cell divisions. Thus, the controlled variable (added volume in this case) spans the pre-initiation and post-initiation states.

define $\beta(x, y, t)$ as the rate of division of post-initiation cells. Under a general asymmetric division event, we assume that the added volume is divided proportionally to the volume of the daughter cells, *i.e.*, if the mother cell's volume is x with added volume y since initiation, and if one daughter cell's volume is $z < x$ and the other daughter cell's volume is $x - z$, the added volume since the division for the first daughter will be set to yz/x while the added volume for the second daughter will be $y(x - z)/x$. The resulting PDE model now involves two coupled densities n_1 and n_2:

$$\frac{\partial n_1(x,y,t)}{\partial t} + \frac{\partial [g_1 n_1]}{\partial x} + \frac{\partial [g_1 n_1]}{\partial y} = -k_{\rm i}(x,y,t)n_1 + 2\int_x^\infty \frac{z}{x} n_2(z, yz/x, t)\tilde{\beta}(z, x, yz/x, t)\mathrm{d}z,$$

$$\frac{\partial n_2(x,y,t)}{\partial t} + \frac{\partial [g_2 n_2]}{\partial x} + \frac{\partial [g_2 n_2]}{\partial y} = -\beta(x,y,t)n_2,$$

$$n_1(x,0,t) = 0, \quad g_2 n_2(x,0,t) = \int_0^x k_{\rm i}(x,y,t)n_1(x,y,t)\mathrm{d}y, \tag{2.2.9}$$

$$\beta(x,y,t) = \int_0^x \tilde{\beta}(x,z,y,t)\mathrm{d}z, \tag{2.2.10}$$

in which we have allowed for different growth rates in the different cell phases. Both n_1 and n_2 are defined in the domain $\{\mathbb{R}^{+2} \cap \{y < x\}\} \times \mathbb{R}^+$. These coupled PDEs are different from the PDE associated with the standard "division-adder" described in Eqs. (2.1.1) and (2.1.3). Here, the added volume is reset to zero not after division, but after initiation.

In [WFL16], a strong size control acting on initiation initiation was proposed where all cells will have initiated DNA replication before reaching some fixed volume x_i. This hypothesis can be implemented in our initiation-adder model by setting $k_\mathrm{i}(x \to x_\mathrm{i}, y, t) \to \infty$. The probability that a cell born at time t_0 has not yet initiated, $e^{-\int_{t_0}^{t} k_\mathrm{i}(x(s),y(s),s)\mathrm{d}s}$, always vanishes for all (t_0, x_{t_0}, y_{t_0}) before some finite time t and $x(t) < x_\mathrm{i}$. Thus, $n_2(x, 0, t)$ is nonzero only in $[0, x_\mathrm{i}]$ for all t. If there exists a constant τ_0 such that $\lim_{\tau \to \tau_0} e^{-\int_{t_0}^{t_0+\tau} k_\mathrm{i}(x(s),y(s),s)\mathrm{d}s} = 0$ for all t_0, then the largest volume that any cell can attain will be $e^{\lambda\tau_0} x_\mathrm{i}$, leading to strict size control and no blowup.

Fig. 2.8 shows numerical solutions to Eq. (2.2.10) using the same birth rate function as that used in Fig. 2.3(d-f). Note that due to cell size control affecting the pre-initiation stage, initial daughter cell sizes stay small at initiation and $n_1(x, y, t)$ is more peaked near $y \approx x$.

If one takes k_i sufficiently large, both daughter cells will nearly instantly initiate DNA replication after division. We have checked numerically that for constant $k_\mathrm{i} = 10^3$, that the densities $n_1(x, y, t)$ are negligible while $n_2(x, y, t)$ approaches the density of the division adder shown in Fig. 2.3 (for the same differential division functions $\tilde{\beta}$). Thus, the initiation adder model converges to the standard division adder model when $k_\mathrm{i} \to \infty$. This can be seen from the first of Eqs. (2.2.10) where n_1 can be neglected and is dominated by the two terms on the right-hand side. Substituting $k_\mathrm{i}(x, y, t)n_1 \approx 2\int_x^\infty \frac{\mathrm{d}z}{x} n_2(z, yz/x, t)$ into the integral terms in the second equation, we find Eq. (2.1.1) for $n_2(x, y, t)$.

2.3 Summary and conclusions

In this chapter, we proposed a PDE model that incorporates an adder mechanism in cell division. In the absence of death, we motivated models for the differential birth rate function

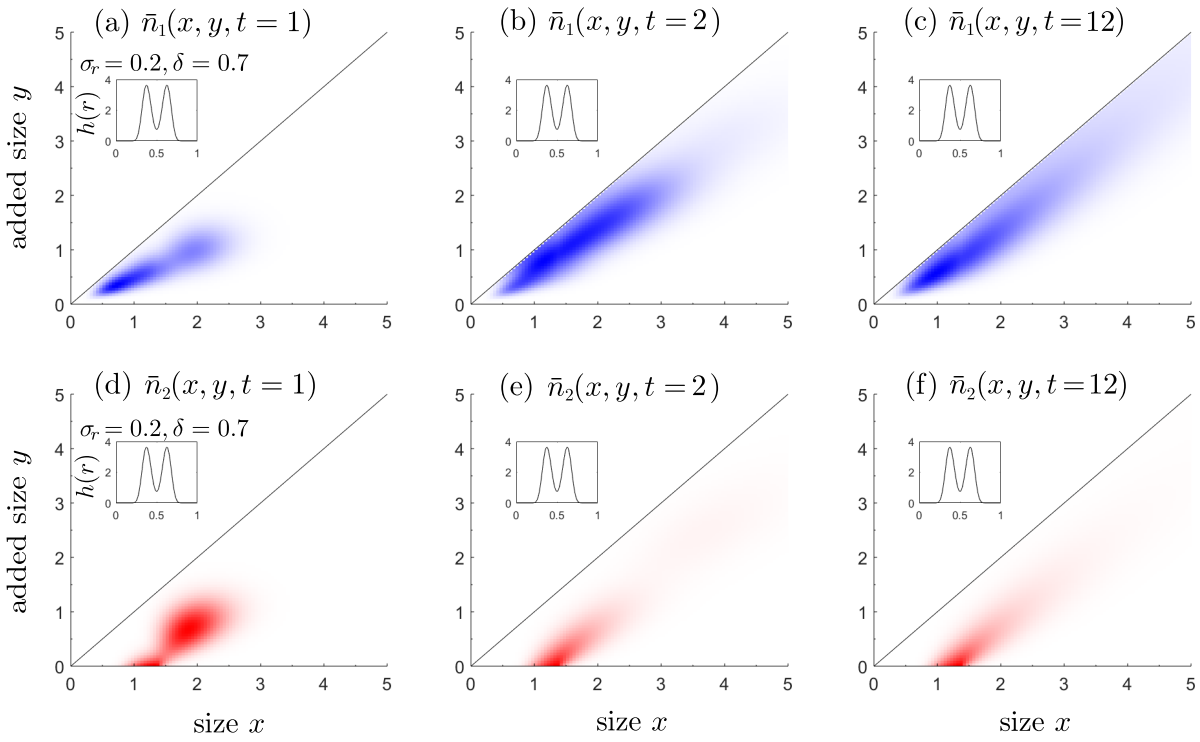

Figure 2.8: Normalized densities of pre-initiation cell populations $\bar{n}_1$ and post-initiation cell populations $\bar{n}_2$ for at various fixed times $t = 1, 2, 12$. Here, we used $k_{\mathrm{i}}(x) = p(x)/\left[1 - \int_0^x p(x')\mathrm{d}x'\right]$ with $p(x) \sim \mathcal{N}(1, 0.1)$ and the same $\tilde{\beta}(x, y, z, t)$ as that used in Fig. 2.3(d-f). (a-c) shows the normalized densities $\bar{n}_1(x, y, t) \equiv n_1(x, y, t)/N(t)$ where $N(t) = \int \mathrm{d}y \int \mathrm{d}x(n_1 + n_2)$. (d-f) shows the normalized post-initiation density $\bar{n}_2(x, y, t)$. For the k_{i} used in this example, the pre-initiation densities span larger volumes and added volumes. The densities are indistinguishable from those at steady state after about $t = 2$.

$\tilde{\beta}(x, y, z, t)$ that are consistent with normalized division probabilities when cell death is neglected. In Appendix A.1.1 we showed existence and uniqueness of a weak solution to the PDE model within a time interval $[0, T]$ during which the solution's support can be bounded. One can prove similar results when both time and space are unbounded as this problem is related to other first-order PDE models that have been studied in more detail.

With a weak solution justified, we explored the sizer-adder PDE via numerical experiments and Monte-Carlo simulations of the underlying stochastic process. Our results show that event-based Monte-Carlo simulations of discrete cells generate realizations of cell configurations that provide accurate samples of the cell densities computed from our PDE model.

When broader differential division rates are used (when cell division is more asymmetric), we find, under the same initial conditions, a broader cell density $n(x, y, t)$ and a broader event density $\rho_{\mathrm{D}}(x, y, T)$. We also demonstrate numerically, the divergence of the mean cell size $\langle x(t) \rangle = M(t)/N(t)$. We showed those division probabilities that are broader in the age or added size (and smaller in magnitude) more likely lead to mean cell sizes that explode with time. While we could not analytically find the specific conditions that lead to blow-up, we found, in the simple case of exponential cell growth, a simple sufficient bound for the division rate below which cell size explosion occurs.

Finally, we translated a stochastic model of the cell growth rate correlation between cells of successive generations [KB18] in to our sizer-adder PDE model. By extending the dimension of the density function to include growth rates and allowing for variability in growth rate, as new cells are born, we developed a PDE model that incorporated the stochastic nature of growth rate inheritance and that describes the evolution of the growth rate distribution of cells. We found that the steady-state value of the mean growth rate depends on the correlation of growth rates between mother and daughter cells. This dependence arises from a subtle interaction between the shape of the growth rate distribution and the distribution of variations in the growth rate from one generation to the next.

PDE-type models can be used to model cell densities that evolve according to timer, sizer, or adder mechanisms, as well as combinations of mechanisms such as the sizer-timer model

and the sizer-adder model studied here. Under a deterministic cell growth assumption, one might propose a growth rate function $g(a, x, y)$ and birth rate $\beta(a, x, y)$ that depend on all three variables, age a, size x, and added size y. Thus, one might propose a full sizer-timer-adder model of the form $(\partial_t+\partial_a)n+\partial_x(gn)+\partial_y(gn) = -\beta(a, x, y, t)n(a, x, y, t)$ with purported boundary condition $g(0, x, 0, t)n(0, x, 0, t) = 2\int_0^\infty \mathrm{d}a' \int_x^\infty \mathrm{d}x' \int_0^{x'} \mathrm{d}y' \, \tilde{\beta}(a', x', y', x, t)n(a', x', y', t)$. However, the three variables $, a, x, y$ are not all independent. For example, if the deterministic added size $y(t)$ is monotonic in time t, the age after birth a and the added size y are functions of each other. More generally, if we can determine the evolution of all three variables (a, x, y) given two of them, we cannot construct a meaningful 3+1-dimensional PDE model. One can understand the loss of independence by noticing that when a cell divides, both its daughter cells' ages and their added volumes reset to $a = y = 0$. For deterministic growth, the age and added size are bijective and are thus not independent. Subsequent deterministic growth of the daughter cells is described by their sizes and *either* their age a or added size y.

Thus, the age and added size variables are not independent, and given timer, sizer, and adder mechanisms, there are only two structurally different PDE models, the sizer-timer PDE, and the sizer-adder PDE studied in detail here. The sizer-timer model can be reduced to just a sizer model or a timer model (McKendrick equation). Thus, assuming deterministic cell growth, one can consider only the timer, sizer, adder, sizer-timer, or sizer-adder PDE models. An adder-only model can be defined only when both the division growth rates depend only on added size y and not on size x. However, if the growth is itself stochastic, one might propose higher-order models that can include all types of cell division mechanisms.

CHAPTER 3

Kinetic theory for stochastic sizer-timer models cell size control

This is the Accepted Manuscript version of an article accepted for publication in *Journal of Physics A: Mathematical and Theoretical*, **54**, (2021), pp.385601. IOP Publishing Ltd is not responsible for any errors or omissions in this version of the manuscript or any version derived from it. The Version of Record is available online at [10.1088/1478-3975/ab9e59].

3.1 Introduction

Across many diverse applications, mathematical models have been formulated to describe the evolution of populations according to a number of individual attributes such as age, size, and/or added size since birth. For example, deterministic age-structured models that incorporate age-dependent birth and death were developed by McKendrick and have been applied to human populations [Foe59]. More recently, there has been renewed interest in cell size control [TBS15, KB18], cellular division mechanisms [RHK14], and structured cell population models [Per08, MD86].

When considering proliferating cell populations, individual cell growth is interrupted by cell division events that generate smaller daughter cells. Cell division is a process that involves many biochemical steps and complex biophysical mechanisms including metabolism, gene expression, protein production, DNA replication, chromosome separation (for eukaryotic cells), and fission or cell wall formation [SM73, HD81, CSW17, DWH17, Wes94]. To simplify the understanding of which factors trigger cell division, three basic models that subsume these complex processes have been proposed. Cells can divide based on their age since birth, volume (size), or added volume since birth y [TBS15, MVG17]. PDE approaches for the timer, sizer, and adder models, as well as combinations of these models, have been well-studied [XGC20, MD86, BDG19]. These PDE approaches implicitly describe the mean density of cells in age, size, and/or added size, and are considered deterministic models.

However, there has been much less development of structured population models that incorporate stochastic effects. In the presence of stochasticity, how would the PDEs be modified? In the sizer-timer type of structured population models, stochasticity can arise in the growth dynamics of each cell as well as in random times of cell division and death (demographic stochasticity).

Stochasticity arising from random times of birth and death (demographic stochasticity) has been considered in timer-like models for age-structured populations [GC16, CG16]. This approach generalized the classic deterministic McKendrick equation to a higher di-

mension (dynamically varying) associated with the number of individuals in the system. This higher-dimensional stochastic "kinetic theory" allows one to systematically connect an age-independent birth-death master equation description to the deterministic age-structured McKendrick model. A comprehensive and general treatment of the age-structured stochastic process using a Doi-Peliti operator formalism has also been developed for the calculation of correlation functions [Gre17]. The full kinetic theory has only been developed for age-structured populations and only includes demographic stochasticity (since chronological age is a deterministic quantity proportional to time). Other approaches using stochastic hybrid systems [VSS16] have been used to incorporate the influence of random birth times of population-level variations in cell size. Intrinsic stochasticity in the growth rate of an individual cell has been treated in terms of Langevin equations for cell size [HLA18], effective potentials [KB18] and stochastic maps [MVG17, KB17]. Recently, Chapman-Kolmogorov equations have also been applied to study the effect of different sources of noise in cellular proliferation [NVP21]. However, stochasticity in the intrinsic growth rate has not been considered within a demographically stochastic kinetic theory.

In this chapter, we derive the kinetic equations for the sizer-timer model of cell proliferation that incorporates both demographic stochasticity and intrinsic stochasticity in the growth of individual cells. In the next section, we derive the Fokker-Planck equation for the size of an individual cell and define the probabilistic quantities needed to construct the full kinetic theory. This equation is then marginalized in Section 3.3 to explicitly isolate and show the feature limits of intrinsic stochasticity and demographic stochasticity. Including both sources of stochasticity renders the calculations of marginalized densities rather technical, but by defining specific moments, we derive a hierarchy of models describing correlations that arise from growth rate stochasticity. These higher-order (and higher dimensional) models cannot be derived from approaches that impose mean-field assumptions and are evident only when a kinetic approach such as ours is employed. The first-order model describing the single-particle density is self-contained and simply reduces the mean-field "sizer-timer" model [RHK14, XGC20]. Higher-order models are connected to each other and the first-

order mean-field model. Marginalization of higher moments of particle numbers can also be constructed from our kinetic theory. These hierarchical models describe demographic stochasticity and are not closed. Our results generalize a large body of work on sizer-timer PDE models to include stochastic processes, both at the individual and population levels.

3.2 Derivation of kinetic theory

Here, we outline the derivation of the kinetic equation for a the population of dividing cells of different ages a and sizes (volumes) x. We start from the SDE for the size [1] of a single cell at time t:

$$\mathrm{d}X_t = g(X_t, A_t, t)\mathrm{d}t + \sigma(X_t, A_t, t)\mathrm{d}W_t, \quad X_t,\, A_t \in \Lambda, \tag{3.2.1}$$

where $\Lambda \coloneqq [0, \infty)$, A_t is the cell's age (time that has elapsed after its birth), $g(X_t, A_t, t) > 0$ is the size- and age-dependent growth rate, and W_t is a standard Wiener process with independent, normally distributed increments $W_t - W_s$, zero mean, and variance $t - s$. The parameter $\sigma(X_t, A_t, t)$ represents the strength of stochasticity in a cell's growth rate. Here, we assume both g and σ are Lipschitz continuous to ensure the existence and uniqueness of X_t given any initial conditions $X_0 > 0, A_0 \geq 0$. We also assume $\sigma \in \mathbf{C}^1, \sigma(0, t, a) = \partial_x\sigma(0, t, a) = 0$ so that the noise vanishes at $x = 0$ and X_t remains positive.

Next, we investigate a system of $m + 2n$ cells, where m is the number of individual cells (singlets) and n is the number of twins (doublets). A twin means two daughter cells generated from the division of a common mother cell, and therefore they have the identical age. In this section, we use the notation

$$\begin{aligned} \mathbf{X}_t^{(m)} &= (X_t^1, X_t^2, ..., X_t^m),\ \mathbf{Y}_t^{(2n)} = (Y_t^1, ..., Y_t^{2n}), \\ \mathbf{A}_t^{(m)} &= (A_t^1, A_t^2, ..., A_t^m),\ \ \mathbf{B}_t^{(n)} = (B_t^1, ..., B_t^n), \end{aligned} \tag{3.2.2}$$

[1] Alternatively, X_t might also represent the log of the cell size

where $\mathbf{A}_t^{(m)}$ and $\mathbf{B}_t^{(n)}$ are ordered ages such that $A_t^i \geq A_t^j \geq 0, B_t^i \geq B_t^j \geq 0, \forall i > j$ and $\mathbf{X}_t^{(m)}$ and $\mathbf{Y}_t^{(2n)}$ are the vectors of the volumes of the m singlets and $2n$ doublets that are of ages $\mathbf{A}_t^{(m)}$ and $\mathbf{B}_t^{(n)}$, respectively, at time t. We first use ordered ages to facilitate our derivations and to better understand the boundary conditions representing newly born cells. Note that two cells in a doublet have the same age but can have different sizes; thus, the age vector $\mathbf{B}_t^{(n)}$ of the $2n$ twins stores n ages, while the size vector $\mathbf{Y}_t^{(2n)}$ stores $2n$ sizes.

Formally solving Eq. (3.2.1), each X_t^i and Y_t^j satisfies

$$\begin{aligned} X_t^i &= X_{t'}^i + \int_{t'}^t g(X_s^i, A_s^i, s)\mathrm{d}s + \int_{t'}^t \sigma(X_s, A_s, s)\mathrm{d}W_s^i, \\ Y_t^j &= Y_{t'}^j + \int_{t'}^t g(Y_s^j, B_s^{[\frac{j+1}{2}]}, s)\mathrm{d}s + \int_{t'}^t \sigma(Y_s^j, B_s^{[\frac{j+1}{2}]}, s)\mathrm{d}W_s^{m+j}, \end{aligned} \tag{3.2.3}$$

where $\mathrm{d}W_s^i, \mathrm{d}W_s^{m+j}$ are intrinsic, independent fluctuations in growth rates. We assume that cell division rates are regulated by a "timer" mechanism and do not depend on cell size, *i.e.*, the probability that a cell in a population of m singlets and n doublets divides during $(t, t+\Delta t]$ is $\beta_{m,n}(A_t, t)\mathrm{d}t + o(\mathrm{d}t)$, a function of its age A_t, time t and population sizes m, n. The mathematical analysis that follow requires that the birth rate is independent of a cell's size X_t. Finally, we take the continuous-time limit and assume that in a finite number of cells, the possibility of two cells dividing in $(t, t+\mathrm{d}t]$ is $o(\mathrm{d}t)$ as $\mathrm{d}t \to 0$.

3.2.1 The forward equation

We evaluate the increment in time by Ito's formula applied to a function

$$f_{m,n}(\mathbf{X}_t^{(m)}, \mathbf{Y}_t^{(2n)}, t | \mathbf{A}_{t'}^{(m)}, \mathbf{B}_{t'}^{(n)}) \tag{3.2.4}$$

of m individual and n twin sizes given initial sizes and ages $\mathbf{A}_{t'}^{(m)}, \mathbf{B}_{t'}^{(n)}$ at $t' < t$, where the ages are defined to be in the descending order $A^1 \geq A^2 ... \geq A^m \geq 0$, $B^1 \geq B^2 ... \geq B^n \geq 0$. Ordering the ages will eventually allow us to easily incorporate cell division as a boundary condition in which newborn cells are represented by $B^n = 0$. We start by constructing the

difference

$$
\begin{aligned}
&f_{m,n}(\mathbf{X}^{(m)}_{t+\mathrm{d}t}, \mathbf{Y}^{(2n)}_{t+\mathrm{d}t}, t+\mathrm{d}t|\mathbf{A}^{(m)}_{t'}, \mathbf{B}^{(n)}_{t'}) - f_{m,n}(\mathbf{X}^{(m)}_{t}, \mathbf{Y}^{(2n)}_{t}, t|\mathbf{A}^{(m)}_{t'}, \mathbf{B}^{(n)}_{t'}) = \\
&\quad \int_t^{t+\mathrm{d}t} \Bigg[\frac{\partial f_{m,n}}{\partial s} + \sum_{i=1}^{m} g(X^i_s, A^i_s, s) \frac{\partial f_{m,n}}{\partial X^i_s} + \sum_{j=1}^{2n} g(Y^j_s, B^{[(j+1)/2]}_s, s) \frac{\partial f_{m,n}}{\partial Y^j_s} \\
&\qquad + \frac{1}{2} \sum_{i=1}^{m} \sigma^2(X^i_s, A^i_s, s) \frac{\partial^2 f_{m,n}}{(\partial X^i_s)^2} + \frac{1}{2} \sum_{j=1}^{2n} \sigma^2(Y^j_s, B^{[(j+1)/2]}_s, s) \frac{\partial^2 f_{m,n}}{(\partial Y^j_s)^2} \Bigg] \mathrm{d}s \\
&\quad + \sum_{i=1}^{m} \int_t^{t+\mathrm{d}t} \sigma(X^i_s, A^i_s, s) \frac{\partial f_{m,n}}{\partial X^i_s} \mathrm{d}W^i_s + \sum_{j=1}^{2n} \int_t^{t+\mathrm{d}t} \sigma(Y^j_s, B^{[(j+1)/2]}_s, s) \frac{\partial f_{m,n}}{\partial Y^j_s} \mathrm{d}\tilde{W}^j_s.
\end{aligned}
\tag{3.2.5}
$$

After taking the expectation of Eq. (3.2.5) we find

$$
\begin{aligned}
&\mathbb{E}[f_{m,n}(\mathbf{X}^{(m)}_{t+\mathrm{d}t}, \mathbf{Y}^{(2n)}_{t+\mathrm{d}t}, t+\mathrm{d}t|\mathbf{A}^{(m)}_{t'}, \mathbf{B}^{(n)}_{t'})] - \mathbb{E}[f_{m,n}(\mathbf{X}^{(m)}_{t}, \mathbf{Y}^{(2n)}_{t}, t|\mathbf{A}^{(m)}_{t'}, \mathbf{B}^{(n)}_{t'})] = \\
&\quad \mathbb{E}\left[\int_t^{t+\mathrm{d}t} \mathrm{d}s \left(\frac{\partial f_{m,n}}{\partial s} + \sum_{i=1}^{m} g(X^i_s, A^i_s, s) \frac{\partial f_{m,n}}{\partial X^i_s} + \sum_{j=1}^{2n} g(Y^j_s, B^{[(j+1)/2]}_s, s) \frac{\partial f_{m,n}}{\partial Y^j_s} \right.\right. \\
&\qquad \left.\left. + \frac{1}{2} \sum_{i=1}^{m} \frac{\partial^2 f_{m,n}}{(\partial X^i_s)^2} \sigma^2(X^i_s, A^i_s, s) + \frac{1}{2} \sum_{j=1}^{2n} \frac{\partial^2 f_{m,n}}{(\partial Y^j_s)^2} \sigma^2(Y^j_s, B^{[(j+1)/2]}_s, s) \right) \right].
\end{aligned}
\tag{3.2.6}
$$

Specifically, we can take $f_{m,n}$ in Eq. (3.2.6) as a distribution of the form

$$
\begin{aligned}
f_{m,n}(\mathbf{X}^{(m)}_{t}, \mathbf{Y}^{(2n)}_{t}, t|\mathbf{A}^{(m)}_{t'}, \mathbf{B}^{(n)}_{t'}) = &\prod_{i=1}^{m} \delta(X^i - X^i_t) \prod_{j=1}^{2n} \delta(Y^j - Y^j_t) \\
&\times S_{1,m}(t|t', \mathbf{A}^{(m)}_{t'}) S_{2,n}(t|t', \mathbf{B}^{(m)}_{t'}),
\end{aligned}
\tag{3.2.7}
$$

where $S_{1,m}$ and $S_{2,n}$ are joint survival possibilities

$$
\begin{aligned}
S_{1,m}(t|t', \mathbf{A}^{(m)}) &= \prod_{i=1}^{m} e^{-\int_{t'}^{t} \beta_{m,n}(A^i - t' + s, s)\mathrm{d}s}, \\
S_{2,n}(t|t', \mathbf{B}^{(n)}) &= \prod_{j=1}^{n} (e^{-\int_{t'}^{t} \beta_{m,n}(B^j - t' + s, s)\mathrm{d}s})^2,
\end{aligned}
\tag{3.2.8}
$$

and the birth rate $\beta \equiv \beta_{m,n}$ can implicitly depend on the populations m, n.

Next, we define $\hat{p}(\mathbf{X}_t^{(m)}, \mathbf{Y}_t^{(2n)}, t | \mathbf{X}_{t'}^{(m)}, \mathbf{Y}_{t'}^{(2n)}, \mathbf{A}_{t'}^{(m)}, \mathbf{B}_{t'}^{(n)})$ as the probability density of m singlets of volumes $\mathbf{X}_t^{(m)}$ and n doublets of volumes $\mathbf{Y}_t^{(2n)}$ at time t, conditioned on there being m singlets of volumes $\mathbf{X}_{t'}^{(m)}$ and ages $\mathbf{A}_{t'}^{(m)}$ and n doublets with volumes $\mathbf{Y}_{t'}^{(2n)}$ and ages $\mathbf{B}_{t'}^{(2n)}$ at time t', and that no cell division occurs during $[t', t]$. The quantity $\hat{p}(\mathbf{X}_t^{(m)}, \mathbf{Y}_t^{(2n)}, t | \mathbf{X}_{t'}^{(m)}, \mathbf{Y}_{t'}^{(2n)}, \mathbf{A}_{t'}^{(m)}, \mathbf{B}_{t'}^{(n)}) S_{1,m}(t|t', \mathbf{A}_{t'}^{(m)}) S_{2,n}(t|t', \mathbf{B}_{t'}^{(m)})$ is thus the probability measure that the cell population at time t contains m singlets of size $\mathbf{X}_t^{(m)}$ and n doublets of size $\mathbf{Y}_t^{(n)}$ with no cell division occurring within $[t', t]$, conditioned on it containing m singlets with volumes $\mathbf{X}_{t'}^{(m)}$ and ages $\mathbf{A}_{t'}^{(m)}$ and n doublets with volumes $\mathbf{Y}_{t'}^{(2n)}$ and ages $\mathbf{B}_{t'}^{(n)}$ at t'.

After substitution of the $f_{m,n}$ defined in Eq. (3.2.7) into Eq. (3.2.6), dividing by $\mathrm{d}t$, and taking the $\mathrm{d}t \to 0$ limit, we obtain

$$
\begin{aligned}
&\frac{\partial}{\partial t}\left(\hat{p}(\mathbf{X}^{(m)}, \mathbf{Y}^{(2n)}, t | \mathbf{X}_{t'}^{(m)}, \mathbf{Y}_{t'}^{(2n)}, \mathbf{A}_{t'}^{(m)}, \mathbf{B}_{t'}^{(n)}) S_{1,m}(t|t', \mathbf{A}_t^{(m)}) S_{2,n}(t|t', \mathbf{B}_t^{(n)})\right) = \\
&\qquad \int_{\Lambda^m} \mathrm{d}\mathbf{X}_t^{(m)} \int_{\Lambda^{2n}} \mathrm{d}\mathbf{Y}_t^{(2n)}\, \hat{p}(\mathbf{X}_t^{(m)}, \mathbf{Y}_t^{(2n)}, t | \mathbf{X}_{t'}^{(m)}, \mathbf{Y}_{t'}^{(2n)}, \mathbf{A}_{t'}^{(m)}, \mathbf{B}_{t'}^{(n)}) \Bigg[\frac{\partial f}{\partial t} \\
&\qquad\qquad + \sum_{i=1}^{m} g(X_t^i, A_t^i, t) \frac{\partial f}{\partial X_t^i} + \sum_{j=1}^{2n} g(Y_t^j, B_t^{[(j+1)/2]}, t) \frac{\partial f}{\partial Y_t^j} \\
&\qquad\qquad + \frac{1}{2} \sum_{i=1}^{m} \frac{\partial^2 f}{\partial (X_t^i)^2} \sigma^2(X_t^i, A_t^i, t) + \frac{1}{2} \sum_{j=1}^{2n} \frac{\partial^2 f}{\partial (Y_t^j)^2} \sigma^2(Y_t^j, B_t^{[(j+1)/2]}, t) \Bigg] \qquad (3.2.9) \\
&= S_{1,m} S_{2,n} \Bigg[- \left(\sum_{i=1}^{m} \beta_{m,n}(A_t^i, t) + 2 \sum_{j=1}^{n} \beta_{m,n}(B_t^j, t) \right) \hat{p}_{m,n} \\
&\qquad\qquad - \sum_{i=1}^{m} \frac{\partial (g(X_t^i, A_t^i, t) \hat{p})}{\partial X_t^i} - \sum_{j=1}^{2n} \frac{\partial (g(Y_t^j, B_t^{[(j+1)/2]}, t) \hat{p})}{\partial Y_t^j} \\
&\qquad\qquad + \frac{1}{2} \sum_{i=1}^{m} \frac{\partial^2 (\sigma^2(X_t^i, A_t^i, t) \hat{p})}{(\partial X_t^i)^2} + \frac{1}{2} \sum_{j=1}^{2n} \frac{\partial^2 (\sigma^2(Y_t^j, B_t^j, t) \hat{p})}{(\partial Y_t^j)^2} \Bigg]
\end{aligned}
$$

where the last equality arises from integration by parts.

Finally, we derive the PDE satisfied by the unconditioned probability density

$$p_{m,n}(\mathbf{X}_t^{(m)}, \mathbf{Y}_t^{(2n)}, \mathbf{A}_t^{(m)}, \mathbf{B}_t^{(n)}, t) \tag{3.2.10}$$

given $p_{m,n}(\mathbf{X}^{(m)}, \mathbf{Y}^{(2n)}, \mathbf{A}^{(m)}, \mathbf{B}^{(n)}, t')$. First, we note that if no division has occurred in $[t', t]$ and $t - t' < \min\{A_t^{(m)}, B_t^{(n)}\}$, a system at t with m singlets of volumes $\mathbf{X}_t^{(m)}$ and ages $\mathbf{A}_t^{(m)}$ and n doublets with volumes $\mathbf{Y}_t^{(2n)}$ and ages $\mathbf{B}_t^{(n)}$ can result only from a system at t' with m singlets with ages $\mathbf{A}_{t'}^{(m)} = \mathbf{A}_t^{(m)} - (t - t')$ and n doublets with ages $\mathbf{B}_{t'}^{(n)} = \mathbf{B}_t^{(n)} - (t - t')$. Thus, we use the Chapman-Kolmogorov relation between the two quantities $\hat{p}(\mathbf{X}_t^{(m)}, \mathbf{Y}_t^{(2n)}, t | \mathbf{X}_{t'}^{(m)}, \mathbf{Y}_{t'}^{(2n)}, \mathbf{A}_{t'}^{(m)}, \mathbf{B}_{t'}^{(n)}) S_{1,m}(t|t', \mathbf{A}_{t'}^{(m)}) S_{2,n}(t|t', \mathbf{B}_{t'}^{(m)})$ and $p_{m,n}$ to construct

$$\begin{aligned} p_{m,n}(\mathbf{X}_t^{(m)}, \mathbf{Y}_t^{(2n)}, \mathbf{A}_{t'}^{(m)} + t - t', \mathbf{B}_{t'}^{(n)} + t - t', t) = \int_{\Lambda^{(m+2n)}} \hat{p}(\mathbf{X}_t^{(m)}, \mathbf{Y}_t^{(2n)}, t | \mathbf{X}_{t'}^{(m)}, \mathbf{Y}_{t'}^{(2n)}, \mathbf{A}_{t'}^{(m)}, \mathbf{B}_{t'}^{(n)}) \\ \times S_{1,m}(t|t', \mathbf{A}_{t'}^{(m)}) S_{2,n}(t|t', \mathbf{B}_{t'}^{(m)}) p_{m,n}(\mathbf{X}_{t'}^{(m)}, \mathbf{Y}_{t'}^{(2n)}, \mathbf{A}_{t'}^{(m)}, \mathbf{B}_{t'}^{(n)}, t') \mathrm{d}\mathbf{X}_{t'}^{(m)} \mathrm{d}\mathbf{Y}_{t'}^{(2n)}. \end{aligned} \tag{3.2.11}$$

Assuming that $p_{m,n}$ is continuous and differentiable, and the integration is interchangeable with differentiation in Eq. (3.2.11), we take derivatives with respect to all variables t, X^i, Y^j, A^i, B^j to obtain

$$\begin{aligned} \frac{\partial p_{m,n}}{\partial t} + \sum_{i=1}^{m} \frac{\partial p_{m,n}}{\partial A_t^i} + \sum_{j=1}^{n} \frac{\partial p_{m,n}}{\partial B_t^j} + \sum_{i=1}^{m} \frac{\partial (g(X_t^i, A_t^i, t) p_{m,n})}{\partial X_t^i} + \sum_{j=1}^{2n} \frac{\partial (g(Y_t^j, B_t^j, t) p_{m,n})}{\partial Y_t^j} \\ = -\left(\sum_{i=1}^{m} \beta_{m,n}(A_t^i, t) + 2 \sum_{j=1}^{n} \beta_{m,n}(B_t^j, t) \right) p_{m,n} \\ + \frac{1}{2} \sum_{i=1}^{m} \frac{\partial^2 (\sigma^2(X_t^i, A_t^i, t) p_{m,n})}{(\partial X_t^i)^2} + \frac{1}{2} \sum_{j=1}^{2n} \frac{\partial (\sigma^2(Y_t^j, B_t^j, t) p_{m,n})}{(\partial Y_t^j)^2}, \end{aligned} \tag{3.2.12}$$

where $p_{m,n} \equiv p_{m,n}(\mathbf{X}_t^{(m)}, \mathbf{Y}_t^{(2n)}, \mathbf{A}_t^{(m)}, \mathbf{B}_t^{(n)}, t)$. Hereafter, we will omit the subscript t for notational simplicity. To facilitate further analysis, we define a symmetrized density $\rho_{m,n}$ that is symmetric to the interchange of variables:

$$\rho_{m,n}(\mathbf{X}^m, \mathbf{Y}^{2n}, \mathbf{A}^m, \mathbf{B}^n, t) = \frac{1}{2^n m! n!} \sum_{\pi^{2n}} p_{m,n}(\mathbf{X}^{(m)}, \pi^{2n}(\mathbf{Y}^{(2n)}), \mathbf{A}^{(m)}, \mathbf{B}^{(n)}, t) \tag{3.2.13}$$

where $\mathbf{A}^{(m)} = (A^{\xi_a(1)}, \ldots, A^{\xi_a(m)})$, $\mathbf{B}^{(n)} = (B^{\xi_b(1)}, \ldots, B^{\xi_b(m)})$ are ordered ages, $\mathbf{X}^{(m)} = (X^{\xi_a(1)}, \ldots, X^{\xi_a(m)})$, $Y^{(2n)} = (Y^{2\xi_b(1)-1}, \ldots, Y^{2\xi_b(n)})$ are the corresponding sizes, and π^{2n} is some permutation $\Lambda^{2n} \to \Lambda^{2n}$ such that $\pi^{2n}(Y^{2i}), \pi^{2n}(Y^{2i-1}) \in \{Y^{2i-1}, Y^{2i}\}, \pi^{2n}(Y^{2i}) \neq \pi^{2n}(Y^{2i-1}), i = 1, ..., n$, *i.e.*, π^{2n} can interchange the sizes of two cells in a doublet. Therefore, there are 2^n total permutations π^{2n}. $\xi_a(1), ..., \xi_a(m)$ is a rearrangement such that $A^{\xi_a(1)} \geq A^{\xi_a(2)} \geq ... \geq A^{\xi_a(m)}$ and $\xi_b(1), ..., \xi_b(n)$ is a rearrangement such that $B^{\xi_b(1)} \geq B^{\xi_b(2)} \geq ... \geq B^{\xi_b(n)}$. Defining such a $\rho_{m,n}$ allows us to remove the restriction that the ages must be presented in a descending order. Moreover, changing the order of two cells within in a doublet will not affect the value of $\rho_{m,n}$. Definite integrals over $\rho_{m,n}$ are then related to those over $p_{m,n}$ via

$$\begin{aligned}\int \mathrm{d}\mathbf{X}^m \mathrm{d}\mathbf{Y}^{2n} \mathrm{d}\mathbf{A}^m \mathrm{d}\mathbf{B}^n \rho_{m,n}(\mathbf{X}^m, \mathbf{Y}^{2n}, \mathbf{A}^m, \mathbf{B}^n, t) = \int_{\Lambda^{(m+2n)}} \mathrm{d}\mathbf{X}^{(m)} \mathrm{d}\mathbf{Y}^{(2n)} \int_\Lambda \mathrm{d}A^{\xi_a(1)} ... \\ ... \int_0^{A^{\xi_a(m-1)}} \mathrm{d}A^{\xi_a(m)} \int_\Lambda \mathrm{d}B^{\xi_b(1)} ... \int_0^{B^{\xi_b(n-1)}} \mathrm{d}B^{\xi_b(n)} \, p_{m,n}(\mathbf{X}^{(m)}, \mathbf{Y}^{(2n)}, \mathbf{A}^{(m)}, \mathbf{B}^{(n)}, t),\end{aligned} \tag{3.2.14}$$

so $\rho_{m,n}$ is also a probability density distribution if $p_{m,n}$ is. Furthermore, the differential equation satisfied by $\rho_{m,n}$ for $\mathbf{A}^m, \mathbf{B}^n > 0$ is the same as the differential equation satisfied by $p_{m,n}$

$$\frac{\partial \rho_{m,n}}{\partial t} + \sum_{i=1}^{m} \frac{\partial \rho_{m,n}}{\partial A^i} + \sum_{j=1}^{n} \frac{\partial \rho_{m,n}}{\partial B^j} + \sum_{i=1}^{m} \frac{\partial (g(X^i, A^i, t)\rho_{m,n})}{\partial X^i} + \sum_{j=1}^{2n} \frac{\partial (g(Y^j, B^{[\frac{j+1}{2}]}, t)\rho_{m,n})}{\partial Y^j}$$
$$= -\Big(\sum_{i=1}^{m} \beta_{m,n}(A^i, t) + 2\sum_{j=1}^{n} \beta_{m,n}(B^j, t) \Big) \rho_{m,n}$$
$$+ \frac{1}{2} \sum_{i=1}^{m} \frac{\partial^2 (\sigma^2(X^i, A^i, t)\rho_{m,n})}{(\partial X^i)^2} + \frac{1}{2} \sum_{j=1}^{2n} \frac{\partial^2 (\sigma^2(Y^j, B^{[\frac{j+1}{2}]}, t)\rho_{m,n})}{(\partial Y^j)^2}. \tag{3.2.15}$$

3.2.2 Boundary conditions

We now specify appropriate boundary conditions for $\rho_{m,n}$ that represent the birth of new cells with age zero. By using ordered ages, it is easy to derive the corresponding boundary conditions for $p_{m,n}$ defined in Eq. (3.2.11), which we omitted here, but which are nonzero if $B^n = 0$ and zero if any entry in $\mathbf{X}^{(m)}, \mathbf{Y}^{(2n)}, \mathbf{A}^{(m)}, \mathbf{B}^{(k<n)}$ is zero. The boundary conditions for $\rho_{m,n}$ are then derived from the boundary conditions for $p_{m,n}$. Homogeneous boundary conditions also arise at any $X^i = 0, \infty$ or $Y^j = 0, \infty$ indicating that no cell can have 0 or infinite size. If one cell divides at time t in a system of m singlets and n doublets, the system could either convert to $m-1$ singlets and $n+1$ doublets when this dividing cell is a singlet, or $m+1$ singlets and n doublets when the dividing cell is one cell in a doublet. A simpler but similar discussion of boundary conditions for the "timer" model which has no size dependence has been discussed [GC16, CG16]. Hereafter, we use the notation $\mathbf{X}^m_{-i} = (X^1, X^2, ..., X^{i-1}, X^{i+1}, ..., X^m)$, $\mathbf{A}^m_{-i} = (A^1, A^2, ..., A^{i-1}, A^{i+1}, ..., A^m)$ to describe vectors of one lower dimension in which element i is removed. The boundary conditions are given by

$$\rho_{m,n} = 0 \begin{cases} \text{if any element in } \{\mathbf{X}^m, \mathbf{Y}^{2n}\} = 0, \infty, \\ \text{or any element in } \mathbf{A}^m = 0, \\ \text{or more than one element in } \mathbf{B}^n = 0, \end{cases} \tag{3.2.16}$$

and

$$\rho_{m,n}(\mathbf{X}^m, \mathbf{Y}^{2n}[Y^{2j-1} = y_1, Y^{2j} = y_2], \mathbf{A}^m, \mathbf{B}^n[B^j = 0], t) =$$
$$\frac{m+1}{n}\int_0^\infty \mathrm{d}s\, \tilde{\beta}_{m+1,n-1}(y_1+y_2, y_1, s, t)\rho_{m+1,n-1}(\mathbf{X}^{m+1}[X^{m+1} = y_1+y_2],$$
$$\mathbf{Y}^{n-1}, \mathbf{A}^{m+1}[A^{m+1} = s], \mathbf{B}^{n-1}, t) \qquad (3.2.17)$$
$$+\frac{2}{m}\sum_{i=1}^{m}\tilde{\beta}_{m-1,n}(y_1+y_2, y_1, A^i, t)\rho_{m-1,n}(\mathbf{X}^m_{-i}, \mathbf{A}^m_{-i}, \mathbf{B}^n[B^n = A^i],$$
$$\mathbf{Y}^{2n}[Y^{2n-1} = X^i, Y^{2n} = y_1+y_2], t),$$

Equation (3.2.16) enforces that no cell can have a zero or infinitely large size and no more than one cell can divide at the same instant (continuous time assumption). In Eq. (3.2.17), the notation $\mathbf{X}^{m+1}[X^i = x]$ indicates that the i^{th} component in $\mathbf{X}^{m+1}$ is x, with similar definitions for $\mathbf{Y}^{2n}[Y^j = y], \mathbf{A}^m[A^i = a], \mathbf{B}^n[B^j = b]$. The first term on the RHS of Eq. (3.2.17) results from the division of a singlet while the second term results from the division of one cell in a doublet, leaving a singlet and giving rise to a new doublet. Division is described by $\tilde{\beta}_{m,n}(x, z, a, t)\mathrm{d}z$, the rate that in a population of m singlets and n doublets, a cell of volume x and age a divides into one cell with volume $\in [z, z+\mathrm{d}z]$. By allowing $\tilde{\beta}_{m,n}$ to explicitly depend on both the mother cell's size x and the daughter cell's size z, we can readily allow for asymmetric division and daughter cells of different sizes. Moreover, from volume conservation, we impose $\tilde{\beta}_{m,n}(x, z, a, t) = \tilde{\beta}_{m,n}(x, x-z, a, t)$. Finally, if we assume the simple form $\tilde{\beta}_{m,n}(x, z, a, t) = h(z/x)\beta_{m,n}(a, t)/x$ [XGC20], $\int_0^x \tilde{\beta}_{m,n}(x, z, a, t)\mathrm{d}z = \beta_{m,n}(a, t)$ is independent of size x as we have assumed. In the Appendix, we explicitly demonstrate that probability conservation is preserved under these boundary conditions.

3.3 Hierarchies and moment equations

In this section, we will assume that $\tilde{\beta}$ and β are independent of the population sizes m, n. Under this assumption, we are able to derive lower-dimensional (*e.g.*, marginalized) projections of our kinetic theory (Eq. (3.2.15)) by integrating over a specific number of cell

sizes:

$$\rho_{m,n}^{(h,k,\ell)}(\mathbf{X}^h;\, \mathbf{Y}_{\mathrm{e}}^{2k+2\ell};\, \mathbf{A}^h,\, \mathbf{B}^{k+\ell};\, t) = \int_{\Lambda} \mathrm{d}\mathbf{X}^{h+1:m} \mathrm{d}\mathbf{Y}_{\mathrm{o}}{}^{2k+2\ell+1:2n} \mathrm{d}\mathbf{A}^{h+1:m} \mathrm{d}\mathbf{B}^{k+\ell+1:n}\, \rho_{m,n}, \qquad (3.3.1)$$

where $\rho_{m,n} \equiv \rho_{m,n}(\mathbf{X}^m;\, \mathbf{Y}^{2n};\, \mathbf{A}^m;\, \mathbf{B}^n;\, t)$, $\Lambda \equiv \Lambda^{(m-h)+(2n-k-2\ell)+(m-h)+(n-k)}$, and we define the notation $\mathbf{X}^{h+1:m} \coloneqq (X^{h+1}, ..., X^m)$, $\mathbf{Y}_{\mathrm{o}}^{2k+2\ell+1:2n} \coloneqq (Y^1, Y^3, ..., Y^{2k-1}, Y^{2k+2\ell+1}, ..., Y^{2n})$, $\mathbf{A}^{h+1:m} \coloneqq (A^{h+1}, ..., A^m)$, $\mathbf{B}^{k+\ell+1:n} \coloneqq (B^{k+\ell+1}, ..., B^n)$ and $\mathbf{Y}_{\mathrm{e}}^{2k+2\ell} \coloneqq (Y^2, Y^4, ..., Y^{2k}$, $Y^{2k+1}, Y^{2k+2}, ..., Y^{2k+2\ell})$. The marginalized densities require three indices to describe because although the size $\mathbf{X}^m$ and age $\mathbf{A}^m$ have a one-to-one correspondence for singlets, the twins, while carrying the same age, almost surely have different sizes due to asymmetric division and independent growth fluctuations immediately after birth. Thus, the number of ways to exit and enter each state depends on which types of cells are "integrated over". By marginalizing over Eq. (3.2.15), we find the kinetic equation satisfied by $\rho_{m,n}^{(h,k,\ell)}$ (in the remaining space $\mathbf{X}^h, \mathbf{Y}_{\mathrm{e}}^{2k+2\ell}, \mathbf{A}^h$, $\mathbf{B}^k > 0$) becomes

$$
\begin{aligned}
&\frac{\partial \rho_{m,n}^{(h,k,\ell)}(\mathbf{X}^h;\, \mathbf{Y}_{\mathrm{e}}^{2k+2l};\, \mathbf{A}^h;\, \mathbf{B}^{k+\ell};\, t)}{\partial t} + \sum_{i=1}^{h} \frac{\partial \rho_{m,n}^{(h,k,\ell)}}{\partial A^i} + \sum_{j=1}^{k+\ell} \frac{\partial \rho_{m,n}^{(h,k,\ell)}}{\partial B^j} + \sum_{i=1}^{h} \frac{\partial (g(X^i, A^i, t)\rho_{m,n}^{(h,k,\ell)})}{\partial X^i} \\
&\quad + \sum_{j=1}^{k} \frac{\partial (g(Y^{2j}, A^j, t)\rho_{m,n}^{(h,k,\ell)})}{\partial Y^{2j}} + \sum_{j=1}^{2\ell} \frac{\partial (g(Y^{2k+j}; A^j, t)\rho_{m,n}^{(h,k,\ell)})}{\partial Y^{2k+j}} \\
&\quad - \frac{1}{2}\sum_{i=1}^{h} \frac{\partial^2 (\sigma^2(X^i;\, A^i;\, t)\rho_{m,n}^{(h,k,\ell)})}{(\partial X^i)^2} - \frac{1}{2}\sum_{j=1}^{k} \frac{\partial^2 (\sigma^2(Y^{2j};\, B^{[\frac{j+1}{2}]}, t)\rho_{m,n}^{(h,k,\ell)})}{(\partial Y^{2j})^2} \\
&\quad - \frac{1}{2}\sum_{j=1}^{2\ell} \frac{\partial^2 (\sigma^2(Y^{2k+j};\, B^{k+[\frac{j+1}{2}]}, t)\rho_{m,n}^{(h,k,\ell)})}{(\partial Y^{2k+j})^2} \\
&= -\sum_{i=1}^{h} \beta(A^i, t)\rho_{m,n}^{(h,k,\ell)}(\mathbf{X}^h;\, \mathbf{Y}_{\mathrm{e}}^{2k+2\ell};\, \mathbf{A}^h;\, \mathbf{B}^{k+\ell};\, t) \\
&\quad - \sum_{j=1}^{k+\ell} 2\beta(B^j, t)\rho_{m,n}^{(h,k,\ell)}(\mathbf{X}^h;\, \mathbf{Y}_{\mathrm{e}}^{2k+2\ell};\, \mathbf{A}^h;\, \mathbf{B}^{k+\ell};\, t) \\
&\quad - (m-h)\int_{\Lambda^2} \mathrm{d}X^{h+1}\mathrm{d}A^{h+1}\, \beta(A^{h+1};\, t)\rho_{m,n}^{(h+1,k,\ell)}(\mathbf{X}^{h+1};\, \mathbf{Y}_{\mathrm{e}}^{2k+2\ell};\, \mathbf{A}^{h+1};\, \mathbf{B}^{k+\ell};\, t) \\
&\quad - 2(n-k-\ell)\int_{\Lambda^2} \mathrm{d}Y^{2k+2}\mathrm{d}B^{k+1}\, \beta(B^{k+1};\, t)\rho_{m,n}^{(h,k+1,\ell)}(\mathbf{X}^h;\, \mathbf{Y}_{\mathrm{e}}^{2k+2\ell+2};\, \mathbf{A}^h;\, \mathbf{B}^{k+\ell+1};\, t) \\
&\quad + \frac{(n-k-\ell)(m+1)}{n}\int_{\Lambda^2} \mathrm{d}X^{h+1}\mathrm{d}A^{h+1}\beta(A^{h+1}, t)\rho_{m+1,n-1}^{(h+1,k,\ell)}(\mathbf{X}^{h+1};\, \mathbf{Y}_{\mathrm{e}}^{2k+2\ell};\, \mathbf{A}^{h+1};\, \mathbf{B}^{k+\ell};\, t) \\
&\quad + \frac{2(n-k-\ell)(m-h)}{m}\int_{\Lambda^2} \mathrm{d}Y^{2k+2}\mathrm{d}B^{k+1}\, \beta(B^{k+1};\, t)\rho_{m-1,n}^{(h,k+1,\ell)}(\mathbf{X}^h;\, \mathbf{Y}_{\mathrm{e}}^{2k+2\ell+2};\, \mathbf{A}^h;\, \mathbf{B}^{k+\ell+1};\, t) \\
&\quad + \frac{2(n-k-\ell)}{m}\sum_{i=1}^{h} \beta(A^i, t) \\
&\qquad \times \rho_{m-1,n}^{(h-1,k+1,\ell)}(\mathbf{X}_{-i}^h,\, \mathbf{Y}_{\mathrm{e}}^{2k+2+2\ell}[Y^{2k+2} = X^i],\, \mathbf{A}_{-i}^h,\, \mathbf{B}^{k+\ell+1}[B^{k+1} = A^i],\, t),
\end{aligned}
\tag{3.3.2}
$$

and the associated boundary conditions become

$$\rho_{m,n}^{(h,k,\ell)}(\mathbf{X}^h;\, \mathbf{Y}_{\mathrm{e}}^{2k+2\ell}[Y^{2k}{=}\, y], \mathbf{A}^h\, \mathbf{B}^{k+\ell}[B^k = 0], t) =$$

$$\begin{aligned}
&\frac{m+1}{n}\int_{\Lambda^2} \mathrm{d}A^{h+1}\mathrm{d}s\, \tilde{\beta}(y+s, y, A^{h+1}, t)\rho_{m+1,n-1}^{(h+1,k-1,\ell)}(\mathbf{X}^{h+1}[X^{h+1}{=}\, y+s], \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \mathbf{Y}_{\mathrm{e}}^{2k+2\ell-2}, \mathbf{A}^{h+1}, \mathbf{B}^{k+\ell-1}, t) \\
&+\frac{2(m-h)}{m}\int_{\Lambda^2} \mathrm{d}B^k\mathrm{d}s\, \tilde{\beta}(y+s, y, B^k, t)\rho_{m-1,n}^{(h,k,\ell)}(\mathbf{X}^h;\, \mathbf{Y}_{\mathrm{e}}^{2k+2\ell}[Y^{2k}{=}\, y+s], \mathbf{A}^h;\, \mathbf{B}^{k+\ell};, t) \\
&+\frac{2}{m}\sum_{i=1}^{h}\int_{\Lambda} \mathrm{d}s\, \tilde{\beta}(y+s, y, A^i, t)\rho_{m-1,n}^{(h-1,k-1,\ell+1)}(\mathbf{X}_{-i}^h, \\
&\qquad\qquad\qquad\qquad \mathbf{Y}_{\mathrm{e}}^{2k+2\ell}[Y^{2k+2\ell-1}{=}\, y+s, Y^{2k+2\ell}{=}\, X^i], \mathbf{A}_{-i}^h, \mathbf{B}^{k+\ell}[B^k = A^i], t),
\end{aligned} \tag{3.3.3}$$

$$\rho_{m,n}^{(h,k,\ell)}(\mathbf{X}^h;\, \mathbf{Y}_{\mathrm{e}}^{2k+2\ell}[Y^{2k+2\ell-1}{=}\, y_1, Y^{2k+2\ell}{=}\, y_2], \mathbf{A}^h;\, \mathbf{B}^{k+\ell}[B^{k+\ell}{=}\, 0], t) =$$

$$\begin{aligned}
&\frac{m+1}{n}\int_{\Lambda} \mathrm{d}A^{h+1}\, \tilde{\beta}(y_1+y_2, y_1, A^{h+1}, t)\rho_{m+1,n-1}^{(h+1,k,\ell-1)}(\mathbf{X}^{h+1}[X^{h+1}{=}\, y_1+y_2], \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \mathbf{Y}_{\mathrm{e}}^{2k+2\ell-2}, \mathbf{A}^{h+1}, \mathbf{B}^{k+\ell-1}, t) \\
&+\frac{2(m-h)}{m}\int_{\Lambda} \mathrm{d}B^{k+1}\, \tilde{\beta}(y_1+y_2, y_1, B^{k+1}, t)\rho_{m-1,n}^{(h,k+1,\ell-1)}(\mathbf{X}^h, \mathbf{Y}_{\mathrm{e}}^{2k+2\ell}[Y^{2k+2}{=}\, y_1+y_2], \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \mathbf{A}^h;\, \mathbf{B}^{k+\ell};, t) \\
&+\frac{2}{m}\sum_{i=1}^{h}\tilde{\beta}(y_1+y_2, y_1, A^i, t)\rho_{m-1,n}^{(h-1,k,\ell)}(\mathbf{X}_{-i}^h, \mathbf{Y}_{\mathrm{e}}^{2k+2\ell}[Y^{2k+2\ell-1}{=}\, y_1+y_2, Y^{2k+2\ell}{=}\, X^i], \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \mathbf{A}_{-i}^h, \mathbf{B}^{k+\ell}[B^{k+\ell} = A^i], t),
\end{aligned} \tag{3.3.4}$$

and

$$\rho_{m,n}^{(h,k,\ell)}(\mathbf{X}^h[X^i=0],\mathbf{Y}_{\rm e}^{2k+2\ell},\mathbf{A}^h,\mathbf{B}^{k+\ell},t)=\rho_{m,n}^{(h,k,\ell)}(\mathbf{X}^h[X^i=\infty],\mathbf{Y}_{\rm e}^{2k+2\ell},\mathbf{A}^h,\mathbf{B}^{k+\ell},t)=0,$$
$$i=1,2,...,h,$$

$$\rho_{m,n}^{(h,k,\ell)}(\mathbf{X}^h,\mathbf{Y}_{\rm e}^{2k+2\ell}[Y^j=0],\mathbf{A}^h,\mathbf{B}^{k+\ell},t)=\rho_{m,n}^{(h,k)}(\mathbf{X}^h,\mathbf{Y}_{\rm e}^{2k+2\ell}[Y^j=\infty],\mathbf{A}^h,\mathbf{B}^{k+\ell},t)=0,$$
$$j=2,4,...,2k,2k+1,...,2k+2\ell,$$

$$\rho_{m,n}^{(h,k,\ell)}(\mathbf{X}^h,\mathbf{Y}_{\rm e}^{2k+2\ell},\mathbf{A}^h[A^i=0],\mathbf{B}^k,t)=0,$$
$$i=1,2,...,h,$$

$$\rho_{m,n}^{(h,k,\ell)}(\mathbf{X}^h,\mathbf{Y}_{\rm e}^{2k+2\ell},\mathbf{A}^h,\mathbf{B}^{k+\ell},t)=0,$$
if two or more entries in $\mathbf{B}^{k+\ell}$ are 0.

(3.3.5)

The first two terms on the RHS of Eq. (3.3.2) represent the division of a singlet/doublet in the current system whose age is specified; the third and fourth terms on the RHS describe the division of a singlet and one cell of a doublet, respectively, whose age is not specified; the fifth term results from the division of a singlet, whose age and volume are unspecified, that induces the state transition $(m+1,n-1)\to(m,n)$. The sixth term arises from division of one cell of a doublet that converts the system from $(m-1,n)$ to (m,n). Finally, the last term represents the division of one cell in a doublet whose age is $A^i, 1\le i\le h$ and its undividing twin has size X^i. In Eqs. (3.3.3) and (3.3.4), the first term on their RHSs represents the division of a singlet and the second term on their RHSs describes the division of one cell in a doublet, giving rise to a newborn doublet and leaving a singlet whose volume and age are integrated over. The last term in the boundary conditions in Eqs. (3.3.3) and (3.3.4) results from the division of a cell in a doublet, giving rise to a newborn doublet and leaving a singlet whose volume and age are $X^i\in\mathbf{X}^h$ and $A^i\in\mathbf{A}^h$, respectively.

The differential equations satisfied by the fully marginalized density $\rho_{m,n}^{(0,0,0)}$ are

$$\begin{aligned}\frac{\partial \rho_{m,n}^{(0,0,0)}(t)}{\partial t} = &\int_{\Lambda^2} \mathrm{d}X^1 \mathrm{d}A^1\, \beta(A^1, t) \left[(m+1)\rho_{m+1,n-1}^{(1,0,0)}(\mathbf{X}^1, \mathbf{A}^1, t) - m\rho_{m,n}^{(1,0,0)}(\mathbf{X}^1, \mathbf{A}^1, t)\right] \\ &+ 2n \int_{\Lambda^2} \mathrm{d}Y^2 \mathrm{d}B^1\, \beta(B^1, t) \left[\rho_{m-1,n}^{(0,1,0)}(\mathbf{Y}_{\mathrm{e}}^2, \mathbf{B}^1, t) - \rho_{m,n}^{(0,1,0)}(\mathbf{Y}_{\mathrm{e}}^2, \mathbf{B}^1, t)\right].\end{aligned} \tag{3.3.6}$$

which, for an age-independent division rate, explicitly reduces to the simple birth-death master equation

$$\frac{1}{\beta(t)} \frac{\partial \rho_{m,n}^{(0,0,0)}(t)}{\partial t} = (m+1)\rho_{m+1,n-1}^{(0,0,0)}(t) - m\rho_{m,n}^{(0,0,0)}(t) + 2n\rho_{m-1,n}^{(0,0,0)}(t) - 2n\rho_{m,n}^{(0,0,0)}(t). \tag{3.3.7}$$

In Eqs. (3.3.6) and (3.3.7), the division rates $\beta(A^1, t)$ and $\beta(t)$ can be replaced by their full (m, n)-dependent forms.

3.3.1 Number-weighted density functions

We now define a class of number-weighted density functions from the marginalized densities $\rho_{m,n}^{(h,k,\ell)}$ that incorporates higher moments and that has useful closure properties:

$$\begin{aligned}u^{(k,\ell)}(\mathbf{x}^k, \mathbf{y}^{2\ell}, \mathbf{a}^k, \mathbf{b}^\ell, t) \coloneqq \sum_{m,n=0}^{\infty} \sum_{r=0}^{k} \sum_{\xi^{(0,r)} \in S_k} &2^{k+\ell-r} (m)_r (n)_{k+\ell-r}\, \rho_{m,n}^{(r,k-r,\ell)}(\mathbf{X}^r[X^i = x^{\xi^{(0,r)}(i)}], \\ &\mathbf{Y}_{\mathrm{e}}^{2(k-r)+2\ell}[Y^{2j} = x^{\xi^{(r,k-r)}(j)}, Y^{2(k-r)+p} = y^p], \mathbf{A}^r[A^i = a^{\xi^{(0,r)}(i)}], \\ &\mathbf{B}^{k-r+\ell}[B^j = a^{\xi^{(r,k-r)}(j)}, B^{k-r+[\frac{p+1}{2}]} = b^{[\frac{p+1}{2}]}], t), \\ &1 \le i \le r, 1 \le j \le k-r, 1 \le p \le 2\ell\end{aligned} \tag{3.3.8}$$

where $\mathbf{x}^k \coloneqq (x^1, ..., x^k), \mathbf{y}^{2\ell} \coloneqq (y^1, ..., y^{2\ell}), \mathbf{a}^k \coloneqq (a^1, ..., a^k), \mathbf{b}^\ell \coloneqq (b^1, ..., b^\ell)$, and $(m)_r = m!/(m-r)!$ is the falling factorial, $S_k = \{1, 2, ..., k\}$. The sum $\sum_{\xi^{(0,r)} \in S_k}$ includes all elements in the set $\xi^{(0,r)} \in \Omega_r$ containing all possible choices of r elements in S_k and $\xi^{(r,k-r)} \coloneqq (\xi(r+1), \xi(r+2), ...\xi(k)) = S_k \backslash \xi^{(0,r)}$. We require $\xi^{(0,r)}(i) < \xi^{(0,r)}(j), \xi^{(r,k-r)}(i) < \xi^{(r,k-r)}(j)$, $\forall i < j$, and $r \le m$, and $k - r \le n$ in Eq. (3.3.8). Note that $u^{(0,0)} \equiv 1$ from normalization. The

lowest order number-weighted density functions $u^{(k,\ell)}$ are explicitly given in Appendix B.

Since our kinetic equations (Eqs. (3.2.15), (3.2.16), and (3.2.17)) subsume all hierarchical equations for $\rho_{m,n}^{(h,k,\ell)}$ (Eqs. (3.3.2), (3.3.3), (3.3.4), and (3.3.5)), equations for $u^{(k,\ell)}$ can be derived. For example, if β is independent of m, n, the PDE satisfied by $u^{(k,\ell)}(\mathbf{x}^k, \mathbf{y}^{2\ell}, \mathbf{a}^k, \mathbf{b}^\ell, t)$ is

$$\begin{aligned}
&\frac{\partial u^{(k,\ell)}}{\partial t} + \sum_{i=1}^{k} \frac{\partial u^{(k,\ell)}}{\partial a^i} + \sum_{j=1}^{\ell} \frac{\partial u^{(k,\ell)}}{\partial b^j} + \sum_{i=1}^{k} \frac{\partial (g(x^i, a^i, t) u^{(k,\ell)})}{\partial x^i} + \sum_{j=1}^{2\ell} \frac{\partial (g(y^j, b^{[\frac{j+1}{2}]}, t) u^{(k,\ell)})}{\partial y^j} \\
&\qquad = -\left(\sum_{i=1}^{k} \beta(a^i, t) + \sum_{j=1}^{\ell} 2\beta(b^j, t) \right) u^{(k,\ell)} \\
&\qquad\quad + \frac{1}{2} \sum_{i=1}^{k} \frac{\partial^2 (\sigma^2(x^i, a^i, t) u^{(k,\ell)})}{(\partial x^i)^2} + \frac{1}{2} \sum_{j=1}^{2\ell} \frac{\partial^2 (\sigma^2(y^j, b^{[\frac{j+1}{2}]}, t) u^{(k,\ell)})}{(\partial y^j)^2},
\end{aligned} \tag{3.3.9}$$

with the boundary conditions

$$\begin{aligned}
u^{(k,\ell)}(\mathbf{x}^k[x^v = x], \mathbf{y}^{2\ell}, \mathbf{a}^k[a^v = 0], \mathbf{b}^\ell, t) &= \sum_{m,n=0}^{\infty} \sum_{r=0}^{k-1} \sum_{\xi^{(0,r)} \in S_k^{-v}} 2^{\ell+k-r} (m)_r (n)_{k+\ell-r} \\
&\quad \times \rho_{m,n}^{(r,k-r,\ell)}(\mathbf{X}^r[X^i = x^{\xi^{(0,r)}(i)}], \mathbf{Y}_{\mathrm{e}}^{2k-2r+2\ell}[Y^{2j} = x^{\xi^{(r,k-r)}(j)}, Y^{2k+p} = y^p], \\
&\qquad \mathbf{A}^r[A^i = a^{\xi^{(0,r)}(i)}], \mathbf{B}^{\ell+k-r}[B^j = a^{\xi^{(r,k-r)}(j)}, B^{k-r+[\frac{p+1}{2}]} = b^{[\frac{p+1}{2}]}], t) \\
&= 2 \int_{\Lambda^2} \mathrm{d}s \, \mathrm{d}a \, \tilde{\beta}(x+s, x, a, t) u^{(k,\ell)}(\mathbf{x}^k[x^k = x+s], \mathbf{y}^{2\ell}, \mathbf{a}^k[a^k = a], \mathbf{b}^\ell, t) \\
&\quad + 2 \sum_{w=1, \neq v}^{k} \int_{\Lambda} \mathrm{d}s \, \tilde{\beta}(x+s, x, a^w, t) u^{(k-2,\ell+1)}(\mathbf{x}_{-v,-w}^k, \mathbf{a}_{-v,-w}^k, \\
&\qquad\qquad \mathbf{y}^{2\ell+2}[y^{2\ell+1} = x^v, y^{2\ell+2} = x+s], \mathbf{b}^{\ell+1}[b^{\ell+1} = a^w], t)
\end{aligned} \tag{3.3.10}$$

$$
\begin{aligned}
&u^{(k,\ell)}(\mathbf{x}^k, \mathbf{y}^{2\ell}[y^{2v-1} = y_1, y^{2v} = y_2], \mathbf{a}^k, \mathbf{b}^\ell[b^v = 0], t) \\
&\quad = \sum_{m,n=0}^{\infty} \sum_{r=0}^{k} \sum_{\xi^{(0,r)} \in S_k} 2^{\ell+k-r} (m)_r (n)_{k+\ell-r} \rho_{m,n}^{(r,k-r,\ell)} (\mathbf{X}^r[X^i = x^{\xi^{(0,r)}(i)}], \\
&\qquad\qquad \mathbf{Y}_{\mathrm{e}}^{2\ell+k-r}[Y^{2j} = x^{\xi^{(r,k-r)}(j)}, Y^{2k+q} = y^q], \mathbf{A}^r[A^i = a^{\xi^{(0,r)}(i)}], \\
&\qquad\qquad \mathbf{B}^{\ell+k-r}[B^j = a^{\xi^{(r,k-r)}(j)}, B^{k-r+[\frac{q+1}{2}]} = b^{[\frac{q+1}{2}]}], t) \\
&\quad = 2 \int_\Lambda \mathrm{d}a\, \tilde{\beta}(y_1 + y_2, y_1, a, t) \\
&\qquad\qquad \times u^{(k+1,\ell-1)}(\mathbf{x}^{k+1}[x^{k+1} = y_1 + y_2], \mathbf{y}^{2\ell}_{-(2v-1),-2v}, \mathbf{a}^{k+1}[a^{k+1} = a], \mathbf{b}^\ell_{-v}, t) \\
&\quad + 2 \sum_{w=1}^{k} \tilde{\beta}(y_1 + y_2, y_1, a^w, t) \\
&\qquad\qquad \times u^{(k-1,\ell)}(\mathbf{x}^k_{-w}, \mathbf{a}^k_{-w}, \mathbf{y}^{2\ell}[y^{2v-1} = y_1 + y_2, y^{2v} = x^w], \mathbf{b}^\ell[b^v = a^w], t),
\end{aligned}
\tag{3.3.11}
$$

where $\mathbf{x}^k_{-v} \coloneqq (x^1, ..., x^{v-1}, x^{v+1}, ..., x^k)$, $\mathbf{a}^k_{-v} \coloneqq (a^1, ..., a^{v-1}, ..., a^{v+1}, ...a^k)$, $\mathbf{x}^k_{-v,-w} \coloneqq (x^1, ..., x^{v-1}, x^{v+1}, ..., x^{w-1}, x^{w+1}, ..., x^k)$, $\mathbf{a}^k_{-v,-w} \coloneqq (a^1, ..., a^{v-1}, a^{v+1}, ..., a^{w-1}, a^{w+1}, ..., a^k)$, $\mathbf{y}^{2\ell}_{-(2v-1),-2v} \coloneqq (y^1, ..., y^{2v-2}, y^{2v+1}, ..., y^{2\ell})$, $\mathbf{b}^\ell_{-v} \coloneqq (b^1, ..., b^{v-1}, b^{v+1}, ..., b^\ell)$ and $S_k^{-v} \coloneqq \{1, 2, ..., v-1, v+1, .., k\}$. The additional conditions,

$$
u^{(k,\ell)}(\mathbf{x}^k, \mathbf{y}^{2\ell}, \mathbf{a}^k, \mathbf{b}^\ell, t) = 0 \begin{cases} \text{if any } x_i, y_j = 0, \infty \\ \text{if two or more } a_i \text{ or } b_j{=}0 \end{cases}
\tag{3.3.12}
$$

are found by using Eq. (3.3.5) in Eq. (3.3.8). Note that the PDE (Eq. (3.3.9)) for each $u^{(k,\ell)}$ is "closed" and does not involve other density functions $u^{(k',\ell')}$. However, the boundary conditions (Eqs. (3.3.10) and (3.3.11)) couple $u^{(k,\ell)}, k + \ell > 1$ with $u^{(k+1,\ell-1)}, u^{(k-1,\ell)}$, or $u^{(k-2,\ell+1)}$, preventing direct closure at the level of each set of indices k, ℓ. Nonetheless, although the full models for $u^{(k,\ell)}, k+\ell > 1$ are not closed, the boundary conditions will only involve $u^{(k',\ell')}$ such that $k' + 2\ell' \leq k + 2\ell$, and therefore all $u^{(k,\ell)}, k + \ell > 1$ can be solved sequentially after we have found $u^{(1,0)}$, which can be completely determined by solving the PDE

$$
\frac{\partial u^{(1,0)}}{\partial t} + \frac{\partial u^{(1,0)}}{\partial a} + \frac{\partial (g u^{(1,0)})}{\partial x} = -\beta(a,t) u^{(1,0)}(x,a,t) + \frac{1}{2} \frac{\partial^2 (\sigma^2 u^{(1,0)})}{(\partial x)^2}
\tag{3.3.13}
$$

with associated boundary conditions specified at $a = 0, x = 0, x = \infty$

$$\begin{aligned} u^{(1,0)}(x,0,t) &= 2n \sum_{m=0}^{\infty} \sum_{n=1}^{\infty} \rho_{m,n}^{(0,1,0)}(\mathbf{Y}_{\mathrm{e}}^2[Y^2 = x], \mathbf{B}^1[B^1 = 0], t) \\ &= 2 \int_x^{\infty} \mathrm{d}z \int_{\Lambda} \mathrm{d}a \, \tilde{\beta}(z,x,a,t) u^{(1,0)}(z,a,t), \\ u^{(1,0)}(0,a,t) &= u^{(1,0)}(\infty,a,t) = 0. \end{aligned} \tag{3.3.14}$$

The model for $u^{(1,0)}$ is essentially the standard mean-field sizer-timer model [RHK14, XGC20] but with an additional diffusion term $\frac{\partial^2(\sigma^2 u^{(1,0)})}{(\partial x)^2}$ representing the random growth rate of each independent cell. To explicitly illustrate how growth rate stochasticity affects the evolution of the structured population, we numerically solve Eqs. (3.3.13) and (3.3.14). We set $g(x,a,t) = x/2$ and a constant rate $\beta(x,a,t) = (2\ln 2)^{-1}$ describing an exponentially distributed division time with mean 2ln2. We also assume $\tilde{\beta} = \beta(x,a,t)\delta(z/x)/x$ where δ is a Dirac measure enforcing symmetric division. The initial condition is $u^{(1,0)}(x,a,0) = xe^{-2a-x/5}$. We use the adaptive spectral method proposed in [XSC21b] to numerically compute $u^{(1,0)}(x,a,t)$ for different growth noise $\sigma = 0, \sqrt{x}, \sqrt{2x}$. We construct the mean cell size

$$\langle x(t) \rangle = \frac{\int_{\Lambda^2} x u^{(1,0)}(x,a,t) \mathrm{d}x\mathrm{d}a}{\int_{\Lambda^2} u^{(1,0)}(x,a,t) \mathrm{d}x\mathrm{d}a} \tag{3.3.15}$$

and plot their evolution in Fig. 3.1.

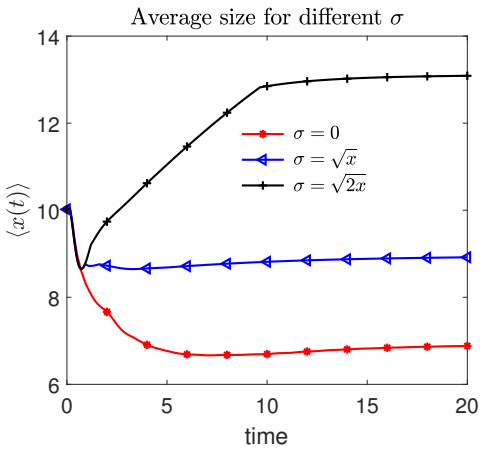

Figure 3.1: Mean cell sizes $\langle x(t) \rangle$ under symmetric division for different growth rate noise functions $\sigma = 0, \sqrt{x}, \sqrt{2x}$. After an initial transient, we see that a larger σ leads to larger average sizes. Even when the mean division times are kept fixed, larger noise in growth rates leads to broader distributions of cell sizes which increases the mean.

Given the solution to $u^{(1,0)}(x, a, t)$, we can calculate $u^{(0,1)}$ and then $u^{(2,0)}$, $u^{(1,1)}$, and so on. How the different $u^{(k,\ell)}$ are connected through the boundary conditions are illustrated in Fig. 3.2, demonstrating the sequence to follow to fully solve the single-density problem. The differential equation satisfied by the lowest order moment $\mathbb{E}[N(t)]$ requires $u^{(1,0)}$, as indicated by the shaded blue arrow in Fig. 3.2(a). The two sequences traced by the boundary conditions (3.3.10) and (3.3.11) are shown in Figs. 3.2(a) and (b), respectively. In Fig. 3.2(c) we show the combined sequence of boundary condition calculations to find $u^{(1,2)}$: the equations satisfied by $u^{(1,0)}$ are fully closed so $u^{(1,0)}$ can be first calculated. In the second step, we use $u^{(1,0)}$ to construct the boundary condition and solve for $u^{(0,1)}$. The third step is to use $u^{(0,1)}$ to construct the boundary condition and solve for $u^{(2,0)}$. The boundary condition dependences of $u^{(1,0)}, u^{(2,0)}$ are indicated by blue arrows. The fourth and fifth steps are to solve for $u^{(1,1)}$ and $u^{(3,0)}$, whose boundary condition dependences are indicated by the green arrows. Next, we calculate $u^{(2,1)}$, $u^{(0,2)}$, and finally $u^{(1,2)}$, whose boundary condition dependences are shown by the red arrows. These higher dimensional results capture the stochasticity arising only from the noisy growth of each cell (through the diffusive terms in Eqs. (3.3.9) and (3.3.13)). When the coefficients satisfy certain conditions, it is also possible to further reduce the full

kinetic models for $u^{(k,\ell)}$ in Eqs. (3.3.9) and (3.3.12) to simpler models, which are derived in previous literature like [Per08, CG16] by integrating over the size variable x or age variable a. This is explicitly shown in Appendix A.2.3.

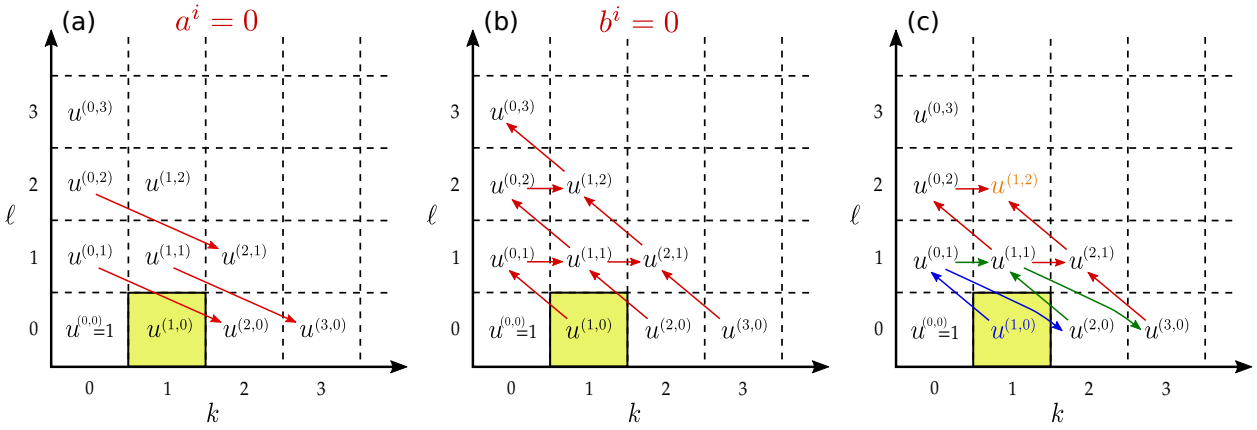

Figure 3.2: A map of boundary condition interdependences for single-density kinetic theory. In (a) we indicate the dependence of the boundary condition for the quantity $u^{(k,\ell)}(\mathbf{x}^k, \mathbf{a}^k, \mathbf{y}^{2\ell}, \mathbf{b}^\ell, t)$ if any $a^i = 0$. The boundary condition for $u^{(k,\ell)}$ depends on itself and $u^{(k-2,\ell+1)}$; for example, $u^{(0,1)}$ is required for the boundary condition for $u^{(2,0)}$, so the red arrow points from $u^{(0,1)}$ to $u^{(2,0)}$. In (b) we indicate the dependence of the boundary condition for $u^{(k,\ell)}(\mathbf{x}^k, \mathbf{a}^k, \mathbf{y}^{2\ell}, \mathbf{b}^\ell, t)$ if any $b^j = 0$. Here, the boundary condition for $u^{(k,\ell)}$ depends on $u^{(k+1,\ell-1)}$ and $u^{(k-1,\ell)}$. (c) An example of an explicit sequence of calculations to find $u^{(1,2)}$ starting from $u^{(1,0)}$.

3.3.2 Moments of the total population

In addition to the number-weighted densities defined in Eq. (3.3.8), one can also investigate moments of the total cell number $N = m + 2n$. The expected moments of the total cell population

$$\mathbb{E}[N^k(t)] = \sum_{m=0}^{\infty}\sum_{n=0}^{\infty}(m+2n)^k \rho_{m,n}^{(0,0,0)}, \tag{3.3.16}$$

can be used to find, for $k = 1$

$$\begin{aligned}\frac{\mathrm{d}\mathbb{E}[N(t)]}{\mathrm{d}t} &= \sum_{m,n=0}^{\infty}\Bigg[m\int_{\Lambda^2}\mathrm{d}X^1\mathrm{d}A^1\,\beta(A^1;\,t)\rho_{m,n}^{(1,0,0)}(\mathbf{X}^1;\,\mathbf{A}^1;\,t) \\ &\qquad\qquad +2n\int_{\Lambda^2}\mathrm{d}Y^2\mathrm{d}B^1\,\beta(B^1;\,t)\rho_{m,n}^{(0,1,0)}(\mathbf{Y}_{\mathrm{e}}^2,\mathbf{B}^1;\,t)\Bigg] \\ &= \int_{\Lambda^2}\mathrm{d}x\,\mathrm{d}a\,\beta(a,t)u^{(1,0)}(x,a,t).\end{aligned} \tag{3.3.17}$$

The differential equation for $\mathbb{E}[N(t)]$ does not involve any boundary condition, but depends on $u^{(1,0)}$. Nonetheless, using the solutions to Eqs. (3.3.13) and (3.3.14) one can explicitly solve Eq. (3.3.17) to find $\mathbb{E}[N(t)]$.

The demographic stochasticity arising from random birth (and possibly death) times affects the total population and is most directly summarized by higher total-population correlations. For example, the differential equation satisfied by $\mathbb{E}[N^2(t)]$ is found to be

$$\begin{aligned}\frac{\mathrm{d}\mathbb{E}[N^2(t)]}{\mathrm{d}t} = \sum_{m,n=0}^{\infty}\Bigg[&(2m^2+4mn+m)\int\mathrm{d}X^1\mathrm{d}A^1\,\beta(A^1;\,t)\rho_{m,n}^{(1,0,0)}(\mathbf{X}^1;\,\mathbf{A}^1;\,t) \\ &+(8n^2+4mn+2n)\int\mathrm{d}Y^2\mathrm{d}B^1\,\beta(B^1;\,t)\rho_{m,n}^{(0,1,0)}(\mathbf{Y}_{\mathrm{e}}^2,\,\mathbf{B}^1;\,t)\Bigg].\end{aligned} \tag{3.3.18}$$

which cannot be solved even knowing all $u^{(k,\ell)}$. However, the expectations decouple if $\beta(t)$ is independent of age and take on the simple form

$$\frac{\mathrm{d}\mathbb{E}[N^k(t)]}{\mathrm{d}t} = \beta(t)\sum_{j=0}^{k-1}\binom{k}{j}\mathbb{E}[N^{j+1}(t)], \tag{3.3.19}$$

which can then be solved by starting with the solution of $\mathbb{E}[N(t)]$. For general age-dependent division rates $\beta(a,t)$, $\mathbb{E}[N^{k>1}(t)]$ cannot be directly computed/approximated without also closing Eq. (3.3.2). Such equations, as well as those for higher-number moments such as $\sum_{m,n} m^k \rho_{m,n}^{(h,k,\ell)}$ are *not* closed and form complex hierarchies that need additional assumptions to close.

3.4 Generalizations and extensions

3.4.1 Incorporation of death

Here, we show how our kinetic theory is modified when an age and size-dependent death, occurring with rate $\mu(a,t)$, is incorporated. By defining

$$\gamma(a,t) = \beta(a,t) + \mu(a,t) \tag{3.4.1}$$

the joint survival probabilities $S_{1,m}$ and $S_{2,n}$ in Eq. (3.2.7) are modified by

$$\tilde{S}_{1,m}(t|t',\mathbf{A}_{t'}^m) = \prod_{i=1}^{m} e^{-\int_{t'}^{t}\gamma(A_{t'}^i - t' + s, s)\mathrm{d}s}, \quad \tilde{S}_{2,n}(t|t',\mathbf{B}_{t'}^n) = \prod_{j=1}^{n}\left[e^{-\int_{t'}^{t}\gamma(B_{t'}^j - t' + s, s)\mathrm{d}s}\right]^2. \tag{3.4.2}$$

Following the previous derivations, we find

$$\begin{aligned}\frac{\partial\rho_{m,n}}{\partial t} + \sum_{i=1}^{m}\frac{\partial\rho_{m,n}}{\partial A^i} + \sum_{j=1}^{n}\frac{\partial\rho_{m,n}}{\partial B^j} + \sum_{i=1}^{m}\frac{\partial(g(X^i,A^i,t)\rho_{m,n})}{\partial X^i} + \sum_{j=1}^{2n}\frac{\partial(g(Y^j,B^j,t)\rho_{m,n})}{\partial Y^j} \\ = -\left(\sum_{i=1}^{m}\gamma(A^i,t) + 2\sum_{j=1}^{n}\gamma(B^j,t)\right)\rho_{m,n}\end{aligned}$$

$$
\begin{aligned}
&+\frac{1}{2}\sum_{i=1}^{m}\frac{\partial^2(\sigma^2(X^i,A^i,t)\rho_{m,n})}{(\partial X^i)^2}+\frac{1}{2}\sum_{j=1}^{2n}\frac{\partial(\sigma^2(Y^j,B^j,t)\rho_{m,n})}{(\partial Y^j)^2} \\
&+(m+1)\int_{\Lambda^2}\mathrm{d}A^{m+1}\mathrm{d}X^{m+1}\,\mu(A^{m+1},t)\rho_{m+1,n}(\mathbf{X}^{m+1},\mathbf{Y}^{2n},\mathbf{A}^{m+1},\mathbf{B}^{n},t) \\
&+\frac{2(n+1)}{m}\sum_{i=1}^{m}\int_{\Lambda}\mathrm{d}x\,\mu(A^i,t)\rho_{m-1,n+1}(\mathbf{X}^{m}_{-i},\mathbf{Y}^{2n+2}[Y^{2n+1}=x,Y^{2n+2}=X^i], \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\mathbf{A}^{m}_{-i},\mathbf{B}^{n+1}[B^{n+1}=A^i],t),
\end{aligned}
\tag{3.4.3}
$$

where the argument of $\rho_{m,n}$ in the first two lines is $(\mathbf{X}^m,\mathbf{Y}^{2n},\mathbf{A}^m,\mathbf{B}^n,t)$.

The boundary conditions for $\rho_{m,n}$ are the same as Eq. (3.2.16) and Eq. (3.2.17) since only cell division contributes to the boundary term. Similarly, we can define the marginal distribution $\rho^{(h,k,\ell)}_{m,n}(\mathbf{X}^h,\mathbf{Y}^{2k+2\ell}_{\mathrm{e}},\mathbf{A}^h,\mathbf{B}^k,t)$ and the higher-dimensional number-weighted densities functions $u^{(k,\ell)}(\mathbf{x}^k,\mathbf{y}^{2\ell},\mathbf{a}^k,\mathbf{b}^\ell,t)$ in the same way as in Eqs. (3.3.2) and (3.3.8), respectively. The $k=1,\ell=0$ density obeys

$$
\frac{\partial u^{(1,0)}}{\partial t}+\frac{\partial(gu^{(1,0)})}{\partial x}+\frac{\partial u^{(1,0)}}{\partial a}=-(\beta(a,t)+\mu(a,t))u^{(1,0)}(x,a,t)+\frac{1}{2}\frac{\partial^2(\sigma^2u^{(1,0)})}{(\partial x)^2},
\tag{3.4.4}
$$

and boundary conditions specified in Eqs. (3.3.14).

3.4.2 Correlated noise in growth rate

In this subsection, we consider a model in which the noise in growth rates is correlated across cells. By defining $\mathbf{Z}^{m,2n}=(\mathbf{X}^m,\mathbf{Y}^{2n})$ and $\mathbf{C}^{m,2n}=(\mathbf{A}^m,B^1,B^1,...,B^n,B^n)$ to be the volumes and ages of m singlets and n doublets at time t, we can describe the growth rate as

$$
\mathrm{d}\mathbf{Z}^{m,2n}_t=G^{m,2n}(\mathbf{Z}^{m,2n}_t,\mathbf{C}^{m,2n}_t,t)\mathrm{d}t+\Sigma^{m,2n}(\mathbf{Z}^{m,2n}_t,\mathbf{C}^{m,2n}_t,t)\mathrm{d}\mathbf{W}^p_t,
\tag{3.4.5}
$$

where $G^{m,2n}\in\mathbb{R}^{m+2n}$, $\Sigma^{m,2n}(\mathbf{Z}^{m,2n}_t,\mathbf{C}^{m,2n}_t,t)=(\sigma)_{ij}\in\mathbb{R}^{(m+2n)\times p}$ and $\mathbf{W}^p_t$ is a p-dimensional i.i.d standard Wiener process [Dur19]. For simplicity, we assume that the i^{th} compo-

nent of $G^{m,2n}$ is $g_i(Z_t^i, C_t^i, t) = g(Z^i, C^i, t)$, indicating that the deterministic part of the growth rate is identical for all cells. Following our derivation in Section 3.2, we find that $\rho_{m,n}(\mathbf{X}^m, \mathbf{Y}^{2n}, \mathbf{A}^m, \mathbf{B}^n, t)$ satisfies

$$\begin{aligned}
&\frac{\partial \rho_{m,n}}{\partial t} + \sum_{i=1}^{m} \frac{\partial \rho_{m,n}}{\partial A^i} + \sum_{j=1}^{n} \frac{\partial \rho_{m,n}}{\partial B^i} + \sum_{i=1}^{m} \frac{\partial (g(t, X^i, A^i)\rho_{m,n})}{\partial X^i} + \sum_{j=1}^{2n} \frac{\partial (g(t, Y^j, B^{[(j+1)/2]})\rho_{m,n})}{\partial Y^j} \\
&= -\Bigg(\sum_{i=1}^{m} \beta(A^i, t) + \sum_{j=1}^{n} 2\beta(B^j, t) \Bigg) \rho_{m,n}(\mathbf{X}^m, \mathbf{Y}^{2n}, \mathbf{A}^m, \mathbf{B}^n, t) + \frac{1}{2} \sum_{s_1, s_2 = 1}^{m+2n} \frac{\partial^2 (D_{s_1,s_2} \rho_{m,n})}{\partial Z^{s_1} \partial Z^{s_2}},
\end{aligned} \tag{3.4.6}$$

where $D_{s_1,s_2} = \sum_{\ell=1}^{p} \sigma_{s_1,\ell} \sigma_{s_2,\ell}$. The boundary conditions for $\rho_{m,n}$ are the same as that described by Eq. (3.2.16) and Eq. (3.2.17). Similarly, we can define the marginal distribution density function $\rho_{m,n}^{(h,k,\ell)}$ in the same way as in Section 3.3, and it can be verified that the differential equations as well as the boundary conditions satisfied by $\rho_{m,n}^{(1,0,0)}(\mathbf{X}^1[X^1 = x], \mathbf{A}^1[A^1 = a], t), \rho_{m,n}^{(0,1,0)}(\mathbf{X}^1[X^1 = x], \mathbf{A}^1[A^1 = a], t)$ are the same as those satisfied by $\rho_{m,n}^{(1,0,0)}(\mathbf{X}^1[X^1 = x], \mathbf{A}^1[A^1 = a], t)$ and $\rho_{m,n}^{(0,1,0)}(\mathbf{Y}^1[Y^1 = x], \mathbf{B}^1[B^1 = a], t)$ in Eq. (3.3.2) and Eq. (3.3.4), although the differential equations satisfied by $\rho_{m,n}$ in Eq. (3.4.6) and in Eq. (3.2.13) are different. If we further assume that the variance in growth rates for all cells is identical: $\sum_{\ell=1}^{p} \sigma_{i,\ell}^2 = \sigma^2$, $\forall i$, then the equation and boundary conditions for the "1-point" density function $u^{(1,0)}(x, a, t)$ are identical to those in Eq. (3.3.13) and Eqs. (3.3.14) since correlations in growth rate noise are not captured by a mean-field description of only one coordinate (x, a). The differences between correlated and uncorrelated growth noise among cells may arise in the differential equations for $u^{(k,\ell)}(\mathbf{x}^k, \mathbf{y}^\ell, \mathbf{a}^k, \mathbf{b}^\ell, t), k + \ell \geq 2$.

3.5 Summary and conclusions

In this chapter, we rigorously constructed a kinetic theory for structured populations, in particular for age- and size-structured cell proliferation models. We considered stochasticity in both an individual cell's growth rate ("intrinsic" stochasticity) and the cell number fluctu-

ations from random birth and death event times ("demographic" stochasticity). Derivations of the kinetic theory require the separation of 'singlet' and 'doublet' populations, as was proposed in [CG16]. However, taking into account both the size and age dependence as well as randomness in growth rates leads to the much more complex computation which we performed here.

One of our main results is the kinetic equations and boundary conditions described by Eqs. (3.2.15), (3.2.16), and (3.2.17). Marginalized densities are also found to obey more complex equations that form a hierarchy (Eqs. (3.3.2), (3.3.4), and (3.3.5)). By taking single-density averages over these equations, we find closed PDEs that govern multi-point density functions (Eq. (3.3.9)). However, the associated boundary conditions, Eq. (3.3.10), couple density functions of different dimensions. Nonetheless, the density function of all dimensions can be successively solved starting from the "1-point" density $u^{(1,0)}(x, a, t)$ which obeys Eqs. (3.3.13) and (3.3.14), a 2+1-dimensional second order PDE and associated boundary conditions that is analogous to the classic McKendrick equation but that includes a diffusive size term arising from stochasticity in growth rates. The explicit equations for the first and second moments of the total population are given by Eqs. (3.3.17) and (3.3.18), respectively.

Generalizations and extensions to our basic kinetic theory are also investigated. For example, we derived the kinetic equations when a Markovian age-dependent death process is included (Eqs. (3.3.14), (3.4.3), and (3.4.4)). We also considered noise in growth rates that are correlated across cells and showed these effects arising in "cross-diffusion" terms in the associated kinetic (and higher moment) equations.

Our unifying kinetic theory enables one to systematically analyze cell populations at both individual and population levels. A full kinetic theory may be useful for studying other processes such as failure in multicomponent systems that age and evolve [PS18]. Further feasible extensions of our kinetic equations are to include spatial distribution [AMR08] or correlations in growth rates across *generations* [XGC20]. It is also possible to consider stochasticity for different cell division strategies [NVP21]. Finally, efficient numerical methods to solve our kinetic equations can be developed, for instance in [XSC21b], equations

similar to Eqs. (3.3.13) and (3.3.14) describing the dynamics of $u^{(1,0)}$ are solved accurately and efficiently.

CHAPTER 4

Kinetic theories of generation-dependent cellular proliferation models

4.1 Introduction

Mathematical models have been formulated to describe the evolution of populations according to a number of individual attributes such as age, size, and/or added size since birth. Such structured population models have various applications across diverse fields. For example, deterministic age-structured models that incorporate age-dependent birth and death were developed by McKendrick and have been applied to human populations [Foe59]. In addition, structured population models have also been applied to investigate cell size control [TBS15, KB18], cellular division mechanisms [RHK14], and structured cell population models [Per08, MD86].

When considering proliferating cell populations, individual cell growth is interrupted by cell division events that generate daughter cells. Therefore, there has been renewed research interest recently in developing kinetic theories for cellular population models that links individual cellular growth and division to the population-level cellular proliferation [GC16, CG16, XC21]. Such kinetic modeling not only established a rigorous mathematical theory that investigates how individual cellular growth and division affects the cellular population's macroscopic quantities such as average cell size or total cell density, but also modeled stochastic effects that arise in both random cellular division times and fluctuating cellular growth rate [HLA18].

Previous kinetic models such as the timer-sizer model in [XC21] can track the evolution of individual cell's internal states like cellular size, mRNA level, or protein level by using stochastic differential equations. Furthermore, marginalizations of those kinetic models can link individual cell states with some key macroscopic quantities of the overall population. However, those kinetic models cannot explicitly track how cellular internal states evolve as cells divide across different generations (the number of times a cell has divided). Furthermore, previous models have not incorporated a death rate or division rate that depends on cells' fluctuating internal states when the evolution of those internal states is described by stochastic differential equations. With the increasing research interest in cellular differentia-

tion and cellular fate described by the Waddington's epigenetic landscape [BZA11, WZX11], and with modern computational and statistical techniques that can efficiently and accurately infer the rate of a cell to produce mRNA [LSZ18] or protein [QZM22, GSP20] given experimental data, developing a mathematical model to investigate how cells' biochemical features evolve over generations is of research importance in understanding how cells differentiate. In our paper, we rigorously propose the first kinetic model that can track cellular internal states (*e.g.*, cellular size, mRNA level, protein level, etc.) which manifest themselves as continuous variables, and the generation of a cell which is denoted by an integer-valued variable. Especially, noise in both growth rates and division times is considered. Our kinetic model can take advantage of inferred rates of producing mRNA or protein to predict how the cellular population evolves as cells divide and how cells differentiate or dedifferentiate across different generations. Through marginalization, the equations that describe the evolution of certain macroscopic of interest can also be derived from our kinetic theory.

In the next section, we propose the kinetic model that describes the evolution of each cell's evolution and division events in a cell population using stochastic differential equations. In Section 4.3, we marginalize the kinetic theories to derive the equations that describe some key mean-filed quantities of biological interest. In Section 4.4, concluding remarks are made and potential future directions are proposed. Here, we shall also provide the list of the common notations we shall use throughout this article.

4.2 Kinetic equation formulation

In this section, we shall formally derive the kinetic theory that describes the evolution of a cell population and tracks each cell's internal state (such as the size, the amount of a certain kind of protein, or the amount of an mRNA). For simplicity, we assume the cell's internal state could be characterized by a one-dimensional quantity $X \in \mathbb{R}$ and the generation of each cell is denoted by a discrete variable $i \in \mathbb{N}^+$. We shall also briefly discuss generalizations that incorporate tracking different quantities at the same time.

Symbol	Definition & explanation
$\vec{n}$	$\vec{n} \coloneqq (n_1, ..., n_k)$: the number of cells in the i^{th} generation is $n_i, I = 1, ..., K$
$\vec{X}_{\vec{n}}$	$\vec{X}_{\vec{n}} \coloneqq (\vec{X}_1, ..., \vec{X}_k)$, $\vec{X}_i \coloneqq (X_{i,1}, ..., X_{i,n_i})$ is the states of cells in the $i^{\text{th}}, I = 1, ..., K^0$ generation
$\vec{n}^0$	$\vec{n} \coloneqq (n_1^0, ..., n_k^0)$: the initial number of cells
$\vec{X}_{\vec{n}^0}$	$\vec{X}_{\vec{n}^0} \coloneqq (\vec{X}_1, ..., \vec{X}_{k^0})$, $\vec{X}_i \coloneqq (X_{i,1}, ..., X_{i,n_i^0})$ is the iniital states of cells
$g_{i,j}(X_{i,j}, t)$	the deterministic growth rate of the j^{th} cell in the i^{th} generation
$\sigma_{i,j}(X_{i,j}, t)$	the strength of noise in the growth of the j^{th} cell in the i^{th} generation
$\beta_{i,j}(X_{i,j}, t)$	the division rate of the j^{th} cell in the i^{th} generation
$\mu_{i,j}(X_{i,j}, t)$	the death rate of the j^{th} cell in the i^{th} generation
$\tilde{\beta}_{i,j}(X_{i,j}, X_1, X_2, t)$	the differential division rate of the j^{th} cell in the i^{th} generation dividing into two cells in the $(i+1)^{\text{th}}$ generation with states X_1, X_2
$\vec{X}_{\vec{n}_{\text{b},-i,-j}}$	the states of the cell population $\vec{X}_{\vec{n}}$ right after the j^{th} cell in the i^{th} generation divides: $\vec{X}_{\vec{n}_{\text{b},-i,-j}}$ differs from $\vec{X}_{\vec{n}^0}$ in that the state variables for the cells in the $(i-1)^{\text{th}}$ generation is $(X_{i-1,1}, ..., X_{i-1,j-1}, X_{i-1,j+1}, ...X_{i-1,n_i})$ and the state variables for the cells in the i^{th} generation are $(X_{i,1}, ..., X_{i,n_i}, X_1, X_2)$
$\vec{X}_{\vec{n}_{\text{d},-i,-j}}$	the states of the cell population $\vec{X}_{\vec{n}}$ right after the j^{th} cell in the i^{th} generation die: $\vec{X}_{\vec{n}_{\text{d},-i,-j}}$ differs from $\vec{X}_{\vec{n}}$ in that the state variables for the cells in the $(i-1)^{\text{th}}$ generation is $(X_{i-1,1}, ..., X_{i-1,j-1}, X_{i-1,j+1}, ...X_{i-1,n_i})$
$\vec{X}_{\vec{n}_{\text{b},i-1,j}}$	the pre-division cellular population: it differs from $\vec{X}_{\vec{n}}$ in that the state variables for the cells in the $(i-1)^{\text{th}}$ generation is $(X_{i-1,1}, ..., X_{i-1,j-1}, Y, X_{i-1,j}, ...)$ and the state variables for the cells in the i^{th} generation are $(X_{i,1}, ..., X_{i,n_i-2})$ (an additional cell with Y in the $(i-1)^{\text{th}}$ generation divides and gives birth to two new daughter cells X_{i,n_i-1}, X_{i,n_i} in the i^{th} generation)
$\vec{X}_{\vec{n}_{\text{d},i,j}}$	the pre-death cellular population: it differs from $\vec{X}_{\vec{n}}$ in that the state variables for the cells in the i^{th} generation is $(X_{i,1}, ..., X_{i,j-1}, Y, X_{i,j}, ...)$ (an additional cell in the i^{th} generation with Y dies)

Table 4.1: **Overview of variables.** A list of the main variables and parameters used in this chapter.

As discussed in [Gar09, CHS20], we can assume that the evolution of $X_{i,j}$ (the internal state of the j^{th} cell in the i^{th} generation) obeys the following law

$$\mathrm{d}X_{i,j}(t) = g_{i,j}(X_{i,j}, t)\mathrm{d}t + \sigma_{i,j}(X_{i,j}, t)\mathrm{d}(B_{i,j})_t \tag{4.2.1}$$

Here $\mathrm{d}(B_{i,j})_t$ are increments of independent Wiener processes for each i, j.We assume that both g_i and σ_i are Lipschitz continuous so the solution $X_{i,j}(t)$ of Eq. (4.2.1) exists and is unique almost surely given any initial condition $X_{i,j}(0)$. The evolution of $X_{i,j}$ is interrupted by the cell division, and we denote the cell division rate by $\beta_i(X_{i,j})$ for a cell in the i^{th} generation with its internal state being $X_{i,j}$. After division, a cell gives birth to two new daughter cells and we denote the birth rate of having two daughters with internal states

X_1, X_2 to be $\tilde{\beta}_{i,j}(X_{i,j}, X_1, X_2)$ with the constraint

$$\int \tilde{\beta}_{i,j}(X_{i,j}, X_1, X_2)\mathrm{d}X_1\mathrm{d}X_2 = \beta_{i,j}(X_{i,j}). \tag{4.2.2}$$

which implies that the rate of giving birth to new cells (the LHS) is equal to the division rate (the RHS).

We denote $p_{\vec{n}}(\vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}, 0)$ to be the probability density function that the population has $\vec{n}$ cells with the internal states $\vec{X}_{\vec{n}^0}$given the initial condition that the system has $\vec{n}^0$ cells with internal states $\vec{X}_{\vec{n}^0}$ at $t = 0$. Actually, $p_{\vec{n}}(\vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}, \vec{n}^0, 0)$ could be written as

$$\begin{aligned}
&p_{\vec{n}}(\vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}, 0) = \\
&\mathbb{E}\Big[\delta(\vec{X}_{\vec{n}}(t) - \vec{X}_{\vec{n}}) \exp\big(-\int_0^t \sum_{i=1}^{k^0}\sum_{j=1}^{n_i^0} \big(\beta(X_{i,j}(s)) + \mu(X_{i,j}(s))\big)\mathrm{d}s\big)\Big|\vec{X}_{\vec{n}^0}, \vec{n}(s) = n^0, s \in [0,t], 0\Big] \\
&\quad + \mathbb{E}\Big[\int_0^t \exp\Big(-\int_0^s \sum_{i=1}^{k^0}\sum_{j=1}^{n_i^0} \big(\beta_{i,j}(X_{i,j}(r)) + \mu_{i,j}(X_{i,j}(r))\big)\mathrm{d}r\Big) \\
&\qquad \times \Big[\sum_{i=1}^{k^0}\sum_{j=1}^{n_i^0} \big(\tilde{\beta}_{i,j}(X_{i,j}(s), X_1, X_2) p_{\vec{n}}(\vec{X}_{\vec{n}}, t-s|\vec{X}_{\vec{n}^0_{\mathrm{b},-i,-j}}(s), 0) \\
&\qquad\quad + \mu_{i,j}(X_{i,j}(t-s)) p_{\vec{n}}(\vec{X}_{\vec{n}}, t-s|\vec{X}_{\vec{n}^0_{\mathrm{d},-i,-j}}(s), 0)\big)\Big]\mathrm{d}s\Big|\vec{X}_{\vec{n}^0}, 0\Big], \text{ if } \vec{n} = \vec{n}^0, \\
&p_{\vec{n}}(\vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}, 0) = \mathbb{E}\Big[\int_0^t \exp\Big(\int_0^s -\sum_{i=1}^{k^0}\sum_{j=1}^{n_i^0} \big(\beta_{i,j}(X_{i,j}(r)) + \mu_{i,j}(X_{i,j}(r))\big)\mathrm{d}r\Big) \\
&\qquad \times \Big[\sum_{i=1}^{k^0}\sum_{j=1}^{n_i^0} \big(\tilde{\beta}_{i,j}(X_{i,j}(s), X_1, X_2) p_{\vec{n}}(\vec{X}_{\vec{n}}, t-s|\vec{X}_{\vec{n}^0_{\mathrm{b},-i,-j}}(s), 0) \\
&\qquad\quad + \mu_{i,j}(X_{i,j}(t-s)) p_{\vec{n}}(\vec{X}_{\vec{n}}, t-s|\vec{X}_{\vec{n}^0_{\mathrm{d},-i,-j}}(s), 0)\big)\Big]\mathrm{d}s\Big|\vec{X}_{\vec{n}^0}, 0\Big], \text{ if } \vec{n} \neq \vec{n}^0
\end{aligned} \tag{4.2.3}$$

where the meanings of notations $\vec{X}_{\vec{n}^0_{\mathrm{b},-i,-j}}(s)$ and $\vec{X}_{\vec{n}^0_{\mathrm{d},-i,-j}}$ are in Table 4.1.

The first term on the RHS of Eq. (4.2.3) is the probability that no division or death happens in the system during time $[0, t]$ and the final internal states of the cell population are $\vec{X}_{\vec{n}}$ and the second term on the RHS of Eq. (4.2.3) denotes the probability that at least

one division or death happens within $[0,t]$ and the final internal states of the cell population are $\vec{X}_{\vec{n}}$.We shall show that under certain conditions the differential equation satisfied by $p_{\vec{n}}(\vec{X}_{\vec{n}},t|\vec{X}_{\vec{n}^0},0)$ is

$$\begin{aligned}
&\frac{\partial p_{\vec{n}}}{\partial t}+\sum_{i=1}^{k}\sum_{j=1}^{n_i}\frac{\partial(g_{i,j}p_{\vec{n}})}{\partial X_{i,j}}-\frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{n_i}\frac{\partial^2(\sigma_{i,j}p_{\vec{n}})}{(\partial X_{i,j})^2}+\sum_{i=1}^{k}\sum_{j=1}^{n_i}\big(\beta_{i,j}(X_{i,j})+\mu_{i,j}(X_{i,j})\big)p_{\vec{n}}\\
&\quad=\sum_{i=2}^{k}\sum_{j=1}^{n_{i-1}+1}\int\tilde{\beta}(Y,X_{i,n_i-1},X_{i,n_i})p_{\vec{n}_{\mathrm{b},i-1,j}}(\vec{X}_{\vec{n}_{\mathrm{b},i-1,j}},t|\vec{X}_{\vec{n}^0}(0),0)\mathrm{d}Y\\
&\qquad+\sum_{i=1}^{\infty}\sum_{j=1}^{n_i+1}\int\mu(Y)p_{\vec{n}_{\mathrm{d},i,j}}(\vec{X}_{\vec{n}_{\mathrm{d},i,j}},t|\vec{X}_{\vec{n}^0}(0),0)\mathrm{d}Y
\end{aligned}\tag{4.2.4}$$

In Eq. (4.2.4), the notations $\vec{X}_{\vec{n}_{\mathrm{b},i-1,j}}$, the pre-division cell population, and $\vec{X}_{\vec{n}_{\mathrm{d},i,j}}$, the pre-death cell population, are defined in Table 4.1. Next, we need the following two propositions to show that $p_{\vec{n}}(\vec{X}_{\vec{n}},t|\vec{X}_{\vec{n}^0},0)$ satisfies Eq. (4.2.4).

Proposition 1. (Forward-type Feynman-Kac formula) If the coefficients $g_{i,j},\sigma_{i,j},\beta_{i,j},\mu_{i,j}$ are smooth, uniform Lipschitz continuous, and uniform bounded, then under certain assumptions the solution to the following PDE

$$\begin{aligned}
&\frac{\partial\hat{p}_{\vec{n}}}{\partial t}(\vec{X}_{\vec{n}},t|\vec{X}_{\vec{n}^0}(0),0)+\sum_{i=1}^{k}\sum_{j=1}^{i}\frac{\partial(g_{i,j}(X_{i,j},s)\hat{p}_{\vec{n}})}{\partial X_{i,j}}-\frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{n_i}\frac{\partial^2(\sigma_{i,j}^2(X_{i,j},s)\hat{p}_{\vec{n}})}{(\partial X_{i,j})^2}\\
&\qquad=-\sum_{i=1}^{k}\sum_{j=1}^{n_i}\big(\beta_{i,j}(X_{i,j})+\mu_{i,j}(X_{i,j})\big)\hat{p}_{\vec{n}},\\
&\hat{p}_{\vec{n}}(\vec{X}_{\vec{n}},0|\vec{X}_{\vec{n}^0}(0),0)=\delta(\vec{X}_{\vec{n}^0}(0)-\vec{X}_{\vec{n}})\text{ if }\vec{n}=\vec{n}^0\text{ and }\hat{p}_{\vec{n}}(\vec{X}_{\vec{n}},0)=0\text{ if }\vec{n}\neq\vec{n}^0
\end{aligned}\tag{4.2.5}$$

is

$$\begin{aligned}
\hat{p}_{\vec{n}}(\vec{X}_{\vec{n}},t|\vec{X}_{\vec{n}^0}(0),0)\coloneqq\mathbb{E}\Bigg[&\delta(\vec{X}_{\vec{n}}(t)-\vec{X}_{\vec{n}})\\
&\times\exp\Big(-\int_0^t\sum_{i=1}^{k}\sum_{j=1}^{n_i}\big(\beta_{i,j}(X_{i,j})+\mu_{i,j}(X_{i,j})\big)\mathrm{d}s\Big)\Big|\vec{n}(s)=\vec{n}^0,s\in[0,t],\vec{X}_{\vec{n}^0}(0),0\Bigg]
\end{aligned}\tag{4.2.6}$$

where $\vec{n}(s)$ is the vector consisting of numbers of cells in each generation at time s, and each

component in $\vec{X}_{\vec{n}}(t)$ satisfies the following SDE

$$X_{i,j}(t) = X_{i,j}(0) + \int_0^t g_{i,j}(X_{i,j}(s), s)\mathrm{d}s + \int_0^t \sigma_{i,j}(X_{i,j}(s), s)\mathrm{d}W_{i,j,s}. \tag{4.2.7}$$

Proposition 1 reveals the partial differential equation satisfied by the probability density of all cells with states $\vec{X}_{\vec{n}_k}$ if not division or death occurs. We shall delay the proof of and the specific mathematical assumptions needed for Prop. 1 in Appendix A.3.1.

When cell division or death happens, the number of total cells in the cell population change. Cell divisions and deaths could be described by a Markov jump process. We need the following proposition to derive the differential equation satisfied by the conditional probability density function $p_{\vec{n}}(\vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}(0), \vec{n}^0, 0)$ defined in Eq. (4.2.3).

Proposition 2. (Markov jump process) Given the initial condition $\vec{n}^0$ with internal states $\vec{X}_{\vec{n}^0}(0)$ at $t = 0$ and a target state at time t with $\vec{n}$ cells and their internal states $\vec{X}_{\vec{n}}$, we let

$$\begin{aligned} &p^0_{\vec{n}}(\vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}(0), 0) = 0, \\ &p^1_{\vec{n}}(\vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}(0), 0) = \hat{p}_{\vec{n}}(\vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}, 0), \end{aligned} \tag{4.2.8}$$

and we recursively define

$$\begin{aligned} p^{m+1}_{\vec{n}}(\vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}(0), 0) &= \hat{p}_{\vec{n}}(\vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}(0), 0) \\ &+ \mathbb{E}\Big[\int_0^t \exp\Big(-\int_0^s \sum_{i=1}^{k^0}\sum_{j=1}^{n_i^0}\big((\beta_{i,j}(X_{i,j}(r)) + \mu_{i,j}(X_{i,j}(r))\big)\mathrm{d}r\Big) \\ &\quad \times \Big[\sum_{i=1}^{k^0}\sum_{j=1}^{n_i^0}\big(\tilde{\beta}_{i,j}(X_{i,j}(s), X_1, X_2)p^m_{\vec{n}}(\vec{X}_{\vec{n}}, t-s|\vec{X}_{\vec{n}^0_{\mathrm{b},-i,-j}}(s), 0) \\ &\quad + \mu_{i,j}(X_{i,j}(s))p^m_{\vec{n}}(\vec{X}_{\vec{n}}, t-s|\vec{X}_{\vec{n}^0_{\mathrm{d},-i,-j}}(s), 0)\big)\Big]\mathrm{d}s\Big|\vec{X}_{\vec{n}^0}(0), 0\Big]. \end{aligned} \tag{4.2.9}$$

Then, $p^{m+1}_{\vec{n}}$ satisfies the following differential equation

$$\frac{\partial p_{\vec{n}}^{m+1}}{\partial t}+\sum_{i=1}^{k}\sum_{j=1}^{n_i}\frac{\partial(g_{i,j}p_{\vec{n}}^{m+1})}{\partial X_{i,j}}-\frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{n_i}\frac{\partial^2(\sigma_{i,j}^2 p_{\vec{n}}^{m+1})}{(\partial X_{i,j})^2}+\sum_{i=1}^{k}\sum_{j=1}^{n_i}\big(\beta_{i,j}(X_{i,j})+\mu_{i,j}(X_{i,j})\big)p_{\vec{n}}^{m+1}$$
$$=\sum_{i=1}^{k-1}\sum_{j=1}^{n_{i-1}^{\mathrm{b}}}\int\tilde{\beta}(Y,X_{i+1,n_{i+1}-1},X_{i+1,n_{i+1}})p_{\vec{n}_{\mathrm{b},i,j}}^{m}(\vec{X}_{\vec{n}_{\mathrm{b},i,j}},t|\vec{X}_{\vec{n}^0}(0),0)\mathrm{d}Y$$
$$+\sum_{i=1}^{\infty}\sum_{j=1}^{n_i^{\mathrm{d}}}\int\mu(Y)p_{\vec{n}_{\mathrm{d},i,j}}^{m}(\vec{X}_{\vec{n}_{\mathrm{d},i,j}},t|\vec{X}_{\vec{n}^0}(0),0)\mathrm{d}Y. \tag{4.2.10}$$

Furthermore, $p_{\vec{n}}^m$ is increasing in m.

We shall delay the proof of Proposition 2 in Appendix A.3.2. From the increasing property of $p_{\vec{n}}^m$ in m, there exists a p^* such that $p^m \to p^*$ a.s. for all $\vec{X}_{\vec{n}^0}(0)$ and $\vec{X}_{\vec{n}}$. Furthermore, by induction on m, after integrating over $\vec{X}_{\vec{n}}$ and summing over all $\vec{n}$ on both sides of the recursive definition Eq. (4.2.9), we can further show that if for $m \in \mathbb{N}^+$

$$\sum_{\vec{n}}\int p_{\vec{n}}^{m-1}(\vec{X}_{\vec{n}},t|\vec{X}_{\vec{n}^0}(0),0)\mathrm{d}\vec{X}_{\vec{n}} \le 1 \tag{4.2.11}$$

for any $\vec{X}_{\vec{n}^0}(0)$, then we have

$$\sum_{\vec{n}}\int p_{\vec{n}}^{m}(\vec{X}_{\vec{n}},t|\vec{X}_{\vec{n}^0}(0),0)\mathrm{d}\vec{X}_{\vec{n}} \le \int\hat{p}_{\vec{n}^0}(\vec{Y}_{\vec{n}^0},t|\vec{X}_{\vec{n}^0}(0),0)\mathrm{d}\vec{Y}_{\vec{n}^0}$$
$$+\mathbb{E}\bigg[\int_0^t\exp\Big(-\int_0^s\sum_{i=1}^{k^0}\sum_{j=1}^{n_i^0}\big((\beta(X_{i,j}(r))+\mu(X_{i,j}(r))\mathrm{d}r\Big)\bigg. \tag{4.2.12}$$
$$\times\bigg[\sum_{i=1}^{k^0}\sum_{j=1}^{n_i^0}\big(\beta(X_{i,j}(s))+\mu(X_{i,j}(s))\big)\bigg]\mathrm{d}s\Big|\vec{X}_{\vec{n}^0}(0),0\bigg] := F^m(t;\vec{X}_{\vec{n}^0}(0),0)$$

for any $\vec{X}_{\vec{n}^0}(0)$, and

$$\frac{\mathrm{d}F^m(t;\vec{X}_{\vec{n}^0}(0),0)}{\mathrm{d}t}=0,\quad F^m(0;\vec{X}_{\vec{n}^0}(0),0)=1. \tag{4.2.13}$$

Therefore, we have

$$\sum_{\vec{n}} \int p_{\vec{n}}^{m}(\vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}(0), 0)\mathrm{d}\vec{X}_{\vec{n}} \leq 1 \tag{4.2.14}$$

for all $m \in \mathbb{N}$. Finally, it is easy to show that $p_{\vec{n}}^{m}(\vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}(0), 0) \geq 0$, so $0 \leq p^* < \infty$ exists a.e.. We assume that the convergence $p^m \to p^*$ is uniform and we also assume that taking the limit w.r.t. m is interchangeable with taking the partial derivatives in Eq. (4.2.10), therefore, p^* is the solution to

$$\begin{aligned}
&\frac{\partial p_{\vec{n}}^*}{\partial t} + \sum_{i=1}^{k}\sum_{j=1}^{n_i} \frac{\partial(g_{i,j}p_{\vec{n}}^*)}{\partial X_{i,j}} - \frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{n_i} \frac{\partial^2(\sigma_{i,j}p_{\vec{n}}^*)}{(\partial X_{i,j})^2} + \sum_{i=1}^{k}\sum_{j=1}^{n_i} \big(\beta(X_{i,j}) + \mu(X_{i,j})\big)p_{\vec{n}}^* = \\
&\quad \sum_{i=1}^{k-1}\sum_{j=1}^{n_i+1} \int \tilde{\beta}_{i,j}(Y, X_{i+1,n_{i+1}-1}, X_{i+1,n_{i+1}})p_{\vec{n}_{\mathrm{b},i,j}}^*(\vec{X}_{\vec{n}_{\mathrm{b},i,j}}, t|\vec{X}_{\vec{n}^0}(0), 0)\mathrm{d}Y \\
&\qquad + \sum_{i=1}^{\infty}\sum_{j=1}^{n_i+1} \int \mu_{i,j}(Y)p_{\vec{n}_{\mathrm{d},i,j}}^*(\vec{X}_{\vec{n}_{\mathrm{d},i,j}}, t|\vec{X}_{\vec{n}^0}(0), 0)\mathrm{d}Y
\end{aligned} \tag{4.2.15}$$

Since p^* can also be written as

$$\begin{aligned}
&p_{\vec{n}}^*(\vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}(0), 0) = \hat{p}_{\vec{n}}(\vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}(0), 0) \\
&\quad + \mathbb{E}\Bigg[\int_0^t \exp\Big(- \int_0^s \sum_{i=1}^{k}\sum_{j=1}^{n_i^0} \big((\beta_{i,j}(X_{i,j}) + \mu_{i,j}(X_{i,j})\big)\mathrm{d}r\Big) \\
&\qquad \times \Big[\sum_{i=1}^{k^0}\sum_{j=1}^{n_i^0} \Big(\int \tilde{\beta}_{i,j}(X_{i,j}(s), X_1, X_2)p_{\vec{n}}^*(\vec{X}_{\vec{n}}, t-s|\vec{X}_{\vec{n}^0_{\mathrm{b},-i,-j}}(s), 0)\mathrm{d}X_1\mathrm{d}X_2 \\
&\qquad\quad + \mu_{i,j}(X_{i,j}(s))p_{\vec{n}}^*(\vec{X}_{\vec{n}}, t-s|\vec{X}_{\vec{n}^0_{\mathrm{d},-i,-j}}(s), 0)\Big)\Big]\mathrm{d}s\Big|\vec{X}_{\vec{n}^0}(0), 0\Bigg]
\end{aligned} \tag{4.2.16}$$

we have shown Eq. (4.2.16) solves the differential equation Eq. (4.2.4). Finally, we assume the normalization condition

$$\sum_{\vec{n}} \int p_{\vec{n}}^*(\vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}(0), 0)\mathrm{d}\vec{X}_{\vec{n}} = 1 \tag{4.2.17}$$

holds for every $\vec{X}_{\vec{n}^0}(0)$. By averaging over the initial distribution of $\vec{X}_{\vec{n}^0}(0)$ denoted by

$p^0_{\vec{n}^0}(\vec{X}_{\vec{n}^0}(0),0)$, we have

$$p^*_{\vec{n}}(\vec{X}_{\vec{n}},t) \coloneqq \sum_{\vec{n}^0}\int_{\vec{X}_{\vec{n}^0}} p_{\vec{n}}(\vec{X}_{\vec{n}},t|\vec{X}_{\vec{n}^0}(0),0)p^0_{\vec{n}^0}(\vec{X}_{\vec{n}^0},0)\mathrm{d}\vec{X}_{\vec{n}^0} \tag{4.2.18}$$

is an unconditional probability density distribution that solves Eq. (4.2.4).

Next, we define the symmetric probability density distribution

$$\rho_{\vec{n}}(\vec{X}_{\vec{n}},t) \coloneqq \Pi^k_{i=1}\frac{1}{n_i!}\sum_{\pi} p^*_{\vec{n}}(\pi(\vec{X}_{\vec{n}}),t) \tag{4.2.19}$$

where $p^*_{\vec{n}}$ is defined in Eq. (4.2.18) and $\pi(\vec{X}_{\vec{n}})$ is a rearrangement that changes the sequence of the state variables $X_{i,j}$ of cells within the same generation and thus the summation is taken over all such rearrangements ($\Pi^k_{i=1}n_i!$ rearrangements in total). If

$$g_{i,j}=g_i,\sigma_{i,j}=\sigma_i,\beta_{i,j}=\beta_i,\mu_{i,j}=\mu_i,\tilde{\beta}_{i,j}=\tilde{\beta}_i, \tag{4.2.20}$$

i.e., all coefficients only depend on the generation i^{th}, then partial differential equation satisfied by $\rho_{\vec{n}}(\vec{X}_{\vec{n}},t)$ in Eq. (4.2.19) is

$$\begin{aligned}\frac{\partial\rho_{\vec{n}}}{\partial t}+\sum_{i=1}^{k}\sum_{j=1}^{n_i}\frac{\partial(g_i\rho_{\vec{n}})}{\partial X_{i,j}}-\frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{n_i}\frac{\partial^2(\sigma_{i,j}\rho_{\vec{n}})}{(\partial X_{i,j})^2}&=-\sum_{i=1}^{k}\sum_{j=1}^{n_i}\big(\beta_{i,j}(X_{i,j})+\mu_{i,j}(X_{i,j})\big)\rho_{\vec{n}}\\
+\sum_{i=1}^{k-1}\frac{n_i+1}{n_{i+1}(n_{i+1}-1)}\sum_{1\le j_1\ne j_2\le n_{i+1}}&\int\tilde{\beta}_i(Y,X_{i+1,j_1},X_{i+1,j_2})\rho_{\vec{n}_{\mathrm{b},i}}(\vec{X}_{\vec{n}_{\mathrm{b},i,j_1,j_2}},t)\mathrm{d}Y\\
+\sum_{i=1}^{\infty}\sum_{j=1}^{n_i+1}&\int\mu_i(Y)\rho_{\vec{n}_{\mathrm{d},i}}(\vec{X}_{\vec{n}_{\mathrm{d},i,j}},t)\mathrm{d}Y,\end{aligned} \tag{4.2.21}$$

where $\vec{X}_{\vec{n}_{\mathrm{b},i,j_1,j_2}}$ differs from $\vec{X}_{\vec{n}}$ in that the state variables for the cells in the i^{th} generation are $(Y,X_{i,1},...X_{i,n_i})$ and the state variables for cells in the $(i+1)^{\text{th}}$ generation misses X_{i+1,j_1} and X_{i+1,j_2}.

Finally, in many biological models, the state variable could be a multi-dimensional vector instead of a scalar, *i.e.*, $\boldsymbol{X}_{i,j}\coloneqq(X_{i,j,1},...,X_{i,j,d})\in\mathbb{R}^h$ for the j^{th} cell in the i^{th} generation in

a cell population and we assume that the evolution of $\boldsymbol{X}_{i,j}$ to obey the following SDE

$$\mathrm{d}\boldsymbol{X}_{i,j} = \gamma_i(\boldsymbol{X}_{i,j}, t)\mathrm{d}t + \boldsymbol{\Sigma}_i(\boldsymbol{X}_{i,j}, t)\mathrm{d}(\boldsymbol{W}_{i,j})_t \tag{4.2.22}$$

where $(\boldsymbol{W}_{i,j})_t$ are independent h_0-dimensional Wiener process and the coefficients $\gamma_i(\boldsymbol{X}_{i,j}, t) \coloneqq (g_{i,1}(\boldsymbol{X}_{i,j}, t), ..., g_{i,h}(\boldsymbol{X}_{i,j}, t)) : \mathbb{R}^h \times \mathbb{R}^+ \to \mathbb{R}^h, \boldsymbol{\Sigma}_i \coloneqq (\sigma_i(\boldsymbol{X}_{i,j}, t))_{mn} : \mathbb{R}^h \times \mathbb{R}^+ \to \mathbb{R}^{h\times h_0}, m = 1, ..., h, n = 1, ..., h_0$ are all smooth, uniform Lipschitz continuous, and uniform bounded. We can also define the symmetric probability density distribution $\rho_{\vec{n}}(\vec{\boldsymbol{X}}_{\vec{n}}, t)$ as in Eqs. (4.2.19) and after applying the multi-dimensional forward Feynman-Kac equation case in [LOR15] we can show that the differential equation satisfied by such $\rho_{\vec{n}}$ is

$$\begin{aligned}
&\frac{\partial \rho_{\vec{n}}}{\partial t} + \sum_{i=1}^{k}\sum_{j=1}^{n_i}\sum_{s=1}^{h}\frac{\partial(g_{i,s}\rho_{\vec{n}})}{\partial X_{i,j,s}} - \frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{n_i}\sum_{1\le s_1,s_2\le h}\frac{\partial^2(\sum_{\ell=1}^{h}(\Sigma_i)_{s_1,\ell}(\Sigma_i)_{s_2,\ell}\rho_{\vec{n}})}{(\partial X_{i,j,s_1}\partial X_{i,j,s_2})} \\
&\quad = -\sum_{i=1}^{k}\sum_{j=1}^{n_i}\big(\beta_i(\boldsymbol{X}_{i,j}) + \mu_i(\boldsymbol{X}_{i,j})\big)\rho_{\vec{n}} \\
&\qquad + \sum_{i=1}^{k-1}\frac{n_i+1}{n_{i+1}(n_{i+1}-1)}\sum_{1\le j_1\ne j_2\le n_{i+1}}\int \tilde{\beta}(\boldsymbol{Y}, \boldsymbol{X}_{i+1,j_1}, \boldsymbol{X}_{i+1,j_2})\rho_{\vec{n}_{\mathrm{b},i}}(\vec{\boldsymbol{X}}_{\vec{n}_{\mathrm{b},i,j}}, t)\mathrm{d}\boldsymbol{Y} \\
&\qquad + \sum_{i=1}^{\infty}\sum_{j=1}^{n_i+1}\int \mu_i(\boldsymbol{Y})\rho_{\vec{n}_{\mathrm{d},i}}(\vec{\boldsymbol{X}}_{\vec{n}_{\mathrm{d},i,j}}, t)\mathrm{d}\boldsymbol{Y}
\end{aligned} \tag{4.2.23}$$

if the coefficients $\beta_{i,j} = \beta_i, \mu_{i,j} = \mu_i, \tilde{\beta}_{i,j} = \tilde{\beta}_i$ are homogeneous for cells in the same generation.

4.3 Mass-action differential equations

Through marginalization of the kinetic equation Eq. (4.2.21) that describes the evolution of population dynamics, we could derive the differential equations that describe the evolution of certain macro quantities such as the total protein and mRNA amount which are of biological and experimental interest. In this section, we shall investigate the governing equations for some macroscopic quantities by marginalizing Eq. (4.2.21) both analytically and numerically.

4.3.1 Evolution of population density

First, we track the cell densities in any given generation and any cellular internal state of interest by defining the marginalized cell density

$$v_{\vec{n}}(\vec{X}_{\vec{n}}, t) = \sum_{\vec{m} \geq \vec{n}} \Pi_{\ell=1}^{\infty} (m_\ell)_{n_\ell} \int_{\vec{X}_{\vec{m}\setminus\vec{n}}} \rho_{\vec{m}}(\vec{X}_{\vec{m}}, t) \mathrm{d}\vec{X}_{\vec{m}\setminus\vec{n}}, \tag{4.3.1}$$

where $\vec{m} \geq \vec{n}$ means that for each component in $\vec{m} \coloneqq (m_1, ..., m_\ell), m_\ell \geq n_\ell$. $(m_\ell)_{n_\ell} \coloneqq m_\ell(m_\ell - 1)...(m_\ell - n_\ell + 1)$ is the falling factorial. The integration is taken over the remaining variables of $\vec{X}_{\vec{m}}$ excluding $\vec{X}_{\vec{n}}$.

We can derive the differential equation satisfied by $v_{\vec{n}}(\vec{X}_{\vec{n}}, t)$

$$\begin{aligned} \frac{\partial v_{\vec{n}}}{\partial t} + \sum_{i=1}^{k} \sum_{j=1}^{n_i} \frac{\partial (g_{i,j} v_{\vec{n}})}{\partial X_{i,j}} - \frac{1}{2} \sum_{i=1}^{k} \sum_{j=1}^{n_i} \frac{\partial^2 (\sigma_{i,j} v_{\vec{n}})}{(\partial X_{i,j})^2} &= - \sum_{i=1}^{k} \sum_{j=1}^{n_i} \big(\beta(X_{i,j}) + \mu(X_{i,j})\big) v_{\vec{n}} \\ + \sum_{i=1}^{k-1} \sum_{j_1 \neq j_2} \int \tilde{\beta}_{i,j}(Y, X_{i+1,j_1}, X_{i+1,j_2}) v_{\vec{n}_{\mathrm{b},i}}(\vec{X}_{\vec{n}_{\mathrm{b},i,j_1,j_2}}, t) \mathrm{d}Y & \\ + \sum_{i=1}^{k-1} \sum_{j=1}^{n_{i+1}} \int \big(\tilde{\beta}_{i,j}(Y, X_{i+1,j}, Z) + \tilde{\beta}_{i,j}(Y, Z, X_{i+1,j})\big) v_{\vec{n}_{\mathrm{b},i}}(\vec{X}_{\vec{n}_{\mathrm{b},i,j}}, t) \mathrm{d}Y \mathrm{d}Z & \end{aligned} \tag{4.3.2}$$

where $\vec{X}_{\vec{n}_{\mathrm{b},i-1,j_1,j_2}}$ differs from $\vec{X}_{\vec{n}}$ in that its i^{th} generation is $(X_{i-1,1}, ..., X_{i-1,n_i}, Y)$ and its $(i+1)^{\text{th}}$ generation does not have the j_1^{th} and j_2^{th} components; $\vec{X}_{\vec{n}_{\mathrm{b},i,j}}$ differs from $\vec{X}_{\vec{n}}$ in that its i^{th} generation is $(X_{i,1}, ..., X_{i,n_i}, Y)$ and its $(i+1)^{\text{th}}$ generation does not have the j^{th} component.

As a special example, $u_{\vec{n}_i}(X, t), \vec{n}_i \coloneqq (0, .., 0, 1) \in \mathbb{R}^i$ tracks the cellular density in the i^{th} generation with respect to the structured variable x over time, and thus $\{u_{\vec{n}_i}(X, t)\}_{i=1}^{\infty}$ could show how the cellular population density evolves over generation through division and differentiation. For instance, we consider the following example in [CHS22] where the coefficients in Eq. (4.3.2)

$$g_{i,j}(x, t) = -x, \quad \sigma_{i,j}^2(x, t) = \exp(-x^2). \tag{4.3.3}$$

In this case, if the cells do not divide or die (*i.e.*, stay at the 1^{textst} generation), then the structured population will converge to the equilibrium

$$v_1(x,t) \to P(x) = \mathcal{N}\exp(2x^2 - \frac{1}{2}\exp(2x^2)),\ t \to \infty \tag{4.3.4}$$

where $\mathcal{N}$ is a normalization constant. Here, we set the division rates

$$\beta_i = \frac{1}{2}, \quad \mu_i = \frac{i-1}{2i}, \quad \tilde{\beta}_i(x,y,t) = \frac{1}{\sqrt{2\pi}}\exp(-\frac{(y-x)^2}{2}). \tag{4.3.5}$$

We set the initial condition to be $u_i(x,t) = \mathbb{1}_{i,1}\frac{1}{100}\mathbb{1}_{-2.5\le x\le 2.5}$ and plot the scaled cellular density

$$\frac{v_i(x,t)/P(x)}{\int_{-\infty}^{\infty} v_i(x,t)\mathrm{d}x} \tag{4.3.6}$$

in the first 10 generations at $t = 2$.

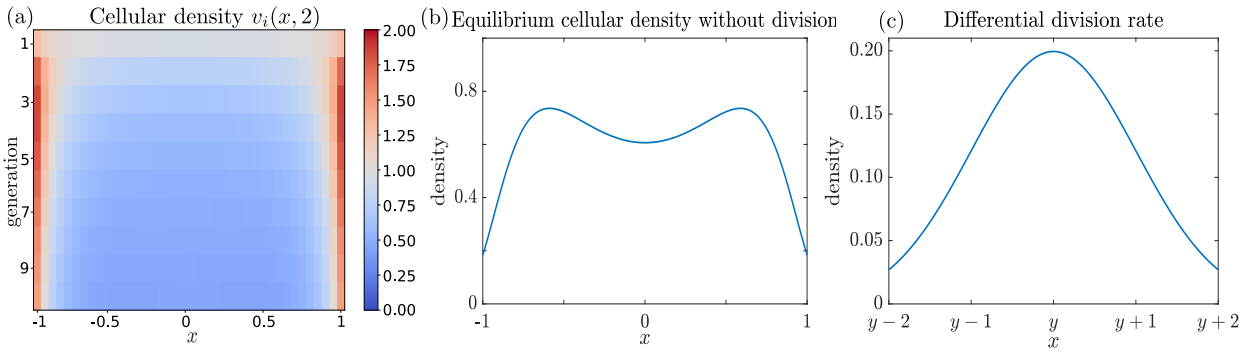

Figure 4.1: (a) The cellular density plot across different generations. It can be observed that the differentiation process prevents the population from reaching the equilibrium ($i \ge 2$) even when the death rate and division rate are irrelevant to x. However, as time increases when no incoming cells are entering a certain generation (such as $i = 1$), the structured population gradually returns to equilibrium. (b) The equilibrium cellular density without division. (c) The differential birth rate $\tilde{\beta}(y,x,t)$.

Specifically, if the coefficients $g, \sigma, \beta, \tilde{\beta}$ only depend on the cellular internal state X and time t, we can ignore the generation number by defining

$$\hat{\rho}_n(\vec{X}_n, t) := \sum_{\sum n_i = n} \frac{1}{n!}\sum_{\pi} p^*_{\vec{n}}(\pi(\vec{X}_{\vec{n}}), t). \tag{4.3.7}$$

where the summation over π is over all possible rearrangements of $(X_1, ..., X_n)$ to be the symmetrized probability density function that has a system of n cells with each cell's state being $X_1, ..., X_n$. $p^*_{\vec{n}}$ is defined in Eq. (4.2.18). It can be shown that the differential equation satisfied by $\hat{\rho}_n$ is

$$\begin{aligned}\frac{\partial \hat{\rho}_n}{\partial t} + \sum_{j=1}^{n} \frac{\partial (g\hat{\rho}_n)}{\partial X_j} - \frac{1}{2}\sum_{j=1}^{n} \frac{\partial^2 (\sigma^2 \hat{\rho}_n)}{(\partial X_j)^2} = -\sum_{j=1}^{n} \big(\beta(X_j) + \mu(X_j)\big)\hat{\rho}_n \\ + \frac{1}{n}\sum_{j_1 \neq j_2} \int \tilde{\beta}(Y, X_{j_1}, X_{j_2}) \hat{\rho}_{n-1}(\vec{X}_{n_{\mathrm{b},j_1,j_2}}, t)\mathrm{d}Y + (n+1)\int \mu(Y)\hat{\rho}_{n+1}(\vec{X}_{n_{\mathrm{d}}}, t)\mathrm{d}Y.\end{aligned} \tag{4.3.8}$$

Here, $\vec{X}_{n_{\mathrm{b},j_1,j_2}}$ is different from $\vec{X}_n$ in that it does not have X_{j_1}, X_{j_2} but has an extra Y; $\vec{X}_{n_{\mathrm{d}}}$ is dffierent from $\vec{X}$ in that it has an extra Y component. In this case, we could define the generation-irrelevant marginalized cell density

$$v_n(X_1, ..., X_n) = \sum_{m \geq n} (m)_n \int \hat{\rho}_m(\vec{X}_m, t)\mathrm{d}\vec{X}_m \backslash \vec{X}_n. \tag{4.3.9}$$

The differential equation satisfied by such $v_n(X_1, X_2, ..., X_n)$ is

$$\begin{aligned}\frac{\partial v_n}{\partial t} + \sum_{j=1}^{n} \frac{\partial (g v_n)}{\partial X_j} - \frac{1}{2}\sum_{j=1}^{n} \frac{\partial^2 (\sigma^2 v_n)}{(\partial X_j)^2} = -\sum_{j=1}^{n} \big(\beta(X_j) + \mu(X_j)\big) v_n \\ + \sum_{j_1 \neq j_2} \int \tilde{\beta}(Y, X_{j_1}, X_{j_2}) v_{n-1_{\mathrm{b},j_1,j_2}}(\vec{X}_{n_{\mathrm{b},j_1,j_2}}, t)\mathrm{d}Y \\ + \sum_{j=1}^{n} \int \big(\tilde{\beta}(Y, X_j, Z) + \tilde{\beta}(Y, Z, X_j)\big) v_{n_{\mathrm{b},j}}(\vec{X}_{n-1_{\mathrm{b},j}}, t)\mathrm{d}Y\mathrm{d}Z.\end{aligned} \tag{4.3.10}$$

Here, $\vec{X}_{n_{\mathrm{b},j}}$ is different from $\vec{X}_n$ in that it does not have X_j but has an extra Y. If we take $n = 1$, we can obtain a closed-form PDE for describing the cell density w.r.t. the scalar state variable X

$$\begin{aligned}\frac{\partial v_1}{\partial t} + \frac{\partial (g v_1)}{\partial X} - \frac{1}{2}\frac{\partial^2 (\sigma_j^2 v_1)}{(\partial X)^2} = -\big(\beta(X) + \mu(X)\big) v_1 \\ + \int \big(\tilde{\beta}(Y, X, Z) + \tilde{\beta}(Y, Z, X)\big) v_1(X, t)\mathrm{d}Y\mathrm{d}Z.\end{aligned} \tag{4.3.11}$$

Note that if we integrate over the age variable for the population density of the kinetic

adder-sizer model in [XC21], we will get v_1 in Eq. (4.3.11).

Finally, if we assume that all coefficients are constant, we can marginalize over the state variables. More specifically, if we define the generation vector $\vec{\boldsymbol{i}} \coloneqq (i_1, ..., i_k), 0 < i_1 < ... < i_k$ and the associated orders of moments $\vec{\boldsymbol{\ell}} \coloneqq (\ell_1, ..., \ell_k), \ell_s > 0$, then we can track the expectation of the product of different orders of the number of cells in different generations

$$\mathbb{E}[\Pi_{s=1}^{k} n_{i_s}^{\ell_s}] \coloneqq \sum_{\vec{n}} \Pi_{s=1}^{k} n_{i_s}^{\ell_s} \int \rho_{\vec{n}}(\vec{X}_{\vec{n}}, t) \mathrm{d}\vec{X}_{\vec{n}}. \tag{4.3.12}$$

The differential equation satisfied by $\mathbb{E}[\Pi_{i=1}^{k} n_i^{\ell_i}]$ can be shown to be

$$\begin{aligned}
\frac{\mathrm{d}\mathbb{E}[\Pi_{s=1}^{k} n_{i_s}^{\ell_s}]}{\mathrm{d}t} &= \sum_{r=1}^{k} \beta_{i_r - 1} \Big(\mathbb{E}[\Pi_{s=1}^{k} (n_{i_s} - \delta_{i_r - 1, i_s} + 2\delta_{i_r, i_s})^{\ell_s} \cdot n_{i_r - 1}] - \mathbb{E}[\Pi_{s=1}^{k} n_{i_s}^{\ell_s} \cdot n_{i_r - 1}] \Big) \\
&+ \sum_{r=1}^{k} \beta_{i_r} \Big(\mathbb{E}[\Pi_{s=1}^{k} (n_{i_s} - \delta_{i_r, i_s} + 2\delta_{i_r + 1, i_s})^{\ell_s} \cdot n_{i_r}] - \mathbb{E}[\Pi_{s=1}^{k} n_{i_s}^{\ell_i} \cdot n_{i_r}] \Big) \\
&- \sum_{r=1}^{k-1} \beta_{i_r} \Big(\mathbb{I}_{\{i_{r+1} - i_r = 1\}} \cdot \big(\mathbb{E}[\Pi_{s=1}^{k} (n_{i_s} - \delta_{i_r, i_s} + 2\delta_{i_r + 1, i_s})^{\ell_s} \cdot n_{i_r}] - \mathbb{E}[\Pi_{s=1}^{k} n_{i_s}^{\ell_s} \cdot n_{i_r}] \big) \Big) \\
&- \sum_{r=1}^{\infty} \mu_{i_r} \Big(\mathbb{E}[\Pi_{s=1}^{k} n_{i_s}^{\ell_s} \cdot n_{i_r}] - \mathbb{E}[\Pi_{s=1}^{k} (n_{i_s} - \delta_{i_s, i_r})^{\ell_s} \cdot n_{i_r}] \Big).
\end{aligned} \tag{4.3.13}$$

where $\delta_{i,r} = 1$ if $i = r$ and $\delta_{i,r} = 0$ otherwise.

Remark: Note that if $\vec{\boldsymbol{i}} = (i)$ is one-dimensional, and $\vec{\boldsymbol{\ell}} = (1)$, then Eq. (4.3.13) reduces to the evolution of the average cell number in the i^{th} generation

$$\frac{\mathrm{d}\mathbb{E}[n_i]}{\mathrm{d}t} = 2\beta_{i-1}\mathbb{E}[n_{i-1}] - \beta_i \mathbb{E}[n_i] - \mu_i \mathbb{E}[n_i]. \tag{4.3.14}$$

4.3.2 Evolution of total biomass

One macro quantity of specific interest is the total biomass (or total protein or mRNA amount). For example, one may also be interested in tracking the expectation of the total

cell's volume $\langle \sum_{j=1}^{n_i} X_{i,j}$ in the $i^{\text{th}} \rangle$ generation, which is denoted by

$$X_i(t) \coloneqq \langle \sum_{j=1}^{n_i} X_{i,j}(t) \rangle = \sum_{\vec{n}} \int (\sum_{j=1}^{n_i} X_{i,j}) \cdot \rho_{\vec{n}}(\vec{X}_{\vec{n}}, t) \mathrm{d}\vec{X}_{\vec{n}} \tag{4.3.15}$$

where $\rho_{\vec{n}}(\vec{X}_{\vec{n}}, t)$ is defined in Eq. (4.2.21). The differential equation satisfied by $X_i(t)$ is usually not closed, but given some constraints on the coefficients, the dynamics for $X_i(t)$ can close by itself. For example, if $\beta_i(X) \coloneqq \beta_i, \mu_i(X) \coloneqq \mu_i$ are constants and $g_i(X) \coloneqq g_i X$ takes the linear form, and X is a conserved quantity at division, then

$$\frac{\mathrm{d}X_i(t)}{\mathrm{d}t} = g_i X_i(t) - \mu_i X_i(t) - \beta_i X_i(t) + \beta_{i-1} X_{i-1}(t). \tag{4.3.16}$$

Furthermore, if the growth rate and division rate are independent of the generation number i, we can define the expected total biomass (or protein or mRNA amount)

$$X(t) = \sum_{\vec{n}} \int (\sum_{i=1}^{k} \sum_{j=1}^{n_i} X_{i,j}) \cdot \rho_{\vec{n}}(\vec{X}_{\vec{n}}, t) \mathrm{d}\vec{X}_{\vec{n}} \tag{4.3.17}$$

and any higher-order moment

$$X^q(t) = \sum_{\vec{n}} \int (\sum_{i=1}^{k} \sum_{j=1}^{n_i} X_{i,j})^q \cdot \rho_{\vec{n}}(\vec{X}_{\vec{n}}, t) \mathrm{d}\vec{X}_{\vec{n}}, q > 1. \tag{4.3.18}$$

Specifically, if μ is a constant and $g(X) = \lambda X$, the differential equations satisfied by the first and second order moments $X(t)$ and $X^2(t)$ are

$$\begin{aligned} \frac{\mathrm{d}X(t)}{\mathrm{d}t} &= \int g(X) u_1(X) \mathrm{d}x - \mu X(t), \\ \frac{\mathrm{d}X^2(t)}{\mathrm{d}t} &= \lambda^2 X^2(t) + \sigma^2 X^1(t) - 2\mu X^2(t) + \mu \int x^2 u_1(x, t) \mathrm{d}x. \end{aligned} \tag{4.3.19}$$

General cases for of the equations satisfied by $X^q(t)$ for arbitrary $q \in \mathbb{N}^+$ are discussed in Appendix A.3.3.

In experiments, we usually cannot directly distinguish live or dead cells during experiments when we measure the total biomass or mRNA level as all cells are killed at the end of the experiment to collect data. Therefore, we may also take into account the contribution of dead cells during experiments. We can treat those dead cells to be in the 0^{th} generation which will not grow or divide, *i.e.*, having $g_0 = \beta_0 = 0$. We also wish to calculate the biomass of all dead cells. We can define $\tilde{p}_{\vec{n}}(\vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}(0), 0)$ to be the conditional probability that having a system of $\vec{n} \coloneqq (n_0, ..., n_k)$ (note here we start from n_0) cells in each generation with states $\vec{X}_{\vec{n}} \coloneqq (X_{0,1}, ..., X_{k,n_k})$ given a system of $\vec{n}^0$ cells with states $\vec{X}_{\vec{n}^0}(0)$ at time $t = 0$. Using similar proofs as in Proposition 2 we can show that under certain conditions $\tilde{p}_{\vec{n}}$ satisfies the differential equation

$$\begin{aligned}
&\frac{\partial \tilde{p}_{\vec{n}}}{\partial t} + \sum_{i=1}^{k}\sum_{j=1}^{n_i} \frac{\partial (g_{i,j}\tilde{p}_{\vec{n}})}{\partial X_{i,j}} - \frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{n_i} \frac{\partial^2 (\sigma_{i,j}^2 \tilde{p}_{\vec{n}})}{(\partial X_{i,j})^2} + \sum_{i=1}^{k}\sum_{j=1}^{n_i} (\beta_{i,j} + \mu_{i,j})\tilde{p}_{\vec{n}} = \\
&\quad \sum_{i=1}^{k-1}\sum_{j=1}^{n_{i-1}+1} \int \tilde{\beta}_{i,j}(Y, X_{i+1,n_{i+1}-1}, X_{i+1,n_{i+1}}, t)\tilde{p}_{\vec{n}_{\mathrm{b},i}}(\vec{X}_{\vec{n}_{\mathrm{b},i,j}}, t|\vec{X}_{\vec{n}^0}(0), 0)\mathrm{d}Y \\
&\qquad + \sum_{i=1}^{\infty}\sum_{j=1}^{n_i+1} \mu(X_{0,n_0})\tilde{p}_{\vec{n}_{\breve{\mathrm{d}},i}}(\vec{X}_{\vec{n}_{\breve{\mathrm{d}},i,j}}, t|\vec{X}_{\vec{n}^0}(0), 0)
\end{aligned} \tag{4.3.20}$$

where $\vec{n}_{\breve{\mathrm{d}},i}$ differs from in that its 0^{th} component is $n_0 - 1$ but its i^{th} component is $n_i + 1$, and $\vec{X}_{\vec{n}_{\breve{\mathrm{d}},i,j}}$ differs from $\vec{X}_{\vec{n}}$ in that the internal states of the 0^{th} generation (dead cells) are $(X_{0,1}, ..., X_{0,n_0-1})$ and the internal states of the cells in the i^{th} generation are $(X_{i,1}, ..., X_{i,j-1}, X_{0,n_0}, X_{i,j}, ..., X_{i,n_i})$. Similarly, we can define the unconditional probability density function $\tilde{p}^*_{\vec{n}}(\vec{X}_{\vec{n}}, t)$ as defined in Eq. (4.2.18) as well as the symmetrized probability density function

$$\tilde{\rho}_{\vec{n}}(\vec{X}_{\vec{n}}, t) \coloneqq \Pi_{i=0}^{k} \frac{1}{n_i!} \sum_{\pi} \tilde{p}^*_{\vec{n}}(\pi(\vec{X}_{\vec{n}}), t). \tag{4.3.21}$$

The PDE satisfied by this $\tilde{\rho}_{\vec{n}}$ is

$$\begin{aligned}\frac{\partial\tilde{\rho}_{\vec{n}}}{\partial t}+\sum_{i=1}^{k}\sum_{j=1}^{n_i}\frac{\partial(g_{i,j}\tilde{\rho}_{\vec{n}})}{\partial X_{i,j}}-\frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{n_i}\frac{\partial^2(\sigma_{i,j}^2\tilde{\rho}_{\vec{n}})}{(\partial X_{i,j})^2}=&-\sum_{i=1}^{k}\sum_{j=1}^{n_i}(\beta_{i,j}+\mu_{i,j})\tilde{\rho}_{\vec{n}}\\ +\sum_{i=1}^{k-1}\frac{n_i+1}{n_{i+1}(n_{i+1}-1)}\sum_{j_1\neq j_2}\int\tilde{\beta}_{i,j}(Y,X_{i+1,j_1},X_{i+1,j_2})\tilde{\rho}_{\vec{n}_{\mathrm{b},i}}(\vec{X}_{\vec{n}_{\mathrm{b},i,j_1,j_2}},t)\mathrm{d}Y&\\ +\frac{1}{n_0}\sum_{i=1}^{\infty}(n_i+1)\sum_{j=1}^{n_0}\mu_{i,j}(X_{0,j})\rho_{\vec{n}_{\tilde{\mathrm{d}},i}}(\vec{X}_{\vec{n}_{\tilde{\mathrm{d}},i,j}},t).&\end{aligned}\tag{4.3.22}$$

The total expected dead cells' biomass is

$$Y(t)=\sum_{\vec{n}}\int(\sum_{j=1}^{n_0}X_{0,j})\cdot\tilde{\rho}_{\vec{n}}(\vec{X}_{\vec{n}},t)\mathrm{d}\vec{X}_{\vec{n}}.\tag{4.3.23}$$

If the death rate for the i^{th} generation is a constant μ_i for each i, then $Y(t)$ satisfies the following differential equation

$$\frac{\mathrm{d}Y(t)}{\mathrm{d}t}=\sum_{i=1}^{\infty}\mu_i X_i(t).\tag{4.3.24}$$

where $X_i(t)$ is the total expected biomass of all cells in the i^{th} generation defined in Eq. (4.3.15).

We can also derive the differential equation satisfied by the second order moment of total dead cells' biomass

$$Y^2(t)=\sum_{\vec{n}}\int(\sum_{j=1}^{n_0}X_{0,j})^2\cdot\tilde{\rho}_{\vec{n}}(\vec{X}_{\vec{n}},t)\mathrm{d}\vec{X}_{\vec{n}}.\tag{4.3.25}$$

as well as the correlation

$$X(t)Y(t)=\sum_{\vec{n}}\int(\sum_{i=1}^{k}\sum_{j=1}^{n_i}X_{i,j})\cdot(\sum_{j=1}^{n_0}X_{0,j})\cdot\tilde{\rho}_{\vec{n}}(\vec{X}_{\vec{n}},t)\mathrm{d}\vec{X}_{\vec{n}}.\tag{4.3.26}$$

If we assume that the death rate is a constant μ for all cells and the state variable is conserved

at division, we can derive the differential equation satisfied by $Y^2(t)$

$$\begin{aligned}\frac{\mathrm{d}Y^2(t)}{\mathrm{d}t} &= 2\mu X(t)Y(t) + \mu \sum_{\vec{n}} \int \sum_{j=1}^{n_i} X_{i,j}^2 \cdot \tilde{\rho}_{\vec{n}}(\vec{X}_{\vec{n}}, t)\mathrm{d}\vec{X}_{\vec{n}} \\ &= 2\mu X(t)Y(t) + \mu \int x^2 u_1(x,t)\mathrm{d}x\end{aligned} \tag{4.3.27}$$

and the differential equation satisfied by $X(t)Y(t)$ is

$$\frac{\mathrm{d}(X(t)Y(t))}{\mathrm{d}t} = \lambda X(t)Y(t) - \mu X(t)Y(t) + \mu X^2(t) - \mu \int x^2 u_1(x)\mathrm{d}x \tag{4.3.28}$$

Higher order moments of X, Y can also be evaluated, which we do not include here for brevity.

4.4 Summary and conclusions

In this work, we used the forward-type Feymann-Kac formula and Markov jump process to formulate a kinetic theory for describing the cellular population density of a generation-characterized cellular population with fluctuating rates of changing internal states as well as random division times. Such a kinetic theory not only tracks each cell's states such as its volume, protein amount, or mRNA amount but also tracks the generation (*i.e.*, how many times a cell has divided) of each cell, which helps simulate cellular proliferation and differentiation or dedifferentiation over generations. After marginalizing the differential equation that describes the cellular population's evolution, we can study different macroscopic quantities of interest, such as cell population in each generation and total biomass or protein amount.

As for future directions, it would be promising to incorporate inferred cellular growth rates or rates of producing protein from experiments to apply our kinetic theory to predict experimental outcomes. Furthermore, including a spatial dependence in our model to track the movement of cellular population [AMR08]. Also, it is prospective to incorporate intercellular

interactions where cellular growth rates and dividing rates could have a dependence on other cells [NDF16]. Including such intercellular interactions could give different kinetic equations and PDEs satisfied by macroscopic quantities such as cellular density. Finally, developing efficient unbounded-domain-based algorithms such as [XSC21a, XSC21b, CSX23, XBC23] that can track globally the change of internal states of each cell in unbounded domains to solve the differential equations for describing the generation-characterized cellular population is prospective.

CHAPTER 5

Efficient Scaling and Moving Techniques for Spectral Methods in Unbounded Domains

This is the Accepted Manuscript version of an article accepted for publication in *SIAM Journal on Scientific Computing*, **40**, pp. A3444–A3268, (2021). It is reproduced here with permission of the publisher. SIAM is not responsible for any errors or omissions in this version of the manuscript or any version derived from it. The Version of Record is available online at [10.1137/20M1347711].

5.1 Introduction

Many scientific models described by PDEs with blowup solutions are set in unbounded domains. For example, in many models of cellular proliferation, a "blowup" in which the average size of a population of cells becomes uncontrolled and diverges over many generations of growth is possible [KB18]. The conditions under which blowup occurs are difficult to determine analytically [BDG19] but have been explored numerically [XGC20]. However, numerically tracking "blowup" behavior over long time is extremely difficult, as it requires solving the problem in a truly unbounded domain to capture the diverging mean size. There are many other cases in which it is desirable to find numerical solutions in an unbounded domain. Scenarios include the analysis of the stability of solitary waves arising from the nonlinear Dirac equation [SQM14, CKS16], diffusion in a parabolic system [MST05], and fractional PDEs that admit solutions with algebraic decay at infinity [TYZ18b, TYZ18a].

Considerable progress has recently been made in using spectral methods for solving PDEs in unbounded domains [SW09]. Among the existing spectral methods, the direct approach that is typically used is based on orthogonal basis functions defined on infinite intervals, such as Hermite and Laguerre functions [CFK90, GWW06, TYZ18a], as well as other rational basis functions of recent interest such as the modified mapped Gegenbauer functions (MMGFs) [TWY20]. It has been demonstrated that the performance of these spectral methods can be greatly improved when proper coordinate scaling is used [Tan93, SW09]. However, it is not clear how to systematically perform the scaling, especially when transient behavior arises. A Hermite spectral method with time-dependent scaling has been proposed for parabolic problems by introducing a time-dependent scaling factor $\beta(t)$ to meet the coercive condition [MST05]. Nonetheless, the form of $\beta(t)$ and related parameters are chosen based on specified knowledge of parabolic models and thus cannot be easily generalized to other problems.

Motivated by the success of adaptive methods in bounded domains [RW00, TT03, LTZ01], we propose two indicators to adaptively allocate a sufficient number of collocation points to represent the unknown solution in the region of interest. The first indicator, designed for

matching the diffusion of unknown solutions, extracts the frequency-space information of intermediate numerical solutions and isolates its high-frequency components. This frequency indicator not only provides a lower bound for the interpolation error but also measures the decay of the derivatives of the reference solution as $|x| \to +\infty$. By tuning a scaling factor in our proposed scaling technique, the frequency indicator can be maintained at a low level. However, the translation of unknown solutions may also amplify the frequency indicator and thus may result in larger errors for excessive scaling. To accommodate this scenario, a second, exterior-error indicator is used to calculate an upper bound for the error in the exterior domain, allowing one to capture translation via moving collocation points. Accordingly, for problems that may involve both translation and diffusion in unbounded domains, the above two indicators are combined in a "first moving then scaling" approach. Numerical experiments demonstrate their ability to recover a faster spectral convergence for time-dependent solutions.

In the following, Section 5.2 introduces the frequency indicator, connects it to the approximation error, and proposes the frequency-dependent scaling technique for diffusion. Section 5.3 proposes the exterior-error-dependent moving technique for translating problems. These two approaches are combined in Section 5.4 to solve time-dependent problems involving both diffusion and translation. Section 5.5 compares the frequency-dependent scaling with a time-dependent scaling proposed in [MST05] for solving parabolic systems. In Section 5.6, we generalize the scaling technique to MMGFs which exhibit algebraic decay at infinity. In Section 5.7, we analyze the efficiency of both the scaling and moving techniques and discuss their dependence on parameters. In Section 5.8, we apply the frequency-dependent scaling method to a PDE model describing structured cell populations to track blowup behavior. Finally, we summarize our approaches and make concluding remarks in Section 5.9.

5.2 Frequency-dependent scaling

We formulate our scaling technique by extracting the frequency domain information on the evolution of numerical solutions, the pseudo-code of which is presented in Alg. 1. Our derivation utilizes the generalized Laguerre functions of degree ℓ

$$\hat{\mathcal{L}}_\ell^{(\alpha,\beta)}(x) = \mathcal{L}_\ell^{(\alpha)}(\beta x)e^{-\frac{\beta}{2}x}, \quad \beta > 0, \tag{5.2.1}$$

which, when using the weight function $\hat{\omega}_\alpha(x) = x^\alpha(\alpha > -1)$, are mutually orthogonal on the half-line $\Lambda \coloneqq (0, +\infty)$. Here $\mathcal{L}_\ell^{(\alpha)}(x)$ denote the usual Laguerre polynomials [GWW06] to which $\hat{\mathcal{L}}_\ell^{(\alpha,\beta)}(x)$ reduce when $\beta = 1$. In this work, we regard β to be the *scaling factor*, and seek a time-dependent spectral approximation of $u(x,t)$ on Λ. Henceforth, for notational simplicity, the t-dependence will usually be omitted.

For any $u \in L^2_{\hat{\omega}_\alpha}(\Lambda)$, the spectral approximation using the interpolation operator $\mathcal{I}_{N,\alpha,\beta}$ is

$$u(x) \approx U_N^{(\alpha,\beta)}(x) = \mathcal{I}_{N,\alpha,\beta}u = \sum_{\ell=0}^{N} u_\ell^{(\alpha,\beta)}\hat{\mathcal{L}}_\ell^{(\alpha,\beta)}(x), \tag{5.2.2}$$

where the coefficients $u_\ell^{(\alpha,\beta)}$ can be computed by using *e.g.*, the Laguerre-Gauss collocation points $x_j^{(\alpha,\beta)}$,

$$u_\ell^{(\alpha,\beta)} = \frac{1}{\gamma_\ell^{(\alpha,\beta)}} \sum_{j=0}^{N} \hat{\mathcal{L}}_\ell^{(\alpha,\beta)}(x_j^{(\alpha,\beta)})u(x_j^{(\alpha,\beta)})\hat{w}_j^{(\alpha,\beta)}, \quad \ell = 0, 1, \ldots, N, \tag{5.2.3}$$

where N is the expansion order (*i.e.*, $N+1$ collocation points or $N+1$ basis functions), $\gamma_\ell^{(\alpha,\beta)} = (\hat{\mathcal{L}}_\ell^{(\alpha,\beta)}, \hat{\mathcal{L}}_\ell^{(\alpha,\beta)})_{\hat{\omega}_\alpha}$ is the $L^2_{\hat{\omega}_\alpha}$ inner product, $\hat{w}_j^{(\alpha,\beta)}$ denotes the corresponding weight for collocation point $x_j^{(\alpha,\beta)}$, and

$$u(x_j^{(\alpha,\beta)}) = U_N^{(\alpha,\beta)}(x_j^{(\alpha,\beta)}) = \mathcal{I}_{N,\alpha,\beta}u(x_j^{(\alpha,\beta)}), \quad j = 0, 1, \ldots, N. \tag{5.2.4}$$

When the scaling factor is updated from β to $\tilde{\beta}$, the collocation points, weights, and $L^2_{\hat{\omega}_\alpha}$

norms are updated according to

$$x_j^{(\alpha,\tilde{\beta})} = \frac{\beta}{\tilde{\beta}} x_j^{(\alpha,\beta)}, \quad \hat{w}_j^{(\alpha,\tilde{\beta})} = \frac{\beta^{\alpha+1}}{\tilde{\beta}^{\alpha+1}} \hat{w}_j^{(\alpha,\beta)}, \quad \gamma_\ell^{(\alpha,\tilde{\beta})} = \frac{\beta^{\alpha+1}}{\tilde{\beta}^{\alpha+1}} \gamma_\ell^{(\alpha,\beta)}. \tag{5.2.5}$$

The expansion coefficients $u_\ell^{(\alpha,\tilde{\beta})}$ can then be estimated through Eq. (5.2.3) where we may use the approximation (5.2.2): $u(x_j^{(\alpha,\tilde{\beta})}) \approx U_N^{(\alpha,\beta)}(x_j^{(\alpha,\tilde{\beta})})$. This procedure constitutes the SCALE subroutine in Lines 9 and 17 of Alg. 1.

To implement the scaling technique, one needs to determine when to apply it and how to choose a new scaling factor $\tilde{\beta}$ such that spectral accuracy can be kept for a prescribed expansion of order N. To this end, we propose a *frequency indicator* acting on the numerical solution $U_N^{(\alpha,\beta)}$:

$$\mathcal{F}(U_N^{(\alpha,\beta)}) = \left(\frac{\sum_{\ell=N-M+1}^{N} \gamma_\ell^{(\alpha,\beta)} \cdot (u_\ell^{(\alpha,\beta)})^2}{\sum_{\ell=0}^{N} \gamma_\ell^{(\alpha,\beta)} \cdot (u_\ell^{(\alpha,\beta)})^2} \right)^{\frac{1}{2}}, \tag{5.2.6}$$

which measures the contribution of the M highest-frequency components to the $L^2_{\hat{\omega}_\alpha}$-norm of $U_N^{(\alpha,\beta)}$. The subroutine FREQUENCY_INDICATOR in Lines 3, 6, 10, and 18 of Alg. 1 calculates this contribution in which we choose $M = [\frac{N}{3}]$ in view of the often-used $\frac{2}{3}$-rule [HL07, Ors71].

If the frequency indicator $\mathcal{F}(U_N^{(\alpha,\beta)})$ increases over time, the contribution of high frequency components to the numerical solution increases, indicating that the numerical solution is decaying more slowly in x and that we need to adjust the scaling factor to enlarge the computational domain $[x_0^{(\alpha,\beta)}, x_N^{(\alpha,\beta)}]$ demarcated by the smallest and largest collocation point positions. In Line 7 of Alg. 1, νf_0 is the threshold at some time t. If the value of the frequency indicator of the current numerical solution $f > \nu f_0$, then we consider scaling. The parameter ν is usually chosen to be slightly larger than 1 to prevent the frequency indicator becoming too large without invoking scaling.

However, the **if** condition is only a necessary condition. Only after we enter the **while** loop in Line 11 will we perform scaling, which aims to ensure that the frequency indicator $\mathcal{F}(U_N^{(\alpha,\beta)})$ will not increase after scaling. Actually, this **while** loop tries to minimize

Algorithm 1 Pseudo-code of spectral methods with frequency-dependent scaling.

1: Initialize N, $\nu > 1$, $q < 1$, Δt, T, α, β, $U_N^{(\alpha,\beta)}(0)$, $\underline{\beta}$
2: $t \leftarrow 0$
3: $f_0 \leftarrow \text{FREQUENCY\_INDICATOR}(U_N^{(\alpha,\beta)}(t))$
4: **while** $t < T$ **do**
5: $\quad U_N^{(\alpha,\beta)}(t+\Delta t) \leftarrow \text{EVOLVE}(U_N^{(\alpha,\beta)}(t), \Delta t)$
6: $\quad f \leftarrow \text{FREQUENCY\_INDICATOR}(U_N^{(\alpha,\beta)}(t+\Delta t))$
7: $\quad$ **if** $f > \nu f_0$ **then**
8: $\quad\quad \tilde{\beta} \leftarrow q\beta$
9: $\quad\quad U_N^{(\alpha,\tilde{\beta})} \leftarrow \text{SCALE}(U_N^{(\alpha,\beta)}(t+\Delta t), \tilde{\beta})$
10: $\quad\quad \tilde{f} \leftarrow \text{FREQUENCY\_INDICATOR}(U_N^{(\alpha,\tilde{\beta})})$
11: $\quad\quad$ **while** $\tilde{f} \leq f$ and $\tilde{\beta} \geq \underline{\beta}$ **do**
12: $\quad\quad\quad \beta \leftarrow \tilde{\beta}$
13: $\quad\quad\quad U_N^{(\alpha,\beta)}(t+\Delta t) \leftarrow U_N^{(\alpha,\tilde{\beta})}$
14: $\quad\quad\quad f_0 \leftarrow \tilde{f}$
15: $\quad\quad\quad f \leftarrow \tilde{f}$
16: $\quad\quad\quad \tilde{\beta} \leftarrow q\beta$
17: $\quad\quad\quad U_N^{(\alpha,\tilde{\beta})} \leftarrow \text{SCALE}(U_N^{(\alpha,\beta)}(t+\Delta t), \tilde{\beta})$
18: $\quad\quad\quad \tilde{f} \leftarrow \text{FREQUENCY\_INDICATOR}(U_N^{(\alpha,\tilde{\beta})})$
19: $\quad\quad$ **end while**
20: $\quad$ **end if**
21: $\quad t \leftarrow t + \Delta t$
22: **end while**

$\mathcal{F}(U_N^{(\alpha,\beta)})$ by geometrically shrinking the scaling factor β (q in Line 16 is the common ratio) to ensure sufficient scaling since $\mathcal{F}(U_N^{(\alpha,\beta)})$ is a lower bound for the numerical error, as shown in Eq. (5.7.2). A more continuous adjustment is preferred by setting q to be slightly less than 1, which may also prevent over-shrinking of the scaling factor within one single time step. Henceforth, we will choose $q = 0.95$ and $\nu = 1/q$. Moreover, at the initial time $t = 0$, we also ensure the frequency indicator is small enough by choosing a suitable initial scaling factor.

In this work, the generalized Laguerre functions with $\alpha = 0$ are used and the relative $L^2_{\hat{\omega}_\alpha}$-error

$$\text{Error} = \frac{\|U_N^{(\alpha,\beta)} - u\|_{\hat{\omega}_\alpha}}{\|u\|_{\hat{\omega}_\alpha}} \tag{5.2.7}$$

is used to measure the quality of the spectral approximation $U_N^{(\alpha,\beta)}(x)$ to the reference solution $u(x)$. We always use the most updated scaling factor to calculate the above error.

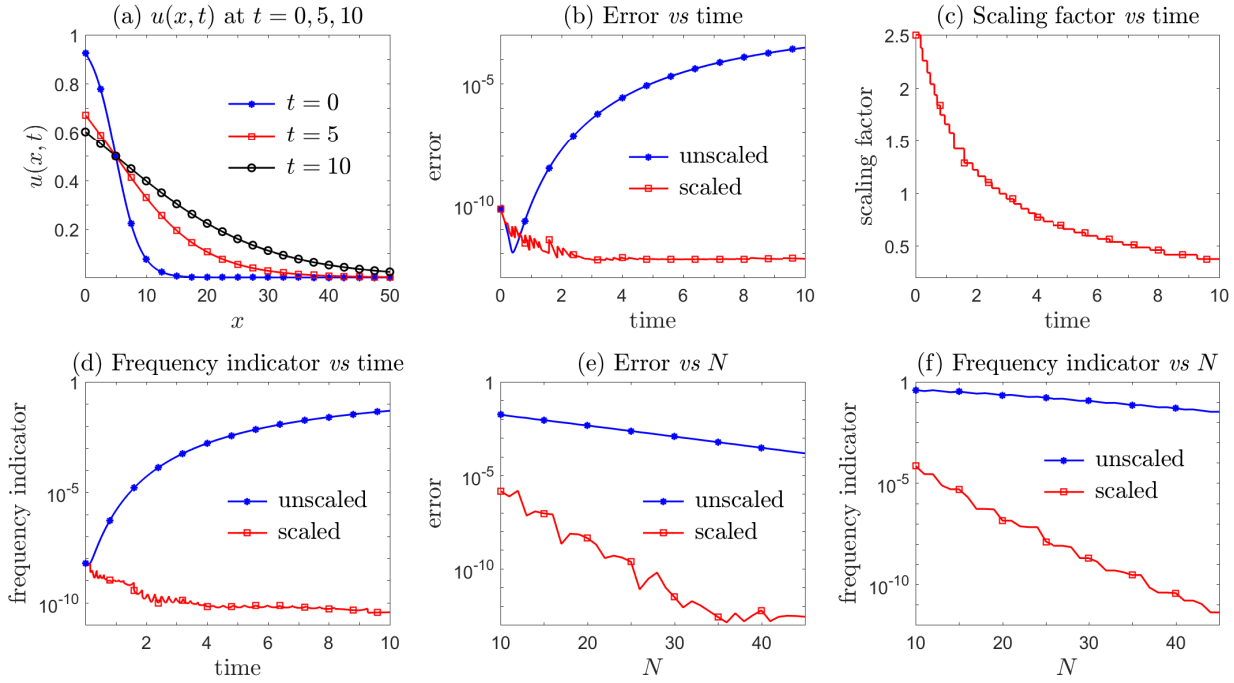

Figure 5.1: Numerical approximation to the diffusive Fermi-Dirac distribution $u(x,t)$ given by Eq. (5.2.8). The scaling algorithm Alg. 1 produces much more accurate solutions and recovers a faster spectral convergence with respect to the expansion order N. As we expected, the frequency indicator defined in Eq. (5.2.6) shows a similar behavior to the error defined in Eq. (5.2.7) against either time or N. The data in last two plots are measured at $t = 10$.

Example 1. We use the spreading Fermi-Dirac distribution

$$u(x,t) = \frac{1}{1 + \mathrm{e}^{\frac{x-5}{2+t}}} \tag{5.2.8}$$

to test the performance of the scaling algorithm Alg. 1. It can be readily verified that the reference solution $u(x,t)$ expands over time as shown in Fig. 5.1(a). The proposed frequency-dependent scaling with $N = 40$ effectively maintains the relative error under 10^{-10} up until time $t = 10$ whereas the error for the corresponding unscaled solution rapidly grows to over 10^{-4} (see Fig. 5.1(b)). We also plot, as $u(x,t)$ evolves, the history of the scaling factor β and frequency indicator $\mathcal{F}(U_N^{(\alpha,\beta)})$ in Figs. 5.1(c) and 5.1(d), respectively. It is clear that the frequency indicator increases for the unscaled solution as time evolves and

that time-dependent scaling is required to preserve the accuracy. The proposed frequency-dependent scaling technique detects the error and shrinks the scaling factor in order to enlarge the computational domain in accordance with the expansion of the reference solution. The spectral convergence as a function of the expansion order N can be also recovered by Alg. 1. The errors at the final time, for the scaled and unscaled approach, are displayed in Fig. 5.1(e). The final scaling factors at $t = 10$ are $0.3213, 0.3560, 0.3747, 0.3945, 0.3945$ for $N = 25, 30, 35, 40, 45$, respectively, having all decreased from the common initial scaling factor of 2.5. Figs. 5.1(e, f) show very similar and expected behavior of the frequency indicator and error as a function of N. Since the error and the frequency indicators behave similarly across time (see Figs. 5.1(b, d)), we also expect them to behave similarly with N. These similarities suggest a possible connection between the error and the frequency indicator.

5.3 Exterior-error-dependent moving

Dynamics in unbounded domains can be much richer than the simple diffusive behavior successfully captured by our frequency-dependent scaling. Other physical mechanisms may induce, for example, translations (Examples 2 and 3) and emerging oscillations (Example 4). A purely scaling approach fails in these cases.

In this section, we develop an exterior-error-dependent moving method that will be able to resolve a solution's decay in an undetermined exterior domain $\Lambda_{\mathrm{e}} \coloneqq (x_L, +\infty)$. Alg. 2 presents the pseudo-code of our exterior-error-dependent moving technique. In the algorithm, we first need to determine the time-dependent left-end point x_L. Next, we move the spectral basis accordingly so that the spectral approximation for an unknown function $u(x)$ in Λ_{e} (denoted by $U^{(\alpha,\beta)}_{N,x_L}(x)$) maintains accuracy. To implement this procedure, we adopt an *exterior-error indicator*:

$$\mathcal{E}(U^{(\alpha,\beta)}_{N,x_L}, x_R) = \frac{\|\partial_x U^{(\alpha,\beta)}_{N,x_L} \cdot \mathbb{I}_{(x_R,+\infty)}\|_{\hat{\omega}_\alpha}}{\|\partial_x U^{(\alpha,\beta)}_{N,x_L} \cdot \mathbb{I}_{(x_L,+\infty)}\|_{\hat{\omega}_\alpha}}, \tag{5.3.1}$$

which measures the proportion of the norm $\|\partial_x U^{(\alpha,\beta)}_{N,x_L} \cdot \mathbb{I}_{(x_R,+\infty)}\|_{\hat{\omega}_\alpha}$ inside a prescribed unbounded domain $(x_L, +\infty)$.

The subroutine EXTERIOR_ERROR_INDICATOR in Lines 5, 8, and 13 of Alg. 2 calculates $\mathcal{E}(U^{(\alpha,\beta)}_{N,x_L}, x_R)$. Here, following the often-used $\frac{2}{3}$-rule [HL07, Ors71], we choose $x_R = x^{(\alpha,\beta)}_{[\frac{N+2}{3}]}$ from the collocation points $x^{(\alpha,\beta)}_j (j = 0, 1, \ldots, N)$ in the exterior domain Λ_{e}.

Intuitively, if $u(x)$ moves rightward in time, such as the moving Fermi-Dirac distribution in Example 2, the spectral approximation at large distances may deteriorate and the exterior-error indicator $\mathcal{E}(U^{(\alpha,\beta)}_{N,x_L})$ will increase. Consequently, the moving mechanism is triggered in Line 9 of Alg. 2, and completed by updating the left end point $x_L = x_L + d_0$ in Line 11. Thus, the starting point of the spectral approximation also moves rightward with time to capture the translation.

The displacement $d_0 = \min\{n\delta, d_{max}\}$ is determined by the MOVE subroutine in Line 10, where n is the smallest integer satisfying $\mathcal{E}(U^{(\alpha,\beta)}_{N,x_L}, x_R + n\delta) < \mu e_0$, δ is the minimum displacement, d_{max} is the maximum displacement, and μ represents the threshold of the increase in the exterior-error indicator that we can tolerate. In practice, d_{max} should be based on a prior knowledge of the maximum translation speed of the function $u(x)$. We usually choose $\mu \gtrsim 1$ to prevent the exterior-error indicator from becoming too large without invoking moving. The MOVE subroutine also generates $U^{(\alpha,\beta)}_{N,x_L+d_0}$ from $U^{(\alpha,\beta)}_{N,x_L}$.

Example 2. In this example, we consider the moving Fermi-Dirac distribution

$$u(x,t) = \frac{1}{1 + \mathrm{e}^{\frac{x-5t}{2}}} \tag{5.3.2}$$

which travels to the right at a speed of 5 without changing shape (see Fig. 5.2(a)). The scaling algorithm Alg. 1, equipped with the same parameters that worked well for the diffusive Fermi-Dirac distribution in Example 1, fails to capture the translation. In fact, the errors of the scaled solutions are larger than those of unscaled ones as shown in Fig. 5.2(b). It appears that the decrease of the scaling factor (black curve with asterisks in Fig. 5.2(c)) cannot compensate for the increase in the frequency indicator (black curve with asterisks in

Algorithm 2 Pseudo-code of spectral methods with exterior-error-dependent moving.

Initialize N, Δt, T, α, β, $U_{N,0}^{(\alpha,\beta)}(0)$, $\mu > 1$, $d_{max} > \delta > 0$
$t \leftarrow 0$
$x_L \leftarrow 0$
$x_R \leftarrow x_{[\frac{N+2}{3}]}^{(\alpha,\beta)}$
$e_0 \leftarrow \text{EXTERIOR\_ERROR\_INDICATOR}(U_{N,x_L}^{(\alpha,\beta)}(0), x_R)$
while $t < T$ **do**
    $U_{N,x_L}^{(\alpha,\beta)}(t + \Delta t) \leftarrow \text{EVOLVE}(U_{N,x_L}^{(\alpha,\beta)}(t), \Delta t)$
    $e \leftarrow \text{EXTERIOR\_ERROR\_INDICATOR}(U_{N,x_L}^{(\alpha,\beta)}(t + \Delta t), x_R)$
    if $e > \mu e_0$ **then**
        $(d_0, U_{N,x_L+d_0}^{(\alpha,\beta)}) \leftarrow \text{MOVE}(U_{N,x_L}^{(\alpha,\beta)}(t + \Delta t), \delta, d_{max}, \mu e_0)$
        $x_L \leftarrow x_L + d_0$
        $x_R \leftarrow x_R + d_0$
        $e_0 \leftarrow \text{EXTERIOR\_ERROR\_INDICATOR}(U_{N,x_L}^{(\alpha,\beta)}(t + \Delta t), x_R)$
    end if
    $t \leftarrow t + \Delta t$
end while

Fig. 5.2(d)). In other words, the scaling algorithm 1 mistakes translation for diffusion and performs excessive scaling. In contrast, the exterior-error-dependent moving algorithm 2 with $\delta = 0.004$, $d_{max} = 0.04$ and $\mu = 1.005$ succeeds in producing a much more accurate approximation to the moving Fermi-Dirac distribution given by Eq. (5.3.2) in the exterior domain Λ_e, with errors kept under 10^{-11} up to time $t = 10$ (red curve with left-pointing triangles in Fig. 5.2(b)). The moving technique recovers a faster spectral convergence with respect to the expansion order N as shown in Fig. 5.2(e).

During the moving process, the exterior-error indicator $\mathcal{E}(U_{N,x_L}^{(\alpha,\beta)}, x_R)$ is well controlled (red curve with left-pointing triangles in Fig. 5.2(f)) and the left-end point of the exterior domain closely tracks the uniform linear motion (red curve with left-pointing triangles in Fig. 5.2(c)). The exterior-error indicator monotonically increases for the unscaled and unmoved solutions (blue curve with squares in Fig. 5.2(f)) and oscillates rapidly for the scaled and unmoved solutions (black curve with asterisks in Fig. 5.2(f)). Moreover, the similarity between the relative error and frequency indicator as a function of time is again confirmed by comparing Fig. 5.2(d) to Fig. 5.2(b), thus providing strong evidence for the effectiveness

of using the frequency indicator (5.2.6). Spectral convergence in N is clearly observed for the moving spectral method in Fig. 5.2(e) while the error decays slowly with N for the unmoved spectral method.

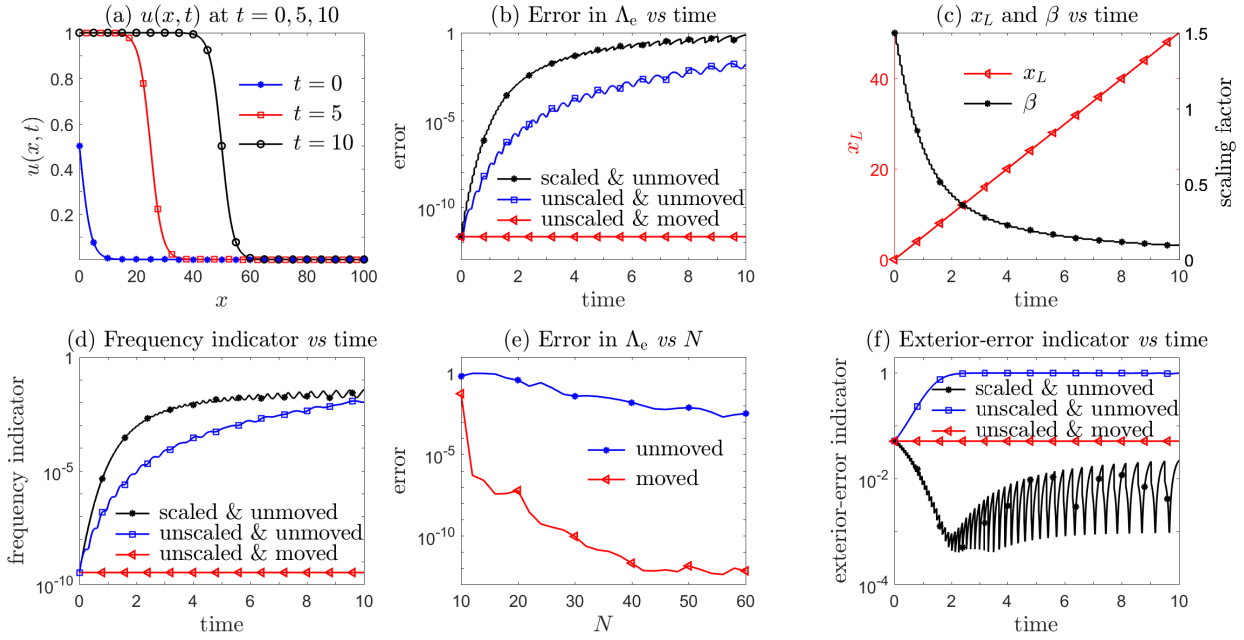

Figure 5.2: Numerical approximation to the moving Fermi-Dirac distribution $u(x,t)$ given by Eq. (5.3.2). The moving algorithm 2 produces much more accurate solutions and recovers a faster spectral convergence with respect to the expansion order N in the exterior domain $\Lambda_e = (x_L, +\infty)$, whereas pure scaling fails to capture this translation. The data in the last plot are measured at $t = 10$.

Example 3. Another class of dynamical systems is described by solitons or solitary waves in which nonlinearities and dispersion counteract. While solitons have been well-studied, there has been recent interest in nonlinear Dirac solitary waves as they emerge naturally in many physical systems [CKS16]. Stability of the nonlinear Dirac solitary waves on the whole line and its connection to the multi-hump structure is a challenging topic of research [SQM14, XST15, BC19]. In this example, we approximate a right-moving two-hump solitary wave, the explicit form of which is given in [ST05] with $v = 0.25$, $\lambda = 0.5$, $m = 1$, $x_0 = -1.5$ and $\Lambda = 0.1$. The reference solutions are plotted in Fig. 5.3(a).

Numerical results are displayed in Fig. 5.3 where we set $\delta = 0.004$, $d_{max} = 0.012$, $\mu = 1.005$. It can be readily observed there that the exterior-error-dependent moving algorithm 2

produces much more accurate solutions with errors kept under 10^{-11} until the final time $t = 15$ (red curve with left-pointing triangles in Fig. 5.3(b)). The moving algorithm also recovers a faster spectral convergence with respect to the expansion order N (see Fig. 5.3(c)). The scaling-only algorithm 1 fails to maintain the accuracy (black curve with asterisks in Fig. 5.3(b)). The similarity between the relative error and frequency indicator is again confirmed by comparing Fig. 5.3(d) to Fig. 5.3(b).

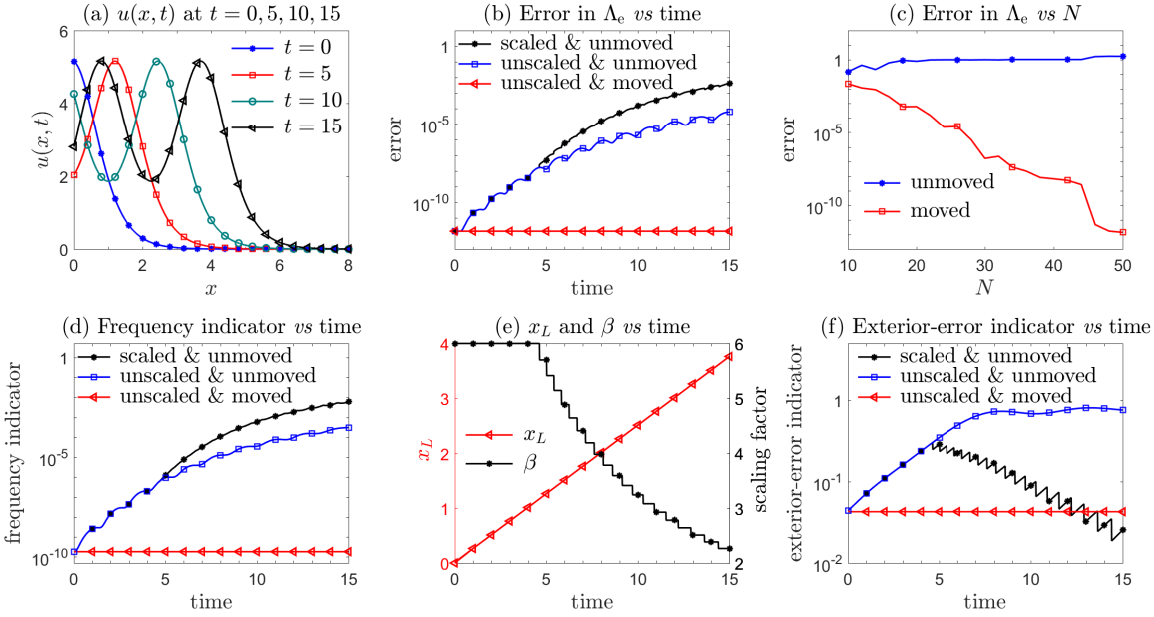

Figure 5.3: Approximating a two-hump nonlinear Dirac solitary wave. The moving algorithm Alg. 2 produces much more accurate solutions and recovers a faster spectral convergence with respect to the expansion order N in the exterior domain $\Lambda_e = (x_L, +\infty)$, whereas a pure scaling approach fails to capture this translation. The data in the last plot are measured at $t = 15$.

In Examples 2 and 3, the exterior-error indicator (5.3.1) efficiently guides us in finding an x_L such that the moved spectral approximation retains accuracy in the resulting exterior domain. This accuracy arises because the exterior-error indicator is related to the upper bound of the error for asymptotically large x. If we assume a large indicator $\mathcal{E}(U^{(\alpha,\beta)}_{N,x_L}, x_R) > \mu$ with $\mu \in (0, 1)$,

$$\mathcal{E}(U^{(\alpha,\beta)}_{N,x_L}, x_R) > \mu \Rightarrow \|\partial_x U^{(\alpha,\beta)}_{N,x_L} \cdot \mathbb{I}_{[x_R,+\infty)}\|_{\hat{\omega}_\alpha} > \mu \|\partial_x U^{(\alpha,\beta)}_{N,x_L} \cdot \mathbb{I}_{[x_L,+\infty)}\|_{\hat{\omega}_\alpha},$$

$$\Rightarrow \|\partial_x U^{(\alpha,\beta)}_{N,x_L} \cdot \mathbb{I}_{[x_R,+\infty)}\|_{\hat{\omega}_{\alpha+1}} > \mu \|\partial_x U^{(\alpha,\beta)}_{N,x_L} \cdot \mathbb{I}_{[x_L,+\infty)}\|_{\hat{\omega}_{\alpha+1}}.$$

Since $\|\partial_x U^{(\alpha,\beta)}_{N,x_L} \cdot \mathbb{I}_{(x_L,+\infty)}\|_{\hat{\omega}_{\alpha+1}}$ is related to the upper bound of the interpolation error $\|(\mathcal{I}_{N,\alpha,\beta}u - u)\mathbb{I}_{(x_L,+\infty)}\|_{\hat{\omega}_\alpha}$ [STW11], a larger exterior-error indicator signals a worsening approximation in the exterior domain (x_R, ∞). The solution in the interior domain $\Lambda_{\rm i} := (0, x_L]$ is not approximated by the basis functions used to approximate the solution in the exterior domain. Obstacles to designing moving mesh methods in unbounded domains include the construction of an *interior numerical solution* and its consistent coupling with the *exterior spectral approximation*. More on these issues will be illustrated in Example 4.

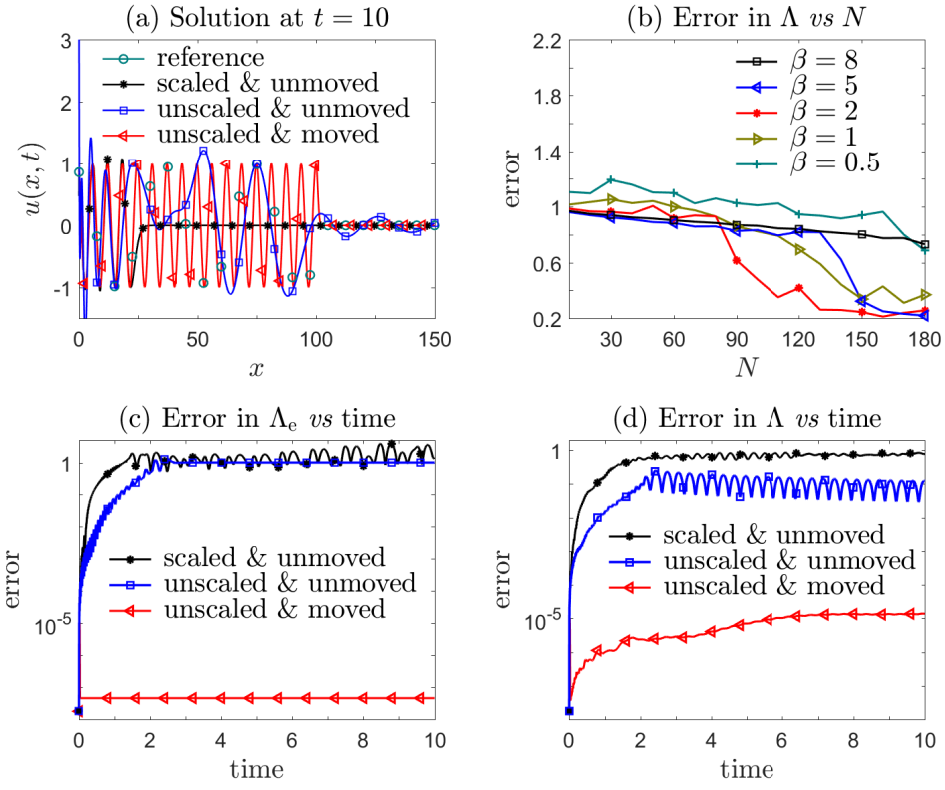

Figure 5.4: Oscillations emanate from the left but the moving algorithm 2 generates accurate solutions in the exterior domain $\Lambda_{\rm e}$, with relative errors under 10^{-7} up to $t = 10$ with $N = 30$ (red curve with left-pointing triangles in (c)). By further coupling with a spectral approximation using 80 Chebyshev polynomials in the interior domain $\Lambda_{\rm i}$, we generate the whole solution with total relative error, up until $t = 10$, under 2×10^{-5}, as shown by the red curves with left-pointing triangles in (a) and (d). The data in (b) are measured at $t = 10$.

Example 4. Let us approximate the following function in Λ:

$$u(x,t) = \begin{cases} \cos(x-10t), & x \le 10t, \\ \mathrm{e}^{-(x-10t)^2}, & x > 10t, \end{cases} \tag{5.3.3}$$

which represents a wave with period 2π traveling to the right with speed 10 and exponentially decaying at infinity. The reference solution $u(x,10)$ is plotted by the green curve with circles in Fig. 5.4(a), which coincides with the red curve with left-pointing triangles that approximates u separately in Λ_{i} and Λ_{e} using different basis functions. As shown by the blue curve with squares in Fig. 5.4(a), applying a Laguerre spectral approximation with $N = 30$ and $\beta = 5$ in Λ fails to accurately approximate $u(x,t)$. This failure arises because more oscillations emerge from $x = 0$ and translate to $+\infty$ as time evolves. Specifically, at $t = 10$, the reference solution $u(x,t)$ possesses 32 extrema while any Laguerre spectral approximation (5.2.2) with $N = 30$ can have at most 30 extrema, implying that the approximation is doomed to fail since all oscillations cannot be captured. Simply increasing the number of basis functions does little to help, even with different scaling factors as shown in Fig. 5.4(b). The ineffectiveness of increasing N is mainly due to the presence of oscillatory components with significantly different frequencies in each of the two different domains. As shown by the black curves with asterisks in Figs. 5.4(a, c, d), the scaling technique is also doomed to fail because it totally neglects this scale difference and only adjusts the scaling factor to redistribute collocation points.

We propose a divide-and-conquer strategy to address Example 4 that can be implemented by applying two subroutines, within each time step. The first step is to use the exterior-error-dependent moving algorithm 2 to determine the exterior spectral approximation for the exponential decay component of the reference solution. The second step is to introduce a new spectral approximation in the remaining bounded interior domain Λ_{i} for the left-side oscillating component. The full numerical solution in the half-line Λ is constructed from concatenating the solution in the exterior domain Λ_{e} to the one in the interior domain Λ_{i}.

Fig. 5.4(c) plots the error in the exterior domain against time and shows that the errors

of of the moved solution with $N = 30$, $\delta = 0.008$, $d_{max} = 0.08$ and $\mu = 1.001$ are kept under 10^{-7} up to time $t = 10$ (red curve with left-pointing triangles), confirming that the Laguerre spectral approximation is accurate in the exterior domain. In fact, the numerical values of x_L obtained by the moving algorithm 2 are consistent with the expected value of $10t$ as shown in Eq. (5.3.3). Coupling the exterior solution with a spectral approximation using 80 Chebyshev polynomials in the interior domain, we find a combined numerical solution with total relative error under 2×10^{-5} up to $t = 10$ (red curves with left-pointing triangles in Figs. 5.4(a, d)) using $111 = 31 + 80$ total basis functions. By contrast, Fig. 5.4(b) shows that the errors for direct refinement using $N = 180$ are larger than 0.2.

It must be pointed out that when solving PDEs in unbounded domains, we may need information about the solution in the exterior domain to construct the interior numerical solution. Further discussion on this point can be found in Example 6.

5.4 Spectral methods incorporating both scaling and moving

For problems that involve both translation and diffusion in unbounded domains, we need to incorporate both the moving and scaling procedures. Since the scaling algorithm Alg. 1 may mistake translation for diffusion and trigger an inappropriate scaling as shown in Examples 2 and 3, we propose a "first moving then scaling" algorithm. The associated pseudo-code is described in Alg. 3. A direct application of Alg. 3 to Example 1 recovers exactly the same results as Alg. 1 since the moving procedure is not invoked. When Alg. 3 is applied to Examples 2 and 3, it gives the same results as Alg. 2 since the scaling mechanism is not triggered. That is, the combined moving-scaling algorithm 3 can deal with both translation-only and diffusion-only problems since it can distinguish translation from diffusion.

Alg. 3 can be extended to unbounded domains in multiple dimensions in a dimension-by-dimension manner by using the tensor product of one-dimensional basis functions. For

example, consider the two-dimensional spectral approximation

$$U^{(\vec{\alpha},\vec{\beta})}_{N,x_L,y_L}(x,y) \coloneqq \sum_{\ell=0}^{N_x}\sum_{m=0}^{N_y} u^{(\vec{\alpha},\vec{\beta})}_{\ell,m}\hat{\mathcal{L}}^{\alpha_x,\beta_x}_{\ell}(x)\hat{\mathcal{L}}^{\alpha_y,\beta_y}_{m}(y) \tag{5.4.1}$$

in $\Lambda^x_{\mathrm{e}} \times \Lambda^y_{\mathrm{e}} := (x_L,+\infty)\times(y_L,+\infty)$ where $\vec{\alpha}=(\alpha_x,\alpha_y)$ and $\vec{\beta}=(\beta_x,\beta_y)$. We choose the exterior-error indicator in x-dimension to be

$$\mathcal{E}_x(U^{(\vec{\alpha},\vec{\beta})}_{N,x_L,y_L}(x,y),x_R) \coloneqq \mathcal{E}(\tilde{U}^{(\alpha_x,\beta_x)}_{N,x_L}(x),x_R), \tag{5.4.2}$$

$$\tilde{U}^{(\alpha_x,\beta_x)}_{N,x_L}(x) \coloneqq \int_{\Lambda^y_{\mathrm{e}}} U^{(\vec{\alpha},\vec{\beta})}_{N,x_L,y_L}(x,y)\mathrm{d}y. \tag{5.4.3}$$

Similarly, $\mathcal{E}_y(U^{(\vec{\alpha},\vec{\beta})}_{N,x_L,y_L}(x,y),y_R)$ gives the exterior-error indicator in y-dimension. Accordingly, we use $\mathcal{E}_x(U^{(\vec{\alpha},\vec{\beta})}_{N,x_L,y_L},x_R)$ to judge the **if** statement in Line 10 of Alg. 3. If satisfied, then the MOVE subroutine in Line 11 of Alg. 3 will move the solution in x-direction via $x_L \to x_L + d^x_0$. Simultaneously, we use $\mathcal{E}_y(U^{(\vec{\alpha},\vec{\beta})}_{N,x_L,y_L}(x,y),y_R)$ to determine the shift in the y-direction.

To allow scaling in x-direction, the corresponding frequency indicator can be defined as

$$\mathcal{F}_x(U^{(\vec{\alpha},\vec{\beta})}_{N,x_L,y_L}) \coloneqq \left(\frac{\sum_{\ell=N_x-M_x+1}^{N_x}\sum_{m=0}^{N_y}\gamma^{(\alpha_x,\beta_x)}_{\ell}\cdot\gamma^{(\alpha_y,\beta_y)}_{m}\cdot(u^{(\vec{\alpha},\vec{\beta})}_{\ell,m})^2}{\sum_{\ell=0}^{N_x}\sum_{m=0}^{N_y}\gamma^{(\alpha_x,\beta_x)}_{\ell}\cdot\gamma^{(\alpha_y,\beta_y)}_{m}\cdot(u^{(\vec{\alpha},\vec{\beta})}_{\ell,m})^2}\right)^{\frac{1}{2}}, \tag{5.4.4}$$

where $M_x = [\frac{N_x}{3}]$ and N_x, N_y are the expansion orders in the x-, y-directions, respectively. Similarly, we define $\mathcal{F}_y$ to be the frequency indicator in y-direction. We first keep β_y fixed and use $\mathcal{F}_x$ to evaluate the **if** statement in Line 16 of Alg. 1 for scaling. If scaling in x-direction is needed, then the **while** loop in Line 20 of Alg. 1 will update the scaling factor to $\tilde{\beta}_x$. Simultaneously, we fix β_x and use $\mathcal{F}_y$ to update the scaling factor in the y-direction to $\tilde{\beta}_y$. After that, the scaling factors for time $t+\Delta t$ are set to $\tilde{\beta}_x$ and $\tilde{\beta}_y$.

Example 5. We will investigate the performance of Alg. 3 in a two-dimensional unbounded

Algorithm 3 Pseudo-code of spectral methods with both scaling and moving.

Initialize N, $\nu > 1$, $q < 1$, Δt, T, α, β, $U_N^{(\alpha,\beta)}(0)$, $\underline{\beta}$, $\mu > 1$, $d_{max} > \delta > 0$, $x_R(0) = x_{[\frac{N+2}{3}]}^{(\alpha,\beta)}$
$x_L, t \leftarrow 0$
$x_R \leftarrow x_{[\frac{N+2}{3}]}^{(\alpha,\beta)}$
$f_0 \leftarrow \text{FREQUENCY\_INDICATOR}(U_{N,x_L}^{(\alpha,\beta)}(x,t))$
$e_0 \leftarrow \text{EXTERIOR\_ERROR\_INDICATOR}(U_{N,x_L}^{(\alpha,\beta)}(0), x_R)$
while $t < T$ **do**
    $x_R \leftarrow x_{[\frac{N+2}{3}]}^{(\alpha,\beta)}$
    $U_{N,x_L}^{(\alpha,\beta)}(x, t+\Delta t) \leftarrow \text{EVOLVE}(U_{N,x_L}^{(\alpha,\beta)}(x,t)), \Delta t)$
    $e \leftarrow \text{EXTERIOR\_ERROR\_INDICATOR}(U_{N,x_L}^{(\alpha,\beta)}(x, t+\Delta t), x_R)$
    if $e > \mu e_0$ **then**
        $(d_0, U_{N,x_L+d_0}^{(\alpha,\beta)}) \leftarrow \text{MOVE}(U_{N,x_L}^{(\alpha,\beta)}(x, t+\Delta t), \delta, d_{max}, \mu e_0)$
        $x_L \leftarrow x_L + d_0$
        $e_0 \leftarrow \text{EXTERIOR\_ERROR\_INDICATOR}(U_{N,x_L}^{(\alpha,\beta)}(x, t+\Delta t), x_R)$
    end if
    $f \leftarrow \text{FREQUENCY\_INDICATOR}(U_{N,x_L}^{(\alpha,\beta)}(x, t+\Delta t))$
    if $f > \nu f_0$ **then**
        $\tilde{\beta} \leftarrow q\beta$
        $U_{N,x_L}^{(\alpha,\tilde{\beta})} \leftarrow \text{SCALE}(U_{N,x_L}^{(\alpha,\beta)}(x, t+\Delta t), \tilde{\beta})$
        $\tilde{f} \leftarrow \text{FREQUENCY\_INDICATOR}(U_{N,x_L}^{(\alpha,\tilde{\beta})})$
        while $\tilde{f} \leq f$ and $\tilde{\beta} \geq \underline{\beta}$ **do**
            $\beta \leftarrow \tilde{\beta}$
            $U_{N,x_L}^{(\alpha,\beta)}(x, t+\Delta t) \leftarrow U_{N,x_L}^{(\alpha,\tilde{\beta})}$
            $f_0 \leftarrow \tilde{f}$
            $f \leftarrow \tilde{f}$
            $\tilde{\beta} \leftarrow q\beta$
            $U_{N,x_L}^{(\alpha,\tilde{\beta})} \leftarrow \text{SCALE}(U_{N,x_L}^{(\alpha,\beta)}(x, t+\Delta t), \tilde{\beta})$
            $\tilde{f} \leftarrow \text{FREQUENCY\_INDICATOR}(U_{N,x_L}^{(\alpha,\tilde{\beta})})$
        end while
    end if
    $t \leftarrow t + \Delta t$
end while

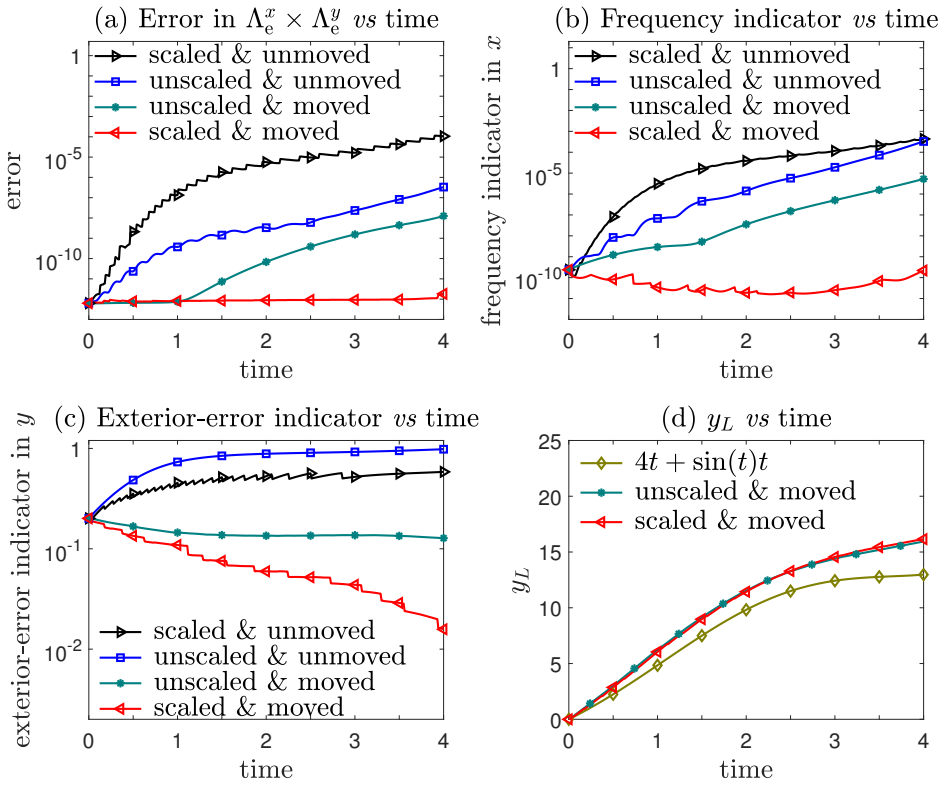

Figure 5.5: A two-dimensional oscillatory function with both translation and diffusion given by Eq. (5.4.5). Only the combined moving-scaling algorithm 3 produces accurate solutions in the exterior domain with errors kept under 10^{-11} up to $t = 4$. The need for combining moving and scaling is evident. For simplicity, we only used $\mathcal{F}_x$ (the frequency indicator in the x-direction), $\mathcal{E}_y$ (the exterior-error indicator in the y-direction), and y_L (the left end of Λ_e^y) as an example. The corresponding curves for $\mathcal{F}_y$, $\mathcal{E}_x$, and x_L are very similar and not shown. Here, we used $N_x = N_y = 40$, and the initial scaling factors: $\beta_x = \beta_y = 2.5$.

domain by considering the function

$$u(x, y, t) = \cos(\frac{xy}{400}) \cdot \frac{1}{1 + \mathrm{e}^{\frac{x-6t-2-t\cos(t)}{2+0.3t}}} \cdot \frac{1}{1 + \mathrm{e}^{\frac{y-4t-2-t\sin(t)}{2+0.4t}}}, \quad x, y, t > 0, \tag{5.4.5}$$

which displays both advective and diffusive behavior. This function exhibits oscillations in space from the factor $\cos(\frac{xy}{400})$, an exponential decay, and a translation to infinity with time-varying velocity $\vec{v} = (v_x, v_y) = (6 + \cos(t), 4 + \sin(t))$. The numerical results shown in Fig. 5.5 are generated using a time step $\Delta t = 0.01$, the same parameters in the x-, y-directions, and $N_x = 40$, $\mu_x = 1.003$, $\delta_x = 0.005$, $d_{max}^x = 0.1$.

As expected, only the combined scaling-moving algorithm 3 keeps the errors in the exterior domain under 10^{-11} (up to the final time $t = 4$), as shown by the error curves in Fig. 5.5(a). This accuracy is achieved because the corresponding frequency indicator and exterior-error indicator are controlled by our "first moving then scaling" techniques, see e.g., $\mathcal{F}_x$ in Fig. 5.5(b) and $\mathcal{E}_y$ in Fig. 5.5(c).

Although the moving algorithm 2 may accurately capture the function near the left end of the exterior domain, the resulting exterior-error indicator does not stay low enough to preserve accuracy in the exterior domain $\Lambda_{\mathrm{e}}^x \times \Lambda_{\mathrm{e}}^y$, as shown by the green curves with asterisks in Figs. 5.5(a, c, d). The moving algorithm neglects the diffusion and thus uses an improper (smaller) x_R and y_R. The right choice for these two variables depends on proper scaling for the diffusion, revealing why we need to update x_R in Line 7 of Alg. 3 after scaling. That is, the moving determines x_L while the scaling determines x_R, making it necessary to combine moving with scaling.

As we have mentioned in Example 4, numerically solving PDEs in unbounded domains requires both the interior solution $U^{\mathrm{interior}}_{x_L(t)}(x,t)$ in $\Lambda_{\mathrm{i}}(t) = (0, x_L(t)]$ and the exterior solution $U^{(\alpha,\beta)}_{N,x_L(t)}(x,t)$ in $\Lambda_{\mathrm{e}}(t) = (x_L(t), +\infty)$ after applying the divide-and-conquer strategy. When using the moving-scaling algorithm 3 to march the solution from t to $t + \Delta t$, if the moving mechanism is not triggered (*i.e.*, x_L is unchanged), then the interior and exterior solutions can be updated individually in the normal way. If it is triggered, extra steps are needed to approximate the solution in the enlarged interior domain $\Lambda_{\mathrm{i}}(t+\Delta t) = \Lambda_{\mathrm{i}}(t) \cup (\Lambda_{\mathrm{e}}(t) \setminus \Lambda_{\mathrm{e}}(t+\Delta t))$ since $x_L(t+\Delta t) = x_L(t) + d_0$ after running Line 12 of Alg. 3.

In the next example, we will test the ability of Alg. 3 to solve a one-dimensional PDE where we will use the intermediate (unmoved) exterior solution $U^{(\alpha,\beta)}_{N,x_L(t)}(x, t+\Delta t)$ (obtained immediately after running Line 8) to interpolate the required function values in $\Lambda_{\mathrm{i}}(t+\Delta t) \setminus \Lambda_{\mathrm{i}}(t)$.

Example 6. We solve the first-order PDE

$$\partial_t u(x,t) + \left(2 + \frac{x-2t}{2+t}\right)\partial_x u(x,t) = 0 \tag{5.4.6}$$

with initial data $u(x,0) = (1+\mathrm{e}^{\frac{x}{2}})^{-1}$ and Dirichlet boundary condition $u(0,t) = (1+\mathrm{e}^{\frac{-2t}{2+t}})^{-1}$. The analytical solution is a moving and spreading Fermi-Dirac distribution: $u(x,t) = (1+\mathrm{e}^{\frac{x-2t}{2+t}})^{-1}$, which travels rightward to infinity at speed 2. A simple numerical scheme for evolving Eq. (5.4.6) is employed here for testing the performance of Alg. 3 within the divide-and-conquer strategy.

Specifically, we adopt the Laguerre spectral approximation (5.2.2) in the exterior domain, the first-order backward finite difference method in the interior domain, and the second-order improved Euler scheme in time. We use a nonuniform mesh, e.g., 10 Gauss-Lobatto points, to avoid possible poor resolution in the tiny interior domain $0 < x_L < d_{\mathrm{max}}$ at short times. For $x_L \geq d_{max}$, a uniform mesh with spacing $\Delta x = \delta = 0.02$ is used so new grid points in $\Lambda_{\mathrm{i}}(t+\Delta t) \setminus \Lambda_{\mathrm{i}}(t)$ can be easily added. The other parameters were set to $N = 40$, $\mu = 1.004$, $d_{max} = 0.2$, and $\Delta t = 0.001$.

The results summarized in Fig. 5.6 clearly show that, up to the final time $t = 5$, the proposed divide-and-conquer strategy maintains the errors in the whole domain $\Lambda = \Lambda_{\mathrm{i}} \cup \Lambda_{\mathrm{e}}$ under 2×10^{-4} (red curve with left-pointing triangles in Fig. 5.6(a)). Alg. 3 succeeds in capturing the translation, as shown by the red curve with left-pointing triangles in Fig. 5.6(b), thus determining the exterior domain Λ_{e}. Without this strategy, a straightforward use of the Laguerre spectral approximation in Λ leads to huge errors as indicated by the blue curve with right-pointing triangles in Fig. 5.6(a).

Fig. 5.6(c) shows that the frequency indicator is always kept under 3×10^{-10} as shown by the black curve with asterisks, a sufficiently small lower error bound for scaling, by continually shrinking the scaling factor shown as the black curve with asterisks in Fig. 5.6(b). The exterior-error indicator is always maintained around 0.2 as shown by the red curve with left-pointing triangles in Fig. 5.6(c), which implies the error in $(x_R, +\infty)$ divided by

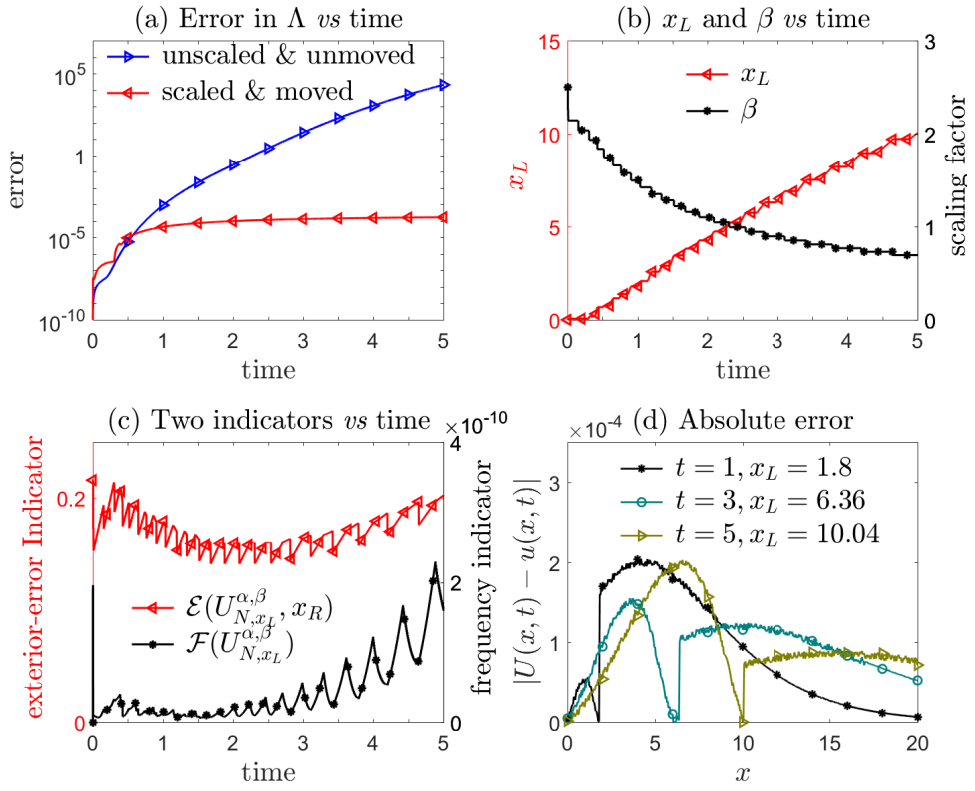

Figure 5.6: Numerical results obtained by the moving-scaling algorithm 3 for the one-dimensional problem in Eq. (5.4.6). The proposed divide-and-conquer strategy maintains the errors in the whole domain $\Lambda = \Lambda_{\rm i} \cup \Lambda_{\rm e}$ under 2×10^{-4} until the final time $t = 5$ where the exterior domain $\Lambda_{\rm e}$ is determined by the "first moving then scaling" technique built in to Alg. 3. We adopt the Laguerre spectral approximation (5.2.2) with $N = 40$ in the exterior domain $\Lambda_{\rm e} = (x_L, +\infty)$, the first order backward finite difference method with spacing $\Delta x = 0.02$ in the interior domain $\Lambda_{\rm i} = (0, x_L]$, and the second order improved Euler time marching scheme with $\Delta t = 0.001$. The last plot displays the absolute difference between the numerical solution $U(x,t)$ and the analytical one $u(x,t)$ at different times.

the error in $\Lambda_{\rm e}$ is almost unchanged, ensuring small errors at infinity. Fig. 5.6(d) plots $|U(x,t) - u(x,t)|$ at different times ($U(x,t)$ and $u(x,t)$ denote the numerical and analytical solution, respectively). There is a clear divide near x_L arising from the different numerical treatments between the interior and exterior domains.

5.5 Performance comparison in solving parabolic PDEs

We now apply the frequency-dependent scaling algorithm 1 to solve

$$\partial_t u(x,t) - \partial_{xx} u(x,t) = f(x,t) \tag{5.5.1}$$

in $\mathbb{R} \times \Lambda$, and compare our results with those obtained with the time-dependent scaling method developed in [MST05]. First, we need to generalize our scaling approach from Λ to $\mathbb{R}$ by using the scaled Hermite functions $\hat{\mathcal{H}}_\ell^{(\beta)}(x) = \mathcal{H}_\ell(\beta x)\mathrm{e}^{-(\beta x)^2/2}$ where $\mathcal{H}_\ell$ are Hermite polynomials [STW11]. Similarly, we use β to denote the *scaling factor* and the frequency indicator defined in Eq. (5.2.6).

A standard Galerkin Hermite spectral method is used to find a solution

$$U_N^{(\beta)} = \sum_{\ell=0}^{N} u_\ell^{(\beta)} \hat{\mathcal{H}}_\ell^{(\beta)}(x) \tag{5.5.2}$$

in $V_N^{(\beta)} = \mathrm{span}\{\hat{\mathcal{H}}_0^{(\beta)}(x), ..., \hat{\mathcal{H}}_N^{(\beta)}(x)\}$ satisfying the initial condition and

$$(\partial_t U_N^{(\beta)}, v) + (\partial_x U_N^{(\beta)}, \partial_x v) = (f, v), \quad \forall\, v \in V_N^{(\beta)}, \tag{5.5.3}$$

where $(\cdot,\cdot)$ is the conventional inner product in $L^2(\mathbb{R})$ space. The Galerkin discretization (5.5.3) is stable in the sense that

$$(\partial_x U_N^{(\beta)}, \partial_x U_N^{(\beta)}) = \sum_{\ell=0}^{N+1} \frac{\ell+1}{2} (u_\ell^{(\beta)})^2 - \sum_{\ell=0}^{N-2} \sqrt{(\ell+1)(\ell+2)}\, u_\ell^{(\beta)} u_{\ell+2}^{(\beta)} \tag{5.5.4}$$

is strictly positive and can be controlled by $(N+1)\|U_N^{(\beta)}\|_2^2 = (N+1)\sum_{\ell=0}^{N}(u_\ell^{(\beta)})^2$. By contrast, a time-dependent scaling factor:

$$\beta(t) = \frac{1}{2\sqrt{\delta_0(\delta t + 1)}} \tag{5.5.5}$$

was taken in [MST05] to fix the instability of the Petrov–Galerkin discretization by tuning the parameters δ_0 and δ.

Example 7. We apply the frequency-dependent scaling algorithm 1 to Example 6.1 in [MST05]. In order to facilitate comparison, we also adopt the same second order-accurate Crank-Nicholson scheme to march Eq. (5.5.3), and the same errors E_N and $E_{N,\infty}$ to measure the accuracy. Table 5.1 presents the numerical errors with different time steps and expansion orders where the second-order accuracy in time and the spectral convergence in space are clearly demonstrated. Table 5.2 compares the errors E_N without scaling to those obtained using the scaling algorithm 1 and the time-dependent scaling method in [MST05] on the same mesh. Both scaling methods produce much more accurate numerical results but the proposed frequency-dependent scaling keeps the errors around or below 10^{-7}, outperforming the time-dependent scaling of [MST05].

The scaling factor adjusted adaptively by the frequency indicator (5.2.6) takes on the value $\beta = 0.5357$ at $t = 1$ for all choices of time steps shown in Table 5.2 whereas the time-dependent scaling factor in [MST05] decreases to $\beta = 0.3536$ at $t = 1$ (Eq. 5.5.5). The smaller scaling factor arises from the stability requirement $\beta'(t) + 2\beta^3(t) \leq 0$, an initial value of 0.5, and using $\delta_0 = \delta = 1$ in Eq. 5.5.5 [MST05], and prevents the error from decreasing when the time step is refined from 1/4000 to 1/16000 (see the third column of Table 5.2). There is no accuracy improvement without scaling when the timestep is decreased as shown in the second column of Table 5.2 where a scaling factor is fixed to $\beta = 0.85$. Regardless of what time step is used in the unscaled method, the error E_N experiences a sudden increase across $t \in [0.3, 0.7]$, rising from below 10^{-6} to about 10^{-4}, as it fails to capture the diffusion. A similar observation was shown in Table 6.1 of [MST05].

5.6 Application to rational basis functions

Apart from the Laguerre and Hermite functions that decay exponentially at infinity, rational basis functions that decay algebraically have been increasingly used [TWY20]. In

Time step	N	$E_N(1)$	Order	$E_{N,\infty}(1)$	Order
10^{-1}	25	2.500e-04		2.182e-04	
10^{-2}		2.499e-07	-2.000	2.227e-06	1.991
10^{-3}		2.500e-09	-2.000	2.227e-08	-2.000
10^{-4}		2.555e-10	-1.991	2.350e-10	-1.977
1/40000	10	2.203e-04		1.619e-04	
	15	2.189e-07	$N^{-16.85}$	4.335e-08	$N^{-20.29}$
	20	1.353e-09	$N^{-17.68}$	8.880e-09	$N^{-13.52}$
	25	4.840e-11	$N^{-14.93}$	6.183e-11	$N^{-11.94}$

Table 5.1: Numerical results for the parabolic problem in Eq. (5.5.1): Errors associated with the frequency-dependent scaling algorithm 1 at $t = 1$ with different time steps and expansion orders N.

Time step	No scaling	Time-dependent scaling in [MST05]	Frequency-dependent scaling in Alg. 1
1/250	3.969e-04	2.598e-06	3.998e-07
1/1000	3.910e-04	1.189e-06	2.503e-08
1/4000	3.390e-04	1.117e-06	2.085e-09
1/16000	3.390e-04	1.117e-06	1.381e-09

Table 5.2: Numerical results for the parabolic problem in Eq. (5.5.1): Comparison of the errors at $t = 1$ with $N = 20$.

this section we generalize our scaling technique to solve a fractional heat equation, the solution of which displays algebraic decay at infinity. We shall use MMGFs [TWY20]: $R_n^{\lambda,\beta}(x) = (1+(\beta x)^2)^{-(\lambda+1)/2} C_n^\lambda(\beta x/\sqrt{1+(\beta x)^2})$ with C_n^λ the Gegenbauer polynomial of order n. In the $|x| \to +\infty$ limit, $R_n^{\lambda,\beta} \sim (\text{sign}(x))^n \frac{(2\lambda)_n}{n!}(1+(\beta x)^2)^{-(1+\lambda)/2}$. We still use β as the scaling factor and define the frequency indicator for the spectral decomposition $U_N^\beta = \sum_{i=0}^N u_i^\beta R_i^{\lambda,\beta}(x)$ in the same way as in Eq. (5.2.6).

Example 8. We numerically solve on $\mathbb{R}$ the fractional heat equation [ZZ17, Yua21]

$$u_t + (-\Delta)^s u = f(x,t), \quad s \in (0,1), \tag{5.6.1}$$

which admits an analytic solution $u(x,t) = ((\frac{x}{t+0.5})^2+1)^{-1/2}$ for an appropriate source function $f(x,t)$. Therefore, we choose MMGFs with $\lambda = 0$ to match the decaying behavior $(1+(\beta x)^2)^{-1/2}$ of the analytic solution. Clearly, the solution is diffusive over time, requiring a decreasing scaling factor β. Fig. 5.7 shows the numerical results for $s = 0.1$, 0.2 and 0.8, where we have adopted the improved Euler scheme in EVOLVE of Alg. 3 and set $\Delta t = 0.005$, $N = 20$, $q = \nu^{-1} = 0.95$, and $\beta_0 = 2$. In Figs. 5.7(c, f, i), it is observed that the scaling factor β matches the intrinsic scaling of the analytic solution and decreases from 2 to about 0.6 over time, during which the errors are well maintained under 10^{-6} for all three fractional orders (red curves in Figs. 5.7(a, d, g)). Failure to adjust β leads to a rapidly increasing frequency indicator (blue curves in Figs. 5.7(b, e, h)) as well as a much larger error (blue curves in Figs. 5.7(a, d, g)).

Comparing the red curves in Figs. 5.7(a, d, g) with those in Figs. 5.7(b, e, h), we confirm the strong correlation between the error and the frequency indicator under these rational MMGF basis functions. We conclude that regardless of s in Eq. (5.6.1), the frequency-dependent scaling is still effective in MMGFs as long as they are able to capture the decaying behavior of the unknown solution at infinity. For functions that display different algebraic decay behavior at infinity, frequency-dependent scaling should also perform well provided an appropriate λ is chosen for the MMGFs to match the algebraic decaying. In short,

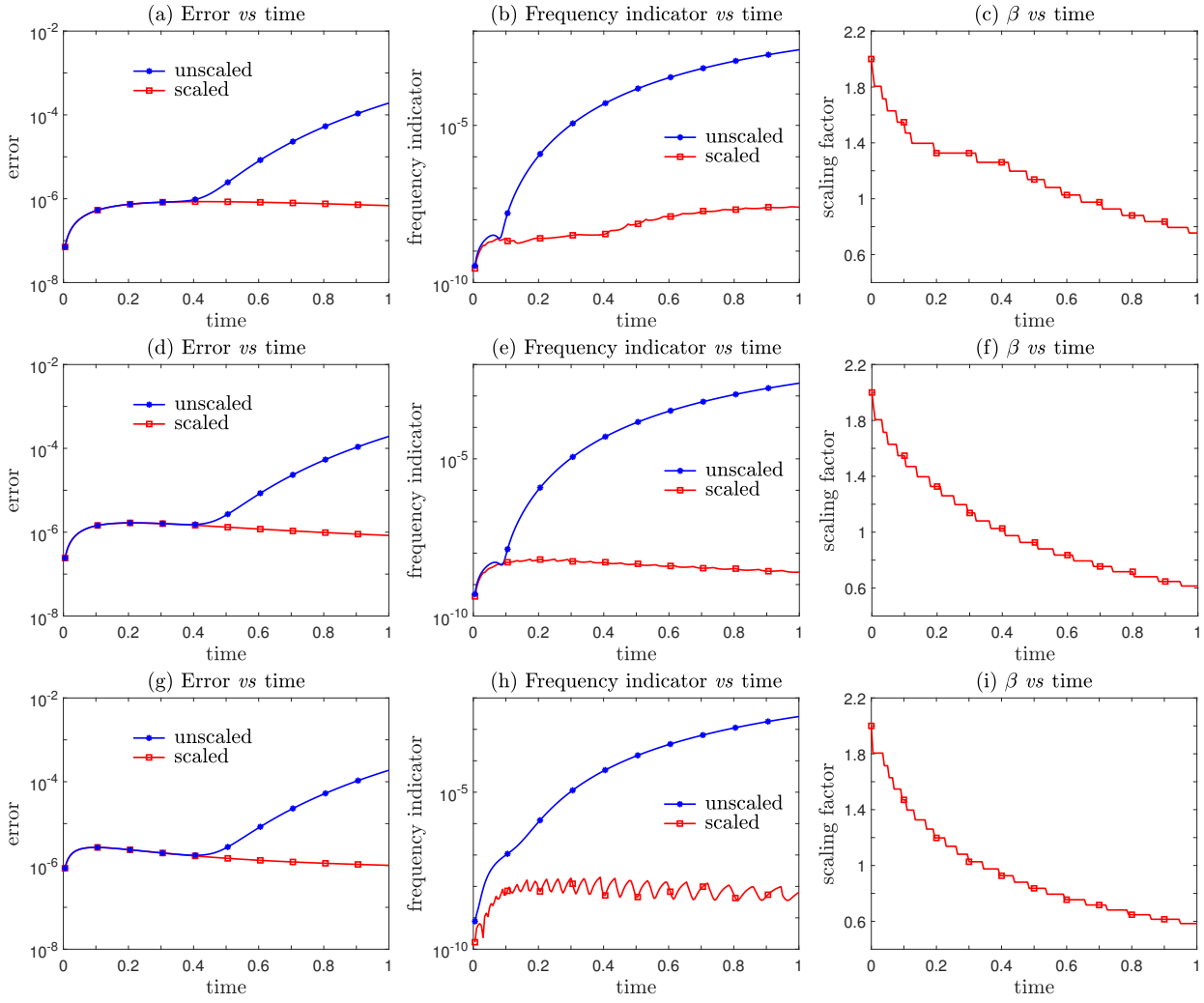

Figure 5.7: Errors, frequency indicators, and scaling factors obtained with MMGFs in solving Eq. (5.6.1) for $s = 0.1$ (first row), 0.5 (second row) and 0.8 (third row). Both the error and the frequency indicator are well maintained under appropriate adjustment of the scaling factor.

the proposed frequency-dependent scaling technique can be successfully generalized to the rational basis functions.

5.7 Analysis of the scaling and moving techniques

In this section, we first illustrate the effectiveness of the frequency-dependent scaling in maintaining small errors. Second, we analyze how the moving technique controls the errors in the exterior domain. Finally, we use two examples to show how both techniques are sensitive to the parameters.

5.7.1 Numerical analysis

The success of the scaling algorithm Alg. 1 is rooted in the connection between the frequency indicator (5.2.6) and the evolution of the information embedded in the numerical solutions. Let $\hat{A}^r_{\alpha,\beta}(\Lambda)$ be the anisotropically weighted Sobolev space. For any integer $r \geq 0$, the seminorm and norm of the solution u are $|u|_{\hat{A}^r_{\alpha,\beta}} = \|\hat{\partial}^r_x u\|_{\hat{\omega}_{\alpha+r}}$ and $\|u\|_{\hat{A}^r_{\alpha,\beta}} = \sqrt{\sum_{k=0}^{r} |u|^2_{\hat{A}^k_{\alpha,\beta}}}$, respectively, where $\hat{\partial}_x u \equiv \partial_x u + \frac{\beta}{2} u$.

For any $u \in \hat{A}^r_{\alpha-1,\beta}(\Lambda) \cap \hat{A}^r_{\alpha,\beta}(\Lambda)$ with integer $r \geq 1$, a direct corollary of Theorem 3.5 in [GWW06] for estimating the interpolation error using the Laguerre functions is

$$\|\mathcal{I}_{N,\alpha,\beta} u - u\|_{\hat{\omega}_\alpha} \leq c(\beta N)^{\frac{1-r}{2}} (\beta^{-1} |u|_{\hat{A}^r_{\alpha-1,\beta}} + (1+\beta^{-\frac{1}{2}})(\ln N)^{\frac{1}{2}} |u|_{\hat{A}^r_{\alpha,\beta}}), \tag{5.7.1}$$

where c denotes a generic positive constant that does not depend on α, β, N, or any function. This error estimate plays a crucial role in the formal development and successful implementation of our scaling and moving techniques. For simplicity, we only consider diffusive behavior without translation, set $x_L = 0$, and drop the subscript x_L.

There are two reasons to use the frequency indicator given in Eq. (5.2.6). Starting with

$M = [\frac{n}{3}]$ and a sufficiently large expansion order N, we have

$$\begin{aligned}\frac{1}{2}\mathcal{F}(U_N^{(\alpha,\beta)}) &\approx \frac{1}{2}\frac{\|\mathcal{I}_{N,\alpha,\beta}u - \mathcal{I}_{N-M,\alpha,\beta}u\|_{\hat{\omega}_\alpha}}{\|\mathcal{I}_{N,\alpha,\beta}u\|_{\hat{\omega}_\alpha}} \\ &\leq \frac{1}{2}\frac{\|u - \mathcal{I}_{N,\alpha,\beta}u\|_{\hat{\omega}_\alpha} + \|u - \mathcal{I}_{N-M,\alpha,\beta}u\|_{\hat{\omega}_\alpha}}{\|\mathcal{I}_{N,\alpha,\beta}u\|_{\hat{\omega}_\alpha}} \leq \frac{\|u - \mathcal{I}_{N-M,\alpha,\beta}u\|_{\hat{\omega}_\alpha}}{\|\mathcal{I}_{N,\alpha,\beta}u\|_{\hat{\omega}_\alpha}}, \end{aligned} \tag{5.7.2}$$

which provides an estimate to the lower bound of $\frac{\|u-\mathcal{I}_{N-M,\alpha,\beta}u\|_{\hat{\omega}_\alpha}}{\|\mathcal{I}_{N,\alpha,\beta}u\|_{\hat{\omega}_\alpha}}$. Minimizing $\mathcal{F}(U_N^{(\alpha,\beta)})$ in Alg. 1 may reduce this lower bound for the relative error. Moreover, a straightforward application of the interpolation error estimator (5.7.1) to the two interpolations in the numerator of the first term of Eq. (5.7.2) yields

$$\left(\sum_{\ell=N-M+1}^{N}\gamma_\ell^{(\alpha,\beta)}(u_\ell^{(\alpha,\beta)})^2\right)^{1/2} \leq c_F(\beta N)^{\frac{1-r}{2}}\left(\beta^{-1}|u|_{\hat{A}^r_{\alpha-1,\beta}} + (1+\beta^{-\frac{1}{2}})(\ln N)^{\frac{1}{2}}|u|_{\hat{A}^r_{\alpha,\beta}}\right), \tag{5.7.3}$$

where the constant $c_F \equiv (1+2^{\frac{r-1}{2}})c$. Thus, we find

$$\mathcal{F}(U_N^{(\alpha,\beta)}) \leq c_F(\beta N)^{\frac{1-r}{2}}\left(\beta^{-1}\frac{|u|_{\hat{A}^r_{\alpha-1,\beta}}}{\|U_N^{(\alpha,\beta)}\|_{\hat{\omega}_\alpha}} + (1+\beta^{-\frac{1}{2}})(\ln N)^{\frac{1}{2}}\frac{|u|_{\hat{A}^r_{\alpha,\beta}}}{\|U_N^{(\alpha,\beta)}\|_{\hat{\omega}_\alpha}}\right), \tag{5.7.4}$$

implying that $\forall\,\varepsilon \in (0,1)$, we may choose a sufficiently large N such that $\mathcal{F}(U_N^{(\alpha,\beta)}) < \varepsilon$. This shows one rationale for using the frequency indicator (5.2.6).

The second reason the frequency indicator $\mathcal{F}(U_N^{(\alpha,\beta)})$ can be used to measure the decay of the reference solution's derivatives as $x \to \infty$ tends to infinity is argued as follows. According to the inequality (5.7.4), if $|u|_{\hat{A}^r_{\alpha-1,\beta}}/\|U_N^{(\alpha,\beta)}\|_{\hat{\omega}_\alpha}$ is fixed, a larger $\mathcal{F}(U_N^{(\alpha,\beta)})$ implies a larger $|u|_{\hat{A}^r_{\alpha,\beta}}/\|U_N^{(\alpha,\beta)}\|_{\hat{\omega}_\alpha}$. Given any $s \in \Lambda$ (*e.g.*, $s = \sqrt{2x_N^{(\alpha,\beta)}}$), and if

$$\mathcal{F}(U_N^{(\alpha,\beta)}) > c_F(\beta N)^{\frac{1-r}{2}}\frac{|u|_{\hat{A}^r_{\alpha-1,\beta}}}{\|U_N^{(\alpha,\beta)}\|_{\hat{\omega}_\alpha}}(\beta^{-1} + s(1+\beta^{-\frac{1}{2}})(\ln N)^{\frac{1}{2}}), \tag{5.7.5}$$

then

$$\int_0^{\frac{s^2}{2}}(\hat{\partial}_x^r u(x))^2 x^{\alpha+r}\mathrm{d}x < \int_{\frac{s^2}{2}}^{+\infty}(\hat{\partial}_x^r u(x))^2 x^{\alpha+r}\mathrm{d}x. \tag{5.7.6}$$

The inequality (5.7.6) can be verified by contradiction. First, combine Eqs. (5.7.4) and (5.7.5) to find

$$s|u|_{\hat{A}^r_{\alpha-1,\beta}} < |u|_{\hat{A}^r_{\alpha,\beta}}. \tag{5.7.7}$$

If (5.7.6) does not hold, we would find

$$\begin{aligned}|u|^2_{\hat{A}^r_{\alpha,\beta}} &= \int_0^{+\infty} (\hat{\partial}^r_x u(x))^2 x^{\alpha+r} \mathrm{d}x \le 2\int_0^{\frac{s^2}{2}} (\hat{\partial}^r_x u(x))^2 x^{\alpha+r} \mathrm{d}x \\ &\le 2\cdot\frac{s^2}{2}\int_0^{\frac{s^2}{2}} (\hat{\partial}^r_x u(x))^2 x^{\alpha+r-1} \mathrm{d}x \le s^2 \int_0^{+\infty} (\hat{\partial}^r_x u(x))^2 x^{\alpha+r-1} \mathrm{d}x = s^2 |u|^2_{\hat{A}^r_{\alpha-1,\beta}},\end{aligned}$$

which would contradict the inequality (5.7.7). Intuitively, basis functions of higher degree decay more slowly than those of lower degree, so an increase in the frequency indicator implies slower decay at infinity. This slower spatial decay as time increases requires the use of a larger computational domain which is achieved by decreasing β. In other words, as the frequency indicator increases, the norm of $\hat{\partial}^r_x u(x)\cdot \mathbb{I}_{(s^2/2,+\infty)}(x)$ becomes larger than that of $\hat{\partial}^r_x u(x)\cdot \mathbb{I}_{(0,s^2/2)}(x)$, implying scaling is indeed needed to enlarge the computational domain because $\|\hat{\partial}^r_x u\cdot \mathbb{I}_{(x>s^2/2)}\|_{\hat{\omega}_\alpha}$ is the dominant component of $\|\hat{\partial}^r_x u\|_{\hat{\omega}_\alpha}$.

Next, we show that increasing x_L in the moving technique can control the errors in the exterior domain when generalized Laguerre functions are adopted. After increasing x_L to x_L+d, $U^{(\alpha,\beta)}_{N,x_L}(x)e^{\beta x/2}$ and $U^{(\alpha,\beta)}_{N,x_L+d}(x)e^{\beta x/2}$ are two identical polynomials of order N since they pass through the same $N+1$ different points: $(x_i^{(\alpha,\beta)}+d,\ U^{(\alpha,\beta)}_{N,x_L}(x_i^{(\alpha,\beta)}+d)e^{\beta(x_i^{(\alpha,\beta)}+d)/2})$, $i=0,...,N$, *i.e.*, $U^{(\alpha,\beta)}_{N,x_L+d}(x) = U^{(\alpha,\beta)}_{N,x_L}(x)$ for any $x\in(x_L+d,\infty)$. Thus,

$$\|U^{(\alpha,\beta)}_{N,x_L+d}(x,t)\mathbb{I}_{(x>x_L+d)}\|^2_{\hat{\omega}_\alpha} = \|U^{(\alpha,\beta)}_{N,x_L}(x,t)\mathbb{I}_{(x>x_L+d)}\|^2_{\hat{\omega}_\alpha} < \|U^{(\alpha,\beta)}_{N,x_L}(x,t)\mathbb{I}_{(x>x_L)}\|^2_{\hat{\omega}_\alpha}, \tag{5.7.8}$$

$$\|(u(x,t)-U^{(\alpha,\beta)}_{N,x_L}(x,t))\mathbb{I}_{(x>x_L+d)}\|_{\hat{\omega}_\alpha} \le \|(u(x,t)-U^{(\alpha,\beta)}_{N,x_L}(x,t))\mathbb{I}_{(x>x_L)}\|_{\hat{\omega}_\alpha}. \tag{5.7.9}$$

That is, both the norm of $U^{(\alpha,\beta)}_{N,x_L}$ and the error $\|(u(x)-U^{(\alpha,\beta)}_{N,x_L})\mathbb{I}_{(x>x_L)}\|_{\hat{\omega}_\alpha}$ will not increase as

x_L increases. Furthermore, we consider solving a time-dependent PDE

$$u_t(x,t) = D_x(t)u(x,t), \quad x,t \in \Lambda \tag{5.7.10}$$

where $D_x(t)$ represents a differential operator that only involves spatial derivatives. Let $U^{(\alpha,\beta)}_{N,x_L}(x,t_n)$ denote the numerical solution at t_n. We show below that there exists $d \geq 0$ such that

$$\|U^{(\alpha,\beta)}_{N,x_L+d}(x,t_{n+1})\mathbb{I}_{(x>x_L+d)}\|_{\hat{\omega}_\alpha} \leq \|U^{(\alpha,\beta)}_{N,x_L}(x,t_n)\mathbb{I}_{(x>x_L)}\|_{\hat{\omega}_\alpha}. \tag{5.7.11}$$

Denote $\mathbf{u}_{N,x_L} = (U^{(\alpha,\beta)}_{N,x_L}(x_0^{(\alpha,\beta)}), ..., U^{(\alpha,\beta)}_{N,x_L}(x_N^{(\alpha,\beta)}))^T$ and define the translation operator matrix $T_N^{(\alpha,\beta)}(s)$ such that $T_N^{(\alpha,\beta)}(s)\mathbf{u}_{N,x_L} = (U^{(\alpha,\beta)}_{N,x_L}(x_0^{(\alpha,\beta)}+s), ..., U^{(\alpha,\beta)}_{N,x_L}(x_N^{(\alpha,\beta)}+s))^T$. It is easy to see that $T_N^{(\alpha,\beta)}(0)$ is the identity matrix and that $T_N^{(\alpha,\beta)}(s+t) = T_N^{(\alpha,\beta)}(s)T_N^{(\alpha,\beta)}(t)$ for $s,t \in \Lambda$. Thus, $T_N^{(\alpha,\beta)}(s \in \Lambda)$ forms a semigroup, has generator $\hat{\mathcal{L}}^{(\alpha,\beta)}_{N,p} \coloneqq \lim_{s\to\infty} \frac{T_N^{(\alpha,\beta)}(s) - T_N^{(\alpha,\beta)}(0)}{s}$, and is expressed as $T_N^{(\alpha,\beta)}(s) = \mathrm{e}^{s\hat{\mathcal{L}}^{(\alpha,\beta)}_{N,p}}$. Since the Laguerre functions tend to 0 at $+\infty$, $\lim_{s\to+\infty} T_N^{(\alpha,\beta)}(s) = 0$, indicating that $\|\mathrm{e}^{\hat{\mathcal{L}}^{(\alpha,\beta)}_{N,p}}\|_{\hat{w}^{\alpha,\beta}} < 1$ where the matrix norm $\|\cdot\|_{\hat{w}^{\alpha,\beta}}$ is induced from the vector norm $\|\mathbf{u}_{N,x_L}\|^2_{\hat{w}^{\alpha,\beta}} \coloneqq \sum_{\ell=0}^N U^2_{N,x_L}(x_\ell^{(\alpha,\beta)})\hat{w}_\ell^{(\alpha,\beta)}$

After discretizing $D_x(t)$ in Eq. (5.7.10) with some numerical scheme, we have $\mathbf{u}_{N,x_L}(t_{n+1}) = D_N^{(\alpha,\beta)}(t_n)\mathbf{u}_{N,x_L}(t_n)$, $D_N^{(\alpha,\beta)}(t_n) \in \mathbb{R}^{(N+1)\times(N+1)}$. We choose

$$d = -\frac{\max\{0, \ln\|D_N^{(\alpha,\beta)}(t_n)\|_{\hat{w}^{\alpha,\beta}}\}}{\ln\|e^{\hat{\mathcal{L}}^{(\alpha,\beta)}_{N,p}}\|_{\hat{w}^{\alpha,\beta}}}, \tag{5.7.12}$$

let $\mathbf{u}_{N,x_L+d}(t_{n+1}) \coloneqq T_N^{(\alpha,\beta)}(d)\mathbf{u}_{N,x_L}(t_{n+1})$, and verify that

$$\|\mathbf{u}_{N,x_L+d}(t_{n+1})\|_{\hat{w}^{\alpha,\beta}} \leq \|\mathbf{u}_{N,x_L}(t_{n+1})\|_{\hat{w}^{\alpha,\beta}}, \tag{5.7.13}$$

which directly gives Eq. (5.7.11) using $\|\mathbf{u}_{N,x_L}(t)\|_{\hat{w}^{\alpha,\beta}} = \|U^{(\alpha,\beta)}_{N,x_L}(x,t)\mathbb{I}_{(x>x_L)}\|_{\hat{\omega}_\alpha}$.

5.7.2 Sensitivity analysis

The scaling in Alg. 1 relies on two parameters q and ν, while the moving in Alg. 2 requires δ and μ. Below use two examples to explore how these parameters affect algorithmic performance and to develop intuition for how to set them.

Example 9. First, we investigate the scaling technique's dependence on q and ν by solving

$$\begin{cases} \dfrac{\partial u}{\partial t} + \dfrac{x-5}{2+t} \cdot \dfrac{\partial u}{\partial x} = 0, \\ \quad u(x,0) = \dfrac{1}{1+\mathrm{e}^{\frac{x-5}{2}}}. \end{cases} \tag{5.7.14}$$

Here, the analytic solution is known: $u(x,t) = (1+\mathrm{e}^{\frac{x-5}{2+t}})^{-1}$. Since it exhibits diffusion numerically approximating it using scaling requires a decreasing β. Table 5.3 lists scaling factors β (bottom-left) and error (upper-right) at $t = 10$ for various ν and q. We have set $\Delta t = 0.004$, $\beta_0 = 3$ and $N = 24$, and called the improved Euler scheme in EVOLVE of Alg. 1. Given the same q, the numerical solution tends to be less accurate for larger ν because if we lift the scaling threshold, the time to invoke scaling will be delayed and errors will accumulate. On the other hand, if we compare errors in each row of Table 5.3, we see that with fixed ν, larger q will yield better results since adjusting the scaling factor in a more "continuous" manner prevents abrupt, possibly delayed, over-adjustment of the scaling factor and allowing the evolution of β to be well-matched to the diffusive evolution of the solution. Thus, we argue that setting $q \lesssim 1$ and $\nu \gtrsim 1$ in the frequency-dependent scaling technique will yield good results.

Example 10. In this example, we will solve

$$\begin{cases} \partial_t u + 2\partial_x u = 0, \\ u(x,0) = \dfrac{1}{1+\mathrm{e}^{\frac{x}{2}}}, \end{cases} \tag{5.7.15}$$

to test how different choices of μ and δ affect Alg. 2's ability to numerically capture the translation of the known analytic solution $u(x,t) = (1+\mathrm{e}^{\frac{x-2t}{2}})^{-1}$, which moves right with

ν \ q	0.6	0.8	0.9	0.95
1.05	5.7526e-07 0.3888	5.8297e-07 0.3221	5.2818e-07 0.2659	5.2500e-07 0.3663
q^{-1}	6.9708e-07 0.2333	5.8293e-07 0.3221	5.3177e-07 0.2954	5.2488e-07 0.3663
$\sqrt{2}$	6.4207e-07 0.3888	5.8117e-07 0.2557	5.2836e-07 0.2954	5.2640e-07 0.3855
$\sqrt{3}$	7.1155e-07 0.3888	6.1260e-07 0.3221	5.4353e-07 0.3283	5.2370e-07 0.3855
2	7.5526e-07 0.3888	6.5640e-07 0.3221	5.5429e-07 0.3647	5.2492e-07 0.4058

Table 5.3: Errors (upper-right) and scaling factors (bottom-left) at $t = 10$ for different ν and q in solving Eq. (5.7.14) with Alg. 1. A smaller ν facilitates scaling and results in more timely scaling and a smaller error, while a larger q can scale the basis functions in a more "continuous" manner, allowing the scaling factor to match the diffusion of the solution. That is, setting $q \lesssim 1$ and $\nu \gtrsim 1$ will be a good choice for the scaling technique.

μ \ δ	0.0005	0.001	0.002	0.005
1.0001	2.0687e-07 9.9985	2.0687e-07 9.9990	2.0687e-07 10.000	8.3063e-07 25.000
1.0002	5.2547e+12 3.0320	2.0687e-07 9.9990	2.0687e-07 10.000	8.3063e-07 25.000
1.0005	8.9856e+16 0.3635	9.3629e+15 1.0420	2.0688e-07 9.9980	8.3062e-07 24.995
1.001	1.1390e+17 0.1745	6.2077e+16 0.3650	5.3087e+15 1.0880	8.3061e+8 24.990

Table 5.4: Errors in Λ_e (upper-right) and x_L (bottom-left) at $t = 5$ for solving Eq. (5.7.15) using different μ and δ and a fixed $d_{\max} = 0.05$ in Alg. 2. Increasing μ renders the moving less sensitive to translation and results in a smaller x_L given the same δ. On the other hand, too large δ increases x_L excessively, leading to unnecessary additional computational cost. Thus, our guideline is to set $\mu \gtrsim 1$ and $\delta \ll 1$ for the exterior-error-dependent moving algorithm.

speed 2. Table 5.4 lists x_L (bottom-left) and errors in Λ_e (upper-right) for various μ and δ at $t = 5$. We have set $N = 40$, $\Delta t = 0.001$, and $\beta_0 = 3$ and invoked the improved Euler scheme in EVOLVE of Alg. 2. For the purpose of testing, we have fixed $d_{\max} = 0.05$, which is much larger than the actual translation of 0.002 associated with a single time step. According to the numerical results reflected in Table 5.4, we see that when fixing δ, increasing μ makes the moving less sensitive so that it eventually fails to capture the translation and leads to large errors.

On the other hand, by fixing μ and comparing x_L in each row for different δ, we discovered that increasing δ allows x_L to increase more in a single timestep, leading to larger values of x_L. Yet when μ is too small or when δ is too large, the moving mechanism may generate an

x_L which is larger than the actual translation of the solution. This results in an unnecessarily large interior domain $(0, x_L)$ requiring more nodes in Λ_{i} in order to find an accurate numerical solution. Hence, to achieve a more accurate reflection of translation, we propose setting $\mu \gtrsim 1$ and $\delta \ll 1$ in Alg. 2.

5.8 Applications to structured cell population models

One example of an application requiring the solution of PDEs in an unbounded domain is the structured population models that track populations of cells endowed with attributes such as their size. The standard sizer-timer model for the density of cells with age near a and size near x is formulated in [MD86], and generalizations to include stochasticity in growth rate is studied in [STH11, Cas05]. Here we address a continuum model describing a stochastic model for cell populations [XC21]:

$$\frac{\partial n}{\partial t} + \frac{\partial n}{\partial a} + \frac{\partial (ng)}{\partial x} - \frac{1}{2}\frac{\partial^2(\sigma^2 n)}{\partial x^2} = -D(x,a,t)n(x,a,t), \quad (x,a) \in \Lambda \times \Lambda, \tag{5.8.1}$$

where $n(a,x,t)$ describes the density of cells with respect to age a and size x at time t, $g(a,x,t)$ is the mean growth rate of an individual cell and $\sigma^2(a,x,t)$ is the variance of stochasticity in the growth rate, *i.e.*, $\mathrm{d}x = g\mathrm{d}t + \sigma \mathrm{d}B_t$, for an individual cell. The fluctuating growth rate manifests itself as a diffusive term. The right-hand-side of Eq. (5.8.1) represents cell division occurring with division rate $D(x,a,t)$. Dirichlet boundary conditions are imposed at $x = 0$, $n(0,a,t) = n_0(a,t)$, and at $x = +\infty$, $n(+\infty,a,t) = 0$ if we assume that there are no cells of infinite size. More importantly, the boundary condition at $a = 0$ should account for two daughter cells (one of size x and one of size $y - x$) from the binary fission of a mother cell of size $y > x$:

$$n(x,0,t) = 2\int_0^{+\infty} \mathrm{d}a \int_x^{+\infty} \mathrm{d}y\, \tilde{D}(y,x,a,t)n(y,a,t), \tag{5.8.2}$$

where $\tilde{D}(y, x, a, t)$ is the differential division rate representing the rate that a cell of age a and size y gives birth to a daughter cell of size $x < y$. Integrating over the daughter cell's size x, D and $\tilde{D}$ satisfy $D(y, a, t) = \int_0^y \tilde{D}(y, x, a, t)\mathrm{d}x$, reflecting cell number conservation. Finally, to maintain biomass conservation during division, $\tilde{D}(y, x, a, t) = \tilde{D}(y, y - x, a, t)$. The prefactor 2 in Eq. (5.8.2) indicates that a cell of size y gives birth to one daughter cell of size $y - x$ and another of size x.

The nonlocal boundary condition Eq. (5.8.2) for cell proliferation plays an essential role in depicting how cell division affects the cell population size and age structure, and presents a major obstacle in numerical computation as the integration is taken in the unbounded domain $(x, +\infty) \times (0, +\infty)$. Another numerical challenge arises from a possible "blowup" behavior in which

$$\lim_{t\to+\infty} \langle x(t)\rangle = \frac{\int_0^{+\infty}\int_0^{+\infty} xn(a, x, t)\mathrm{d}a\mathrm{d}x}{\int_0^{+\infty}\int_0^{+\infty} n(a, x, t)\mathrm{d}a\mathrm{d}x} = +\infty. \tag{5.8.3}$$

Whether blowups can occur is of biological interest [KB18, XGC20] and has been predicted within certain cell proliferation models Eq. (5.8.1) under specific conditions [KB18].

Existing numerical methods such as the finite volume method in [XGC20] typically truncate the unbounded domain into a bounded domain and therefore cannot accurately capture long time blowup behavior of $\langle x(t)\rangle$. The need for numerical solutions in the unbounded domain $\Lambda \times \Lambda$ for Eqs. (5.8.1) and (5.8.2) is thus evident. We apply the scaling technique built in to Alg. 1 only in x-dimension for tracking the increasing $\langle x(t)\rangle$, considering the age distribution is often presumed to be stable since no cell could live too long without division. A standard two-dimensional pseudo-spectral method with the generalized Laguerre functions are used in (a, x)-space, coupled with a third-order TVD Runge-Kutta time discretization in t.

Example 11. We solve Eqs. (5.8.1) and (5.8.2) with $g(x, a, t) = t+7$, $\sigma^2(x, a, t) = 2(t+6)x$, $D(x, a, t) = x/(t + 5)$, $\tilde{D}(y, x, a, t) = 1/(t + 5)$. These parameters leads to the analytic solution $n(x, a, t) = e^t e^{-2a} \exp(-x/(5 + t))$, which produces the mean size $\langle x(t)\rangle = 5 + t$.

This result shows that the average size is unbounded as it grows linearly in time and thus, for general cases, requires proper scaling in x-dimension. Here we adopt the same expansion order N in both size x- and age a-dimensions. For the nonlocal boundary condition given in Eq. (5.8.2), we also use $N+1$ Laguerre-Lobatto collocation points in each dimension to perform the numerical integration.

Fig. 5.8 presents the numerical results with the initial scaling factors $(\beta_x, \beta_a) = (0.9, 1)$ and a timestep of 0.002. We observe that the frequency-dependent scaling algorithm 1 in x-dimension shows a faster spectral convergence with N than that of the unscaled algorithm (see Fig. 5.8(a)). That is, both the sizer-timer model (5.8.1) in the unbounded domain and the nonlocal boundary condition Eq. (5.8.2) are well resolved by the Laguerre spectral approximation with frequency-dependent scaling. When fixing $N = 20$, the unscaled numerical solution experiences an error growth to 1.143e-02 till $t = 10$ for using inappropriate scaling factors, whereas the error of the scaled solution is less than 8.662e-06 (see Fig. 5.8(b)). The frequency indicator in the x-dimension is kept around 10^{-6} (red curve with left-pointing triangles in Fig. 5.8(c)) by continuously shrinking the scaling factor β_x from 0.9 to 0.2766 for tracking the blowup (black curve with asterisks in Fig. 5.8(d)). The average size of the scaled solution behaves almost exactly like $\langle x(t) \rangle = 5 + t$ and the value at $t = 10$ is 15.001 (see red curve with left-pointing triangles in Fig. 5.8(d)). Note that the scaling in a-dimension will really not be triggered even when we apply the scaling algorithm for both x- and a-dimensions.

5.9 Summary and conclusions

The key to making spectral approximations in unbounded domains more efficient is to allocate collocation points in an economical manner such that crucial regimes of unknown solutions can be resolved accurately. This is essentially an adaptive numerical method for PDEs in unbounded domains, for which there are very few studies compared with its bounded-domain counterpart. Using the standard language of adaptive methods, we proposed a

scaling technique based on the frequency indicator, which can be regarded as r-adaptivity, since collocation points are redistributed through the evolution of a scaling factor.

We also proposed a moving technique based on the exterior-error indicator which is similar to h-adaptive methods since collocation points are added to the interior subdomain. Both indicators utilize only the numerical solution and do not require prior knowledge of unknown solutions.

It will be promising in future work to generalize the scaling and moving techniques to the hyperbolic cross space [SW10a] which may greatly reduce the cost of higher dimensional simulations. Promising future directions involve performing a more rigorous numerical analysis of the proposed techniques as well as providing a more detailed discussion on how to choose key parameters when numerically solving different problems. Finally, apart from the proposed scaling and moving techniques, a p-adaptive technique [XSC21b] which can be applied to time-dependent problems with oscillatory behavior will be further explored.

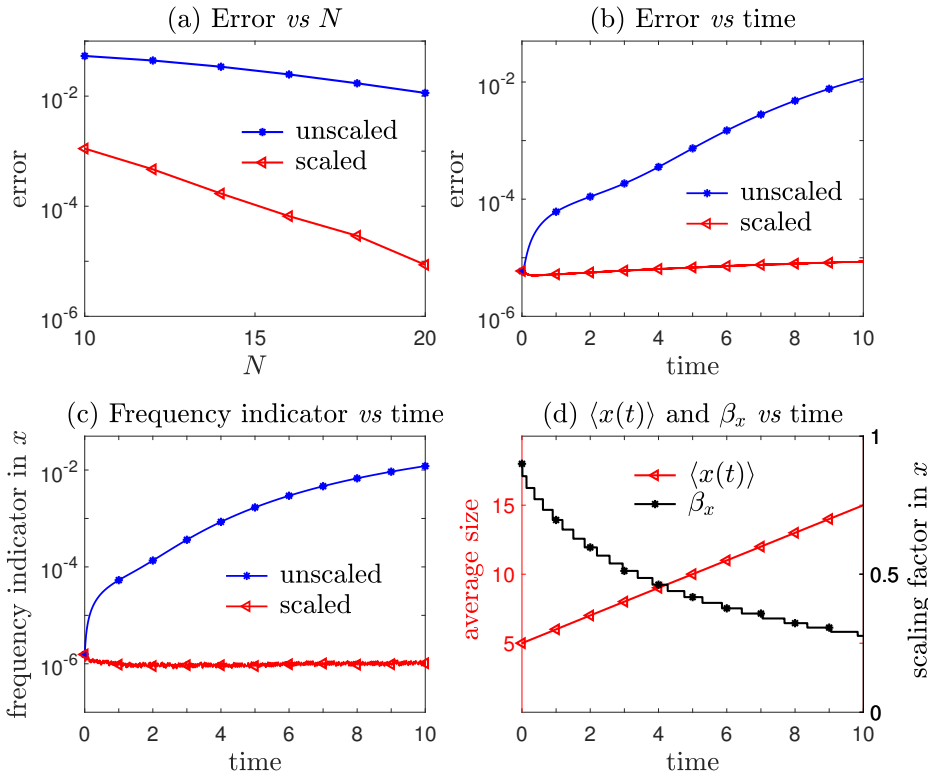

Figure 5.8: Numerical results obtained by the scaling algorithm Alg. 1 for the structured cell population proliferation model Eq. (5.8.1) with the nonlocal boundary Eq. (5.8.2): The scaled method gives better results than the unscaled one till $t = 10$. The latter experiences a growth in error because inappropriate scaling factors are used, whereas the former gains a faster spectral convergence in the expansion order N. We adopt the same N in both size x- and age a-dimensions and set $N = 20$ for the last three plots. The frequency-dependent scaling is applied only in x-dimension for tracking the blowup behavior in Eq. (5.8.3). The frequency indicator in x-dimension is kept around 10^{-6} through constantly shrinking the scaling factor β_x to capture the blowup. The average size of the scaled solution is in good agreement with that of the analytical solution, *i.e.*, $\langle x(t)\rangle = 5 + t$.

CHAPTER 6

A frequency-dependent p-adaptive technique for spectral methods

This is the Accepted Manuscript version of an article accepted for publication in *Journal of Computational Physics*, **446**, pp. 110627, (2021). It is an open-access paper. The Version of Record is available online at [10.1016/j.jcp.2021.110627].

6.1 Introduction

Unbounded domain problems arise in many scientific applications [Tsy98, XGC20] and adaptive numerical methods are needed on many occasions, for instance, in solving the Schrödinger equation in unbounded domains when the solution's behavior varies over time and we wish to capture the solution's behavior in the whole domain. As an important class of numerical algorithms, adaptive methods have witnessed numerous advances in their efficiency and accuracy [TT03, BFH12, RW00, LLM02]. However, despite considerable progress that has been made for spectral methods in unbounded domains [SW09], there are few adaptive methods that apply in unbounded domains.

In [XSC21b], adaptive scaling and moving techniques were proposed for spectral methods in unbounded domains and it was noted that adjusting the expansion order is necessary when the function displays oscillatory behavior that varies over time. In this chapter, we first develop a frequency-dependent technique for spectral methods which adjusts the expansion order N ($N+1$ basis functions are used to approximate the solution). This technique takes advantage of the frequency indicator defined in [XSC21b] and corresponds to p-adaptivity [SLC11, DKT07, KX01, CNS17, ACV19]. By adjusting the expansion order efficiently, our p-adaptive technique can be used to accurately solve problems with varying oscillatory behavior.

Although the work reported in [XSC21b] motivated us to develop the p-adaptive technique using the same frequency indicator for the numerical solutions, the work reported here complements, but is fundamentally different from that in [XSC21b]. The biggest difference is that the expansion order in [XSC21b] is fixed, while in this work it is adaptively adjusted according to the dynamic behavior of unknown functions. For instance, as demonstrated in Examples 17 and 18 and in the subsequent discussions, solutions to Schrödinger equations that become increasingly oscillatory over time cannot be well approximated by the previously proposed scaling and moving techniques in [XSC21b]. On the other hand, the p-adaptive technique, which adjusts the expansion order alone, cannot successfully deal with diffusing

and advecting solutions, which are captured by the moving and scaling techniques described in [XSC21b], as shown in Examples 15, 16, and 19. Thus, an efficient and more complete adaptive spectral method should integrate moving, scaling, and p-adaptive techniques as charted in Fig. 6.5.

By combining this p-adaptive technique with scaling and moving methods, we develop an adaptive spectral method that can capture diffusion, advection, and oscillations in unbounded domains. Since scaling and adjusting the expansion order both depend on the frequency indicator, we also investigate the interdependence of these two techniques. We demonstrate that appropriately adjusting the expansion order can facilitate scaling to more efficiently distribute allocation points. In turn, proper scaling can help avoid unnecessary increases in the expansion order when it does not increase accuracy, thereby avoiding unnecessary computational burden.

The significance of this adaptive spectral method is that it can capture the solution's behavior in the whole domain. We demonstrate the utility of our method by solving Schrödinger's equation in $\mathbb{R}$. Here, the unboundedness and the oscillatory nature of the solution pose two major numerical challenges [YZ14]. Specifically, in the semiclassical regime, when the wavelength of the solution is small, the function becomes extremely oscillatory. Moreover, in certain situations, one has to work with a very large computational domain that is difficult to automatically determine.

Previous numerical methods which solve Schrödinger's equation in unbounded domains usually truncate the domain into a finite subdomain and impose artificial boundary conditions, which may be nonlocal and complicated [HJW05, YZ14, LZZ18, AAB08]. Our adaptive spectral method tackles the oscillatory problem directly in the original unbounded domain without the need to truncate it or to devise an artificial boundary condition.

This chapter is organized as follows. In the next section, we first present a p-adaptive technique for spectral methods and use examples to illustrate its efficiency. In Section 6.3, we incorporate and study this technique within existing scaling and moving techniques and devise an adaptive spectral method in unbounded domains. Application of our adaptive

spectral methods to numerically solving Schrödinger's equation is given in Section 6.4. We summarize our results in Section 6.5 and propose directions for future work.

6.2 Frequency-dependent p-adaptivity

We present a frequency-dependent p-adaptive spectral method based on information extracted from only the numerical solution of time-dependent problems. In [XSC21b], we showed that a frequency indicator defined for spectral methods is particularly useful in measuring the contribution of high-frequency modes in the numerical solution. Because high-frequency modes decay more slowly, this indicator could be used to determine scaling in spectral methods applied to unbounded domains.

In this work, we will show that the frequency indicator can also be used to determine whether more or fewer basis functions are needed to refine or coarsen the numerical solution. Proper refining allows one to maintain accuracy when the solution becomes more oscillatory, while appropriate coarsening can reduce computational costs without sacrificing accuracy.

Given a set of orthogonal basis functions $\{B_i(x)\}_{i=0}^{\infty}$ under a specific weight function $\omega(x) > 0$ in a domain Λ, the frequency indicator associated with the interpolation of a function

$$\mathcal{I}_N u(x) = U_N(x) = \sum_{i=0}^{N} u_i B_i(x) \tag{6.2.1}$$

is defined as in [XSC21b]

$$\mathcal{F}(U_N) \coloneqq \left(\frac{\sum_{i=N-M+1}^{N} \gamma_i u_i^2}{\sum_{i=0}^{N} \gamma_i u_i^2} \right)^{\frac{1}{2}}, \tag{6.2.2}$$

where $\gamma_i = \int_\Lambda B_i^2(x)\omega(x)\mathrm{d}x$ is the square of L^2_ω-weighted norm of the basis function $B_i(x)$. This frequency indicator measures the contribution of the M highest-frequency components to the L^2_ω-weighted norm of U_N. Here M is often chosen to be $[\frac{N}{3}]$ following the $\frac{2}{3}$-rule [HL07, Ors71]. This indicator provides a lower bound for the error divided by the norm of

the numerical solution $\frac{\|u-\mathcal{I}_{N-M}u\|_\omega}{\|\mathcal{I}_N u\|_\omega}$ which is illustrated in [XSC21b]. Thus, the quality of the numerical interpolation U_N can be measured by $\mathcal{F}(U_N)$.

For a time-dependent problem, the expansion order N may need adjusting dynamically, which can be reflected by the frequency indicator. If the frequency indicator increases, the lower bound for $\frac{\|u-\mathcal{I}_{N-M}u\|_\omega}{\|\mathcal{I}_N u\|_\omega}$ will also increase. On the other hand, as N increases, the error $\|u - \mathcal{I}_N u\|_\omega$ as well as $\mathcal{F}(U_N)$ are expected to decrease. By sufficiently increasing the expansion order N, the frequency indicator as well as the error can be kept small. If the frequency indicator decreases, we can also consider decreasing N to relieve computational costs without compromising accuracy, as was done in [SLC11]. The pseudo-code of the proposed p-adaptive technique is given in Alg. 4.

The p-adaptive spectral method in Alg. 4 for time-dependent problems consists of two ingredients: refinement (increasing N) and coarsening (decreasing N). The method maintains accuracy when there are emerging oscillations by increasing the expansion order N. It also decreases N when the expansion order is larger than needed to avoid unnecessary computation. In Alg. 4, the FREQUENCY_INDICATOR subroutine evaluates the frequency indicator defined in Eq. (6.2.2) for the numerical solution U_N while the EVOLVE subroutine is to obtain the numerical solution $U_N(t + \Delta t)$ at the next timestep from $U_N(t)$.

In Line 11 of Alg. 4, the REFINE subroutine uses U_N to generate a new numerical solution with a larger expansion order U_{N+1} (refine), and in Line 20 of Alg. 4 the COARSEN subroutine uses U_N to generate a new numerical solution with a smaller expansion order U_{N-1} (coarsen). The refinement or coarsening is achieved by reconstructing the function values of U_{N+1} or U_{N-1} at the new set of collocation points $\{x_i\}$:

$$U_{N\pm1}(x_i, t) = U_N(x_i, t), \quad i = 0, ..., N \pm 1, \tag{6.2.3}$$

where U_{N+1} uses $N + 2$ basis functions for refinement and U_{N-1} uses N basis functions for coarsening.

In Alg. 4, ηf_0 is the refinement threshold such that if the current frequency indicator

Algorithm 4 Pseudo-code of the p-adaptive technique which may increase (refine) or decrease (coarsen) the expansion order N.

Initialize N, N_0, $\gamma \geq 1, \eta_0 = \eta > 1$, Δt, T, α, β, $U_N(0)$, $N_{\max}$, $N_{\min}$
$t \leftarrow 0$
$f_0 \leftarrow$ FREQUENCY_INDICATOR$(U_N(t))$
while $t < T$ **do**
    $U_N(t + \Delta t) \leftarrow$ EVOLVE$(U_N(t), \Delta t)$
    $f \leftarrow$ FREQUENCY_INDICATOR$(U_N(t + \Delta t))$
    $l \leftarrow 0$
    if $f > \eta f_0$ **then** # refinement is needed
        while $f > \eta f_0$ **and** $l \leq N_{\max}$ **do**
            $l \leftarrow l + 1$
            $U_{N+1} \leftarrow$ REFINE$(U_N(t + \Delta t))$
            $N \leftarrow N + 1$
            $f \leftarrow$ FREQUENCY_INDICATOR(U_N)
        end while
        $f_0 \leftarrow f$
        $\eta \leftarrow \gamma\eta$ # renew η
    else if $f < f_0/\eta_0$ **then** # coarsening could be considered
        $r \leftarrow$ **False**
        while $f < f_0/\eta_0$ **and** $N > N_{\min}$ **and not** r_1 **do**
            $\tilde{U}_{N-1}(t + \Delta t) \leftarrow$ COARSEN$(U_N(t + \Delta t))$
            $f \leftarrow$ FREQUENCY_INDICATOR$(\tilde{U}_{N-1}(t + \Delta t))$
            if $f < f_0$ **then**
                $f_1 \leftarrow f$
                $r \leftarrow$ **True**
                $U_{N-1}(t + \Delta t) \leftarrow \tilde{U}_{N-1}(t + \Delta t)$
                $N \leftarrow N - 1$
            end if
        end while
        if r **then**
            $f_0 \leftarrow f_1$
        end if
    end if
    $t \leftarrow t + \Delta t$
end while

$f > \eta f_0$, we increase the expansion order N. The **while** loop starting in Line 9 ensures we either refine enough such that the frequency indicator, after increasing N, is smaller than the threshold ηf_0, or the maximal allowable expansion order increment within a single step N_{max} is reached.

After increasing N, f_0 is set to the current frequency indicator and η is multiplied by a factor $\gamma \geq 1$, enabling us to dynamically adjust the refinement threshold for the next refinement in order to prevent increasing N too fast without substantially increasing accuracy. On the other hand, when an extremely large N is needed to match the increasingly oscillatory behavior of the numerical solution, we can set $\gamma \succeq 1$ or even $\gamma = 1$, as we will do in Examples 17 and 20. We have observed numerically, as expected, that the larger η_0, γ are, the more difficult it is to increase the expansion order.

We also consider reducing N when a large expansion order is not really needed and f_0/η_0 is the threshold for decreasing the expansion order. If the condition in Line 17 in Alg. 4 is satisfied and $N > N_{\min}$, the minimal allowable expansion order, and we have not increased N in the current step, then we consider decreasing the expansion order below Line 17 in Alg. 4. As long as the frequency indicator of the new numerical solution with the decreased expansion order $\mathcal{F}(U_{N-1})$ is smaller than f_0, the frequency indicator recorded after previously adjusting the expansion order, reducing the expansion order is accepted; else reducing the expansion order is declined. Therefore, f_0 after coarsening will not surpass f_0 before coarsening. This procedure is described by the **If** condition in Line 22 in Alg. 4. If N is decreased, f_0 will become the latest frequency indicator. In addition, if the current frequency indicator $f \in [\frac{f_0}{\eta_0}, \eta f_0]$, neither the refinement nor the coarsening subroutine is activated.

Alg. 4 can be generalized to higher dimensions in a dimension-by-dimension manner. The expansion order for each dimension can change simultaneously within each timestep by using the tensor product of one-dimensional basis functions, in much the same way moving and scaling algorithms were generalized to higher dimensions [XSC21b]. For example, for a two-dimensional problem, given

Computational domain	Bounded interval	$(0,\infty)$	$(-\infty,\infty)$
Basis functions	Jacobi polynomials	Laguerre polynomials/functions	Hermite polynomials/functions

Table 6.1: Typical choices of basis functions $\{B_i\}_{i=0}^{\infty}$ and computational domain Λ.

$$U_{\vec{N}}(x,y) := \sum_{i=0}^{N_x}\sum_{j=0}^{N_y} u_{i,j}B_i(x)B_j(y) \tag{6.2.4}$$

where $\vec{N} = (N_x, N_y)$, the frequency indicator in the x-direction is defined as

$$\mathcal{F}_x(U_{\vec{N}}) := \left(\frac{\sum_{i=N_x-M_x+1}^{N_x}\sum_{j=0}^{N_y}\gamma_i\gamma_j u_{i,j}^2}{\sum_{i=0}^{N_x}\sum_{j=0}^{N_y}\gamma_i\gamma_j u_{i,j}^2} \right)^{\frac{1}{2}}, \tag{6.2.5}$$

while the frequency indicator in y-direction is similarly defined. At each timestep, we keep N_y fixed and use $\mathcal{F}_x$ to judge whether or not to renew $N_x \to \tilde{N}_x$; simultaneously, we fix N_x and use $\mathcal{F}_y$ to renew $N_y \to \tilde{N}_y$ if adjusting the expansion order in y dimension is needed. Finally N_x, N_y are updated to $\tilde{N}_x, \tilde{N}_y$.

In this work, the relative L^2_ω-error

$$\text{Error} = \frac{\|U_N - u\|_\omega}{\|u\|_\omega}, \tag{6.2.6}$$

is used to measure the quality of the spectral approximation $U_N(x)$ compared to the reference solution $u(x)$. Table 6.1 lists some typical choices of orthogonal basis functions for different domains Λ that we use in this chapter.

We provide two examples of using this p-adaptive technique in Alg. 4 below, where the generalized Jacobi polynomials [STW11] are used. Theorem 3.41 in [STW11] gives an estimation for the interpolation error of a function u in the Jacobi-weighted Sobolev space

for $\alpha, \beta > -1$ as follows

$$\|\partial_x^l (I_{N,\alpha,\beta} u - u)\|_{\omega_{\alpha+l,\beta+l}} \leq c\sqrt{\frac{(N-m+1)!}{N!}} N^{l-(m+1)/2} \|\partial_x^m u\|_{\omega_{\alpha+m,\beta+m}}, \ 0 \leq l \leq m \leq N+1, \tag{6.2.7}$$

where c is a positive constant independent of m, N and u. When $m > 0$ and $l = 0$, the left-hand side becomes the interpolation error $\|(I_{N,\alpha,\beta} u - u)\|_{\omega_{\alpha,\beta}}$ which may decrease with N. Therefore, by increasing the expansion order for the Jacobi polynomials it is generally true that the interpolation will be more accurate. Theorem 7.16 and Theorem 7.17 in [STW11] give similar error estimates for Laguerre and Hermite interpolations, which reveals that under some smoothness assumptions, the interpolation error decreases when the expansion order N increases.

Since unbounded domain problems may involve diffusive and advective behavior, we discuss and develop adaptive spectral methods in unbounded domains in the next section.

Example 12. We numerically solve the PDE

$$\partial_t u = \left(\frac{x+2}{t+1}\right) \partial_x u, \quad x \in [-1, 1], \tag{6.2.8}$$

with a Dirichlet boundary condition specified at $x = 1$ given as $u(1, t) = \cos 3(t + 1)$. This PDE admits an analytical solution

$$u(x, t) = \cos((t + 1)(x + 2)). \tag{6.2.9}$$

We solve it numerically by using Chebyshev polynomials with Chebyshev-Gauss-Lobatto quadrature nodes and weights. The Chebyshev polynomials are orthogonal under the weight function $\omega(x) = (1 - x^2)^{-\frac{1}{2}}$, *i.e.*, they correspond to Jacobi polynomials with $\alpha = \beta = -\frac{1}{2}$. Since $u(x, t)$ becomes increasingly oscillatory over time, an increasing expansion order is required to capture these oscillations. We start with $N = 10$ at $t = 0$, the parameters $\eta = 1.5, \gamma = 1.1, N_{\max} = 3, N_{\min} = 0$, and a timestep $\Delta t = 0.001$. We use a third-order

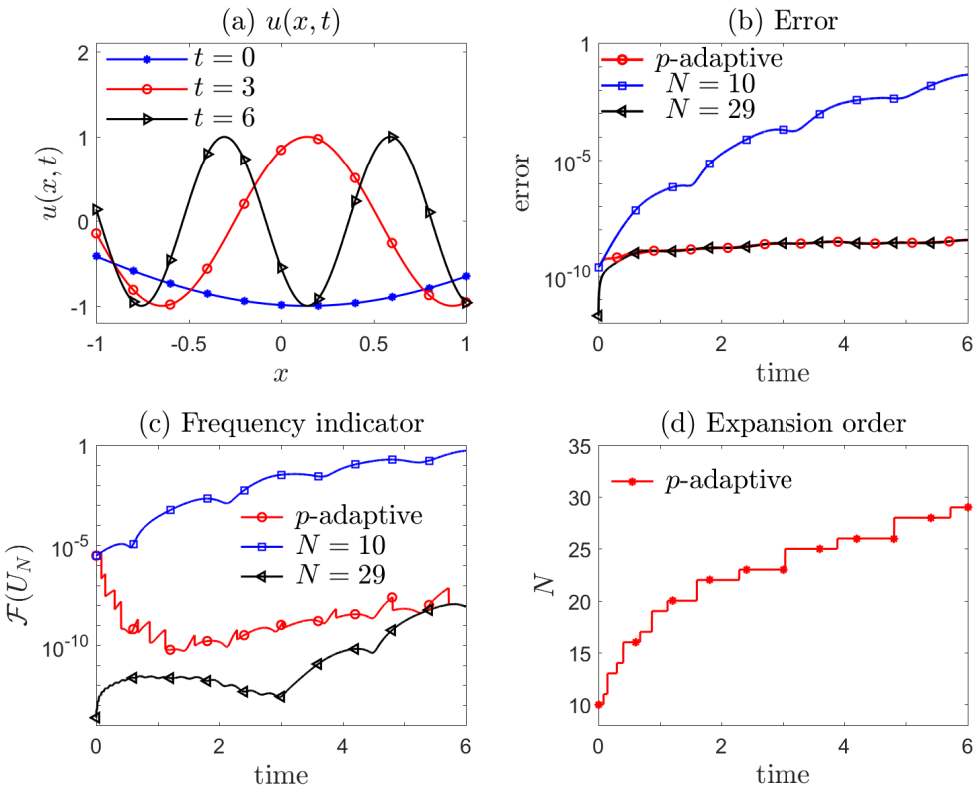

Figure 6.1: Numerically solving Eq. (6.2.8) with Chebyshev polynomials using Alg. 4. For solutions that become increasingly oscillatory, the p-adaptive technique can increase the expansion order effectively to capture the oscillations and maintain a small error by keeping the frequency indicator low. A fixed $N = 10$ fails to maintain the frequency indicator and results in a larger error, whereas using a fixed $N = 29$, the largest expansion order appearing during the p-adaptive procedure, will not result in higher accuracy at $t = 6$ than the p-adaptive technique but requires a higher computational cost. The p-adaptive technique dynamically selects an expansion order N that saves computational costs while maintaining accuracy.

explicit Runge-Kutta scheme to advance time.

The reference solution $u(x,t)$ is plotted in Fig. 6.1(a). The increasing oscillations lead to a fast rise in the frequency indicator as the contribution from high-frequency modes increases. Keeping the same number of basis functions over time will fail as it will be eventually incapable of capturing the shorter wavelength oscillations.

However, a much more accurate approximation can be obtained (see Fig. 6.1(b)) with our p-adaptive method which maintains the frequency indicator (see Fig. 6.1(c)) by increasing the number of basis functions (shown in Fig. 6.1(d)). Furthermore, the coarsening subroutine

for decreasing the expansion order described in the **while** loop in Line 17 will not be triggered (shown in Fig. 6.1(d)). The largest expansion order $N = 29$ appearing during the p-adaptive procedure occurs at $t = 6$. For comparison, we also plot the error and the frequency indicator for fixed $N = 29$ (black curves in Figs. 6.1(b, c)). Using a fixed $N = 29$ does not lead to a more accurate result at $t = 6$ than the p-adaptive technique starting from $N = 10$ because a larger expansion order is not needed when t is small. On our laptop, the solution in $t \in [0, 6]$, using fixed $N = 29$, took 339.0469 seconds to evaluate while p-adaptive method required only 294.9531 seconds. Therefore, the p-adaptive method maintained accuracy while requiring a lower computational cost. In this and subsequent examples, we record the runtime as a measure of computational costs. All computations were performed using MATLAB with double precision running on a laptop with a 4-core i7-8550U Intel CPU running at 1.80 GHz.

When directly approximating the reference solution in Eq. (6.2.9), we can achieve 10^{-8} accuracy with only 20 basis functions. However, when numerically solving Eq. (6.2.8), the error will accumulate due to the increasing oscillatory behavior which will require even more basis functions to achieve the same accuracy as the direct approximation to Eq. (6.2.9). Thus, the oscillatory behavior of the solution poses additional difficulties and requires even more refinement when numerically solving a PDE.

Next, we present an example in which we apply the proposed p-adaptive technique to approximate a function that is oscillatory over time and contains a singularity.

Example 13. We approximate the function

$$u(x,t) = \begin{cases} (1-x)^{-0.01} \sin\left((2t+1)(x+1)\right), t \in [0,1), \\ (1-x)^{-0.01} \sin\left((5-2t)(x+1)\right), t \in [1,2), \\ (1-x)^{-0.01} \sin\left((2(t-2)+1)(x+1)\right), t \geq 2, \end{cases} \tag{6.2.10}$$

where $x \in [-1, 1]$ and $t \in [0, 6]$, by Chebyshev polynomials with Chebyshev-Gauss quadrature nodes and weights. The function carries a singularity at $x = 1$ as $\lim_{x \to 1} |u(x,t)| = +\infty$ except when t satisfies $u(1,t) = 0$. Away from the singularity, the function becomes more

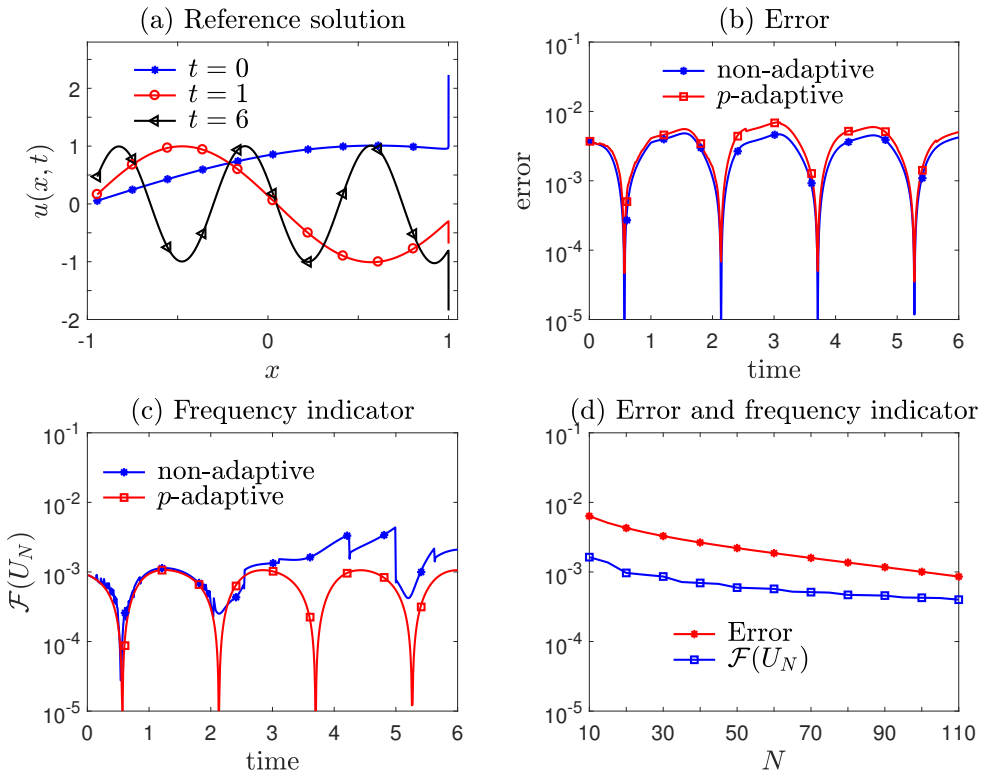

Figure 6.2: The p-adaptive technique applied to evaluating the singular function in Eq. (6.2.10). The function $u(x,t)$ becomes more oscillatory when $t \in [0,1] \cup [2,6]$ and less oscillatory when $t \in [1,2]$ and has a singularity at $x = 1$. The error of the approximation decreases very slowly with increasing expansion orders due to this singularity. Applying the p-adaptive technique straightforwardly in the whole domain $[-1,1]$ cannot substantially increase accuracy due to failure to approximate the singularity.

oscillatory in $t \in [0,1] \cup [2,6]$ and less oscillatory when $t \in [1,2]$ as shown in Fig. 6.2(a). The expansion order should be increased or decreased appropriately when the function becomes more or less oscillatory to maintain accuracy or to relieve the computational burden.

As clearly shown in [GW01], the error when using spectral methods to approximate $u(x,t)$ in Eq. (6.2.10) with the Jacobi basis functions is about 10^{-4}, as shown in Figs. 6.2(b, c), where we approximate Eq. (6.2.10) with and without the p-adaptive technique by setting the initial expansion order $N = 25$, $\eta = 1.1$ and $\gamma = 1.1$. It can be seen that directly approximating u in Eq. (6.2.10) leads to a large error even with the p-adaptive technique (red curve in Fig. 6.2(b)). Large errors are accompanied by large frequency indicators as shown in Fig. 6.2(c). This error cannot be significantly reduced by simply increasing the expansion order (up to $N = 110$ shown in Fig. 6.2(d)) because of the singularity at $x = 1$. Both the error and frequency indicator decay very slowly with increasing N due to the failure in approximating the singularity.

In order to accurately approximate $u(x,t)$, we divide the interval $x \in [-1,1]$ into $I_\ell = [-1, 0.99)$ and $I_r = [0.99, 1]$ to isolate a small neighborhood around the singularity and approximate the function separately in the two subdomains. Fig. 6.3(a) plots the distribution of errors associated with approximating $u(x,6)$ in the whole domain $[-1,1]$ by using a fixed expansion order $N = 38$ and by using a fixed expansion order $N = 26$ in the subdomain I_ℓ and $N = 11$ in I_r. By separating the domain, the resulting errors are smaller in both subdomains. Next, we apply the p-adaptive technique in both subdomains. In I_ℓ, the function is nonsingular and its varying oscillatory behavior resulting from the factor $\sin((2t+1)(x+1)), t \in [0,1), \sin\left((5-2t)(x+1)\right), t \in [1,2)$ or $\sin((2(t-2)+1)(x+1)), t \geq 2$ in Eq. (6.2.10) requires proper adjustment of the expansion order. In I_r, the function is dominated by the singular term $(1-x)^{-0.01}$. In this two-subdomain approximation, we set $\eta = 1.05$, $\gamma = 1.1$, an initial expansion order $N = 10$, $N_{\max} = 15$, $N_{\min} = 0$ for the numerical solution U_ℓ in subdomain I_ℓ, and $\eta = 1.2$, $\gamma = 1.1$, an initial expansion order $N = 5$, $N_{\max} = 9$, $N_{\min} = 0$ for the numerical solution U_r in subdomain I_r.

Fig. 6.3 displays the numerical results of first dividing the domain into two subdomains

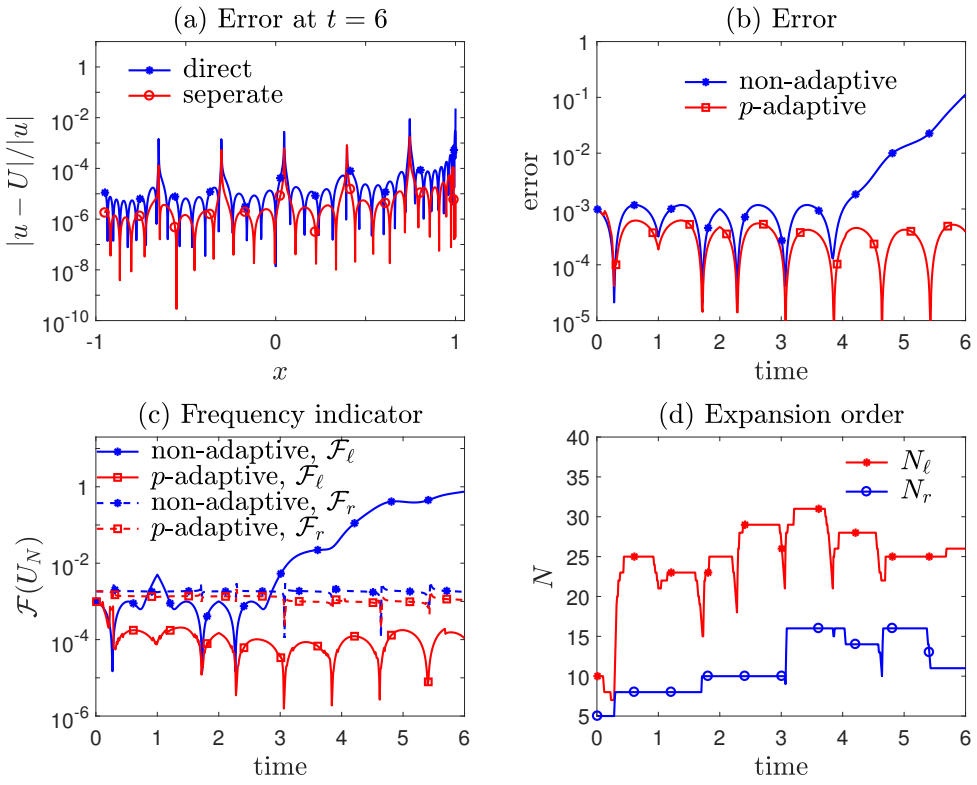

Figure 6.3: Dividing the function in Eq. (6.2.10) into the domains $[-1, 1] = [-1, 0.99) \cup [0.99, 1]$ and using the p-adaptive technique to separately approximate $u(x, t)$ in each subdomain. Dividing the domain and separating the neighborhood of the singularity leads to improved accuracy compared to approximating $u(x, t)$ in the whole function $[-1, 1]$. In the subdomain I_ℓ, oscillatory behavior dominates, and properly adjusting the expansion order N_ℓ by the p-adaptive technique is necessary (red curve in (d)). In the subdomain I_r, adjusting the expansion order N_r is not essential (blue curve in (d)).

$I_\ell \cup I_r$ and then approximating the function separately in each of them. As shown in Fig. 6.3(b), the errors for the p-adaptive methods in the subdomains I_ℓ and I_r are smaller presumably because the frequency indicators in both subdomains can be better controlled, as shown in Fig. 6.3(c). To approximate the varying oscillatory behavior in I_ℓ we need to adjust N_ℓ (the expansion order for U_ℓ) while to approximate $u(x, t)$ in the neighborhood of the singularity in I_r does not rely on adjusting the expansion order (Fig. 6.3(d)). Thus, by using this domain separation strategy to isolate the neighborhood of the singularity, the p-adaptive technique can be used to capture varying oscillatory behavior, leading to higher accuracy if the singular behavior is appropriately captured.

Finally, we present an example of a two-dimensional problem in $[-1,1]^2$.

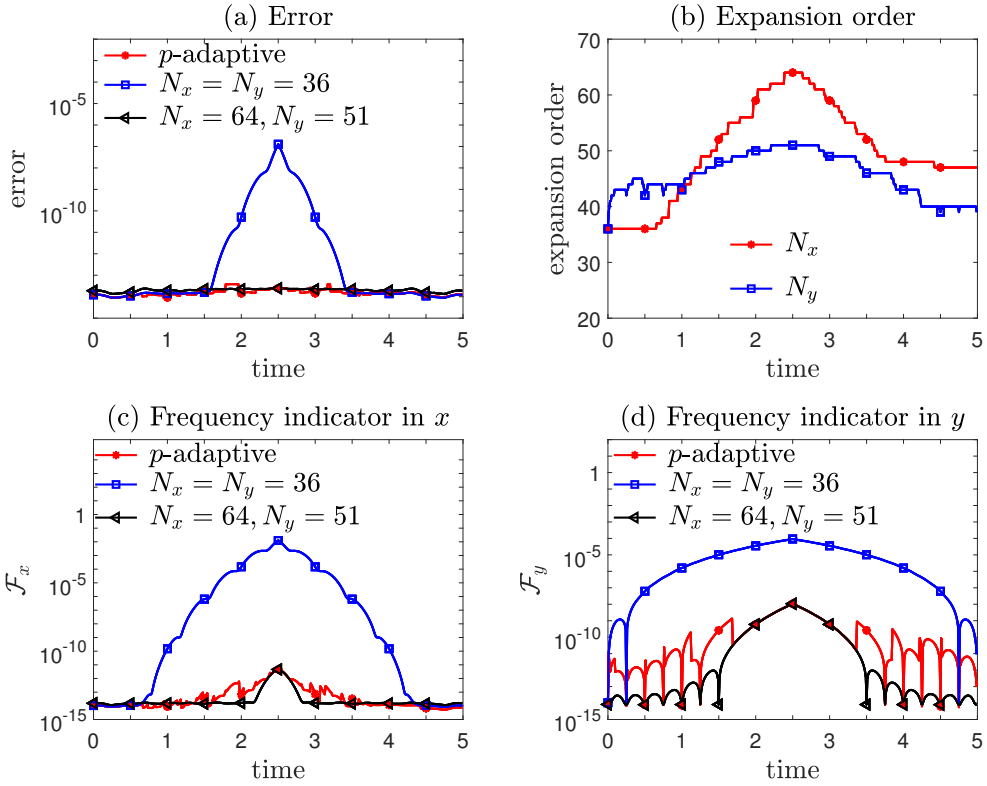

Figure 6.4: Using the p-adaptive technique to approximate the two-dimensional function in Eq. (6.2.11) with Legendre polynomials. Refinement is applied in each direction simultaneously to capture increasing oscillations in both directions. Coarsening is applied when large expansion orders are not needed. Anisotropic oscillatory behavior requires adjusting the expansion order in each direction differently. The frequency indicators in both dimensions are kept low, leading to a small error.

Example 14. We approximate the function

$$u(x,y,t) = \cos\left(xy(5-2|t-\tfrac{5}{2}|)\right) + y^{10-4|t-5/2|}\sin\left(4x(5-2|t-\tfrac{5}{2}|)\right),\ (x,y)\in[-1,1]^2 \tag{6.2.11}$$

by Legendre polynomials (corresponding to Jacobi polynomials with $\alpha = \beta = 0$) with Legendre-Gauss-Lobatto quadrature nodes and weights in both dimensions. Within $t \in [0, \frac{5}{2}]$, the function becomes more oscillatory over time in both dimensions, requiring increasing expansion orders. For $t \in [\frac{5}{2}, \frac{7}{2}]$, the error for approximation with fixed expansion orders in both dimensions decreases because the function becomes less oscillatory, and therefore a

reduction in expansion orders in both directions can be used to reduce computational effort without compromising accuracy. Since the function is not symmetric in x and y, the adjustment of expansion order is anisotropic. We show that Alg. 4 can appropriately increase N_x, N_y when $t < \frac{5}{2}$ and reduce N_x, N_y when $t \geq \frac{5}{2}$. We take $N_x = N_y = 36$ at $t = 0$ with a timestep $\Delta t = 0.01$, and $\gamma_x = \gamma_y = 1.1, \eta_x = \eta_y = 1.1, N_{\max,x} = N_{\max,y} = 3, N_{\min,x} = N_{\min,y} = 0$. The maximum expansion orders during $t \in [0, 5]$ are $N_x = 64$ and $N_y = 51$ for the p-adaptive method, which we also use as fixed expansion orders for comparison.

It is evident from Fig. 6.4(a) that fixing the number of basis functions to $N_x = N_y = 37$ in each dimension leads to an approximation that deteriorates while the proposed p-adaptive spectral method can keep the error small. The p-adaptive technique can maintain the same accuracy as using $N_x = 64$ and $N_y = 51$, as shown in Fig. 6.4(a), but requires only 2.3681×10^3 seconds of run-time compared to 2.5365×10^3 seconds when using fixed $N_x = 64$, $N_y = 51$. This example confirms that the p-adaptive technique can reduce computational burden by appropriate adjustment of the expansion order. Furthermore, when $t \in [\frac{5}{2}, 5]$ we see that with fixed expansion orders $N_x = 64$ and $N_y = 51$ the approximation error decreases, indicating that coarsening can be tolerated to relieve computational burden while maintaining accuracy. Alg. 4 first tracks increasing oscillations by increasing expansion orders in both x and y directions. When $t \geq \frac{5}{2}$, Alg. 4 senses a decrease in the frequency indicator and decreases both N_x and N_y adaptively (Fig. 6.4(b)) without compromising accuracy (blue and black curves in Fig. 6.4(a)). Overall, Alg. 4 preserves accuracy at all times while avoiding excessive values of N_x and N_y when they are not needed.

Since $\sin(4x(5 - 2|t - \frac{5}{2}|))$ is the most oscillatory term in $u(x, y, t)$, the function is more oscillatory in x than in y for $t \in [0, \frac{5}{2}]$. Therefore, the expansion orders should be adjusted anisotropically and we expect N_x needs to be increased more than N_y in order to keep $\mathcal{F}_x$ small, which is indeed observed (Fig. 6.4(b)). That is, the proposed p-adaptive technique can successfully sense the function's heterogeneity and adjust the expansion orders differently in each dimension. Over time, both $\mathcal{F}_x$ and $\mathcal{F}_y$ in the p-adaptive approximation are maintained as well as when $N_x = 64$ and $N_y = 51$ are fixed (Figs. 6.4(c, d)), leading to satisfactory error

control.

6.3 Adaptive spectral methods in unbounded domains

Unbounded domain problems are often more difficult to solve numerically than bounded domain problems. Diffusion and advection in unbounded domains necessitates knowledge of the solution's behavior at infinity. To distinguish and handle diffusive and advective behavior in unbounded domains, techniques for scaling and moving basis functions are proposed in [XSC21b]. When combining scaling, moving, refinement, and coarsening, we can devise a comprehensive adaptive spectral approach for unbounded domains. A flow chart of our overall approach is given in Fig. 6.5. The scaling, refinement, and coarsening techniques all rely on a common frequency indicator.

As is stated in [XSC21b], advection may cause a false increase in the frequency indicator. Thus, we must first compensate for advection by the moving technique before we consider either scaling or adjusting the expansion order. Next, as the cost of changing the scaling factor is lower than increasing the expansion order, we implement scaling before adjusting the expansion order. Only if scaling cannot maintain the frequency indicator below the refinement threshold do we consider increasing the expansion order. Coarsening is also considered after scaling if the frequency indicator decreases below the threshold.

As was done in [XSC21b], we also decrease the scaling factor β by multiplying it by a common ratio $q < 1$ if the current frequency indicator is larger than the scaling threshold $f > \nu f_1$. The scaling we perform here contains an additional step: when the current frequency indicator decreases and is below f_1, we consider increasing the scaling factor β by dividing it by the common ratio q as long as the frequency indicator decreases after increasing β. When $f \in [f_1, \nu f_1]$, β is neither increased nor decreased. Thus, at each step, the scaling factor β may be either increased or decreased as long as the frequency indicator decreases after adjusting the scaling factor. A decrease in the scaling factor indicates that the allocation points are more efficiently distributed. These changes avoid the unnecessary computational

burden that may arise if N is excessively increased. We briefly describe our modified scaling subroutine for one timestep in Alg. 5.

Algorithm 5 Pseudo-code of the frequency-dependent scaling technique which may increase or decrease the scaling factor β.

$f \leftarrow \text{FREQUENCY\_INDICATOR}(U_N^{(\alpha,\beta)}(t+\Delta t))$
if $f > \nu f_1$ **then** # try decreasing β
  $\tilde{\beta} \leftarrow q\beta$
  $U_N^{(\alpha,\tilde{\beta})} \leftarrow \text{SCALE}(U_N^{(\alpha,\beta)}(t+\Delta t), \tilde{\beta})$
  $\tilde{f} \leftarrow \text{FREQUENCY\_INDICATOR}(U_N^{(\alpha,\tilde{\beta})})$
  while $\tilde{f} \leq f$ **and** $\tilde{\beta} \geq \underline{\beta}$ **do**
    $\beta \leftarrow \tilde{\beta}$
    $U_N^{(\alpha,\beta)}(t+\Delta t) \leftarrow U_N^{(\alpha,\tilde{\beta})}$
    $f_1 \leftarrow \tilde{f}$
    $f \leftarrow \tilde{f}$
    $\tilde{\beta} \leftarrow q\beta$
    $U_N^{(\alpha,\tilde{\beta})} \leftarrow \text{SCALE}(U_N^{(\alpha,\beta)}(t+\Delta t), \tilde{\beta})$
    $\tilde{f} \leftarrow \text{FREQUENCY\_INDICATOR}(U_N^{(\alpha,\tilde{\beta})})$
  end while
else if $f < f_1$ **then** # try increasing β
  $\tilde{\beta} \leftarrow \beta/q$
  $U_N^{(\alpha,\tilde{\beta})} \leftarrow \text{SCALE}(U_N^{(\alpha,\beta)}(t+\Delta t), \tilde{\beta})$
  $\tilde{f} \leftarrow \text{FREQUENCY\_INDICATOR}(U_N^{(\alpha,\tilde{\beta})})$
  while $\tilde{f} \leq f$ **and** $\tilde{\beta} \leq \overline{\beta}$ **do**
    $\beta \leftarrow \tilde{\beta}$
    $U_N^{(\alpha,\beta)}(t+\Delta t) \leftarrow U_N^{(\alpha,\tilde{\beta})}$
    $f_1 \leftarrow \tilde{f}$
    $f \leftarrow \tilde{f}$
    $\tilde{\beta} \leftarrow \beta/q$
    $U_N^{(\alpha,\tilde{\beta})} \leftarrow \text{SCALE}(U_N^{(\alpha,\beta)}(t+\Delta t), \tilde{\beta})$
    $\tilde{f} \leftarrow \text{FREQUENCY\_INDICATOR}(U_N^{(\alpha,\tilde{\beta})})$
  end while
end if

For simplicity, we assume that the function is moving rightward so we need to move the basis functions rightward. Therefore, (x_R, ∞) is the "exterior domain" of the spectral approximation on which we wish to control the error as illustrated in [XSC21b]. For Laguerre polynomials/functions the parameter x_L in the algorithm in Fig. 6.5 denotes the starting

point for the approximation, while for Hermite polynomials/functions x_L represents the translation of Hermite polynomials/functions, *i.e.*, we use $\{\mathcal{H}_i(x-x_L)\}$ or $\{\hat{\mathcal{H}}_i(x-x_L)\}$. Let $U^{(\beta)}_{N,x_L}$ be the spectral approximation with the scaling factor β. The exterior-error indicator for the semi-unbounded domain is defined in [XSC21b] and we can generalize it to $\mathbb{R}$ when using Hermite polynomials/functions

$$\mathcal{E}(U^{(\beta)}_{N,x_L}, x_R) = \frac{\|\partial_x U^{(\beta)}_{N,x_L} \cdot \mathbb{I}_{(x_R,+\infty)}\|_{\omega_\beta}}{\|\partial_x U^{(\beta)}_{N,x_L} \cdot \mathbb{I}_{(-\infty,+\infty)}\|_{\omega_\beta}}, \tag{6.3.1}$$

where ω_β is the weight function and x_R is taken to be $x^{(\beta)}_{[\frac{2N+2}{3}]}$ for Hermite functions/polynomials and $x^{(\alpha,\beta)}_{[\frac{N+2}{3}]}$ for Laguerre functions/polynomials [XSC21b] in view of the often-used $\frac{2}{3}$-rule. The difference between the choices of x_R for Hermite and Laguerre basis functions arises because the allocation points for Hermite functions are symmetrically distributed around their center while those for Laguerre functions are one-sided, to the right of the starting point x_L in the axis. If the solution $u(x)$ moves rightward in time, the spectral approximation at large distances may deteriorate and the exterior-error indicator $\mathcal{E}(U^{(\beta)}_{N,x_L}, x_R)$ will increase. This means that the moving mechanism will be triggered and the starting point of the spectral approximation will need to be updated by $x_L \to x_L + d_0$ with the displacement determined by $d_0 = \min\{n\delta, d_{max}\}$. Here n is the smallest integer satisfying $\mathcal{E}(U^{(\beta)}_{N,x_L}, x_R + n\delta) < \mu e_0$, δ is the minimum displacement, d_{max} is the maximum displacement, and μ represents the threshold of the increase in the exterior-error indicator (the current value of which is given by e_0) that we can tolerate.

For the scaling subroutine, we need the following parameters: the common ratio $q < 1$ that we use to geometrically shrink/increase the scaling factor, the parameter describing the threshold for considering shrinking the scaling factor ν; a predetermined lower bound for the scaling factor $\underline{\beta}$ and an upper bound $\overline{\beta}$. For the moving subroutine, the required parameters include the minimal displacement for the moving technique δ, the maximal displacement within a single timestep d_{max}, and the parameter of the threshold for activating the moving technique μ. At the beginning $t = 0$, we need to ensure the frequency indicator and the

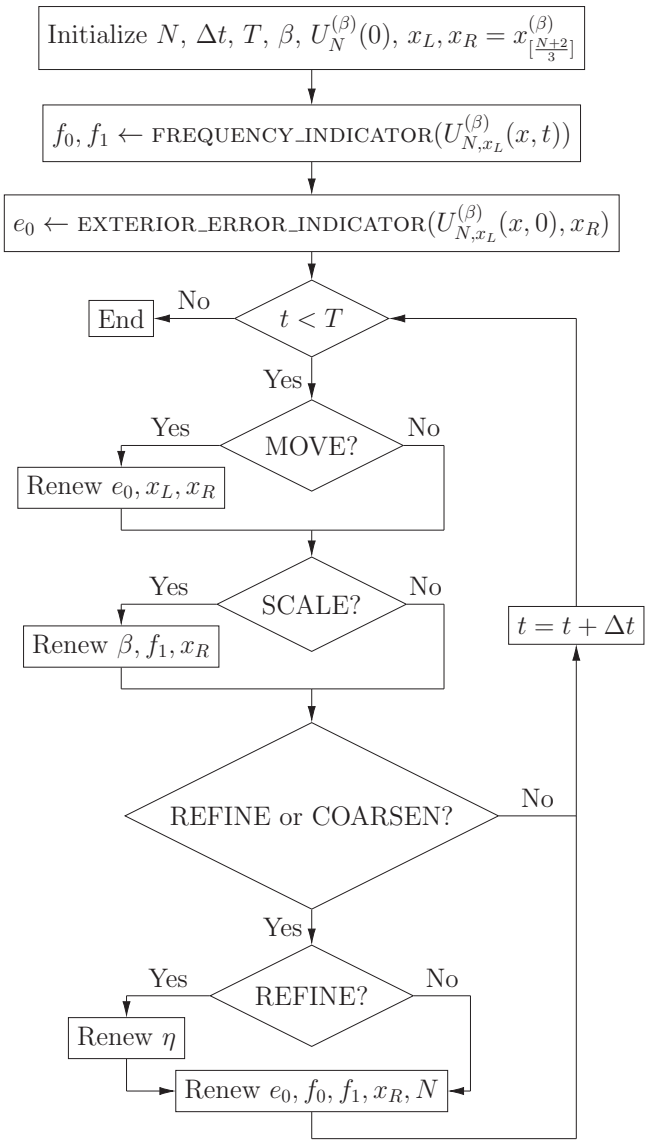

Figure 6.5: Flow chart of an adaptive spectral method in unbounded domains that includes moving, scaling, refinement, and coarsening techniques.

γ \ η	1.2	1.5	2	4
1.05	1.434, 67 1.305e-05	1.434, 64 2.346-05	1.362, 58 7.687e-05	2.053, 55 6.030e-05
1.1	1.510, 62 2.500e-05	1.434, 69 5.396e-05	1.673, 66 5.513e-05	2.053, 55 6.030e-05
1.2	1.853, 56 4.451e-05	1.673, 56 5.512e-05	1.673, 53 8.927e-05	1.853, 53 8.706e-05
1.5	1.589, 52 1.369e-04	1.673, 53 8.927e-05	1.761, 52 1.099e-04	1.951, 53 8.702e-05

Table 6.2: Error, β, and N at $t = 5$ for different η and γ with both p-adaptive and scaling techniques.

exterior-error indicator are relatively small by choosing a suitable initial scaling factor β and an appropriate initial translation of basis functions x_L, respectively. The SCALE and MOVE in Fig. 6.5 are the scaling and moving subroutines and EXTERIOR_ERROR_INDICATOR calculates the exterior-error indicator for the moving subroutine. Detailed discussions on the scaling and moving techniques are given in [XSC21b]. Note that after the expansion order N has changed, we need to renew the threshold for scaling, the threshold for subsequent adjustment of the expansion order, as well as the threshold for moving, as indicated in Fig. 6.5. After first applying the moving technique, adjusting the expansion order and scaling both depend on the frequency indicator and aim to keep the frequency indicator low to control the error. The relationship and interdependence between them is key to understanding and justifying the first-scaling-then-adjusting expansion-order procedure in Fig. 6.5. Thus, we need to investigate how the proposed scaling technique will affect our p-adaptive technique and how these two techniques interact with each other. We use two examples containing both diffusive and oscillatory behavior to investigate how the two techniques will be activated and influence each other. In Example 15, both refinement and reducing β are needed for matching increasing oscillatory and diffusive behavior of the solution; in Example 16, a less oscillatory and diffusive solution over time implies that coarsening and increasing β may be considered.

γ \ η	1.2	1.5	2	4
1.05	79 1.316e-05	71 3.720e-05	65 9.724e-05	59 2.364e-04
1.1	70 4.372e-05	67 6.544e-05	64 9.823e-05	58 2.607e-04
1.2	65 9.724e-05	62 1.534e-04	61 1.597e-04	58 2.607e-04
1.5	59 2.364e-03	59 2.364e-04	58 2.607e-04	56 3.508e-04

Table 6.3: Error and N at $t = 5$ for different η and γ with the p-adaptive technique but without the scaling technique, $\beta = 4$.

Example 15. We approximate the function

$$u(x,t) = \exp\left[-\frac{x}{(bt+a)}\right]\cos x, \quad t \in \mathbb{R}^+ \tag{6.3.2}$$

with the generalized Laguerre function basis $\{\hat{\mathcal{L}}_i^{(\alpha,\beta)}(x)\}_{i=0}^{\infty}$ discussed in [XSC21b] with the parameter $\alpha = 0$. The magnitude of oscillations for this function, $\exp(-\frac{x}{(bt+a)})$, increases over time, requiring proper scaling. Under a variable transformation $y = \frac{x}{bt+a}$, $u(x,t)$ can be rewritten as $u(y,t) = \cos\left((bt+a)y\right)\exp(-y)$, indicating that the solution is increasingly oscillatory in y as time increases. Thus, if we reduce the scaling factor β to match the diffusive behavior of the solution, proper refinement is also required. In other words, diffusive and oscillatory behavior is coupled in this example. We carry out numerical experiments using the algorithm described in Fig. 6.5 with different (η, γ) to investigate how scaling and refinement influence each other. We deactivate the moving technique by setting $d_{max} = 0$ since the solution exhibits no intrinsic advection. Even if we had allowed moving, it was hardly activated. We set $\Delta t = 10^{-3}, N = 50$ at $t = 0$ and $a = 2, b = 0.7$. $q = v^{-1} = 0.95, \underline{\beta} = 0.3, \overline{\beta} = 10, N_{\min} = 0, N_{\max} = 3$ and choose the initial scaling factor $\beta = 4$.

In Tables 6.2 and 6.3 the error in $\mathbb{R}^+$ is recorded in the lower-left part of each entry while the scaling factor β and expansion order N at $t = 5$ is recorded in the upper-right. By comparing entries in each column/row for smaller η, γ, both tables show the expansion order N is likely to be increased more when the threshold for refinement ηf_0 is lower.

We see from Table 6.2 that with more refinement β tends to be smaller. This interaction between p-adaptivity and scaling arises because more refinement leads to a larger expansion order N and a smaller scaling threshold νf_1. Since scaling will only be performed if the frequency indicator after scaling decreases, proper refinement is not likely to lead to over-scaling. Moreover, by comparing N at $t = 5$ between Tables 6.2 and 6.3, we see that N tends to be smaller with the scaling technique for the same γ, η. This implies that without scaling, the refinement procedure is more often activated, leading to a larger N to compensate for the incapability of scaling alone to maintain a low-frequency indicator. This results in a larger computational burden without an improvement in accuracy. This behavior has been expected from the design of Alg. 6.5 since we put scaling before refinement so that redistribution of collocation points is tried first to avoid unnecessary refinement when the increase in frequency indicator results from diffusion instead of oscillation.

Example 16. We approximate the function

$$u(x,t) = \exp\left[-(bt+a)x\right]\cos x,\ \ x,t \in \mathbb{R}^+ \tag{6.3.3}$$

with the generalized Laguerre function basis with the parameter $\alpha = 0$. The magnitude of oscillations for this function, $\exp(-(bt+a)x)$, decreases over time and increasing the scaling factor β to more densely redistribute the allocation points is needed. Furthermore, under the variable transformation $y = (bt+a)x$, $u(x,t)$ can be rewritten as $u(y,t) = \cos(\frac{y}{bt+a})\exp(-y)$. Since the oscillations decrease with y, one can reduce the expansion order. We consider coarsening with or without scaling to investigate whether increasing β can facilitate coarsening (and save computational effort) or result in higher accuracy. We carry out numerical experiments using the algorithm described in Fig. 6.5 and different (η, γ) and also deactivate the moving technique by setting $d_{max} = 0$ since the solution exhibits no intrinsic advection. We set $\Delta t = 10^{-3}, N = 50$ at the beginning and set the parameters $a = \frac{1}{2}, b = 0.5$, $q = v^{-1} = 0.95, \underline{\beta} = 0.3, \overline{\beta} = 10, N_{\min} = 0, N_{\max} = 3$ and initial scaling factor $\beta = 4$. We use a different threshold η_0 for coarsening and we have checked numerically that the parameter

η	1.2	1.5	2	4
Scaled	5.728, 11 6.514e-10	7.032, 13 8.885e-12	6.347, 13 1.260e-11	6.681, 17 1.255e-14
Unscaled	4, 20 2.707e-12	4, 28 8.127e-11	4, 31 9.417e-15	4, 30 9.672e-15

Table 6.4: Error, β and N at $t = 5$ for different η_0 and γ with/without scaling for the p-adaptive technique.

γ in the refinement subroutine will not affect the coarsening subroutine in this example.

In Table 6.4 the error in $\mathbb{R}^+$ is recorded in the lower-left part of each entry while the scaling factor β and expansion order N at $t = 5$ is recorded in the upper-right. By comparing entries in each row we see that a smaller η_0 will lead to easier coarsening and a smaller N at $t = 5$. Since the approximation with larger N is always better, whether we can achieve the same level of accuracy with a smaller expansion order N, if proper scaling is implemented, is of interest. The initial approximation error is 1.960×10^{-9} and the approximation will not worsen after coarsening regardless of η_0 because in the p-adaptive subroutine coarsening is allowed only when the post-coarsening frequency indicator remains below the previous threshold f_0. Moreover, by comparing the two rows in Table 6.4 we see that if the solution concentrates and becomes less diffusive, increasing β and more efficiently redistributing the allocation points allows the scaling technique to achieve high accuracy with fewer expansion orders than without the scaling technique.

The time-dependent errors and expansion orders are plotted in Fig. 6.6 where the p-adaptive method is compared with the non-p-adaptive method when scaling is applied. From Figs. 6.6(a, c) we can observe that both scaled and unscaled methods maintain the error below the initial approximation error. Yet, upon comparing Fig. 6.6(b) to Fig. 6.6(d) it is readily seen that the scaled method leads to appropriate coarsening while succeeding in maintaining low error, but the unscaled method will increase N when increasing the expansion order is not actually needed, resulting in an additional unnecessary computational burden. In Fig. 6.6(e) the scaled and p-adaptive spectral method with $\eta_0 = 4$ is compared with the scaling-only spectral method. We see that the errors for both methods are almost the same but the p-adaptive method can reduce unnecessary computation by decreasing N

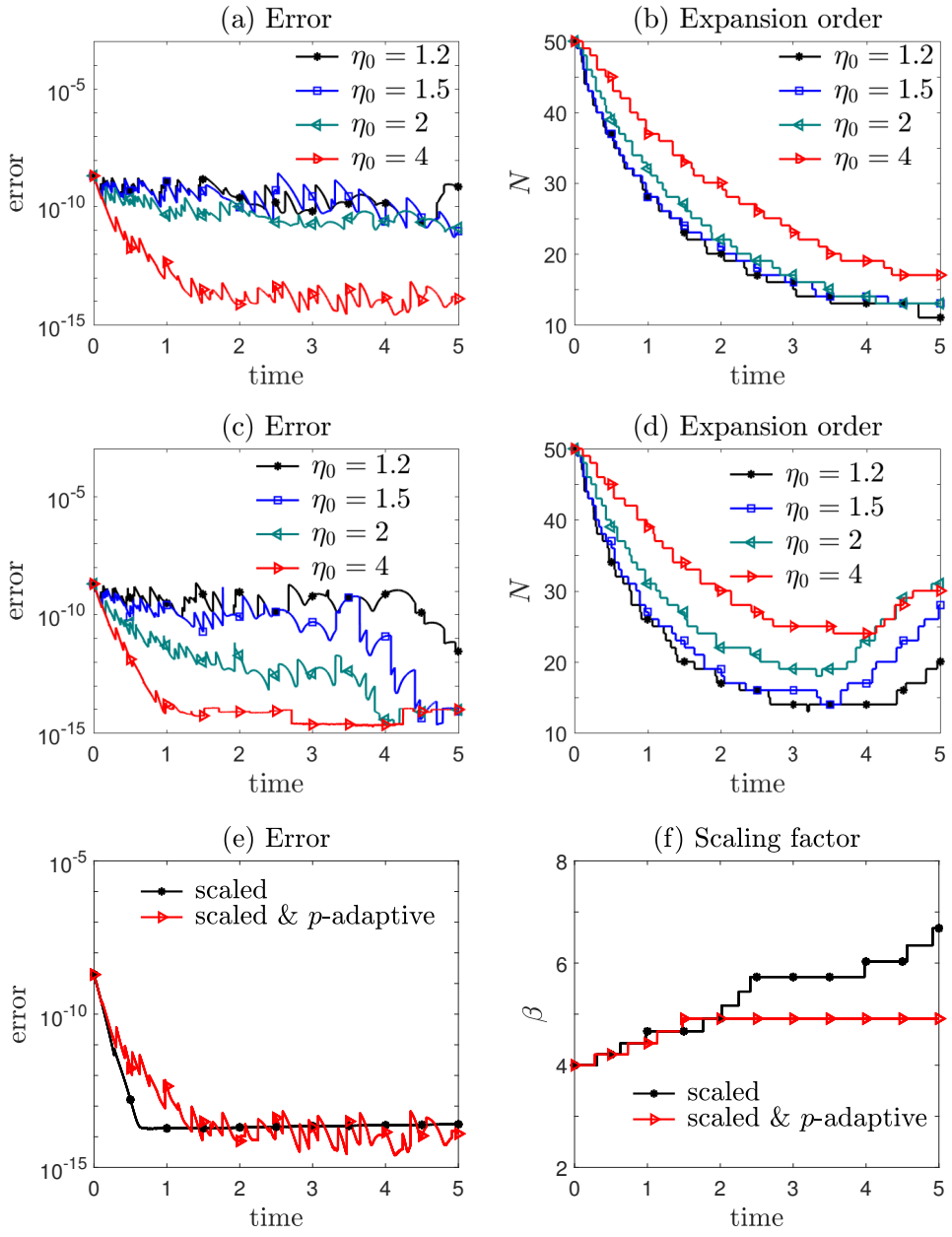

Figure 6.6: Approximation to Eq. (6.3.3) with scaling and p-adaptive spectral methods. Increasing β by scaling can save computational burden while maintaining accuracy by more efficiently redistributing allocation points. The approximation error is controlled below the initial approximation error for both scaled and unscaled p-adaptive methods, but the expansion order of the scaled method is smaller. On the other hand, adjusting the scaling factor without decreasing N will not achieve higher accuracy even with a much larger expansion order.

adaptively while still maintaining a low error, and the approximation error for the p-adaptive method fluctuates due to a decreasing N.

Fig. 6.6(f) shows that the scaling factor β is increased more in the p-adaptive method, implying that the reason why the p-adaptive method can achieve the same accuracy as the non-p-adaptive method with a smaller expansion order is that it can redistribute the allocation points more efficiently.

Finally, we conclude that all three methods: scaling, p-adaptive+scaling, and p-adaptive methods can maintain the error well below the initial approximation error, but the combined p-adaptive+scaling method can achieve this with the smallest expansion order and is therefore the most efficient method among them.

6.4 Applications in solving the Schrödinger equation

In this section, we apply our adaptive spectral methods described in Fig. 6.5 to solve the Schrödinger equation in unbounded domains

$$i\partial_t \psi(x,t) = -\partial_x^2 \psi(x,t) + V(x)\psi(x,t) + V_{\mathrm{ex}}(x,t)\psi(x,t), \quad x \in \mathbb{R}, \tag{6.4.1}$$

which is equivalent to the PDE discussed in [LZZ18]

$$i\partial_t u(x,t) = \left[-(\partial_x + iA(x,t))^2 + V(x,t)\right] u(x,t) \tag{6.4.2}$$

under the transformation $u(x,t) = e^{i\int_0^t V_{\mathrm{ex}}(x,s)\mathrm{d}s}\psi(x,t)$. Here, we shall use spectral methods with the Hermite function basis. The solution is complex, so in the spectral decomposition, the coefficients of the basis functions are complex. The major difference here is that in [LZZ18] the Schrödinger equation is solved in a bounded domain (x_-, x_+) with absorbing boundary conditions. Using spectral methods, we are able to solve the Schrödinger equation without truncating the domain.

We solve the weak form of Eq. (6.4.1)

$$(\partial_t\psi, v) = -i(\partial_x\psi, \partial_x v) - i((V(x) + V_{\text{ex}}(x,t))\psi, v), \quad v \in L^2(-\infty, \infty), \tag{6.4.3}$$

which is to find $\Psi^{\beta}_{N,x_L}(t,x) \coloneqq \sum_{i=0}^{N} \psi^{\beta}_{i,x_L}(t)\hat{\mathcal{H}}^{\beta}_i(x - x_L)$ in $V^{\beta}_{N,x_L} = \text{span}\{\hat{\mathcal{H}}^{\beta}_i(x - x_L)\}_{i=0}^{N}$ satisfying the initial condition and

$$(\partial_t\Psi^{\beta}_{N,x_L}, v) + i(\partial_x\Psi^{\beta}_{N,x_L}, \partial_x v) = -i((V(x) + V_{\text{ex}}(x,t))\Psi_{N,x_L}, v), \quad \forall v \in V^{\beta}_{N,x_L}. \tag{6.4.4}$$

We denote $\boldsymbol{\psi}^{\beta}_{N,x_L}(t) \coloneqq (\psi^{\beta}_{0,x_L}(t), ..., \psi^{\beta}_{N,x_L}(t))$, which can be analytically solved to advance time

$$\boldsymbol{\psi}^{\beta}_{N,x_L}(t_{n+1}) = \exp\left[-i\int_{t_n}^{t_{n+1}} (D^{\beta}_N + V^{\beta}_{N,x_L}(t))\mathrm{d}t\right]\boldsymbol{\psi}^{\beta}_{N,x_L}(t_n) \tag{6.4.5}$$

where $D^{\beta}_N \in \mathbb{R}^{(N+1)\times(N+1)}$ is a symmetric matrix with entries

$$(D^{\beta}_N)_{\ell j} = \begin{cases} -\beta^2\sqrt{\ell(\ell+1)} & j = \ell+2, \\ -\beta^2\sqrt{(\ell-2)(\ell-1)} & j = \ell-2, \\ \beta^2\dfrac{\ell}{2} & j = \ell, \\ 0 & \text{otherwise}, \end{cases} \tag{6.4.6}$$

and the matrix $V^{\beta}_{N,x_L}(t) \in \mathbb{R}^{(N+1)\times(N+1)}$ has entries

$$(V^{\beta}_{N,x_L}(t))_{\ell j} = \int_{-\infty}^{\infty} (V(x) + V_{\text{ex}}(x,t))\hat{\mathcal{H}}^{\beta}_{\ell-1}(x - x_L)\hat{\mathcal{H}}^{\beta}_{j-1}(x - x_L)\mathrm{d}x. \tag{6.4.7}$$

The evaluation of $\exp(-i\int_{t_n}^{t_{n+1}}(D^{\beta}_N + V^{\beta}_{N,x_L}(t))\mathrm{d}t)\boldsymbol{\psi}^{\beta}_{N,x_L}(t_n)$ is performed as follows. First, we denote $\tilde{V}^{\beta}_{N,x_L} \approx \int_{t_n}^{t_{n+1}} V^{\beta}_{N,x_L}(t)\mathrm{d}t$ where the integration is evaluated by Gauss-Legendre formula. Therefore, when calculating the matrix-vector product $\tilde{V}^{\beta}_{N,x_L}\mathbf{X}_N$ for a vector $\mathbf{X}_N \coloneqq (X_1, ..., X_N) \in \mathbb{R}^{N+1}$, its ℓ^{th} component is

$$(\tilde{V}_{N,x_L}\mathbf{X}_N)_\ell = \sum_{j=0}^{N}\sum_{s=0}^{N}\hat{H}_{\ell-1}^{\beta}(x_s^\beta)\hat{H}_j^\beta(x_s^\beta)\Big[V(x_s^\beta+x_L)+\tfrac{5}{18}V_{\mathrm{ex}}(x_s^\beta+x_L,t_n+\tfrac{1}{2}(1-\sqrt{\tfrac{3}{5}})\mathrm{d}t)$$
$$+\tfrac{4}{9}V_{\mathrm{ex}}(x_s^\beta+x_L,t_n+\tfrac{\mathrm{d}t}{2})+\tfrac{5}{18}V_{\mathrm{ex}}(x_s^\beta+x_L,t_n+\tfrac{1}{2}(1+\sqrt{\tfrac{3}{5}})\mathrm{d}t)\Big]X_j\Delta t \tag{6.4.8}$$

where $\Delta t = t_{n+1}-t_n$. We can first calculate

$$\sum_{j=0}^{N}\hat{H}_j^\beta(x_s^\beta)\Big[V(x_s^\beta+x_L)+\tfrac{5}{18}V_{\mathrm{ex}}(x_s^\beta+x_L,t_n+\tfrac{1}{2}(1-\sqrt{\tfrac{3}{5}})\mathrm{d}t)$$
$$+\tfrac{4}{9}V_{\mathrm{ex}}(x_s^\beta+x_L,t_n+\tfrac{\mathrm{d}}{2})+\tfrac{5}{18}V_{\mathrm{ex}}(x_s^\beta+x_L,t_n+\tfrac{1}{2}(1+\sqrt{\tfrac{3}{5}})\mathrm{d}t)\Big]X_j\Delta t \tag{6.4.9}$$

for each subindex s; then, evaluating $(\tilde{V}_{N,x_L}^\beta\mathbf{X}_N)_\ell$ for each subindex ℓ will only require an $O(N)$ operation. In this way, given any arbitrary potentials $V(x), V_{\mathrm{ex}}(x,t)$ we can calculate $\tilde{V}_{N,x_L}^\beta\mathbf{X}_N$ in $O(N^2)$ operations without explicitly calculating entries in $\tilde{V}_{N,x_L}^\beta$. We approximate the matrix-vector product $\exp\left[-i(D_N^\beta\Delta t+\tilde{V}_{N,x_L})\right]\boldsymbol{\psi}_{N,x_L}^\beta(t_n)$ in the following way: we rewrite $\exp\left[-i(D_N\Delta t+\tilde{V}_{N,x_L})\right]\boldsymbol{\psi}_{N,x_L}^\beta(t_n)=\exp\left[-\frac{1}{m}i(D_N\Delta t+\tilde{V}_{N,x_L})\right]^m\boldsymbol{\psi}_{N,x_L}^\beta(t_n)$, which is introduced as the "scaling and squaring" method in [MV78], and approximate the matrix-vector product $\exp\left[-\frac{1}{m}i(D_N\Delta t+\tilde{V}_{N,x_L})\right]\mathbf{X}_N$ by truncating the infinite Taylor expansion series $\sum_{j=0}^{\infty}\frac{1}{m^j j!}\left[-i(D_N\Delta t+\tilde{V}_{N,x_L})\right]^j\mathbf{X}_N$. Here, we take $m=3$.

As mentioned in Section 6.1, two main numerical difficulties when solving the Schrödinger equation are the unboundedness and the oscillatory behavior of the solutions. In fact, the solution may be increasingly oscillatory behavior at infinity over time, making it very hard to solve in the unbounded domain. However, with our adaptive spectral methods, we can efficiently solve the Schrödinger equation in unbounded domains accurately and capture these oscillations.

We now revisit the examples of linear Schrödinger equations discussed in [LZZ18], nonlinear Schrödinger equations discussed in [TA84], and their semiclassical limits studied in

[IKS19]. In the following examples, curves labeled "adaptive" indicate that scaling, moving, and p-adaptive techniques are all applied as described in Fig. 6.5, while curves labeled "non-adaptive" indicate none of the three adaptive techniques is applied. Results obtained by applying some of the three techniques are marked by curves named by the corresponding techniques that are applied.

Example 17. We numerically solve the Schrödinger equation which is solved in Example 1 of [LZZ18] and take $V = V_{ex} = 0$ in Eq. (6.4.1), admitting the analytic solution

$$\Psi(x,t) = \frac{1}{\sqrt{\zeta + it}} \exp\left[ik(x - kt) - \frac{(x - 2kt)^2}{4(\zeta + it)}\right], \tag{6.4.10}$$

where k is related to the propagation speed of the beam and ζ determines the width of the beam. The absolute values of the real part of $\Psi(x, t = 0, 0.5, 1)$ are plotted in Fig. 6.7(a), illustrating the increasingly oscillatory and diffusive behavior in the rightward propagating solution. Treatment of this solution will thus require scaling, moving, and p-adaptive techniques. The imaginary parts of the reference solution (not plotted) over time are also increasingly oscillatory. We shall apply the algorithm described in Fig. 6.5. We set $\zeta = 0.3, k = 1$, and initialize $N = 50$ at $t = 0$. Other parameters are set to $q = \nu^{-1} = 0.95, \mu = 1.0002, d_0 = 0.005, \underline{\beta} = 0.3, \overline{\beta} = 2, d_{\max} = 0.1, N_{\max} = 6, N_{\min} = 0, \eta = 1.1, \gamma = 1.05$, and $\Delta t = 0.005$. Note that with zero potential, Eq. (6.4.5) reduces to

$$\boldsymbol{\psi}_N^{\beta}(t_{n+1}) = \exp(-iD_N^{\beta}\mathrm{d}t)\boldsymbol{\psi}_N^{\beta}(t_n). \tag{6.4.11}$$

When all four techniques are applied, the error is the smallest (shown in Fig. 6.7(b)) since we can keep the exterior error indicator in (x_R, ∞) small (shown in Fig. 6.7(c)) by matching the solution's intrinsic advection. We can simultaneously prevent the frequency indicator from growing too fast (shown in Fig. 6.7(d)), thus ensuring a small error bound.

From the reference solution, one observes that increasing the expansion order over time is an intrinsic requirement and failure to do so prevents the capture of the increasing oscillations, leading to a huge error. As the function becomes increasingly oscillatory as $x \to \infty$,

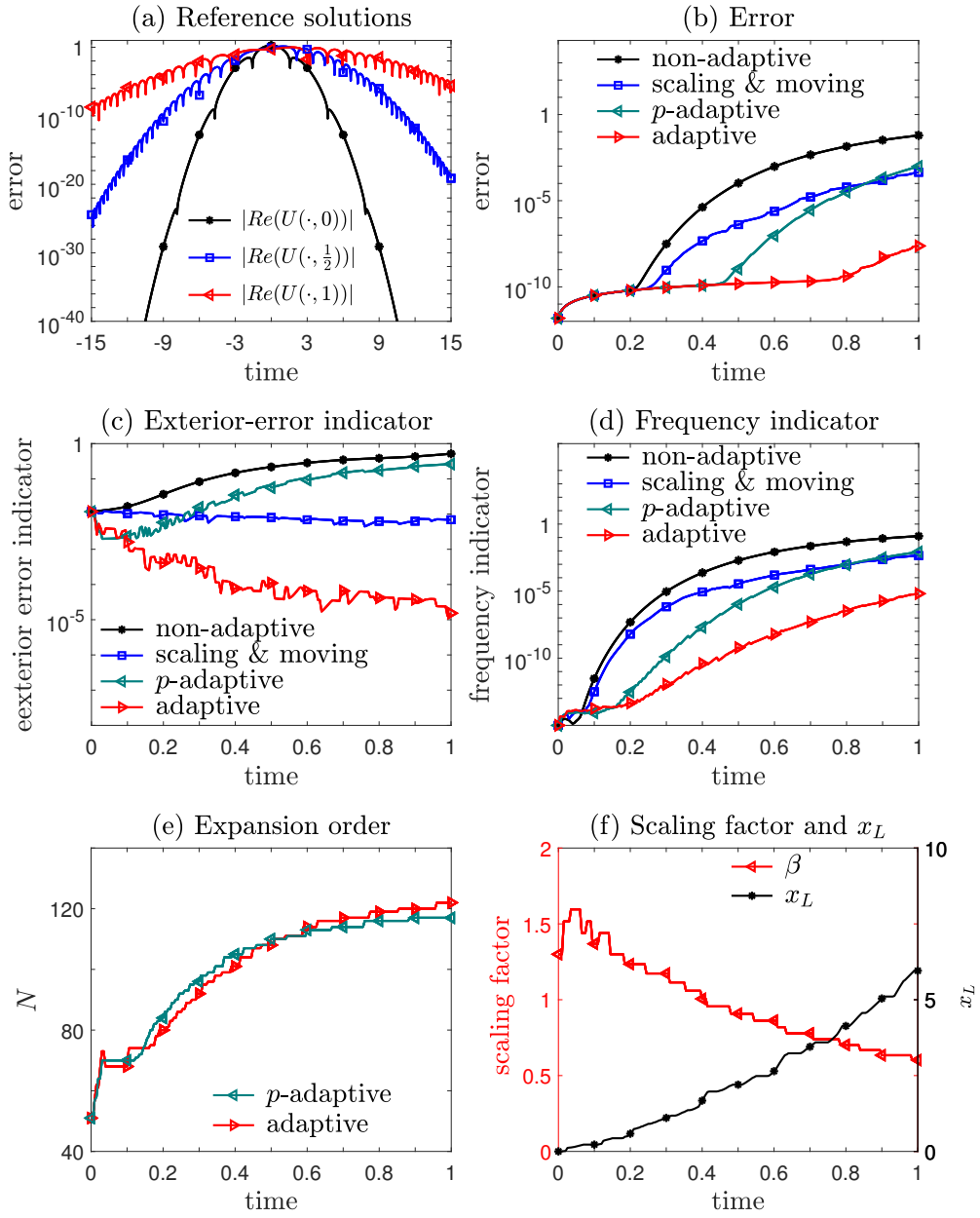

Figure 6.7: Numerically solving the Schrödinger equation with vanishing potentials. Applying scaling, moving, and p-adaptive techniques can successfully capture diffusive advective, and oscillatory behavior of the solution and yields an accurate numerical solution that prevents the frequency indicator from growing too fast. The exterior-error indicator is also kept small by moving the basis functions rightward to avoid a deteriorating approximation at ∞. Failure to incorporate any of the moving, scaling, or p-adaptive techniques results in a much larger error.

moving the basis rightward requires correspondingly more refinement (shown in Fig. 6.7(e)). However, the p-adaptive method alone cannot compensate for the inability to capture diffusion and advection, resulting in an inaccurate approximation. We have also checked that apart from what is shown in Fig. 6.7, applying any single scaling, moving or p-adaptive technique, or combining any two of them will all result in a much larger error than employing all three techniques indicated in Fig. 6.5. The adaptive spectral method produced an error of 3.4572×10^{-8} and required 162.6 seconds of laptop run-time. If we use a fixed $N = 116$, which is the largest expansion order during $t \in [0, 1]$, while activating the scaling and moving techniques 199.2 seconds of run-time is required for an error of 2.0661×10^{-8}. This example verifies that the p-adaptive technique can provide computational savings through adaptive adjustment of the expansion order while maintaining an equivalent accuracy as by using the largest expansion order.

Example 18. We solve the following 2-D Schrödinger equation in $\mathbb{R}^2 \times \mathbb{R}^+$:

$$i\partial_t \psi(x, y, t) = -\Delta\psi(x, y, t), \tag{6.4.12}$$

that admits the analytic solution

$$\Psi(x, y, t) = \frac{1}{\sqrt{\zeta_x + it}} \exp\left[-\frac{x^2}{4(\zeta_x + it)}\right] \cdot \frac{1}{\sqrt{\zeta_y + it}} \exp\left[\frac{y^2}{4(\zeta_y + it)}\right], \tag{6.4.13}$$

which is diffusive and becomes increasingly oscillatory over time in both dimensions. Also, the function is heterogeneous and requires different scaling and adjustment of the expansion orders in each dimension. We shall still solve the weak form by similar schemes described in Eqs. (6.4.4) and (6.4.5) to forward time. Since the function is not advecting over time, we deactivate the moving technique by setting the maximal displacement to 0. We set $\zeta_x = 0.5, \zeta_y = 0.3$ in Eq. (6.4.13) with which the function is more oscillatory and diffusive in x, the initial scaling factors: $\beta_x = 1, \beta_y = 1.2$, the initial expansion orders: $N_x = N_y = 50$, $q_x = q_y = \nu_x^{-1} = \nu_y^{-1} = 0.95$ for scaling, and $\eta_x = \eta_y = 1.02, \gamma_x = \gamma_y = 1$ for p-adaptivity.

Because of the application of the frequency indicators in both directions, the spectral

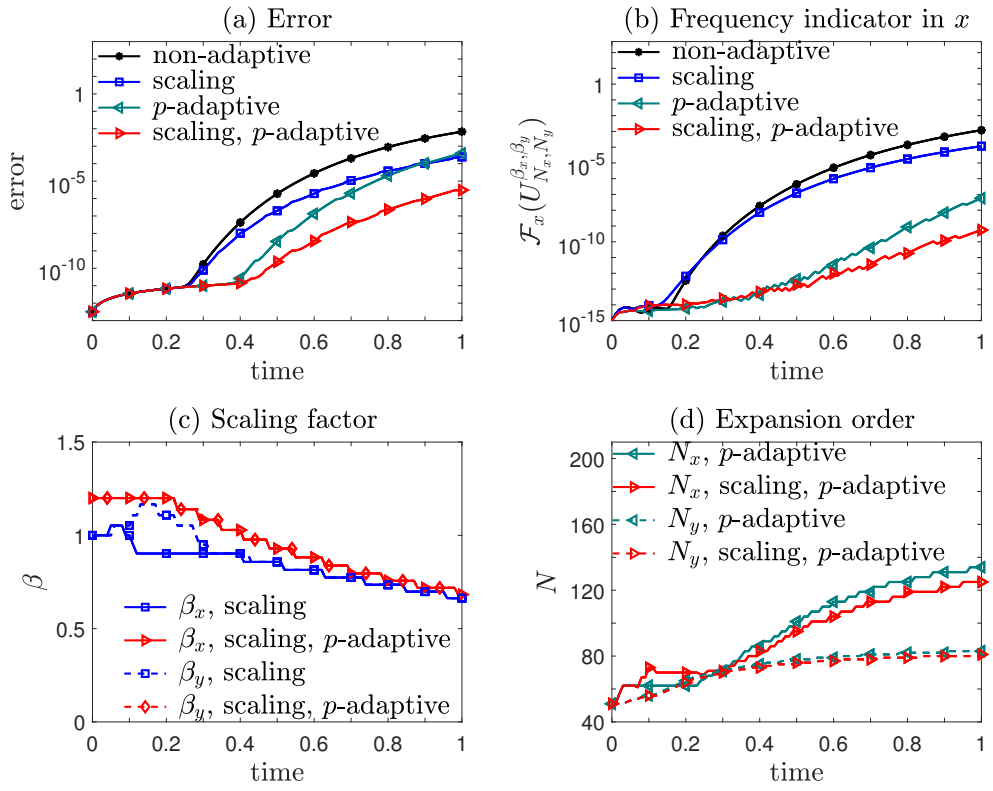

Figure 6.8: Numerically solving the 2-D Schrödinger equation Eq. (6.4.12). Applying scaling and p-adaptive techniques can capture diffusive and oscillatory behavior of the solution. The solution is heterogeneous in each dimension and requires adjusting the scaling factors and frequency indicators differently in x- and y-directions.

method with the p-adaptive and scaling techniques leads to the smallest error (Fig. 6.8(a)). In Fig. 6.8(b) we plot frequency indicators in the x-direction and show that the spectral method with both scaling and p-adaptive techniques gives rise to the smallest values. Furthermore, since the function is more diffusive and oscillatory in x, β_x, the scaling factor in x is decreased more than β_y when only scaling is used, as shown in Fig. 6.8(c). Expansion orders N_x in both scaling and scaling + p-adaptive methods increases faster than N_y, as shown in Fig. 6.8(d). Fig. 6.8(d) also shows that proper scaling can avoid unnecessary increases in the expansion order, in this case, N_y. As shown in Fig. 6.8(a), increasing the expansion order without scaling leads to an even larger error at $t = 1$ than that obtained under pure scaling.

For comparison, we also use the spectral method without the p-adaptive technique and fix the expansion orders $N_x = 124, N_y = 80$, the largest values reached under the p-adaptive

and scaling techniques. The error 3.1852×10^{-5} is comparable to the error 3.0366×10^{-5} of the adaptive method, but the run-time 7.1705×10^4 seconds (when using fixed $N_x = 124, N_y = 80$) is significantly larger than the 4.9508×10^4 seconds necessary for the adaptive method. Therefore, using the p-adaptive technique to adjust the expansion order efficiently can significantly reduce computational costs without sacrificing accuracy.

Example 19. We numerically solve the following nonlinear Schrödinger equation in $\mathbb{R}$ [TA84]:

$$i\partial_t \psi = \partial_x^2 \psi + 2|\psi|^2 \psi, \quad \psi(x, 0) = 2e^{-2ix}\text{sech}(2x), \tag{6.4.14}$$

which admits an analytic solution

$$\psi(x, 0) = 2e^{-2ix}\text{sech}(2x - 8t). \tag{6.4.15}$$

Clearly, the function translates rightward, requiring a corresponding translation of the basis functions. In Example 16 we have demonstrated that successfully capturing the diffusive behavior of the numerical solution by the scaling technique can prevent the unnecessary increase in the expansion order. Here, we show that successfully capturing the advective behavior of the solution can also help avoid the unnecessary increase of the expansion order, reducing computational costs. Since the solution does not exhibit diffusive behavior, we use the algorithm described in Fig. 6.5 but deactivate the scaling technique.

For the p-adaptive technique, we set $\eta = 1.3, \gamma = 1$, $N_{\max} = 20$, and $N_{\min} = 0$ while for the moving technique we set $\mu = 1.00005, \delta = 0.0005$, $d_{\max} = 0.01$, the initial expansion order $N = 120$, the scaling factor $\beta = 1.3$, and set the timestep $\Delta t = 0.0005$. An explicit third-order Runge Kutta scheme is used to forward time.

Spectral methods including the moving technique can achieve the highest accuracy and maintain the smallest frequency indicator as shown by the green and red curves in Figs. 6.9(a, b), respectively. Since the oscillatory behavior of the solution does not vary over time

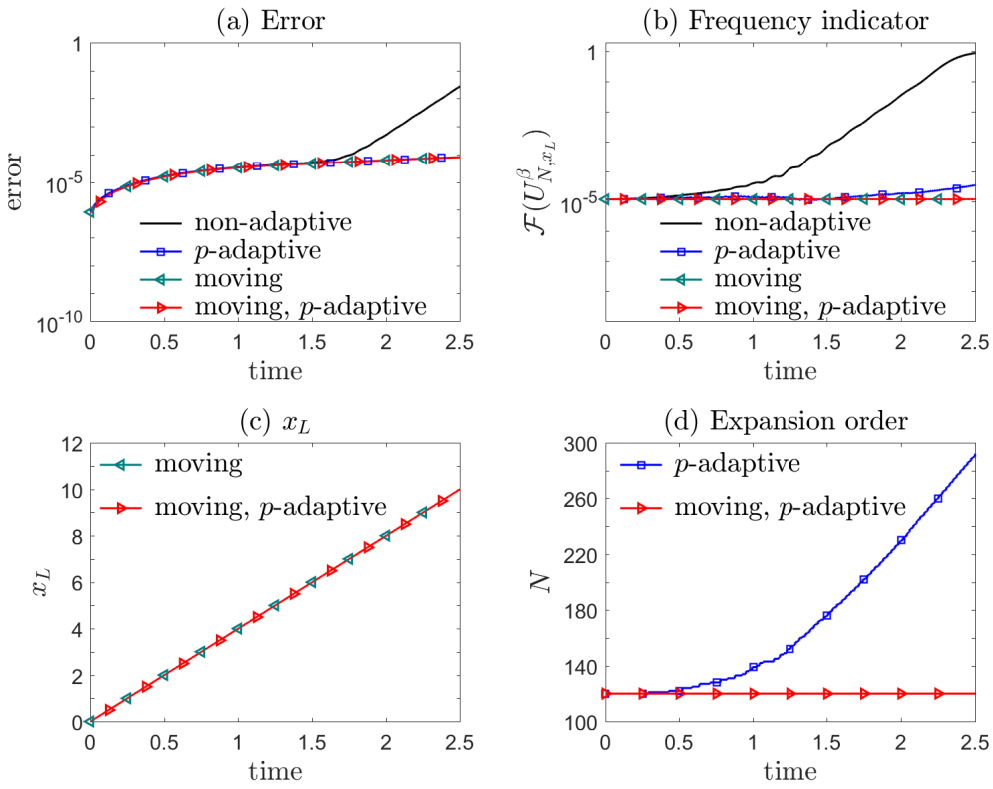

Figure 6.9: Numerically solving the nonlinear Schrödinger equation in Eq. (6.4.14). The solution translates rightward which may cause a false increase in the frequency indicator leading to a large error if the moving technique is not applied. If moving is not applied, the expansion order will need to be increased to give an accurate solution. However, by properly moving the basis functions rightward using the moving technique, accuracy can be maintained without increasing the expansion order. Therefore, the moving technique is required in addition to the p-adaptive method.

and the function translates to the right with speed $4s^{-1}$, the p-adaptive technique will not be activated as long as we properly move the basis functions rightward. Translation can be accurately captured by the moving technique (Figs. 6.9(c, d)). However, as shown in Fig. 6.9(d), without the moving technique, the p-adaptive method increases the expansion order to maintain accuracy, resulting in a higher computational cost. As mentioned in [XSC21b], translation will induce a false increase in the frequency indicator. Successfully resolving the translation can be used to prevent unnecessary increases in the expansion order. In this example, the moving and p-adaptive technique requires 2.1569×10^3 seconds of run-time while the p-adaptive technique without moving required 3.8235×10^3 seconds. Thus, it is important to first properly move the basis functions before adjusting the expansion order.

Next, we numerically solve the Schrödinger equation with non-vanishing potentials.

Example 20. We numerically solve the following standard Schrödinger equation Eq. (6.4.1) equivalent to Example 2 in [LZZ18] with the potentials

$$V_{ex}(x,t) = \frac{50}{\sqrt{\pi}}\sin(10t)\int_{-\infty}^{x}\exp(-z^2)\mathrm{d}z,\ V(x) = -10\left[e^{-10(x-1)^2} + e^{-10(x+1)^2}\right]. \quad (6.4.16)$$

Given an even function as the initial condition for Example 2 in [LZZ18], the solution is also an even function and the solution of Eq. (6.4.1) obeys $|\psi(-x,t)| = |\psi(x,t)|$. No bias towards $-\infty$ or $+\infty$ is preferred. Therefore, we use the Hermite function basis and apply the algorithm described in Fig. 6.5 but deactivate the moving technique by setting $d_{max} = 0$. We use the same initial condition as in Example 17 and set $\eta = 1.025, \gamma = 1, q = 0.95, \nu = q^{-1}, N_{\min} = 0, N = 200, \underline{\beta} = 0.3, \overline{\beta} = 2$, and $\beta_0 = 1.3$ at $t = 0$ with the maximum expansion order increment in each step $N_{\max} = 20$.

The reason why we set $\gamma = 1$ is that the expansion order N needs to be increased quickly to catch up with the rapidly increasing oscillatory behavior of the numerical solution. We set a uniform timestep $\Delta t = 0.01$ and use only the scaling technique with fixed $N = 2500$ to find the reference solution. For the p-adaptive method, we added an additional restriction that the expansion order cannot exceed $N = 2500$ of the reference solution.

We can easily see that the spectral method with both scaling and p-adaptive techniques outperforms the non-adaptive spectral method or with only one of these two techniques employed (shown in Fig. 6.10(a)). The frequency indicator of using both scaling and p-adaptive techniques is also the smallest (Fig. 6.10(b)), and the similarity between the frequency indicator and error is again confirmed as stated in [XSC21b]. Moreover, the unscaled method will result in a larger expansion order (Fig. 6.10(a)), leading to excessive refinement with no improvement in accuracy (Fig. 6.10(a)). In this example, the coarsening procedure will not lead to a large increase of frequency indicator and does not significantly compromise the accuracy (Figs. 6.10(b, c)). Finally, the scaling factors of the p-adaptive spectral method and the non-p-adaptive spectral method trend similarly over time; they both decrease after

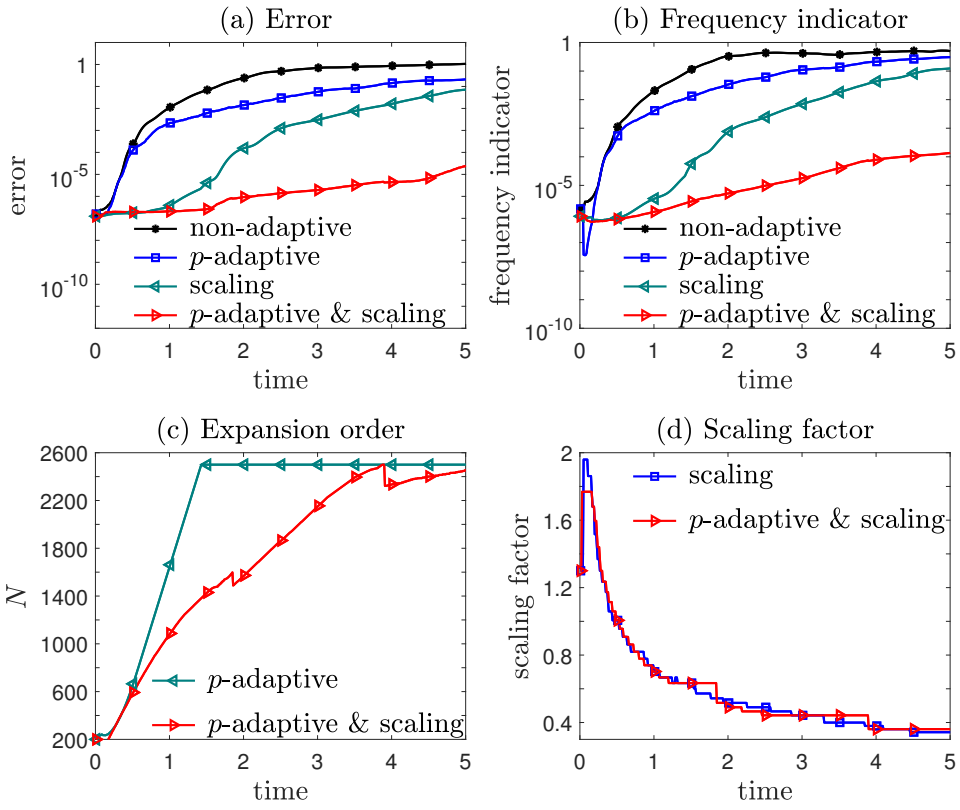

Figure 6.10: Numerically solving the Schrödinger equation with non-vanishing potentials. Rapidly increasing oscillations of the solution over time require much refinement and proper scaling to maintain accuracy. It is again verified that proper scaling can avoid unnecessary refinement and avoid unnecessary computational burden by adaptively adjusting the scaling factor. Without scaling, the expansion order soon reaches the upper bound for N (the expansion order of the reference solution) and the approximation soon deteriorates due to an inability to further increase N or adjust β and maintain a low frequency indicator. Failure to accommodate the p-adaptive technique will also result in a larger error because of an inability to capture the oscillatory behavior.

experiencing an initial, transient increase (Fig. 6.10(d)).

Example 21. Finally, we consider solving a Schrödinger equation,

$$i\partial_t \psi(x,t) = -\varepsilon \partial_x^2 \psi(x,t) + \frac{1}{\varepsilon} V(x,t), \tag{6.4.17}$$

in the semiclassical regime in which the solution can become more oscillatory over time, especially when $\varepsilon \to 0^+$ as illustrated in [IKS19]. A p-adaptive technique to increase the expansion order is thus needed. We investigate whether our p-adaptive technique can suc-

cessfully solve Eq. (6.4.17) as $\varepsilon \to 0^+$ and how the evolution of expansion order N depends on ε.

We assume a time-dependent potential

$$V(x,t) = \sin(\frac{t}{8}) \exp(-10x^2) \tag{6.4.18}$$

which is even in $\mathbb{R}$ and use the same even initial condition as in Example 17. Since the solution remains even, no translation of the basis functions is needed. Therefore, as in Example 20, we deactivate the moving technique by setting $d_{max} = 0$. We set $\eta = 1.02, \gamma = 1, q = 0.95, \nu = q^{-1}, N_{\min} = 0, N = 100, \underline{\beta} = 0.3, \overline{\beta} = 2$, and $\beta_0 = 1.3$ at $t = 0$ and impose a maximum expansion order increment $N_{\max} = 24$ at each timestep $\Delta t = 0.01$. As a reference solution, we solve Eq. (6.4.17) using a fixed $N = 2500$ with the scaling technique and investigate the cases $\varepsilon = 0.1, 0.01$, and 0.001.

Since the solution increases its oscillations faster as ε becomes smaller, the expansion order needs to be increased accordingly. As shown in Fig. 6.11, under the p-adaptive technique, the smaller the ε, the faster the rate of increase of the expansion order N. In this example, no intrinsic diffusion of the solution is detected and scaling is not activated in the p-adaptive technique as long as the expansion order is properly increased. Figs. 6.11(b, c, d) show the errors for $\varepsilon = 0.1, 0.01$, and 0.001, respectively. Without adaptive methods, the errors increase significantly as ε decreases, but by employing p-adaptivity, the expansion order is adjusted to control the errors, dramatically reducing their increase across time, as shown in Figs. 6.11(b, c).

When ε becomes even smaller (cf $\varepsilon = 0.001$ in Fig. 6.11(d)), the solution becomes extremely oscillatory and the expansion order needs to be increased too fast even for the current implementation of the p-adaptive method to accommodate. Very small ε in Eq. (6.4.17) poses an intrinsic numerical difficulty that requires an extremely large expansion order N. In the extremely small ε regime, methods like Magnus-Zassenhaus splittings [IKS19], which becomes more accurate as $\varepsilon \to 0^+$, could be used. Thus, our p-adaptive technique is most ap-

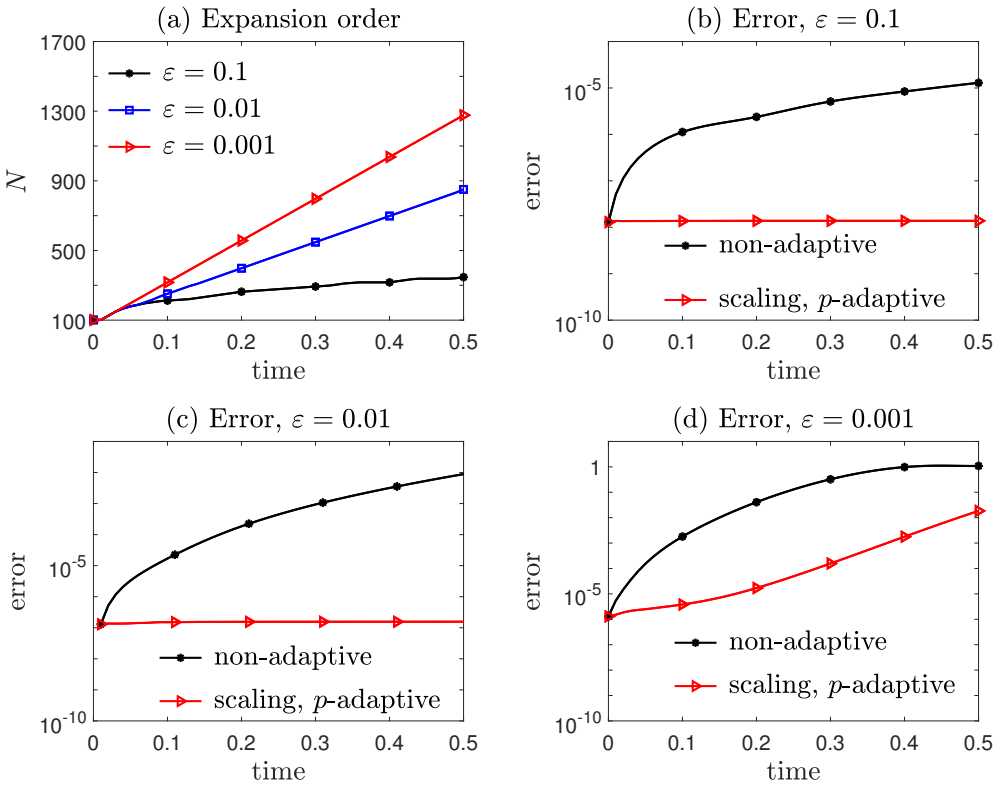

Figure 6.11: Numerically solving the Schrödinger equation in Eq. (6.4.17) with a time-dependent potential in Eq. (6.4.18) as $\varepsilon \to 0^+$. Rapidly increasing oscillations of the solution require significant refinement by the p-adaptive technique in order to maintain accuracy. The expansion order increases faster over time as ε becomes smaller. In general, the p-adaptive technique is appropriate for solving Eq. (6.4.17) in the mesoscopic regime for ε that is not too small.

plicable to the mesoscopic regime of Schrödinger equations of the form Eq. (6.4.17) in which ε is not too small. Nonetheless, our p-adaptive technique can efficiently capture oscillations and complements existing methods designed for very small ε.

6.5 Summary and conclusions

In this chapter, we proposed a frequency-dependent p-adaptive technique that adjusts the expansion order for spectral methods. We demonstrated its applicability to time-dependent problems with varying oscillatory behavior. In order to develop efficient numerical methods for problems requiring solutions in unbounded domains, we also combined the p-adaptive

technique with scaling (r-adaptivity) and moving (h-adaptivity) methods to devise a complete adaptive spectral method that can successfully deal with diffusion, advection, and oscillation. Through a number of numerical examples, we explored the relationship among the three building blocks: the scaling, moving, and p-adaptive techniques. In particular, the proposed p-adaptive technique enables us to adjust the expansion order dynamically, boosting the efficiency of the spectral method. We also investigated the relationship between scaling and p-adaptive techniques for spectral methods in unbounded domains, both of which depend on the same frequency indicator.

Our adaptive spectral method was also used to numerically solve examples of different forms and limits of the Schrödinger equation. The solutions to these examples can contain rapid oscillations across the whole domain that evolve in time, posing numerical difficulties for existing numerical methods that truncate the domain. However, this type of problem can be efficiently resolved by our p-adaptive spectral methods. We find the proposed approaches are most effective in the mesoscopic, semiclassical regime of the Schrödinger equation where ε is not too small.

Further analysis of the proposed methods can be performed. The relationship among the adaptive techniques for spectral methods, scaling, moving, refinement, and coarsening, can be further studied and rigorous numerical analysis for these techniques should be investigated. Furthermore, fast algorithms with mapped Chebyshev polynomials for solving PDEs in unbounded domains have been developed using the fast Fourier transform [STW20]. Thus, generalizing these adaptive methods for mapped Jacobi polynomials may be a compelling future research direction.

CHAPTER 7

Adaptive Hermite spectral methods in unbounded domains

This is the Accepted Manuscript version of an article accepted for publication in *Applied Numerical Mathematics*, **183**, (2023), pp.201-220. It is an open-access paper. The Version of Record is available online at [10.1016/j.apnum.2022.09.003].

7.1 Introduction

Unbounded domain problems require efficient numerical methods for computation. For example, resolving the decay of the solution of Schrödinger's equations at infinity requires efficient unbounded domain algorithms [LZZ18]. In population dynamics, tracking cell volume blowup in structured population PDE models demands high-accuracy numerical methods in unbounded domains [XC21, XGC20]. Furthermore, in solid-state physics, numerical methods for unbounded domains are required for studying long-range particle interactions [HDO12, MHR11]. Despite these numerous applications, there has been little research on developing efficient and accurate algorithms for solving models in unbounded domains.

Adaptive methods, such as re-defining grids for finite difference methods [RW00] and re-generating meshes for finite element methods [ACV19, BFH12, LLM02, TT03], which are applied to PDEs defined on finite domains, can dramatically improve not only accuracy but computational efficiency. Recently, novel adaptive techniques for spectral methods have been developed and incorporated into efficient algorithms for numerically solving PDEs in unbounded domains that posed substantial numerical difficulties when using previous numerical methods [XSC21b, XSC21a]. These adaptive spectral techniques require tuning of three key parameters: the scaling factor β, the displacement of the basis function x_0, and the spectral expansion order N. For example, if we use the generalized Hermite functions [STW11] as basis functions on $\mathbb{R}$, the variables β, x_0, and N appear in a spectral expansion according to

$$U^{\beta}_{N,x_0} \coloneqq \sum_{i=0}^{N} u^{\beta}_{i,x_0} \hat{\mathcal{H}}_i(\beta(x - x_0)), \tag{7.1.1}$$

where u^{β}_{i,x_0} is the coefficient of the i^{th}-order Hermite function $\hat{\mathcal{H}}_i$. The adaptive techniques for spectral methods consist of three separate but interdependent procedures: the scaling technique which adjusts the shape of the basis functions, the moving technique which adjusts the displacement of the basis function, and the p-adaptive technique which adjusts the expansion order of the numerical solution. For example, for PDEs involving a spatial variable $x \in \mathbb{R}$ and a temporal variable $t \in [0, T]$, we typically impose a spectral expansion using

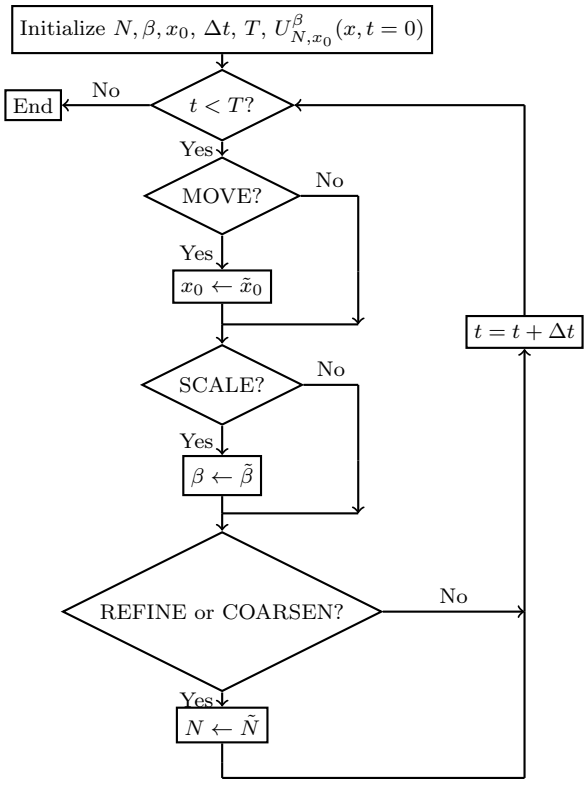

Figure 7.1: Flow chart of an adaptive Hermite spectral method equipped with scaling, moving, and p-adaptive techniques. x_0 and $\tilde{x}_0$ are the displacements before and after the moving technique is used. β and $\tilde{\beta}$ are the scaling factors before and after scaling when the scaling technique is used. N and $\tilde{N}$ are the expansion orders before and after adjusting the expansion order when the p-adaptive technique is used.

generalized Hermite functions of x and forward time t starting from an initial condition at $t = 0$. Adaptive spectral techniques are implemented as shown in Fig. 7.1.

The major advantage of the proposed adaptive spectral method Alg. 7.1 is that it depends only on the numerical solution U^{β}_{N,x_0} and thus does not require any prior knowledge of how the solution will evolve. The adaptive spectral method can automatically interrogate the behavior of the solution through a *frequency indicator* that measures a numerical solution's spread and oscillation and an *exterior-error indicator* that measures the solution's error outside a given domain. Both of these indicators are defined in [XSC21a]. Despite the numerical success of adaptive spectral methods when applied on unbounded domains, there exists no theoretical analysis of how the parameters β, x_0, and N affect the algorithm's performance and therefore there is thus far no general rule on how to best adjust these

parameters to minimize errors. Since the improper adjustment of β, x_0, and N can lead to large errors [Tan93, XG22], properly choosing them is crucial for the effective implementation of adaptive spectral methods.

In this chapter, we carry out a numerical analysis of the adaptive spectral method to specify how algorithm parameters affect the accuracy of numerical results. We restrict ourselves to generalized Hermite functions as basis functions for a parabolic model problem and explore how parameters in the adaptive spectral algorithm control the tuning of the three key quantities β, x_0, and N and thereby its numerical performance.

Depending on the inverse inequality for generalized Hermite functions [STW11], such analyses for numerically solving unbounded-domain PDEs provide a posterior error estimate. This error estimate only relies on the numerical solution and the adjustment of β, x_0, and N. Our main result is

Theorem 3. The L^2-error at time T when solving a parabolic PDE in $(x,t) \in \mathbb{R} \times [0,T]$ with the generalized Hermite functions and using adaptive techniques is bounded by

$$e(T) \coloneqq \|u(x,T) - U^{\beta}_{N,x_0}(x,T)\|_2 \leq e_0 + e_{\mathrm{S}} + e_{\mathrm{M}} + e_{\mathrm{C}}, \tag{7.1.2}$$

where U^{β}_{N,x_0} is the numerical solution; e_0 is the error of numerically solving the PDE without adjusting β, x_0, N. e_{S} is the error bound arising from changing the scaling factor from β to $\tilde{\beta}$; e_{M} is the error bound for changing the displacement from x_0 to $\tilde{x}_0$; e_{C} is the error bound for coarsening, *i.e.*, reducing the expansion order from N to $\tilde{N}$. More specifically, $e_{\mathrm{S}}, e_{\mathrm{M}}$, and e_{C} take the following forms

$$\begin{aligned}
e_{\mathrm{S}} &\coloneqq \sum_{scale} \frac{|\tilde{\beta} - \beta|\sqrt{1 + \frac{\tilde{\beta}}{\beta}}}{\sqrt{2}\tilde{\beta}} \|x\partial_x U^{\beta}_{N,x_0}(x,t)\|_2, \\
e_{\mathrm{M}} &\coloneqq \sum_{move} |x_0 - \tilde{x}_0| \|\partial_x U^{\beta}_{N,x_0}(x,t)\|_2, \\
e_{\mathrm{C}} &\coloneqq \sum_{coarsen} \|(I - \pi^{\beta}_{\tilde{N},x_0}) U^{\beta}_{N,x_0}(x,t)\|_2,
\end{aligned} \tag{7.1.3}$$

symbol	definition
$\hat{\mathcal{H}}^{\beta}_{i,x_0}$	generalized i^{th}-order Hermite function with a scaling factor β and displacement x_0, defined in $\mathbb{R}$ as $\hat{\mathcal{H}}^{\beta}_{i,x_0} \coloneqq \hat{\mathcal{H}}_i(\beta(x-x_0))$
P^{β}_{N,x_0}	function space $P_N^{x_0,\beta} \coloneqq \{\hat{\mathcal{H}}^{\beta}_{i,x_0}\}_{i=0}^N$
I	the identity operator
π^{β}_{N,x_0}	the projection operator $\pi^{\beta}_{N,x_0} : L^2(\mathbb{R}) \to P_N^{x_0,\beta}$ such that $(\pi_N^{x_0,\beta}u(x),\, u(x) - \pi_N^{x_0,\beta}u(x)) = 0$
$\mathcal{I}^{\beta}_{N,x_0}$	the interpolation operator $\mathcal{I}^{\beta}_{N,x_0} : L^2(\mathbb{R}) \to P_N^{x_0,\beta}$ such that $\mathcal{I}_N^{x_0,\beta}u(x_i) = u(x_i)$ where $\{x_i^N\}_{i=0}^N$ are collocation points of $\{\hat{\mathcal{H}}^{\beta}_{i,x_0}\}_{i=0}^N$
U^{β}_{N,x_0}	spectral expansion $U^{\beta}_{N,x_0} = \sum_{i=0}^N u^{\beta}_{i,x_0}\hat{\mathcal{H}}_i(\beta(x-x_0))$
N	expansion order of the spectral expansion
β	scaling factor of the generalized Hermite functions
x_0	displacement of the generalized Hermite functions
$\mathcal{E}_R(U^{\beta}_{N,x_0}), \mathcal{E}_L(U^{\beta}_{N,x_0})$	$\mathcal{E}_R$: the right exterior-error indicator of the spectral expansion U^{β}_{N,x_0}; $\mathcal{E}_L$: the left exterior-error indicator of the spectral expansion U^{β}_{N,x_0}
$\mathcal{F}(U^{\beta}_{N,x_0})$	frequency indicator for the spectral expansion U^{β}_{N,x_0}
q	scaling factor update (β to $\tilde{\beta}$) ratio ($\tilde{\beta} \leftarrow q^n\beta$ or $q^{-n}\beta, n \in \mathbb{N}^+$) in the scaling technique
ν	threshold for activating the scaling technique
δ	minimal displacement of updating the displacement x_0 to $\tilde{x}_0$ ($\tilde{x}_0 \leftarrow x_0 + nx_0$ or $x_0 - nx_0, n \in \mathbb{N}^+$) in the moving technique
μ	threshold for activating the moving technique
η	threshold for increasing the number of basis functions
η_0	threshold for decreasing the number of basis functions
γ	post-refinement adjustment factor for refinement threshold $\tilde{\eta} \leftarrow \gamma\eta$
$L^2(a,b;V)$	space of functions $\{f : [a,b] \to V$ (V is a Banach space) such that f is measurable for $\mathrm{d}t$ and $\int_a^b f(t)^2\mathrm{d}t < \infty\}$
$X(t_1,t_2)$	function space $\{f : f(x,s) \in L^2((t_1,t_2),t;H^1(\mathbb{R})), \partial_s f(x,s) \in L^2((t_1,t_2),t;H^1(\mathbb{R}))\}$
$e(t)$	L^2-norm of the error $\|u(\cdot,t) - U^{\beta}_{N,x_0}(\cdot,t)\|_{L^2}$ at time t

Table 7.1: **Overview of variables and notations.** List of the main variables and notations associated with the overall adaptive spectral method. Three key variables for adaptive spectral methods with generalized Hermite functions are the scaling factor β that determines the shape of the basis functions, the displacement of the basis functions x_0, and the expansion order N of the spectral decomposition.

where the sum $\sum_{scale}$ is taken over all scaling steps, the sum $\sum_{move}$ is taken over all translation steps, and $\sum_{coarsen}$ is taken over all coarsening steps. The operators I and $\pi^{\beta}_{\tilde{N},x_0}$ are defined in Table 7.1.

This result allows us to provide general guidelines for selecting the parameters in the adaptive spectral algorithm that lead to the proper tuning of β, x_0, and N. Note that the last three terms in Eq. (7.1.2) depend only on the numerical solution since the adaptive techniques depend only on the numerical solution and do not need any prior knowledge of the solution itself. From this theorem, we can conclude that the smaller the adjustment in the scaling factor or in the displacement of the basis functions, the smaller the error bounds $e_{\mathrm{S}}, e_{\mathrm{M}}$ for carrying out the adaptive techniques. However, given that improper β or x_0 leads to very large e_0, proper dynamic adjustment of β and x_0 are still needed to keep e_0 small, possibly at the expense of accumulating more error in $e_{\mathrm{S}}, e_{\mathrm{M}}$. Overall, it is desirable to adjust β, x_0 optimally to maintain a small e_0 while also avoiding large $e_{\mathrm{S}}, e_{\mathrm{M}}$.

Furthermore, by increasing the threshold of coarsening to prevent using too small an expansion order, the error resulting from reducing the expansion order e_{C} is smaller. However, there is a trade-off: increasing the threshold of coarsening to make it harder to decrease the expansion order would result in a larger N and thus higher computational cost. Although the effect of increasing the expansion order N (refinement) does not explicitly affect the error bound Eq. (7.1.2), lowering the refinement error to use a larger N usually leads to smaller errors as a result of more accurately solving PDEs (a smaller e_0).

In the next section, we formulate the model problem using generalized Hermite functions and perform numerical analysis. In Section 7.3, numerical analysis for applying the adaptive techniques is carried out and Theorem 3 is proved. In Section 7.4, numerical experiments are carried out, and an improvement of the adaptive spectral method is proposed. For completeness, we list the common variables and notations in Table 7.1 that we use throughout this chapter

7.2 Errors in solving a model problem with generalized Hermite functions

In this section, we first formulate a parabolic equation in the weak form [MST05]:

$$\big(\partial_t u(x,t), v(x)\big) + a\big(u(x,t), v(x)\big) = \big(f(x,t), v(x)\big), \quad x \in \mathbb{R}, t \in [0,T], \quad \forall v(x) \in H^1(\mathbb{R}), \tag{7.2.1}$$

$$\big(u(x,0), \tilde{v}(x)\big) = \big(u_0(x), \tilde{v}(x)\big), \quad \forall v(x) \in H^1(\mathbb{R}), \tag{7.2.2}$$

where $u_0(x) \in L^2(\mathbb{R})$ is the initial condition, $f(x,t)$ is the inhomogeneous source term (*e.g.* heat source in the heat equation), and $a(u,v)$ is a coercive symmetric bilinear form such that there exist constants $0 < c_0 < C_0$ satisfying

$$|a(u,v)| \le C_0 \|u\|_{H^1} \|v\|_{H^1} \quad \text{and} \quad c_0 \|u\|_{H^1}^2 \le a(u,u), \quad \forall u(x), v(x) \in H^1(\mathbb{R}). \tag{7.2.3}$$

In Eqs. (7.2.1), (7.2.2), and (7.2.3) and hereafter, the inner product is taken over the spatial variable x, and the norm $\|\cdot\|$ denotes the L^2-norm taken over x unless otherwise specified.

The solution to the model problem, Eqs. (7.2.1) and (7.2.2), exists and is unique [DL92], and the solution u is in the so-called Bochner-Sobolev space.

$$W\left(0,t; H^1(\mathbb{R}), H^{-1}(\mathbb{R})\right) \coloneqq \left\{ u : u(x,s) \in L^2\left(0,t; H^1(\mathbb{R})\right), \partial_s u(x,s) \in L^2\left(0,t; H^{-1}(\mathbb{R})\right) \right\} \tag{7.2.4}$$

where $H^{-1}(\mathbb{R})$ is the dual space of $H^1(\mathbb{R})$.

For simplicity, we assume that $f(x,t) \in C(\mathbb{R} \times [0,t]), \partial_s u(\cdot, s) \in L^2(0,t; H^1(\mathbb{R}))$ and

therefore $u \in X(0,t)$, and its norm is given by

$$\|u\|^2_{X(0,t)} = \int_0^t \Big(\|u(\cdot,s)\|^2_{H^1(\mathbb{R})} + \|\partial_s u(\cdot,s)\|^2_{H^1(\mathbb{R})} \Big) \mathrm{d}s + \|u(x,0)\|^2. \tag{7.2.5}$$

Analysis of finite element methods for solving Eqs. (7.2.1) and (7.2.2) for bounded x has already been performed [US19]. Here, we wish to numerically solve Eqs. (7.2.1) and (7.2.2) using spectral methods with generalized Hermite functions. We first fix the scaling factor β, the displacement x_0 of the basis functions $\hat{\mathcal{H}}^\beta_{i,x_0}$, and the expansion order N of the trial and test functions. Integrating Eq. (7.2.1) w.r.t time, we wish to find a $U^\beta_{N,x_0}(x,t) \in L^2(0,t;P^\beta_{N,x_0}(\mathbb{R}))$ such that for any test function $v^\beta_{N,x_0}(x,t) \in L^2(0,t;P^\beta_{N,x_0}(\mathbb{R}))$ and $\tilde{v}^\beta_{N,x_0} \in P^\beta_{N,x_0}(\mathbb{R})$,

$$\begin{aligned} \int_0^t \Big[\big(\partial_s U^\beta_{N,x_0}(x,s), v^\beta_{N,x_0}(x,t)\big) + a\big(U^\beta_{N,x_0}(x,t), v^\beta_{N,x_0}(x)\big) \Big] \mathrm{d}s + \big(U^\beta_{N,x_0}(x,0), \tilde{v}^\beta_{N,x_0}(x)\big) \\ = \int_0^t \big(f(x,s), v^\beta_{N,x_0}(x,s)\big)\mathrm{d}s + \big(u(x,0), \tilde{v}^\beta_{N,x_0}(x)\big), \\ \forall (v^\beta_{N,x_0}, \tilde{v}^\beta_{N,x_0}) \in L^2(0,t;P^\beta_{N,x_0}(\mathbb{R})) \times P^\beta_{N,x_0}(\mathbb{R}). \end{aligned} \tag{7.2.6}$$

For notational simplicity, we denote

$$\mathbf{v}^\beta_{N,x_0} \coloneqq (v^\beta_{N,x_0}, \tilde{v}^\beta_{N,x_0}), \quad Y^\beta_{N,x_0} \coloneqq L^2(0,t;P^\beta_{N,x_0}(\mathbb{R})) \times P^\beta_{N,x_0}(\mathbb{R}), \tag{7.2.7}$$

and equip $\mathbf{v}^\beta_{N,x_0} \in Y^\beta_{N,x_0}$ with the norm

$$\|\mathbf{v}^\beta_{N,x_0}\|^2_{Y^\beta_{N,x_0}} = \|\big(v^\beta_{N,x_0}(x,t), \tilde{v}^\beta_{N,x_0}(x)\big)\|^2_{Y^\beta_{N,x_0}} \coloneqq \int_0^t \|v^\beta_{N,x_0}(\cdot,s)\|^2_{H^1(\mathbb{R})}\mathrm{d}s + \|\tilde{v}^\beta_{N,x_0}(x)\|^2. \tag{7.2.8}$$

The solution $U^\beta_{N,x_0} \coloneqq \sum_{i=0}^N u^\beta_{i,x_0}(t)\hat{\mathcal{H}}^\beta_{i,x_0}(x)$ of Eq. (7.2.6) can be explicitly evaluated through the matrix equation

$$\boldsymbol{u}^\beta_{N,x_0}(t) = e^{-\boldsymbol{A}^\beta_N t}\boldsymbol{u}^\beta_{N,x_0}(0) + e^{-\boldsymbol{A}^\beta_N t}\int_0^t e^{\boldsymbol{A}^\beta_N s}\boldsymbol{F}_{N,x_0}(s)\mathrm{d}s, \tag{7.2.9}$$

where

$$\begin{aligned}\boldsymbol{u}^{\beta}_{N,x_0}(s) &\coloneqq \big(u^{\beta}_{0,x_0}(s),...,u^{\beta}_{N,x_0}(s)\big)^T,\\ \boldsymbol{F}^{\beta}_{N,x_0}(s) &\coloneqq \big(f^{\beta}_{0,x_0}(s),...,f^{\beta}_{N,x_0}(s)\big)^T,\ f^{\beta}_{i,x_0} = \big(f(x,s),\hat{\mathcal{H}}^{\beta}_{i,x_0}(x)\big)\end{aligned} \tag{7.2.10}$$

are the vectors consisting of coefficients in the spectral expansion U^{β}_{N,x_0} and the coefficients of the spectral expansion of the RHS term f in Eq. (7.2.6). The matrix $\boldsymbol{A}^{\beta}_{N}$ is defined by

$$(\boldsymbol{A}^{\beta}_{N})_{ij} = a(\hat{\mathcal{H}}^{\beta}_{i,x_0},\hat{\mathcal{H}}^{\beta}_{j,x_0}) \tag{7.2.11}$$

where a is the bilinear operator in Eq. (7.2.1). The initial values $u^{\beta}_{i,x_0} \coloneqq \big(u(x,0),\hat{\mathcal{H}}^{\beta}_{i,x_0}(x)\big)$.

Our goal is to analyze the error $e(t) = \|U^{\beta}_{N,x_0}(x,t) - u(x,t)\|$, where u gives the solution to the model problem (Eqs. (7.2.1) and (7.2.2)) and U^{β}_{N,x_0} is the numerical solution of Eq. (7.2.6).

Theorem 4. Suppose u solves Eqs. (7.2.1) and (7.2.2) and U^{β}_{N,x_0} solves Eq. (7.2.6), then the error $e(t) = \|U^{\beta}_{N,x_0}(x,t) - u(x,t)\|$ can be bounded by

$$e(t) \le \sqrt{2(t+1)}\frac{b_{N,\beta}+B_0}{b_{N,\beta}} \inf_{z^{\beta}_{N,x_0}\in Y^{\beta}_{N,x_0}} \|u - z^{\beta}_{N,x_0}\|_{X(0,t)}, \tag{7.2.12}$$

where B_0 is a constant that depends on the bilinear operator $a(\cdot,\cdot)$ and $b_{N,\beta}$ is a constant that depends on $a(\cdot,\cdot)$, the scaling factor β, and the dimension of the space P^{β}_{N,x_0}.

Proof. For simplicity, we define the operator (denoting the LHS of Eq. (7.2.6))

$$B(u,\mathbf{v}^{\beta}_{N,x_0}) \coloneqq \int_0^t \Big[(\partial_s u, v^{\beta}_{N,x_0}) + a(u,v^{\beta}_{N,x_0})\Big]\mathrm{d}s + (u_0,\tilde{v}^{\beta}_{N,x_0}),\ \ u\in X(0,t), \mathbf{v}^{\beta}_{N,x_0}\in Y^{\beta}_{N,x_0}. \tag{7.2.13}$$

It can be proved that $B(u,\mathbf{v}^{\beta}_{N,x_0})$ is a continuous operator, *i.e.*, there exists a constant B_0 such that

$$B(u,\mathbf{v}^{\beta}_{N,x_0}) \le B_0\|u\|_{X(0,t)}\|\mathbf{v}^{\beta}_{N,x_0}\|_{Y^{\beta}_{N,x_0}}. \tag{7.2.14}$$

Furthermore, there exists a positive constant that depends on the dimension of the basis function space P^{β}_{N,x_0} as well as the scaling factor β denoted by $b_{N,\beta}$ such that

$$\inf_{0\le U^{\beta}_{N,x_0}\in X^{\beta}_{N,x_0}} \sup_{0\le \mathbf{v}^{\beta}_{N,x_0}\in X^{\beta}_{N,x_0}} \frac{B(U^{\beta}_{N,x_0},\mathbf{v}^{\beta}_{N,x_0})}{\|U^{\beta}_{N,x_0}\|_{X(0,t)}\|\mathbf{v}^{\beta}_{N,x_0}\|_{Y^{\beta}_{N,x_0}}} \ge b_{N,\beta}. \tag{7.2.15}$$

Actually, we can take

$$\mathbf{v}^{\beta}_{N,x_0} = (U^{\beta}_{N,x_0}(x,s) + \frac{c_0}{(2N\beta^2+1)(C_0+1)^2}\partial_s U^{\beta}_{N,x_0}(x,s),\, U^{\beta}_{N,x_0}(x,0)) \tag{7.2.16}$$

where c_0, C_0 are the constants in Eq. (7.2.3). Therefore, by substituting v as defined in Eq. (7.2.16) into Eq. (7.2.13), we find

$$\begin{aligned}
B(U^{\beta}_{N,x_0},\mathbf{v}^{\beta}_{N,x_0}) \ge& \frac{1}{2}\big(\|U^{\beta}_{N,x_0}(x,0)\|^2 + \|U^{\beta}_{N,x_0}(x,t)\|^2\big) \\
&+ c_0\int_0^t \left(\|U^{\beta}_{N,x_0}\|^2_{H^1} + \frac{1}{(2N\beta^2+1)(C_0+1)^2}\|\partial_s U^{\beta}_{N,x_0}\|^2\right)\mathrm{d}s \\
&- \frac{c_0}{2}\int_0^t \left(\|U^{\beta}_{N,x_0}\|^2_{H^1} + \frac{C_0^2}{(2N\beta^2+1)^2(C_0+1)^4}\|\partial_s U^{\beta}_{N,x_0}\|^2_{H^1}\right)\mathrm{d}s \\
\ge& \frac{1}{2}\big(\|U^{\beta}_{N,x_0}(x,0)\|^2 + \|U^{\beta}_{N,x_0}(x,t)\|^2\big) + c_0\int_0^t \|U^{\beta}_{N,x_0}\|^2_{H^1}\mathrm{d}s \\
&- \frac{c_0}{2}\int_0^t \|U^{\beta}_{N,x_0}\|^2_{H^1}\mathrm{d}s + \frac{c_0}{(2N\beta^2+1)^2(C_0+1)^2}\int_0^t \|\partial_s U^{\beta}_{N,x_0}\|^2_{H^1}\mathrm{d}s \\
&- \frac{c_0}{2(2N\beta^2+1)^2(C_0+1)^2}\int_0^t \|\partial_s U^{\beta}_{N,x_0}\|^2_{H^1}\mathrm{d}s \\
\ge& \min\left\{\frac{1}{2}, \frac{c_0}{2}, \frac{c_0}{2(2N\beta^2+1)^2(C_0+1)^2}\right\}\|U^{\beta}_{N,x_0}\|^2_{X(0,t)} \\
\ge& \min\left\{\frac{1}{4}, \frac{c}{4}, \frac{c_0}{2(2N\beta^2+1)^2(C_0+1)^2}\right\}\|U^{\beta}_{N,x_0}\|_{X(0,t)}\ \|\mathbf{v}^{\beta}_{N,x_0}\|_{Y^{\beta}_{N,x_0}},
\end{aligned} \tag{7.2.17}$$

where in the second inequality we have used the inverse inequality of generalized Hermite functions [STW11] that states

$$\|\partial_s U^{\beta}_{N,x_0}\|^2_{H^1} \le (2N\beta^2+1)\|\partial_s U^{\beta}_{N,x_0}\|^2. \tag{7.2.18}$$

Here, $b_{N,\beta} \coloneqq \min\{\frac{1}{4}, \frac{c_0}{4}, \frac{c_0}{2(2N\beta^2+1)^2(C_0+1)^2}\}$ is the constant that satisfies Eq. (7.2.15).

For any $\mathbf{v}^\beta_{N,x_0} \in Y^\beta_{N,x_0}$, if U^β_{N,x_0} solves Eq. (7.2.6) and u solves Eqs. (7.2.1) and (7.2.2),

$$B(U^\beta_{N,x_0}, \mathbf{v}^\beta_{N,x_0}) = B(u, \mathbf{v}^\beta_{N,x_0}) = \int_0^t \big(f(x,s), v^\beta_{N,x_0}(x,s)\big)\mathrm{d}s + (u_0(x), \tilde{v}^\beta_{N,x_0}(x)). \tag{7.2.19}$$

By combining Eqs. (7.2.15) and (7.2.19), we find

$$\|U^\beta_{N,x_0}\|_{X(0,t)} \le \frac{1}{b_{N,\beta}} \sup_{\mathbf{v}^\beta_{N,x_0}} \frac{B(U^\beta_{N,x_0}, \mathbf{v}^\beta_{N,x_0})}{\|\mathbf{v}^\beta_{N,x_0}\|_{Y^\beta_{N,x_0}}} = \sup_{\mathbf{v}^\beta_{N,x_0}} \frac{1}{b_{N,\beta}} \frac{B(u, \mathbf{v}^\beta_{N,x_0})}{\|\mathbf{v}^\beta_{N,x_0}\|_{Y^\beta_{N,x_0}}} \le \frac{B_0}{b_{N,\beta}} \|u\|_{X(0,t)}. \tag{7.2.20}$$

Finally, by the triangular inequality, we can conclude that the approximation error is bounded:

$$\begin{aligned} \|u - U^\beta_{N,x_0}\|_{X(0,t)} &\le \inf_{z^\beta_{N,x_0} \in Y^\beta_{N,x_0}} (\|u - z^\beta_{N,x_0}\|_{X(0,t)} + \|U^\beta_{N,x_0} - z^\beta_{N,x_0}\|_{X(0,t)}) \\ &\le \frac{b_{N,\beta} + B_0}{b_{N,\beta}} \inf_{z^\beta_{N,x_0} \in Y^\beta_{N,x_0}} \|u - z^\beta_{N,x_0}\|_{X(0,t)}. \end{aligned} \tag{7.2.21}$$

Notice that the L^2-error $e(t) = \|u(x,t) - U^\beta_{N,x_0}(x,t)\|$ at time t can be bounded by $2(t+1)\|u - U^\beta_{N,x_0}\|_{X(0,t)}$ using the triangular inequality and the Hölder inequality

$$\begin{aligned} \|u(x,t) - U^\beta_{N,x_0}(x,t)\|^2 &\le \|u(x,0) - U^\beta_{N,x_0}(x,0) + \int_0^t \partial_s\big(u(x,s) - U^\beta_{N,x_0}(x,s)\big)\mathrm{d}s\|^2 \\ &\le 2(t+1)\int_0^t \|\partial_s\big(u(x,s) - U^\beta_{N,x_0}(x,s)\big)\|^2 \mathrm{d}s \\ &\quad + 2\|u(x,0) - U^\beta_{N,x_0}(x,0)\|^2 \\ &\le 2(t+1)\|u(x,t) - U^\beta_{N,x_0}\|^2_{X(0,t)}. \end{aligned} \tag{7.2.22}$$

Therefore, Eq. (7.2.12) holds. □

We can also use generalized Hermite functions to numerically solve the D-dimensional model problem Eq. (7.2.6),

$$\begin{aligned}&\int_0^t \Big[\big(\partial_s U^{\beta}_{\boldsymbol{N},\boldsymbol{x}_0}(\boldsymbol{x},s), v^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x}_0}(\boldsymbol{x},s)\big) + a\big(U^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x}_0}(\boldsymbol{x},s), v^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x}_0}(\boldsymbol{x},s)\big)\Big]\,\mathrm{d}s \\ &+ \big(U^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x}_0}(\boldsymbol{x},0), \tilde{v}^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x}_0}(\boldsymbol{x})\big) = \int_0^t \big(f(\boldsymbol{x},s), v^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x}_0}(\boldsymbol{x},s)\big)\mathrm{d}s + \big(u(\boldsymbol{x},0), \tilde{v}^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x}_0}(\boldsymbol{x})\big),\end{aligned} \tag{7.2.23}$$

where

$$\boldsymbol{\beta} \coloneqq (\beta^1,\ldots,\beta^D),\ \ \boldsymbol{x}_0 \coloneqq (x_0^1,\ldots,x_0^D),\ \ \boldsymbol{N} \coloneqq (N^1,\ldots,N^D) \tag{7.2.24}$$

are the D-dimensional scaling factors, displacements, and expansion orders and

$$U^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x}_0}(\boldsymbol{x},s), v^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x}_0}(\boldsymbol{x},s) \in L^2\big(0,t;\bigotimes_{h=1}^{D} P^{\beta^h}_{N^h,x_0^h}(\mathbb{R})\big), \quad \tilde{v}^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x}_0} \in \bigotimes_{h=1}^{D} P^{\beta^h}_{N^h,x_0^h}(\mathbb{R}). \tag{7.2.25}$$

A multiple dimension version of the error bound Eq. (7.2.12) can be similarly derived

$$\|u(\boldsymbol{x},t) - U^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x}_0}(\boldsymbol{x},t)\| \le \sqrt{2(t+1)}\frac{b_{\boldsymbol{N},\boldsymbol{\beta}} + B_0}{b_{\boldsymbol{N},\boldsymbol{\beta}}} \inf_{z^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x}_0} \in Y^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x}_0}} \|u - z^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x}_0}\|_{X(0,t)}, \tag{7.2.26}$$

where $b_{\boldsymbol{N},\boldsymbol{\beta}} \coloneqq \min\{\frac{1}{4}, \frac{c_0}{4}, \frac{c_0}{2(\sum_{h=1}^{D} 2N_i\beta_i^2+1)(C_0+1)^2}\}$. The function spaces are

$$\begin{aligned}&X(0,t) \coloneqq \Big\{u : u(\boldsymbol{x},s) \in L^2\left(0,t;H^1(\mathbb{R}^D)\right), \partial_s u(\boldsymbol{x},s) \in L^2\left(0,t;H^1(\mathbb{R}^D)\right)\Big\}. \\ &Y^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x}_0} \coloneqq L^2\big(0,t;\bigotimes_{h=1}^{D} P^{\beta^h}_{N^h,x_0^h}(\mathbb{R})\big) \times \bigotimes_{h=1}^{D} P^{\beta^h}_{N^h,x_0^h}(\mathbb{R}).\end{aligned} \tag{7.2.27}$$

7.3 Errors of adaptive techniques

In this section, we analyze the errors directly associated with the moving, scaling, and p-adaptive techniques that automatically change the shape, the translation, and the order of the numerical solution through adjustment of β, x_0, and N, respectively [XSC21b, XSC21a]. We derive the error bound when solving Eq. (7.2.6) and prove Theorem 3 presented in Introduction. Doing so explicitly shows how changing β, x_0, and N affects the error, thus

providing insight into how to choose parameters in the adaptive algorithm that leads to the proper tuning of β, x_0, and N.

Instead of using collocation methods to carry out the scaling, moving, or p-adaptive methods as was done in previous work [XSC21b, XSC21a] (*i.e.*, enforcing the updated numerical solution to be the same as the original numerical solution on the new collocation points), we now use the Galerkin method (*i.e.*, projecting the numerical solution onto the space of adjusted basis functions). For example, given the numerical solution $U^{\beta}_{N,x_0}(x,t)$ at time t, if we change its scaling factor from β to $\tilde{\beta}$, previous implementation in [XSC21b, XSC21a] replaces $U^{\beta}_{N,x_0}(x,t)$ with $\mathcal{I}^{\tilde{\beta}}_{N,x_0}U^{\beta}_{N,x_0} \in P^{\tilde{\beta}}_{N,x_0}$ as the new numerical solution. This new numerical solution $\mathcal{I}^{\tilde{\beta}}_{N,x_0}U^{\beta}_{N,x_0}$ takes on the same values as U^{β}_{N,x_0} at the collocation points for the new basis functions $\{\hat{\mathcal{H}}^{\tilde{\beta}}_{i,x_0}\}_{i=0}^{N}$. Therefore, the error after changing β to $\tilde{\beta}$ and replacing $U^{\beta}_{N,x_0}(x,t)$ with $\mathcal{I}^{\tilde{\beta}}_{N,x_0}U^{\beta}_{N,x_0}$ can be bounded by

$$\|u - \mathcal{I}^{\tilde{\beta}}_{N,x_0}U^{\beta}_{N,x_0}\| \leq \|u - U^{\beta}_{N,x_0}\| + \|(I - \mathcal{I}^{\tilde{\beta}}_{N,x_0})U^{\beta}_{N,x_0}\|. \tag{7.3.1}$$

In this chapter, we project the numerical solution onto $P^{\tilde{\beta}}_{N,x_0} \coloneqq \{\hat{\mathcal{H}}^{\tilde{\beta}}_{i,x_0}\}_{i=0}^{N}$, *i.e.*, using $\pi^{\tilde{\beta}}_{N,x_0}U^{\beta}_{N,x_0}$ as the new numerical solution. Therefore, the error bound after changing the scaling factor is

$$\|u - \pi^{\tilde{\beta}}_{N,x_0}U^{\beta}_{N,x_0}\| \leq \|u - U^{\beta}_{N,x_0}\| + \|(I - \pi^{\tilde{\beta}}_{N,x_0})U^{\beta}_{N,x_0}\|. \tag{7.3.2}$$

The second term on the RHSs of Eqs. (7.3.1) and (7.3.2) can be viewed as an additional error bound resulting from changing the scaling factor. Furthermore, we can show

$$\|(I - \mathcal{I}^{\tilde{\beta}}_{N,x_0})U^{\beta}_{N,x_0}\| \geq \|(I - \pi^{\tilde{\beta}}_{N,x_0})U^{\beta}_{N,x_0}\|. \tag{7.3.3}$$

The proof is straightforward. Assuming the spectral expansion of U^{β}_{N,x_0} under the new basis

functions $\{\hat{\mathcal{H}}^{\beta}_{i,x_0}\}$ is

$$U^{\beta}_{N,x_0} = \sum_{i=0}^{\infty} u^{\beta}_{i,x_0} \hat{\mathcal{H}}^{\beta}_{i,x_0}. \tag{7.3.4}$$

By definition,

$$\pi^{\tilde{\beta}}_{N,x_0} U^{\beta}_{N,x_0} = \sum_{i=0}^{N} u^{\beta}_{i,x_0} \hat{\mathcal{H}}^{\beta}_{i,x_0}, \quad \mathcal{I}^{\tilde{\beta}}_{N,x_0} U^{\beta}_{N,x_0} = \sum_{i=0}^{N} \tilde{u}^{\beta}_{i,x_0} \hat{\mathcal{H}}^{\beta}_{i,x_0}. \tag{7.3.5}$$

Therefore,

$$\begin{aligned} \|(I - \mathcal{I}^{\tilde{\beta}}_{N,x_0}) U^{\beta}_{N,x_0}\| &= \left[\sum_{i=0}^{N} (\tilde{u}^{\beta}_{i,x_0} - u^{\beta}_{i,x_0})^2 \|\hat{\mathcal{H}}^{\beta}_{i,x_0}\|^2 + \sum_{i=N+1}^{\infty} (u^{\beta}_{i,x_0})^2 \|\hat{\mathcal{H}}^{\beta}_{i,x_0}\|^2 \right]^{\frac{1}{2}} \\ &\geq \left[\sum_{i=N+1}^{\infty} (u^{\beta}_{i,x_0})^2 \|\hat{\mathcal{H}}^{\beta}_{i,x_0}\|^2 \right]^{\frac{1}{2}} = \|(I - \pi^{\tilde{\beta}}_{N,x_0}) U^{\beta}_{N,x_0}\|. \end{aligned} \tag{7.3.6}$$

With Eq. (7.3.3), using the projected $\pi^{\tilde{\beta}}_{N,x_0} U^{\beta}_{N,x_0}$ as the new numerical solution instead of the interpolated $\mathcal{I}^{\tilde{\beta}}_{N,x_0} U^{\beta}_{N,x_0}$ might lead to a smaller error bound.

7.3.1 Posterior error estimate

We derive the posterior error estimates that depend on the numerical solution $U^{\beta}_{N,x_0} \in P^{\beta}_{N,x_0}$ and on how β, x_0, and N are changed. Combining the error estimate of the adaptive techniques with Theorem 4, the error estimate for numerically solving Eqs. (7.2.1) and (7.2.2), our ultimate goal is to prove Theorem 3, the error estimate for adaptive spectral methods.

First, we derive the error bound after changing the scaling factor β. Suppose at time t, we change β to $\tilde{\beta}$ and replace the numerical solution U^{β}_{N,x_0} with $\pi^{\tilde{\beta}}_{N,x_0} U^{\beta}_{N,x_0} \in P^{\tilde{\beta}}_{N,x_0}$, the error is

$$\|u - \pi^{\tilde{\beta}}_{N,x_0} U^{\beta}_{N,x_0}\| \leq \|u - U^{\beta}_{N,x_0}\| + \|(I - \pi^{\tilde{\beta}}_{N,x_0}) U^{\beta}_{N,x_0}\| \tag{7.3.7}$$

where the first term on the RHS is the error before scaling and the second term on the RHS

is the additional error bound from changing the scaling factor ("scaling error"). Denoting $\beta' = \tilde{\beta}/\beta$, we can further bound the scaling error by

$$\begin{aligned}
\|(I - \pi^{\tilde{\beta}}_{N,x_0})U^{\beta}_{N,x_0}\| &\leq \|U^{\beta}_{N,x_0}(x,t) - U^{\beta}_{N,x_0}(\beta' x, t)\| \\
&= \left[\int_{\mathbb{R}} \left(\int_{\beta' x}^{x} \partial_y U^{\beta}_{N,x_0}(y,t)\mathrm{d}y\right)^2 \mathrm{d}x\right]^{\frac{1}{2}} \\
&\leq \left[\int_{\mathbb{R}} |1-\beta'| x \left(\int_{\beta' x}^{x} \left(\partial_y U^{\beta}_{N,x_0}(y,t)\right)^2 \mathrm{d}y\right) \mathrm{d}x\right]^{\frac{1}{2}} \\
&= \frac{|1-\beta'|\sqrt{1+\beta'}}{\sqrt{2}\beta'} \|x\partial_x U^{\beta}_{N,x_0}(x)\|.
\end{aligned} \tag{7.3.8}$$

Therefore, the error after changing the scaling factor from β to $\tilde{\beta}$ is bounded by

$$\|u - \pi^{\tilde{\beta}}_{N,x_0} U^{\beta}_{N,x_0}\| \leq \|u - U^{\beta}_{N,x_0}\| + \frac{|1-\beta'|\sqrt{1+\beta'}}{\sqrt{2}\beta'} \|x\partial_x U^{\beta}_{N,x_0}(x)\|. \tag{7.3.9}$$

From Eq. (7.3.9), the second term in the last equality is the additional error bound resulting from scaling. The factor $\frac{|1-\beta'|\sqrt{1+\beta'}}{\sqrt{2}\beta'}$ is directly related to how much the scaling factor is changed while $\|x\partial_x U^{\beta}_{N,x_0}(x)\|$ depends on the spatial derivative of the pre-scaled solution.

Next, we derive the error bound associated with changing the displacement x_0. Given the numerical solution U^{β}_{N,x_0}, if we change the displacement of the basis functions from x_0 to $\tilde{x}_0$ and set $\pi^{\beta}_{N,\tilde{x}_0} U^{\beta}_{N,\tilde{x}_0} \in P^{\beta}_{N,\tilde{x}_0}$ as the new numerical solution, the error is

$$\|u - \pi^{\beta}_{N,\tilde{x}_0} U^{\beta}_{N,\tilde{x}_0}\| \leq \|u - U^{\beta}_{N,x_0}\| + \|(I - \pi^{\beta}_{N,\tilde{x}_0})U^{\beta}_{N,x_0}\|, \tag{7.3.10}$$

where the second term on the RHS is the additional error bound from changing x_0 ("moving error"). Furthermore, it is bounded by

$$\begin{aligned}\|(\pi^{\beta}_{N,\tilde{x}_0} - I)U^{\beta}_{N,x_0}(x,t)\| &\leq \|U^{\beta}_{N,x_0}(x,t) - U^{\beta}_{N,x_0}(x-\tilde{x}_0+x_0,t)\| \\ &\leq \left[\int_{\mathbb{R}} |x_0-\tilde{x}_0| \left(\int_{x-\tilde{x}_0+x_0}^{x} \left(\partial_y U^{\beta}_{N,x_0}(y,t)\right)^2 \mathrm{d}y\right) \mathrm{d}x\right]^{\frac{1}{2}} \\ &= d\|\partial_x U^{\beta}_{N,x_0}(x)\|,\end{aligned} \tag{7.3.11}$$

where $d \coloneqq |x_0 - \tilde{x}_0|$. Therefore, the error after changing the displacement from x_0 to $\tilde{x}_0$ is bounded by

$$\|u - \pi^{\beta}_{N,x_0} U^{\beta}_{N,\tilde{x}_0}\| \leq \|u - U^{\beta}_{N,x_0}\| + d\|\partial_x U^{\beta}_{N,x_0}(x)\|. \tag{7.3.12}$$

We see that the additional error bound associated with moving depends on the change in the displacement x_0 and the spatial derivative $\partial_x U^{\beta}_{N,x_0}(x)$ of the pre-translated numerical solution.

Finally, we analyze the error associated with the p-adaptive technique. When projecting the numerical solution U^{β}_{N,x_0} onto the new space $P^{\beta}_{\tilde{N},x_0}$, no extra error will be introduced when $\tilde{N} > N$ (refinement) because the basis functions $\{\hat{\mathcal{H}}^{\beta}_{i,x_0}\}_{i=0}^{\tilde{N}}$ form an orthogonal set of basis functions and $\pi^{\beta}_{\tilde{N},x_0} U^{\beta}_{N,x_0} = U^{\beta}_{N,x_0}$, *i.e.*,

$$\|u - \pi^{\beta}_{\tilde{N},x_0} U^{\beta}_{N,x_0}\| = \|u - U^{\beta}_{N,x_0}\|, \;\; \tilde{N} > N. \tag{7.3.13}$$

When we reduce the number of basis functions from N to $\tilde{N} < N$ (coarsening), we use $\pi^{\beta}_{\tilde{N},x_0} U^{\beta}_{N,x_0} = U^{\beta}_{\tilde{N},x_0}$ as the new numerical solution. $\pi^{\beta}_{\tilde{N},x_0} U^{\beta}_{N,x_0}$ leaves out the last $N-\tilde{N}$ terms in the spectral expansion of U^{β}_{N,x_0}. Therefore, the error after coarsening can be bounded by

$$\|u - \pi^{\beta}_{\tilde{N},x_0} U^{\beta}_{N,x_0}\| \leq \|u - U^{\beta}_{N,x_0}\| + \|(I - \pi^{\beta}_{\tilde{N},x_0}) U^{\beta}_{N,x_0}\|, \;\; \tilde{N} < N. \tag{7.3.14}$$

In Eq. (7.3.14), the second term in the last inequality is the additional error bound that results from truncating the spectral expansion and leaving out the last $N - \tilde{N}$ terms.

Next, we generalize Theorem 4 to forward time from t_0 to t_1 given $U^{\beta}_{N,x_0}(x,t_0)$. We assume that no adaptive technique is activated within $t \in (t_0,t_1)$ and denote $e(x,t) = u(x,t) - U^{\beta}_{N,x_0}(x,t), t \in [t_0,t_1]$, where u is the solution to Eqs. (7.2.1) and (7.2.2). The error at t_1, $e(x,t_1) = u(x,t_1) - U^{\beta}_{N,x_0}(x,t_1)$, can be decomposed as $e(x,t_1) = e_1(x,t_1) + e_2(x,t_1)$ where $e_1(x,t_1)$ is the error $u(x,t_1) - \tilde{U}^{\beta}_{N,x_0}(x,t_1)$ with $\tilde{U}^{\beta}_{N,x_0}$ solving Eq. (7.2.6) with initial condition $u(x,t_0)$. The second error term $e_2(x,t_1) \in L^2(t_0,t_1;P^{\beta}_{N,x_0})$ satisfies

$$\begin{aligned}\int_{t_0}^{t_1} &(\partial_s e_2(x,s), v(x,s)) + a\left(e_2(x,s), v(x,s)\right) \mathrm{d}s + (e_2(x,t_0), \tilde{v}(x,t_0)) \\ &= (e(x,t_0), \tilde{v}(x,t_0)), \ \ \forall v \in L^2(t_0,t_1;P^{\beta}_{N,x_0}),\ \tilde{v} \in P^{\beta}_{N,x_0}.\end{aligned} \tag{7.3.15}$$

From Theorem 4,

$$\|e_1(x,t_1)\| \leq \frac{b_{N,\beta}+B_0}{b_{N,\beta}} \sqrt{2(1+(t_1-t_0))}\, \|(I-\pi^{\beta}_{N,x_0})u\|_{X(t_0,t_1)}. \tag{7.3.16}$$

Additionally, since the bilinear form $a(\cdot,\cdot)$ is positive definite, substituting $v(x,t) = e_2(x,t)$ and $\tilde{v} = e(x,t_0)$ into Eq. (7.3.15), we conclude that $\|e_2(x,t_1)\| \leq \|e(x,t_0)\| = e(t_0)$. Therefore,

$$e(t_1) \leq e(t_0) + \frac{b_{N,\beta}+B_0}{b_{N,\beta}} \sqrt{2(1+(t_{i+1}-t_i))}\, \|(I-\pi^{\beta}_{N,x_0})u\|_{X(t_i,t_{i+1})}. \tag{7.3.17}$$

Specifically, this error bound does not depend on the step size $\Delta t = t_{i+1} - t_i$ if we use

$$\boldsymbol{u}^{\beta}_{N,x_0}(t+\Delta t) = e^{-\boldsymbol{A}^{\beta}_N \Delta t} \boldsymbol{u}^{\beta}_{N,x_0}(t) + e^{-\boldsymbol{A}^{\beta}_N \Delta t} \int_t^{t+\Delta t} e^{\boldsymbol{A}^{\beta}_N (s-t)} \boldsymbol{F}_{N,x_0}(s) \mathrm{d}s, \tag{7.3.18}$$

with $\boldsymbol{u}^{\beta}_{N,x_0}$, $\boldsymbol{F}^{\beta}_{N,x_0}$ defined by Eq. (7.2.10) and $\boldsymbol{A}^{\beta}_N$ defined by Eq. (7.2.11). Now, we are ready to prove Theorem 3, the overall error bound using the adaptive spectral methods. We define the times of the ℓ^{th} scaling, the ℓ^{th} translation, and the ℓ^{th} changing the expansion order to be t^s_ℓ, t^m_ℓ, and t^c_ℓ, respectively. We denote the scaling factors right before the ℓ^{th} scaling, moving, and changing the expansion order to be $\beta^s_\ell, \beta^m_\ell$, and β^c_ℓ, the displacements right before the ℓ^{th} scaling, moving, and changing the expansion order to be $x_{0\ell}^s, x_{0\ell}^m$, and $x_{0\ell}^c$, and the expansion

orders right before the ℓ^{th} scaling, moving, and changing the expansion order to be N_ℓ^s, N_ℓ^m, and N_ℓ^c, respectively. After the ℓ^{th} scaling, we denote the new scaling factor to be $\tilde{\beta}_\ell^s$ and the ratio $\beta'^s_\ell \coloneqq \tilde{\beta}_\ell^s/\beta_\ell^s$; after the ℓ^{th} moving, we denote the new displacement to be $\tilde{x_0}_\ell^m$ and $d_\ell^m \coloneqq |\tilde{x_0}_\ell^m - {x_0}_\ell^m|$; after the ℓ^{th} change of the expansion order, we denote the new expansion order as $\tilde{N}_\ell^c$.

The times at which the scaling factor or the displacement of the basis functions is changed, or the expansion order is reduced, are indicated by t_i in chronological order $0 = t_0 \leq t_1 ... \leq t_i \leq t_{K^s+K^m+K^c+1} = T$, where K^s, K^m, and K^c are the total number of scalings, translations, and changing the expansion order within $t \in [0, T]$. Specifically, if $t_i = t_{i+1}$, then more than one adaptation is triggered simultaneously. The corresponding constant that satisfies the inequality Eq. (7.2.12) during $[t_i, t_{i+1}]$ is denoted as $(b_{N_i,\beta_i} + B_0)/b_{N_i,\beta_i}$. From the error estimates of the scaling, moving, and p-adaptive techniques in Eqs. (7.3.9), (7.3.11), (7.3.14), and Eq. (7.3.17), we conclude

$$
\begin{aligned}
e(T) \leq & \sum_{i=0}^{K^s+K^m+K^c} \frac{b_{N_i,\beta_i}+B_0}{b_{N_i,\beta_i}} \sqrt{2(1+(t_{i+1}-t_i))}\, \|(I-\pi^{\beta_i}_{N_i,x_{0i}})u\|_{X(t_i,t_{i+1})} \\
& + \sum_{\ell=1}^{K^s} \frac{|1-\beta'^s_\ell|\sqrt{1+\beta'^s_\ell}}{\sqrt{2}\beta'^s_\ell}\, \|x\partial_x U^{\beta^s_\ell}_{N^s_\ell,x_{0^s_\ell}}(x,t^s_\ell)\| \\
& + \sum_{\ell=1}^{K^m} d^m_\ell \|\partial_x U^{\beta^m_\ell}_{N^m_\ell,x_{0^m_\ell}}(x,t^m_\ell)\| \\
& + \sum_{r=1}^{K^c} \|(I-\pi^{\beta^c_\ell}_{\tilde{N}^c_\ell,x_{0^c_\ell}})U^{\beta^c_\ell}_{N^c_\ell,x_{0^c_\ell}}\| \\
\leq & \sum_{i=0}^{K^s+K^m+K^c} \frac{b_{N_i,\beta_i}+B_0}{b_{N_i,\beta_i}} \sqrt{2(1+(t_{i+1}-t_i))}\, \|(I-\pi^{\beta_i}_{N_i,x_{0i}})u\|_{X(t_i,t_{i+1})} \\
& + \sum_{\ell=1}^{K^s} \frac{|1-\beta'^s_\ell|\sqrt{1+\beta'^s_\ell}}{\sqrt{2}\beta'^s_\ell}(2N^s_\ell+1)\, \|U^{\beta^s_\ell}_{N^s_\ell,x_{0^s_\ell}}(x,t^s_\ell)\| \\
& + \sum_{\ell=1}^{K^m} \sqrt{(2N^m_\ell+1)}\beta^m_\ell d^m_\ell\, \|U^{\beta^m_\ell}_{N^m_\ell,x_{0^m_\ell}}(x,t^m_\ell)\| \\
& + \sum_{\ell=1}^{K^c} \|(I-\pi^{\beta^c_\ell}_{\tilde{N}^c_\ell,x_{0^c_\ell}})U^{\beta^c_\ell}_{N^c_\ell,x_{0^c_\ell}}\|
\end{aligned}
\tag{7.3.19}
$$

where we have used the three-term recurrence relation for generalized Hermite functions and the inverse inequality Eq. (7.2.18) to bound $\|x\partial_x U^{\beta^s_\ell}_{N^s_\ell,x_{0^s_\ell}}(x,t^s_\ell)\|$ and $\|\partial_x U^{\beta^m_\ell}_{N^m_\ell,x_{0^m_\ell}}(x,t^m_\ell)\|$ in the second inequality. Note that in the first term of Eq. (7.3.19), if $t_i = t_{i+1}$ then we define $\|(I-\pi^{\beta_i}_{N_i,x_{0i}})u\|_{X(t_i,t_{i+1})} \coloneqq 0$. The first term on the RHS of the last inequality corresponds to e_0 in Theorem 3, and the second, third, and last terms on the RHS of last inequality correspond to $e_{\mathrm{S}}, e_{\mathrm{M}}$, and e_{C}, respectively.

From Eq. (7.3.19), the errors caused by scaling and moving (the second and third terms of the equation) suggest that the smaller the adjustment in β or x_0, the smaller the factors $|1-\beta'^s_\ell|$ and d^m_ℓ in the scaling or moving errors. Therefore, we should set the triggering parameters $q \lesssim 1$ ($\lesssim$ means smaller but close to) and $0 \lesssim \delta$ in Table 7.1 so that the scaling factor β and the displacement x_0 can be tuned more accurately without over-adjustment that may lead to larger errors.

When coarsening, decreasing the expansion order N too much will increase the coarsening error through the last term in Eq. (7.3.19). Increasing the coarsening threshold η_0 to make it harder to decrease N can preserve accuracy but possibly at the expense of keeping a higher computational burden. Note that although the effect of refinement does not explicitly reveal itself in the error bound Eq. (7.3.19), both a smaller initial refinement threshold η and a smaller γ (the ratio of increasing the refinement threshold) could lead to larger N and thus smaller errors (the first term of the second equation in Eq. (7.3.19)).

However, if N increases, so will the computational cost. Using the numerical example presented in the next section, we will discuss how to set γ and η so that high accuracy can be achieved without significant degradation of computational efficiency. Since the adaptive techniques do not require prior information on the solution, the last three terms in Eq. (7.3.19), *i.e.*, errors from adaptive techniques, depend only on the latest numerical solution itself.

Note that the numerical error in solving Eqs. (7.2.1) and (7.2.2) is no less than the projection error

$$e(T) = \|u(x,t) - U^{\beta}_{N,x_0}(x,t)\| \geq \|(I - \pi^{\beta}_{N,x_0})u(x,t)\|, \tag{7.3.20}$$

and it has also been shown that improper scaling of generalized Hermite functions can lead to large projection errors [Tan93]. Furthermore, in Examples 2, 3, 5 in [XSC21b] and Example 2 in [XSC21a], improper displacement x_0 or a too-small expansion order N will also lead to projection errors, implying a large $e(T)$. Therefore, timely and accurate implementation of the adaptive techniques is important for controlling the lower error bound (the projection error) Eq. (7.3.20). Consequently, to adjust them properly, we need to set $1 \lesssim \nu$ and $1 \lesssim \mu$ in the scaling and moving technique algorithms, respectively.

A D-dimensional generalization of Eq. (7.3.19) for spatial variables $\boldsymbol{x} = (x_1, ..., x_D) \in \mathbb{R}^D$ can be similarly derived for the numerical solution

$$U^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x_0}}(\boldsymbol{x},t) \coloneqq \sum_{i_1=0}^{N^1} ... \sum_{i_D=0}^{N^D} u^{\boldsymbol{\beta}}_{i^1,...,i^D,\boldsymbol{x}_0}(t) \Pi_{h=1}^{D} \hat{\mathcal{H}}^{\beta^h}_{i^h,x_0^h}(x) \tag{7.3.21}$$

as

$$
\begin{aligned}
e(T) =& \|u(\boldsymbol{x},t) - U^{\boldsymbol{\beta}}_{\boldsymbol{N},\boldsymbol{x_0}}(\boldsymbol{x},t)\| \\
\leq & \sum_{i=0}^{\boldsymbol{K}^s+\boldsymbol{K}^m+\boldsymbol{K}^c} \frac{b_{\boldsymbol{N}_i,\boldsymbol{\beta}_i}+B_0}{b_{\boldsymbol{N}_i,\boldsymbol{\beta}_i}}\sqrt{2(1+(t_{i+1}-t_i))}\|(I-\pi^{\boldsymbol{\beta}_i}_{\boldsymbol{N}_i,\boldsymbol{x}_{0i}})u\|_{X(t_i,t_{i+1})} \\
& + \sum_{h=1}^{D}\sum_{\ell=1}^{K^{h,s}} \frac{|1-{\beta'}_\ell^{h,s}|\sqrt{1+{\beta'}_\ell^{h,s}}}{\sqrt{2}{\beta'}_\ell^{h,s}}(2N_\ell^{h,s}+1)\|U^{\boldsymbol{\beta}_\ell^{h,s}}_{\boldsymbol{N}_\ell^{h,s},{\boldsymbol{x_0}_\ell}^{h,s}}(\boldsymbol{x},t_\ell^{h,s})\| \\
& + \sum_{h=1}^{D}\sum_{\ell=1}^{K^{h,m}} \sqrt{2N_\ell^{h,m}+1}\beta_\ell^{h,m}d_\ell^{h,m}\|U^{\boldsymbol{\beta}_\ell^{h,m}}_{\boldsymbol{N}_\ell^{h,m},{\boldsymbol{x_0}_\ell}^{h,m}}(\boldsymbol{x},t_\ell^{h,m})\| \\
& + \sum_{h=1}^{D}\sum_{r=1}^{K^{h,c}} \|(I-\pi^{\boldsymbol{\beta}_r^{h,c}}_{\tilde{\boldsymbol{N}}_r^{h,c},{\boldsymbol{x_0}_\ell}^{h,c}})U^{\boldsymbol{\beta}_r^{h,c}}_{{\boldsymbol{N}_r}^{h,c},{\boldsymbol{x_0}_\ell}^{h,c}}(\boldsymbol{x},t_\ell^{h,c})\|
\end{aligned}
\tag{7.3.22}
$$

where $\boldsymbol{\beta}, \boldsymbol{x_0}$, and $\boldsymbol{N}$ are the corresponding D-dimensional scaling factor, displacement, and expansion order defined in Eq. (7.2.24). $\boldsymbol{K}^s = \sum_{h=1}^{D} K^{h,s}, \boldsymbol{K}^m = \sum_{h=1}^{D} K^{h,m}, \boldsymbol{K}^c = \sum_{h=1}^{D} K^{h,c}$ are the total number of times of performing scaling, moving, and changing the expansion orders, across all dimensions ($K^{h,s}, K^{h,m}, K^{h,c}$ are the numbers of using the scaling, moving, or p-adaptive technique in the h^{th} dimension, respectively), the constant $(b_{\boldsymbol{N}_i,\boldsymbol{\beta}_i} + B_0)/b_{\boldsymbol{N}_i,\boldsymbol{\beta}_i}$ is the RHS constant in the inequality (7.2.26) during $[t_i, t_{i+1}]$, and $t_\ell^{h,s}, t_\ell^{h,m}, t_\ell^{h,c}$ are the times of the ℓ^{th} scaling, moving, or changing the expansion order in the h^{th} dimension, respectively. The second, third and last terms in Eq. (7.3.22) describe scaling error bounds in all dimensions, moving error bounds in all dimensions, and coarsening error bounds in all dimensions.

In Eq. (7.3.22), $\boldsymbol{\beta}_\ell^{h,s} \coloneqq (\beta_\ell^{1,s}, ..., \beta_\ell^{D,s}), \boldsymbol{\beta}_\ell^{h,m}$, and $\boldsymbol{\beta}_\ell^{h,c}$ are the D-dimensional scaling factors right before the ℓ^{th} scaling, moving, or changing the expansion order in the h^{th} dimension. Similarly, ${\boldsymbol{x_0}_\ell}^{h,s} \coloneqq ({x_0}_\ell^{1,s}, ..., {x_0}_\ell^{D,s}), {\boldsymbol{x_0}_\ell}^{h,m}$, and ${\boldsymbol{x_0}_\ell}^{h,c}$ are the D-dimensional displacements right before the ℓ^{th} scaling, moving, or change of expansion order in the h^{th} dimension, and $\boldsymbol{N}_\ell^{h,s} \coloneqq (N_\ell^{1,s}, ..., N_\ell^{D,s}), \boldsymbol{N}_\ell^{h,m}$, and $\boldsymbol{N}_\ell^{h,c}$ are the D-dimensional expansion orders right before the ℓ^{th} scaling, moving, or change of expansion order in the h^{th} dimension. ${\beta'}_\ell^{h,s}$ is the ratio $\tilde{\beta}_\ell^{h,s}/\beta_\ell^{h,s}$ where $\tilde{\beta}_\ell^{h,s}$ is the scaling factor after the ℓ^{th} scaling in the h^{th} dimension,

$d_\ell^{h,m} := |\tilde{x_0}_\ell^{h,m} - {x_0}_\ell^{h,m}|$ ($\tilde{x_0}_\ell^{h,m}$ is the new displacement) is the absolute value of the change in displacement in the ℓ^{th} moving step in the h^{th} dimension, and $\tilde{N}_\ell^{h,c}$ is the expansion order after the ℓ^{th} changing the expansion order in the h^{th} dimension. t_i is the time for carrying out the i^{th} scaling, moving, or p-adaptive technique in any dimension and if within the same time step more than one of those techniques in any dimension is used, those t_i may be the same but are listed in the order of carrying out those techniques.

Equation (7.3.22) can be proved in a dimension-by-dimension manner to evaluate the error caused by scaling Eq. (7.3.9), moving Eq. (7.3.11), and coarsening Eq. (7.3.14). As with Eq. (7.3.19), we also conclude that in multi-dimension cases the optimal strategy for choosing parameters is to set $q^h \lesssim 1$ and $0 \lesssim \delta^h$ in each dimension so that the change in the scaling factor or the displacement results in numerical accuracy but does not result in over-scaling or over-shifting. From the error lower bound in Eq. (7.3.20), $1 \lesssim \nu^h$ and $1 \lesssim \mu^h$ are required so that β^h and x_0^h are adjusted in each dimension h without incurring too large a projection error.

As for coarsening across higher dimensions, a larger η_0^h could lead to a larger minimal expansion order in each dimension and improve accuracy, but larger expansion orders lead to higher computational cost, especially for high-dimensional problems (as the total number of coefficients are $\Pi_{h=1}^D N^h$). Similarly, decreasing the initial refinement threshold η^h or γ^h, or the adjustment ratio η^h in the h^{th} direction, will lead to smaller errors and higher computational costs.

7.3.2 Prior error estimate

In addition to the posterior upper error bound of Eq. (7.3.19), we can also derive a prior error upper bound of using the adaptive spectral method to solve Eq. (7.2.6) in which the error estimate only depends on the solution itself. First, for the scaling technique, when we change the scaling factor from β to $\tilde{\beta}$ and use $\pi_{N,x_0}^{\tilde{\beta}} U_{N,x_0}^{\beta}$ as the new numerical solution, the error $\|u - \pi_{N,x_0}^{\tilde{\beta}} U_{N,x_0}^{\beta}\|$ can be bounded by

$$\begin{aligned}\|u - \pi^{\tilde{\beta}}_{N,x_0} U^{\beta}_{N,x_0}\| &\leq \|(I - \pi^{\tilde{\beta}}_{N,x_0})u\| + \|\pi^{\tilde{\beta}}_{N,x_0}(u - U^{\beta}_{N,x_0})\|, \\ &\leq \|(I - \pi^{\tilde{\beta}}_{N,x_0})u\| + \|u - U^{\beta}_{N,x_0}\|.\end{aligned} \tag{7.3.23}$$

In Eq. (7.3.23), the term $\|(i - \pi^{\tilde{\beta}}_{N,x_0})u\|$ in the last equation is the increment in the error bound resulting from scaling (scaling error). Similarly, if we carry out the moving technique and change the displacement of the basis function from x_0 to $\tilde{x}_0$ and use $\pi^{\beta}_{N,\tilde{x}_0} U^{\beta}_{N,x_0}$ as the new numerical solution, the error $\|u - \pi^{\beta}_{N,\tilde{x}_0} U^{\beta}_{N,x_0}\|$ can be bounded by

$$\begin{aligned}\|u - \pi^{\beta}_{N,\tilde{x}_0} U^{\beta}_{N,x_0}\| &\leq \|(I - \pi^{\beta}_{N,\tilde{x}_0})u\| + \|\pi^{\beta}_{N,\tilde{x}_0}(u - U^{\beta}_{N,x_0})\| \\ &\leq \|(I - \pi^{\beta}_{N,\tilde{x}_0})u\| + \|u - U^{\beta}_{N,x_0}\|.\end{aligned} \tag{7.3.24}$$

As for the p-adaptive technique, refinement will not bring any additional error since $\pi^{\beta}_{\tilde{N},x_0} U^{\beta}_{N,x_0} = U^{\beta}_{N,x_0}, \tilde{N} > N$. However, the error after coarsening and using $\pi^{\beta}_{\tilde{N},x_0} U^{\beta}_{N,x_0}, \tilde{N} < N$ to replace the original numerical solution U^{β}_{N,x_0} can be bounded by

$$\begin{aligned}\|u - \pi^{\beta}_{\tilde{N},x_0} U^{\beta}_{N,x_0}\| &\leq \|u - \hat{U}^{\beta}_{N,x_0}\| + \|(\pi^{\beta}_{N,x_0} - \pi^{\beta}_{\tilde{N},x_0})u\| \\ &\leq \|u - U^{\beta}_{N,x_0}\| + \|(\pi^{\beta}_{N,x_0} - \pi^{\beta}_{\tilde{N},x_0})u\|\end{aligned} \tag{7.3.25}$$

where

$$\hat{U}^{\beta}_{N,x_0} = \pi^{\beta}_{\tilde{N},x_0} U^{\beta}_{N,x_0} + \sum_{i=\tilde{N}+1}^{N} \hat{u}^{\beta}_{i,x_0} \hat{\mathcal{H}}^{\beta}_{i,x_0}(x), \;\; \hat{u}^{\beta}_{i,x_0} = (u(x,t), \hat{\mathcal{H}}^{\beta}_{i,x_0}(x)). \tag{7.3.26}$$

Finally, as with the derivation of Eq. (7.3.19), we can obtain an error bound that only

depends on the solution u

$$
\begin{aligned}
e(T) \leq & \sum_{i=0}^{K^s+K^m+K^c} \frac{b_{N_i,\beta_i}+B_0}{b_{N_i,\beta_i}} \sqrt{2(1+(t_{i+1}-t_i))} \|(I-\pi^{\beta_i}_{N_i,x_{0i}})u\|_{X(t_i,t_{i+1})} \\
& + \sum_{\ell=1}^{K^s} \|(I-\pi^{\tilde{\beta}^s_\ell}_{N^s_\ell,x_{L^s_\ell}})u(x,t^s_\ell)\| \\
& + \sum_{\ell=1}^{K^m} \|(I-\pi^{\beta^m_\ell}_{N^m_q,\tilde{x_0}^m_\ell})u(x,t^m_\ell)\| \\
& + \sum_{\ell=1}^{K^c} \|(\pi^{\beta}_{N,x_0}-\pi^{\beta^c_\ell}_{\tilde{N}^c_\ell,x_0{}^c_\ell})u(x,t^c_\ell)\|.
\end{aligned}
\tag{7.3.27}
$$

Therefore, the posterior error estimate Eq. (7.3.19) gives us more information on how we should choose the parameters in the adaptive techniques to determine β, x_0, N. Posterior error bounds for adaptive spectral methods for $(D+1)$-dimensional model problems ($x \in \mathbb{R}^D$) can be straightforwardly derived but is excluded for brevity.

7.4 Numerical results

In our numerical examples, we numerically solve Eq. (7.2.6) by discretizing time according to $t_j = j\Delta t$ and using the scheme Eq. (7.3.18) to forward time from t_j to t_{j+1}. Adaptive techniques will be used to adjust the basis functions at different timesteps t_j. The matrix-vector product $e^{-\boldsymbol{A}^\beta_N(t_{j+1}-t_j)}\boldsymbol{u}^\beta_{N,x_0}(t_j)$ in Eq. (7.3.18) is calculated using a "scaling and squaring" method in [MV78], *i.e.*, we rewrite

$$
e^{-\boldsymbol{A}^\beta_N(t_{j+1}-t_j)}\boldsymbol{u}^\beta_{N,x_0}(t_j) = \big(e^{-\frac{\boldsymbol{A}^\beta_N(t_{j+1}-t_j)}{3}}\big)^3 \boldsymbol{u}^\beta_{N,x_0}(t_j) \tag{7.4.1}
$$

and evaluate $e^{-\frac{\boldsymbol{A}^\beta_N(t_{j+1}-t_j)}{3}}\boldsymbol{u}^\beta_{N,x_0}(t_j)$ by Taylor expansion. The integral $\int_{t_j}^{t_{j+1}} e^{-\boldsymbol{A}^\beta_N(t_{j+1}-t_j)}\boldsymbol{F}^\beta_{N,x_0}(t)\mathrm{d}t$ on the RHS of Eq. (7.3.18) is evaluated by the Gauss-Legendre formula described in [XSC21b].

In all examples, the error denotes the relative L^2-error

$$\frac{\|u(\cdot,t) - U^{\beta}_{N,x_0}(\cdot,t)\|}{\|u(\cdot,t)\|}. \tag{7.4.2}$$

First, we numerically investigate how the parameters of the scaling and moving techniques affect the performance of the adaptive spectral method and the conclusions drawn from Eq. (7.3.19), namely, to set $q \lesssim 1, 1 \lesssim \nu$ for scaling, and $0 \lesssim \delta, 1 \lesssim \mu$ for moving in order to accurately adjust the scaling factor and translation of the basis functions. We also wish to explore how to appropriately set the parameters in the p-adaptive technique, the refinement threshold η, the coarsening threshold η_0, and the η adjustment ratio to achieve higher accuracy while reducing the computational cost. In this chapter, all computations were performed using Matlab R2017a on a laptop with a 4-core Intel(R) Core(TM) i7-8550U CPU @ 1.80 GHz.

Example 22. We consider solving the following parabolic equation in the weak form

$$\begin{aligned}
&(u_t(x,t), v) + \big(u_x(x,t), v_x(x,t)\big) = \big(f(x,t), v(x,t)\big), \forall v(x) \in H^1(\mathbb{R}),\\
&\quad u(x,0) = \exp(\mathrm{i}x) \cdot \exp(-\tfrac{x^2}{4}),\\
&f(x,t) = \frac{(x-2t) + (t+1)^3 + 2\mathrm{i}(x-t)(1+t)}{(t+1)^{\frac{3}{2}}} \exp\left[\mathrm{i}(t+1)x - \frac{(x-2t)^2}{4(t+1)}\right]
\end{aligned} \tag{7.4.3}$$

which admits an analytic solution

$$u(x,t) = \frac{1}{\sqrt{t+1}} \exp\left[\mathrm{i}(t+1)x - \frac{(x-2t)^2}{4(t+1)}\right]. \tag{7.4.4}$$

Not only is the center of the solution translating rightward at speed $2t$, its magnitude $|u(x,t)| = \frac{1}{\sqrt{t+1}} \exp\left(-\frac{(x-2t)^2}{4(1+t)}\right)$ decays more slowly for larger $|x|$. The solution also incurs higher frequency spatial variations as time increases due to the $\exp(\mathrm{i}(t+1)x)$ factor.

Therefore, all three adaptive techniques are expected to be required. Upon setting $\Delta t = 2 \times 10^{-4}$ and solving Eq. (7.4.3) up to $t = 2$, we investigate how the parameters in the

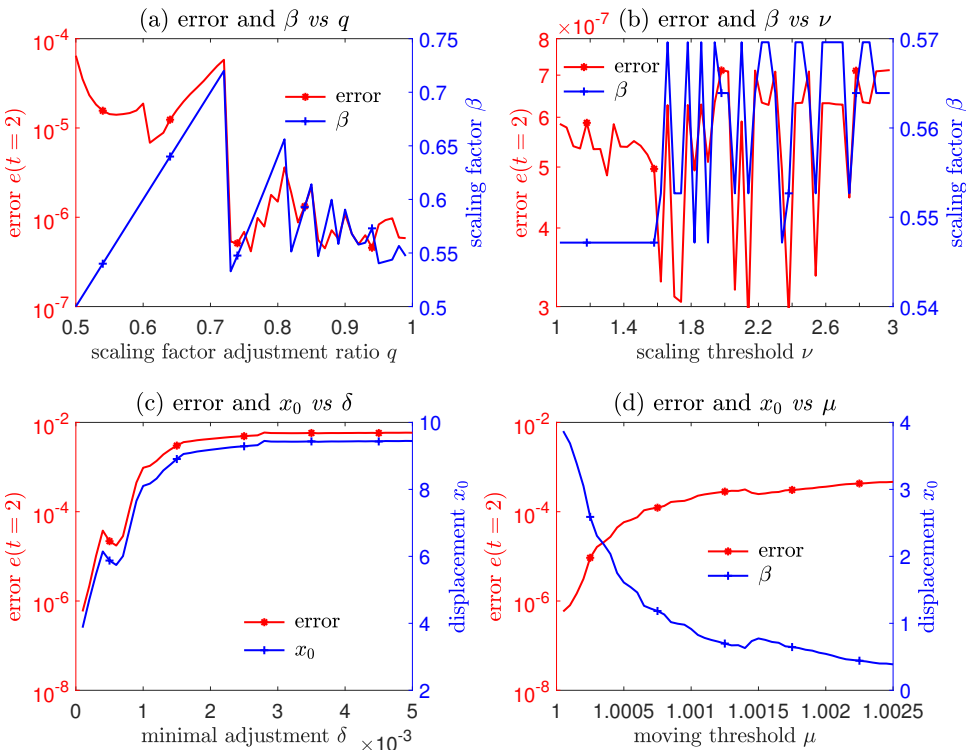

Figure 7.2: Plots of the error at $t = 2$ and the scaling factor β or the displacement x_0 when tuning the scaling factor adjustment ratio q and the scaling threshold ν or the minimum displacement δ and the moving threshold μ. (a) The error tends to be smaller as q decreases to 1, indicating that $q \lesssim 1$ is crucial for proper adjustment of the scaling factor. (b) As ν is increased, the scaling technique could be impeded, but the error is not very sensitive to ν if q is small. (c) The error is strongly correlated with x_0 and a large δ can lead to over-adjustment of the displacement x_0, resulting in a larger error. (d) A large μ will make it harder to activate the moving technique, leading to a smaller x_0 and a larger error.

three adaptive techniques affect performance. The initial scaling factor, displacement, and expansion order are set to $\beta = 1$, $x_0 = 0$, and $N = 40$. First, we test how the scaling threshold ν, the scaling factor adjustment ratio q, the moving μ, and the minimum displacement step δ affect the performance of the scaling and moving techniques. We keep the expansion order fixed since it has been illustrated that the effects of improper scaling or moving can be offset by increasing the expansion order N but at the expense of increased computational cost [XSC21a]. Initially, we set the parameters $q = 0.99, \nu = 1.02, \delta = 10^{-4}$, and $\mu = 1.00005$, and then change each of them one at a time. Imposing the maximal allowable displacement within each timestep $d_{\max} = 0.01$, the upper scaling factor limit $\overline{\beta} = 0.2$, and lower scaling factor limit $\overline{\beta} = 5$, we plot the relative L^2-error $e(t = 2)$ along with the scaling factor when we change q and ν, and we plot $e(t = 2)$ along with x_0 when we change δ and μ.

Fig. 7.2(a) shows that $q \lesssim 1$ is required for the scaling technique to properly adjust the scaling factor. When $q \lesssim 1$ and we vary ν from 1 to 2, the error, as well as the scaling factor β, do not change much, indicating that the scaling technique is more sensitive to q than to ν. Therefore, keeping $q \lesssim 1$ is more important than keeping $1 \lesssim \nu$. Fig. 7.2(c) shows that the error is highly correlated with x_0, suggesting that it is critical to properly move the basis functions to capture the displacement of the solution. Having $0 \lesssim \delta$ is important so that the displacement x_0 is not over-adjusted. Finally, as shown in Fig. 7.2(d), increasing μ will make the moving technique less sensitive to the translation of the basis functions and lead to a larger error. Thus, $1 \lesssim \mu$ is recommended for the moving technique.

Next, we investigate how the initial refinement threshold η, the refinement threshold adjustment ratio γ, and the coarsening threshold η_0 affect the p-adaptive technique's performance when $q = 0.99, \nu = 1.02, \delta = 10^{-4}$, and $\mu = 1.00005$ are fixed, and the initial variables are set to $\beta = 1, x_0 = 0, N = 40$. Fixing the maximum increment to $N_{\max} = 6$, we start with the initial parameter values $\gamma = 1.02$, $\eta = 1.05$, and $\eta_0 = 1.02$, and vary each of them one by one and plot the relative L^2-error and N. Fig. 7.3(a) shows that apart from translating rightward and decaying more slowly, the analytic solution is increasingly oscillatory which requires adjusting the expansion order N of the numerical solution. Fig. 7.3(b) shows

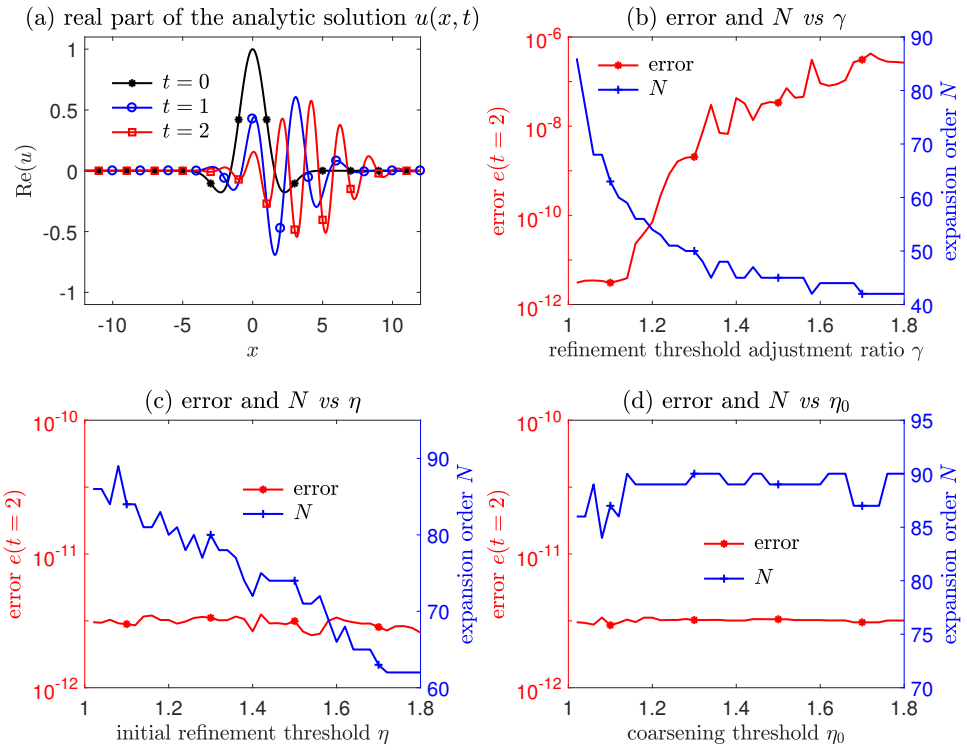

Figure 7.3: Plots of the real part of the analytic solution $\mathrm{Re}(u)(x,t)$ at different times, the error and the expansion order N at $t=2$ when we vary the refinement threshold adjustment ratio γ, the initial refinement threshold η, and the coarsening threshold η_0. (a) The real part of the analytic solution, which translates rightward, becomes more diffusive, and is increasingly oscillatory over time. (b) The error increases with γ while the expansion order decreases with γ. A larger γ implies a faster-increasing refinement threshold η. (c) A larger initial refinement threshold η results in a smaller expansion order at $t=2$, yet the error is not reduced as η decreases and N increases with the initial γ. This indicates that as long as γ is small enough, a larger initial η can be tolerated to lead to a smaller computational cost without compromising accuracy. (d) The expansion order N tends to increase as the coarsening threshold η_0 increases.

that if γ is large, then the threshold for increasing the expansion order η will increase more quickly. This renders the p-adaptive technique unable to sufficiently adjust the expansion order, leading to smaller expansion orders N and larger errors. Fig. 7.3(c) shows that the larger the initial threshold η for increasing the expansion order, the smaller the expansion order. In the depicted regime, larger initial values of η do not degrade accuracy since $N \gtrsim 65$ is sufficient to maintain high accuracy. Therefore, to maintain accuracy while reducing the computational burden, it is crucial to set $1 \lesssim \gamma$ so that the p-adaptive technique can capture oscillatory behavior over long periods of time. Using a smaller initial η may lead to more computational costs but does not lead to improvement in accuracy. Overall, since the function exhibits higher frequency spatial oscillations as time increases, coarsening is typically not activated. However, a large coarsening threshold η_0 can still impede coarsening resulting in a slightly larger N than a smaller η_0 (Fig. 7.3(d)).

Finally, as shown in Figs. 7.2 and 7.3, we numerically verify that the appropriate strategy for the adaptive spectral parameters is to set $q \lesssim 1, 1 \lesssim \nu, 0 \lesssim \delta$, and $1 \lesssim \mu$. In fact, for good performance, the scaling procedure strongly requires $q \lesssim 1$ and the moving procedure requires both $0 \lesssim \delta$ and $1 \lesssim \mu$. For an effective refinement, it is more important to set $1 \lesssim \gamma$ rather than to set the initial $1 \lesssim \eta$ (*i.e.*, setting $1 \lesssim \gamma$ rather than setting the initial $1 \lesssim \eta$ leads to more accurate results with smaller computational costs).

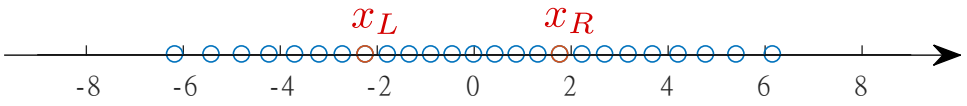

Figure 7.4: Distribution of the collocation points of generalized Hermite functions $\{\hat{\mathcal{H}}^{\beta}_{i,x_0}\}_{i=0}^{N}$ with $\beta = 1, x_0 = 0$, and $N = 24$. $x_L \coloneqq x^{\beta}_{[\frac{N}{3}]}$ and $x_R \coloneqq x^{\beta}_{[\frac{2N+2}{3}]}$ are marked in red. The number of collocation points that are in the right-exterior region (x_R, ∞) for calculating $\mathcal{E}_R$ and in the left-exterior region $(-\infty, x_L)$ for calculating $\mathcal{E}_L$ are both approximately $N/3$.

When using the generalized Hermite functions defined in $\mathbb{R}$, the desired solution might move leftward or rightward, requiring both leftward and rightward displacement of the basis functions. Since only rightward basis function shifts have been previously considered

[XSC21b, XSC21a], here, we generalize the moving technique to allow for bidirectional adjustment of the displacement x_0. We first propose a left exterior-error indicator

$$\mathcal{E}_L(U^{\beta}_{N,x_0}) = \frac{\|\partial_x U^{\beta}_{N,x_0} \cdot \mathbb{I}_{(-\infty,x_L)}\|}{\|\partial_x U^{\beta}_{N,x_0} \cdot \mathbb{I}_{(-\infty,+\infty)}\|}, \tag{7.4.5}$$

where we use $x_L = x^{\beta}_{[\frac{N}{3}]}$ following the often-used $\frac{2}{3}$-rule [HL07, Ors71]. The left exterior-error indicator (7.4.5) can be seen as the upper bound for the ratio of the error in $(-\infty, x_L)$ to the error across the whole space $\mathbb{R}$, in analogy to the (right) exterior-error indicator defined in [XSC21a]

$$\mathcal{E}_R(U^{\beta}_{N,x_0}) = \frac{\|\partial_x U^{\beta}_{N,x_0} \cdot \mathbb{I}_{(x_R,\infty)}\|}{\|\partial_x U^{\beta}_{N,x_0} \cdot \mathbb{I}_{(-\infty,+\infty)}\|}, \tag{7.4.6}$$

where $x_R = x^{\beta}_{[\frac{2N+2}{3}]}$. The number of nodes in the left-exterior region $(-\infty, x_L)$ and in the right-exterior region (x_R, ∞) are both roughly $\frac{N}{3}$. It was shown in [XSC21a] that if the right exterior-error indicator (7.4.6) increases, then the ratio of the error in the right exterior region $(x_R, +\infty)$ to the total error may also increase, suggesting that one should move the basis functions rightward (increase x_0). In Fig. 7.4, we show the positions of collocation nodes of generalized Hermite functions $\{\mathcal{H}^{\beta}_{i,x_0}\}_{i=0}^{N}$ with $\beta = 1, x_0 = 0$, and $N = 24$. The endpoints x_L and x_R are shown in red, showing that the right and left exterior regions, (x_R, ∞) and $(-\infty, x_L)$, are near-symmetric. The left exterior-error indicator (7.4.5) also measures the relative error in the left exterior region $(-\infty, x_L)$, and, if it increases, one can consider shifting the basis functions leftward (decrease x_0). With both left and right exterior-error indicators, we propose the following bidirectional moving scheme.

In Alg. 6, the LEFT_EXTERIOR_ERROR_INDICATOR subroutine calculates the right exterior-error indicator by Eq. (7.4.5) and the RIGHT_EXTERIOR_ERROR_INDICATOR calculates the left exterior-error indicator by Eq. (7.4.6). If the right or left exterior-error indicator is larger than their corresponding thresholds, *i.e*, $\mathcal{E}_R > \mu\tilde{\mathcal{E}}_R$ or $\mathcal{E}_L > \mu\tilde{\mathcal{E}}_L$, the moving technique is activated, calculating the rightward displacement d_0 or the leftward displacement d_1 of the

Algorithm 6 Pseudo-code of the bidirectional exterior-error-dependent moving technique.

1: Initialize N, Δt, T, β, x_0, $U^{\beta}_{N,x_0}(x,0)$, $\mu > 1$, $d_{max} > \delta > 0$
2: $t \leftarrow 0$
3: $x_R \leftarrow x^{\beta}_{[\frac{2N+2}{3}]}$
4: $x_L \leftarrow x^{\beta}_{[\frac{N}{3}]}$
5: $\tilde{\mathcal{E}}_R \leftarrow$ RIGHT_EXTERIOR_ERROR_INDICATOR$(U^{\beta}_{N,x_0}(x,0))$
6: $\tilde{\mathcal{E}}_L \leftarrow$ LEFT_EXTERIOR_ERROR_INDICATOR$(U^{\beta}_{N,x_0}(x,0))$
7: **while** $t < T$ **do**
8: $\quad U^{\beta}_{N,x_0}(x,t+\Delta t) \leftarrow$ EVOLVE$(U^{\beta}_{N,x_0}(x,t),\Delta t)$
9: $\quad \mathcal{E}_R \leftarrow$ RIGHT_EXTERIOR_ERROR_INDICATOR$(U^{\beta}_{N,x_0}(x,t+\Delta t))$
10: $\quad \mathcal{E}_L \leftarrow$ LEFT_EXTERIOR_ERROR_INDICATOR$(U^{\beta}_{N,x_0}(x,t+\Delta t))$
11: $\quad$ **if** $\mathcal{E}_R > \mu\tilde{\mathcal{E}}_R \,||\, \mathcal{E}_L > \mu\tilde{\mathcal{E}}_L$ **then**
12: $\quad\quad d_R \leftarrow$ MOVE_RIGHT$(U^{\beta}_{N,x_0}(t+\Delta t),\delta,d_{max},\mu e_0)$
13: $\quad\quad d_L \leftarrow$ MOVE_LEFT$(U^{\beta}_{N,x_0}(t+\Delta t),\delta,d_{max},\mu e_1)$
14: $\quad\quad U^{\beta}_{N,x_0}(x,t) \leftarrow \pi^{\beta}_{N,x_0+d_R-d_L} U^{\beta}_{N,x_0}(x,t+\Delta t)$
15: $\quad\quad x_0 \leftarrow x_0 + d_R - d_L$
16: $\quad\quad x_L \leftarrow x_L + d_R - d_L$
17: $\quad\quad x_R \leftarrow x_R + d_R - d_L$
18: $\quad\quad \tilde{\mathcal{E}}_R \leftarrow$ RIGHT_EXTERIOR_ERROR_INDICATOR$(U^{\beta}_{N,x_0}(x,t+\Delta t))$
19: $\quad\quad \tilde{\mathcal{E}}_L \leftarrow$ LEFT_EXTERIOR_ERROR_INDICATOR$(U^{\beta}_{N,x_0}(x,t+\Delta t))$
20: $\quad$ **end if**
21: $\quad t \leftarrow t + \Delta t$
22: **end while**

basis functions. In [XSC21b], the rightward displacement $d_R = \min\{n_R\delta, d_{\max}\}$ is determined by the MOVE_RIGHT subroutine in Line 12, where n is the smallest integer satisfying $\mathcal{E}_R(U^{(\alpha,\beta)}_{N,x_0}(x - n_R\delta, t)) < \mu\tilde{\mathcal{E}}_R$. Similarly, the leftward displacement $d_L = \min\{n_L\delta, d_{\max}\}$ is determined by the MOVE_LEFT subroutine in Line 13, where n_L is the smallest integer satisfying $\mathcal{E}_L(U^{(\alpha,\beta)}_{N,x_0}(x + n_L\delta, t)) < \mu\tilde{\mathcal{E}}_L$.

Example 23. Consider numerically solving the following parabolic equation in the weak form in $\mathbb{R} \times \mathbb{R}^+$

$$\begin{aligned} (u_t(x,t),v) + \big(u_x(x,t),v_x(x,t)\big) &= \big(f(x,t),v(x,t)\big), \forall v(x) \in H^1(\mathbb{R}), \\ u(x,0) &= \sin(x)\exp(-x^2), \end{aligned} \tag{7.4.7}$$

where

$$f(x,t)=\begin{cases}\Big[\Big(3-2(x+vt)\big(v+2(x+vt)\big)\Big)\sin(x+vt) & \\ \qquad +\Big(v+4(x+vt)\Big)\cos(x+vt)\Big]e^{-(x+vt)^2} & t\le 2,\\ \Big[\Big(3-4\big(x+v(4-t)\big)^2+2v\big(x+v(4-t)\big)\Big)\sin(x-v(t-4)) & \\ \qquad +\Big(4x+v(15-4t)\Big)\cos(x-v(t-4))\Big]e^{-(x-v(t-4))^2} & t\ge 2.\end{cases} \tag{7.4.8}$$

This PDE is solved by

$$u(x,t)=\begin{cases}e^{-(x+vt)^2}\sin(x+vt) & t\le 2,\\ e^{-(x-vt+4v)^2}\sin(x-vt+4v) & t\ge 2.\end{cases} \tag{7.4.9}$$

We set $v=2$ in Eq. (7.4.8) so that the center of the solution moves with velocity -2 from $x=0$ to $x=-4$ when $t\in[0,2]$, and when $t\in[2,6]$ the center of the solution moves from $x=-4$ to $x=4$ with velocity $+2$. Since the solution displays only convective behavior, we deactivate the scaling and p-adaptive procedures and apply only the moving technique. Since the translation switches from leftward to rightward at $t=2$, the moving technique needs to allow for both leftward and rightward displacement of the basis functions. The parameters in the moving technique are set to be $\mu=1.0005, \delta=0.0005$, and the maximal displacement within a timestep $d_{\max}=0.2$. We take the scaling factor, the expansion order, and the initial displacement of the basis function to be $\beta_0=1.2, N_0=24, x_0=0$, respectively, and plot the results obtained with no moving technique, the leftward-only moving technique, the rightward-only moving technique, and the bidirectional moving technique.

Fig. 7.5(a) shows that the spectral method equipped with the bidirectional moving technique (red) can maintain the smallest error because the displacement x_0 can be decreased when $t\in[0,2]$ and increased when $t>2$ (see Fig. 7.5(b)). The spectral method with the leftward-only moving technique (blue) can maintain a small error in $[0,2]$ when the center of the function moves leftward but fails to keep the error small when $t>2$ due to its inability

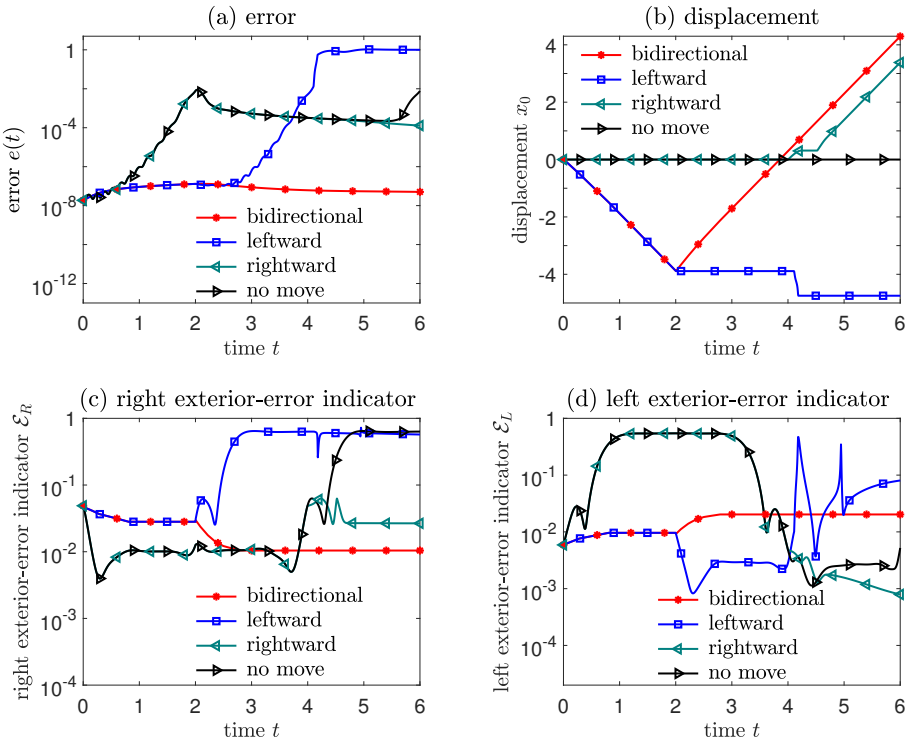

Figure 7.5: Plots of the error, x_0, the left exterior-error indicator Eq. (7.4.5), and the right exterior-error indicator Eq. (7.4.6). (a) The bidirectional moving technique Alg. 6 can main the smallest error while failure to accommodate either leftward or rightward displacement leads to much larger errors. (b,c,d) The displacement x_0, the left exterior-error indicator, and the right exterior-error indicator of spectral methods with the bidirectional, the leftward-only, the rightward-only moving technique, and the spectral method without any moving.

to increase x_0. When $t < 2$, the rightward-only moving technique (green) cannot decrease the displacement x_0 and therefore the error for the rightward-only moving technique is large at $t = 2$. Furthermore, large error accumulation before $t = 4$ of the rightward-only moving technique makes it unable to properly increase x_0 for $t > 4$ when the center of the solution moves to the right of the origin $x = 0$. The right and left exterior-error indicators for the bidirectional moving technique Alg. 6 can be well controlled as shown in Fig. 7.5(c,d), while for the leftward-only moving technique the right exterior-error indicator grows dramatically when $t > 2$ and for the rightward-only moving technique, the left exterior-error indicator grows when $t < 2$. Therefore, the leftward- and rightward-only moving techniques both fail to maintain a small error in at least one exterior region (x_R, ∞) or $(-\infty, x_L)$. The left exterior-error indicator grows when $t < 2$ (the center moves to the left of the origin) and the right exterior-error indicator grows when $t > 4$ (the center moves to the right of the origin) for the spectral method without the moving technique (black), suggesting that it cannot maintain a small error in both exterior regions.

7.5 Summary and conclusions

In this chapter, we carried out a numerical analysis of recently proposed adaptive spectral methods in unbounded domains using generalized Hermite functions. Specifically, our analysis helps guide parameter choice across three adaptive spectral techniques, *i.e.*, the scaling procedure, the moving procedure, and the p-adaptive technique to properly adjust the three key variables associated with these techniques, the scaling factor, the displacement, and the spectral expansion order. Based on our analyses, rules for properly choosing parameters in the scaling, moving, and p-adaptive techniques to most efficiently and accurately solve PDEs are derived. Numerical experiments were carried out to verify our theoretical results. Furthermore, we developed a new bidirectional moving technique to accommodate both leftward and rightward displacements.

What remain are analyses of adaptive spectral techniques using other classes of basis

functions that have been of recent interest [TWY20]. These include generalized Laguerre functions in $\mathbb{R}^+$ and the modified mapped Gegenbauer functions in $\mathbb{R}$. Another potentially useful extension is to explore developing methods to automatically determine and adjust the decay rate of solutions at infinity by adaptively switching among different classes of basis functions in order to match underlying physics or observations.

CHAPTER 8

Spectrally adapted physics-informed neural networks for solving unbounded domain problems

This is the Accepted Manuscript version of an article accepted for publication in *Machine Learning: Science and Technology*, **4**, (2023), pp.025024. It is an open-access paper. The Version of Record is available online at [10.1088/2632-2153/acd0a1].

8.1 Introduction

The use of neural networks as universal function approximators [Hor91, PYL20] led to various applications in simulating [RPK19, KKL21] and controlling [ABA22b, BAA22, BA22, LJY20] physical, biological, and engineering systems. Training neural networks in function-approximation tasks is typically realized in two steps. In the first step, an observable u_s associated with each distinct sample or measurement point $(x,t)_s \equiv (x_s, t_s)$, $s = 1, 2, \ldots, n$ is used to construct the corresponding loss function (*e.g.*, the mean squared loss) in order to find representations for the constraint $u_s \equiv u(x_s, t_s)$ or infer the equation that the function $u(x,t)$ obeys. In many physical settings, the variables x and t denote the space and time variables, respectively. Thus, the data points $(x,t)_s$ in many cases can be classified in two groups, $\{x_s\}$ and $\{t_s\}$, and the information they contain may be manifested differently in an optimization process. In the second step, the loss function is minimized by backpropagating gradients to adjust neural network parameters Θ. If the number of observations n is limited, additional constraints may help to make the training process more effective [KGC17].

To learn and represent the dynamics of physical systems, the constraints used in physics-informed neural networks (PINNs) [RPK19, KKL21] provide one possible option of an inductive bias in the training process. The key idea underlying PINN-based training is that the constraints imposed by the known equations of motion for some parts of the system are embedded in the loss function. Terms in the loss function associated with the differential equation can be evaluated using a neural network, which could be trained via backpropagation and automatic differentiation. In accordance with the distinction between Lagrangian and Hamiltonian formulations of the equations of motion in classical mechanics, physics-informed neural networks can be also divided into these two categories [LRP19, RRB20, ZDC19]. Another formulation of PINNs uses variational principles [KZK19] in the loss function to further constrain the types of functions used. Such variational PINNs rely on finite element (FE) methods to discretize partial differential equation (PDE)-type constraints.

Many other PINN-based numerical algorithms have been recently proposed. A space-

time domain decomposition PINN method was proposed for solving nonlinear PDEs [JK20]. In other variants, physics-informed Fourier neural operators have also been proposed to learn the underlying PDE models [LZK21]. In general, PINNs link modern neural network methods with traditional complex physical models and allow algorithms to efficiently use higher-order numerical schemes to (i) solve complex physical problems with high accuracy, (ii) infer model parameters, and (iii) reconstruct physical models in data-driven inverse problems [RPK19]. Therefore, PINNs have become increasingly popular as they can avoid certain computational difficulties encountered when using traditional FE/FD methods to find solutions to physics models.

The broad utility of PINNs is revealed by their numerous applications to problems in aerodynamics [MJK20], surface physics [FZ19], power systems [MVC20], cardiology [SYP20], and soft biological tissues [LLS20]. PINNs have also been integrated into the multi-task learning [TNF21] and meta-learning [PZN23] frameworks. When implementing PINN algorithms to find functions in an unbounded system, the unbounded variables cannot be simply normalized, precluding the reconstruction of solutions outside the range of data. Nonetheless, many problems in nature are associated with long-ranged potentials [BH21, SB19] (*i.e.*, unbounded spatial domains) and processes that are subject to algebraic damping [BOY11] (*i.e.*, unbounded temporal domains), and thus need to be solved in unbounded domains. For example, to capture the oscillatory and decaying behavior at infinity of the solution to Schrödinger's equation, efficient numerical methods are required in the unbounded domain $\mathbb{R}$ [LZZ18]. As another example, in structured cellular proliferation models in mathematical biology, efficient unbounded domain numerical methods are required to detect and better resolve possible blow-up in mean cell size [XGC20, XC21]. Finally, in solid-state physics, long-range interactions [MHR11, HDO12] require algorithms tailored for unbounded domain problems to accurately simulate particle interactions over long distances.

Solving unbounded domain problems is thus a key challenge in various fields that cannot be addressed with standard PINN-based solvers. In static problems, if the solution's behavior at infinity is known, one can use boundary-layer methods to truncate the unbounded domain

by discretizing space [BE22]. However, in spatiotemporal problems, it is often the case that the solution's behavior is evolving over time or otherwise unknown. Solving a PDE in this situation requires proper detection and capturing of the function's long-range behavior over time. Thus, simply discretizing space or truncating the domain is usually not effective in spatiotemporal problems. To efficiently solve PDEs in unbounded domains, we will treat the information carried by the x_s data using spectral decompositions of the function $u(x,t)$ in the x variable. Typically, a spatial initial condition of the desired solution is given and some spatial regularity is assumed from the underlying physical process. As a consequence, we suppose that at time t, we can use a spectral expansion in x to record spatial information. On the other hand, a solution's behavior in time t is unknown and one still has to numerically step forward in time to obtain the solution. Thus, we combine PINNs with spectral methods and propose a spectrally adapted PINN (s-PINN) method that can utilize recently developed adaptive function expansion techniques [XSC21a, XSC21b].

In contrast to traditional numerical spectral schemes that can only furnish solutions at discrete, predetermined timesteps, our approach uses time t as an input variable into the neural network combined with the PINN method to define a loss function, which enables (i) easy implementation of high-order Runge-Kutta schemes to relax the constraint on timesteps and (ii) easy extrapolation of the numerical solution at any time. However, our approach is distinct from that taken in standard PINNs, variational-PINNs, or physics-informed neural operator approaches. We do not input spatial positions x into the network or try to learn the x-dependence of $u(x,t)$; instead, we assume that the function $u(x,t)$ can be approximated by a spectral expansion in x with appropriate basis functions. Rather than learning the explicit spatial dependence directly, we train the neural network to learn the time-dependent expansion coefficients. Our main contributions include (i) integrating spectral methods into multi-output neural networks to approximate the spectral expansions of functions when partial information is available, (ii) incorporating recently developed adaptive spectral methods in our s-PINNs to allow accurate solutions of unbounded-domain spatiotemporal PDEs, and (iii) presenting explicit examples illustrating how s-PINNs can be used to solve unbounded

domain problems, recover spectral convergence, and more easily solve inverse-type PDE inference problems. We show how s-PINNs provide a unified, easy-to-implement method for solving PDEs and performing parameter inference given noisy observation data and how complementary adaptive spectral techniques can further improve efficiency, especially for solving problems in unbounded domains.

In Section 8.2.7, we show how neural networks can be combined with modern adaptive spectral methods to outperform standard neural networks in function approximation tasks. As a first application, we show in Section 8.3 how efficient PDE solvers can be derived from spectral PINN methods. In Section 8.4, we discuss another application that focuses on reconstructing underlying physical models and inferring model parameters given observational data. In Section 8.5, we summarize our work and discuss possible directions for future research. A summary of the main variables and parameters used in this study is given in Table 8.1. Our source codes are publicly available at `https://gitlab.com/ComputationalScience/spectrally-adapted-pinns`.

8.2 Combining Spectral Methods with Neural Networks

In this section, we first introduce the basic features of function approximators that rely on neural networks and spectral methods designed to handle variables that are defined in unbounded domains. In a dataset (x_s, t_s, u_s), $s \in \{1, \ldots, n\}$, x_s are values of the sampled "spatial" variable x which can be defined in an unbounded domain. We will also assume that our problem is defined within a finite time horizon so that t_s are time points restricted to a bounded domain, and are thus normalizable. Our key assumption is that the solution's behavior in x can be represented by a spectral decomposition, while u's behavior in t remains unknown and is to be learned from the neural network. This is achieved by isolating the possibly unbounded spatial variables x from the bounded variables t by expressing u in terms of suitable basis functions in x with time-dependent weights. As indicated in Fig. 8.1(a), we

Symbol	Definition
n	Number of observations
N	Spectral expansion order
N_H	Number of intermediate layers in the neural network
H	Number of neurons per layer
η	Learning rate of stochastic gradient descent
Θ	Neural network parameters (weights and biases)
K	Order of the Runge–Kutta scheme
$\mathcal{L}$	Loss function, *e.g.*, sum of squared errors (SSEs)
β	Scaling factor in basis functions $\phi^{\beta}_{i,x_L}(x) \coloneqq \phi_i(\beta(x - x_L))$
x_L	Translation of basis functions $\phi^{\beta}_{i,x_L} \coloneqq \phi_i(\beta(x - x_L))$
u^{β}_{N,x_L}	Spectral expansion of order N generated by the neural network: $u^{\beta}_{N,x_L} = \sum_{i=0}^{N} w^{\beta}_{i,x_L} \phi_i(\beta(x - x_L))$
$\mathcal{F}(u^{\beta}_{N,x_L})$	Frequency indicator for the spectral expansion u^{β}_{N,x_L}
$\hat{\mathcal{H}}^{\beta}_{i,x_L}$	Generalized Hermite function of order i, scaling factor β, and translation x_L
P^{β}_{N,x_L}	Function space defined by the first $N+1$ generalized Hermite functions $P^{\beta}_{N,x_L} \coloneqq \{\hat{\mathcal{H}}^{\beta}_{i,x_L}\}_{i=0}^{N}$
q	Scaling factor (β) adjustment ratio
ν	Threshold for adjusting the scaling factor β
ρ, ρ_0	Threshold for increasing, decreasing N
γ	Ratio for adjusting ρ

Table 8.1: **Overview of variables.** Definitions of the main variables and parameters used in this chapter.

approximate u_s using

$$u_s \coloneqq u(x_s, t_s) \approx u_N(x_s, t_s) \coloneqq \sum_{i=0}^{N} w_i(t_s)\phi_i(x_s), \tag{8.2.1}$$

where $\{\phi_i\}_{i=0}^{N}$ are suitable basis functions that can be used to approximate u in an unbounded domain (see Fig. 8.1(b) for a schematic of a basis function $\phi_i(x)$ that decays with x). Examples of such basis functions include, for example, the generalized Laguerre functions in $\mathbb{R}^+$ and the generalized Hermite functions in $\mathbb{R}$ [STW11]. In addition to being defined on an unbounded domain, spectral expansions allow high accuracy [Tre00] calculations with

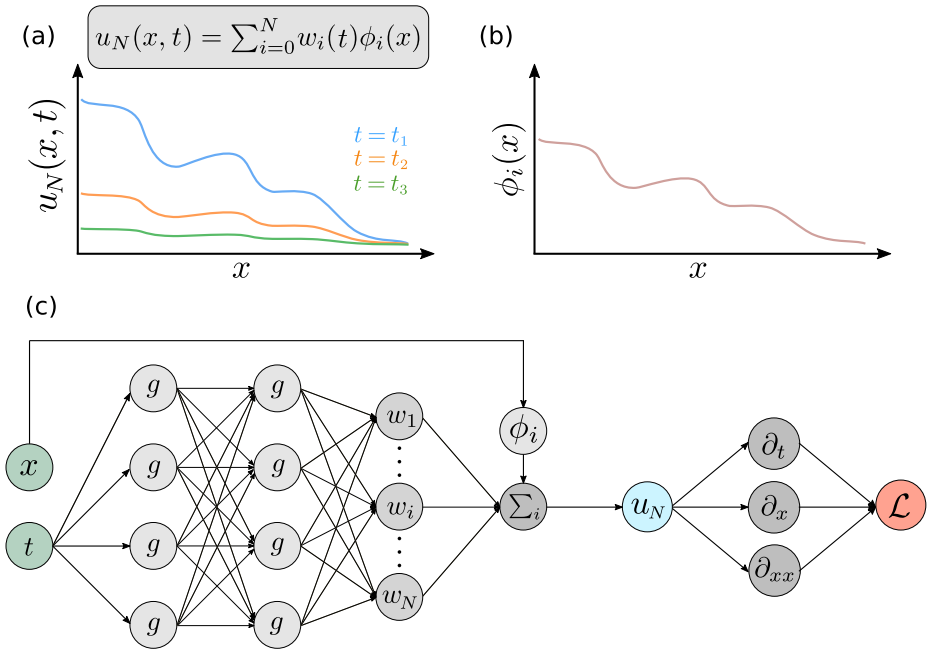

Figure 8.1: Solving unbounded domain problems with spectrally adapted physics-informed neural networks for functions $u_N(x,t)$ that can be expressed as a spectral expansion $u_N(x,t) = \sum_{i=0}^{N} w_i(t)\phi_i(x)$. (a) An example of a function $u_N(x,t)$ plotted at three different time points. (b) Decaying behavior of a corresponding basis function element $\phi_i(x)$. (c) PDEs in unbounded domains can be solved by combining a PINN with a neural network approximation of the spectral representation, $u_N(x,t,;\Theta) = \sum_{i=0}^{N} w_i(t;\Theta)\phi_i(x)$, and minimizing the loss function $\mathcal{L}$. Spatial derivatives of basis functions are explicitly defined and easily obtained. Here, g denotes an activation function such as the ReLU function.

errors that decay exponentially (spectral convergence) in space if the target function $u(x,t)$ is smooth.

Figure 8.1(c) shows a schematic of our proposed spectrally adapted PINN algorithm. The variable x is directly fed into the basis functions ϕ_i instead of being used as an input in the neural network. If one wishes to connect the output $u_N(x,t;\Theta)$ of the neural network (here, Θ represents the parameters of the neural network) to the solution of a PDE and perform backpropagation to minimize a loss functional $\mathcal{L}[u_N(x,t;\Theta), u_s(x,t)]$, it must contain spatial derivatives of u_N intrinsic to the underlying PDE. Derivatives that involve the variable x can be easily and explicitly calculated by taking derivatives of the basis functions with high accuracy while derivatives with respect to t can be obtained via automatic

differentiation [Lin76, PGC17].

If a function u can be written in terms of a spectral expansion in some dimensions (*e.g.*, x in Eq. (8.2.1)) with appropriate spectral basis functions, we can approximate u using a multi-output neural network by solving the corresponding least squares optimization problem

$$\min_{\Theta} \left\{ \sum_{s=1}^{n} \left| u_N(x_s, t_s; \Theta) - u_s \right|^2 \right\}, \quad u_N(x, t; \Theta) = \sum_{i=0}^{N} w_i(t; \Theta) \phi_i(x), \tag{8.2.2}$$

where n is the number of sample points. The neural network outputs the t-dependent vector of coefficients $w_i(t; \Theta)$. This representation will be used in the appropriate loss function depending on the application. The neural network can achieve arbitrarily high accuracy in the minimization of the loss function if it is deep enough and contains sufficiently many neurons in each layer [HSW89]. Since the solution's spatial behavior has been approximated by the spectral expansion which could achieve high accuracy with proper ϕ_i, we shall show that solving Eq. (8.2.2) can be more accurate and efficient than directly fitting to u_s by a neural network without using a spectral expansion. The proper choice of basis function $\phi_i(x)$ usually depends on the domain and how the solution decays at infinity. Overviews of asymptotic properties of basis functions are given in [STW11, BVO20]. For instance, in bounded domains, using any set of basis functions in the Jacobi polynomial family leads to the same convergence order for smooth functions and usually similar performance; in a semi-unbounded domain $\mathbb{R}^+$, the generalized Laguerre functions are often used; in the whole unbounded domain $\mathbb{R}^+$, the generalized Hermite functions are a common choice if the function decays exponentially at infinity. If the solution is expected to decay algebraically at infinity, the mapped Jacobi functions, such as the modified mapped Gegenbauer functions (MMGFs) are to be used [STW11].

As a motivating example, we compare the approximation error of a neural network that is fed both x_s and t_s with that of the s-PINN method in which only t_s are inputted, but with the information contained in x_s imposed on the solution via the basis functions $\{\phi_i(x)\}_{i=0}^{N}$. We show that taking advantage of the prior knowledge on the x-data greatly improves training

efficiency and accuracy. All neural networks that we use in our examples are based on fully connected linear layers with ReLU activation functions. Weights and biases in each layer are initially distributed according to a uniform distribution $\mathcal{U}(-\sqrt{a}, \sqrt{a})$, where a is the inverse of the number of input features. To normalize hidden-layer outputs, we apply the batch normalization technique [IS15]. Neural network parameters are optimized using stochastic gradient descent.

Example 24.: Function approximation

Consider approximating the function

$$u(x,t) = \frac{8x \sin 3x}{(x^2+4)^2}\, t, \tag{8.2.3}$$

which decays algebraically as $u(x \to \infty, t) \sim t/|x|^3$ when $|x| \to \infty$. To numerically approximate Eq. (8.2.3), we choose the loss function to be the mean-squared error

$$\text{MSE} = \frac{1}{n} \sum_{s=1}^{n} \left| u_N(x_s, t_s) - u_s \right|^2. \tag{8.2.4}$$

A standard feed-forward neural network approach is applied by inputting *both* x_s and t_s into a 5-layer, 15 neuron-per-layer network defined by the neural network parameters $\tilde{\Theta}$ to find a numerical approximation to

$$u_N(x_s, t_s) \coloneqq \tilde{u}(x_s, t_s; \tilde{\Theta}) \tag{8.2.5}$$

by minimizing Eq. (8.2.4) with respect to $\tilde{\Theta}$.

To apply a multi-output neural network to this problem, we need to choose an appropriate spectral representation of the spatial dependence of Eq. (8.2.3), in the form of Eq. (8.2.2). To capture an algebraic decay at infinity as well as the oscillatory behavior resulting from the $\sin(3x)$ term, we start from the modified mapped Gegenbauer functions (MMGFs) [TWY20]

$$R_i^{\lambda,\beta}(x) = (1+(\beta x)^2)^{-(\lambda+1)/2} C_i^{\lambda}\left(\beta x / \sqrt{1+(\beta x)^2}\right), \quad x \in \mathbb{R}, \tag{8.2.6}$$

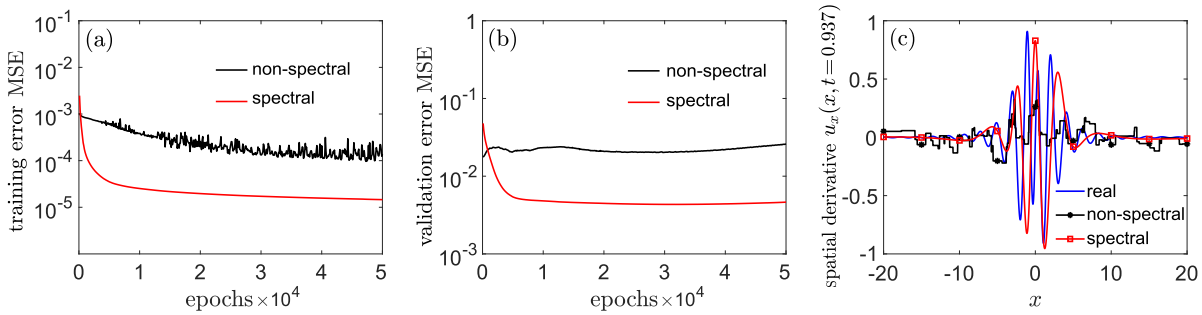

Figure 8.2: Example 24: Function approximation. Approximation of the target function Eq. (8.2.3) using both standard feed-forward neural networks and a spectral multi-output neural network that learns the coefficients $w_i(t;\Theta)$ in the spectral expansion Eq. (8.2.1). Comparison of the approximation error using a spectral multi-output neural network (red) with the error incurred when using a standard neural-network function approximator (black). Here, both the spectral and non-spectral function approximators use the same number of parameters, but the spectral multi-output neural network converges much faster on the training set and has a smaller validation error than the standard feed-forward neural network. (a) The training curve of the spectral multi-output neural network decreases much faster than that of the standard feed-forward neural network. (b) Since the spectral multi-output neural network is better at fitting the data by taking advantage of the spectral expansion in x, its validation error is also much smaller and decreases faster. (c) Asymptotic behavior of the spatial derivatives of the analytic solution $\partial_x u(x,t)$, the feed-forward neural network $\partial_x \tilde{u}(x,t;\tilde{\Theta})$ (Eq. (8.2.5)), and the spectral neural network $\partial_x u_N(x,t;\Theta)$ (Eq. (8.2.7)). The feed-forward neural network fails to capture the function's behavior when $|x|$ is large because $\partial_x \tilde{u}(x,t;\tilde{\Theta})$ is not vanishing for large $|x|$, but the spectral approximation Eq. (8.2.7) leads to smaller errors because $\partial_x u_N(x,t;\Theta)$ better approximates $\partial_x u(x,t)$ especially when $|x|$ is large. Here, $t = 0.937$ is randomly chosen from one of the training samples.

where $C_i^{\lambda}(\cdot)$ is the Gegenbauer polynomial of order i. At infinity, the MMGFs decay as $R_i^{\lambda,\beta}(x) \sim \text{sign}(x)^i \frac{(2\lambda)^{(i)}}{i!}(1+(\beta x)^2)^{-(\lambda+1)/2}$, where $(2\lambda)^{(i)}$ is the i^{th} rising factorial of 2λ. A suitable basis ϕ_i needs to include functions that decay more slowly than x^{-3}. If we choose $\beta = 1/4$ and the special case $\lambda = 0$, the basis function is defined as $\phi_i(x) = R_i^{0,\beta}(x) \equiv (1+(\beta x)^2)^{-1/2} T_i(\beta x/\sqrt{1+(\beta x)^2})$, where T_i are the Chebyshev polynomials. We thus use

$$u_N(x_s, t_s; \Theta) = \sum_{i=0}^{N=9} w_i(t_s;\Theta) R_i^{0,\beta}(x_s) \tag{8.2.7}$$

in Eq. (8.2.4) and use a 4-layer neural network with 15 neurons per layer to learn the coefficients $\{w_i(t;\Theta)\}_{i=0}^{1}4$ by minimizing the MSE (Eq. (8.2.4)) with respect to Θ. The total numbers of parameters for both the 4-layer spectral multi-output neural network and the

normal 5-layer neural network are the same. The training set and the validation set each contain $n = 200$ pairs of values $(x,t)_s = (x_s, t_s)$ where x_s are sampled from the Cauchy distribution, $x_s \sim \mathcal{C}(12, 0)$, and $t_s \sim \mathcal{U}(0, 1)$. For each pair (x_s, t_s), we find $u_s \equiv u(x_s, t_s)$ using equation (8.2.3). The positions x_s are sampled from the unbounded domain $\mathbb{R}$ and cannot be normalized (the expectation and variance of the Cauchy distribution do not exist). The minimum (maximum) value of x in the training set and the validation set are -18.65 (50.32) and -721.50 (120.01), respectively.

We set the learning rate $\eta = 5 \times 10^{-4}$ and plot the training and validation MSEs (Eq. (8.2.4)) as a function of the number of training epochs in Fig. 8.2. Figures 8.2(a) and (b) show that the spectral multi-output neural network yields smaller errors since it naturally and efficiently captures the oscillatory and decaying feature of the underlying function u from Eq. (8.2.3). Directly fitting $u \approx \tilde{u}$ leads to over-fitting on the training set which does nothing to reduce the validation error. We can see from Fig. 8.2(c) that using the feed-forward neural network, Eq. (8.2.5) results in a nonvanishing spatial derivative when $|x|$ is large. Such an approximation to the original function $u(x,t)$, which vanishes for large $|x|$, is thus inaccurate. On the other hand, the spatial derivative of the spectral neural network Eq. (8.2.7) better fits $u(x,t)$ especially as $|x| \to \infty$. Therefore, it is important to take advantage of the data structure, in this case, using the spectral expansion to represent the function's known oscillations and decay as $x \to \infty$.

In this and subsequent examples, all computations are performed using Python 3.8.10 on a laptop with a 4-core Intel® i7-8550U CPU @ 1.80 GHz.

8.3 Application to Solving PDEs

In this section, we show that spectrally adapted neural networks can be combined with physics-informed neural networks (PINNs) which we shall call spectrally adapted PINNs (s-PINNs). We apply s-PINNs to numerically solve PDEs, and in particular, spatiotemporal PDEs in unbounded domains for which standard PINN approaches cannot be directly

applied. Although we mainly focus on solving spatiotemporal problems, s-PINNs are also applicable to other types of PDEs.

Again, we assume that the problem is defined over a finite time horizon t while the spatial variable x may be defined in an unbounded domain. Assuming the solution's asymptotic behavior in x is known, we approximate it by a spectral expansion in x with suitable basis functions (*e.g.*, MMGFs in Example 24 for describing algebraic decay at infinity). Assuming $\mathcal{M}$ is an operator that only involves the spatial variable x (*e.g.*, ∂_x, ∂_x^2, etc.), we can represent the solution to the spatiotemporal PDE $\partial_t u = \mathcal{M}[u](x,t)$ by the spectral expansion in Eq. (8.2.2) with expansion coefficients $\{w_i(t;\Theta)\}$ to be learned by a neural network with parameters Θ. If the solution's behavior in both x and t are known and one can find proper basis functions in both the x and t directions, then one could use a spectral expansion in both x and t to solve the PDE directly without time-stepping. However, it is often the case that the time dependence is unknown and $u(x,t)$ needs to be solved step-by-step in time.

As in standard PINNs, we use a high-order Runge–Kutta scheme to advance time by uniform timesteps Δt. What distinguishes our s-PINNs from standard PINNs is that only the intermediate times t_s between timesteps are provided as inputs to the neural network, while the outputs contain global spatial information (the spectral expansion coefficients), as shown in Fig. 8.1(c). Over a longer time scale, the optimal basis functions in the spectral expansion Eq. (8.2.2) may change. Therefore, one can use new adaptive spectral methods proposed in [XSC21b, XSC21a]. Using s-PINNs to solve PDEs has the advantages that they can (i) accurately represent spatial information via spectral decomposition, (ii) convert solving a PDE into an optimization and data fitting problem, (iii) easily implement high-order, implicit schemes to advance time with high accuracy, and (iv) allow the use of recently developed spectral-adaptive techniques that dynamically find the most suitable basis functions.

The approximated solution to the PDE $\partial_t u = \mathcal{M}[u](x,t)$ can be written at discrete

timesteps $t_{j+1} - t_j = \Delta t$ as

$$u_N(x, t_{j+1}; \Theta_{j+1}) = \sum_{i=0}^{N} w_i(t_{j+1}; \Theta_{j+1}) \phi_i(x), \tag{8.3.1}$$

where $\Theta_{j+1}, j \geq 1$ is the parameter set of the neural network used in the time interval $(j\Delta t, (j+1)\Delta t)$. In order to forward time from $t_j = j\Delta t$ to $t_{j+1} = (j+1)\Delta t$, we can use, *e.g.*, a K^{th}-order implicit Runge–Kutta scheme, with $0 < c_s < 1$ $(s = 1, \ldots, K)$ as parameters describing different collocation points in time and a_{rs}, b_r $(r = 1, \ldots, K)$ the associated coefficients.

Given $u(x, t_j)$, the K^{th}-order implicit Runge–Kutta scheme aims to approximate $u(x, t_j + c_s\Delta t)$ and $u(x, t_j + \Delta t)$ through

$$\begin{aligned} u_N(x, t_j + c_s\Delta t) &= u(x, t_j) + \sum_{r=1}^{K} a_{rs} \mathcal{M}\big[u_N(x, t_j + c_r\Delta t)\big], \\ u_N(x, t_j + \Delta t) &= u(x, t_j) + \sum_{r=1}^{K} b_r \mathcal{M}\big[u_N(x, t_j + c_r\Delta t)\big]. \end{aligned} \tag{8.3.2}$$

With the starting point $u_N(t_0, x; \Theta_0) \coloneqq u_N(t_0, x)$ defined by the initial condition at t_0, we define the target function as the sum of squared errors

$$\begin{aligned} \text{SSE}_j = \sum_{s=1}^{K} \Big\| u_N(x, t_j + c_s\Delta t; \Theta_{j+1}) - u_N(x, t_j; \Theta_j) - \sum_{r=1}^{K} a_{sr} \mathcal{M}[u_N(x, t_j + c_r\Delta t; \Theta_{j+1})] \Big\|_2^2 \\ + \Big\| u_N(x, t_j + \Delta t; \Theta_{j+1}) - u_N(x, t_j; \Theta_j) - \sum_{r=1}^{K} b_r \mathcal{M}[u_N(x, t_j + c_r\Delta t; \Theta_{j+1})] \Big\|_2^2, \end{aligned} \tag{8.3.3}$$

where the L^2 norm is taken over the spatial variable x. Minimization of Eq. (8.3.3) provides a numerical solution at t_{j+1} given its value at t_j. If coefficients in the PDE are sufficiently smooth, we can use the basis function expansion in Eq. (8.3.1) for u_N and find that the weights at the intermediate Runge–Kutta timesteps can be written as the Taylor expansion

$$w_i(t_j + c_r\Delta t; \Theta_{j+1}) = \sum_{\ell=0}^{\infty} \frac{w_i^{(\ell)}(t_j; \Theta_{j+1})}{\ell!}(c_r\Delta t)^\ell, \tag{8.3.4}$$

where $w_i^{(\ell)}(t_j)$ is the ℓ^{th} derivative of w_i with respect to time, evaluated at t_j. Therefore, the neural network is learning the mapping $t_j + c_s\Delta t \to \sum_{\ell=0}^{\infty} w_i^{(\ell)}(t_j)(c_s\Delta t)^\ell/\ell!$ for every i by minimizing the loss function Eq. (8.3.3).

Example 25.: Solving bounded domain PDEs

Before focusing on the application of s-PINNs to PDEs whose solution is defined in an unbounded domain, we first consider the numerical solution of a PDE in a bounded domain to compare the performance of the spectral PINN method (using recently developed adaptive methods) to that of the standard PINN.

Consider the following PDE:

$$\begin{aligned} \partial_t u &= \left(\frac{x+2}{t+1}\right)\partial_x u, \;\; x \in (-1, 1), \\ u(x, 0) = \cos(x+2), &\;\; u(1, t) = \cos(3(t+1)), \end{aligned} \tag{8.3.5}$$

which admits the analytical solution $u(x, t) = \cos((t + 1)(x + 2))$. In this example, we use Chebyshev polynomials $T_i(x)$ as basis functions and the corresponding Chebyshev-Gauss-Lobatto quadrature collocation points and weights such that the boundary $u(1, t) = \cos(3(t + 1))$ can be directly imposed at a collocation point $x = 1$. Since the solution becomes increasingly oscillatory in x over time, an ever-increasing expansion order (*i.e.*, the number of basis functions) is needed to accurately capture this behavior. Between consecutive timesteps, we employ a recently developed p-adaptive technique for tuning the expansion order [XSC21b]. This method is based on monitoring and controlling a frequency indicator $\mathcal{F}(u_N)$ defined by

$$\mathcal{F}(u_N) = \left(\frac{\sum_{i=N-[\frac{N}{3}]+1}^{N} \gamma_i w_i^2}{\sum_{i=0}^{N} \gamma_i w_i^2} \right)^{\frac{1}{2}}, \tag{8.3.6}$$

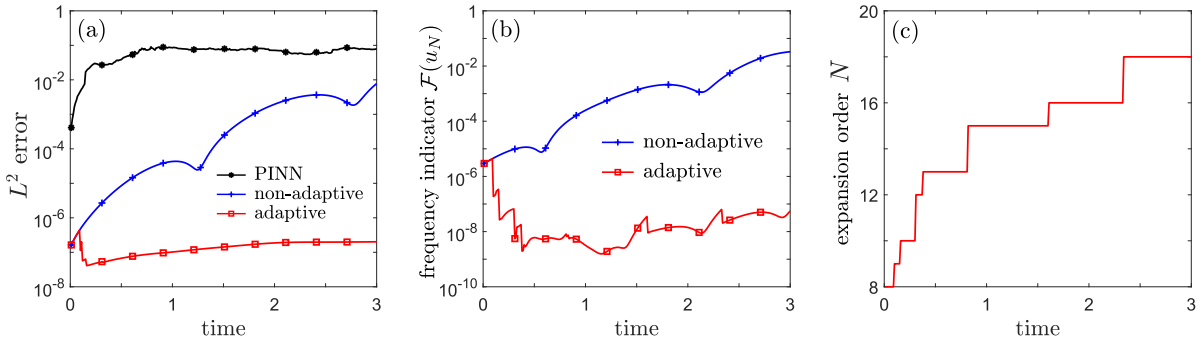

Figure 8.3: Example 25: Solving Eq. (8.3.5) in a bounded domain. L^2 errors, frequency indicators, and expansion order associated with the numerical solution of Eq. (8.3.5) using the adaptive s-PINN method with a timestep $\Delta t = 0.01$. (a) In a bounded domain, the s-PINNs, with and without the adaptive spectral technique, have smaller errors than the standard PINN (black). Moreover, the s-PINN method combined with a p-adaptive technique that dynamically increases the number of basis functions (red) exhibits a smaller error than the non-adaptive s-PINN (blue). The higher accuracy of the adaptive s-PINN is a consequence of maintaining a small frequency indicator Eq. (8.3.6), as shown in (b). (c) Keeping the frequency indicator at small values is realized by increasing the spectral expansion order.

where $\gamma_i := \int_{-1}^{1} T_i^2(x)(1-x^2)^{-1/2}\mathrm{d}x$. The frequency indicator $\mathcal{F}(u_N)$ measures the proportion of high-frequency waves and serves as a lower error bound of the numerical solution $u_N(x,t;\Theta) := \sum_{i=0}^{N} w_i(t;\Theta)T_i(x)$. When $\mathcal{F}(u_N)$ exceeds its previous value by more than a factor ρ, the expansion order is increased by one. The indicator is then updated and the factor ρ also is scaled by a parameter $\gamma \geq 1$.

We use a fourth-order implicit Runge–Kutta method to advance time in the loss function (8.3.3) and in order to adjust the expansion order in a timely way, we take $\Delta t = 0.01$. The initial expansion order $N = 8$, and the two parameters used to determine the threshold of adjusting the expansion order are set to $\rho = 1.5$ and $\gamma = 1.3$. A neural network with $N_H = 4$ layers and $H = 200$ neurons per layer is used in conjunction with the loss function ((8.3.3)) to approximate the solution of equation (8.3.5). We compare the results obtained using the s-PINN method with those obtained using a fourth-order implicit Runge–Kutta scheme with $\Delta x = \frac{1}{256}, \Delta t = 0.01$ in a standard PINN approach [RPK19], also using $N_H = 4$ and $H = 200$.

Figure 8.3 shows that s-PINNs can be used to greatly improve accuracy because the

spectral method can recover exponential convergence in space, and when combined with a high-order accurate implicit scheme in time, the overall error is small. In particular, the large error shown in Fig. 8.3 of the standard PINN suggests that the error of applying auto-differentiation to calculate the spatial derivative is significantly larger than the spatial derivatives calculated using spectral methods. Moreover, when equipping spectral PINNs with the p-adaptive technique to dynamically adjust the expansion order, the frequency indicator can be controlled, leading to even smaller errors as shown in Fig. 8.3(b,c).

Computationally, using our 4-core laptop in this example, the standard PINN method requires $\sim 10^6$ seconds while the s-PINN approach with and without adaptive spectral techniques (dynamically increasing the expansion order N) required 1711 and 1008 seconds, respectively. Thus, s-PINN methods can be computationally more efficient than the standard PINN approach. This advantage can be better understood by noting that training of standard PINNs requires time $\sim \mathcal{O}(\sum_{i=0}^{N_H} H_i H_{i+1})$ (H_i is the number of neurons in the i^{th} layer) to calculate each spatial derivative (*e.g.*, $\partial_x u, \partial_x^2 u, ...$) by autodifferentiation [BPR18]. However, in an s-PINN, since a spectral decomposition $u_N(x,t;\Theta)$ has been imposed, the computational time to calculate derivatives of all orders is $\mathcal{O}(N)$, where N is the expansion order. Since $\sum_{i=0}^{N_H} H_i H_{i+1} \geq \sum_{i=0}^{N_H} H_i$ and the total number of neurons $\sum_{i=0}^{N_H} H_i$ is usually much larger than the expansion order N, using s-PINNs can substantially reduce computational cost.

In bounded-domain problems, there are many other good machine-learning-based PDE solvers against which we can compare, such as the DeepONet method [LJP21], its PINN extension [WWP21], and the Fourier neural operator method [LKA20]. However, what distinguishes s-PINNs from the standard PINN framework is that the latter uses spatial and temporal variables as neural-network inputs, implicitly assuming that all variables are normalizable especially when batch-normalization techniques are applied while training the underlying neural network. Our s-PINN approach relies on spectral expansions to represent the dependence of a function $u(x,t)$ on the spatial variable x, which can then be defined in unbounded domains and does not need to be normalizable. Thus, our s-PINN method

provides a novel machine-learning-based PDE solver for unbounded-domain spatiotemporal problems. In the following example, we shall explore how our s-PINN is applied to solving a PDE defined in $(x,t) \in \mathbb{R}^+ \times [0,T]$.

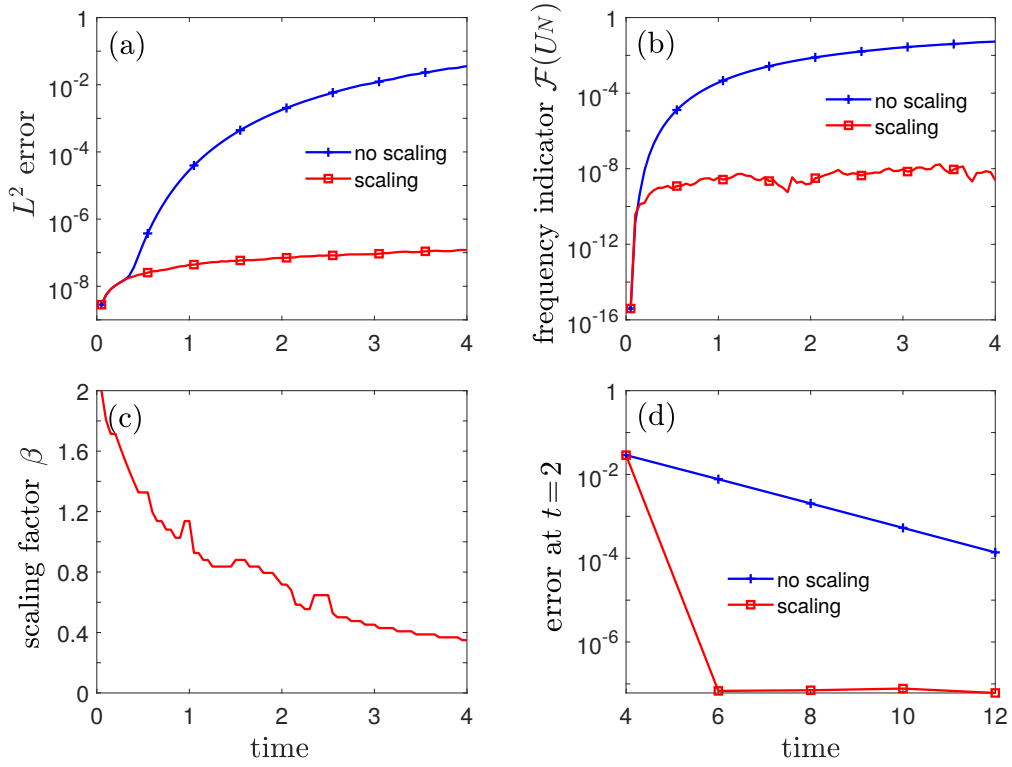

Figure 8.4: Example 26: Solving equation (8.3.7) in an unbounded domain. L^2 error, frequency indicator, and expansion order associated with the numerical solution of equation (8.3.7) using the s-PINN method combined with the spectral scaling technique. (a) The s-PINN method with the scaling technique (red) has a smaller error than the s-PINN without scaling (blue). The higher accuracy of the adaptive s-PINN is a consequence of maintaining a small frequency indicator equation (8.3.6), as shown in (b). (c) Keeping the frequency indicator at small values is possible by reducing the scaling factor so that the basis functions decay more slowly at infinity. The timestep is $\Delta t = 0.05$. (d) The errors for the spectral method with and without scaling at $t = 2$. When the scaling factor is properly adjusted, very high accuracy can be obtained with only a few basis functions. Not dynamically adjusting the scaling factor leads to a much slower convergence.

Example 26.: Solving unbounded domain PDEs

Consider the following PDE, which is similar to Eq. (8.3.5) but is defined in $(x,t) \in \mathbb{R}^+ \times [0,T]$:

$$\partial_t u = -\left(\frac{x}{t+1}\right)\partial_x u, \quad u(x,0) = e^{-x}, \;\; u(0,t) = 1. \tag{8.3.7}$$

Equation (8.3.7) admits the analytical solution $u(x,t) = \exp[-x/(t+1)]$. In this example,

we use the basis functions $\{\hat{\mathcal{L}}_i^{\beta}(x)\} \coloneqq \{\hat{\mathcal{L}}_i^{(0)}(\beta x)\}$ where $\hat{\mathcal{L}}_i^{(0)}(x)$ is the generalized Laguerre function of order i defined in [STW11]. Here, we use the Laguerre-Gauss quadrature collocation points and weights so that $x = 0$ is *not* included in the collocation node set. We use a fourth-order implicit Runge–Kutta method to minimize the SSE Eq. (8.3.3) by advancing time. In order to address the boundary condition, we augment the loss function in Eq. (8.3.3) with terms that represent the cost of deviating from the boundary condition:

$$\begin{aligned}\mathrm{SSE}_j = &\sum_{s=1}^{K} \left\| u_N(x, t_j + c_s\Delta t; \Theta_{j+1}) - u_N(x, t_j; \Theta_j) - \sum_{r=1}^{K} a_{sr}\mathcal{M}[u_N(x, t_j + c_r\Delta t; \Theta_{j+1})] \right\|_2^2 \\ &+ \left\| u_N(x, t_j + \Delta t; \Theta_{j+1}) - u_N(x, t_j; \Theta_j) - \sum_{r=1}^{K} b_r\mathcal{M}[u_N(x, t_j + c_r\Delta t; \Theta_{j+1})] \right\|_2^2 \\ &+ \sum_{s=1}^{K} \left[u_N(0, t_j + c_s\Delta t; \Theta_{j+1}) - u(0, t_j + c_s\Delta t)\right]^2 + \left[u_N(0, t_{j+1}; \Theta_{j+1}) - u(0, t_{j+1})\right]^2,\end{aligned}$$

where the last two terms push the constraints associated with the Dirichlet boundary condition at $x = 0$ at all time points:

$$u_N(0, t_j + c_s\Delta t; \Theta_{j+1}) = u(0, t_j + c_s\Delta t), \quad u_N(0, t_{j+1}; \Theta_{j+1}) = u(0, t_{j+1}), \tag{8.3.8}$$

where in this example, $u(0, t_j + c_s\Delta t) = u(0, t_{j+1}) \equiv 1$.

Because the solution of Eq. (8.3.7) becomes more diffusive with x (*i.e.*, decays more slowly at infinity), it is necessary to decrease the scaling factor β to allow basis functions to decay more slowly at infinity. Between consecutive timesteps, we adjust the scaling factor by applying the scaling algorithm proposed in [XSC21a]. Thus, we dynamically adjust the basis functions in Eq. (8.2.1). As with the p-adaptive technique we used in Example 25, the scaling technique also relies on monitoring and controlling the frequency indicator given in Eq. (8.3.6). In order to efficiently and dynamically tune the scaling factor, we set $\Delta t = 0.05$. The initial expansion order is $N = 8$, the initial scaling factor is $\beta = 2$, the scaling factor adjustment ratio is set to $q = 0.95$, and the threshold for tuning the scaling factor is set to $\nu = 1/(0.95)$. A neural network with 3 intermediate layers and 100 neurons per layer is

used in conjunction with the loss function given in equation ((8.3.3)). Figure 8.4(a) shows that s-PINNs can achieve very high accuracy even when a relatively large timestep ($\Delta t = 0.05$) is used. Scaling techniques to dynamically control the frequency indicator are also successfully incorporated into s-PINNs, as shown in Figs. 8.3.7(b,c), and very high accuracy can be achieved with only a few basis functions, as shown in Fig. 8.3.7(d). Actually, such spatiotemporal diffusive behavior in unbounded domains distinguishes unbounded-domain problems from bounded-domain problems, as we have to dynamically adjust the scaling factor over time using the scaling technique in [XSC21a].

In Eq. (8.3.7), we imposed a Dirichlet boundary condition by modifying the SSE Eq. (8.3.8) to include boundary terms. Other types of boundary conditions can be applied in s-PINNs by including boundary constraints in the SSE as in standard PINN approaches.

In the next example, we focus on solving a PDE with two spatial variables, x and y, each defined on an unbounded domain.

Example 27.: Solving 2D unbounded domain PDEs
Consider the two-dimensional heat equation on $(x, y) \in \mathbb{R}^2$

$$\partial_t u(x, y, t) = \Delta u(x, y, t), \quad u(x, y, 0) = \frac{1}{\sqrt{2}} e^{-x^2/12 - y^2/8}, \tag{8.3.9}$$

which admits the analytical solution

$$u(x, y, t) = \frac{1}{\sqrt{(t+3)(t+2)}} \exp\left[-\frac{x^2}{4(t+3)} - \frac{y^2}{4(t+2)}\right]. \tag{8.3.10}$$

Note that the solution spreads out over time in both dimensions, *i.e.*, it decays more slowly at infinity as time increases. Therefore, we apply the scaling technique to capture the increasing spread by adjusting the scaling factors β_x and β_y of the generalized Hermite basis functions. Generalized Hermite functions of orders $i = 0, \ldots, N_x$ and $\ell = 0, \ldots, N_y$ are used in the x and y directions, respectively.

In order to solve Eq. (8.3.9), we multiply it by any test function $v \in H^1(\mathbb{R})$ and integrate

the resulting equation by parts to convert it to the weak form $(\partial_t u, v) = -(\nabla u, \nabla v)$. Solving the weak form of Eq. (8.3.9) ensures numerical stability. When implementing the spectral method, the goal is to find

$$u_{N_x,N_y}^{\beta_x,\beta_y}(x,y,t) = \sum_{i=0}^{N_x}\sum_{\ell=0}^{N_y} w_{i,\ell}(t)\hat{\mathcal{H}}_{i,0}^{\beta_x}(x)\hat{\mathcal{H}}_{\ell,0}^{\beta_y}(y), \tag{8.3.11}$$

where $\hat{\mathcal{H}}_{i,0}^{\beta_x}$, $\hat{\mathcal{H}}_{\ell,0}^{\beta_y}$ are generalized Hermite functions defined in Table 8.1 such that $(\partial_t u, v) = -(\nabla u, \nabla v)$ $t \in (t_j, t_{j+1})$ for all $v \in P_{N_x,0}^{\beta_x} \times P_{N_y,0}^{\beta_y}$, $t \in (t_j, t_{j+1})$. This allows one to advance time from t_j to t_{j+1} given $u_{N_x,N_y}^{\beta_x,\beta_y}(x,y,t_j)$.

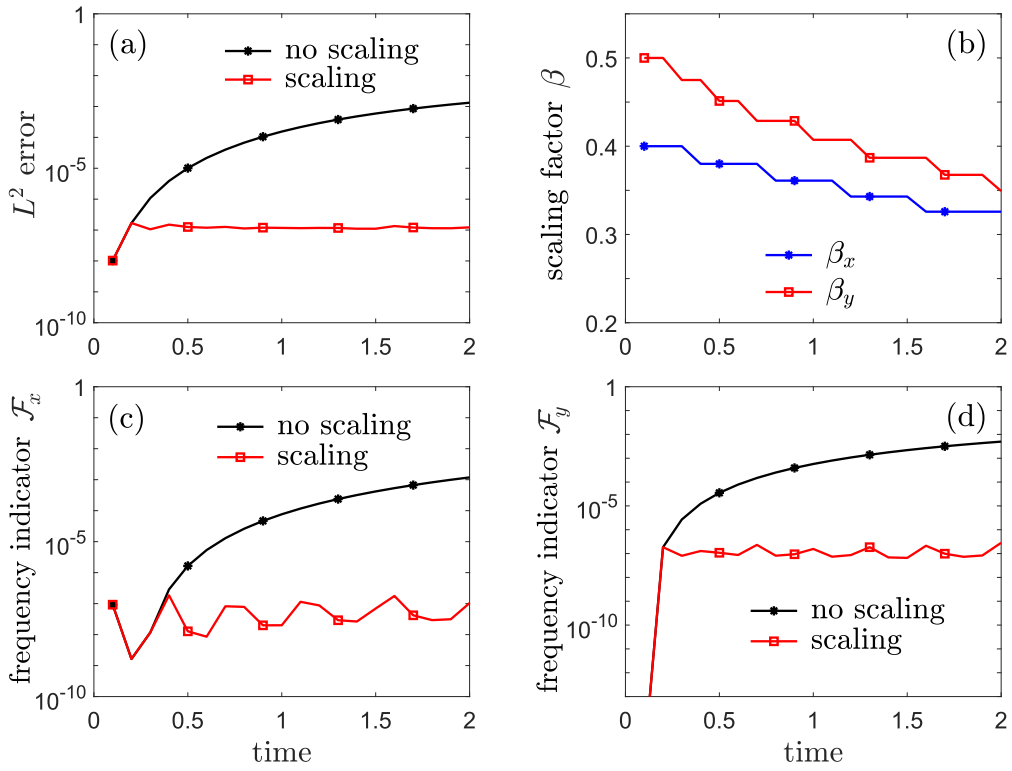

Figure 8.5: Example 27: Solving a 2D unbounded domain PDE (Eq. (8.3.9)). L^2 error, scaling factor, and frequency indicators associated with the numerical solution of equation (8.3.9) using s-PINNs, with and without dynamic scaling. (a) L^2 error as a function of time. The s-PINNs that are equipped with the scaling technique (red) achieve higher accuracy than those without (black). (b) The scaling factors β_x (blue) and β_y (red) as functions of time. Both scaling factors are decreased to match the spread of the solution in both the x and y directions. Scaling factors are adjusted to maintain small frequency indicators in the x-direction (c), and in the y-direction (d). In all computations, the timestep is $\Delta t = 0.1$.

Tuning the scaling factors β_x, β_y across different timesteps is achieved by monitoring the frequency indicators in the x- and y-directions, $\mathcal{F}_x$ and $\mathcal{F}_y$, as detailed in [XSC21b]. We use initial expansion orders $N_x = N_y = 8$ and scaling factors $\beta_x = 0.4, \beta_y = 0.5$. The ratio and threshold for adjusting the scaling factors are set to be $q = 0.95$ and $\nu^{-1} = 0.95$. The timestep $\Delta t = 0.1$ is used to adjust both scaling factors in both dimensions in a timely manner and a fourth-order implicit Runge–Kutta scheme is used for numerical integration. The neural network that we use to learn $w_{i,\ell}(t)$ has 5 intermediate layers with 150 neurons in each layer.

The results depicted in Fig. 8.5(a) show that an s-PINN using the scaling technique can achieve high accuracy by using high-order Runge–Kutta schemes in minimizing the SSE Eq. (8.3.3) and by properly adjusting β_x and β_y (shown in Fig. 8.5(b)) to control the frequency indicators $\mathcal{F}_x$ and $\mathcal{F}_y$ (shown in Fig. 8.5(c) and (d)). The s-PINNs can be extended to higher spatial dimensions by calculating the numerical solution expressed in the tensor product form as in Eq. (8.3.11).

Since our method outputs spectral expansion coefficients, using the full tensor product in the spatial spectral decomposition leads to a number of outputs that increase exponentially with dimensionality. The very wide neural networks needed for such high-dimensional problems result in less efficient training. However, unlike other recent machine–learning–based PDE solvers or PDE learning methods [BWW21, LKA20] that explicitly rely on a spatial discretization of grids or meshes, the curse of dimensionality can be partially mitigated in our s-PINN method. By using a hyperbolic cross space [SW10b], we can effectively reduce the number of coefficients needed to accurately reconstruct the numerical solution. In the next example, we solve a 3D parabolic spatiotemporal PDE, similar to that in Example 27, but we demonstrate how implementing a hyperbolic cross space can reduce the number of outputs and boost training efficiency.

Example 28.: Solving 3D unbounded domain PDEs

Consider the (3+1)-dimensional heat equation

$$\partial_t u(x,y,z,t) = \Delta u(x,y,z,t), \quad u(x,y,0) = \frac{1}{\sqrt{6}} e^{-x^2/12 - y^2/8 - z^2/4}, \tag{8.3.12}$$

which admits the analytical solution

$$u(x,y,z,t) = \frac{1}{\sqrt{(t+3)(t+2)(t+1)}} \exp\left[-\frac{x^2}{4(t+3)} - \frac{y^2}{4(t+2)} - \frac{z^2}{4(t+1)}\right] \tag{8.3.13}$$

for $(x,y,z) \in \mathbb{R}^3$. If we use the full tensor product of spectral expansions with expansion orders $N_x = N_y = N_z = 9$, we will need to output $10^3 = 1000$ expansion coefficients, and in turn, a relatively wide neural network with many parameters will be needed to generate the corresponding weights as shown in Fig. 8.1(c). Training such wide networks can be inefficient. However, many of the spectral expansion coefficients are close to zero and can be eliminated without compromising accuracy. One way to select expansion coefficients is to use the hyperbolic cross space technique [SW10b] to output coefficients of the generalized Hermite basis functions only in the space

$$\begin{aligned} V_{N,\gamma_\times}^{\vec{\beta},\vec{x}_0} &\coloneqq \operatorname{span}\Big\{\hat{\mathcal{H}}_{N_x}(\beta_x x)\hat{\mathcal{H}}_{N_y}(\beta_y y)\hat{\mathcal{H}}_{N_z}(\beta_z z) : |\vec{N}|_{\text{mix}} \|\vec{N}\|_\infty^{-\gamma_\times} \le N^{1-\gamma_\times}\Big\}, \\ \vec{N} &\coloneqq (N_x, N_y, N_z), \quad |\vec{N}|_{\text{mix}} \coloneqq \max\{N_x,1\}\max\{N_y,1\}\max\{N_z,1\}, \end{aligned} \tag{8.3.14}$$

where the hyperbolic space index $\gamma_\times \in (-\infty, 1)$. Taking $\gamma_\times = -\infty$ in Eq. (8.3.14) corresponds to the full tensor product with $N+1$ basis functions in each dimension. $\beta_x, \beta_y, \beta_x$ are the scaling factors for the basis functions in the x, y, z directions, and N_x, N_y, N_z are the orders of the basis function expansions in the x, y, z directions. For fixed N in Eqs. (8.3.14), the number of total basis function tends to decrease with increasing $\gamma_\times$. We set $N = 9$ in Eq. (8.3.14) and use the initial scaling factors $\beta_x = 0.4, \beta_y = 0.5, \beta_z = 0.7$. Using a fourth-order implicit Runge–Kutta scheme with a timestep $\Delta t = 0.2$, we set the ratio and threshold for adjusting the scaling factors are set to $q = 0.95$ and $\nu^{-1} = 0.95$ in each dimension.

H \ $\gamma_\times$	$-\infty$	-1	0	$\frac{1}{2}$
200	2.217×10^{-3}, (22911)	1.651×10^{-4}, (4309)	5.356×10^{-5}, (2886)	3.173×10^{-4}, (3956)
400	1.072×10^{-3}, (26725)	2.970×10^{-5}, (7014)	5.356×10^{-5}, (3309)	3.173×10^{-4}, (2356)
700	2.276×10^{-3}, (43923)	2.900×10^{-5}, (3133)	5.356×10^{-5}, (3229)	3.173×10^{-4}, (2098)
1000	7.871×10^{-5}, (55880)	2.901×10^{-5}, (3002)	5.356×10^{-5}, (2016)	3.173×10^{-4}, (1894)

Table 8.2: Example 28: Applying hyperbolic cross space and s-PINNs to the (3+1) dimensional PDE equation (8.3.12). Applying the hyperbolic cross space (equation (8.3.14)), we record the L^2 error as well as the training time (in seconds). The number of coefficients (outputs in the neural network) for $\gamma_\times = -\infty, -1, 0, \frac{1}{2}$ are $1000, 205, 141, 110$, respectively. Using $\gamma_\times = -1$ or 0 leads to the most accurate results. The training time tends to increase with the number of outputs (a smaller $\gamma_\times$ corresponds to more outputs). By comparing the results in different rows for the same column, it can be seen that more outputs require a wide neural network for training.

To illustrate the potential numerical difficulties arising from outputting large numbers of coefficients when solving higher-dimensional spatiotemporal PDEs, we use a neural network with two hidden layers and different numbers of neurons in the intermediate layers. We also adjust $\gamma_\times$ to explore how decreasing the number of coefficients can improve training efficiency. Our results are listed in Table 8.2.

The results shown in Table 8.2 indicate that, compared to using the full tensor product $\gamma_\times = -\infty$, implementing the hyperbolic cross space with a moderate $\gamma_\times = -1$ or 0, the total number of outputs is significantly reduced, leading to faster training and better accuracy. However, increasing the hyperbolicity to $\gamma_\times = \frac{1}{2}$, the error increases relative to using $\gamma_\times = -1, 0$ because some useful, nonzero coefficients are excluded. Also, comparing the results across different rows, wider layers lead to both more accurate results and faster training speed. The sensitivity of our s-PINN method to the number of intermediate layers in the neural network and the number of neurons in each layer are further discussed in Example 30. Overall, in higher-dimensional problems, there is a balance between computational cost and accuracy as the number of outputs needed will grow fast with dimensionality. Spectrally-adapted PINNs can easily incorporate a hyperbolic cross space so that the total number

of outputs can be reduced to a manageable number for moderate-dimensional problems. Finding the optimal hyperbolicity index $\gamma_\times$ for the cross space Eq. (8.3.14) will be problem-specific.

In the next example, we explore how s-PINNs can be used to solve the Schrödinger's equation in $x \in \mathbb{R}$. Solving this complex-valued equation poses substantial numerical difficulties as the solution exhibits diffusive, oscillatory, and convective behavior [LZZ18].

Example 29.: Solving an unbounded domain Schrödinger equation
We seek to numerically solve the following Schrödinger equation defined on $x \in \mathbb{R}$

$$\mathrm{i}\partial_t \psi(x,t) = -\partial_x^2 \psi(x,t), \quad \psi(x,0) = \frac{1}{\sqrt{\zeta}} \exp\left[\mathrm{i}kx - \frac{x^2}{4\zeta}\right]. \tag{8.3.15}$$

For reference, Eq. (8.3.15) admits the analytical solution

$$\psi(x,t) = \frac{1}{\sqrt{\zeta + \mathrm{i}t}} \exp\left[\mathrm{i}k(x-kt) - \frac{(x-2kt)^2}{4(\zeta+\mathrm{i}t)}\right]. \tag{8.3.16}$$

As in Example 27, we shall numerically solve Eq. (8.3.15) in the weak form

$$(\partial_t \Psi(x,t), v) + \mathrm{i}(\partial_x \Psi(x,t), \partial_x v) = 0, \quad \forall v \in H^1(\mathbb{R}). \tag{8.3.17}$$

Since the solution to Eq. (8.3.15) decays as $\sim \exp[-x^2/(4\sqrt{(\zeta^2+t^2)})]$ at infinity, we shall use the generalized Hermite functions as basis functions. The solution is rightward-translating for $k > 0$ and increasingly oscillatory and spread out over time. Hence, as detailed in [XSC21b], we apply three additional adaptive spectral techniques to improve efficiency and accuracy: (i) a scaling technique to adjust the scaling factor β over time in order to capture diffusive behavior, (ii) a moving technique to adjust the center of the basis function x_L to capture convective behavior, and (iii) a p-adaptive technique to increase the number of basis functions N to better capture the oscillations. We set the initial parameters $\beta = 0.8, x_L = 0, N = 24$ at $t = 0$. The scaling factor adjustment ratio and the threshold for adjusting the scaling factor are $q = \nu^{-1} = 0.95$, the minimum and maximum change in displacements of the basis

functions are 0.004 and 0.1 within each timestep, respectively, and the threshold for moving is 1.001. Finally, the thresholds of the p-adaptive technique are set to $\rho = \rho_0 = 2$ and $\gamma = 1.4$.

Generally speaking, it is desirable to set the adaptive spectral method scaling hyperparameters to $\nu \gtrsim 1 \gtrsim q$. When implementing adaptive moving, it is desirable to make the change in the basis functions' displacement as accurate as possible by setting a small minimum change in displacement per timestep, a large maximum change in displacement per timestep, and a threshold for moving which is slightly larger than 1. For the p-adaptive technique that adjusts the spectral expansion order, there is a cost-accuracy tradeoff; setting ρ and γ to small values but ρ_0 to a large value leads to the smallest errors but higher computational costs. A more detailed and theoretical discussion of how the choices of those hyperparameters influence the results is given in [CSX23].

To numerically solve Eq. (8.3.17), a fourth-order implicit Runge–Kutta scheme is applied to advance time with timestep $\Delta t = 0.1$. The neural network underlying the s-PINN that we use in this example contains 13 layers with 100 neurons in each layer. Figure 8.6(a) shows that the s-PINN with adaptive spectral techniques leads to very high accuracy as it can properly adjust the basis functions over a longer timescale (across different timesteps), while not adapting the basis functions results in larger errors. Figs. 8.6(b–d) show that the scaling factor β decreases over time to match the spread of the solution, the displacement of the basis function x_L increases in time to capture the rightward movement of the basis functions, and the expansion order N increases to capture the solution's increasing oscillatory behavior. Our results indicate that our s-PINN method can effectively utilize all three adaptive algorithms.

We now explore how the timestep and the order of the implicit Runge–Kutta method affect the approximation error, *i.e.*, to what extent can we relax the constraint on the timestep and maintain the accuracy of the basis functions, or, if higher-order Runge–Kutta schemes are better. Another feature to explore is the neural network structure, such as the number of layers and neurons per layer, and how it affects the performance of s-PINNs. In the following example, we carry out a sensitivity analysis.

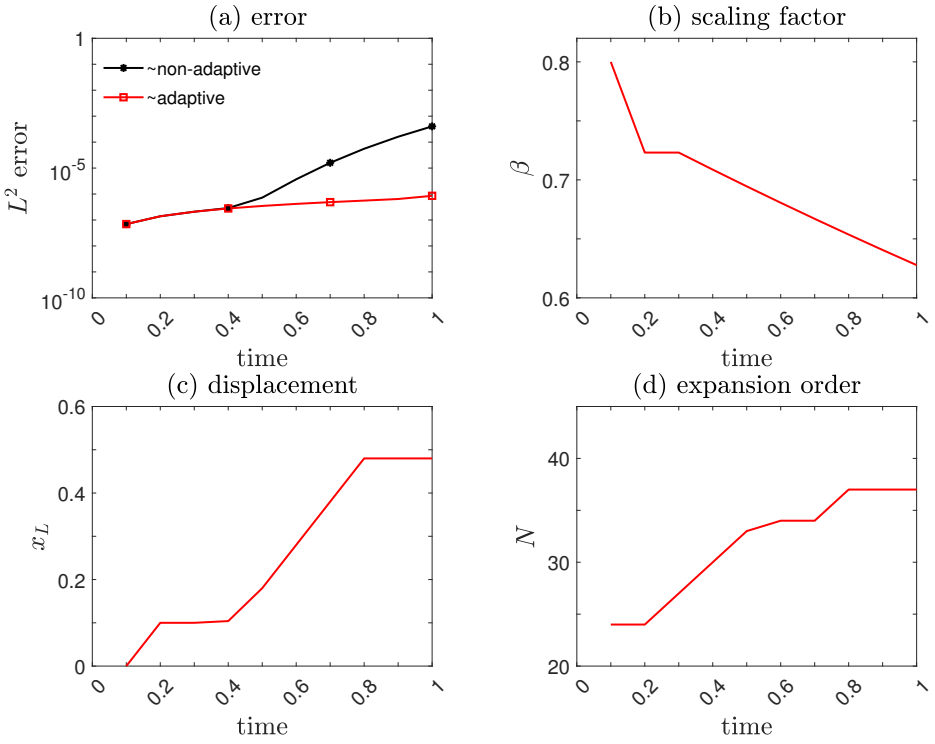

Figure 8.6: Example 29: Solving the Schrödinger equation (Eq. (8.3.15)) in an unbounded domain. Approximation error, scaling factor, displacement, and expansion order associated with the numerical solution of Eq. (8.3.15) using adaptive (red) and non-adaptive (black) s-PINNs. (a) Errors for numerically solving Eq. (8.3.15) with and without adaptive techniques. (b) The change in the scaling factor which decreases over time as the solution becomes more spread out. (c) The displacement of the basis functions x_L which is increased as the solution moves rightwards. (d) The expansion order N increases over time as the solution becomes more oscillatory. A timestep $\Delta t = 0.1$ was used.

Example 30.: Sensitivity analysis of s-PINN

To explore how the performance of an s-PINN depends on algorithmic set-up and parameters, we apply it to solving the heat equation defined on $x \in \mathbb{R}$,

$$\partial_t u(x,t) = \partial_x^2 u(x,t) + f(x,t), \quad u(x,0) = e^{-x^2/4} \sin x \tag{8.3.18}$$

using generalized Hermite functions as basis functions. For the source $f(x,t) = [x\cos x + (t+1)\sin x]\,(t+1)^{-3/2}\,\exp[-\frac{x^2}{4(t+1)}]$, Eq. (8.3.18) admits the analytical solution

$$u(x,t) = \frac{\sin x}{\sqrt{t+1}} \exp\left[-\frac{x^2}{4(t+1)}\right]. \tag{8.3.19}$$

We solve Eq. (8.3.18) in the weak form by multiplying any test function $v \in H^1(\mathbb{R})$ on both sides and integrating by parts to obtain

$$(\partial_t u, v) = -(\partial_x u, \partial_x v) + (f, v), \ \ \forall v \in H^1(\mathbb{R}). \tag{8.3.20}$$

The solution diffusively spreads over time, requiring one to decrease the scaling factor β of the generalized Hermite functions $\{\hat{\mathcal{H}}_i^\beta(x)\}$. We shall first study how the timestep and the order of the implicit Runge–Kutta method associated with solving the minimization problem (8.3.3) affect our results. We use a neural network with five intermediate layers and 200 neurons per layer, and set the learning rate $\eta = 5 \times 10^{-4}$. The initial scaling factor is set to $\beta = 0.8$. The scaling factor adjustment ratio and threshold are set to $q = 0.98$, and $\nu = q^{-1}$, respectively. For comparison, we also apply a Crank-Nicolson scheme for numerically solving Eq. (8.3.20), *i.e.*,

$$\frac{U_N^\beta(t_{j+1}) - U_N^\beta(t_j)}{\Delta t} = D_N^\beta \frac{\left[U_N^\beta(t_{j+1}) + U_N^\beta(t_j)\right]}{2} + \frac{F_N^\beta(t_{j+1}) + F_N^\beta(t_j)}{2}. \tag{8.3.21}$$

where $U_N^\beta(t), F_N^\beta(t)$ are the $N+1$-dimensional vectors of spectral expansion coefficients of the numerical solution and of the source, respectively. $D_N^\beta \in \mathbb{R}^{(N+1)\times(N+1)}$ is the tridiagonal block matrix representing the discretized Laplacian operator ∂_x^2:

$$D_{i,i-2} = \beta^2 \frac{\sqrt{(i-2)(i-1)}}{2}, \ \ D_{i,i} = -\beta^2\Big(i - \frac{1}{2}\Big), \ \ D_{i,i+2} = \beta^2 \frac{\sqrt{i(i+1)}}{2},$$

and $D_{i,j} = 0$, otherwise.

Table 8.3 shows that since the error from temporal discretization Δt^{2K} is already quite small for $K \geq 4$, using a higher-order Runge–Kutta method does not significantly improve accuracy for all choices of Δt. Using higher-order ($K \geq 4$) schemes tends to require longer run times. Higher orders require fitting over more data points (using the same number of parameters) leading to slower convergence when minimizing Eq. (8.3.3), which can result in larger errors. Compared to the second-order Crank-Nicolson scheme, whose error is $O(\Delta t^2)$,

Δt \ K	C-N scheme	2	4	6	10
0.02	12, 8.252e-08, 0.545	27, 4.011e-08, 0.545	*54, 1.368e-08, 0.545*	279, 2.545e-07, 0.545	7071, 6.358e-05, 0.695
0.05	5, 5.157e-05, 0.545	12, 2.799e-08, 0.545	23, 1.651e-08, 0.545	105, 2.566e-07, 0.545	3172, 1.052e-06, 0.545
0.1	3, 2.239e-04, 0.695	6, 1.331e-06, 0.695	10, 1.314e-06, 0.695	72, 1.346e-06, 0.695	1788, 2.782e-06, 0.695
0.2	2, 9.308e-04, 0.695	**3, 3.760e-06, 0.695**	9, 2.087e-06, 0.695	317, 2.107e-06, 0.695	1310, 1.925e-03, 0.753

Table 8.3: Example 30: Sensitivity analysis of s-PINN. Computational runtime (in seconds), error, and the final scaling factor for different timesteps Δt, different implicit order-K Runge–Kutta schemes, and the traditional Crank-Nicolson scheme. In each box, the run time (in seconds) and the SSE are listed, with the final scaling factor given just below. The results associated with the smallest error are highlighted in italics while the results associated with the shortest run time for our s-PINN method are indicated in bold.

the errors of our s-PINN method do not grow significantly when Δt increases. In fact, the accuracy using the smallest timestep $\Delta t = 0.02$ in the Crank-Nicolson scheme was still inferior to that of the s-PINN method using the second order or fourth order Runge-Kutta scheme with $\Delta t = 0.2$. Moreover, the run time of our s-PINN method using a second or fourth-order implicit Runge–Kutta scheme for the loss function is not significantly larger than that of the Crank-Nicolson scheme. Thus, compared to traditional spectral methods for numerically solving PDEs, our s-PINN method, even when incorporating some lower-order Runge–Kutta schemes, can greatly improve accuracy without significantly increasing computational cost.

In Table 8.3, the smallest run time of our s-PINN method, which occurs for $K = 2, \Delta t = 0.2$, is shown in blue. The smallest error case, which arises for $K = 4, \Delta t = 0.02$, is shown in red. The run time always increases with the order K of the implicit Runge–Kutta scheme and always decreases with Δt due to fewer timesteps. Additionally, the error always increases with Δt regardless of the order of the Runge–Kutta scheme. However, the expected convergence order is not observed, implying that the increase in error results from increased lag in the adjustment of the scaling factor β when Δt is too large, rather than from an insufficiently small time discretization error Δt^{2K}. Using a fourth-order implicit Runge–Kutta scheme with $\Delta t = 0.05$ to solve Eq. (8.3.20) seems to both achieve high accuracy and avoid large computational costs.

H \ N_H	3	5	8	13
50	1348(0.0014), 6.317e-04, 0.74	798(0.0015), 9.984e-05, 0.70	995(0.0020), 1.891e-04, 0.58	778(0.0039), 4.022e-04, 0.70
80	784(0.0015), 7.164e-04, 0.65	234(0.0016), 1.349e-06, 0.70	216(0.0023), 1.345e-06, 0.70	376(0.0043), 1.982e-06, 0.70
100	1080(0.0018), 8.804e-05, 0.70	*114(0.0017), 1.344e-06, 0.70*	102(0.0024), 1.346e-06, 0.70	145(0.0043), 1.348e-06, 0.70
200	219(0.0022), 1.349e-06, 0.70	72(0.0035), 1.346e-06, 0.70	**43(0.0048), 1.347e-06, 0.70**	64(0.0057), 1.345e-06, 0.70

Table 8.4: Example 30: Sensitivity analysis of our s-PINN for different numbers of intermediate layers N_H and neurons per layer H. The first line gives the total computational runtime (seconds) and the runtime per epoch (in parentheses), while the second line lists the SSE (equation ((8.3.3))) and the final scaling factor. Results associated with the smallest error are marked in italics while those associated with the shortest run time are highlighted in bold.

We also investigate how the total number of parameters in the neural network and the structure of the network affect efficiency and accuracy. We use a sixth-order implicit Runge–Kutta scheme with $\Delta t = 0.1$. The learning rate is set to $\eta = 5 \times 10^{-4}$ for all neural networks.

As shown in Table 8.4, the computational cost tends to decrease with the number of neurons H in each layer as it takes fewer epochs to converge when minimizing Eq. (8.3.3). The run time tends to decrease with N_H due to a faster convergence rate, until about $N_H = 8$. The errors when $H = 50$ are significantly larger as the training terminates (after a maximum of 100000 epochs) before it converges. For $N_H = 3$, the corresponding s-PINN always fails to achieve accuracy within 100000 epochs unless $H \gtrsim 200$. Actually, the mean run time for training one epoch increases with H, N_H. Nonetheless, a neural network with 8 intermediate layers and 200 neurons in each layer performs the best with the smallest total run time. Therefore, overparametrization is indeed helpful in improving the neural network's performance, leading to faster convergence rates, in contrast to most traditional optimization methods that take longer to converge with more parameters. Similar observations have been made in other optimization tasks that involve deep neural networks [ACH18, CCZ20]. Consequently, our s-PINN method retains the advantages of deep and wide neural networks for improving accuracy and efficiency.

8.4 Parameter Inference and Source Reconstruction

As with standard PINN approaches, s-PINNs can also be used for parameter inference in PDE models or reconstructing unknown sources in a physical model. Assuming observational data at uniform time intervals $t_j = j\Delta t$ associated with a partially known underlying PDE model, s-PINNs can be trained to infer model parameters θ by minimizing the sum of squared errors, weighted from both ends of the time interval (t_j, t_{j+1}),

$$\mathrm{SSE}_j = \mathrm{SSE}_j^{\mathrm{L}} + \mathrm{S}SE_j^{\mathrm{R}}, \tag{8.4.1}$$

where

$$\begin{aligned}
\mathrm{SSE}_j^{\mathrm{L}} = \sum_{s=1}^{K} \Big\| & u(x, t_j + c_s\Delta t; \theta_{j+1}; \Theta_{j+1}) - u(x, t_j; \theta_j) \\
& - \sum_{r=1}^{K} a_{sr} \mathcal{M}\big[u(x, t_j + c_r\Delta t; \theta_{j+1}; \Theta_{j+1})\big] \Big\|_2^2, \\
\mathrm{SSE}_j^{\mathrm{R}} = \sum_{s=1}^{K} \Big\| & u(x, t_j + c_s\Delta t; \theta_{j+1}; \Theta_{j+1}) - u(x, t_{j+1}; \theta_{j+1}) \\
& - \sum_{r=1}^{K} (a_{sr} - b_r) \mathcal{M}\big[u(x, t_j + c_r\Delta t; \theta_{j+1}; \Theta_{j+1})\big] \Big\|_2^2.
\end{aligned}$$

Here, θ_{j+1} are the set of model parameters to be found using the sample points $c_s\Delta t$ between t_j and t_{j+1}. The most obvious advantage of s-PINNs over standard PINN methods is that they can deal with models defined on unbounded domains, extending PINN-based methods that are typically applied to finite domains. Note that the revised loss function Eq. (8.4.1) differs from Eq. (8.3.3) because now the solutions at t_j and t_{j+1} are both known, while for Eq. (8.3.3) the solution at t_{j+1} is to be solved.

Given observations over a certain time interval, one may wish to both infer parameters θ_j in the underlying physical model and reconstruct the solution u at any given time. Here, we provide an example in which both a parameter in the model is to be inferred and the numerical solution is to be obtained.

Example 31.: Parameter (diffusivity) inference

As a starting point for a parameter-inference problem, we consider diffusion with a source defined on $x \in \mathbb{R}$

$$\partial_t u(x,t) = \kappa \partial_x^2 u(x,t) + f(x,t), \quad u(x,0) = e^{-x^2/4} \sin x, \tag{8.4.2}$$

where the constant parameter κ is the thermal conductivity (or diffusion coefficient) in the entire domain. In this example, we set $\kappa = 2$ as a reference and assume the source

$$f(x,t) = \left[\frac{2\left(x\cos x + (t+1)\sin x\right)}{(t+1)^{3/2}} - \frac{x^2}{4(t+1)^2} + \frac{\sin x}{2(t+1)^{3/2}} \right] \exp\left[-\frac{x^2}{4(t+1)} \right]. \tag{8.4.3}$$

In this case, the analytical solution to Eq. (8.4.2) is given by Eq. (8.3.19). We numerically solve Eq. (8.4.2) in the weak form of Eq. (8.3.20). If the form of the spatiotemporal heat equation is known (such as Eq. (8.4.2)), but some parameters such as κ is unknown, reconstructing it from measurements is usually performed by defining and minimizing a loss function as was done in [Hun21]. It can also be shown that $\kappa = \kappa(t)$ in Eq. (8.4.2) can be uniquely determined by the observed solution $u(x,t)$ [Iva93, Jon62, Bez74] under certain conditions. Here, however, we assume that observations are taken at discrete time points $t_j = j\Delta t$ and seek to reconstruct both the parameter κ and the numerical solution at $t_j + c_s \Delta t$ (defined in Eqs. (8.4.2)) by minimizing Eq. (8.4.1). We use a neural network with 13 layers and 100 neurons per layer with a sixth-order implicit Runge–Kutta scheme. The timestep Δt is 0.1. At each timestep, we draw the function values from

$$u(x,t_j) = \frac{\sin x}{\sqrt{t_j+1}} \exp\left[-\frac{x^2}{4(t_j+1)} \right] + \xi(x,t_j), \tag{8.4.4}$$

where $\xi(x,t)$ is the noise term that is both spatially and temporally uncorrelated, and $\xi(x,t) \sim \mathcal{N}(0,\sigma^2)$, where $\mathcal{N}(0,\sigma^2)$ is the normal distribution of mean 0 and variance σ^2 (*i.e.*, $\langle \xi(x,t)\xi(y,s)\rangle = \sigma^2 \delta_{x,y}\delta_{s,t}$). For different levels of noise σ, we take one trajectory of the measured solution with noise $u(x,t_j)$ to reconstruct the parameter κ, which is presumed to be a constant in $[t_j, t_{j+1})$, and simultaneously obtain the numerical solutions at the in-

termediate time points $t_j + c_s\Delta t$. We are interested in how different levels of noise and the increasing spread of the solution will affect the SSE and the reconstructed parameter $\hat{\kappa}$. Figure 8.7 shows the deviation of the reconstructed $\hat{\kappa}$ from its true value, $|\hat{\kappa} - 2|$, the SSE, the scaling factor, and the frequency indicator as functions of time for different noise levels. Figure 8.7(a) shows that the larger the noise, the less accurate the reconstructed

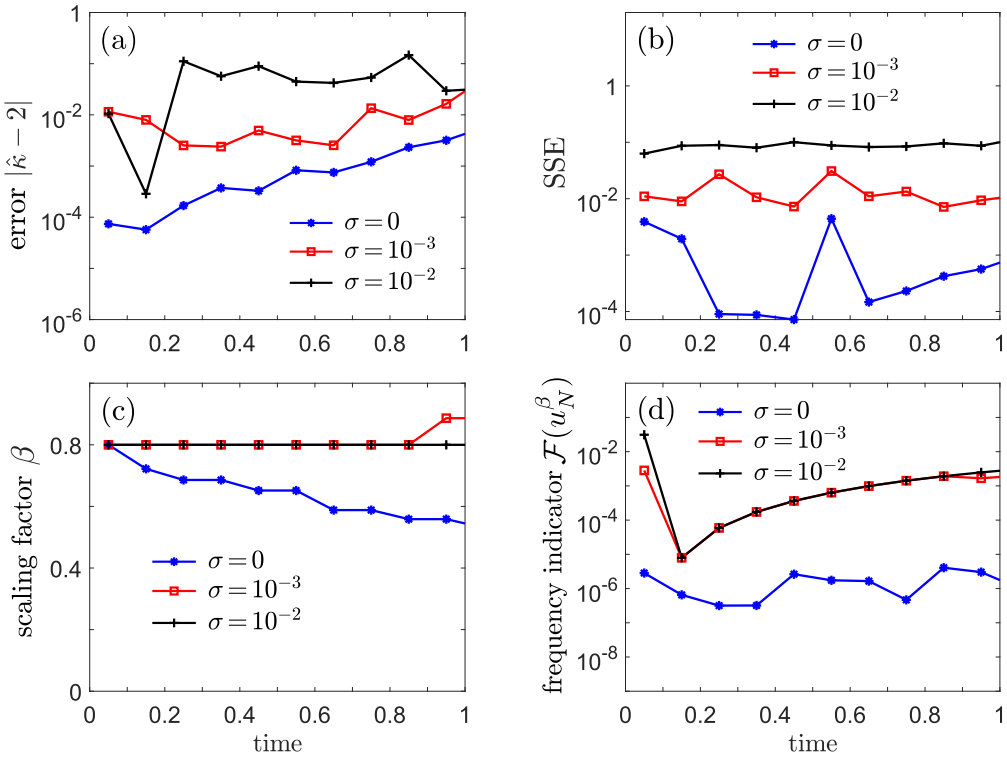

Figure 8.7: Example 31: Parameter (diffusivity) inference. The parameter κ inferred within successive time windows of $\Delta t = 0.1$, the SSE error Eq. (8.4.1), the scaling factor, and the frequency indicators associated with solving Eq. (8.4.2), for different noise levels σ. Here, the SSE was minimized to find the estimate $\hat{\theta} \equiv \hat{\kappa}$ and the solutions u_N at intermediate timesteps $t_j + c_s\Delta t$. (a, b) Smaller σ leads to smaller SSE Eq. (8.4.2) and a more accurate reconstruction of $\hat{\kappa}$. When the function has spread out significantly at long times, the reconstructed $\hat{\kappa}$ becomes less accurate, suggesting that unboundedness and small function values render the problem susceptible to numerical difficulties. (c, d) Noisy data results in a larger proportion of high-frequency waves and thus a large frequency indicator, impeding proper scaling.

κ. Moreover, as the function becomes more spread out (when $\sigma = 0$), the error in both the reconstructed diffusivity and the SSE increases across time, as shown in Fig. 8.7(b). This behavior suggests that a diffusive solution that decays more slowly at infinity can give rise to inaccuracies in the numerical computation of the intermediate timestep solutions and in

reconstructing model parameters. Finally, as indicated in Fig. 8.7(c,d), larger variances in the noise will impede the scaling process since the frequency indicator cannot be as easily controlled because larger variances in the noise usually correspond to high-frequency and oscillatory components of a solution.

In Example 31, both the parameter and the unknown solution were inferred. Apart from reconstructing the coefficients in a given physical model, in certain applications, we may also wish to reconstruct the underlying physical model by inferring, *e.g.*, the heat source $f(x,t)$. Source recovery from observational data commonly arises and has been the subject of many previous studies [YYF09, YDY11, YF10]. We now discuss how the s-PINN methods presented here can also be used for this purpose. For example, in Eq. (8.3.18) or Eq. (8.4.2), we may wish to reconstruct an unknown source $f(x,t)$ by also approximating it with a spectral decomposition

$$f(x,t) \approx f_N(x,t) = \sum_{i=0}^{N} h_i(t)\phi^{\beta}_{i,x_L}(x), \tag{8.4.5}$$

and minimizing an SSE that is augmented by a penalty on the coefficients $h_i, i = 0,\ldots,N$. We learn the expansion coefficients h_i within $[t_j, t_{j+1}]$ by minimizing

$$\mathrm{SSE}_j = \; \mathrm{S}SE_j^{\mathrm{L}} + \mathrm{S}SE_j^{\mathrm{R}} + \lambda \textstyle\sum_{s=1}^{K} \left\|\mathbf{h}_N(t_j + c_s\Delta t;\Theta_{j+1})\right\|_2^2, \quad \lambda \geq 0,$$

$$\begin{aligned}
\mathrm{SSE}_j^{\mathrm{L}} = \sum_{s=1}^{K} \Big\| & u(x,t_j + c_s\Delta t) - u(x,t_j) \\
& - \sum_{r=1}^{K} a_{sr}\big[\partial_{xx}u(x,t_j + c_r\Delta t) + f_N(x,t_j + c_r\Delta t;\Theta_{j+1})\big]\Big\|_2^2, \\
\mathrm{SSE}_j^{\mathrm{R}} = \sum_{s=1}^{K} \Big\| & u(x,t_j + c_s\Delta t) - u(x,t_{j+1}) \\
& - \sum_{r=1}^{K} (a_{sr} - b_r)\big[\partial_{xx}u(x,t_j + c_r\Delta t) + f_N(x,t_j + c_r\Delta t;\Theta_{j+1})\big]\Big\|_2^2,
\end{aligned}$$

where $\mathbf{h}_N(t_j + c_s\Delta t; \Theta_{j+1}) \equiv \big(h_1(t_j + c_s\Delta t; \Theta_{j+1}), \ldots, h_N(t_j + c_s\Delta t; \Theta_{j+1})\big)$ and u (or the spectral expansion coefficients w_i of u) is assumed known at all intermediate time points $c_s\Delta t$ in (t_j, t_{j+1}).

The last term in Eq. (8.4.6) adds an L^2 penalty term on the coefficients of f which tends to reconstruct smoother and smaller-magnitude sources as λ is increased. Other forms of regularization such as L^1 can also be considered [WT19]. In the presence of noise, an L^1 regularization further drives small expansion weights to zero, yielding an inferred source f_N described by fewer nonzero weights.

Since the reconstructed heat source f_N is expressed in terms of a spectral expansion in Eq. (8.4.5), and minimizing the loss function Eq. (8.4.6) depends on the global information of the observation u, f at any location x also contains global information intrinsic to u. In other words, for such inverse problems, the s-PINN approach extracts global spatial information and is thus able to reconstruct global quantities. We consider an explicit case in the next example.

Example 32.: Source recovery

Consider the canonical source reconstruction problem [Can68, JL07, HP14] of finding $f(x,t)$ in the heat equation model in Eq. (8.3.18) for which observational data are given by Eq. (8.4.4) but evaluated at $t_j + c_s\Delta t$. A physical interpretation of the reconstruction problem is identifying the heat source $f(x,t)$ using measurement data in conjunction with Eq. (8.3.18). As in Example 5, we numerically solve the weak form Eq. (8.3.20). To study how the L^2 penalty term in Eq. (8.4.6) affects source recovery and whether increasing the regularization λ will make the inference of f more robust against noise, we minimize Eq. (8.4.1) for different values of λ and σ.

We use a neural network with 13 layers and 100 neurons per layer to reconstruct $f_i(t)$ in the decomposition Eq. (8.4.5) with $N = 16$, *i.e.*, the neural network outputs the coefficients h_i at the intermediate timesteps $t_j + c_s\Delta t$. The basis functions $\phi^{\beta}_{i,x_L}(x)$ are chosen to be Hermite functions $\hat{\mathcal{H}}^{\beta}_{i,x_L}(x)$. For simplicity, we consider the problem only at times within the first time interval $[0, 0.2]$ and a fixed scaling factor $\beta = 0.8$ as well as a fixed displacement

σ \ λ	0	10^{-3}	10^{-2}	10^{-1}
0	0.1370, (1.543×10^{-8})	0.1370, (1.368×10^{-5})	0.1477, (0.00132)	0.3228, (0.0888)
10^{-3}	0.1821, (2.837×10^{-6})	0.1818, (2.736×10^{-5})	0.1702, (1.387×10^{-3})	0.3222, (0.08964)
10^{-2}	1.0497, (0.001517)	1.0383, (1.579×10^{-3})	0.8031, (6.078×10^{-3})	0.3434, (0.1168)
10^{-1}	11.505, (0.2976)	11.458, (0.3032)	8.2961, (0.6905)	1.3018, (2.9330)

Table 8.5: The error SSE_0 from Eq. (8.4.2) and the error of the reconstructed source Eq. (8.4.6) (in parentheses), under different strengths of data noise and regularization coefficients λ.

$x_L = 0$.

In Table 8.5, we record the L^2 error

$$\left\| f(x,t) - \sum_{i=0}^{16} h_i(t;\Theta)\hat{\mathcal{H}}^{\beta}_{i,x_L}(x) \right\|_2 \tag{8.4.6}$$

the lower-left of each entry and the SSE_0 in the upper-right. Observe that as the variance of the noise increases, the reconstruction of f via the spectral expansion becomes increasingly inaccurate. In the noise-free case, taking $\lambda = 0$ in equation (8.4.6) achieves the smallest

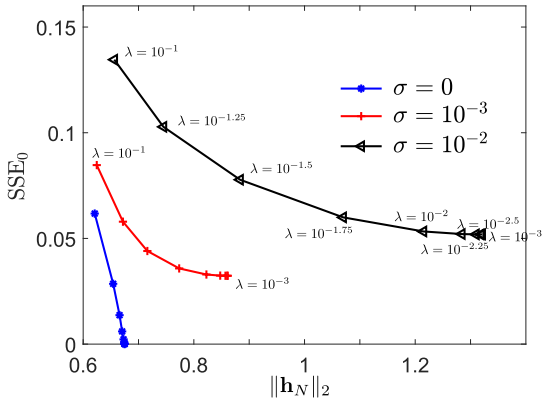

Figure 8.8: Example 32: Source recovery. SSE_0 plotted against the reconstructed heat source $\|\mathbf{h}_N\|_2$ as given by equation (8.4.6), as a function of λ for various values of σ (an "L-curve"). When λ is large, the norm of the reconstructed heat source $\|\mathbf{h}_N\|_2$ always tends to decrease while the "error" SSE_0 tends to increase. When $\lambda = 10^{-1}$, $\|\mathbf{h}_N\|_2$ is small and the SSE_0 is large. A moderate $\lambda \in [10^{-2}, 10^{-3}]$ could reduce the error SSE_0, compared to using a large λ, while also generating a heat source with smaller $\|\mathbf{h}_N\|_2$.

SSE_0 and the smallest reconstruction error. However, with increasing noise σ, using an L^2 regularization term in Eqs. (8.4.6) can prevent over-fitting of the data although SSE_0 increases with the regularization strength λ. When $\sigma = 10^{-3}$, taking $\lambda = 10^{-2}$ achieves the smallest reconstruction error Eq. (8.4.6); when $\sigma = 10^{-2}, 10^{-1}$, $\lambda = 10^{-1}$ achieves the smallest reconstruction error. However, if λ is too large, coefficients of the spectral approximation to f are pushed to zero. Thus, it is important to choose an intermediate λ so that the reconstruction of the source is robust to noise. In Fig. 8.8, we plot the norm of the reconstructed heat source $\|\mathbf{h}_N\|_2$ and the "error" SSE_0 which varies as λ changes for different σ.

8.5 Summary and Conclusions

In this chapter, we propose an approach that blends standard PINN algorithms with adaptive spectral methods and show through examples that this hybrid approach can be applied to a wide variety of data-driven problems including function approximation, solving PDEs, parameter inference, and model selection. The underlying feature that we exploit is the physical differences across classes of data. For example, by understanding the difference between space and time variables in a PDE model, we can describe the spatial dependence in terms of basis functions, obviating the need to normalize spatial data. Thus, s-PINNs are ideal for solving problems in unbounded domains. The only additional "prior" needed is an assumption on the asymptotic spatial behavior and an appropriate choice of basis functions. Additionally, adaptive techniques have been recently developed to further improve efficiency and accuracy, making spectral decomposition especially suitable for unbounded-domain problems that the standard PINN cannot easily address.

We applied s-PINNs (exploiting adaptive spectral methods) across a number of examples and showed that they can outperform simple feed-forward neural networks for function approximation and existing PINNs for solving certain PDEs. Three major advantages are that s-PINNs can be applied to unbounded domain problems, more accurate by recovering spectral convergence in space, and more efficient as a result of faster evaluation of spatial

derivatives of all orders compared to standard PINNs that use autodifferentiation. These advantages are rooted in separated data structures, allowing for spectral computation and high-accuracy numerics. A straightforward implementation of s-PINNs retains most of the advantageous features of deep PINN architectures, making s-PINNs ideal for data-driven inference problems. However, in the context of solving higher-dimensional PDEs, a trade-off is necessary when using s-PINNs instead of PINNs. For s-PINNs, the network structure needs to be significantly widened to output an exponentially increasing (with dimensionality) number of expansion coefficients, while in standard PINNs, the network structure remains largely preserved but an exponentially larger number of trajectories are needed for sufficient training. We found that by restricting the spatial domain to a hyperbolic cross space, the number of outputs required for s-PINNs can be appreciably decreased for problems of moderate dimensions. While using a hyperbolic cross space cannot reduce the number of outputs sufficiently to allow s-PINNs to be effective for very high-dimensional problems, the standard PINNs approach to problems in very high dimensions could require an unattainable number of samples for sufficient training.

In Table 8.6, we compare the advantages and disadvantages of the standard PINN and s-PINN methods. Potential improvements and extensions include applying techniques for selecting basis functions that best characterize the expected underlying process and inferring forms of the underlying model PDEs [LLM18, Rai18]. While standard PINN methods deal with local information (*e.g.*, $\partial_x u, \partial_x^2 u$), spectral decompositions capture global information making them a natural choice for also efficiently learning and approximating nonlocal terms such as convolutions and integral kernels. Potential future extensions of our s-PINN method may include adapting it to solve higher-dimensional problems by more systematically choosing a proper hyperbolic space or using other coefficient-reducing techniques, as well as using wavelets as activation functions [UGA23] to solve nonlinear differential equations. Also, recent Gaussian–process–based smoothing techniques [BMA23] can be considered to improve the robustness of our s-PINN method against noise/errors in measurements, and noise-aware physics-informed machine learning techniques [TMN22] can be incorporated when applying

our s-PINN for inverse-type PDE discovery problems. Finally, one can incorporate a recently proposed Bayesian-PINN (B-PINN) [YMK21] method into our s-PINN method to quantify uncertainty when solving inverse problems under noisy data.

Methods \ Solvers	Traditional	PINN
Non-spectral	+ leverages existing numerical methods + low-order FD/FE schemes easily implemented + efficient evaluation of function and derivatives -- mainly restricted to bounded domains -- complicated time-extrapolation -- complicated implementation of higher-order schemes -- algebraic convergence, less accurate -- more complicated inverse-type problems -- more complicated temporal and spatial extrapolation -- requires understanding of the problem to choose suitable discretization	+ easy implementation + efficient deep-neural-network training + easy extrapolation + easily handles inverse-type problems -- mainly restricted to bounded domains -- less accurate -- less interpretable spatial derivatives -- limited control of spatial discretization -- expensive evaluation of neural networks -- incompatible with existing numerical methods
Spectral	+ suitable for bounded and unbounded domains + spectral convergence in space, more accurate + leverage existing numerical methods + efficient evaluation of function and derivatives -- information required for choosing basis functions -- more complicated inverse-type problems -- more complicated implementation -- more complicated temporal extrapolation in time -- usually requires a "regular" domain *e.g.* rectangle, $\mathbb{R}^d$, a ball, etc.	+ suitable for both bounded and unbounded domains + easy implementation + spectral convergence in space, more accurate + efficient deep-neural-network training + more interpretable derivatives of spatial variables + easy extrapolation + easily handles inverse-type problems + compatible with existing adaptive techniques -- requires some information to choose basis functions -- expensive evaluation of neural networks -- usually requires a "regular" domain

Table 8.6: Advantages and disadvantages of traditional and PINN-based numerical solvers. This table provides an overview of the advantages ('+') and disadvantages ('--') associated with different methods and solvers. Finite difference (FD), finite-element (FE), and spectral methods can be used in a traditional sense without relying on neural networks.

CHAPTER 9

Why case fatality ratios can be misleading: individual- and population-based mortality estimates and factors influencing them

This is the Accepted Manuscript version of an article accepted for publication in *Physical Biology*, **17**, 065003, (2020). IOP Publishing Ltd is not responsible for any errors or omissions in this version of the manuscript or any version derived from it. The Version of Record is available online at [10.1088/1751-8121/abf532].

9.1 Introduction

Mortality metrics are key quantities describing the severity of a viral disease [VOD20]. During an outbreak, these metrics typically evolve in time before converging to a constant value and can be defined in a number of ways. Commonly used metrics are the case fatality ratio, case fatality rate, and case fatality risk, which are all confusingly denoted "CFR" [KC13, DC].

Fatality *rate* implies a change in deaths per unit time, *risk* implies an individual probability, while *ratio* implies a fraction of two numbers, typically populations. CFR is most often defined as the ratio of the total estimated number of deaths to date, $D(t)$, to the estimated number of all confirmed cases to date $N_{\mathrm{c}}(t)$ [GLD09, XSW20, WM20, VOD20]. These numbers are key to estimating disease severity. Usually, antibody [tes20a] and reverse transcription-polymerase chain reaction (RT-PCR) testing [tes20b] is used to confirm SARS-CoV-2-positive patients. To find $D(t)$, the number of patients who actually die of COVID-19 must also be quantified. In Italy, deaths of patients with positive RT-PCR testing for SARS-CoV-2 are reported as COVID-19 deaths, but the criteria for COVID-19-related deaths are currently not clearly defined and may vary from region to region [ORB20].

Some studies define CFR as the "case fatality risk" and associate it with the probability of death of an individual confirmed case within "a period of time" [LDF15]. Yet others define case fatality ratio as simply "case fatality" and reserve the term case fatality ratio to mean the ratio of the case fatalities of two different diseases [DC]. Infection fatality ratios (IFR), the number of deaths to date divided by the number of all infected individuals, have also been used [JAH20, Fam20, OH20] although the IFR $= D(t)/N(t)$ requires an estimate of $N(t)$, the number of total (including unconfirmed) infected individuals. Similarly, IFR has also been called the "infection fatality risk," the probability of an individual dying conditioned on being infected. This individual-based definition of IFR is thus equivalent to the individual-based case fatality risk. However, in nearly all practical cases, both the CFR and IFR are estimated from aggregated population data from past outbreaks [GLD09] as well as from

those of the recent SARS-CoV-2 outbreaks [VOD20, OH20, JAH20, APV20, MC20, Rua20, SBK20].

Since the case fatality ratio is the most commonly used, we henceforth define CFR = $D(t)/N_c(t)$. We show examples of CFR curves (orange), which typically vary significantly both by region and in time, in Fig. 9.1 and in the Results and Discussion section. During the severe 2003 acute respiratory syndrome (SARS) outbreak in Hong Kong, the World Health Organization (WHO) also used the aforementioned estimate to obtain an initial CFR $\sim 3\%$ while the final values, after the resolution of infections, approached 17.0% [YLL05a, YLL05b] (see Fig. 9.1(a)).

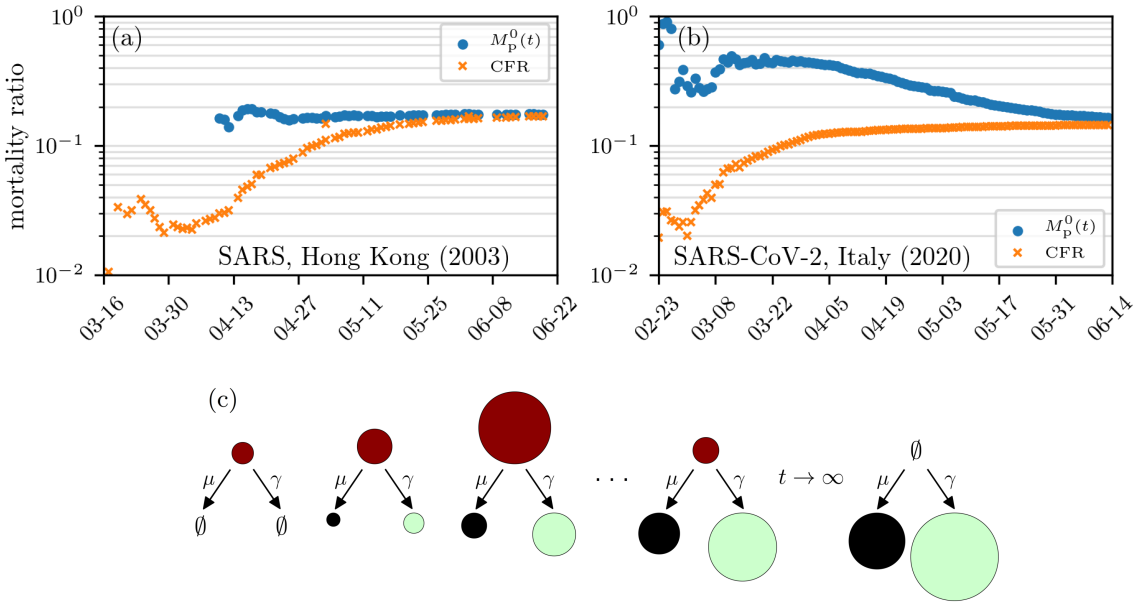

Figure 9.1: **Mortality estimates**. (a–b) Estimates of mortality ratios (see Eqs. (9.2.9) and (9.2.14)) of SARS-CoV infections in Hong Kong (2003) [Org20a] and SARS-CoV-2 infections in Italy. (c) Evolution of the cumulative number of infected (red), death (black), and recovered (green) cases. The size of the circles indicates the number of cases in the respective compartments on a certain day. Note that CFR and $M_p^0(t)$ have exhibited qualitatively similar behavior across different epidemics. The data are based on Ref. [DDG20].

Another population-based mortality ratio is $M_p(t) = D(t)/(D(t) + R(t))$, the number of deaths divided by the sum of death and recovered cases (the number of *resolved* cases), up to time t is shown in blue in Figs. 9.1(a–b). In principle, $M_p(t)$ should be a better measure of the likelihood of death, but it is underestimated by the CFR = $D(t)/N_c(t)$. For example, $M_p(t)$ is

reference	CFR
Xu *et al.* [XSW20, XLT20] and Mahase [Mah20]	2%
Wu *et al.* [WM20]	0.1-1% (outside Wuhan)
World Health Organization [Org20b, Org20c]	2-4%
Porcheddu *et al.* [PSK20]	2.3% (Italy and China)
Peeri [PSR20] *et al.*	2%

Table 9.1: Different CFR estimates of COVID-19.

currently (as of April 25, 2020) $203,164/(203,164+836,612) \approx 20\%$, significantly higher than the April 25, 2020 $\mathrm{CFR}(t) = D(t)/N_{\mathrm{c}}(t) = 203,164/2,919,404 \approx 7\%$ estimate [cor20]. Despite this underestimation, the CFR is still commonly used by the WHO and other health officials, such as in the ongoing SARS-CoV-2 outbreaks [SBK20, MC20, OH20, Fam20, VOD20] (see Table 9.1). As shown in Fig. 9.1(c), the CFR would correspond to the mortality ratio only if all tested infected individuals recover. Such underestimations by CFRs may lead to insufficient countermeasures and a more severe epidemic [BWA15, BWG16].

Since meaningful and accurate mortality metrics are critical for assessing the risks associated with epidemic outbreaks, we first unambiguously define the probability $M_1(t)$ that a single, newly infected individual will die of the disease by a given time. This probability has also been called the case fatality risk, but without specifying its dependence on time after infection [LDF15]. This intrinsic mortality or probability of death can be identified as one minus the survival probability of a single infected individual. It should be an *intrinsic* property of the virus and the infected individual, depending on age, health, access to health care, etc., and not *directly* on the population-level dynamics of infected and recovered individuals. Whether this individual infects others does not directly affect his probability of eventually dying [1].

In the next section, we derive a survival probability model for $M_1(t)$ similar to that in Ghani *et al.* [GDC05]. Importantly, our individual survival model incorporates the duration of infection (including an incubation period) before a patient tests positive at time $t = 0$. However, the CFR and other mortality measures are typically reported based on population

[1] Of course, at the population level, if there are many deaths, medical facilities may be stressed, which can indirectly lead to an increase in death rates.

data. Do these population-based measures, including CFR, provide reasonable measures of the probability of death of an individual? To address these and related issues, we develop an analogous population-based mortality metric based on a disease duration-structured SIR model. While population-based estimates of CFR are typically not a meaningful measure of individual mortality, under simplifying assumptions, the population-based mortality ratio $M_{\mathrm{p}}(t)$ is more closely related to the true probability of death $M_1(t)$ [GDC05].

We will use the same rate parameters in our individual and population models to compute and compare the different mortality measures. By critically analyzing and comparing these estimates, the CFR, and a "delayed" case fatality ratio $\mathrm{CFR}_{\mathrm{d}}$, we illustrate and interpret the differences among these measures and discuss how changes or uncertainty in the data affect them. In the Results and Discussion section, we identify a correction factor to transform population-level mortality estimates into individual mortality probabilities, and we discuss the effects of other possible confounding factors such as heterogeneous populations and undertesting (unconfirmed cases).

9.2 Mortality Measures

In this section, we present different mortality measures for *confirmed* cases and outline their underlying mathematical models.

9.2.1 Intrinsic individual mortality rate

Consider an individual that, at the time of positive testing ($t = 0$), had been infected for a duration τ_1. A "survival" probability density can be defined such that $P(\tau, t|\tau_1)\mathrm{d}\tau$ is the probability that the patient is still alive and infected (not recovered) at time $t > 0$ and has been infected for a duration between τ and $\tau + \mathrm{d}\tau$. Since τ_1 is unknown, it must be estimated or averaged over some distribution. The individual survival probability evolves according to [2].

[2]Since the disease timescale is much shorter than the timescale over which aging appreciably affects death or recovery, the age-dependent transport terms $\partial P/\partial a$ can be neglected.

$$\frac{\partial P(\tau,t|\tau_1)}{\partial t}+\frac{\partial P(\tau,t|\tau_1)}{\partial \tau}=-(\mu(\tau,t|\tau_1)+\gamma(\tau,t|\tau_1))P(\tau,t|\tau_1), \tag{9.2.1}$$

where the death and recovery rates, $\mu(\tau,t|\tau_1)$ and $\gamma(\tau,t|\tau_1)$, depend explicitly on the duration of infection at time t and can be further implicitly stratified according to patient age, gender, health condition, etc. [RHC20, VOD20]. They may also depend explicitly on time t to reflect changes in clinical policy or available health care. For example, enhanced medical care may decrease the death rate μ, giving the individual's intrinsic physiological processes a chance to cure the patient.

If we assume an initial condition of one individual having been infected for time τ_1 at the time of confirmation, Eq. 9.2.1 can be solved using the method of characteristics shown in the Appendix. From the solution $P(\tau=t+\tau_1,t|\tau_1)$, one can derive the probabilities of death and recovery by time t as

$$P_{\mathrm{d}}(t|\tau_1)=\int_0^t \mathrm{d}s\,\mu(\tau_1+s,s)P(\tau_1+s,t|\tau_1),\quad P_{\mathrm{r}}(t|\tau_1)=\int_0^t \mathrm{d}s\,\gamma(\tau_1+s,s)P(\tau_1+s,t|\tau_1). \tag{9.2.2}$$

The probability that an individual died before time t, conditioned on resolution (either death or recovery), is then defined as

$$M_1(t|\tau_1)=\frac{P_{\mathrm{d}}(t|\tau_1)}{P_{\mathrm{d}}(t|\tau_1)+P_{\mathrm{r}}(t|\tau_1)}. \tag{9.2.3}$$

Equations (9.2.2) and (9.2.3) also depend on all other relevant patient attributes such as age, accessibility to health care, etc. In the long-time limit, when the resolution has occurred ($P_{\mathrm{d}}(\infty|\tau_1)+P_{\mathrm{r}}(\infty|\tau_1)=1$), the individual mortality ratio is simply $M_1(\infty|\tau_1)=P_{\mathrm{d}}(\infty|\tau_1)$. In order to capture the dependence of death and recovery rates on the time an individual has been infected, we propose a constant recovery rate γ and a piecewise constant death rate $\mu(\tau|\tau_1)$ that is not explicitly a function of time t:

$$\gamma(\tau, t|\tau_1) = \gamma, \quad \mu(\tau|\tau_1) = \begin{cases} 0 & \tau \leq \tau_{\text{inc}} \\ \mu_1 & \tau > \tau_{\text{inc}} \end{cases} . \tag{9.2.4}$$

Here, τ_{inc} is the incubation time during which the patient is asymptomatic, has a negligible chance of dying, but can recover by clearing the virus. In other words, some patients fully recover without ever developing serious symptoms.

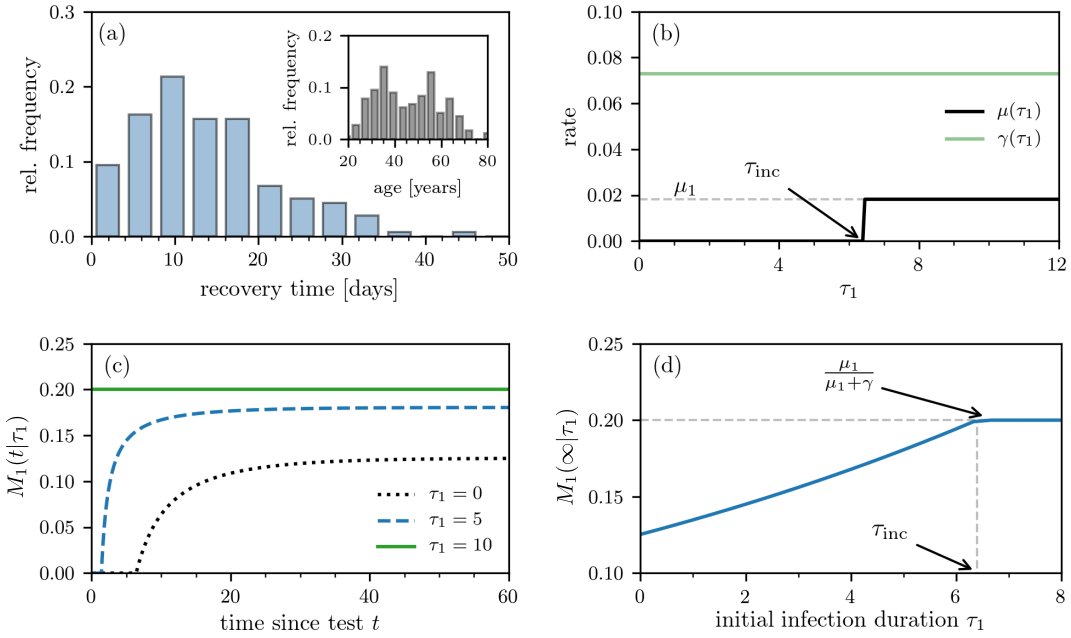

Figure 9.2: **Individual mortality**. (a) Recovery time after first symptoms occurred based on individual data of 178 patients [COV20]. The inset shows the age distribution of these patients. (b) Death- and recovery rates as defined in Eq. (9.2.4). The death rate $\mu(\tau_1)$ approaches μ_1 for $\tau_1 > \tau_{\text{inc}}$, where τ_{inc} is the incubation period and τ_1 is the time the patient has been infected before first being tested positive. (c) The individual mortality ratio $M_1(t|\tau_1)$ for $\tau_{\text{inc}} = 6.4$ days at different values of τ_1. Note that the individual death probability $P_{\text{d}}(t|\tau_1)$ and $M_1(t|\tau_1)$ are nonzero only after $t > \tau_{\text{inc}} - \tau_1$. (d) The asymptotic individual mortality ratio $M_1(\infty)$ (see Eq. (9.2.3)) as a function of τ_1.

For coronavirus infections, the incubation period appears to be highly variable with a mean of $\tau_{\text{inc}} \approx 6.4$ days [LSK20]. We can estimate μ_1 and γ using recent individual patient data from Singapore where 178 patients (mean age: 46 years) had been tracked from the date on which their first symptoms occurred until they recovered [COV20], on average, after 13.7 days. We show the recovery-time distribution in Fig. 9.2(a). Compared to other existing

datasets, the Singapore COVID-19 dataset provides complete line lists for a large number of patients and is being updated regularly.

We then use the global mortality of all resolved cases ($\approx 20\%$ [cor20]) to determine the dependence between μ_1 and γ via $\mu_1/(\mu_1+\gamma) \approx 1/5$ (or $\gamma/\mu_1 \approx 4$). The constant recovery and post-incubation death rates [KR11] are thus

$$\gamma \approx \frac{1}{13.7}/\text{day} = 0.073/\text{day} \quad \text{and} \quad \mu_1 \approx \gamma/4 = 0.018/\text{day}. \tag{9.2.5}$$

Using these numbers, the recovery and death rate functions $\gamma(\tau, t|\tau_1)$ and $\mu(\tau|\tau_1)$ are plotted as functions of τ in Fig. 9.2(b). We show the evolution of $M_1(t|\tau_1)$ at different values of τ_1 in Fig. 9.2(c). The corresponding long-time limit $M_1(\infty|\tau_1)$ is readily apparent in Fig. 9.2(d): for $\tau_1 \geq \tau_{\text{inc}}$, $M_1(\infty|\tau_1) = \mu_1/(\mu_1+\gamma) \approx 0.2$, while $M_1(\infty|\tau_1) < \mu_1/(\mu_1+\gamma)$ when $\tau_1 < \tau_{\text{inc}}$. The smaller expected mortality associated with early identification of infection arises from the remaining incubation time during which the patient has a chance to recover without possibility of death. When conditioned on testing positive at or after the incubation period, the patient immediately experiences a positive death rate, increasing his $M_1(\infty|\tau_1)$.

In order to infer M_1 (and also indirectly μ and γ) during an outbreak, a number of statistical issues must be considered. First, if the outbreak is ongoing, there may not be sufficient long-time cohort data. Second, τ_1 is unknown. Since testing typically occurs at the onset of symptoms, most positive patients will have been infected a few days earlier. The uncertainty in τ_1 can be represented by a probability density $\rho(\tau_1)$ for the individual. The expected mortality can then be constructed as an average over $\rho(\tau_1)$:

$$\bar{M}_1(t) = \frac{\bar{P}_{\text{d}}(t)}{\bar{P}_{\text{d}}(t) + \bar{P}_{\text{r}}(t)}, \tag{9.2.6}$$

where $\bar{P}_{\text{d}}(t)$ and $\bar{P}_{\text{r}}(t)$ are the τ_1-averaged probabilities death and cure probabilities.

Some properties of the distribution $\rho(\tau_1)$ can be inferred from the behavior of patients. Before symptoms arise, only very few patients will know they have been infected, seek

medical care, and get their case confirmed (*i.e.*, $\rho(\tau_1) \approx 0$ for $\tau_1 \approx 0$). The majority of patients will seek care when they have been infected for approximately τ_{inc}. We choose the gamma distribution

$$\rho(\tau_1; n, \lambda) = \frac{\lambda^n}{\Gamma(n)} \tau_1^{n-1} e^{-\lambda \tau_1} \tag{9.2.7}$$

with shape parameter $n = 8$ and rate parameter $\lambda = 1.25/\text{day}$ so that the mean n/λ is equal to $\tau_{\text{inc}} = 6.4$. Note that, independent of the distribution ρ, the average $\bar{M}_1(t)$ is bounded from above by $M_1(\infty) = \mu_1/(\mu_1 + \gamma)$ for all times t.

Upon using the rates in Eqs. (9.2.4) and averaging over $\rho(\tau_1)$, we derived expressions for $\bar{P}(t), \bar{P}_{\text{d}}(t)$, and $\bar{P}_{\text{r}}(t)$ which are explicitly given in the Appendix. Using the values in Eq. (9.2.5) we find an expected individual mortality ratio $\bar{M}_1(t)$ (which are subsequently plotted in Fig. 9.3) and its asymptotic value $\bar{M}_1(\infty) = \bar{P}_{\text{d}}(\infty) \approx 0.19$ (slightly less than $M_1(\infty|\tau_1)$ due to averaging over $\rho(\tau_1)$). Of course, it is also possible to account for more complex time-dependent forms of γ and μ_1 [BA20], but we will primarily use Eqs. (9.2.4) in our subsequent analyses. We stress that $M_1(t)$ tracks mortality of a cohort of individuals infected at about the same time, and does not include mortality of newly infected individuals. Thus, it can be trivially stratified according to different age groups and defined as the mortality $M_1(t|\mu)$ of each age subpopulation with death rate μ.

In the next subsection, we define population-based estimates for mortality ratios, $M_{\text{p}}(t)$, and explore how they can be computed using SIR-type models. By comparing $\bar{M}_1(t)$ to $M_{\text{p}}(t)$, we gain insight into whether population-based metrics are good proxies for individual mortality ratios.

9.2.2 Relation to infection duration-dependent SIR model

While individual mortalities can be estimated by tracking many individuals from infection to recovery or death, often, the available data are not resolved at the individual level and only total populations are given. Typically, one only has the total number of confirmed

cases accumulated up to time t, $N_{\rm c}(t)$, the number of deaths to date $D(t)$, and the number of cured/recovered patients to date $R(t)$ (see Fig. 9.1). Note that $N_{\rm c}(t)$ includes unresolved cases and that $N_{\rm c}(t) \geq R(t) + D(t)$. Resolution (death or recovery) of all patients, $N_{\rm c}(\infty) = R(\infty) + D(\infty)$, occurs only well after the epidemic completely passes.

A variant of the CFR commonly used in the literature is the delayed CFR [XSW20, WM20]

$$\mathrm{CFR}_{\rm d}(t, \tau_{res}) = \frac{D(t)}{N_{\rm c}(t - \tau_{res})}, \tag{9.2.8}$$

which uses an earlier and smaller case number to compensate for underestimation by the standard CFR

$$\mathrm{CFR}(t) = \frac{D(t)}{N_{\rm c}(t)} \equiv \mathrm{CFR}_{\rm d}(t, \tau_{res} = 0). \tag{9.2.9}$$

The delay $\tau_{\rm res}$ used is typically the time between the day symptoms first occurred and the day of death or recovery. To determine a realistic value of the delay time $\tau_{\rm res}$ (which can be qualitatively interpreted as a resolution time), we use data on death/recovery periods of 36 tracked COVID-19 patients [20120] and find that patients recover/die, on average, $\tau_{\rm res} \approx$ 2 weeks after first symptoms occurred. The delayed $\mathrm{CFR}_{\rm d}(t, \tau_{\rm res} > 0)$ also underestimates the individual mortality in previous epidemic outbreaks of SARS [GDC05, YLL05a] and Ebola [AWN15], but is highly sensitive to $\tau_{\rm res}$. If the delay between the time of infection and the time of resolution were vanishingly small, we can set $\tau_{\rm res} = 0$ and find that the $\mathrm{CFR}_{\rm d}$ and CFR are equivalent (see Eq. (9.2.9)).

Alternatively, a simple and interpretable population-level mortality is $M_{\rm p}(t) = D(t)/(R(t)+D(t))$, the ratio of infected deaths to all resolved cases of confirmed infections. To provide a concrete model for $D(t)$ and $R(t)$, and hence $M_{\rm p}(t)$, we will use a variant of the standard infection duration-dependent susceptible-infected-recovered (SIR)-type model described by [Web08, WLB20]

$$\frac{\mathrm{d}S(t)}{\mathrm{d}t} = -S(t) \int_0^\infty \mathrm{d}\tau' \, \beta(\tau', t) I(\tau', t),$$

$$\frac{\partial I(\tau,t)}{\partial t}+\frac{\partial I(\tau,t)}{\partial \tau}=-(\mu(\tau,t)+\gamma(\tau,t))I(\tau,t), \tag{9.2.10}$$

and $\mathrm{d}R(t)/\mathrm{d}t=\int_0^\infty \mathrm{d}\tau\gamma(\tau,t)I(\tau,t)$, where $S(t)$ is the number of susceptibles, $I(\tau,t)$ is density of individuals at time t who have been infected for time τ, and $R(t)$ is the number of recovered individuals. The rate at which an individual infected for time τ at time t infects susceptibles is denoted by $\beta(\tau,t)S(t)$. For simplicity, we assume only community spread and neglect immigration of infected individuals, which could be straightforwardly included [WLB20].

Note that the equation for $I(\tau,t)$ is identical to the equation for the survival probability described by Eq. (9.2.1). It is also equivalent to McKendrick age-structured models [McK26, CG16]. In both the individual model (Eq. (9.2.1)) and population model (Eq. (9.2.10)), the death and recovery rates are insensitive to changes in age a over the $\lesssim 1$ year epidemic timescale. In this limit, we consider only infection-duration dependence on the population dynamics. However, in contrast to the individual survival probability, new infections of susceptibles are described by the boundary condition (or renewal equation)

$$I(\tau=0,t)=S(t)\int_0^\infty \mathrm{d}\tau'\,\beta(\tau',t)I(\tau',t), \tag{9.2.11}$$

which is similar to that used in age-structured models to represent birth [McK26]. The initial time $t=0$ is arbitrary as long as the initial condition $I(\tau,0)$ is defined. We use an initial condition corresponding to a single infected with the infection duration density given by Eq. (9.2.7): $I(\tau,0)=\rho(\tau;n=8,\lambda=1.25)$. Note that Eq. (9.2.11) assumes that all newly infected individuals are immediately identified; *i.e.*, these newly infected individuals start with $\tau_1=0$. After solving for the infected population density, we find the total number of deaths, recoveries, and total cases to date,

$$\begin{aligned} &D^0(t)=\int_0^t \mathrm{d}t'\int_0^\infty \mathrm{d}\tau\,\mu(\tau,t')I(\tau,t'),\quad R^0(t)=\int_0^t \mathrm{d}t'\int_0^\infty \mathrm{d}\tau\,\gamma(\tau,t')I(\tau,t'),\\ &N_0(t)=R^0(t)+D^0(t)+\int_0^\infty \mathrm{d}\tau\,I(\tau,t), \end{aligned} \tag{9.2.12}$$

and use $D^0(t)$ and $N_0(t)$ for $D(t)$ and $N_c(t)$ in definitions of $\mathrm{CFR}(t)$ and $\mathrm{CFR}_d(t, \tau_{res})$ (Eq. (9.2.8)). In the definitions of $D^0(t)$, $R^0(t)$, and $N_0(t)$, we account for all possible death and recovery cases to date (see Appendix) and that newly infected individuals are immediately identified. We use these case numbers as approximations of the reported case numbers to study the evolution of mortality ratio estimates. Mortalities based on these numbers underestimate the actual individual mortality M_1 (see the previous "Intrinsic individual mortality rate" subsection) since they involve individuals that have been infected for different durations τ, particularly recently infected individuals who have not yet died.

An alternative way to compute populations is to exclude new infections and consider only an initial cohort. The corresponding populations in this case are defined as

$$D^1(t) = \int_0^t \mathrm{d}t' \int_{t'}^{\infty} \mathrm{d}\tau\, \mu(\tau, t') I(\tau, t'), \quad R^1(t) = \int_0^t \mathrm{d}t' \int_{t'}^{\infty} \mathrm{d}\tau\, \gamma(\tau, t') I(\tau, t'). \tag{9.2.13}$$

Since $D^1(t)$ and $R^1(t)$ do not include infected individuals with $\tau < t$, they exclude the effect of newly infected individuals and may yield more meaningful mortalities as they would be based on an initial cohort of individuals in the distant past. It is superfluous to define CFR using $D^1(t)/N_c$ because the corresponding N_c of a cohort is a constant. The infections that occur after $t = 0$ contribute only to $I(\tau < t, t)$; thus, $D^1(t)$ and $R^1(t)$ do not depend on the transmission rate β, possible immigration of infected individuals, or the number of susceptibles $S(t)$. Note that all the populations derived above implicitly average over $\rho(\tau_1; n, \gamma)$ for the first cohort of identified infected individuals (but not subsequent infections). Moreover, the population density $I(\tau \geq t, t)$ follows the same equation as $\bar{P}(t|\tau_1)$ provided the same $\rho(\tau_1; n, \lambda)$ is used in their respective calculations.

The two different ways of partitioning populations (Eqs. (9.2.12) and (9.2.13)) lead to two different population-level mortality ratios

$$M_p^0(t) \equiv \frac{D^0(t)}{D^0(t) + R^0(t)} \quad \text{and} \quad M_p^1(t) \equiv \frac{D^1(t)}{D^1(t) + R^1(t)}. \tag{9.2.14}$$

Since the populations $D^0(t)$ and $R^0(t)$, and hence $M_p^0(t)$, depend on disease transmission

through $\beta(\tau, t)$ and $S(t)$, we expect $M_{\rm p}^0(t)$ to carry a different interpretation from $M_1(t)$ and $M_{\rm p}^1(t)$.

In the special case in which μ and γ are constants, the time-integrated populations $\int_0^t \mathrm{d}t' \int_0^\infty \mathrm{d}\tau\, I(\tau, t')$ and $\int_0^t \mathrm{d}t' \int_{t'}^\infty \mathrm{d}\tau\, I(\tau, t')$ factor out of $M_{\rm p}^0(t)$ and $M_{\rm p}^1(t)$, rendering them time-independent and

$$M_{\rm p}^{0,1} = \frac{\mu_1}{\mu_1 + \gamma} = M_1. \tag{9.2.15}$$

Thus, only in the special time-homogeneous case do both population-based mortality ratios become *independent* of the population (and transmission β) and coincide with the individual death probability.

To illustrate, in more general cases, the differences between $M_1(t), M_{\rm p}^{0,1}(t)$ and $\mathrm{CFR}_{\rm d}(t, \tau_{\rm res})$, we use the simple death and recovery rate functions given by Eqs. (9.2.4) in solving Eqs. (9.2.1) and (9.2.10). For $\beta(\tau, t)$ in Eq. (9.2.11), we use a recently inferred infectiousness profile [HLW20] which is described by a gamma distribution

$$\beta(\tau) = \beta_0 \rho(\tau; n, \lambda) \tag{9.2.16}$$

with a peak that occurs shortly before the onset of symptoms at the time $\tau_{\rm inc}$ and coincidentally has $n = 8$ and $\lambda = 1.25$/day as in the testing time distribution $\rho(\tau_1)$ from a single infected (Eq. (9.2.7)). The constant dimensionless prefactor β_0 sets the amplitude of the transmission rate. For the chosen parameters n and λ, the gamma distribution $\rho(\tau; n, \lambda)$ reaches a maximum at $\tau \approx 5.6$ days, about one day before $\tau_{\rm inc} = 6.4$ days [HLW20]. Assuming that the susceptible pool is not appreciably depleted, $S(t) \approx S_0$ and Eq. (9.2.11) becomes $I(\tau = 0, t) = \beta_0 S_0 \int_0^\infty \mathrm{d}\tau'\, \rho(\tau; n, \lambda) I(\tau', t)$. The amplitude $\beta_0 S_0$ can be found by assuming a single infected for $I(\tau, t)$ in the renewal equation and using the estimated basic reproduction number. The basic reproduction number $\mathcal{R}_0$ is the average number of secondary infections that result from any single infected individual before he dies or recovers [KR11]. There are

two terms to consider when determining $\mathcal{R}_0$: (i) $\beta(\tau)\,\mathrm{d}\tau$ is the probability that an infection occurs in $[\tau, \tau+\mathrm{d}\tau]$ and (ii) $\exp\left[-\int_0^\tau(\mu(\tau')+\gamma)\mathrm{d}\tau'\right]$ is the probability that a single infected individual has not died or recovered prior to time τ. If we integrate over the product of these quantities and multiply by the total susceptible population S_0 (which is equivalent to the boundary condition (9.2.11) applied to a single infected individual), we obtain the average number of susceptibles infected by one infected individual, *i.e.*, $\mathcal{R}_0$. Thus, upon using Eq. (9.2.16), $\beta_0 S_0$ can be found by solving

$$S_0\int_0^\infty \beta(\tau)\exp\left[-\int_0^\tau(\mu(\tau')+\gamma)\mathrm{d}\tau'\right]\mathrm{d}\tau = \beta_0 S_0\int_0^\infty \rho(\tau;n,\lambda)\exp\left[-\int_0^\tau(\mu(\tau')+\gamma)\mathrm{d}\tau'\right]\mathrm{d}\tau$$
$$= \mathcal{R}_0 \approx 2.91. \tag{9.2.17}$$

Using the death and recovery rate functions given by Eqs. 9.2.4 and 9.2.5, we find $\beta_0 S_0 \approx 4.64$/day. Using this value, we numerically solve Eqs. (9.2.10) and (9.2.11) (see Appendix for further details) and use these solutions to compute $D^{0,1}(t)$, $R^{0,1}(t)$, and $N_{0,1}(t)$, which are then used in Eqs. (9.2.14) and $\mathrm{CFR}_\mathrm{d}(t,\tau_\mathrm{res})$.

9.3 Results and Discussion

9.3.1 Comparison of mortalities

Here, we evaluate and compare the different mortality metrics and show how some of them qualitatively resemble the measured mortality estimates shown in Fig. 9.1. In Fig. 9.3(a), we show the unbounded subpopulations $I^0(t), D^0(t)$, and $R^0(t)$ computed using Eqs. (9.2.10), (9.2.11), and (9.2.12) when the susceptible population is assumed constant. Fig. 9.3(b) shows the populations when a strict quarantine ($S(t>t_\mathrm{q})=0$) is applied after $t_\mathrm{q}=50$ days.

The mortalities plotted in Figs. 9.3(c) show that $M_\mathrm{p}^1(t)$ approaches the individual mortality ratio $\bar{M}_1(\infty)\approx 0.19$ given in the "Intrinsic individual mortality rate" subsection above.

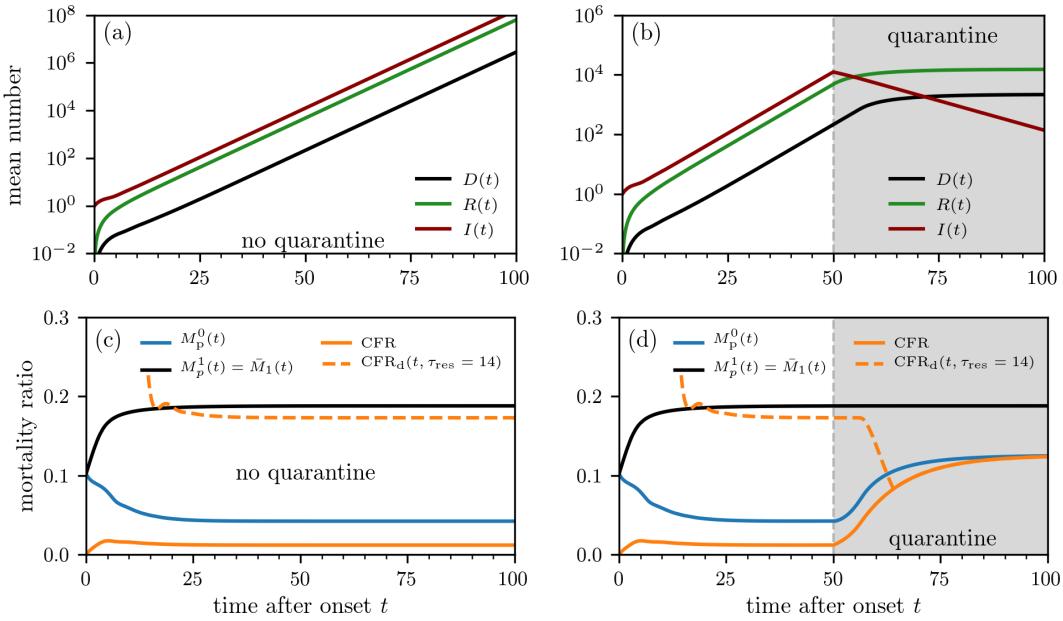

Figure 9.3: **Population-level mortality estimates.** Outbreak evolution and mortality ratios without containment measures (a,c) and with quarantine (b,d). The curves are based on numerical solutions of Eqs. (9.2.10) using the initial condition $I(\tau, 0) = \rho(\tau; 8, 1.25)$ (see Eq. (9.2.7)). The death and recovery rates are defined in Eqs. (9.2.4) and (9.2.5). We use an infection rate (Eq. (9.2.16)) defined by $\beta_0 S_0 = 4.64$/day, which we estimated from the basic reproduction number of SARS-CoV-2 [LSK20]. To model quarantine effects, we set $\beta_0 S_0 = 0$ for $t > 50$ days. We show the mortality-ratio estimates $M_p^0(t)$ and $M_p^1(t)$ (see Eq. (9.2.14)) and $\mathrm{CFR}_d(t, \tau_{res})$ (see Eqs. (9.2.8), (9.2.12), and (9.2.14)). $\mathrm{CFR}_d(t, \tau_{res} = 14 \text{ days})$ behaves very differently from CFR, initially decreasing for $\tau_{res} > 0$ and significantly *overestimating* $M_p^0(t)$ but providing a reasonable estimate of $\bar{M}_1(t) = M_p^1(t)$ without quarantine. Note that under quarantine, $\mathrm{CFR}(\infty)$, $\mathrm{CFR}_d(\infty)$, and $M_p^0(\infty)$ approach the same value since they reflect the mortality ratio of the total cohort at the time of quarantine. On the other hand, $\bar{M}_1(t) = M_p^1(t)$ reflects the ratio of the initial cohort at the start of the outbreak and remains unchanged from the no-quarantine case.

This occurs because the model for $P(\tau, t)$ and $I(\tau, t)$ are equivalent and we assumed the same initial distribution $\rho(\tau; 8, 1.25)$ for both quantities. However, the population-level mortality ratios $\mathrm{CFR}_d(t, \tau_{res})$ and $M_p^0(t)$ also take into account recently infected individuals who may recover before symptoms. This difference yields different mortality ratios because newly infected individuals are implicitly assumed to be detected immediately and all have $\tau_1 = 0$. Thus, the underlying infection-time distribution is not the same as that used to compute $\bar{M}_p^1(t)$ (see Appendix for further details). The mortality ratio $M_p^0(t)$ should not be used to quantify the individual mortality probability $\bar{M}_1(t)$ of individuals who tested positive, while the accuracy of $\mathrm{CFR}_d(t, \tau_{res})$ is sensitive to τ_{res} and quarantining. Moreover, due to the

evolution of the disease, $D(t)$, $R(t)$, and $N(t)$ do not change with the same rates during an outbreak, the population-level mortality measures $\mathrm{CFR}_{\mathrm{d}}(t, \tau_{\mathrm{res}})$ and $M_{\mathrm{p}}^{0}(t)$ reach their final steady-state values only after sufficiently long times. Figs. 9.3(d) shows the corresponding mortalities with quarantining after $t_{\mathrm{q}} = 50$ days.

The population-level ratios $M_{\mathrm{p}}^{0}(t)$ and $\mathrm{CFR}(t)$ implicitly depend on new infections and the transmission rate β. Despite this confounding factor, $M_{\mathrm{p}}^{0}(t)$ and $\mathrm{CFR}_{\mathrm{d}}(t, \tau_{\mathrm{res}})$ approach $e^{-\gamma \tau_{\mathrm{inc}}} \mu_1/(\mu_1+\gamma)$ as $t \to \infty$, where $e^{-\gamma \tau_{\mathrm{inc}}}$ is the probability that no recovery occurred during the incubation time τ_{inc}. Based on these results, we can establish the following connection between the different mortality ratios for initial infection times with distribution $\rho(\tau_1; n, \lambda)$ and mean $\bar{\tau} = n/\lambda$:

$$\mathrm{CFR}_{\mathrm{d}}(\infty) = M_{\mathrm{p}}^{0}(\infty) \approx e^{-\gamma \bar{\tau}} M_{\mathrm{p}}^{1}(\infty) = e^{-\gamma \bar{\tau}} \bar{M}_1(\infty). \tag{9.3.1}$$

According to Eq. (9.3.1), population-level mortality estimates (*e.g.*, CFR and M_{p}^{0}), can be transformed, at least approximately, into individual mortality probabilities using the correction factor $e^{-\gamma \bar{\tau}}$ with $\bar{\tau} \approx \tau_{\mathrm{inc}}$.

Although population-level quarantining does not directly affect the individual mortality $M_1(t|\tau_1)$ or $\bar{M}_1(t)$, it can be easily incorporated into the SIR-type population dynamics equations through changes in $\beta(\tau, t)S(t)$. For example, we have set $S(t > t_{\mathrm{q}}) = 0$ to represent the implementation of a perfect quarantine after $t_{\mathrm{q}} = 50$ days of the outbreak. After $t_{\mathrm{q}} = 50$ days, no new infections occur and the estimates $\mathrm{CFR}(t)$ and $M_{\mathrm{p}}^{0}(t)$ start to converge towards their common larger value (see Fig. 9.3(d)). In other words, without quarantining, the infected and recovered populations are continuously increasing, keeping CFR and $M_{\mathrm{p}}^{0}(t)$ low. Since the number of deaths decreases after the implementation of quarantine measures, the delayed $\mathrm{CFR}_{\mathrm{d}}(t, \tau_{\mathrm{res}} = 14 \text{ days})$ is first decreasing until $t = t_{\mathrm{q}} + \tau_{\mathrm{res}} = 64$ days. For $t > 64$ days, the $\mathrm{CFR}_{\mathrm{d}}(t, \tau_{\mathrm{res}} = 14 \text{ days})$ measures no new cases and is thus equal to the CFR.

The overall time evolution of some of the mortalities in Fig. 9.3 qualitatively resembles

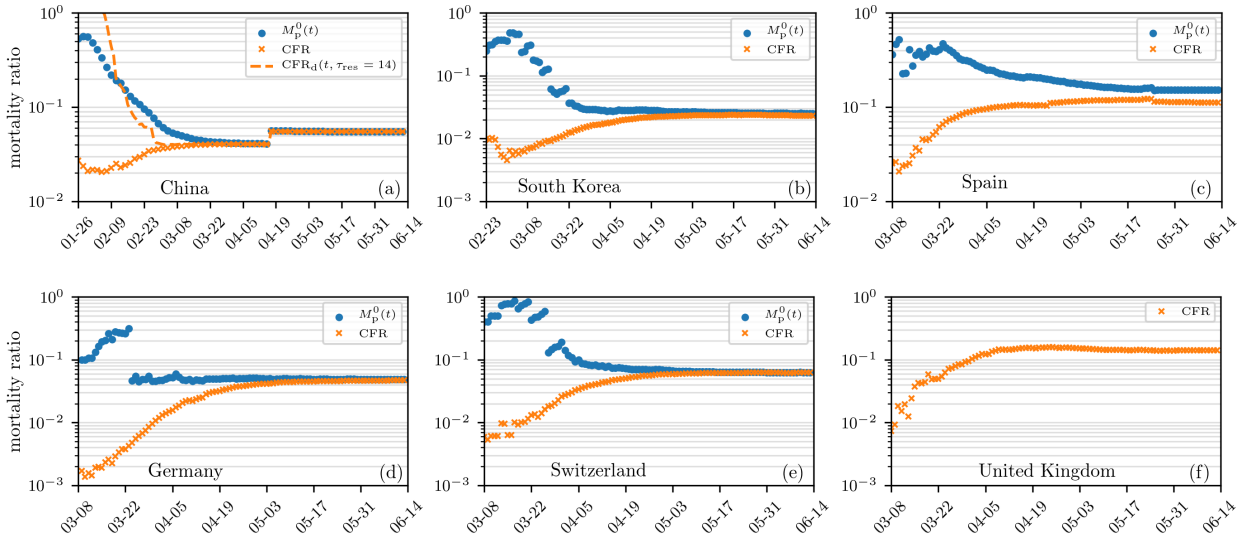

Figure 9.4: **Mortality estimates in different countries**. Estimates of mortality ratios (see Eqs. (9.2.8) and (9.2.14)) of SARS-CoV-2 infections in different countries. The data are derived from Ref. [DDG20]. The case fatality rate, CFR, corresponds to the number of deaths to date divided by the total number of cases to date. The "delayed" mortality-ratio estimate CFR_d corresponds to the number of deaths to date divided by the total number of cases at time $t - \tau_\mathrm{res}$ is also shown for China. The population-based mortality ratios $M_\mathrm{p}(t)$ are also shown, except for the UK which has reported an inexplicable $M_\mathrm{p}^0(t) \sim 1$.

the behavior of the mortality estimates in Fig. 9.1. As shown in Fig. 9.1, the CFR is increasing over time whereas M_p^0 provides a more stable mortality estimate for the SARS-CoV outbreak in Hong Kong (2003) and seems to follow a similar behavior in the current SARS-CoV-2 outbreak in Italy. In Fig. 9.4, we show additional examples of mortality-ratio estimates for China, South Korea, Spain, Germany, Switzerland, and the United Kingdom. After an initial transient, the CFR, in most cases, increases to a new asymptote after the epidemic passes. As in Fig. 9.1, we observe, consistent with their definitions, that the population-based mortality ratio $M_\mathrm{p}^0(t)$ is larger than the corresponding CFR in all cases. $M_\mathrm{p}^0(t)$ also appears to be a temporally more stable metric. Differences in the evolution of mortality ratios in different regions could result from changing practices in data collection or from explicitly time-inhomogeneous parameters $\mu(\tau, t)$, $\gamma(\tau, t)$, and/or $\beta(\tau, t)$.

Differences in demographics can easily be a source of variability in mortality rates measured across different regions. Older patients and those with underlying medical conditions

typically have a higher death rate $\mu(\tau, t)$ and/or lower recovery rate γ. Since we focus on mortality, the different subpopulations within the confirmed population matter only through their differences in μ and/or γ. For the $\bar{M}_1(t)$ and $M_\mathrm{p}^1(t)$ metrics, no new infections are used in their determination. Thus, these metrics are associated with the mean death and recovery rates of the original group of infected individuals, *i.e.*, the ratios $\bar{M}_1(t|\mu, \gamma)$ and $M_\mathrm{p}^1(t|\mu, \gamma)$ refer to the mortality ratios of each subpopulation or individual described by μ and γ. The effective $M_\mathrm{p}^1(t)$ over the entire confirmed population can be trivially constructed by population-averaging $D^1(t|\mu, \gamma)$ and $R^1(t|\mu, \gamma)$ over μ and γ before constructing $M_\mathrm{p}^1(t)$.

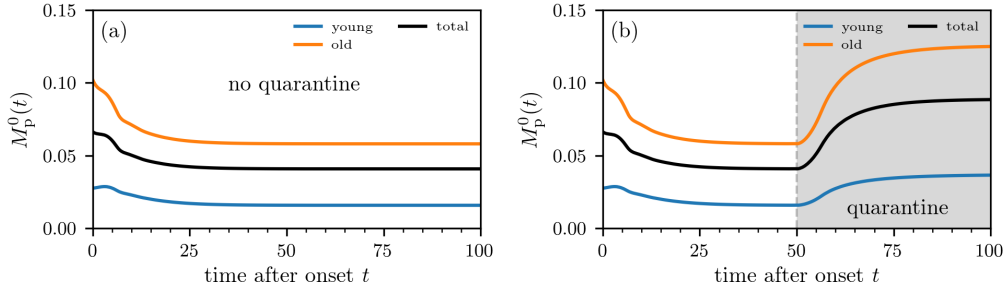

Figure 9.5: **Population-level mortality estimate for two age groups.** The mortality ratio $M_\mathrm{p}^0(t)$ without containment measures (a) and under quarantining (b). The curves are based on numerical solutions of Eqs. (9.3.2) and (9.3.3) assuming constant $S(t) \approx S_0$ and using the initial condition $I_\mathrm{a}(\tau, 0) = I_\mathrm{b}(\tau, 0) = \rho(\tau; 8, 1.25)/2$ (see Eq. (9.2.7)), where the subscripts "a" and "b" denote the young and old age group, respectively. The death and recovery rates for the younger age group are defined in Eqs. (9.2.4) and (9.2.5). For the older age group, we set $\mu_\mathrm{b} = 4\mu_\mathrm{a}$ and $\gamma_\mathrm{b} = \gamma_\mathrm{a}$. We use an infection rate (Eq. (9.2.16)) defined by $\beta_{\mathrm{a\,a}} S_0 = 4.64/\mathrm{day}$, which we estimated from the basic reproduction number of SARS-CoV-2 [LSK20]. The remaining infection rates are defined via $\beta_{\mathrm{a\,a}} = \sqrt{2}\beta_{\mathrm{b\,a}} = \sqrt{2}\beta_{\mathrm{a\,b}} = 2\beta_{\mathrm{b\,b}}$. To model quarantine effects, we set $\beta_0 S_0 = 0$ for $t > 50$ days in (b).

For the other confirmed mortalities $M_\mathrm{p}^0(t)$ and $\mathrm{CFR}(t)$, new infections are taken into account and subpopulations with different death and recovery rates can infect each other. Suppose there are two subpopulations "a" and "b" (*e.g.*, young and old) with associated death and recovery rates $\mu_{\mathrm{a,b}}$ and $\gamma_{\mathrm{a,b}}$, respectively. The equations for each subpopulation are

$$\frac{\partial I_\mathrm{a}(\tau, t)}{\partial t} + \frac{\partial I_\mathrm{a}(\tau, t)}{\partial \tau} = -(\mu_\mathrm{a}(\tau, t) + \gamma_\mathrm{a}(\tau, t)) I_\mathrm{a}(\tau, t),$$

$$\frac{\partial I_{\rm b}(\tau,t)}{\partial t}+\frac{\partial I_{\rm b}(\tau,t)}{\partial \tau}=-(\mu_{\rm b}(\tau,t)+\gamma_{\rm b}(\tau,t))I_{\rm b}(\tau,t), \tag{9.3.2}$$

indicating that each subpopulation follows its own dynamics for $\tau > 0$. However, the subpopulations interact with each other through the coupled boundary conditions

$$\begin{aligned} I_{\rm a}(0,t) &= S(t)\int_0^\infty {\rm d}\tau' \ [\beta_{\rm aa}(\tau',t)I_{\rm a}(\tau',t)+\beta_{\rm ab}(\tau',t)I_{\rm b}(\tau',t)] \\ I_{\rm b}(0,t) &= S(t)\int_0^\infty {\rm d}\tau' \ [\beta_{\rm ab}(\tau',t)I_{\rm a}(\tau',t)+\beta_{\rm bb}(\tau',t)I_{\rm b}(\tau',t)] \end{aligned} \tag{9.3.3}$$

that describe cross-infections between the "a" and "b" subpopulations. Thus, the infection levels in each subpopulation also depend on the transmission rates $\beta_{\rm aa}$, $\beta_{\rm ab}$, and $\beta_{\rm bb}$. To compute the overall confirmed mortality $M_{\rm p}^0(t)$ or CFR(t) of the entire population, we must solve Eqs. (9.3.2) and (9.3.3) for $I_{\rm a}$ and $I_{\rm b}$, and hence $D_{\rm a}(t)$, $D_{\rm b}(t)$, and $D(t) = D_{\rm a}(t)+D_{\rm b}(t)$.

In Fig. 9.5, we show the evolution of $M_{\rm p}^0(t)$ for two age groups representing young and old individuals with different mortality and infection rates. The behavior of $M_{\rm p}^0(t)$ for the entire population is qualitatively similar to, but falls in between those of each age group (see Fig. 9.3). Whether the overall mortality is closer to that of the young or old population depends on the relative populations of the young and old infected, their death and recovery rates, and their cross transmission rates $\beta_{\rm ab}$. For age-stratified case data, the subpopulation model outlined above, or other approaches such as scaling approximations [Seo21] may be useful for capturing age-dependent variations in $M_{\rm p}^0(t)$.

9.3.2 Undertesting and unconfirmed cases

Another important confounding factor is the large number of untested and often asymptomatic infected individuals. The mortality rate often quoted in the literature ranges from $< 1-3\%$, which is much smaller than the resolved mortality ratios we have used for illustration. Our estimates of $M_{\rm p}^{0,1}(t)$ and ${\rm CFR}_{\rm d}(t,\tau_{\rm res})$ using $I(\tau,t)$ actually describe the mor-

tality of the population *conditioned* on being tested positive. Since we used Eqs. (9.2.10) to compute infected populations, we implicitly assumed that all infected individuals have been tested/confirmed. However, the total infected population is comprised of tested and untested individuals, which may or may not carry different death and recovery rates. Typically, only a small fraction f of the total number of infected individuals might be tested and confirmed positive.

Our confirmed mortalities (derived from only the positively tested population) can be extended to the entire population, tested or untested. The "true" $\mathcal{M}_{\rm p}^0$ and the fatality ratio *conditioned on having been infected* (the IFR) would typically be much smaller than the $M_{\rm p}^0$ and CFR calculated using only confirmed cases. How the testing fraction $f < 1$ might qualitatively affect the "true" underlying mortality measures (the mortality conditioned on simply being infected) is illustrated in Fig. 9.6.

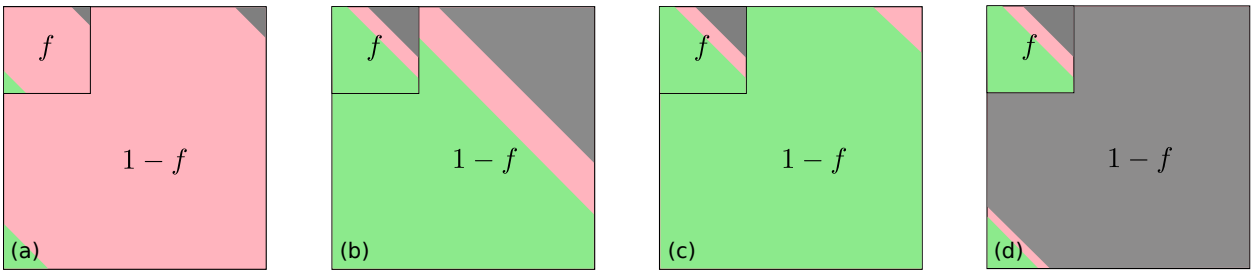

Figure 9.6: **Fractional testing.** An example of fractional testing in which a fixed fraction f of the real total infected population is assumed to be tested. The remaining $1-f$ proportion of infected individuals is untested. Equivalently, if the total tested fraction has a unit population, then the fraction of the population that remains untested is $1/f - 1$. (a) At short times after an outbreak, most of the infected patients, tested and untested, have not yet resolved (red). Only a small number have died (gray) or have recovered (green). (b) At later times, if the untested population dies at the same rate as the tested population, $M_{\rm p}(t)$ and CFR remain accurate estimates for the entire infected population. (c) If the untested population is, say, asymptomatic and rarely dies, the true mortality $\mathcal{M}_{\rm p}^{0,1}(\infty) \approx f M_{\rm p}^{0,1}(\infty)$ can be significantly overestimated by the tested mortality $M_{\rm p}^{0,1}(t)$. (d) Finally, in a scenario in which untested infected individuals die at a higher rate than tested ones, $M_{\rm p}^{0,1}(t)$ and CFR based on the tested fraction *underestimate* the true mortality $\mathcal{M}_{\rm p}^{0,1}$.

Estimates for SARS-CoV-2 show that f is small (e.g., $f \approx 14\%$ in China before January 23, 2020) [LPC20]. At early times (Fig. 9.6(a)) most patients, tested or untested, have not yet resolved. A reported/tested fraction $f < 1$ *would not* directly affect or alter the CFRs or mortality ratios if the unreported/untested population dies and recovers in the

same proportion as those tested, as depicted in Fig. 9.6(b). That is, undertesting would still provide a good estimate of the true mortality if the entire population were homogeneous in death and recovery rates. However, if the untested (presumably mildly or asymptomatic infected) are less likely to die than the tested infected individuals, undertesting would give rise to $M_{\rm p}^0(t)$ and CFR(t) that overestimate the true mortality $\mathcal{M}_{\rm p}^0(t)$ and the infection fatality ratio (IFR). If untested infected individuals do not die at all, as depicted in Fig. 9.6(c), the true long-time mortality $\mathcal{M}_{\rm p}^{0,1}(\infty) \approx f M_{\rm p}^{0,1}(\infty)$. In the unlikely scenario in which untested individuals do not receive medical care and hence die at a faster rate (Fig. 9.6(d)), $M_{\rm p}^{0,1}(\infty)$ and CFR based on the tested fraction would underestimate the true long-time mortality $\mathcal{M}_{\rm p}^{0,1}(\infty)$ and IFR, respectively.

To quantitatively estimate the underlying mortality of the population conditioned simply on being infected, we have to quantify the number of confirmed and untested infected individuals, $I_{\rm c}(\tau, t)$ and $I_{\rm u}(\tau, t)$, which can be further divided into subpopulations with intrinsically different transmission, death, and recovery rates. The act of confirmation itself may change behavior and/or treatment, further changing transmission, death, and recovery parameters.

Metric / Subpopulation	Fatality Ratios	Resolved Mortality w/inf	Resolved Mortality w/o inf	Individual Risk
confirmed (tested)	CFR $= \frac{D_{\rm c}^0}{N_{\rm c}^0}$	$M_{\rm p}^0 = \frac{D_{\rm c}^0}{D_{\rm c}^0 + R_{\rm c}^0}$	$M_{\rm p}^1 = \frac{D_{\rm c}^1}{D_{\rm c}^1 + R_{\rm c}^1}$	$\bar{M}_1 = \frac{\bar{P}_{\rm d}}{\bar{P}_{\rm d} + \bar{P}_{\rm r}}$
total (tested+untested)	IFR $= \frac{D_{\rm c}^0 + D_{\rm u}^0}{N_{\rm c}^0 + N_{\rm u}^0}$	$\mathcal{M}_{\rm p}^0 = \frac{D_{\rm c}^0 + D_{\rm u}^0}{D_{\rm c}^0 + D_{\rm u}^0 + R_{\rm c}^0 + R_{\rm u}^0}$	$\mathcal{M}_{\rm p}^1 = \frac{D_{\rm c}^1 + D_{\rm u}^1}{D_{\rm c}^1 + D_{\rm u}^1 + R_{\rm c}^1 + R_{\rm u}^1}$	not defined

Table 9.2: **Definitions of the main metrics.** The superscript "0" and "1" denote quantities that are based on the total population (including new infections) and a cohort (excluding new infections), respectively. Quantities with subscript "c" and "u" denote confirmed and untested pools (for example, $N_{\rm u}^0(t)$ is the total number of untested individuals at time t) that must be inferred using other measurements such as random testing. The columns denoted by "w/inf" and "w/o inf" denote the mortalities associated with no new infections and with including new infections, respectively. We have suppressed the time dependences for notational simplicity.

By constructing the accumulated deaths and recoveries associated with $I_{\rm c}(\tau, t)$ and $I_{\rm u}(\tau, t)$, $D_{{\rm c},u}^{0,1}(t)$ and $R_{{\rm c},u}^{0,1}(t)$, respectively, we can define true, whole population mortality ratios as

listed in Table II. For example,

$$D^0_{\mathrm{c,u}}(t) = \int_0^t \mathrm{d}t' \int_0^\infty \mathrm{d}\tau\, \mu_{\mathrm{c,u}}(\tau, t') I_{\mathrm{c,u}}(\tau, t'), \quad R^0_{\mathrm{c,u}}(t) = \int_0^t \mathrm{d}t' \int_0^\infty \mathrm{d}\tau\, \gamma_{\mathrm{c,u}}(\tau, t') I_{\mathrm{c,u}}(\tau, t'), \tag{9.3.4}$$

where $\mu_{\mathrm{c,u}}$ and $\gamma_{\mathrm{c,u}}$ are the death and recovery rates associated with infected individuals who are confirmed and untested, respectively. Analogous expressions arise for $D^1_{\mathrm{c,u}}(t)$ and $R^1_{\mathrm{c,u}}(t)$. If the confirmed and untested populations are further subdivided, the $\mu_{\mathrm{c,u}} I_{\mathrm{c,u}}$ and $\gamma_{\mathrm{c,u}} I_{\mathrm{c,u}}$ integrands would be replaced by a Hadamard (*i.e.*, element-wise) product of two vectors representing subpopulations and their corresponding rates. The populations $I_{\mathrm{c,u}}$ themselves can be found from a specific disease transmission model that also includes a testing process that converts I_{u} to I_{c}.

9.4 Summary and conclusions

The CFR has been predominantly used but appears to evolve in qualitatively similar ways as epidemics evolve. Although $\mathrm{CFR}_{\mathrm{d}}(t, \tau_{\mathrm{res}})$ is based on a delay reflecting the timescales for recovery, in general, there is no clear mechanistic interpretation for using the CFR or IFR as mortality ratios.

Here, we stress that more mechanistically meaningful and interpretable metrics can be readily defined and just as easily estimated from population data as CFRs are. Our proposed mortality ratios for viral epidemics are defined in terms of (i) individual survival probabilities and (ii) population ratios using numbers of deaths and recovered individuals. Both of these measures are based on the within-host evolution of the disease, and in the case of $M^{0,1}_{\mathrm{p}}(t)$, the population-level transmission dynamics. On a single patient level, $\bar{M}_1(t)$ is the metric of interest. However, to estimate this, one needs accurate cohort data, for which few exist for coronavirus. Nonetheless, cumulative population-based mortalities can provide insight.

Among the metrics we describe, $M^1_{\mathrm{p}}(t)$ is structurally closest to the individual mortality $\bar{M}_1(t)$ in that both are independent of disease transmission since new infections are not

counted. Both of these mortality ratios converge after an incubation time $\tau_{\rm inc}$ to a value smaller than or equal to $\mu_1/(\mu_1+\gamma)$ and are best interpreted as approximately the mortality probability *conditioned on being tested positive.* The most accurate estimates of $\bar{M}_1$ can be obtained if we keep track of the fate of cohorts who were confirmed within a small time window in the past. By following only these individuals, one can track how many of them die as a function of time. As more cases arise, one should stratify them according to their estimated times since infection to obtain better statistics for $M_1(\infty)$. With the further spread of SARS-CoV-2 in different countries, data on more individual cases of death and recovery can also be more easily stratified according to other central factors in COVID-19 mortality: age, sex, and health condition. Population heterogeneity and uncertainty in intrinsic disease parameters such as the incubation period and the time τ_1 a patient had been infected before confirmation can affect the mortality measures.

Besides demographic heterogeneity and the highly variable estimates of COVID-19 deaths due to different clinical protocols for assigning cause of death, undertesting also confounds accurate estimation of the true underlying mortality. Infected individuals in the population at large who are untested comprise an unknown population $I_{\rm u}$ which contributes to deaths and recovery, and needs to be factored into the "true" mortalities $\mathcal{M}_{\rm p}^{0,1}$ or the IFR.

These untested/unconfirmed populations can, in principle, be computed from a multicompartment mathematical model for disease transmission and testing. The relevant expressions for $\mathcal{M}_{\rm p}^{0,1}$ are listed in Table II. Even though $M_{\rm p}(t)$ typically overestimates the true mortality, tracking $\bar{M}_1(t)$ or $M_{\rm p}^1(t)$ of an initially confirmed cohort can still provide a reasonable estimate of the mortality ratio, especially if untested infected individuals die at the same rate as confirmed individuals.

CHAPTER 10

Controlling epidemics through optimal allocation of test kits and vaccine doses across networks

This is the Accepted Manuscript version of an article accepted for publication in *IEEE Transactions on Network Science and Engineering*, **9**, pp. 1422-1436, (2021). It is an open-access paper. The Version of Record is available online at [10.1109/TNSE.2022.3144624].

10.1 Introduction

Limiting the spread of novel pathogens such as SARS-CoV-2 requires efficient testing [ABM20, YAT20] and quarantine strategies [QCH21], especially when vaccines are not available or effective [SDW22]. Even if effective vaccines are available at scale, their population-wide distribution is a complex and time-consuming endeavor, influenced by, for example, age-structure [MHD21, M98, ZNL21], vaccine hesitancy [PG21], and different objectives [ADB21].

Until a sufficient level of immunity within a population is reached, distancing and quarantine policies can also be used to help slow the spread and evolutionary dynamics [CDC21] of infectious diseases. Epidemic modeling and control-theoretic approaches are useful for identifying both efficient testing and vaccination policies. For an epidemic model of SARS-CoV-2 transmission, Pontryagin's maximum principle (PMP) has been used to derive optimal distancing and testing strategies that minimize the number of COVID-19 cases and intervention costs [CS21, NLV21]. Optimal control theory has also been applied to a multi-objective control problem that uses isolation and vaccination to limit the epidemic size and duration [BBD19]. These recent investigations describe the underlying infectious disease dynamics through compartmental models without underlying network structure, meaning that all interactions among different individuals are assumed to be homogeneous.

Multicompartment models that may be associated with contact networks have been investigated. For example, optimal vaccination strategies have been derived for a rapidly spreading disease in a highly mobile multi-compartment susceptible-infected-recovered (SIR) model using PMP [OM00]. The application of optimal control methods and PMP to heterogeneous node-based susceptible-infected-recovered-susceptible (SIRS) models were also studied in the context of multiplex networks [WXC21] and rumor spreading [LB20].

Complementing these control-theory-based investigations, reinforcement learning (RL) has been recently used to identify infectious high-degree nodes ("superspreaders") in temporal networks [WSD20]. It has been found that RL was able to outperform intervention policies derived from purely structural node characterizations that are, for instance, based

on centrality measures [WSD20]. However, these RL methods could only be applied to rather small networks with about 400 nodes. For social networks describing much larger populations, early work by May and Anderson employed effective degree models to study the population-level dynamics of human immunodeficiency virus (HIV) infections [MA88]. These degree-based models and later variants [BBP04, PV01a, PV01b] did not account for degree correlations. Effective degree models for susceptible-infected-susceptible (SIS) dynamics with degree correlations were derived in [BPV03] and applied to SIR dynamics in [BBP05]. A further generalization of these methods to model SIR dynamics with networked and well-mixed transmission pathways was presented in [KGK06]. For a detailed summary of degree-based epidemic models, see [LMD11].

In this work, we focus on formulating both optimal control and RL-based target policies on a degree-based epidemic model [New18] that is constrained only by the maximum degree and not by the system size (*i.e.*, number of nodes). We construct effective control strategies to slow down disease spread across heterogeneous network models which include both degree distributions and higher-order correlations of the degree distribution. Our approach is not limited by size as agent-based models are [WSD20], is simpler because we do not resolve interpersonal contact times or other individual details, and is thus easier to solve. On the other hand, unlike simple multi-compartment epidemic control models [CS21, BBD19], we take into account a heterogeneous contact network and control measures that depend on both time and node degree.

In the next section, we propose and justify a degree-based epidemic, testing, and quarantining model. An optimal control framework for this model is presented in Sec. 10.3 and, given limited testing resources, an optimal testing strategy is computed. We extend the same underlying disease model to include vaccination in Sec. 10.4 and derive optimal vaccination strategies that minimize infection given a limited vaccination rate. We summarize our results and discuss how they depend on network and dynamical features of the model in Sec. 10.5. For comparison, we also present in Appendix A.5.3 a reinforcement-learning-based algorithm that can approximate optimal testing strategies for the model introduced in Sec. 10.2.

Finally, we implemented a stochastic Monte-Carlo simulation of disease transmission, testing, and vaccination on networks. By using the optimal strategies computed using the PMP on ODE-based deterministic models, we find significant differences in the stochastic model. In Appendix A.5.4, we show that these differences arise from higher correlations in network connectivity that arise in the discrete stochastic model used.

10.2 Degree-based epidemic and testing model

For the formulation of optimal testing policies that allocate testing resources to different individuals in a contact network, we adopt an effective degree model of SIR dynamics with testing in a static network of N nodes. Nodes represent individuals, and edges between nodes represent corresponding contacts. Therefore, the degree of a node represents the number of its contacts. If K is the maximum degree across all nodes, we can divide the population into K distinct subpopulations, each of size N_k ($k = 1, 2, \ldots, K$) such that all nodes in the k^{th} group have degree k. Therefore, $N = \sum_{k=1}^{K} N_k$.

In our epidemic model, we distinguish between untested and tested infected individuals. Let $S_k(t)$, $I_k^{\text{u}}(t)$, $I_k^*(t)$, and $R_k(t)$ denote the numbers of susceptible, untested infected, tested infected, and recovered nodes with degree k at time t, respectively. Since these subpopulations together represent the entire population (the total number of nodes N), both N and N_k are constants in our model. Their values satisfy the normalization condition $S_k + I_k^{\text{u}} + I_k^* + R_k = N_k$. The corresponding fractions are

$$\begin{aligned} s_k(t) &= S_k(t)/N, \quad i_k^{\text{u}}(t) = I_k^{\text{u}}(t)/N, \\ i_k^*(t) &= I_k^*(t)/N, \quad r_k(t) = R_k(t)/N, \end{aligned} \tag{10.2.1}$$

such that $\sum_k (s_k + i_k^{\text{u}} + i_k^* + r_k) = 1$. Using an effective-degree approach [MA88, KGK06],

we describe the evolution of the above subpopulations by

$$\frac{\mathrm{d}s_k(t)}{\mathrm{d}t} = - ks_k(t) \sum_{\ell=1}^{K} \frac{P(\ell|k)}{P(\ell)} \big(\beta^{\mathrm{u}} i_\ell^{\mathrm{u}}(t) + \beta^* i_\ell^*(t)\big), \tag{10.2.2}$$

$$\begin{aligned}\frac{\mathrm{d}i_k^{\mathrm{u}}(t)}{\mathrm{d}t} =& ks_k(t) \sum_{\ell=1}^{K} \frac{P(\ell|k)}{P(\ell)} \big(\beta^{\mathrm{u}} i_\ell^{\mathrm{u}}(t) + \beta^* i_\ell^*(t)\big) \\ & - \gamma^u i_k^{\mathrm{u}}(t) - \frac{f_k(t)}{N_k} i_k^{\mathrm{u}}(t),\end{aligned} \tag{10.2.3}$$

$$\frac{\mathrm{d}i_k^*(t)}{\mathrm{d}t} = - \gamma^* i_k^*(t) + \frac{f_k(t)}{N_k} i_k^{\mathrm{u}}(t), \tag{10.2.4}$$

$$\frac{\mathrm{d}r_k(t)}{\mathrm{d}t} = \gamma^{\mathrm{u}} i_k^{\mathrm{u}}(t) + \gamma^* i_k^*(t), \tag{10.2.5}$$

where $P(\ell) = N_\ell/N$ is the degree distribution. $P(\ell|k)$ is the conditional probability that a chosen node with degree k is connected to a node with degree ℓ. By defining $E_{\ell,k}$ as the number of edges connecting a node with degree k with another node with degree ℓ in a given network, the conditional probability can be directly evaluated as $P(\ell|k) = E_{\ell,k}/(kN_k)$. Our degree-based formulation of SIR dynamics with testing, Eqs. (10.2.2)–(10.2.5), is an approximation of the full node-based dynamics assuming that nodes of the same degree are equally likely to be infected at any given time [New18].

Susceptible individuals become infected through contact with untested and tested infected individuals at rates β^{u} and β^*, respectively. Untested and tested infected individuals recover at rates γ^{u} and γ^*, respectively. Differences in the recovery rates γ^{u} and γ^* reflect differences in disease severity and treatment options for untested and tested infected individuals. Once recovered, individuals develop long-lasting immunity that protects them from reinfection. Temporary immunity can be easily modeled by using an SIS-type model with or without delays. Reduced transmissibility of tested infected (and potentially quarantined) individuals corresponds to setting $\beta^* \ll \beta^{\mathrm{u}}$.

The testing rate of nodes with degree k is defined as $f_k(t)$, such that $f_k(t)\Delta t$ is the total number of tests given to nodes with degree k in the time window Δt. Tests given to recovereds, susceptibles, and already-tested infecteds do not lead to quarantining and will

not affect the disease dynamics. However, a fraction $I_k^{\mathrm{u}}/(S_k+I_k^{\mathrm{u}}+I_k^*+R_k) \equiv I_k^{\mathrm{u}}/N_k$ of these $f_k(t)\Delta t$ tests will be administered to untested infecteds. Once infected nodes have been identified by testing, they can be quarantined and removed from the disease transmission dynamics. If infected individuals who already have been tested strictly avoid future testing, more tests will be available for the other subpopulations, increasing the rate at which the remaining untested infecteds will be tested. In this case, the fraction of tests administered to untested infecteds is modified: $I_k^{\mathrm{u}}/(S_k + I_k^{\mathrm{u}} + R_k) \equiv I_k^{\mathrm{u}}/(N_k - I_k^*)$. After normalizing by the total population N to write tested fractions in terms of Eqs. (10.2.1), the testing term becomes $-f_k(t)i_k^{\mathrm{u}}/N_k$ (Eq. (10.2.3)) or $-f_k(t)i_k^{\mathrm{u}}/[N_k(1 - I_k^*/N_k)]$, respectively.

Biased testing can also be represented by using a testing fraction of the form $I_k^{\mathrm{u}}e^b/(I_k^{\mathrm{u}}e^b + S_k + I_k^* + R_k)$, where $b > 0$ increases the fraction of tests given to infecteds. To correct for false-positive tests, Eqs. (10.2.2)–(10.2.5) can be modified by including an additional term that transfers the I_k^* population back to S_k. False negatives can be accounted for by a reduction in $f_k(t)/N_k$. For a detailed overview of statistical models that account for testing errors and bias, see [BDC21, BDC22].

What remains is to assign network structures, extract $P(\ell|k)$ from them, and determine reasonable parameter values before calculating the optimal testing protocol $f_k(t)$. We apply our disease-control framework to (i) a Barabási-Albert (BA) network [BA99, AB02] and (ii) a stochastic block model (SBM) [HLL83] with four communities and a probability matrix

$$P = 10^{-4}\begin{pmatrix} 1 & 2 & 2 & 2 \\ 2 & 4 & 2 & 2 \\ 2 & 2 & 5 & 2 \\ 2 & 2 & 2 & 3 \end{pmatrix}. \tag{10.2.6}$$

These two network types exhibit properties, such as hub nodes with high degrees and community structure, that are observable in real-world contact networks [BNM13, ZLZ12]. In the construction of the BA network, each new node is connected to 2 existing nodes. Figure 10.1(a) shows the degree distribution of a 99,817-node BA network that we use in this

study.

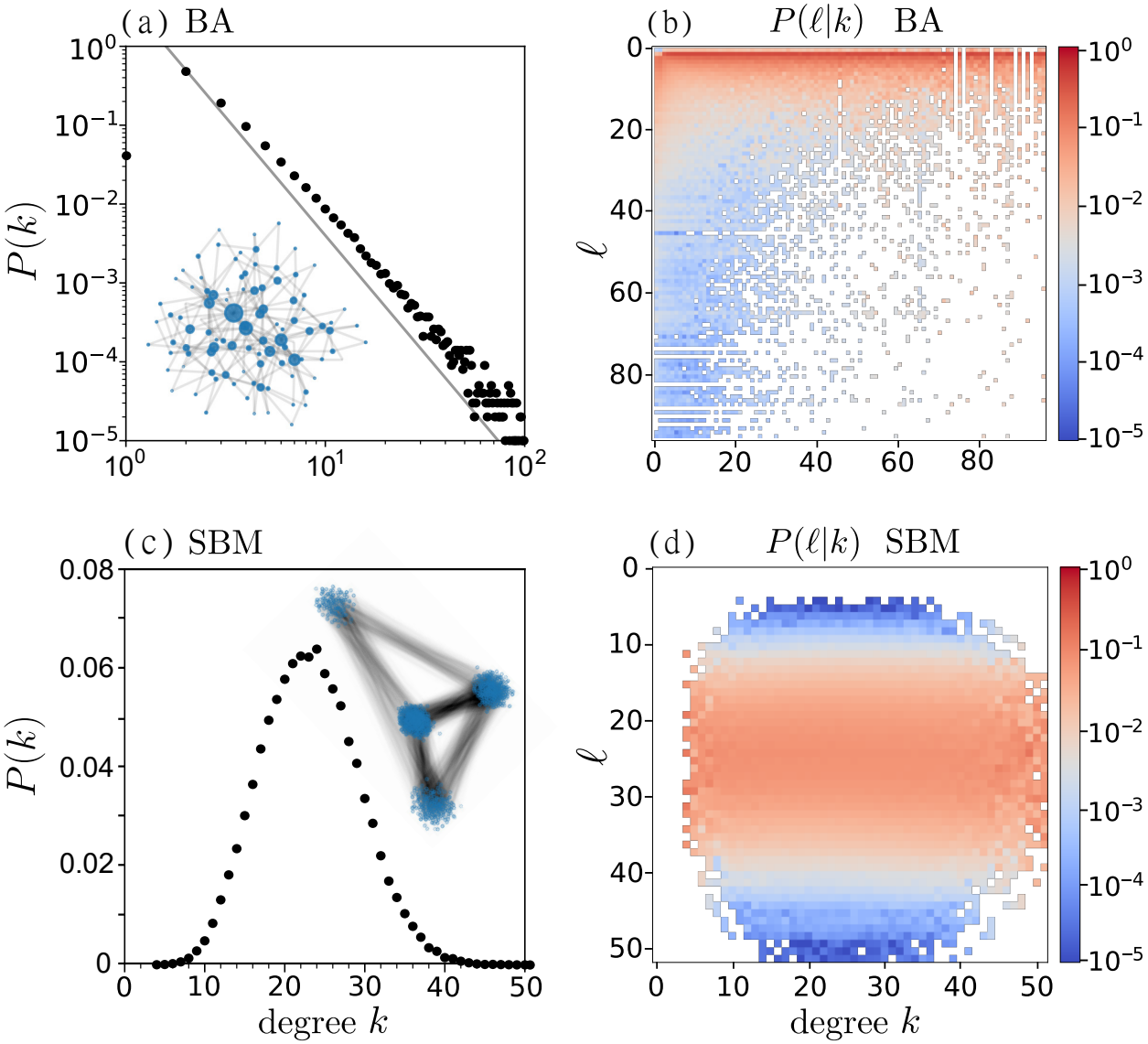

Figure 10.1: Degree distribution of a Barabási–Albert network and a stochastic block model. (a) The degree distribution of a Barabási–Albert network with 99,817 nodes. To generate the network, we start with a dyad and iteratively add new nodes until we reach 100,000 nodes. Each new node has 2 edges that connect it to existing nodes using the linear preferential attachment. Isolated nodes or nodes with degrees larger than 100 [BNM13] are then removed from the network. The grey solid line is a guide-to-the-eye with slope -3 [AB02]. For illustration, the inset shows a realization of a Barabási–Albert network with 100 nodes. Node size scales with their betweenness centrality. (b) The conditional probability $P(\ell|k)$ associated with the Barabási–Albert network generated in (a). (c) The degree distribution of a stochastic block model with four blocks and 100,000 nodes. The inset shows a realization of a stochastic block model with 800 nodes, but using the same block probability matrix. (d) The conditional probability $P(\ell|k)$ associated with the SBM. In both (b) and (d), all elements that are strictly zero are uncolored.

A heatmap of the conditional degree distribution matrix of the BA network with the degree distribution $P(k)$ shown in Fig. 10.1(a) is given in Fig. 10.1(b). The degree distribution and the conditional degree distribution matrix of the 100,000-node SBM network are shown in Figs. 10.1(c) and (d), respectively. Taking into account empirical findings on the degree distributions in real-world contact networks [BNM13], we use a degree cutoff of $k \leq K = 100$. We will use the specific configurations of the BA and SBM networks shown in Fig. 10.1 for our subsequent analysis of Eqs. (10.2.2)-(10.2.5).

Next, to constrain the parameter values, we first invoke estimates of the basic reproduction number (*i.e.*, the average number of secondary cases that results from one case in a completely susceptible population), which for a network model is defined as [DW02, DHM90]

$$\mathcal{R}_0 = \rho(JV^{-1}) \tag{10.2.7}$$

in which $\rho(\cdot)$ is the largest eigenvalue (spectral radius), $V \equiv \text{diag}(\gamma^{\text{u}}) \in \mathbb{R}^{K\times K}$, and $J \in \mathbb{R}^{K\times K}$ is the Jacobian of the linearized dynamical system (Eqs. (10.2.2) and (10.2.3)) about the disease-free state with $s_k(t = 0) = N_k/N$ and $f_k = 0$ corresponding to the initial, untested, and uncontrolled spread of the infection:

$$J_{ij} = iP(j|i)\frac{N_i}{N_j}\beta^{\text{u}}, \quad i, j \leq K. \tag{10.2.8}$$

This "next-generation" method associates $\mathcal{R}_0$ with the largest eigenvalue inherent to the dynamical system. Additional expressions for $\mathcal{R}_0$ for an uncorrelated degree network are given in Appendix A.5.1.

Empirically, the basic reproduction number for COVID-19 varies across different regions. For the early outbreak in Wuhan [WYW20], $\mathcal{R}_0$ was estimated to be 3.49, while for the early outbreak in Italy $\mathcal{R}_0 \sim 2.43 - 3.10$ [DC20]. Here we set $\mathcal{R}_0 = 4.5$ which was suggested in [KMB20] as the basic reproduction number of early COVID-19 spread in the absence of any intervention. For a given value of the recovery rate γ^{u} of untested individuals, which can be inferred from empirical data [KMB20, BRB20], we determine the transmissibility

β^{u} by numerically solving $\mathcal{R}_0(\beta^{\mathrm{u}}) = 4.5$ for β^{u}. Our source codes are publicly available at https://gitlab.com/ComputationalScience/epidemic-control.

10.3 Allocating limited testing resources

Without any testing constraints, it would be most effective for disease control to use a testing rate $f_k(t)$ sufficiently large to keep the fraction of untested individuals, $i_k^{\mathrm{u}}(t)$, close to zero. In general, the testing rates are constrained by

$$f_k^{\min} \leq \frac{f_k(t)}{N_k} \leq f_k^{\max}, \tag{10.3.1}$$

and the total testing rate is also bounded by availability and logistics of testing

$$\sum_{k=1}^{K} f_k(t) = F(t). \tag{10.3.2}$$

The goal is to determine, under these constraints, the function $f_k(t)$ or $f_k(t)/N_k$ that most effectively reduces the total number of infections. In practice, high-degree nodes (*e.g.*, highly social individuals) might be subject to more testing (and quarantining if positive) than low-degree nodes because of their higher expected rate of infecting others. This rationale translates to $f_k(t)/N_k > f_{k'}(t)/N_{k'}$ if $k > k'$. In our numerical experiments, we use sufficiently broad bounds of $f_k(t)$ and set $f_k^{\min} = f_{\min}$ and $f_k^{\max} = f_{\max}$.

To minimize the number of total infections over time, while simultaneously stressing the importance of reducing early infections, we define a loss function as

$$L(T) = \int_0^T \mathrm{d}t\, \delta^t \sum_{k=1}^{K} k s_k(t) \sum_{\ell=1}^{K} \frac{P(\ell|k)}{P(\ell)} \big(\beta^{\mathrm{u}} i_\ell^{\mathrm{u}}(t) + \beta^* i_\ell^*(t)\big), \tag{10.3.3}$$

where $\delta \in (0, 1]$ denotes a discount factor, which describes how we balance between minimizing current infections and future infections. The smaller the parameter δ, the less attention we pay to future infections, and the more we focus on reducing early infections. For exam-

ple, medical resources can better handle confirmed patients and new treatments can be given time to develop if the number of infections are spread over longer time periods. These effects can be effectively incorporated into the loss function by using $\delta < 1$. Minimizing the loss Eq. (10.3.3) is equivalent to minimizing the number of infections, weighted by the discount factor δ^t, in the time horizon $[0, T]$.

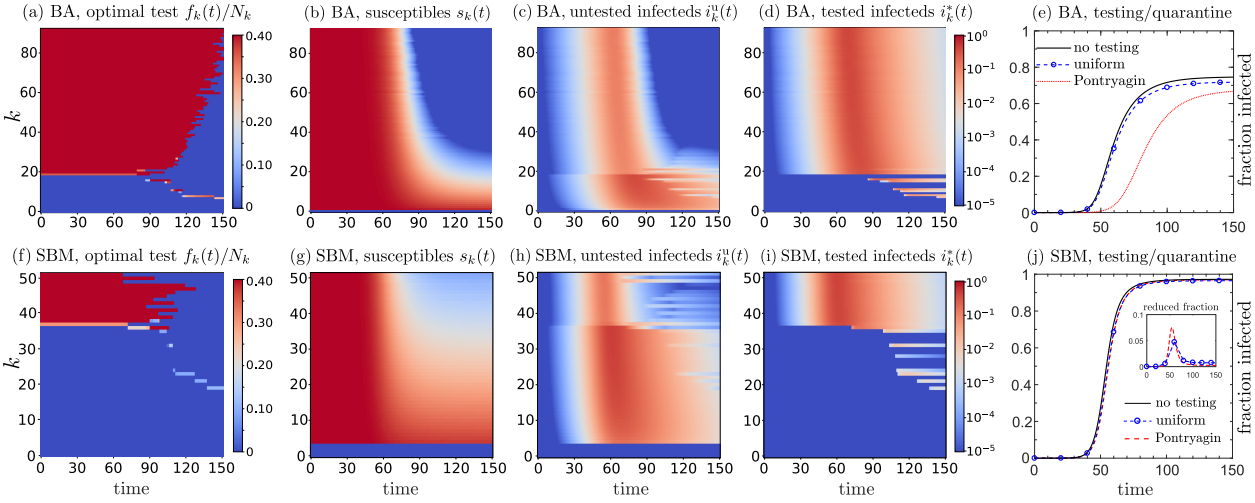

Figure 10.2: Optimal testing and quarantining strategy for $T = 200$ and discount factor $\delta = 0.95$. We plot the optimal strategies and the corresponding susceptible, untested infected, and tested infected fractions at each degree k across time $t = n\Delta t$. (a) A heatmap of the PMP-optimal testing strategy (see Alg. 7) for the BA network. The corresponding populations of degree-k susceptibles, untested infecteds, and tested infecteds are plotted in (b-d), respectively. (e) Time-evolution of the total fraction infected $1 - \sum_{k=1}^{K} s_k(t)$ under the PMP-optimal testing strategy (dashed red). The fractions infected under hypothetical uniform testing (dashed blue/circle) and no testing (black) scenarios are shown for comparison. For the BA network, optimal testing both delays and suppresses epidemic spreading more effectively than uniform testing. The bottom row (f-j) shows analogous results for the SBM network. Panels (f-i) show the corresponding optimal testing rates, susceptible, untested infected, and untested infected populations with degree k as a function of time. Panel (j) shows the fraction infected as a function of time. Although optimal testing and quarantining reduce the fraction infected relative to uniform or no testing, its effects are only modestly better. Given the same testing budget constraint, the effects of optimal testing strategies are greater in the BA network because its distribution of node degrees is more heterogeneous and testing and quarantining high-degree nodes can more effectively control disease spread. However, since the node degree distribution in the SBM network is sharply peaked, an optimal testing strategy is less effective overall.

To search for the optimal testing function $f_k(t)$ that minimizes Eq. (10.3.3), we invoke

Pontryagin's maximum principle (PMP) and construct the associated Hamiltonian

$$\begin{aligned} H &= \delta^t \sum_{k=1}^{K} k s_k(t) \sum_{\ell=1}^{K} \frac{P(\ell|k)}{P(\ell)} \big(\beta^{\mathrm{u}} i_\ell^{\mathrm{u}}(t) + \beta^* i_\ell^*(t)\big) \\ &\quad + \sum_{k=1}^{K} \left(\lambda_k^s \frac{\mathrm{d}s_k(t)}{\mathrm{d}t} + \lambda_k^{\mathrm{u}} \frac{\mathrm{d}i_k^{\mathrm{u}}(t)}{\mathrm{d}t} + \lambda_k^* \frac{\mathrm{d}i_k^*(t)}{\mathrm{d}t} \right) \\ &= \sum_{k=1}^{K} (\delta^t - \lambda_k^s + \lambda_k^{\mathrm{u}}) k s_k(t) \sum_{\ell=1}^{K} \frac{P(\ell|k)}{P(\ell)} \big(\beta^{\mathrm{u}} i_\ell^{\mathrm{u}}(t) + \beta^* i_\ell^*(t)\big) \\ &\quad + \sum_{k=1}^{K} \Big[\frac{f_k(t)}{N_k} (\lambda_k^* - \lambda_k^{\mathrm{u}}) i_k^{\mathrm{u}}(t) - \gamma^{\mathrm{u}} i_k^{\mathrm{u}}(t) \lambda_k^{\mathrm{u}} - \gamma^* i_k^*(t) \lambda_k^* \Big], \end{aligned} \tag{10.3.4}$$

where λ_k^s, λ_k^{u}, and λ_k^* are adjoint variables associated with s_k, i_k^{u}, and i_k^*, respectively. PMP states that a necessary condition for the loss-minimizing control $f_k(t)$ is that it minimizes H (or maximizes $-H$) at every time point t. This method of optimal control has been applied to many other contexts, including control of economic growth [AK07]. In our problem, applying PMP under the total budget constraint $\sum_{k=1}^{K} f_k(t) = F(t)$, we explicitly find the minimizing testing function $(f_k^*) = \mathrm{argmin}_f H$, which we will assume to be optimal control that minimizes $L(T)$. The dynamics for $(\lambda_k^s, \lambda_k^{\mathrm{u}}, \lambda_k^*)$ obey

$$\begin{aligned} \frac{\mathrm{d}\lambda_k^s}{\mathrm{d}t} &= -\frac{\partial H}{\partial s_k} = -\delta^t k \sum_{\ell=1}^{K} \frac{P(\ell|k)}{P(\ell)} \big(\beta^{\mathrm{u}} i_\ell^{\mathrm{u}}(t) + \beta^* i_\ell^*(t)\big) \\ &\quad + \lambda_k^s k \sum_{\ell=1}^{K} \frac{P(\ell|k)}{P(\ell)} \big(\beta^{\mathrm{u}} i_\ell^{\mathrm{u}}(t) + \beta^* i_\ell^*(t)\big) \\ &\quad - \lambda_k^{\mathrm{u}} k \sum_{\ell=1}^{K} \frac{P(\ell|k)}{P(\ell)} \big(\beta^{\mathrm{u}} i_\ell^{\mathrm{u}}(t) + \beta^* i_\ell^*(t)\big), \end{aligned} \tag{10.3.5}$$

$$\begin{aligned} \frac{\mathrm{d}\lambda_k^{\mathrm{u}}}{\mathrm{d}t} &= -\frac{\partial H}{\partial i_k^{\mathrm{u}}} = -\frac{\beta^{\mathrm{u}}}{P(k)} \sum_{j=1}^{K} P(k|j) s_j(t) (\delta^t - \lambda_j^s + \lambda_j^{\mathrm{u}}) \\ &\quad + \gamma^{\mathrm{u}} \lambda_k^{\mathrm{u}} + \frac{f_k(t)}{N_k} (\lambda_k^{\mathrm{u}} - \lambda_k^*), \end{aligned} \tag{10.3.6}$$

$$\begin{aligned}\frac{\mathrm{d}\lambda_k^*}{\mathrm{d}t} &= -\frac{\partial H}{\partial i_k^*} \\ &= -\frac{\beta^*}{P(k)}\sum_{j=1}^{K} P(k|j)s_j(t)(\delta^t - \lambda_j^s + \lambda_j^{\mathrm{u}}) + \gamma^*\lambda_k^*,\end{aligned} \tag{10.3.7}$$

with end conditions $\lambda_k^s(T) = \lambda_k^{\mathrm{u}}(T) = \lambda_k^*(T) = 0$. To minimize H with respect to the testing rates $f_k(t)$, we have to minimize the term

$$\sum_{k=1}^{K} \frac{f_k(t)}{N_k}(\lambda_k^* - \lambda_k^{\mathrm{u}})i_k^{\mathrm{u}}(t) \tag{10.3.8}$$

given the budget constraints Eqs. (10.3.1) and (10.3.2). Hence, after giving each subpopulation the minimal testing resources $f_{\min}N_k$, we maximize the testing rates $f_k(t)$ with the smallest coefficients $(\lambda_k^* - \lambda_k^{\mathrm{u}})i_k^{\mathrm{u}}(t)/N_k$ of $f_{\max}N_k$ as long as a sufficient testing budget is available. In other words, we should give testing resources to those groups presumed to be at the highest risk, as quantified by the quantity $(\lambda_k^* - \lambda_k^{\mathrm{u}})i_k^{\mathrm{u}}(t)/N_k$. We use the PMP-based algorithm outlined in Appendix A.5.2 to iteratively calculate the loss function (10.3.3) and optimal testing strategy.

In accordance with empirical data on COVID-19 patients [BRB20, WSJ20, HM22], we set $\gamma = \gamma^{\mathrm{u}} = \gamma^* = (1/14)/\text{day}$ and $\beta^* = \beta^{\mathrm{u}}/10$. The transmissibility of untested individuals, β^{u}, is calculated according to Eq. (10.2.7) as $\beta^{\mathrm{u}} = 0.0411/\text{day}$ for the BA network and $\beta^{\mathrm{u}} = 0.0130/\text{day}$ for the SBM network. We set the discount factor $\delta = 0.95$ so that initial infections contribute more to the loss function (10.3.3). The total daily number of SARS-CoV-2 tests in the US after an initial ramping-up phase in 2020 is about 0.6%/day [BDC22]. Hence, we set

$$\sum_k f_k(t) = 0.006N, \tag{10.3.9}$$

and $f_{\min} = 0, f_{\max} = 0.4N_k$. As the initial condition, we use

$$\begin{aligned} s_k(0) &= P(k) - i_k^{\mathrm{u}}(0), \;\; i_k^*(0) = 0, \\ i_k^{\mathrm{u}}(0) &= 10^{-6}P(k), \qquad r_k(0) = 0, \end{aligned} \tag{10.3.10}$$

corresponding to about 0.1 of an infected individual uniformly distributed on $N \approx 10^5$ susceptible nodes. The optimal testing strategy is supposed to identify those nodes that are most likely to be infected and transmit the disease to others. Upon using $T = 200, \Delta t = 0.1$ and $\delta = 0.95$, we find the optimal testing strategy $f_k(t)/N_k$ for our BA network and plot it in Fig. 10.2(a). Here Eqs. (10.2.2)–(10.2.5) and (10.3.5)–(10.3.7) are solved using an improved Euler method. For the BA network, the value of the loss function defined in Eq. (10.3.3) is $L(T = 200) = 0.0109$ under the optimal testing strategy, while it is $L(T = 200) = 0.0325$ under uniform testing

$$f_k = F_0 \frac{N_k}{N}. \tag{10.3.11}$$

Figs. 10.2(b-d) show the associated populations under optimal testing, while (e) shows the dynamics of the fraction of nodes infected, $1 - \sum_{k=1}^{K} s_k(t)$. The disease spread under optimal testing is significantly slowed relative to the no testing (black) and uniform testing (dashed blue/circle) cases. Fig. 10.2(f) plots the optimal testing rate for the SBM network. Panels (g-i) show the corresponding subpopulations, and panel (j) plots the fraction of nodes infected under PMP-optimal, uniform, and no-testing conditions. For the SBM network, $L(T = 200) = 0.0564$ under the optimal strategy and $L(T = 200) = 0.0571$ under the uniform testing strategy, suggesting that the PMP approach yields better solutions than uniform testing. However, the improvement is modest and the SBM network is rather insensitive to testing and quarantining. The slight improvement from testing is shown by the *reduction* in the fraction infected relative to the no testing case (inset).

In both networks, nodes with larger degrees are more likely to be tested at the beginning of the outbreak [Figs. 10.2(a,f)], indicating that people with more contacts are more likely to infect others or get infected, and should be given priority to get tested. Yet, in both networks,

as time evolves, the optimal testing strategy tends to shift focus from higher-degree nodes to nodes with smaller degrees because testing those nodes that were infected and have already recovered is not meaningful in terms of disease control.

Comparing Figs. 10.2(e) and (j), we see that the differences between optimal and uniform testing are larger for the BA network compared to the SBM. A possible explanation for this behavior is that in the BA network, the degree distribution $P(k)$ decays algebraically. Therefore, as long as testing focuses primarily on high-degree nodes, the spreading of the disease can be controlled very effectively since the majority of nodes have small degrees and are more unlikely to be infected. On the other hand, for our SBM network, the degrees of most nodes are close to each other and larger than 10, indicating that nodes with a small degree are more likely to be infected compared to the BA network. Even if we use the same uniform testing rates [see Eq. (10.3.11)] in both networks, the proportion of infections in the BA network is less than that in the SBM network.

10.4 Optimal vaccination policy

Optimal vaccination has also been studied within the classic SIR model [GKC17]. However, devising vaccination strategies based on social network structure may provide a more refined and efficient way of administering vaccines and extinguishing an epidemic. Our simple testing model presented in the previous section can be straightforwardly adapted to describe vaccination on a network. The goal is to determine the optimal allocation of vaccine doses to a population with heterogeneous contacts to minimize the impact of the infection across the entire population.

For COVID-19, there are a variety of vaccines that require one or two shots [PRK21]. In our simulations, we assume that the administered vaccine provides full protection after one shot and that a vaccinated individual will instantly leave the susceptible group and enter the recovered group. This means that vaccinated individuals will no longer be infectious and can be treated as "recovered" after receiving one vaccination dose. Furthermore, we assume

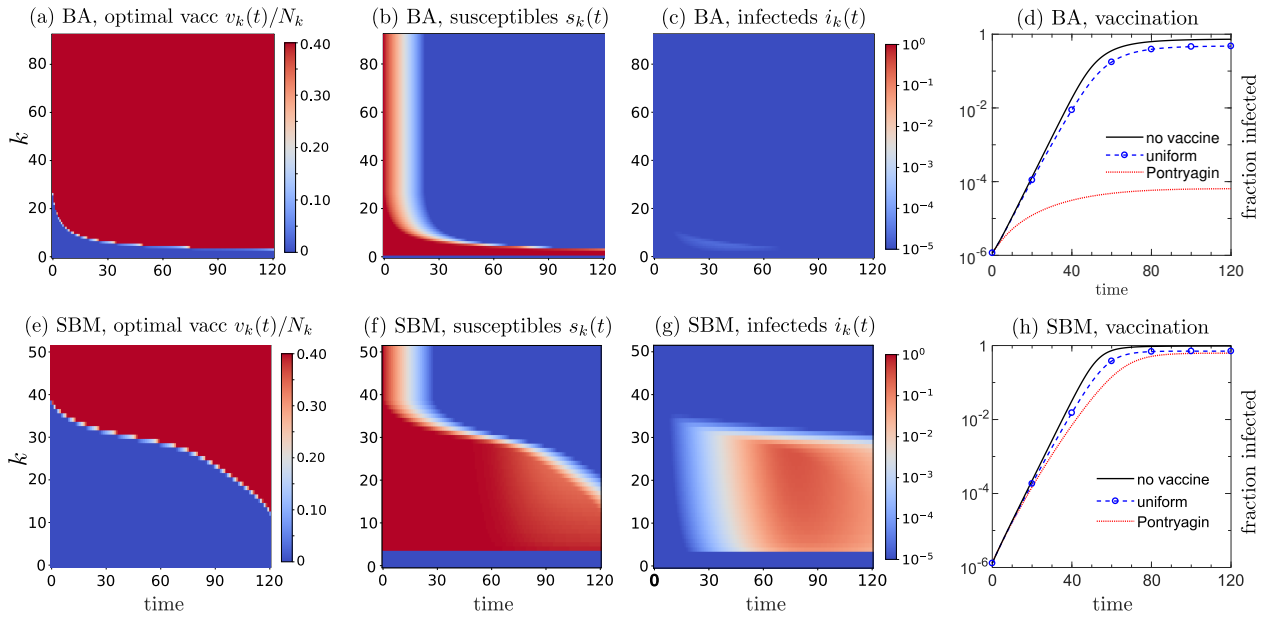

Figure 10.3: Vaccination model optimized for $T = 150$ under different constraints. We plot the optimal strategies and the corresponding susceptible, untested infected, and tested infected fractions at each degree k across time $t = n\Delta t$. (a) Heatmap of the optimal vaccination strategy $v_k(t)/(s_k(t)N_k)$ for the BA network given by Alg. 7. Panels (b,c) show the corresponding susceptible and infected subpopulations $s_k(t)$ and $i_k(t)$, while (d) plots the fraction infected as a function of time, derived from solving Eqs. (10.4.1)–(10.4.3) under optimal vaccination using a discount factor $\delta = 0.95$. The dashed red curve indicates the fraction infected under optimal vaccination. For comparison, the infected population with no vaccination (solid black) and constant, uniform (dashed blue/circles) vaccination are also plotted and show how optimizing vaccination significantly suppresses infectivity. Panels (e-h) show the corresponding quantities for the SBM network. Optimal vaccination is less effective at decreasing infection in the SBM network than in the BA network, again because of the SBM's peaked (more homogeneous) node degree distribution. Note from the logarithmic scale that vaccination is qualitatively more effective in reducing infections than testing and quarantining.

that only susceptible persons will be vaccinated. Other mechanisms such as prime-boost protocols and time delays between vaccination and onset of immune response can also be accounted for in similar models as detailed in [BN21].

We reformulate Eqs. (10.2.2)-(10.2.5) to study optimal vaccination protocols that are constrained by vaccine supplies in a heterogeneous population. For simplicity, we do not take into account the effect of testing and quarantining when devising optimal vaccinating strategies, although testing and vaccination can be performed concurrently. The resulting

rate equations are

$$\frac{\mathrm{d}s_k(t)}{\mathrm{d}t} = -\beta k s_k(t) \sum_{\ell=1}^{K} \frac{P(\ell|k)}{P(\ell)} i_\ell(t) - \frac{v_k(t)}{N}, \tag{10.4.1}$$

$$\frac{\mathrm{d}i_k(t)}{\mathrm{d}t} = \beta k s_k(t) \sum_{\ell=1}^{K} \frac{P(\ell|k)}{P(\ell)} i_\ell(t) - \gamma i_k(t), \tag{10.4.2}$$

$$\frac{\mathrm{d}r_k(t)}{\mathrm{d}t} = \gamma i_k(t) + \frac{v_k(t)}{N}, \tag{10.4.3}$$

where $v_k(t)$ is the rate of vaccination of susceptibles with degree k at time t. Once vaccinated, susceptibles become "recovered" because they are immunized and no longer susceptible to the infection. The total rate of administering vaccines at time t is defined as

$$\sum_{k=1}^{K} v_k(t) = V(t). \tag{10.4.4}$$

In other words, in time increment Δt at time t, we can administer only $V(t)\Delta t$ doses. Eq. (10.4.1) assumes that vaccination is resource-limited and that the rate of protecting susceptibles is proportional only to the rate $v_k(t)$ of administering vaccines. In addition, we assume that the vaccination rates for different subpopulations are confined to the interval

$$v_{\min} \leq \frac{v_k(t)}{N s_k(t)} \leq v_{\max}, \tag{10.4.5}$$

where $v_{\min}, v_{\max} \in [0, 0.4]$/day are minimum and maximum vaccination rates. Note that vaccines are allocated only to susceptibles, while tests are typically given to individuals of all categories: susceptible, infected, and recovered, according to their relative proportions. To formulate the vaccine distribution problem in a heterogeneous contact network, we use the following loss function

$$L(T) = \int_0^T \mathrm{d}t\, \delta^t \sum_{k=1}^{K} k s_k(t) \sum_{\ell=1}^{K} \frac{P(\ell|k)}{P(\ell)} \beta(t) i_\ell(t), \tag{10.4.6}$$

to minimize the total number of infections over time (with a constant discount factor $\delta \in$

$(0, 1]$) by appropriately distributing vaccines to groups with different degrees k at different rates.

To minimize the loss function (10.4.6), we construct the Hamiltonian

$$\begin{aligned} H =&\beta\delta^t \sum_{k=1}^{K} k s_k(t) \sum_{\ell=1}^{K} \frac{P(\ell|k)}{P(\ell)} i_\ell(t) \\ &+ \sum_{k=1}^{K} \Big(\lambda_k^s \frac{\mathrm{d}s_k(t)}{\mathrm{d}t} + \lambda_k^i \frac{\mathrm{d}i_k(t)}{\mathrm{d}t} \Big) \\ =&\beta \sum_{k=1}^{K} (\delta^t - \lambda_k^s + \lambda_k^i) k s_k(t) \sum_{\ell=1}^{K} \frac{P(\ell|k)}{P(\ell)} i_\ell(t) \\ &+ \sum_{k=1}^{K} \Big(\frac{v_k(t)}{N} \lambda_k^s(t) - \gamma i_k(t) \Big), \end{aligned} \tag{10.4.7}$$

where λ_k^s and λ_k^i are the adjoint variables satisfying the differential equations

$$\frac{\mathrm{d}\lambda_k^s}{\mathrm{d}t} = -\frac{\partial H}{\partial s_k} = -\beta k \sum_{\ell=1}^{K} \frac{P(\ell|k)}{P(\ell)} i_\ell(t) (\delta^t - \lambda_k^s + \lambda_k^i) \tag{10.4.8}$$

$$\begin{aligned} \frac{\mathrm{d}\lambda_k^i}{\mathrm{d}t} &= -\frac{\partial H}{\partial i_k} \\ &= -\frac{\beta}{P(k)} \sum_{\ell=1}^{K} P(k|\ell) \ell s_\ell(t) (\delta^t - \lambda_\ell^s + \lambda_\ell^i) + \gamma \lambda_k^s. \end{aligned} \tag{10.4.9}$$

Again, PMP explicitly generates the control that minimizes Eq. (10.4.7) for every t, which we assume also minimizes the target loss function Eq. (10.4.6). From the constraints (10.4.4) and (10.4.5), minimizing the Hamiltonian is achieved by giving vaccination doses to those subpopulations with the smallest λ_k^s. We use the same Alg. 7 to solve the minimization problem (10.4.6) numerically and obtain the optimal strategy.

In the US, about two million doses of SARS-CoV-2 vaccines were delivered in May 2021 [MRO21], most of which were two-dose vaccines. Since approximately 0.3% of the entire US population is fully vaccinated daily, we set $V(t) = 0.003N/\text{day}$, $v_{\min} = 0/\text{day}$, and $v_{\max} = 0.4/\text{day}$ in the constraint (10.4.5). The infection rates β are set to be $0.0411/\text{day}$ for

the BA network and 0.0130/day for the SBM network, and the recovery rate $\gamma = (1/14)$/day. For comparison, we also simulate a vaccination strategy with a uniform vaccination rate

$$v_k(t) = \frac{s_k(t)V(t)}{\sum_{k=1}^{K} s_k(t)}. \tag{10.4.10}$$

In all simulations, we use the following initial condition:

$$i_k(0) = 10^{-6}P(k),\ r_k(0) = 0, s_k(0) = P(k) - i_k(0). \tag{10.4.11}$$

We plot the PMP-optimal vaccination strategy v_k/N_k in Figure 10.3(a) and the corresponding susceptible and infected k-degree subpopulations $s_k(t)$ and $i_k(t)$ in (b) and (c). We set $T = 150, \Delta t = 0.1$ and we use an improved Euler method to numerically solve Eqs. (10.4.1)–(10.4.3), (10.4.8)–(10.4.9). Alg. 7 is applied (without the infected and tested compartment) to determine the optimal vaccination strategy by the PMP approach. For the BA network, $L(T = 150) = 1.165 \times 10^{-5}$ under the PMP-optimal strategy and $L(T = 150) = 0.01953$ under a uniform vaccination rate. Figure 10.3(d) shows that the optimal vaccination strategy on a BA network significantly reduces the fraction infected compared to the uniform vaccination strategy. Panels (e-h) show the corresponding quantities for the SBM network for which $L(T = 150) = 0.0210$ under the optimal vaccination strategy and $L(T = 150) = 0.0360$ under a constant, uniform vaccination strategy. In both networks, the optimal vaccination strategies obtained via Alg. 7 tend to prioritize those nodes with higher degrees first and eventually expand to those nodes with smaller degrees [see Figs. 10.3(a) and (e)]. As with testing and quarantining, the reduction in the fraction infected by vaccination is greater in the BA network. Since the BA network has a degree distribution with algebraic decay, the effect of the optimal vaccination strategy will be more pronounced than for the SBM, whose nodes have similar degrees.

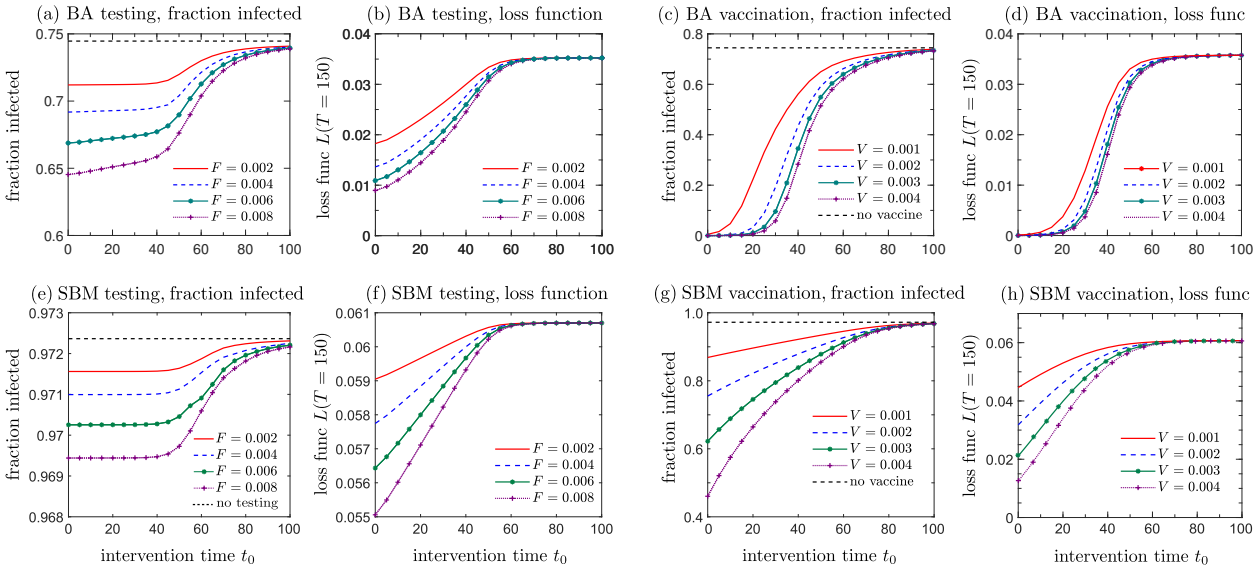

Figure 10.4: Total fraction infected under testing or vaccination model as a function of different intervention starting times t_0. We minimize the corresponding loss function at $T = 150$ and use $\delta = 0.95$. (a) The fraction infected in the BA network as a function of start times for different testing amplitudes F. The total infected fraction is fairly insensitive to intervention starting times, especially for small intervention delays. The effect of delayed vaccination on the fraction infected is shown in (c), with the corresponding loss function shown in (d). For the SBM network, the fraction infected as a function of the testing start time shown in (e) reflects the small effect of testing on the infected population. However, the loss functions shown in (f) are monotonic in the starting time. This implies that an early intervention time on the SBM network is able to "flatten" the curve by postponing infection so even if total infections stay roughly the same when t_0 varies in $\sim [0, 50]$, the earlier the intervention time, the fewer the earlier infections, with little change in the final total infected fraction. The starting time dependence of the fraction infected on an optimally vaccinated SBM network in (g) shows a monotonic and smooth decrease in effectiveness as vaccination is delayed. In (h), the loss function for vaccination on the SBM network also monotonically increases with the start time.

10.5 Discussion

Effective testing and vaccination strategies are an essential part of epidemic management. In this chapter, we derived optimal testing and vaccination policies by applying Pontryagin's maximum principle to a degree-based epidemic model in a heterogeneous contact network. We complemented our analytical results with reinforcement learning (RL) approaches that identify effective policies (see Appendix A.5.3). On occasions when the best optimal strategy can be analytically solved, the controls derived from the Pontryagin's maximum

principle outperform RL-based interventions. However, reinforcement learning is useful for epidemic management problems when an efficient procedure for computing optimal solutions is not available. Our results show that the two approaches can complement each other; preliminary findings from optimal-control analysis can be used to pre-train and restrict the space of possible actions, which may lead to more efficient RL algorithms. In addition to RL-based control strategies, it may also be worthwhile to apply neural ODE control frameworks [ABA22a, BAA22] to resource allocation problems since they have exhibited better performance than RL and numerical adjoint system solvers.

Our analytical results show that optimal testing and vaccination policies under resource constraints initially tend to prioritize nodes with higher degrees to control the spread of the disease. In situations where the number of contacts of individuals is known or can be estimated with reasonable precision, Algs. 7 and 8 may be useful for identifying effective epidemic management strategies. Using our control-theoretic approach, we also explored the relative effectiveness of testing and vaccination under different conditions. If more information on contact patterns of individual nodes is available, it is possible to further refine the proposed policies using interventions that rely not only on node degrees but also on other structural features such as percolation and betweenness centrality [New18, PPH13].

10.5.1 Effects of delayed intervention

First, we consider the effectiveness of interventions as a function of the time between the first infection and the implementation of testing or vaccination. The initial conditions are set to be the same as Eqs. (10.3.10) and (10.4.11). Fig. 10.4 shows the total fraction infected and the loss function at $T = 150$, for both the BA and SBM networks, as a function of intervention starting time t_0. We set $F = F(t)\mathbb{1}_{t>t_0}$ or $V = V(t)\mathbb{1}_{t>t_0}$ and explore the effects of different constant levels of test kits or vaccine availability, $F(t) = 0.002, 0.004, 0.006, 0.008N/\text{day}$ and $V(t) = 0.001, 0.002, 0.003, 0.004N/\text{day}$, respectively. The transmissibility rates $\beta^{\text{u}}, \beta^*, \beta$ and the recovery rates $\gamma^{\text{u}}, \gamma^*, \gamma$ are set to the same values as those used in Section 10.3 for the testing model and those used in Section 10.4 for the vaccination model.

In the BA network, the total infected fraction shown in Fig. 10.4(a) is fairly insensitive to starting times for all testing rates F, especially at small starting times $t_0 \lesssim 50$. However, the loss functions corresponding to all testing rates increase monotonically with the testing starting time t_0, as shown in Fig. 10.4(b). On the other hand, vaccination of a BA network leads to infected fractions that change significantly with delay time, but with an overall vaccination-rate-dependent starting time before which disease spread can be nearly completely suppressed, as shown in (c). For the vaccination model applied to both networks, an earlier intervention time will always lead to fewer infected nodes. Overall, we found that earlier and stronger intervention measures lead to more effective control of disease spread and a smaller loss function defined by Eqs. (10.3.3), (10.4.6). For all cases, the testing loss functions monotonically increase with t_0.

Similarly, for the SBM model, the final infections shown in Fig. 10.4(d) are insensitive to starting times $t_0 \lesssim 50$ for each of the four choices of total testing budgets F. Earlier intervention times t_0 lead to smaller testing loss functions that indicate more effective early-time disease control and fewer early infections (which are followed by larger later infections) than those associated with later start times t_0. Vaccination of the SBM network reveals more smoothly monotonically increasing infected fraction and loss functions and does not display the sigmoidal dependence on intervention time t_0 as exhibited by the infected fraction and loss function for the BA network. For the vaccination model applied to both networks, an earlier intervention time will always lead to fewer infected nodes.

Overall, higher levels of F and V lead to high-k nodes being addressed sooner and total infections can be reduced. In summary, for both networks, when the discount factor $\delta < 1$, earlier intervention starting times t_0 more effectively reduce early infections although it might be at the cost of larger later infections.

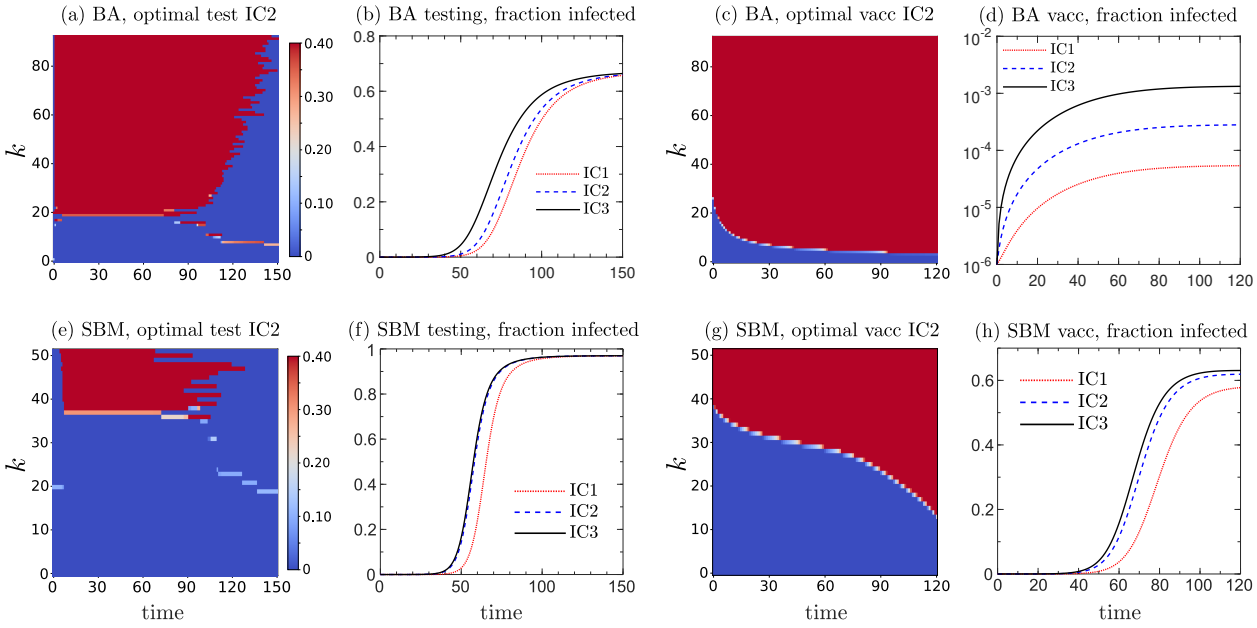

Figure 10.5: Dependence of intervention effectiveness on the degree of the initial infected individual. (a) The PMP-optimal testing strategy computed using IC2 ($k_{\mathrm{i}} = 20$) on the BA network. Strategies for IC1 ($k_{\mathrm{i}} = 3$) and IC3 ($k_{\mathrm{i}} = 90$) are qualitatively similar (not shown) with small differences at the beginning leading to the different delays in the infection dynamics shown in (b). Specifically, for IC1 and IC3, the initial transient of the optimal testing strategy maximizes the testing rate for the subpopulation with the same degree as k_1 and k_3, respectively, indicating that the optimal testing strategy is sensitive to the degree properties of the initial seed infection. Once the disease spreads out, the testing strategies "forget" the initial condition and converge to each other. Despite optimal testing, initial infecteds with larger degrees, such as IC3, lead to the earlier spread of the epidemic. Results are found by using a discount factor $\delta = 0.95$, the optimal strategy given in Alg. 7, and solving Eqs. (10.2.2)–(10.2.5). (c-d) The optimal vaccination strategy for IC2 and the associated fraction infected for the BA network. As with testing, the vaccination strategies associated with IC1 ($k_{\mathrm{i}} = 5$) and IC3 ($k_{\mathrm{i}} = 30$) lead to differences in infection magnitudes. However, the optimal vaccination strategies are insensitive to different initial conditions, even at early times. Since the mechanism of vaccination is always to protect high-degree susceptibles, the vaccination strategies are not as dependent on the current infected population as the testing strategies are. Panel (e) shows the optimal testing strategy for the SBM network, assuming IC2 ($k_{\mathrm{i}} = 20$). (f) The fraction infected exhibits slower dynamics for smaller-degree initial conditions. (g) Optimal vaccination strategy for IC2 in the SBM network, and (h), the associated infected fraction showing both delay and amplitude changes with changes in the initial condition.

10.5.2 Dependence on initial conditions

Besides the start time of testing or vaccination, initial conditions may also affect the optimal strategy. For example, the initial propagation of the disease may depend on the degree k of the first infected individual [OCK21]. Instead of an initial infectious source that is uniformly

distributed across all nodes, as described in Eqs. (10.3.10) and (10.4.11), we vary the degree of the first infected node and explore how the strategies change as a function of concentrated initial condition $i_k(0) = N_0\delta_{k,k_\mathrm{i}}/N$. We take $N_0 = 10^{-6}N$ for both networks, $k_\mathrm{i} = 3, 20, 90$ for the BA network, and $k_\mathrm{i} = 5, 20, 30$ for the SBM network. These different initial conditions are denoted IC1, IC2, and IC3 for each network, respectively.

Optimal testing strategies are found to be subtly dependent on the initial conditions, *i.e.*, the degree of the initial infected patient. In Fig. 10.5 we show only the optimal strategy associated with IC2, but plot the time-dependent fraction infected under optimal strategies for all ICs. Under both optimal testing and vaccination, a smaller degree of the first infected source typically leads to a smaller subsequent infected population. Specifically for testing, this decrease is greatest at intermediate times because at early stages there are fewer infecteds. At later times, the testing strategy becomes insensitive to the initial condition because persons with all degrees are infected and those with a higher degree tend to be infected sooner.

The optimal vaccination strategies obtained through Alg. 7 are also relatively insensitive to initial conditions in both networks, particularly at longer times. Although not shown, the optimal strategies associated with different ICs are mostly the same because nodes with larger degrees tend to always be vaccinated first to minimize the loss function Eq. (10.4.6). Susceptibles with higher degrees are more vulnerable and should be vaccinated first to mitigate subsequent infection events. For vaccination, the different ICs lead to long-term differences in infected fractions because low-degree ICs allows more time for vaccination to more effectively remove susceptibles.

10.5.3 Monte-Carlo simulation of stochastic network model

Our results are derived from a mass-action ODE model and it is unclear how they apply to stochastic network dynamics. To compare these different representations of the disease, we define the discrete stochastic versions of our network models and impose a stochastic

version of the optimal strategies found using the PMP on our ODEs. In Appendix A.5.4 we implemented the optimal testing and vaccination strategies on the BA and SBM network realizations used in the PMP study. Infection, recovery, testing, and vaccination processes are described as Markov events in continuous time. Results from rejection-free, event-based Monte-Carlo simulation indicate that the degree-based mean-field ODE model tends to overestimate new infections because it assumes that all subpopulations interact in a well-mixed manner, thus neglecting certain structural features of the considered networks. The loss function derived from the stochastic model and using the PMP-derived optimal strategy is shown to be lower than that of the ODE model, except for vaccination on the SBM network for which they are comparable. Nonetheless, optimal strategies derived from the ODE model still outperform a uniform or unstructured testing or vaccination strategy applied to the stochastic model. Therefore, although not optimal, the PMP-optimal strategies nonetheless provide advantages in the real-life agent-based stochastic process.

10.6 Summary and conclusions

Our overall results indicate that different network structures (*e.g.*, BA *vs.* SBM) have different susceptibilities to optimal intervention strategies. Thus, policies such as selective social distancing can potentially be used to shift network structure towards one that is more sensitive to direct testing and vaccination strategies.

We have analyzed testing and vaccination separately, but in practice, both are simultaneously implemented. The relative efforts of these two interventions, as a function of time, will depend on their constraints and costs as well as the desired loss function time T. A further generalization of either our mass-action or stochastic versions of our network models may be to derive different loss functions other than Eqs. (10.3.3) and (10.4.6) to take into account factors such as economic effects or prioritization of certain groups (*e.g.*, healthcare workers or individuals with comorbidities). Formulating more specific loss functions would allow one to balance mitigation and suppression strategies, as studied in a well-mixed SIR

model [NLV21]. Another important and straightforward extension of our model is to consider the effects of waning protection of vaccination, which has become a relevant feature of disease control in the context of booster shots. Recovered individuals that include previously vaccinated or infected individuals can become susceptible again at a rate equal to the rate of loss of immunity. Thus, another timescale (months) is introduced which is comparable to timescales T that are used to define the loss function. We expect even wider variety and richness in the analysis of optimization problems under waning immunity.

Finally, the discrepancies between the effective degree ODE model and the Monte-Carlo simulations, under the same ODE-derived optimal strategies appear to arise from the differences in the underlying disease propagation. The discrete stochastic models tend to show lower infected fractions than the corresponding mass-action ODE models since its discreteness and finite infection lifetimes prevents high-degree nodes in some network regions to be infected while the mass-action model allows all nodes to be partially infected. Further analysis of fluctuations in real-world stochastic models could provide insight into a better estimation of optimal strategies without simulating the large space of intervention strategies. This and many other important extensions will be topics of future exploration.

APPENDIX A

Appendix

A.1 Appendix for Chapter 2

A.1.1 Existence and uniqueness of a weak solution for the adder-sizer model

Here, we show the existence and uniqueness of the solution to the sizer-adder model PDE. The full problem is defined as

$$\begin{cases} \dfrac{\partial n}{\partial t} + \dfrac{\partial (ng)}{\partial x} + \dfrac{\partial (ng)}{\partial y} = -\beta(x,y,t)n(x,y,t), \\ g(x,0,t)n(x,0,t) = 2\displaystyle\int_x^\infty \mathrm{d}x' \int_0^{x'} \mathrm{d}y\, \tilde{\beta}(x',y,x,t)n(x',y,t), \\ \beta(x,y,t) \coloneqq \displaystyle\int_0^x \tilde{\beta}(x,y,z,t)\mathrm{d}z, \\ \tilde{\beta}(x,y,z',t) = \tilde{\beta}(x,y,z-z',t),\ \tilde{\beta}(x,y,0,t) = 0,\ n(x,x,t) = 0, \\ n(x,y,t=0) \coloneqq n_0(x,y). \end{cases} \tag{A.1.1}$$

where the independent variables $(x,y,t) \in \mathbf{R}^2 \cap \{y < x\} \times \mathbf{R}^+$.

First, we assume that

$$\begin{aligned} &0 < g_{\min} \leq g \in \mathbf{C}^1(\mathbf{R}^+ \times \mathbf{R}^2 \cap \{y \leq x\}), \\ &n_0(x,y) \in \mathbf{L}^1 \cap \mathbf{L}^\infty(\mathbf{R}^+ \cap \{y < x\}), \\ &0 \leq \tilde{\beta} \in \mathbf{L}^\infty \cap \mathbf{L}^1 \cap \mathbf{C}^1(\mathbf{R}^+ \times (\mathbf{R}^+)^3 \cap \{y < x, z < x\}) \\ &\beta(x,y,t) \in \mathbf{L}^\infty \cap \mathbf{L}^1(\mathbf{R}^+ \times (\mathbf{R}^+)^2) \end{aligned} \tag{A.1.2}$$

and nondimensionalize the size and added size by Δ, the fixed added size parameter that

represents the adder mechanism. We also impose an additional assumption on g:

$$|g(x,y,t)| < K(t+x+1), \quad K < \infty. \tag{A.1.3}$$

We also assume the initial distribution $n_0(x,y)$ has compact support bounded in $(0,\Omega) \times [0,\Omega), \Omega < \infty$. From this assumption and A.1.3, the closure of $n(x,y,T)$'s support is compact for any finite time T since $n \neq 0$ only when $y < x$ and

$$\frac{\mathrm{d}x}{\mathrm{d}t} \leq K(x+t+1) \leq K(x+T+1).$$

From Grönwall's Inequality $x(s) \leq Ce^{Ms} - (1+T)$, where $C < 1+T+\Omega$ is given by the initial condition. At any finite time T, the support of $n(x,y,T)$ is bounded and we assume it is contained in $[0,\Omega(T)) \times [0,\Omega(T))$. Furthermore, by setting $g, \beta, \tilde{\beta} = 0$ at the given time T when (x,y) is out of the support of n, we can assume the closure of $g, \beta, \tilde{\beta}$'s support to be compact. One can generalize the definition of the weak solution n to $[0,\infty) \times (\mathbf{R}^+)^2$ as in [Per08].

Definition A.1 Given time $T < \infty$ and assuming A.1.2, for a function $n \in \mathbf{L}^1((([0,\Omega(T)])^2 \bigcap \{y < x\}) \times [0,T]), \Omega(T) < \infty$ with $n(x,y,t) \neq 0$ in $[0,\Omega(T)) \times [0,\Omega(T)), y < x, t \in [0,T]$, we say that n satisfies the adder-sizer PDE in the weak sense in time $[0,T]$, if

$$\begin{aligned} -\int_0^T \mathrm{d}t \int_0^\infty \mathrm{d}x \int_0^x \mathrm{d}y\, n(x,y,t) \left[\frac{\partial \Psi}{\partial t} + g(x,y,t)\frac{\partial \Psi}{\partial x} + g(x,y,t)\frac{\partial \Psi}{\partial y} - \beta(x,y,t)\Psi(x,y,t) \right] \\ = \int_0^\infty \mathrm{d}x \int_0^x \mathrm{d}y\, n_0(x,y)\Psi_0(x,y) + \int_0^T \mathrm{d}t \int_0^\infty \mathrm{d}x\, \Psi(x,0,t) n(x,0,t) g(x,0,t), \end{aligned} \tag{A.1.4}$$

holds for all test function $\Psi \in \mathbf{C}^1(([0,\Omega(T)])^2 \bigcap \{y \leq x\}) \times [0,T])$ satisfying $\Psi(x,y,T) \equiv 0, \Psi(\Omega(T),y,t) = 0$ and $\Psi(x,x,t) = 0$, where we set $g, \tilde{\beta}, \beta = 0$ for $x \geq \Omega(T), x \leq y$ or $x \leq z$. Upon using the boundary condition in A.1.1, the right-hand-side becomes

$$\int_0^\infty \mathrm{d}x \int_0^x \mathrm{d}y\, n_0(x,y)\Psi_0(x,y) + 2\int_0^T \mathrm{d}t \int_0^\infty \mathrm{d}x \int_0^x \mathrm{d}y \int_0^x \mathrm{d}z\, \Psi(z,0,t)\tilde{\beta}(x,y,z,t)n(x,y,t).$$

Note that if $n \in \mathbf{C}^1(\mathbf{R}^+ \times ((\mathbf{R}^+)^2 \cap \{y < x\}))$ is a classical solution to the PDE (Eq. A.1.1), then it must also satisfy Eq. A.1.4 in any time interval $[0, T]$. We refer to [Per08] for proof of the existence and uniqueness of a weak solution of a related, simpler renewal equation. However, our adder-sizer PDE is more complicated, requiring additional steps to prove the existence and uniqueness of a weak solution.

A.1.2 Uniqueness

First, we prove the uniqueness of the solution to A.1.4. Assume there are two weak solutions $n^{(0)}$ and $n^{(1)}$ for the adder-sizer PDE satisfying A.1.4 with the same initial condition $n_0^{(0)}(x,y) = n_0^{(1)}(x,y)$. Taking the difference between using these purported solutions, we obtain

$$\begin{aligned} -\int_0^T \mathrm{d}t \int_0^\infty \mathrm{d}x \int_0^x \mathrm{d}y\, \Delta n(x,y,t) \left[\frac{\partial \Psi}{\partial t} + g(x,y,t)\frac{\partial \Psi}{\partial x} + g(x,y,t)\frac{\partial \Psi}{\partial y} - \beta(x,y,t)\Psi(x,y,t)\right] \\ = 2\int_0^T \mathrm{d}t \int_0^\infty \mathrm{d}x \int_0^x \mathrm{d}y \int_0^x \mathrm{d}z\, \Psi(z,0,t)\tilde{\beta}(x,y,z,t)\Delta n(x,y,t), \end{aligned} \tag{A.1.5}$$

where $\Delta n = n^{(1)} - n^{(0)}$.

A.1.2.1 Adjoint Problem

First, we consider the adjoint problem for Ψ in the given time interval $[0, T]$ and with a with a source term $S(x,y,t)$ for $0 \le y < x$:

$$\frac{\partial \Psi}{\partial t} + g(x,y,t)\frac{\partial \Psi}{\partial x} + g(x,y,t)\frac{\partial \Psi}{\partial x} - \beta(x,y,t)\Psi(x,y,t) = -2\int_0^x \Psi(z,0,t)\tilde{\beta}(x,y,z,t)\mathrm{d}z - S(x,y,t),$$
$$\Psi(x,y,T) = 0,\ \Psi(\Omega(T),y,t) = 0,\ \Psi(x,x,t) = 0. \tag{A.1.6}$$

Theorem A.1 Assume A.1.2, and $S \in \mathbf{C}^1([0,T]\times[0,\Omega(T)]^2), S(\Omega(T),y,t) = 0$, and $S = 0$ when $x \le y$. Then, there exists a unique $\mathbf{C}^1$ solution to the adjoint problem.

Proof: We can transform the above equation into an ODE along the characteristic line, and then use contraction mapping, which is a standard practice in functional analysis to prove the existence and uniqueness of the solution to an ODE problem. On the left-hand side of Eq. A.1.6, we apply the characteristic line method. Setting $X(c,t) = (x(c,t), y(c,t))$ on the characteristic lines leads to

$$\begin{cases} \dfrac{\partial X(c,s)}{\partial s} = (g(x,y,s), g(x,y,s)), \quad t \le s \le T, \\ X(c,t) = (x_t, y_t), \quad 0 \le y_t < x_t, x_t - y_t = c. \end{cases}$$

Since we have $x(s) - y(s) = x_t - y_t$, the above equation can be simplified as

$$\frac{\partial X(c,s)}{\partial s} = \tilde{g}(X(c,s),s), x(c,t) = x_t, y(c,t) = x_t - c$$

where $\tilde{g}(X(c,s),s) = (g(x(c,s), x(c,s)-c, s), g(x(c,s), x(c,s)-c, s))$. Once c is fixed and x_t is given, the above equation becomes an ordinary differential equation. Given x_t, we define

$$\begin{cases} \tilde{\Psi}(c,s) \coloneqq \Psi(X(c,s),s)e^{-\int_t^s \beta(X(c,v),v)dv}, \\ U(c,z,s) \coloneqq 2\tilde{\beta}(X(c,s),z,s)e^{-\int_t^s \beta(X(c,v),v)dv},\ \tilde{S}(c,s) \coloneqq S(X(c,s),s)e^{-\int_t^s \beta(X(c,v),v)dv}. \end{cases}$$

Thus, along the characteristic line, we can write A.1.6 as

$$\frac{\partial}{\partial s}\tilde{\Psi}(c,s) = -\int_0^{x(c,s)} \Psi(z,0,s)U(c,z,s)\mathrm{d}z - \tilde{S}(c,s). \tag{A.1.7}$$

Since $\tilde{\Psi}(c,T)=0$ and $\tilde{\Psi}(c,t)=\Psi(x_t,x_t-c,t)$,

$$\Psi(x_t,x_t-c,t)=\int_t^T \tilde{S}(c,s)\mathrm{d}s+\int_t^T \mathrm{d}s\int_0^{x(c,s)} \mathrm{d}z\,\Psi(z,0,s)U(c,z,s),\quad 0<c\le x_t. \quad \text{(A.1.8)}$$

We can see that if $x\le y$ or $x_t\ge\Omega(T)$, $\Psi(t,x_t,x_t-c)=\Psi(t,x,x)=0$ since $U,\tilde{S}=0$ for $c\le 0$ or $x_t>\Omega(T)$. Using $c=x_t$, Eq. A.1.8 becomes

$$\Psi(x_t,0,t)=\int_t^T \tilde{S}(x_t,s)\mathrm{d}s+\int_t^T \mathrm{d}s\int_0^{x(x_t,s)} \mathrm{d}z\,\Psi(z,0,s)U(x_t,z,s). \quad \text{(A.1.9)}$$

From condition A.1.3 we obtain $x(s)\le(x_t+1+T)e^{K(s-t)}-(1+T)$. From condition A.3, we define $\tilde{B}=2\|\tilde{\beta}\|_\infty<\infty$. Next, we choose $s=\max\{T-\frac{1}{K}\ln(1+\frac{1}{2\tilde{B}(1+T)}),T-\frac{1}{K}\ln 2,T-1\}$ such that $e^{K(T-t)}\le 1+\frac{1}{2\tilde{B}(1+T)}, s\le t\le T$, and choose x_s small enough such that $x_s<\min\{1,\frac{1}{8\tilde{B}(T-s)}\}$. We denote a mapping T defined on the functional space as

$$T(\Psi)(x_t,0,t)=\int_t^T \tilde{S}(x_t,s)\mathrm{d}s+\int_t^T \mathrm{d}s\int_0^{x(s,x_t)} \mathrm{d}z\,\Psi(z,0,s)U(x_t,z,s),\quad t\in[s,T],x_t\in[0,x_s].$$

It is easy to verify that T is a contraction mapping for $\Psi(x_t,0,t)$ and thus there exists a unique solution Ψ_0 satisfying A.1.6 in D_0 defined as $D_0=\{(x,t)|s\le t\le T,0\le x\le x(x_s,t)\}$, then we let $x_s^1>x_s$ and define $D_1=\{(x,t)|s\le t\le T,0\le x\le x(x_s^1,t)\}$ such that the difference of the area of the region D_1 and D_0 is less than $\tilde{B}^{-1}$. So we can define a second mapping T_1 as

$$\begin{cases} T_1(\Psi)(x_t,0,t)=\displaystyle\int_t^T \mathrm{d}s\int_{x(x_s,s)}^{x(x_t,s)} \mathrm{d}z\,\Psi(z,0,s)U(x_t,z,s)+I(x_s,t),\ \ t\in[s,T],x_t\in[x(t,x_s),x_s^1],\\ I(x_s,t)=\displaystyle\int_t^T \mathrm{d}s\tilde{S}(x_t,s)+\int_t^T \mathrm{d}s\int_0^{x(x_s,s)} \mathrm{d}z\,\Psi_0(z,0,s)U(x_t,z,s). \end{cases}$$

T_1 is also a contraction mapping and we can obtain a Ψ_1 on D_1 such that $T(\Psi_1)=\Psi_1$.

Denote

$$\begin{cases} \Psi(x,0,t) = \Psi_0(x,0,t),\ (x,t) \in D_0, \\ \Psi(x,0,t) = \Psi_1(x,0,t),\ (x,t) \in D_1, \end{cases} \tag{A.1.10}$$

and it is easy to verify that Ψ is C^1 continuous on $D_0 \bigcap D_1$ by first proving it is continuous and then taking the partial derivatives, and Ψ satisfy A.1.6 in the region $D_0 \bigcup D_1$.

Following the same procedure, we can extend Ψ to satisfy A.1.6 in the region $t \in [s,T]$. Then, for $[0,s]$, we choose a $\tilde{s}$ close enough to s and use the same strategy by defining T_2 as

$$\begin{cases} T_2(\Psi)(x_t,0,t) = \int_t^s \mathrm{d}r\, \tilde{S}(x_t,r) + \int_t^s \mathrm{d}r \int_0^{x(x_t,r)} \mathrm{d}z\, \Psi(z,0,r) U(x_t,z,r) + \tilde{I}(t,x_s), t \in [\tilde{s},s], \\ \tilde{I}(x_s,t) = \int_s^T \mathrm{d}r\, \tilde{S}(x_t,r) + \int_s^T \mathrm{d}r \int_0^{x(x_t,r)} \mathrm{d}z\, \Psi(z,0,r) U(x_t,z,r). \end{cases} \tag{A.1.11}$$

We finally obtain a unique function Ψ satisfying A.1.6 in $[0,T] \times [0,\infty)$.

From A.1.8, the value of Ψ is determined by $\tilde{S}, \Psi(x,0,t), U$ and we conclude that there exists a unique $\mathbf{C}^1$ solution for A.1.6.

A.1.2.2 Uniqueness of weak solution for the adder-sizer model

From Section A.1.1 we obtain the existence and uniqueness of Ψ of the adjoint problem. Given any time T and $S(x,y,t) \in \mathbf{C}^1(\mathbf{R}^+ \times (\mathbf{R}^+)^2)$ satisfying the condition in Theorem A.1, since we can set $g, \beta, \tilde{\beta}$'s support to be compact in $[0,T]$, we can find a unique $\mathbf{C}^1$ continuous Ψ satisfying A.1.6. By substituting A.1.6 into A.1.5, we obtain

$$\int_0^T \mathrm{d}t \int_0^{\Omega(T)} \mathrm{d}x \int_0^x \mathrm{d}y\, \Delta n(x,y,t) S(x,y,t) = 0 \tag{A.1.12}$$

for any $S(x,y,t) \in \mathbf{C}^1(\mathbf{R}^+ \times \mathbf{R}^{+2})$ satisfying $S(x,y,t) = 0, x \leq y, S(x \geq \Omega(T), y, t) = 0$, which implies $n \equiv 0$ a.e. in $y < x \leq \Omega(T)$. So at any given time T the weak solution, if

exists, is unique.

One can also set the condition for $\tilde{\beta}, g$ weaker even when we define the weak solution in the unbounded region $[0,\infty)\times(\mathbf{R}^+)^2\cap\{y<x\}$. In [Per08] such work is done for the renewal equation. We do not discuss this generalization in detail here.

A.1.3 Existence of the weak solution

We construct a series of functions $\{n_i\}$ with a limit n for this series satisfying A.1.6 for all test functions Ψ. We use semi-discrete approximation to discretize the PDE and obtain piecewise solutions. As the mesh size becomes smaller, we expect the piecewise solution to converge to a function n satisfying A.1.4.

A.1.3.1 Semi-discrete approximation for the PDE

We choose a uniform grid with mesh size $h>0$ fixed in both x and y axis and let time t be continuous. We denote

$$\begin{aligned}
(x_i,y_j)=(ih,jh),(x_{i+\frac{1}{2}},\quad y_{j+\frac{1}{2}})&=((i+\frac{1}{2})h,(j+\frac{1}{2})h),\quad j<i\in\mathbf{N},\\
\beta_{i+\frac{1}{2},j+\frac{1}{2}}(t)&=\frac{1}{h^2}\int_{ih}^{(i+1)h}\mathrm{d}y\int_{jh}^{(j+1)h}\mathrm{d}x\,\beta(x,y,t),\quad j<i\in\mathbf{N},\\
\tilde{\beta}_{i+\frac{1}{2},j+\frac{1}{2}}((s+\frac{1}{2})h,t)&=\frac{1}{h^3}\int_{ih}^{(i+1)h}\mathrm{d}z\int_{jh}^{(j+1)h}\mathrm{d}y\int_{sh}^{(s+1)h}\mathrm{d}x\,\tilde{\beta}(x,y,z,t),\quad s\le i,\\
g_{i,j}(t)&=g(ih,jh,t),\quad j<i\in\frac{1}{2}\mathbf{N}.
\end{aligned}\tag{A.1.13}$$

Here, $\beta_{i+\frac{1}{2},j+\frac{1}{2}}(t)=h\sum_{s=0}^{i}\tilde{\beta}_{i+\frac{1}{2},j+\frac{1}{2}}((s+\frac{1}{2})h,t)$. Given a fixed time T, we wish to find a solution of point-wise function $n^h(t)$, which takes values on the grid points $(x_{i+\frac{1}{2}},y_{j+\frac{1}{2}})$ as we denote above. Then n^h can be seen as a vector function. According to our assumption there exists Ω such that the initial value n^0 is nonzero within the region $\{(x,y)|y<x,x<\Omega\}$, and from our previous illustration there exists $\Omega(T)<\infty$ such that n is nonzero within the

region $\{(x,y)|y<x, x<\Omega(T)\}$. So we can take $h_k=\frac{\Omega(T)}{k}$ so as k tends to infinity the width of the mesh grid will tend to zero.

By discretizing A.1.1, we expect the vector function $n^h(t)$ to satisfy the below equations for $t\in[0,T]$ and $0<j<i<L$ (L is the number of discretization points along one direction):

$$
\begin{aligned}
&h^2\frac{\mathrm{d}n_{i+\frac{1}{2},j+\frac{1}{2}}(t)}{\mathrm{d}t}+h(g_{i+1,j+\frac{1}{2}}(t)n_{i+1,j+\frac{1}{2}}(t)-g_{i,j+\frac{1}{2}}(t)n_{i,j+\frac{1}{2}}(t))\\
&\quad+h(g_{i+\frac{1}{2},j+1}(t)n_{i+\frac{1}{2},j+1}(t)-g_{i+\frac{1}{2},j}(t)n_{i+\frac{1}{2},j}(t))+h^2\beta_{i+\frac{1}{2},j+\frac{1}{2}}(t)n_{i+\frac{1}{2},j+\frac{1}{2}}(t)=0,\\
&\qquad 0\le j<i-1\\
&h^2\frac{\mathrm{d}n_{i+\frac{1}{2},j+\frac{1}{2}}(t)}{\mathrm{d}t}+hg_{i+1,j+\frac{1}{2}}(t)n_{i+1,j+\frac{1}{2}}(t)-hg_{i+\frac{1}{2},j}(t)n_{i+\frac{1}{2},j}(t)\\
&\quad+h^2\beta_{i+\frac{1}{2},j+\frac{1}{2}}(t)n_{i+\frac{1}{2},j+\frac{1}{2}}(t)=0,\quad 0\le j=i-1
\end{aligned}
\tag{A.1.14}
$$

$$
g_{i+\frac{1}{2},0}(t)n_{i+\frac{1}{2},-\frac{1}{2}}(t)=2h^2\sum_{l=i}^{K-1}\sum_{j=0}^{l-1}\tilde{\beta}_{l+\frac{1}{2},j+\frac{1}{2}}((i+\frac{1}{2})h,t)n_{l+\frac{1}{2},j+\frac{1}{2}}(t),
$$

$$
n_{i+\frac{1}{2},j+\frac{1}{2}}(0)=\frac{1}{h^2}\int_{x_i}^{x_{i+1}}\mathrm{d}y\int_{y_j}^{y_{j+1}}\mathrm{d}x\qquad n_0(x,y),\ n_{i+\frac{1}{2},i+\frac{1}{2}}(t)=0,
\tag{A.1.15}
$$

where we henceforth omit the h superscript in the proof. In the two-dimensional upwind scheme, derivatives in one direction are neglected on neighboring sites in the other direction: $n_{i,j\pm\frac{1}{2}}=n_{i-\frac{1}{2},j\pm\frac{1}{2}}, n_{i\pm\frac{1}{2},j}=n_{i\pm\frac{1}{2},j-\frac{1}{2}}$. The boundary condition $n(x,x,t)=0$ is implemented by $n_{i+\frac{1}{2},i+\frac{1}{2}}(t)=0$ for any t and i.

We will obtain a uniform bound irrelevant of h for n. All coefficients in the above ODE equations are $\mathbf{C}^1$ continuous, which means that there exists a unique solution in time $[0,T], T<\infty$.

Theorem A.2 For $t \in [0, T]$ and assuming A.1.2 hold, we find the bound

$$\sum_{i=1}^{L-1}\sum_{j=0}^{i} |n_{i+\frac{1}{2},j+\frac{1}{2}}(t)| \leq e^{Mt} \sum_{i=1}^{L-1}\sum_{j=0}^{i} |n_{i+\frac{1}{2},j+\frac{1}{2}}(0)|, \tag{A.1.16}$$

where $\tilde{B} = 2\|\tilde{\beta}\|_\infty, M = 2B - b, B = \|\beta\|_\infty$, and $b = \min_t \min_{i,j} \beta_{i+\frac{1}{2},j+\frac{1}{2}}(t)$.

And the $\mathbf{L}^\infty$ bound is given as

$$\|n^h(t)\|_\infty \leq e^{(2\tilde{g}')t} R \tag{A.1.17}$$

where $R = \max\{\frac{1}{g_{min}} \tilde{B} e^{MT} \|n(0)\|_1, \|n^h(0)\|_\infty\}$, $\tilde{g}'$ is the $\mathbf{L}^\infty$ bound of g's spatial partial derivatives.

Proof For the summation of n over all grid points, we multiply the first equation in Eq.(A.1.15) by $\text{sign}(n_{i+\frac{1}{2},j+\frac{1}{2}})$ for each $i, j \leq i$ we have,

$$h^2 \frac{\mathrm{d}}{\mathrm{d}t} |n_{i+\frac{1}{2},j+\frac{1}{2}}(t)| + h g_{i+1,j+\frac{1}{2}} |n_{i+\frac{1}{2},j+\frac{1}{2}}| + h g_{i+\frac{1}{2},j+1} |n_{i+\frac{1}{2},j+\frac{1}{2}}| + h^2 \beta_{i+\frac{1}{2},j+\frac{1}{2}} |n_{i+\frac{1}{2},j+\frac{1}{2}}| \leq$$
$$h g_{i,j+\frac{1}{2}} |n_{i-\frac{1}{2},j+\frac{1}{2}}| + h g_{i+\frac{1}{2},j} |n_{i+\frac{1}{2},j-\frac{1}{2}}| \tag{A.1.18}$$

By multiplying the second equation in Eq. (A.1.15) by $\text{sign}(n_{i+\frac{1}{2},j+\frac{1}{2}})$ for each $i, j \leq i$ pair and summing over index $\sum_{i=1}^{L-1} \sum_{j=0}^{i-1}$,

$$h^2 \sum_{i=1}^{L-1}\sum_{j=0}^{i-1} |n_{i+\frac{1}{2},j+\frac{1}{2}}(t)| + h \sum_{i=1}^{L-1} g_{i+\frac{1}{2},i-\frac{1}{2}}(t) |n_{i+\frac{1}{2},i-\frac{1}{2}}(t)| + h \sum_{j=0}^{i-1} g_{L,j+\frac{1}{2}}(t) |n_{L-1+\frac{1}{2},j+\frac{1}{2}}(t)| +$$
$$h^2 \sum_{i=1}^{L-1}\sum_{j=0}^{i-1} \beta_{i+\frac{1}{2},j+\frac{1}{2}} |n_{i+\frac{1}{2},j+\frac{1}{2}}| \leq h \sum_{i=0}^{L-1} g_{i+\frac{1}{2},0}(t) |n_{i+\frac{1}{2},-\frac{1}{2}}(t)|.$$

We can simplify the above expression to

$$
\begin{aligned}
h^2 \frac{d}{dt} \sum_{i=1}^{L-1} \sum_{j=0}^{i-1} |n_{i+\frac{1}{2},j+\frac{1}{2}}(t)| &+ h^2 \sum_{i=1}^{L-1} \sum_{j=0}^{i-1} \beta_{i+\frac{1}{2},j+\frac{1}{2}} |n_{i+\frac{1}{2},j+\frac{1}{2}}(t)| \\
&\leq 2h^3 \sum_{i=0}^{L-1} |\sum_{l=i}^{L-1} \sum_{j=0}^{l-1} \tilde{\beta}_{l+\frac{1}{2},j+\frac{1}{2}}((i+1/2)h,t) n_{l+\frac{1}{2},j+\frac{1}{2}}(t)| \\
&\leq 2h^2 \sum_{l=1}^{L-1} \sum_{j=0}^{l-1} |\beta_{l+\frac{1}{2},j+\frac{1}{2}}(t)||n_{l+\frac{1}{2},j+\frac{1}{2}}(t)|.
\end{aligned}
$$

We then have

$$
\frac{\mathrm{d}}{\mathrm{d}t} \sum_{i=1}^{L-1} \sum_{j=0}^{i-1} |n_{i+\frac{1}{2},j+\frac{1}{2}}(t)| \leq (2B-b) \sum_{i=1}^{L-1} \sum_{j=0}^{i-1} |n_{i+\frac{1}{2},j+\frac{1}{2}}(t)|,
$$

which yields

$$
\sum_{i=1}^{L-1} \sum_{j=0}^{i-1} |n_{i+\frac{1}{2},j+\frac{1}{2}}(t)| \leq e^{Mt} \sum_{i=1}^{L-1} \sum_{j=0}^{i-1} |n_{i+\frac{1}{2},j+\frac{1}{2}}(t)(0)|. \tag{A.1.19}
$$

A.1.19 states that the l^1 norm of all the values on the grid points is uniformly bounded and the upper bound is not relevant to h. Next, estimate the $\mathbf{L}^\infty$ bound of n^h. First, we consider $j=0$ and assume $S(t) = \max\limits_{1\leq i\leq L-1} |n_{i+\frac{1}{2},\frac{1}{2}}(t)|e^{-\tilde{g}'t}$ for $t \in [0,T]$. For the maximum value of S at some index i, we find

$$
\begin{aligned}
h^2 \frac{\mathrm{d}|n_{i+\frac{1}{2},\frac{1}{2}}(t)|}{\mathrm{d}t} + h(g_{i+1,\frac{1}{2}}(t)|n_{i+\frac{1}{2},\frac{1}{2}}(t)| - g_{i,\frac{1}{2}}(t)|n_{i-\frac{1}{2},\frac{1}{2}}(t)|) + \\
h(g_{i+\frac{1}{2},1}(t)|n_{i+\frac{1}{2},\frac{1}{2}}(t)| - g_{i+\frac{1}{2},0}(t)|n_{i+\frac{1}{2},-\frac{1}{2}}(t)|) \leq 0,
\end{aligned}
$$

$$
h^2 \frac{\mathrm{d}|n_{i+\frac{1}{2},\frac{1}{2}}(t)|}{\mathrm{d}t} + h g_{i+1,\frac{1}{2}}(t)|n_{i+\frac{1}{2},\frac{1}{2}}(t)| - g_{i+\frac{1}{2},0}(t)|n_{i+\frac{1}{2},-\frac{1}{2}}(t)| \leq 0, \quad i=1.
$$

and

$$
\frac{\mathrm{d}(|n_{i+\frac{1}{2},\frac{1}{2}}(t)|e^{-\tilde{g}'t})}{\mathrm{d}t} + h^{-1} g_{i+\frac{1}{2},1}(t)|n_{i+\frac{1}{2},\frac{1}{2}}(t)|e^{-\tilde{g}'t} \leq h^{-1} g_{i+\frac{1}{2},0}(t)|n_{i+\frac{1}{2},-\frac{1}{2}}(t)|e^{-\tilde{g}'t},
$$

By the assumption that $g(x,y,t) \geq g_{min}(t) \geq g_{min} > 0$ and $g < K(T+1+\Omega(T))$, we have

$$\frac{\mathrm{d}(|n_{i+\frac{1}{2},\frac{1}{2}}(t)|e^{-\tilde{g}'t})}{\mathrm{d}t}+h^{-1}g_{\min}(t)(|n_{i+\frac{1}{2},\frac{1}{2}}(t)|e^{-\tilde{g}'t}) \leq h^{-1}\left(\frac{g_{\min}(t)}{g_{\min}}\right)\max_{1\leq i\leq L-1}|g_{i+\frac{1}{2},0}(t)n_{i+\frac{1}{2},-\frac{1}{2}}(t)|. \tag{A.1.20}$$

Finally defining $G(t)=h^{-1}\int_0^t g_{\min}(s)\mathrm{d}s$ yields

$$\frac{\mathrm{d}(|n_{i+\frac{1}{2},\frac{1}{2}}(t)|e^{-\tilde{g}'t}e^{G(t)})}{\mathrm{d}t} \leq \frac{g_{\min}(t)}{h}(\frac{1}{g_{\min}})\max_{1\leq i\leq L-1}|g_{i+\frac{1}{2},0}(t)n_{i+\frac{1}{2},-\frac{1}{2}}(t)|e^{G(t)}.$$

From the $\mathbf{L}^1$ bound, we can deduce

$$\max_t \max_{1\leq i\leq L-1}|g_{i+\frac{1}{2}}(t)n_{i+\frac{1}{2},-\frac{1}{2}}(t)| \leq h^2\tilde{B}e^{MT}\|n^h(0)\|_1 \leq \tilde{B}e^{MT}\|n(0)\|_1, \quad t>0$$

and conclude that for the function $S(t)e^{G(t)}$

$$S(t)e^{G(t)} \leq S(0)+\frac{1}{g_{\min}}\tilde{B}e^{MT}\|n(0)\|_1(e^{G(t)}-1), \tag{A.1.21}$$

and $S(t)\leq \max_{1\leq i\leq L-1}\{n_{i+\frac{1}{2},\frac{1}{2}}(0),\frac{1}{g_{\min}}\tilde{B}e^{MT}\|n(0)\|_1\}$, which then gives the $\mathbf{L}^\infty$ bound for the point-wise solution n^h when $j=0$.

Now, we set $R=\max\{\frac{1}{g_{\min}}\tilde{B}e^{MT}\|n(0)\|_1,\|n^h(0)\|_\infty\}$ and estimate $|n_{i+\frac{1}{2},j+\frac{1}{2}}(t)|$ for $j\geq 0$ by setting $P(t)=\max_{0\leq i\leq L-1,0\leq j\leq i-1}|n_{i+\frac{1}{2},j+\frac{1}{2}}(t)|e^{-2\tilde{g}'t}$ and $\tilde{S}(t)=S(t)e^{-\tilde{g}'t}\leq S(t)$. At a fixed time t, $P(t)$ is taken on a certain $(i+\frac{1}{2},j+\frac{1}{2})$, so either $P(t)=\tilde{S}(t)$ or $P(t)$ is taken somewhere $j>0$.

If $i-1>j>0$, we have

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}(|n_{i+\frac{1}{2},j+\frac{1}{2}}(t)|e^{-2\tilde{g}'t}) \leq (&\frac{g_{i,j+\frac{1}{2}}(t)|n_{i+\frac{1}{2},j+\frac{1}{2}}(t)|-g_{i+1,j+\frac{1}{2}}(t)|n_{i+\frac{1}{2},j+\frac{1}{2}}(t)|}{h}\\ &+\frac{g_{i+\frac{1}{2},j}(t)|n_{i+\frac{1}{2},j+\frac{1}{2}}(t)|-g_{i+\frac{1}{2},j+1}(t)|n_{i+\frac{1}{2},j+\frac{1}{2}}(t)|}{h}-2\tilde{g'}|n_{i+\frac{1}{2},\frac{1}{2}}(t)|)e^{-2\tilde{g}'t}\leq 0;\end{aligned}$$

if $j=i-1>0$, we have

$$\frac{\mathrm{d}}{\mathrm{d}t}(|n_{i+\frac{1}{2},j+\frac{1}{2}}(t)|e^{-2\tilde{g}'t}) \leq \left[h^{-1}\left(g_{i+\frac{1}{2},j}(t) - g_{i+1,j+\frac{1}{2}}(t)\right)|n_{i+\frac{1}{2},\frac{1}{2}}(t)| - 2\tilde{g}'|n_{i+\frac{1}{2},j+\frac{1}{2}}(t)|\right]e^{-2\tilde{g}'t}$$
$$\leq 0.$$

For any $t \in (0,T]$ we can set $\tilde{t} < t$ to be the lower bound that $P(v) > \tilde{S}(v), v \in (\tilde{t},t]$, if $\tilde{t} = 0$ then $P(t) \leq P(0) = \|n^h(0)\|_\infty$ from above equation, or if $\tilde{t} < t$ then $P(t) \leq P(\tilde{t}) \leq S(\tilde{t}) \leq \max\limits_{0\leq t\leq T} S(t)$ (since $P(t)$ is nonincreasing in $[\tilde{t},t]$ and is evaluated at some $j > 0$). If $\tilde{t} = t$ then $P(t) = S(t) \leq \max\limits_{0\leq t\leq T} S(t)$ so $P(t) \leq \max\{\max\limits_{0\leq t\leq T} S(t), \|n^h(0)\|_\infty\} = R$, and

$$\|n^h(t)\|_\infty \leq e^{2\tilde{g}'t}R. \tag{A.1.22}$$

We arrive at the second conclusion in Theorem A.2, which states that the $\mathbf{L}^\infty$ bound again not related to h.

A.1.3.2 Existence of the weak solution

Given the time $T < \infty$, if we take the limit $h \to 0$, we will obtain a vector functions family $\{n^{h(k)}\}$, just take k as integer numbers and let $h(k) = \Omega(T)/k$. Now we can obtain piecewise functions based on the vector functions $n^{h(k)}$. By setting $n^h_{i+\frac{1}{2},i+\frac{1}{2}}(t) = 0$, we define $n^h(x,y,t)$ and related $\beta^h, \tilde{\beta}^h$ as

$$\begin{aligned}
n^h(x,y,t) =& \sum_{i=0}^{L-1}\sum_{j=0}^{i-1} n^h_{i+\frac{1}{2},j+\frac{1}{2}}(t)\chi_{\{ih\leq x<(i+1)h, jh\leq y<(j+1)h\}},\\
\beta^h(x,y,t) =& \sum_{i=0}^{L-1}\sum_{j=0}^{i} \beta_{i+\frac{1}{2},j+\frac{1}{2}}(t)\chi_{\{ih\leq x<(i+1)h, jh\leq y<(j+1)h\}},\\
\tilde{\beta}^h(x,y,z,t) =& \sum_{i=0}^{L-1}\sum_{j=0}^{i-1}\sum_{l=0}^{i-1} \tilde{\beta}_{i+\frac{1}{2},j+\frac{1}{2}}((l+\frac{1}{2})h,t)\chi_{\{ih\leq x<(i+1)h, jh\leq y<(j+1)h, lh\leq z<(l+1)h\}},\\
n^h(x,0,t) =& n^h_{i+\frac{1}{2},-\frac{1}{2}}(t), \quad ih \leq x < (i+1)h,
\end{aligned}$$

where χ is the indicator function. Since there is an upper bound for both β and $\tilde{\beta}$, and both $\beta, \tilde{\beta}$ are continuous, we have the following result

$$\begin{aligned}
\lim_{k\to\infty} \beta^{h(k)}(x,y,t) &\to \beta(x,y,t) \ a.e. \quad 0 \le \beta^{h(k)} \le \|\beta\|_\infty < \infty, \\
\lim_{k\to\infty} \tilde{\beta}^{h(k)}(x,y,z,t) &\to \beta(x,y,z,t) \ a.e. \quad 0 \le \tilde{\beta}^{h(k)} \le \|\tilde{\beta}\|_\infty < \infty, \\
\lim_{k\to\infty} n^{h(k)}(x,y,0) &\to n(x,y,0) \ a.e..
\end{aligned}$$

Then, we can easily extend Theorem A.2 for our piecewise constant functions $n^{h(k)}$.

Corollary A.3 Under the conditions of Theorem A.2, we have for any $t \in [0,T]$ and any h,

$$\int_0^{\Omega(T)} \mathrm{d}y \int_0^{\Omega(T)} \mathrm{d}x \, |n^h(x,y,t)| \le e^{Mt} \int_0^{\Omega(0)} \mathrm{d}y \int_0^{\Omega(0)} \mathrm{d}x \, |n^h(x,y,0)|, \tag{A.1.23}$$

and

$$\|n^h(t)\|_\infty \le \max\{\|n(0)\|_\infty, Be^{MT}\|n(0)\|_1\} e^{2\tilde{g'}t}, \tag{A.1.24}$$

where $B, M, \tilde{g}'$ are defined in Theorem A.2. The proof is the direct consequence of Theorem A.2.

The piecewise constant functions $\{n^{h(k)}\}$ are uniformly bounded and $n^{h(k)} \in \mathbf{L}^1 \bigcap \mathbf{L}^\infty([0,T] \times [0,\Omega(T)]^2)$, so n^h are all $\mathbf{L}^2$ functions. We have the fact that their exists a function $n \in \mathbf{L}^2([0,T] \times [0,\Omega(T)]^2)$ and $b(x,t)$ such that there exists a series $k_i \to \infty$ and

$$n^{h(k)} \to n, w^* - \mathbf{L}^2([0,T] \times [0,\Omega(T)]^2 \cap \{y < x\}) \tag{A.1.25}$$

Since $\mathbf{L}^2[0,T] \times [0,\Omega(T))^2$ implies $\mathbf{L}^1$ bound, we can deduce that n, b are $\mathbf{L}^1$ functions as desired. For the piecewise-constant-in-space function $n^{h(k)}, k \in \mathbf{N}^+$, there exists a function $n \in L^2([0,T] \times [0,\Omega(T)]^2$ sequence $k_i \to \infty$ such that $n^{h(k_i)} \to n, w^* - L^2([0,T] \times [0,\Omega(T)]^2)$.

To prove this, we need only to verify that there exists a sequence $n^{h(k_i)}$ such that for all test functions $f \in \mathbf{L}^2$, $\int_0^T \mathrm{d}t \int_0^{\Omega(T)} \mathrm{d}x \int_0^x \mathrm{d}y \, n^{h(k_i)} f \to \int_0^T \mathrm{d}t \int_0^{\Omega(T)} \mathrm{d}x \int_0^x \mathrm{d}y \, nf$. Since $\mathbf{L}^2$ space

is separable, we have a countable set of basis function $\{b_i(x,y,t)\}$ for the space $\mathbf{L}^2([0,T]\times[0,\Omega(T)]^2\cap\{y<x\})$. Thus, every $n^{h(k)}$ can be decomposed as $n^{h(k)}=\sum_{i=1}^{\infty}\alpha_i^k b_i$. The $n^{h(k)}$s are uniformly $\mathbf{L}^\infty$ bounded, so $\sum\alpha_k^2$ are all uniformly bounded. If the bound is S, we can select a sequence $\{n^{h(k_i)}\}$ from $\{n^{h(k)}\}$ satisfying $\lim_{i\to\infty}\alpha_j^{k_i}=\alpha_j<\infty$ so that $\sum_{i=1}^{\infty}\alpha_j^2\le S<\infty$. If we decompose $n=\sum_{i=1}^{\infty}\alpha_i b_i$, then by decomposing any test function $\Psi\in\mathbf{L}^2([0,T]\times[0,\Omega(T)]^2\cap\{y<x\})$ by $\Psi=\sum_{i=1}^{\infty}\gamma_i b_i$, we have

$$\lim_{i\to\infty}\left|\int_0^T\mathrm{d}t\int_0^{\Omega(T)}\mathrm{d}x\int_0^x\mathrm{d}y\,\left(n^{h(k_i)}-n\right)\Psi\right|=\left|\sum_{s=1}^{\infty}(\alpha_s^{k_i}-\alpha_s)\gamma_s\right|=0, \tag{A.1.26}$$

which gives the result $n^{h(k)}\to n, w^*-\mathbf{L}^2([0,T]\times[0,\Omega(T)]^2\cap\{y<x\})$ as desired.

We can now show that n is a weak solution by using the first equation in Eq.(A.1.15). For any test function $\Psi\in\mathbf{C}^1([0,T]\times[0,\Omega(T)]^2)$, we have $\Psi(x,y,T)=0,\Psi(x,y,t)=0, y\ge x$. We define

$$\Psi_{i+\frac{1}{2},j+\frac{1}{2}}(t)=\frac{1}{h^2}\int_{x_i}^{x_{i+1}}\mathrm{d}x\int_{y_j}^{y_{j+1}}\mathrm{d}y\,\Psi(x,y,t),\quad j\le i.$$

For a given $L\in\mathbf{N}^+$ and $h=\frac{\Omega(T)}{L}$,

$$\begin{aligned}&\int_0^T\mathrm{d}t\sum_{i=1}^{L-1}\sum_{j=0}^{i-1}\Big(h^2\frac{\mathrm{d}n^h_{i+\frac{1}{2},j+\frac{1}{2}}(t)}{\mathrm{d}t}\Psi_{i+\frac{1}{2},j+\frac{1}{2}}(t)+h[g_{i+1,j+\frac{1}{2}}(t)n^h_{i+\frac{1}{2},j+\frac{1}{2}}(t)-g_{i,j+\frac{1}{2}}(t)n^h_{i-\frac{1}{2},j+\frac{1}{2}}]\Psi_{i+\frac{1}{2},j+\frac{1}{2}}(t)\\&+h[g_{i+\frac{1}{2},j+1}(t)n^h_{i+\frac{1}{2},j+\frac{1}{2}}(t)-g_{i+\frac{1}{2},j}(t)n^h_{i+\frac{1}{2},j-\frac{1}{2}}]\Psi_{i+\frac{1}{2},j+\frac{1}{2}}(t)+h^2\beta_{i+\frac{1}{2},j+\frac{1}{2}}(t)n^h_{i+\frac{1}{2},j+\frac{1}{2}}\Psi_{i+\frac{1}{2},j+\frac{1}{2}}(t)\Big)\\&=-\int_0^T\mathrm{d}t\sum_{i=1}^{L-1}hg_{i+\frac{1}{2},i-\frac{1}{2}}(t)n^h_{i+\frac{1}{2},i-\frac{1}{2}}.\end{aligned}$$

Integrating the above equation by parts with respect to time, we find

$$\int_0^T \mathrm{d}t \left[\sum_{i=1}^{L-1} \sum_{j=0}^{i-1} h^2 n^h_{i+\frac{1}{2},j+\frac{1}{2}}(t) \frac{\mathrm{d}\Psi_{i+\frac{1}{2},j+\frac{1}{2}}(t)}{\mathrm{d}t} + h \sum_{i=1}^{L-2} \sum_{j=0}^{i-1} g_{i+1,j+\frac{1}{2}}(t) n^h_{i+\frac{1}{2},j+\frac{1}{2}}(t)(\Psi_{i+\frac{3}{2},j+\frac{1}{2}}(t) - \Psi_{i+\frac{1}{2},j+\frac{1}{2}}(t)) \right.$$
$$\left. + h \sum_{i=1}^{L-1} \sum_{j=0}^{i-2} g_{i+\frac{1}{2},j+1}(t) n^h_{i+\frac{1}{2},j+\frac{1}{2}}(t)(\Psi_{i+\frac{1}{2},j+\frac{3}{2}}(t) - \Psi_{i+\frac{1}{2},j+\frac{1}{2}}(t)) \right] + h \int_0^T \mathrm{d}t \sum_{i=1}^{L-1} g_{i+\frac{1}{2},0}(t) n^h_{i+\frac{1}{2},-\frac{1}{2}}(t) \Psi_{i+\frac{1}{2},\frac{1}{2}}(t)$$
$$- h \int_0^T \mathrm{d}t \left[\sum_{j=0}^{L-2} g_{L,j+\frac{1}{2}}(t) n^h_{L-\frac{1}{2},j+\frac{1}{2}}(t) \Psi_{L-\frac{1}{2},j+\frac{1}{2}}(t) + \sum_{i=1}^{L-1} \sum_{j=0}^{i-1} h^2 \beta_{i+\frac{1}{2},j+\frac{1}{2}}(t) n^h_{i+\frac{1}{2},j+\frac{1}{2}}(t) \Psi_{i+\frac{1}{2},j+\frac{1}{2}}(t) \right] =$$
$$h^2 \sum_{i=0}^{L-1} \sum_{j=0}^{i-1} n^h_{i+\frac{1}{2},j+\frac{1}{2}}(0) \Psi_{i+\frac{1}{2},j+\frac{1}{2}}(0). \tag{A.1.27}$$

Since $\Psi_{i+\frac{3}{2},j+\frac{1}{2}}(t) - \Psi_{i+\frac{1}{2},j+\frac{1}{2}}(t) = \int_{ih}^{(i+1)h} \mathrm{d}x \int_{jh}^{(j+1)h} \mathrm{d}y \int_x^{x+h} \mathrm{d}s \, \frac{\partial \Psi}{\partial s}(s,y,t)$, $|n^h|$ is uniformly bounded while g is $\mathbf{C}^1$ continuous. From above we can pick a series in $\{n^{h(k)}\}$, denoted by $\{n^{h(k_i)}\}$ satisfying A.1.25. We take $n^h = n^{h(k_i)}$ in the above formula, since $\Psi \in \mathbf{C}^1[0,T] \times [0,\Omega(T)]^2$, then given any Ψ we have a positive upper bound $R(\Psi) < \infty$ for both Ψ and its any first-order partial derivatives. Thus,

$$\left| \int_0^T \mathrm{d}t \sum_{i=1}^{L-1} \sum_{j=0}^{i-1} \left(h^2 n^{h(k_i)}_{i+\frac{1}{2},j+\frac{1}{2}}(t) \frac{\mathrm{d}\Psi_{i+\frac{1}{2},j+\frac{1}{2}}(t)}{\mathrm{d}t} \right) + \int_0^T \mathrm{d}t \int_0^{\Omega(T)} \mathrm{d}x \int_0^x \mathrm{d}y \, n^{h(k_i)}(x,y,t) \frac{\partial \Psi(x,y,t)}{\partial t} \right|$$
$$\leq \int_0^T \mathrm{d}t \sum_{i=0}^{L-1} \int_{ih}^{(i+1)h} \mathrm{d}x \int_{ih}^x \mathrm{d}y \left| n^{h(k_i)}(x,y,t) \frac{\partial \Psi(x,y,t)}{\partial t} \right|.$$

As i tends to infinity, $|\int_0^T \mathrm{d}t \sum_{i=0}^{L-1} \int_{ih}^{(i+1)h} \mathrm{d}x \int_{ih}^x \mathrm{d}y \, n^{h(k_i)}(x,y,t) \frac{\partial \Psi(x,y,t)}{\partial t}|$ tends to zero since $\frac{\partial \Psi}{\partial t}$ and $n^{h(k_i)}$ are all bounded, and

$$\int_0^T \mathrm{d}t \int_0^{\Omega(T)} \mathrm{d}x \int_0^x \mathrm{d}y \, h^2 n^{h(k_i)}(x,y,t) \frac{\partial \Psi(x,y,t)}{\partial t} \to \int_0^T \mathrm{d}t \int_0^{\Omega(T)} \mathrm{d}x \int_0^x \mathrm{d}y \, h^2 n(x,y,t) \frac{\partial \Psi(x,y,t)}{\partial t} \tag{A.1.28}$$

as i tends to infinity, so the first term in A.1.27 tends to the limit in A.1.28.

By the same procedure and using the condition that g is uniformly continuous in $[0,T] \times$

$[0, \Omega(t)]^2 (g$ is $\mathbf{C}^1)$, it is easy to verify that the second term in the LHS of A.1.27 tends to $\int_0^T \mathrm{d}t \int_0^{\Omega(T)} \mathrm{d}x \int_0^x \mathrm{d}y \, (gn)(x,y,t) \frac{\partial \Psi}{\partial x}$, and the third term in the LHS of A.1.27 tends to $\int_0^T \mathrm{d}t \int_0^{\Omega(T)} \mathrm{d}x \int_0^x \mathrm{d}y \, (gn)(x,y,t) \frac{\partial \Psi}{\partial y}$.

We turn to the right-hand side of A.1.27, by the same procedure, it is easy to verify the first term tends to $\int_0^T \mathrm{d}t \int_0^{\Omega(T)} \mathrm{d}x \int_0^x \mathrm{d}y \, n(x,y,0) \Psi(x,y,0)$, and the second term tends to $\int_0^T \mathrm{d}t \int_0^{\infty} \mathrm{d}x \int_0^x \mathrm{d}y \int_0^x \mathrm{d}z \, \Psi(z,0,t) \tilde{\beta}(x,y,z,t) n(x,y,t)$. The third term will tend to 0 since Ψ is $\mathbf{C}^1$ continuous and takes 0 on the boundary $x = y$ and $x = \Omega(T)$. Thus, there exists a uniform upper bound on g, n^h and the last term tends to $\int_0^T \mathrm{d}t \int_0^{\Omega(T)} \mathrm{d}x \int_0^x \mathrm{d}y \, \beta(x,y,t) n(x,y,t) \Psi(x,y,t)$.

By passing to the limit $i \to \infty$, we obtain that n exactly satisfies the condition of a weak solution in A.1.4. One can follow the proof in [Per08] and generalize the conclusions to $\mathbf{R}^+ \times (\mathbf{R}^+)^2 \bigcap \{y < x\}$.

A.1.4 Numerical scheme

We denote $\mathbf{u}(t) = \{\mathbf{n}_1(t), \mathbf{n}_2(t), \ldots, \mathbf{n}_{L-1}(t)\}^\intercal$ where $\mathbf{n}_j(t) = \{n_{\frac{1}{2}, j-\frac{1}{2}}, n_{1+\frac{1}{2}, j-\frac{1}{2}}, \ldots, n_{L-\frac{1}{2}, j-\frac{1}{2}}\}$ and $n_{i \le j} = 0$. Equations 2.1.18 and 2.1.19 can then be written in the form $\mathbf{u}(t + \Delta t) = \mathbf{A}(t)\mathbf{u}(t)$, where

$$\mathbf{A}(t) = \begin{bmatrix} \mathbf{B}_1 + \mathbf{C}_1 & \mathbf{C}_2 & \mathbf{C}_3 & \mathbf{C}_4 & \cdots & \mathbf{C}_{L-2} & \mathbf{C}_{L-1} \\ \mathbf{D}_2 & \mathbf{B}_2 & 0 & 0 & \cdots & 0 & 0 \\ 0 & \mathbf{D}_3 & \mathbf{B}_3 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \mathbf{D}_{L-1} & \mathbf{B}_{L-1} \end{bmatrix}, \tag{A.1.29}$$

$$\mathbf{B}_i = \begin{bmatrix} 0 & 0 & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & 1 - 2\mu g_{i+1,i-\frac{1}{2}} - \beta_{i+\frac{1}{2},i-\frac{1}{2}}(t) & 0 & \ldots & 0 \\ 0 & \mu g_{i+1,i-\frac{1}{2}} & 1 - 2\mu g_{i+2,i-\frac{1}{2}} - \beta_{i+\frac{3}{2},i-\frac{1}{2}}(t) & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & \mu g_{L-1,i-\frac{1}{2}} & 1 - 2\mu g_{L,i-\frac{1}{2}} - \beta_{L-\frac{1}{2},i-\frac{1}{2}}(t) \end{bmatrix},$$

$$\mathbf{C}_i = \begin{bmatrix} 0 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & \tilde{\beta}_{(i+1)-\frac{1}{2},i-\frac{1}{2}}(\frac{3}{2}h,t) & \cdots & \tilde{\beta}_{L-\frac{1}{2},i-\frac{1}{2}}(\frac{3}{2}h,t) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & \tilde{\beta}_{L-\frac{1}{2},i-\frac{1}{2}}((L-\frac{3}{2})h,t) \\ 0 & \cdots & 0 & 0 & 0 \end{bmatrix},$$

and

$$\mathbf{D}_i = \begin{bmatrix} 0 & 0 & 0 & ... & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & ... & \mu g_{i+\frac{1}{2},i-1} & 0 & ... & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & ... & 0 & 0 & \mu g_{L-\frac{1}{2},i-1} \end{bmatrix}$$

A.1.5 Monte-Carlo simulations

In this section, we describe the implementation of our Monte-Carlo simulations of the process underlying the adder-sizer mechanism. Suppose we have a list of cells at time t given as $S(t) = \{c_1(x_i, y_i, t, b_1), ..., c_i(x_i, y_{,i}, t, b_i)\}$, where x_i is cell c_i's volume and y_i is its added volume. The cell's division factor b_i is determined at birth, which is drawn from a uniform distribution $\mathbf{U}(0, 1)$.

Suppose we have a β of the form Eq. 2.1.11 and $\tilde{\beta}$ of the form Eq. 2.1.13. We set a time step $\Delta t = 0.01$, the maximum allowable time step, and determine the next state of the system at time t' by the following

- Step 1: For each cell i, calculate its age a_i at time t by the exponential growth law $\frac{\mathrm{d}x}{\mathrm{d}t} = \lambda x$. We require that $G_i = \int_0^{a_i} \gamma(a')\mathrm{d}a' < b_i$ at the beginning of each step for every i.

- Step 2: For each cell, calculate $G_i = \int_0^{a_i+\Delta t} \gamma(a')\mathrm{d}a'$. If $G_i \geq b_i$, then we numerical calculate a Δt_i such that $\int_0^{a_i+\Delta t_i} \gamma(a')\mathrm{d}a' \approx b_i$.

- Step 3: Choose the smallest Δt_i among all possible Δt_is as the new time step, set time $t' = t + \Delta t_i$ and let all cells gain an extra volume $\lambda x_i \Delta t_i$. If there is no such Δt_i, which means $G_i < b_i$ for every i, go to step 5.

- Step 4: Remove cell i from $S(t')$, record its volume x at t', and generate one random number $r \in (0, 1)$ observing a distribution which has a probability density function of $h(r)$, add two new cells in $S(t')$ as $c_m(rx, 0, t)$ and $c_{m+1}(x - rx, 0, t)$.

- Step 5: If $G_i < b_i$ for all i, then set $t' = t$ and let all cells gain an extra volume $\lambda x_i \Delta t_i$.

- Step 6: Return to step 1 until $t' > t_{max}$, the maximum time of the simulation.

Here, we set the initial added volume of all cells to zero so the condition in step 1 above is automatically satisfied at $t = 0$. For our runs, we used 10 cells of initial volume 0.5 and $t_{max} = T$ is the same as the maximum time for the numerical PDE experiments. We also generalize the model to incorporate the mother-daughter growth coefficient correlation by including a new label λ_i to each cell.

A.2 Appendix for Chapter 3

A.2.1 Conservation of probability

We now define probability fluxes

$$
\begin{aligned}
J_{m,n;m+1,n-1}(t) = (m+1)\int \mathrm{d}\mathbf{X}^m \mathrm{d}\mathbf{Y}^{2n-2}\mathrm{d}\mathbf{A}^m \mathrm{d}\mathbf{B}^{n-1} \int_{\mathbf{L}^3} \mathrm{d}y_1 \mathrm{d}y_2 \mathrm{d}s \\
\tilde{\beta}_{m+1,n-1}(y_1+y_2, y_1, s, t) \\
\times \rho_{m+1,n-1}(\mathbf{X}^{m+1}[\mathbf{X}^{m+1} = y_1+y_2], \mathbf{Y}^{2n-2}, \mathbf{A}^{m+1}[\mathbf{A}^{m+1} = s], \mathbf{B}^{n-1}, t), \\
J_{m,n;m-1,n}(t) = \frac{2n}{m}\int \mathrm{d}\mathbf{X}^m \mathrm{d}\mathbf{Y}^{2n-2}\mathrm{d}\mathbf{A}^m \mathrm{d}\mathbf{B}^{n-1} \int_{\mathbf{L}^2} \mathrm{d}y_1 \mathrm{d}y_2 \sum_{i=1}^{m} \\
\tilde{\beta}_{m-1,n}(y_1+y_2, y_1, A^i, t) \\
\times \rho_{m-1,n}(t, \mathbf{X}^m_{-i}, \mathbf{Y}^{2n}[Y^{2n-1} = X^i, Y^{2n} = y_1+y_2], \mathbf{A}^m_{-i}, \mathbf{B}^n[B^n = A^i], t), \\
J_{m,n;m',n'}(t) = 0, \quad \text{if } m+2n-m'-2n' \neq 1.
\end{aligned}
\tag{A.2.1}
$$

$J_{m,n;m',n'}(t)\mathrm{d}t$ is the probability flux within time $[t, t+\mathrm{d}t]$ from state (m', n') to state (m, n) arising from from cell division. When $\mathrm{d}t$ is sufficiently small, the probability that more than one cell divides during $[t, t+\mathrm{d}t]$ is $o(\mathrm{d}t)$, which is negligible, allowing us to set $J_{m,n;m',n'}(t) = 0$ if $m+2n-m'-2n' \neq 1$. We now verify the conservation of probability flux

$$
\begin{aligned}
& J_{m-1,n+1;m,n}(t) + J_{m+1,n;m,n}(t) \\
&\quad = \int \mathrm{d}\mathbf{X}^m \mathrm{d}\mathbf{Y}^{2n} \mathrm{d}\mathbf{A}^m \mathrm{d}\mathbf{B}^n \left(\sum_{i=1}^{m} \beta_{m,n}(A^i, t) + \sum_{i=j}^{n} 2\beta_{m,n}(B^j, t) \right) \rho_{m,n} \\
&\quad = \int \mathrm{d}\mathbf{X}^m \mathrm{d}\mathbf{Y}^{2n} \mathrm{d}\mathbf{A}^m \mathrm{d}\mathbf{B}^n \left(m\beta_{m,n}(A^m, t) + 2n\beta_{m,n}(B^n, t) \right) \rho_{m,n},
\end{aligned}
\tag{A.2.2}
$$

where $\rho_{m,n} = \rho_{m,n}(\mathbf{X}^m, \mathbf{Y}^{2n}, \mathbf{A}^m, \mathbf{B}^n, t)$. The first term is

$$
\begin{aligned}
J_{m-1,n+1;m,n}(t) &= m \int \mathrm{d}\mathbf{X}^{m-1}\mathrm{d}\mathbf{Y}^{2m}\mathrm{d}\mathbf{A}^{m-1}\mathrm{d}\mathbf{B}^{n} \int_{\mathbf{L}^3} \mathrm{d}y_1\mathrm{d}y_2\mathrm{d}A^m \\
&\qquad \tilde{\beta}_{m,n}(y_1+y_2, y_1, A^m, t)\rho_{m,n}(\mathbf{X}^m[X^m = y_1+y_2], \mathbf{Y}^{2n}, \mathbf{A}^m, \mathbf{B}^n, t) \\
&= m \int \mathrm{d}\mathbf{X}^{m-1}\mathrm{d}\mathbf{Y}^{2n}\mathrm{d}\mathbf{A}^{m-1}\mathrm{d}\mathbf{B}^{n} \int_{\mathbf{L}^2} \mathrm{d}A^m\mathrm{d}y \int_0^y \mathrm{d}s \\
&\qquad \tilde{\beta}_{m,n}(y, s, A^m, t)\rho_{m,n}(\mathbf{X}^m[X^m = y], \mathbf{Y}^{2n}, \mathbf{A}^m, \mathbf{B}^n, t) \\
&= m \int \mathrm{d}\mathbf{X}^{m-1}\mathrm{d}\mathbf{Y}^{2n}\mathrm{d}\mathbf{A}^{m-1}\mathrm{d}\mathbf{B}^{n} \int_{\mathbf{L}^2} \mathrm{d}A^m\mathrm{d}X^m\, \beta_{m,n}(A^m, t)\rho_{m,n}
\end{aligned}
\tag{A.2.3}
$$

which is exactly the first term on the right-hand side of Eq. (A.2.2). The second term

$$
\begin{aligned}
J_{m+1,n;m,n}(t) &= \frac{2n}{m+1} \int \mathrm{d}\mathbf{X}^{m+1}\mathrm{d}\mathbf{Y}^{2n}\mathrm{d}\mathbf{A}^{m+1}\mathrm{d}\mathbf{B}^{n-1} \int_{\mathbf{L}^2} \mathrm{d}y_1\mathrm{d}y_2 \sum_{i=1}^{m+1} \\
&\qquad \tilde{\beta}_{m,n}(y_1+y_2, y_1, A^i, t) \\
&\qquad \times \rho_{m,n}(\mathbf{X}_{-i}^{m+1}, \mathbf{Y}^{2n}[Y^{2n-1} = X^i, Y^{2n} = y_1+y_2], \mathbf{A}_{-i}^{m+1}, \mathbf{B}^n[B^n = A^i], t) \\
&= \frac{2n}{m+1} \sum_{i=1}^{m+1} \int \mathrm{d}\mathbf{X}^{m+1}\mathrm{d}\mathbf{Y}^{2n-2}\mathrm{d}\mathbf{A}^{m+1}\mathrm{d}\mathbf{B}^{n-1} \int_{\mathbf{L}} \mathrm{d}y \int_0^y \mathrm{d}s\, \tilde{\beta}_{m,n}(y, s, A^i, t) \\
&\qquad \times \rho_{m,n}(\mathbf{X}_{-i}^{m+1}, \mathbf{Y}^{2n}[Y^{2n-1} = X^i, Y^{2n} = y], \mathbf{A}_{-i}^{m+1}, \mathbf{B}^n[B^n = A^i], t) \\
&= 2n \int \mathrm{d}\mathbf{X}^{m}\mathrm{d}\mathbf{Y}^{2n}\mathrm{d}\mathbf{A}^{m}\mathrm{d}\mathbf{B}^{n}\, \beta_{m,n}(B^n, t)\rho_{m,n}
\end{aligned}
\tag{A.2.4}
$$

which is precisely the second term on the right-hand side of Eq. (A.2.2). We have thus verified that the probability flux out of state (m, n) due to cell division is the sum of probability currents into $(m-1, n+1)$ and into $(m+1, n)$. Summing up over m and n, we obtain for $m + n > 0$

$$\sum_{m,n=0}^{\infty}\big(J_{m-1,n+1;m,n}(t)+J_{m+1,n;m,n}(t)\big)= \\ \sum_{m,n=0}^{\infty}\int \mathrm{d}\mathbf{X}^m\mathrm{d}\mathbf{Y}^{2n}\mathrm{d}\mathbf{A}^m\mathrm{d}\mathbf{B}^n\big(m\beta_{m,n}(A^m,t)+2n\beta_{m,n}(B^n,t)\big)\rho_{m,n}. \tag{A.2.5}$$

Finally, it is readily observed that

$$\begin{aligned}&\sum_{m,n=0}^{\infty}\int \mathrm{d}\mathbf{X}^m\mathrm{d}\mathbf{Y}^{2n}\mathrm{d}\mathbf{A}^m\mathrm{d}\mathbf{B}^n\frac{\partial\rho_{m,n}}{\partial t}\\ &\quad=\sum_{m,n=0}^{\infty}\sum_{j=1}^{n}\int \mathrm{d}\mathbf{X}^m\mathrm{d}\mathbf{Y}^{2n}\mathrm{d}\mathbf{A}^m\mathrm{d}\mathbf{B}^n_{-j}\,\rho_{m,n}(\mathbf{X}^m,\mathbf{Y}^{2n},\mathbf{A}^m,\mathbf{B}^n[B^j=0],t)\\ &\qquad-\sum_{m,n=0}^{\infty}\int \mathrm{d}\mathbf{X}^m\mathrm{d}\mathbf{Y}^{2n}\mathrm{d}\mathbf{A}^m\mathrm{d}\mathbf{B}^n\bigg(\sum_{i=1}^{m}\beta_{m,n}(A^i,t)+2\sum_{j=1}^{n}\beta_{m,n}(B^j,t)\bigg)\rho_{m,n}\\ &\quad=\sum_{m=1}^{\infty}\sum_{n=0}^{\infty}(J_{m,n;m-1,n}-J_{m-1,n+1;m,n})-\sum_{m,n=0}^{\infty}J_{m+1,n;m,n}+\sum_{m=0}^{\infty}\sum_{n=1}^{\infty}J_{m,n;m+1,n-1}=0\end{aligned} \tag{A.2.6}$$

Therefore, we have verified that

$$\sum_{m=0}^{\infty}\sum_{n=0}^{\infty}\int \mathrm{d}\mathbf{X}^m\mathrm{d}\mathbf{Y}^{2n}\mathrm{d}\mathbf{A}^m\mathrm{d}\mathbf{B}^n\rho_{m,n}(\mathbf{X}^m,\mathbf{Y}^{2n},\mathbf{A}^m,\mathbf{B}^n,t)$$

is time-independent.

A.2.2 Explicit expressions for $u^{(k,\ell)}$

Below, we display the explicit expressions of $u^{(k,\ell)}$ in terms of $\rho^{(h,k,\ell)}_{m,n}$ for $k+\ell\le 2$:

$$\begin{aligned}u^{(1,0)}(x,a,t)=&\sum_{m,n=0}^{\infty}m\rho^{(1,0,0)}_{m,n}(X^1=x,A^1=a,t)\\ &+\sum_{m,n=0}^{\infty}2n\rho^{(0,1,0)}_{m,n}(\mathbf{Y}^2_{\mathrm{e}}[Y^2=x],B^1=a,t),\end{aligned} \tag{A.2.7}$$

$$u^{(0,1)}(y_1, y_2, b_1, t) = \sum_{m,n=0}^{\infty} 2n\rho_{m,n}^{(0,0,1)}(Y^1 = y_1, Y^2 = y_2, B^1 = b_1, t), \tag{A.2.8}$$

$$\begin{aligned} u^{(1,1)}(x_1, y_1, y_2, a_1, b_1, t) = \\ \sum_{m,n=0}^{\infty} 4n(n-1)\rho_{m,n}^{(0,1,1)}(\mathbf{Y}_{\mathrm{e}}^4[Y^2 = x_1, Y^3 = y_1, Y^4 = y_2], B^1 = a_1, B^2 = b_1, t) \\ + \sum_{m,n=0}^{\infty} 2mn\rho_{m,n}^{(1,0,1)}(X^1 = x_1, Y^1 = y_1, Y^2 = y_2, A^1 = a_1, B^1 = b_1, t), \end{aligned} \tag{A.2.9}$$

$$\begin{aligned} u^{(2,0)}(x_1, x_2, a_1, a_2, t) = \\ \sum_{m,n=0}^{\infty} m(m-1)\rho_{m,n}^{(2,0,0)}(X^1 = x_1, X^2 = x_2, A^1 = a_1, A^2 = a_2, t) \\ + \sum_{m,n=0}^{\infty} 2mn\rho_{m,n}^{(1,1,0)}(X^1 = x_1, A^1 = a_1, \mathbf{Y}_{\mathrm{e}}^2[Y^2 = x_2], B^1 = a_2, t) \\ + \sum_{m,n=0}^{\infty} 2mn\rho_{m,n}^{(1,1,0)}(X^1 = x_2, A^1 = a_2, \mathbf{Y}_{\mathrm{e}}^2[Y^2 = x_1], B^1 = a_1, t) \\ + \sum_{m,n=0}^{\infty} 4n(n-1)\rho_{m,n}^{(0,2,0)}(\mathbf{Y}_{\mathrm{e}}^4[Y^2 = x_1, Y^4 = x_2], B^1 = a_1, B^2 = a_2, t), \end{aligned} \tag{A.2.10}$$

$$u^{(0,2)}(y_1, y_2, y_3, y_4, b_1, b_2, t) = \sum_{m,n=0}^{\infty} 4n(n-1)\rho_{m,n}^{(0,0,2)}(\mathbf{Y}^4 = [y_1, y_2, y_3, y_4], \mathbf{B}^2 = [b_1, b_2], t). \tag{A.2.11}$$

A.2.3 Reduction to simpler models

Besides the general marginalizations we have considered (Eqs. (3.3.2) and (3.3.8)), we can define other useful quantities by *e.g.*, integrating over all volumes or all ages. Under some additional assumptions, these additional integrations reduce the kinetic theory simpler, known models. For example, if the solution $u^{(k,\ell)}$ of Eqs. (3.3.9) and (3.3.12) satisfies $\lim_{x_i \to \infty} \frac{\partial(\sigma^2 u^{(k,\ell)})}{\partial x_i} = \lim_{y_j \to \infty} \frac{\partial(\sigma^2 u^{(k,\ell)})}{\partial y_j} = 0$ for any i, j, integrating $u^{(k,\ell)}(\mathbf{x}^k, \mathbf{y}^{2\ell}, \mathbf{a}^k, \mathbf{b}^\ell, t)$ over all sizes $\mathbf{x}^k$ and $\mathbf{y}^{2\ell}$ yields

$$u_a^{(k,\ell)}(\mathbf{a}^k,\mathbf{b}^\ell,t) \coloneqq \int_{\mathbf{L}^{k+2\ell}} \mathrm{d}\mathbf{x}^k \mathrm{d}\mathbf{y}^{2\ell}\, u^{(k,\ell)}(\mathbf{x}^k,\mathbf{y}^{2\ell},\mathbf{a}^k,\mathbf{b}^\ell,t) \tag{A.2.12}$$

which satisfies

$$\frac{\partial u_a^{(k,\ell)}(\mathbf{a}^k,\mathbf{b}^{2\ell},t)}{\partial t} + \sum_{i=1}^{k} \frac{\partial u_a^{(k,\ell)}}{\partial a^i} + \sum_{j=1}^{\ell} \frac{\partial u_a^{(k,\ell)}}{\partial b^j} = -\bigg(\sum_{i=1}^{k} \beta(a^i,t) + \sum_{j=1}^{\ell} 2\beta(b^j,t) \bigg) u_a^{(k,\ell)}, \tag{A.2.13}$$

with corresponding boundary conditions

$$\begin{aligned} u_a^{(k,\ell)}(\mathbf{a}^k[a^v=0],\mathbf{b}^\ell,t) =& 2\int_{\mathbf{L}} \mathrm{d}a\, \beta(a,t) u_a^{(k,\ell)}(\mathbf{a}^k[a^k=a],\mathbf{b}^\ell,t) \\ &+ 2\sum_{w=1,\neq v}^{k} \beta(a^w,t) u_a^{(k-2,\ell+1)}(\mathbf{a}^k_{-v,-w},\mathbf{b}^{\ell+1}[b^{\ell+1}=a^w],t), \end{aligned} \tag{A.2.14}$$

$$\begin{aligned} u_a^{(k,\ell)}(\mathbf{a}^k,\mathbf{b}^\ell[b^v=0],t) =& 2\int_{\mathbf{L}} \mathrm{d}a\, \beta(a,t) u_a^{(k+1,\ell-1)}(\mathbf{a}^{k+1}[a^{k+1}=a],\mathbf{b}^\ell_{-v},t) \\ &+ 2\sum_{w=1}^{k} \beta(a^w,t) u_a^{(k-1,\ell)}(\mathbf{a}^k_{-w},\mathbf{b}^\ell[b^v=a^w],t), \end{aligned} \tag{A.2.15}$$

and $u_a^{(k,\ell)}(\mathbf{a}^k,\mathbf{b}^\ell,t)=0$ if two or more $a_i=0$ or $b_j=0$. This model describes age-structured cell populations similar to that discussed in [CG16].

On the other hand, integrating over age variables α, $\mathbf{b}$ defines size-dependent weighted densities

$$u_s^{(k,\ell)}(\mathbf{x}^k,\mathbf{y}^{2\ell},t) \coloneqq \int_{\mathbf{L}^{k+\ell}} \mathrm{d}\mathbf{a}^k \mathrm{d}\mathbf{b}^\ell\, u^{(k,\ell)}(\mathbf{x}^k,\mathbf{y}^{2\ell},\mathbf{a}^k,\mathbf{b}^\ell,t). \tag{A.2.16}$$

In this case, if $\tilde{\beta},\beta,g,\sigma$ do not depend on a, $n_s^{(k,\ell)}(\mathbf{x}^k,\mathbf{y}^{2\ell},t)$ is found to obey

$$
\begin{aligned}
&\frac{\partial u_s^{(k,\ell)}}{\partial t}(\mathbf{x}^k,\mathbf{y}^{2\ell},t)+\sum_{i=1}^{k}\frac{\partial(g(x_i,t)u_s^{(k,\ell)})}{\partial x^i}+\sum_{j=1}^{2\ell}\frac{\partial(g(y_j,t)u_s^{(k,\ell)})}{\partial y^j}= \\
&\quad -\sum_{i=1}^{k}(k+2\ell)\beta(t)u_s^{(k,\ell)}+\frac{1}{2}\sum_{i=1}^{k}\frac{\partial^2(\sigma^2(x^i,t)u_s^{(k,\ell)})}{(\partial x^i)^2}+\frac{1}{2}\sum_{j=1}^{2\ell}\frac{\partial^2(\sigma^2(y^j,t)u_s^{(k,\ell)})}{(\partial y^j)^2} \\
&\quad +2\sum_{v=1}^{k}\int_{\mathbf{L}}\mathrm{d}s\,\tilde{\beta}(x^v+s,x^v,t)u_s^{(k,\ell)}(\tilde{\mathbf{x}}^k[\tilde{x}^v=x^v+s],\mathbf{y}^{2\ell},t) \\
&\quad +2\sum_{v=1}^{k}\sum_{w=1,\neq v}^{k}\int_{\mathbf{L}}\mathrm{d}s\,\tilde{\beta}(x^w+s,x^w,t)u_s^{(k-2,\ell+1)}(\mathbf{x}^k_{-w,-v},\mathbf{y}^{2\ell+1}[y^{2\ell+1}=x^w,y^{2\ell+2}=x^v+s],t) \\
&\quad +2\sum_{v=1}^{\ell}\tilde{\beta}(y^{2v-1}+y^{2v},y^{2v},t)u_s^{(k+1,\ell-1)}(\mathbf{x}^{k+1}[x^{k+1}=y^{2v-1}+y^{2v}],\mathbf{y}^{2\ell}_{-(2v-1),-2v},t) \\
&\quad +2\sum_{v=1}^{\ell}\sum_{w=1}^{k}\tilde{\beta}(y^{2v-1}+y^{2v},y^{2v},t)u_s^{(k-1,\ell)}(\mathbf{x}^k_{-w},\tilde{\mathbf{y}}^{2\ell}[\tilde{y}^{2v-1}=y^{2v-1}+y^{2v},\tilde{y}^{2v}=x^w],t),
\end{aligned}
\tag{A.2.17}
$$

where $\tilde{\mathbf{x}}^k$ shares the same components with $\mathbf{x}^k$ except for the v^{th} element and $\tilde{\mathbf{y}}^{2\ell}$ shares $2\ell-2$ common components with $\mathbf{y}^{2\ell}$ except for the $(2v-1)^{\text{th}}$ and $(2v)^{\text{th}}$ elements, as indicated by the replacements [...] following each variable. By integrating over age, the boundary conditions in Eqs. (3.3.10) and (3.3.11) for newborn cells have been assimilated into Eq. (A.2.17). The remaining conditions are

$$
\begin{aligned}
u_s^{(k,\ell)}(\mathbf{x}^k[x^i=0],\mathbf{y}^{2\ell},t)=&u_s^{(k,\ell)}(\mathbf{x}^k[x^i=\infty],\mathbf{y}^{2\ell},t)=0, \\
u_s^{(k,\ell)}(\mathbf{x}^k,\mathbf{y}^{2\ell}[y^j=0],t)=&u_s^{(k,\ell)}(\mathbf{x}^k,\mathbf{y}^{2\ell}[y^j=\infty],t)=0.
\end{aligned}
\tag{A.2.18}
$$

Notice that if $k=1,\ell=0$, the last three terms on the RHS of Eq. (A.2.17) vanish and the equation reduces to the size-structured PDE model [Per08] except for the additional diffusion term describing growth noise.

A.3 Appendix for Chapter 4

A.3.1 Proof of Proposition 1

Here, we shall give proof of Prop. 1 as well as the additional assumptions we need. We shall apply Theorem 6.2 in [LOR15]. If $\vec{n} \neq \vec{n}^0$, then by definition $\hat{p}_{\vec{n}} = 0$, which solves Eq. (4.2.5). If $\vec{n} = \vec{n}^0$, for any smooth function $\phi \in C^\infty(\mathbb{R}^{|\vec{n}|_1})$, we define the measure

$$\begin{aligned}\gamma^m(\phi, t) &= \int_{\mathcal{C}^{|\vec{n}|}} \phi(\vec{X}_{\vec{n}}(t;\omega)) \exp\Big(- \int_0^t \sum_{i=1}^k \sum_{j=1}^{n_i} \big(\beta_{i,j}(X_{i,j}(s;\omega) + \mu_{i,j}(X_{i,j}(s;\omega)\big) \mathrm{d}s\Big) \mathrm{d}m(\omega), \\ \vec{X}_{\vec{n}}(0) &= \vec{X}_{\vec{n}^0}(0)\end{aligned} \tag{A.3.1}$$

where $\mathcal{C}^d := \mathcal{C}([0,t], \mathbb{R}^d)$ (the integration is taken all realization of $\vec{X}_{\vec{n}}(t;\omega)$). Using Theorem 6.2 in [LOR15], $\gamma^m(\phi, t)$ solves the PDE

$$\begin{aligned}\frac{\partial \gamma^m}{\partial t} + \sum_{i=1}^k \sum_{j=1}^{n_i} \frac{\partial (g_{i,j}(X_{i,j}, t)\gamma^m)}{\partial X_{i,j}} - \frac{1}{2} \sum_{i=1}^k \sum_{j=1}^{n_i} \frac{\partial^2 (\sigma_{i,j}^2(X_{i,j}, t)\gamma^m)}{(\partial X_{i,j})^2} \\ = - \sum_{i=1}^k \sum_{j=1}^{n_i} \big(\beta_{i,j}(X_{i,j}(t)) + \mu_{i,j}(X_{i,j}(t))\big)\gamma^m\end{aligned} \tag{A.3.2}$$

in the sense of distributions. Letting $K^\epsilon = \frac{1}{\epsilon^{|\vec{n}|_1}} K(\cdot)$ where $K(\cdot)$ is a smooth mollifier and we define

$$u^\epsilon(\vec{X}_{\vec{n}}, t) = \gamma^m(K^\epsilon(\cdot - \vec{X}_{\vec{n}}), t), \tag{A.3.3}$$

i.e.,

$$u^\epsilon(\vec{X}_{\vec{n}}, t) = \mathbb{E}\bigg[K^\epsilon(\vec{X}_{\vec{n}}(t) - \vec{X}_{\vec{n}}) \exp\Big(- \int_0^t \sum_{i=1}^k \sum_{j=1}^{n_i} \big(\beta_{i,j}(X_{i,j}(s)) + \mu_{i,j}(X_{i,j}(s))\big) \mathrm{d}s\Big) \Big| \vec{X}_{\vec{n}^0}(0), 0 \bigg]. \tag{A.3.4}$$

By Eq. (A.3.2), we have

$$
\begin{aligned}
\frac{\partial u^\epsilon}{\partial t}(\vec{X}_{\vec{n}},t) = \mathbb{E}\Big[& \sum_{i=1}^{k}\sum_{j=1}^{n_i} \partial_{X_{i,j}(t)} K^\epsilon(\vec{X}_{\vec{n}}(t)-\vec{X}_{\vec{n}}) \\
& \times g_{i,j}(X_{i,j}(t),t)\cdot \exp\Big(-\int_0^t \sum_{i=1}^{k}\sum_{j=1}^{n_i}\big(\beta_{i,j}(X_{i,j})+\mu_{i,j}(X_{i,j})\big)\mathrm{d}s\Big)\Big|\vec{X}_{\vec{n}^0}(0),0\Big] + \\
\mathbb{E}\Big[& \sum_{i=1}^{k}\sum_{j=1}^{n_i} \frac{1}{2}\partial^2_{X_{i,j}(t)} K^\epsilon(\vec{X}_{\vec{n}}(t)-\vec{X}_{\vec{n}}) \\
& \times \sigma^2_{i,j}(X_{i,j}(t),t)\cdot \exp\Big(-\int_0^t \sum_{i=1}^{k}\sum_{j=1}^{n_i}\big(\beta_{i,j}(X_{i,j})+\mu_{i,j}(X_{i,j})\big)\mathrm{d}s\Big)\Big|\vec{X}_{\vec{n}^0}(0),0\Big] - \\
\mathbb{E}\Big[& \sum_{i=1}^{k}\sum_{j=1}^{n_i} \big(\beta_{i,j}(X_{i,j}(t))+\mu_{i,j}(X_{i,j}(t))\big) K^\epsilon(\vec{X}_{\vec{n}}-\vec{X}_{\vec{n}}(t)) \\
& \times \exp\Big(-\int_0^t \sum_{i=1}^{k}\sum_{j=1}^{n_i}\big(\beta_{i,j}(X_{i,j})+\mu_{i,j}(X_{i,j})\big)\mathrm{d}s\Big)\Big|\vec{X}_{\vec{n}^0}(0),0\Big].
\end{aligned}
\tag{A.3.5}
$$

The assumption that we shall impose for Prop. 1 is that: i) the limit

$$
u \coloneqq \lim_{\epsilon\to 0^+} u^\epsilon = \mathbb{E}\Big[\delta(\vec{X}_{\vec{n}}(t)-\vec{X}_{\vec{n}})\exp\Big(-\int_0^t \sum_{i=1}^{k}\sum_{j=1}^{n_i}\big(\beta(X_{i,j})+\mu(X_{i,j})\big)\mathrm{d}s\Big)\Big|\vec{X}_{\vec{n}^0}(0),0\Big] \tag{A.3.6}
$$

exists, and ii) taking the limit $\epsilon \to 0^+$ is interchangeable with taking the expectation and taking the derivative w.r.t. t and $X_{i,j}$, *i.e.*,

$$\begin{aligned}
\frac{\partial u}{\partial t}(\vec{X}_{\vec{n}}, t) = \mathbb{E}\Big[& \sum_{i=1}^{k}\sum_{j=1}^{n_i} \partial_{X_{i,j}(t)}\delta(\vec{X}_{\vec{n}}(t) - \vec{X}_{\vec{n}}) \cdot g_{i,j}(X_{i,j}(t), t) \\
& \times \exp\Big(- \int_0^t \sum_{i=1}^{k}\sum_{j=1}^{n_i} \big(\beta_{i,j}(X_{i,j}) + \mu_{i,j}(X_{i,j})\big)\mathrm{d}s\Big)\Big| \vec{X}_{\vec{n}^0}(0), 0\Big] \\
+ \mathbb{E}\Big[& \sum_{i=1}^{k}\sum_{j=1}^{n_i} \frac{1}{2}\partial^2_{X_{i,j}(t)}\delta(\vec{X}_{\vec{n}}(t) - \vec{X}_{\vec{n}}) \\
& \times \sigma^2_{i,j}(X_{i,j}(t), t) \cdot \exp\Big(- \int_0^t \sum_{i=1}^{k}\sum_{j=1}^{n_i} \big(\beta_{i,j}(X_{i,j}) + \mu_{i,j}(X_{i,j})\big)\mathrm{d}s\Big)\Big| \vec{X}_{\vec{n}^0}(0), 0\Big] \\
- \mathbb{E}\Big[& \sum_{i=1}^{k}\sum_{j=1}^{n_i} (\beta_{i,j}(X_{i,j}(t)) + \mu_{i,j}(X_{i,j}(t))) \cdot \delta(\vec{X}_{\vec{n}}(t) - \vec{X}_{\vec{n}}) \\
& \times \exp\Big(- \int_0^t \sum_{i=1}^{k}\sum_{j=1}^{n_i} \big(\beta_{i,j}(X_{i,j}) + \mu_{i,j}(X_{i,j})\big)\mathrm{d}s\Big)\Big| \vec{X}_{\vec{n}^0}(0), 0\Big].
\end{aligned} \tag{A.3.7}$$

By integration in parts, the partial differential equation satisfied by u is

$$\frac{\partial u}{\partial t} + \sum_{i=1}^{k}\sum_{j=1}^{n_i} \frac{\partial(g_i(X_{i,j})u)}{\partial X_{i,j}} - \frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{n_i} \frac{\partial^2(\sigma_i^2(X_{i,j})u)}{(\partial X_{i,j})^2} = -\sum_{i=1}^{k}\sum_{j=1}^{n_i} \big(\beta(X_{i,j}) + \mu(X_{i,j})\big)u. \tag{A.3.8}$$

Finally, we can also write u as

$$\mathbb{E}\Big[\delta(\vec{X}_{\vec{n}}(t) - \vec{X}_{\vec{n}}) \exp\Big(- \int_0^t \sum_{i=1}^{k}\sum_{j=1}^{n_i} \big(\beta(X_{i,j}) + \mu(X_{i,j})\big)\mathrm{d}s\Big)\Big| \vec{n}(s) = \vec{n}^0, s \in [0,t], \vec{X}_{\vec{n}^0}(0), 0\Big] \tag{A.3.9}$$

because the number of particles is a constant, which proves Proposition 1.

A.3.2 Proof of Proposition 2

We prove this lemma by induction on m. Clearly, when $m = 0, 1$, $p^{(0)}$ and $p^{(1)}$ solve Eq. (4.2.10) by using Proposition 1. If the conclusion holds for $m - 1, m \geq 2$, then if

$\vec{n} \neq \vec{n}^0$, we have

$$
\begin{aligned}
\frac{\partial p_{\vec{n}}^{m+1}}{\partial t} &= \mathbb{E}\Big[\exp\Big(- \int_0^t \sum_{i=1}^{k^0} \sum_{j=1}^{n_i^0} \big(\beta_{i,j}(X_{i,j}(r)) + \mu_{i,j}(X_{i,j}(r))\big) \mathrm{d}r \Big) \\
&\times \Big[\sum_{i=1}^{k^0} \sum_{j=1}^{n_i^0} (\tilde{\beta}_{i,j}(X_{i,j}(t), X_1, X_2) p_{\vec{n}}^m(\vec{X}_{\vec{n}}, 0 | \vec{X}_{\vec{n}^0_{\mathrm{b},-i,-j}}(t), 0)) + \sum_{i=1}^{k^0} \sum_{j=1}^{n_i^0} (\mu_{i,j}(X_{i,j}(t)) p_{\vec{n}}^m(\vec{X}_{\vec{n}}, 0 | \vec{X}_{\vec{n}^0_{\mathrm{d},-i,-j}}(t), 0)) \Big] \Big| \vec{X}_{\vec{n}^0}(0), 0 \Big] \\
&+ \mathbb{E}\Big[\int_0^t \exp\Big(- \int_0^s \sum_{i=1}^{k^0} \sum_{j=1}^{n_i^0} \big(\beta_{i,j}(X_{i,j}(r)) + \mu(X_{i,j}(r))\big) \mathrm{d}r \Big) \cdot \Big[\sum_{i=1}^{k^0} \sum_{j=1}^{n_i^0} \big(\tilde{\beta}_{i,j}(X_{i,j}(s), X_1, X_2) \frac{\partial p_{\vec{n}}^{(d)}}{\partial t}(\vec{X}_{\vec{n}}, t-s | \vec{X}_{\vec{n}^0_{\mathrm{b},-i,-j}}(s), 0) \\
&\quad + \mu_{i,j}(X_{i,j}(s)) \frac{\partial p_{\vec{n}}^m}{\partial t}(\vec{X}_{\vec{n}}, t-s | \vec{X}_{\vec{n}^0_{\mathrm{d},-i,-j}}(s), 0)) \Big] \mathrm{d}s \Big| \vec{X}_{\vec{n}^0}(0), 0 \Big] \\
&= - \sum_{i=1}^{k} \sum_{j=1}^{n_i} \frac{\partial (g_{i,j}(X_{i,j}, t) p_{\vec{n}}^{m+1})}{\partial X_{i,j}} + \frac{1}{2} \sum_{i=1}^{k} \sum_{j=1}^{n_i} \frac{\partial^2 (\sigma_{i,j}^2(X_{i,j}, t) p_{\vec{n}}^{m+1})}{(\partial X_{i,j})^2} + \sum_{i=1}^{k-1} \sum_{j=1}^{n_i^{\mathrm{b},i}} \int \tilde{\beta}_{i,j}(Y, X_{i+1,n_{r+1}-1}, X_{r+1,n_{i+1}}) \\
&\quad \times \mathbb{E}\Big[\delta(X_{i,j} - Y) \delta(\vec{X}_{\vec{n}^0_{\mathrm{b},-i,-j}}(t) - \vec{X}_{\vec{n}}) \delta_{\vec{n}^0_{\mathrm{b},-i}, \vec{n}} \exp\Big(- \int_0^t \sum_{i=1}^{k^0} \sum_{j=1}^{n_i^0} \big(\beta_{i,j}(X_{i,j}) + \mu_{i,j}(X_{i,j})\big) \mathrm{d}s \Big) \Big| \vec{X}_{\vec{n}^0}, 0 \Big] \mathrm{d}Y \\
&+ \sum_{i=1}^{k-1} \sum_{j=1}^{n_i^{\mathrm{b},i}} \int \tilde{\beta}_{i,j}(Y, X_{i+1,n_{i+1}-1}, X_{i+1,n_{i+1}}) \mathbb{E}\Big[\int_0^t \exp\Big(- \int_0^s \sum_{r=1}^{k^0} \sum_{\ell=1}^{n_r^0} \big(\beta_{r,\ell}(X_{r,\ell}) + \mu_{r,\ell}(X_{r,\ell})\big) \mathrm{d}r \Big) \\
&\quad \times \Big[\sum_{r=1}^{k^0} \sum_{\ell=1}^{n_r^0} \big(\tilde{\beta}_{r,\ell}(X_{r,\ell}, X_1, X_2) p_{\vec{n}_{\mathrm{b},i}}^{m-1}(\vec{X}_{\vec{n}_{\mathrm{b},i,j}}, t-s | \vec{X}_{\vec{n}^0_{\mathrm{b},-r,-\ell}}(s), 0) \\
&\quad + \mu_{r,\ell}(X_{r,\ell}) p_{\vec{n}_{\mathrm{b},i}}^{m-1}(\vec{X}_{\vec{n}_{\mathrm{b},i,j}}, t-s | \vec{X}_{\vec{n}^0_{\mathrm{d},-r,-\ell}}(s), 0)) \Big] \mathrm{d}s \Big| \vec{X}_{\vec{n}^0}(0), 0 \Big] \mathrm{d}Y + \sum_{i=1}^{\infty} \sum_{j=1}^{n_i^{\mathrm{d},i}} \int \mu_{i,j}(Y) \\
&\quad \times \mathbb{E}\Big[\delta(X_{i,j} - Y) \delta(\vec{X}_{\vec{n}^0_{\mathrm{d},-i,-j}}(t) - \vec{X}_{\vec{n}}) \delta_{\vec{n}, \vec{n}^0_{\mathrm{d},-i}} \exp\Big(- \int_0^t \sum_{r=1}^{k^0} \sum_{\ell=1}^{n_r^0} \big(\beta_{r,\ell}(X_{r,\ell}) + \mu_{r,\ell}(X_{r,\ell})\big) \mathrm{d}s \Big) \Big| \vec{X}_{\vec{n}^0}(0), 0 \Big] \mathrm{d}Y \\
&\quad + \sum_{i=1}^{\infty} \sum_{j=1}^{n_i^{\mathrm{d},i}} \int \mu_{i,j}(Y) \mathbb{E}\Big[\int_0^t \exp\Big(- \int_0^s \sum_{r=1}^{k^0} \sum_{\ell=1}^{n_r^0} \big(\beta_{r,\ell}(X_{r,\ell}(v), v) + \mu_{r,\ell}(X_{r,\ell}(v), v)\big) \mathrm{d}v \Big) \\
&\times \Big[\sum_{r=1}^{k^0} \sum_{\ell=1}^{n_r^0} \big(\tilde{\beta}_{r,\ell}(X_{r,\ell}(s), X_1, X_2) p_{\vec{n}_{\mathrm{d},i,j}}^m(\vec{X}_{\vec{n}_{\mathrm{d},i,-j}}, t-s | \vec{X}_{\vec{n}^0_{\mathrm{b},-r,-\ell}}, 0) \\
&\quad + \mu_{r,\ell}(X_{r,\ell}) p_{\vec{n}_{\mathrm{d},i,j}}^m(\vec{X}_{\vec{n}_{\mathrm{d},i,j}}, t-s | \vec{X}_{\vec{n}^0_{\mathrm{d},-r,-\ell}}, 0)) \Big] \mathrm{d}s \Big| \vec{X}_{\vec{n}^0}(0), 0 \Big] \mathrm{d}Y \\
&- \sum_{i=1}^{k} \sum_{j=1}^{n_i} \big(\beta_{i,j}(X_{i,j}) + \mu_{i,j}(X_{i,j})\big) p_{\vec{n}}^{m+1} = - \sum_{i=1}^{k} \sum_{j=1}^{n_i} \frac{\partial (g_{i,j} p_{\vec{n}}^{m+1})}{\partial X_{i,j}} + \frac{1}{2} \sum_{i=1}^{k} \sum_{j=1}^{n_i} \frac{\partial^2 (\sigma_{i,j} p_{\vec{n}}^{m+1})}{(\partial X_{i,j})^2} \\
&\quad + \sum_{i=1}^{k-1} \sum_{j=1}^{n_i^{\mathrm{b},i}} \int \tilde{\beta}(Y, X_{i+1,n_{i+1}-1}, X_{i+1,n_{i+1}}) p_{\vec{n}_{\mathrm{b},i}}^m(\vec{X}_{\vec{n}_{\mathrm{b},i,j}}, t | \vec{X}_{\vec{n}^0}, \vec{n}^0, 0) \mathrm{d}Y + \sum_{i=1}^{\infty} \sum_{j=1}^{n_i^{\mathrm{d},i}} \int \mu(Y) p_{\vec{n}_{\mathrm{d},i}}^m(\vec{X}_{\vec{n}_{\mathrm{d},i,j}}, t | \vec{X}_{\vec{n}^0}(0), 0) \mathrm{d}Y.
\end{aligned}
\tag{A.3.10}
$$

Here, $n_i^{\mathrm{b},i} \coloneqq n_i + 1$ denotes the number of cells in the i^{th} generation before the j^{th} cell in the i^{th} generation divides, and $n_i^{\mathrm{d},i} \coloneqq n_i + 1$ denotes the number of cells in the i^{th} generation before the j^{th} cell in the i^{th} generation dies. Similarly, using Proposition 1 to observe that

the quantity

$$\mathbb{E}\Big[\delta(\vec{X}_{\vec{n}}(t)-\vec{X}_{\vec{n}})\exp\Big(-\int_0^t\sum_{i=1}^{k^0}\sum_{j=1}^{n_i^0}\big(\beta_{i,j}(X_{i,j}(s))+\mu_{i,j}(X_{i,j}(s))\big)\mathrm{d}s\Big)\Big|\vec{X}_{\vec{n}^0}(0),0\Big] \quad \text{(A.3.11)}$$

satisfies Eq. (4.2.5), we can check that Eq. (4.2.10) also holds for $\vec{n}^0=\vec{n}$. Additionally, if $p_{\vec{n}}^m\geq p_{\vec{n}}^{m-1}$ for any $\vec{n},\vec{X}_{\vec{n}}$, denoting $\Delta_{\vec{n}}^m=p_{\vec{n}}^m-p_{\vec{n}}^{m-1}$, we have

$$\begin{aligned}\Delta_{\vec{n}}^{m+1}=\mathbb{E}\Bigg[&\int_0^t\exp\Big(-\int_0^s\sum_{i=1}^{k^0}\sum_{j=1}^{n_i^0}\big(\beta_{i,j}(X_{i,j}(r))+\mu_{i,j}(X_{i,j}(r))\big)\mathrm{d}r\Big)\\&\times\Bigg[\sum_{i=1}^{k^0}\sum_{j=1}^{n_i^0}\big(\tilde{\beta}_{i,j}(X_{i,j}(s),X_1,X_2)\Delta_{\vec{n}}^m(\vec{X}_{\vec{n}},t-s|\vec{X}_{\vec{n}^0_{\mathrm{b},-i,-j}}(s),0)\\&\quad+\mu_{i,j}(X_{i,j})\Delta_{\vec{n}}^m(\vec{X}_{\vec{n}},t-s|\vec{X}_{\vec{n}_{\mathrm{d},-i,-j}}(s),0)\big)\Bigg]\mathrm{d}s\Bigg]\geq 0.\end{aligned} \quad \text{(A.3.12)}$$

Therefore, $p_{\vec{n}}^{m+1}\geq p_{\vec{n}}^m$. Since $p_{\vec{n}}^{m+1}\geq p_{\vec{n}}^m$ holds for $m=0$, $p_{\vec{n}}^m$ is non-decreasing for all $m\in\mathbb{N}$ by induction.

A.3.3 Differential equations satisfied by $X^q(t), q \in \mathbb{N}^+$

With $X^q(t)$ defined in Eq. (4.3.18), it can be shown that

$$\begin{aligned}
\frac{\mathrm{d}X^q(t)}{\mathrm{d}t} &= q\sum_{\vec{n}}\int (\sum_{i=1}^{k}\sum_{j=1}^{n_i} X_{i,j})^{q-1}\cdot\sum_{i=1}^{k}\sum_{j=1}^{n_i} g_i(X_{i,j},t)\cdot\rho_{\vec{n}}(\vec{X}_{\vec{n}},t)\mathrm{d}\vec{X}_{\vec{n}} \\
&\quad + \frac{q(q-1)}{2}\sum_{\vec{n}}\int (\sum_{i=1}^{k}\sum_{j=1}^{n_i} X_{i,j})^{q-2}\cdot\sum_{i=1}^{k}\sum_{j=1}^{n_i}\sigma_i^2(X_{i,j},t)\cdot\rho_{\vec{n}}(\vec{X}_{\vec{n}},t)\mathrm{d}\vec{X}_{\vec{n}} \\
&\quad - \sum_{\vec{n}}\int \Big(\sum_{i=1}^{k}\sum_{j=1}^{n_i}\mu_i(X_{i,j},t)\cdot\big(\sum_{r=1}^{q}(-1)^{r-1}\binom{q}{r}(\sum_{i'=1}^{k}\sum_{j'=1}^{n_{i'}} X_{i',j'})^{q-r}\cdot X_{i,j}^r\big)\Big)\rho_{\vec{n}}(\vec{X}_{\vec{n}},t)\mathrm{d}\vec{X}_{\vec{n}} \\
&\quad - \sum_{\vec{n}}\int (\sum_{i=1}^{k}\sum_{j=1}^{n_i} X_{i,j})^{q}\cdot\sum_{i=1}^{k}\sum_{j=1}^{n_i}\beta_i(X_{i,j})\cdot\rho_{\vec{n}}(\vec{X}_{\vec{n}},t)\mathrm{d}\vec{X}_{\vec{n}} \\
&\quad + \sum_{\vec{n}}\int (\sum_{i=1}^{k}\sum_{j=1}^{n_i} X_{i,j})^{q} \\
&\qquad \times\sum_{i=1}^{k-1}\frac{n_i+1}{n_{i+1}(n_{i+1}-1)}\sum_{j_1\neq j_2}\int\tilde{\beta}(Y,X_{i+1,j_1},X_{i+1,j_2})\rho_{\vec{n}_{\mathrm{b},i}}(\vec{X}_{\vec{n}_{\mathrm{b},i,j_1,j_2}},t)\mathrm{d}Y\mathrm{d}\vec{X}_{\vec{n}},\quad q>1.
\end{aligned} \tag{A.3.13}$$

Specifically, if X is a conserved quantity at division, then the evolution of the second-order moment can be further simplified as

$$\begin{aligned}
\frac{\mathrm{d}X^q(t)}{\mathrm{d}t} &= q\sum_{\vec{n}}\int (\sum_{i=1}^{k}\sum_{j=1}^{n_i} X_{i,j})^{q-1}\cdot\sum_{i=1}^{k}\sum_{j=1}^{n_i} g_i(X_{i,j},t)\cdot\rho_{\vec{n}}(\vec{X}_{\vec{n}},t)\mathrm{d}\vec{X}_{\vec{n}} \\
&\quad + \sum_{\vec{n}}\frac{q(q-1)}{2}\int (\sum_{i=1}^{k}\sum_{j=1}^{n_i} X_{i,j})^{q-2}\cdot(\sum_{i=1}^{k}\sum_{j=1}^{n_i}\cdot\sigma_i^2(X_{i,j},t)\cdot\rho_{\vec{n}}(\vec{X}_{\vec{n}},t)\mathrm{d}\vec{X}_{\vec{n}} \\
&\quad - \sum_{\vec{n}}\int \Big(\sum_{i=1}^{k}\sum_{j=1}^{n_i}\mu_i(X_{i,j},t)\cdot\big(\sum_{r=1}^{q}(-1)^{r-1}\binom{q}{r}(\sum_{i'=1}^{k}\sum_{j'=1}^{n_{i'}} X_{i',j'})^{q-r}\cdot X_{i,j}^r\big)\Big)\cdot\rho_{\vec{n}}(\vec{X}_{\vec{n}},t)\mathrm{d}\vec{X}_{\vec{n}}
\end{aligned} \tag{A.3.14}$$

Eq. (A.3.14) can be further simplified if the coefficients g_i and σ_i satisfy certain conditions. For example, if the cells grow exponentially, *i.e.*, $g_i(X_{i,j},t)=\lambda X_{i,j}$ and $\sigma_i^2(X_{i,j},t)=\sigma^2 X_{i,j}$

Eq. (A.3.14) can be simplified as

$$\frac{\mathrm{d}X^q(t)}{\mathrm{d}t} = \lambda q X^q(t) + \sigma^2 \frac{q(q-1)}{2} X^{q-1}(t)$$
$$- \sum_{\vec{n}} \int \Big(\sum_{i=1}^{k} \sum_{j=1}^{n_i} \mu_i(X_{i,j}, t) \cdot \big(\sum_{r=1}^{q} (-1)^{i-1} \binom{q}{r} \big(\sum_{i'=1}^{k} \sum_{j'=1}^{n_{i'}} X_{i',j'} \big)^{q-r} \cdot X_{i,j}^r \big) \Big) \cdot \rho_{\vec{n}}(\vec{X}_{\vec{n}}, t) \mathrm{d}\vec{X}_{\vec{n}}. \tag{A.3.15}$$

A.3.4 Birth-induced boundary conditions

We can also consider X which has a component that is reset to 0 at division, *e.g.*, cell's age. Here, we shall consider a simple case where one new cell with age 0 will be created at division, while the mother cell's age as well as its generation will not change ("budding"). In this case, we shall be tracking the cell's volume denoted by X, and the cell's age denoted by A. We assume that in the i^{th} generation, there are n_i singlets with sizes $(X_{i,1}, ..., X_{i,n_i})$ and ages $(A_{i,1}, ..., A_{i,n_i})$. Similarly to Prop. 1, we can show that the solution to

$$\frac{\partial \hat{p}_{\vec{n}}}{\partial t}(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t | \vec{A}_{\vec{n}^0}(0); \vec{X}_{\vec{n}^0}(0), 0) + \sum_{i=1}^{k} \sum_{j=1}^{n_i} \frac{\partial (g_{i,j}(A_{i,j}, X_{i,j}, t) \hat{p}_{\vec{n}})}{\partial X_{i,j}} - \frac{1}{2} \sum_{i=1}^{k} \sum_{j=1}^{n_i^s} \frac{\partial^2 (\sigma_{i,j}^2(A_{i,j}, X_{i,j}, t) \hat{p}_{\vec{n}})}{(\partial X_{i,j})^2}$$
$$+ \sum_{i=1}^{k} \sum_{j=1}^{n_i} \frac{\partial \hat{p}_{\vec{n}}}{\partial A_{i,j}} = - \sum_{i=1}^{k} \sum_{j=1}^{n_i^s} \big(\beta_{i,j}(A_{i,j}, X_{i,j}) + \mu_{i,j}(A_{i,j}, X_{i,j}) \big) \hat{p}_{\vec{n}},$$
$$\hat{p}_{\vec{n}}(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, 0 | \vec{A}_{\vec{n}^0}(0); \vec{X}_{\vec{n}^0}(0), 0) = \delta(\vec{X}_{\vec{n}^0}(0) - \vec{X}_{\vec{n}}) \delta(\vec{A}_{\vec{n}^0}(0) - \vec{A}_{\vec{n}}), \text{ if } \vec{n} = \vec{n}^0,$$
$$\hat{p}_{\vec{n}}(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, 0) = 0 \text{ if } \vec{n} \neq \vec{n}^0 \tag{A.3.16}$$

is

$$\hat{p}_{\vec{n}}(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t | \vec{A}_{\vec{n}^0}(0), \vec{X}_{\vec{n}^0}(0), 0) := \mathbb{E} \Big[\delta(\vec{X}_{\vec{n}}(t) - \vec{X}_{\vec{n}}) \delta(\vec{A}_{\vec{n}}(t) - \vec{A}_{\vec{n}}))$$
$$\times \exp \Big(- \int_0^t \sum_{i=1}^{k} \sum_{j=1}^{n_i} \big(\beta_{i,j}(A_{i,j}, X_{i,j}) + \mu_{i,j}(A_{i,j}, X_{i,j}) \big) \mathrm{d}s \Big) \Big| \vec{A}_{\vec{n}^0}(0), \vec{X}_{\vec{n}^0}(0), 0 \Big], \text{ if } \vec{n} = \vec{n}^0,$$
$$\hat{p}_{\vec{n}}(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t) = 0, \text{ if } \vec{n} \neq \vec{n}^0 \tag{A.3.17}$$

Furthermore, if we recursively define

$$
\begin{aligned}
&p^0(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t|\vec{A}_{\vec{n}^0}(0), \vec{X}_{\vec{n}^0}(0), 0) = 0,\\
&p^1(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t|\vec{A}_{\vec{n}^0}(0), \vec{X}_{\vec{n}^0}(0), 0) = \hat{p}_{\vec{n}}(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t|\vec{A}_{\vec{n}^0}(0), \vec{X}_{\vec{n}^0}(0), 0)
\end{aligned}
\tag{A.3.18}
$$

and

$$
\begin{aligned}
&p_{\vec{n}}^{m+1}(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t|\vec{A}_{\vec{n}^0}(0), \vec{X}_{\vec{n}^0}(0), 0) = \hat{p}_{\vec{n}}(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t|\vec{A}_{\vec{n}^0}(0), \vec{X}_{\vec{n}^0}(0), 0)\\
&\quad + \mathbb{E}\bigg[\int_0^t \exp\Big(-\int_0^s \sum_{i=1}^{k}\sum_{j=1}^{n_i^0}\big(\beta_{i,j}(A_{i,j}(r), X_{i,j}(r)) + \mu_{i,j}(A_{i,j}(r), X_{i,j}(r))\big)\mathrm{d}r\Big)\\
&\qquad \times \bigg[\sum_{i=1}^{k^0}\sum_{j=1}^{n_i^0}\big(\tilde{\beta}_{i,j}(A_{i,j}, X_{i,j}, Y_{i,j}, Y_{i+1,n_{i+1}^0+2})p_{\vec{n}}^m(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t-s|\vec{A}_{\vec{n}^0_{\mathrm{b};i,j}}(s), \vec{X}_{\vec{n}^0_{\mathrm{b};i,j}}(s), 0)\\
&\qquad\quad + \mu_{i,j}(A_{i,j}, X_{i,j})p_{\vec{n}}^m(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t-s|\vec{A}_{\vec{n}^0_{\mathrm{d},-i,-j}}(s), \vec{X}_{\vec{n}^0_{\mathrm{d},-i,-j}}(s), 0)\big)\bigg]\mathrm{d}s\Big|\vec{A}_{\vec{n}^0}(0), \vec{X}_{\vec{n}^0}(0), 0\bigg],\\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{if } \vec{A}_{\vec{n}}(0) > 0,\\
&p_{\vec{n}}^{m+1}(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t|\vec{A}_{\vec{n}^0}(0), \vec{X}_{\vec{n}^0}(0), 0) =\\
&\qquad \mathbb{E}\Big[\sum_{i=1}^{k}\sum_{j=1}^{n_i}\tilde{\beta}(A_{i,j}, Y, X_{i,j}, X_{i+1,n_{i+1}})p_{\vec{n}}^m(\vec{A}_{\vec{n}_{\mathrm{b},-i,-j}}(t), \vec{X}_{\vec{n}_{\mathrm{b},-i,-j}}(t), t|\vec{A}_{\vec{n}}(0), \vec{X}_{\vec{n}}(0), 0)\Big|\vec{A}_{\vec{n}^0}(0), \vec{X}_{\vec{n}^0}(0), 0\Big],\\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{if } A_{i+1,n_{i+1}} = 0.
\end{aligned}
\tag{A.3.19}
$$

Here, $\tilde{\beta}(A_{i,j}, Y, X_{i,j}, X_{i+1,n_{i+1}})$ is the rate of a cell in the i^{th} generation giving birth to a cell in the $(i+1)^{\text{th}}$ generation with the state $X_{i+1,n_{i+1}}$ and its own state shifting to $X_{i,j}$. $\vec{A}_{\vec{n}^0_{\mathrm{b},-i,-j}}$ differs from $\vec{A}_{\vec{n}^0}$ in that its $(i+1)^{\text{th}}$ generation has an extra component $A_{i+1,n_{i+1}^0+1} = 0$, and $\vec{A}_{\vec{n}^0_{\mathrm{b},i,j}}$ differs from $\vec{X}_{\vec{n}^0}$ in that its j^{th} component in the i^{th} generation is $Y_{i,j}$ while its $(i+1)^{\text{th}}$ generation has an extra component Y_{i+1,n_{i+1}^0+1}. $\vec{A}_{\vec{n}_{\mathrm{b},-i,-j}}(t)$ differs from $\vec{A}_{\vec{n}}(t)$ in that its $(i+1)^{\text{th}}$ generation does not have the $(n_{i+1})^{\text{th}}$ component, while $\vec{X}_{\vec{n}_{\mathrm{b},-i,-j}}(t)$ differs from $\vec{A}_{\vec{n}}(t)$ in that its j^{th} component in the i^{th} generation is Y while it does not have the $(n_{i+1})^{\text{th}}$ component in the $(i+1)^{\text{th}}$ generation, respectively. Then similar to the proof of

Proposition 2 as shown in Appendix A.3.2, $p_{\vec{n}}^{m+1}$ satisfies the following PDE

$$\begin{aligned}
&\frac{\partial p_{\vec{n}}^{m+1}}{\partial t}(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t) + \sum_{i=1}^{k}\sum_{j=1}^{n_i} \frac{\partial (g_{i,j}(A_{i,j}, X_{i,j}, t) p_{\vec{n}}^{m+1})}{\partial X_{i,j}} - \frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{n_i} \frac{\partial^2 (\sigma_{i,j}^2(A_{i,j}, X_{i,j}, t) p_{\vec{n}}^{m+1})}{(\partial X_{i,j})^2} \\
&\quad + \sum_{i=1}^{k}\sum_{j=1}^{n_i^s} \frac{\partial p_{\vec{n}}^{m+1}}{\partial A_{i,j}} = -\sum_{i=1}^{k}\sum_{j=1}^{n_i} \big(\beta_{i,j}(A_{i,j}, X_{i,j}) + \mu_{i,j}(A_{i,j}, X_{i,j})\big) p_{\vec{n}}^{m+1} \\
&\qquad + \sum_{i=1}^{\infty}\sum_{j=1}^{n_i^{\mathrm{d}}} \int \mu(Y, A) p_{\vec{n}_{\mathrm{d},i}}^{m}(\vec{A}_{\vec{n}_{\mathrm{d},i,j}}, \vec{X}_{\vec{n}_{\mathrm{d},i,j}}, t | \vec{A}_{\vec{n}^0}(0), \vec{X}_{\vec{n}^0}(0), \vec{n}^0, 0) \mathrm{d}Y \mathrm{d}A \text{ , if } \vec{A}_{\vec{n}} > 0 \\
&p_{\vec{n}}^{m+1}(\vec{X}_{\vec{n}}, \vec{A}_{\vec{n}}, t | \vec{A}_{\vec{n}^0}(0), \vec{X}_{\vec{n}^0}(0), 0) = \\
&\quad \int \sum_{i=1}^{k}\sum_{j=1}^{n_i} \tilde{\beta}_{i,j}(Y, A_{i,j}, X_{i+1,n_{i+1}-1}, X_{i+1,n_{i+1}}) \\
&\qquad \times p_{\vec{n}}^{m}(\vec{A}_{\vec{n}_{\mathrm{b},-i,-j}^0}(t), \vec{X}_{\vec{n}_{\mathrm{d},-i,-j}^0}(t), t | \vec{A}_{\vec{n}^0}(0), \vec{X}_{\vec{n}^0}(0), 0) \mathrm{d}X_{i,j} \mathrm{d}A_{i,j}, \\
&\qquad \text{if } A_{i+1,n_{i+1}} = 0.
\end{aligned} \tag{A.3.20}$$

Likewise, it could be shown that $p_{\vec{n}}^{m}$ is non-negetive, increasing in m, and

$$\sum_{\vec{n}} \int p_{\vec{n}}^{m}(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t | \vec{X}_{\vec{n}^0}(0), \vec{A}_{\vec{n}^0}(0), 0) \mathrm{d}\vec{X}_{\vec{n}} \mathrm{d}\vec{A}_{\vec{n}} \leq 1. \tag{A.3.21}$$

Therefore, there exists a limit $p^*_{\vec{n}} = \lim_{d\to\infty} p^m_{\vec{n}}$ which satisfies the PDE

$$\begin{aligned}
&\frac{\partial p^*_{\vec{n}}}{\partial t}(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t) + \sum_{i=1}^{k}\sum_{j=1}^{n_i^s} \frac{\partial (g_{i,j}(A_{i,j}, X_{i,j}, s)p^*_{\vec{n}})}{\partial X_{i,j}} - \frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{n_i} \frac{\partial^2(\sigma^2_{i,j}(A_{i,j}, X_{i,j}, t)p^*_{\vec{n}})}{(\partial X_{i,j})^2} \\
&+ \sum_{i=1}^{k}\sum_{j=1}^{n_i^s} \frac{\partial p^*_{\vec{n}}}{\partial A_{i,j}} = -\sum_{i=1}^{k}\sum_{j=1}^{n_i} \big(\beta_{i,j}(A_{i,j}, X_{i,j}) + \mu_{i,j}(A_{i,j}, X_{i,j})\big)p^*_{\vec{n}} \\
&+ \sum_{i=1}^{\infty}\sum_{j=1}^{n_i^{\mathrm{d}}} \int \mu_{i,j}(A, Y)p^*_{\vec{n}_{\mathrm{d},i}}(\vec{A}_{\vec{n}_{\mathrm{d},i,j}}, \vec{X}_{\vec{n}_{\mathrm{d},i,j}}, t|\vec{A}_{\vec{n}^0}(0), \vec{X}_{\vec{n}^0}(0), 0)\mathrm{d}Y\,\mathrm{d}A \text{ , if } \vec{A}_{\vec{n}} > 0 \qquad \text{(A.3.22)} \\
&p^*_{\vec{n}}(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t|\vec{X}_{\vec{n}^0}(0), \vec{A}_{\vec{n}^0}(0), 0) = \\
&\int \sum_{i=1}^{k}\sum_{j=1}^{n_i} \tilde{\beta}_{i,j}(A_{i,j}, Y, X_{i+1,n_{i+1}-1}, X_{i+1,n_{i+1}}) \\
&\qquad \times p^*_{\vec{n}}(\vec{A}_{\vec{n}^0_{\mathrm{d},-i,-j}}(t), \vec{X}_{\vec{n}^0_{\mathrm{d},-i,-j}}(t), t|\vec{A}_{\vec{n}^0}(0), \vec{X}_{\vec{n}^0}(0), 0)\mathrm{d}X_{i,j}\mathrm{d}A_{i,j}, \text{ if } A_{i+1,n_{i+1}} = 0.
\end{aligned}$$

We can also define the unconditional probability density by averaging the initial probability density $p^{\mathrm{i}}_{\vec{n}^0}$

$$p^*_{\vec{n}}(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t) \coloneqq \sum_{\vec{n}^0} \int p_{\vec{n}}(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t|\vec{A}_{\vec{n}^0}(0), \vec{X}_{\vec{n}^0}(0), 0)p^{\mathrm{i}}_{\vec{n}^0}(\vec{A}_{\vec{n}^0}, \vec{X}_{\vec{n}^0}, 0)\mathrm{d}\vec{X}_{\vec{n}^0}\mathrm{d}\vec{A}_{\vec{n}^0}. \qquad \text{(A.3.23)}$$

From Eq. (A.3.23), we can define the symmetric probability density distribution

$$\rho_{\vec{n}}(\vec{A}_{\vec{n}}, \vec{X}_{\vec{n}}, t) \coloneqq \Pi_{i=1}^{k}\frac{1}{n_i!}\sum_{\pi} p^*_{\vec{n}}(\pi(\vec{A}_{\vec{n}}), \pi(\vec{X}_{\vec{n}}), t), \qquad \text{(A.3.24)}$$

from which we could derive the macroscopic quantities such as the marginalized cell density. We shall omit detailed discussions on equations satisfied by these macroscopic quantities here for brevity.

A.4 Appendix for Chapter 9

A.4.1 Numerical scheme

To numerically solve Eqs. (9.2.10) and (9.2.11), we use a uniform discretization $\tau_k = k\Delta\tau, k = 0, 1, \ldots, K$. A backward difference operator $[I(\tau_k, t) - I(\tau_{k-1}, t)]/(\Delta\tau)$ is used to approximate $\partial_\tau I(\tau, t)$ and a predictor-corrector Euler scheme is used to advance time [PTV07]. Setting the cut-offs $I(-\Delta\tau, t) \equiv 0$ and $I(K\Delta\tau, t) \equiv 0$, the resulting discretized equations for the full SIR model are

$$\begin{aligned}
S(t+\Delta t) =& S(t) - \Delta t S(t) \sum_{k=0}^{K} \beta(\tau_k, t) I(\tau_k, t) \Delta\tau, \\
\tilde{I}(\tau_k, t) =& I(\tau_k, t) - \Delta t \frac{I(\tau_k, t) - I(\tau_{k-1}, t)}{\Delta\tau} - \Delta t(\gamma(\tau_k, t) + \mu(\tau_k, t)) I(\tau_k, t), \\
I(\tau_k, t+\Delta t) =& \tilde{I}(\tau_k, t) - \frac{\Delta t}{2} \left[\frac{I(\tau_k, t) - I(\tau_{k-1}, t)}{\Delta\tau} + (\gamma(\tau_k, t) + \mu(\tau_k, t)) I(\tau_k, t) \right. \\
& \left. + \frac{\tilde{I}(\tau_k, t) - \tilde{I}(\tau_{k-1}, t)}{\Delta\tau} + (\gamma(\tau_k, t+\Delta t) + \mu(\tau_k, t+\Delta t)) \tilde{I}(\tau_k, t) \right] \\
& + \delta_{k,0} \frac{\Delta t}{\Delta\tau} S(t) \sum_{j=0}^{K} \beta(\tau_j, t) I(\tau_j, t) \Delta\tau,
\end{aligned} \tag{A.4.1}$$

where $\tilde{I}$ is the initial predicted guess, and the last term proportional to $\delta_{k,0}$ encodes the boundary condition Eq. (9.2.11). Note that we use $\sum_{k=0}^{K} \beta(\tau_k, t) I(\tau_k, t) \Delta\tau$ to indicate the numerical evaluation of $\int_0^\infty \mathrm{d}\tau' \beta(\tau', t) I(\tau', t)$. Quadrature methods such as the Simpson's rule and the trapezoidal rule can be used to approximate the integral more efficiently.

The total number of dead, recovered, and infected individuals at the time t are found by

$$
\begin{aligned}
D^0(m\Delta t) &= \frac{1}{2}\sum_{j=0}^{m}\sum_{k=0}^{K} c(k\Delta\tau, j\Delta t)\left[I(k\Delta\tau, j\Delta t) + \tilde{I}(k\Delta\tau, j\Delta t)\right]\Delta\tau\Delta t,\\
R^0(t) &= \frac{1}{2}\sum_{j=0}^{m}\sum_{k=0}^{K} \mu(k\Delta\tau, j\Delta t)\left[I(j\Delta\tau, j\Delta t) + \tilde{I}(k\Delta\tau, j\Delta t)\right]\Delta\tau\Delta t,\\
I(m\Delta t) &= \sum_{k=0}^{K} I(k\Delta\tau, m\Delta t)\Delta\tau,
\end{aligned}
$$

with analogous expressions for $D^1(m\Delta t)$ and $R^1(m\Delta t)$. To obtain a stable integration scheme, the time steps Δt and $\Delta\tau$ have to satisfy $\Delta t/(2\Delta\tau) < 1$. In all of our numerical computations, we thus set $\Delta t = 0.002, \Delta\tau = 0.02$, and $K = 10^4$. In the next section, we show additional plots of the magnitude of $I(\tau, t)$ in the $t - \tau$ plane.

A.4.2 Solutions for τ_1-averaged probabilities

Using the method of characteristics, we find the formal solution to Eq. (9.2.1):

$$
P(\tau, t|\tau_1) = \delta(\tau - t - \tau_1)e^{-\int_0^t (\mu(\tau - t + s, s|\tau_1) + \gamma(\tau - t + s, s|\tau_1))\mathrm{d}s}, \tag{A.4.2}
$$

which can be used to construct the death and cure probabilities

$$
\begin{aligned}
P_{\mathrm{d}}(t|\tau_1) &= \int_0^t \mathrm{d}t'\, \mu(\tau_1 + t', t')e^{-\int_0^{t'} (\mu(\tau_1 + s, s) + \gamma(\tau_1 + s, s))\mathrm{d}s}\\
P_{\mathrm{r}}(t|\tau_1) &= \int_0^t \mathrm{d}t'\, \gamma(\tau_1 + t', t')e^{-\int_0^{t'} (\mu(\tau_1 + s, s) + \gamma(\tau_1 + s, s))\mathrm{d}s}.
\end{aligned} \tag{A.4.3}
$$

If we now invoke the functional forms of μ and γ given in Eq. (9.2.4), we find explicitly

$$P_{\rm d}(\tau,t|\tau_1)=\begin{cases}\dfrac{\mu_1}{\mu_1+\gamma}\left(1-e^{-(\mu_1+\gamma)t}\right) & \tau>t+\tau_{inc}\\ 0 & \tau_{inc}\geq\tau>\tau_1\\ \dfrac{\mu_1 e^{-\gamma(\tau_{inc}-\tau_1)}}{\mu_1+\gamma}\left(1-e^{-(\mu_1+\gamma)(\tau-\tau_{\rm inc})}\right) & \tau>\tau_{inc}\geq\tau_1\end{cases} \tag{A.4.4}$$

and

$$P_{\rm r}(\tau,t|\tau_1)=\begin{cases}\dfrac{\gamma}{\mu_1+\gamma}\left(1-e^{-(\mu_1+\gamma)t}\right) & \tau>t+\tau_{inc}\\ 1-e^{-\gamma t} & \tau_{inc}\geq\tau>\tau_1\\ 1-e^{-\gamma(\tau_{inc}-\tau_1)}+\dfrac{\gamma e^{-\gamma(\tau_{inc}-\tau_1)}}{\mu_1+\gamma}\left(1-e^{-(\mu_1+\gamma)(\tau-\tau_{\rm inc})}\right) & \tau>\tau_{inc}\geq\tau_1.\end{cases} \tag{A.4.5}$$

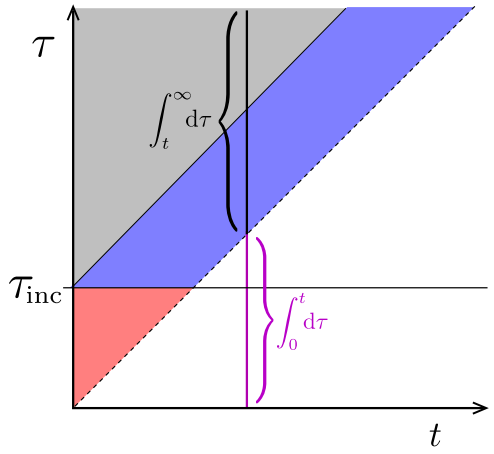

Figure A.1: **Phase plot for $P(\tau>t,t)$ and $I(\tau>t,t)$.** The regions delineate the different forms of the solution (Eq. (A.4.6)). Here, we have included an incubation time τ_{inc} before which no death occurs. The solution for $\bar{P}(\tau,t)$ or $I(\tau,t)$ in the $\tau<t$ region must be self-consistently solved using the boundary condition Eq. (9.2.11). At any fixed time, the integral of $I(\tau,t)$ over $t<\tau\leq\infty$ captures only the initial population, excludes newly infected individuals, and is used to compute $D^1(t), R^1(t)$, and $M_{\rm p}^1(t)$. To compute $D^0(t), R^0(t)$, and $M_{\rm p}^0(t)$, we integrate across all infected individuals (including the integral over $t>\tau\geq 0$ shown in magenta).

Finally, we can also find the τ_1-averaged probabilities for $\tau\geq t$ by weighting over $\rho(\tau_1;n,\lambda)$. For example,

$$\bar{P}(\tau,t) = \begin{cases} \rho(\tau - t; n, \lambda) e^{-(\mu_1+\gamma)t} & \tau \geq t + \tau_{inc} \\ \rho(\tau - t; n, \lambda) e^{-\gamma t} & \tau_{inc} \geq \tau > t \\ \rho(\tau - t; n, \lambda) e^{-\gamma t} e^{-\mu_1(\tau - \tau_{inc})} & t + \tau_{inc} \geq \tau > \tau_{inc} \end{cases} .$$

These solutions hold for the different regions shown in the phase plot of Fig. A.1 and are equivalent to those for $I(\tau > t, t)$. Corresponding expressions for $\bar{P}_{\rm d}(t)$ and $\bar{P}_{\rm r}(t)$ can be found and used to construct $M^1_{\rm p}(t)$. Fig. A.2(a) shows the magnitude of $I(\tau, t)$ in the

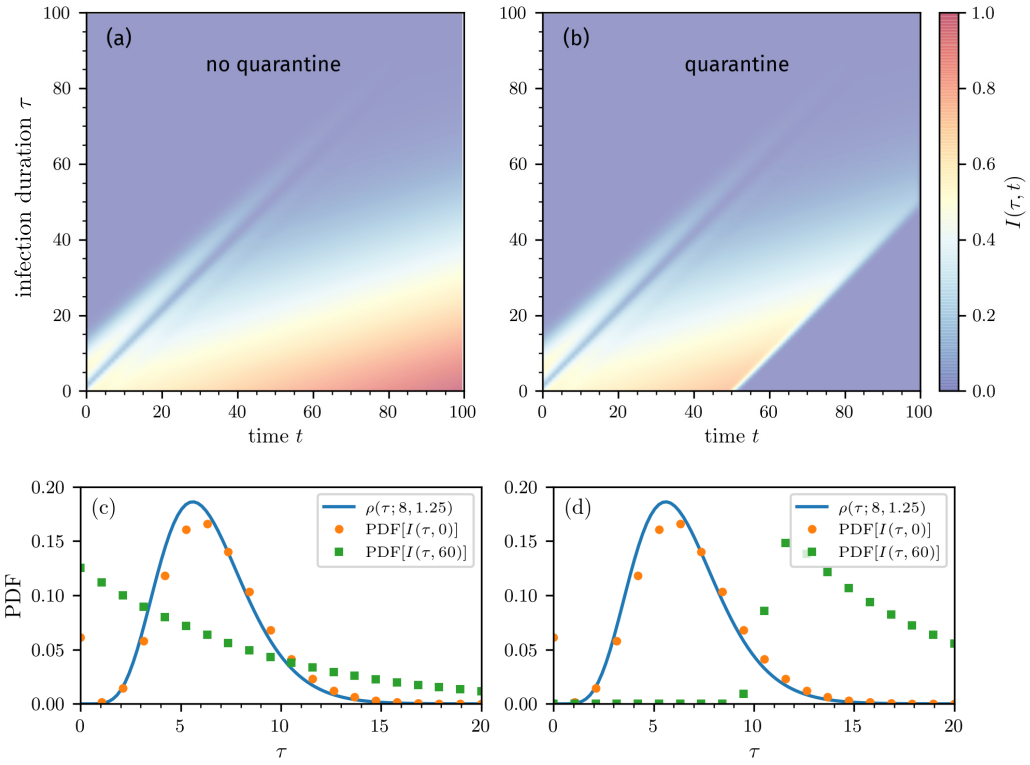

Figure A.2: **Density plots of $I(\tau, t)$ in the $t - \tau$ plane.** Numerical solution of the equation for $I(\tau, t)$ in Eqs. (9.2.10) under the assumption of a fixed susceptible size and $\beta_0 S_0 = 4.64$/day. (a) The density without quarantine monotonically grows with time t in the region $\tau < t$ as an unlimited number of susceptibles continually produces infections. (b) With quarantining after $t_{\rm q} = 50$ days, we set $\beta_0 S_0 = 0$ for $t > t_{\rm q}$, which shuts off new infections. Both plots were generated using the same initial density $\rho(\tau_1)$ defined in Eq. (9.2.7). In both cases, the density $I(\tau > t)$ is identical to $P(\tau > t)$ if the same $\rho(\tau_1)$ is used and is independent of disease transmission, susceptible dynamics, etc. (c-d) Probability-density functions (PDFs) of the number of infected individuals $I(\tau, t)$ for $t = 0, 60$ days (b) without and (c) with quarantine. The blue solid line corresponds to the initial distribution $\rho(\tau; n = 8, \lambda = 1.25)$ (see Eq. (9.2.7)).

$t - \tau$ plane when we use Eq. (9.2.16), set $S(t) = S_0$ constant (so that the first equation

in Eq. (A.4.1) does not apply) and assign $\beta_0 S_0 = 4.64$/day. In this case, the epidemic continues to grow in time, but the mortality rates $M_{\rm p}^{0,1}(t)$ nonetheless converge as $t \to \infty$. In Fig. A.2(b), we set $\beta_0 S_0 = 0$ for $t > t_{\rm q}$ to model strict quarantining after $t_{\rm q} = 50$ days. We observe no new infection after the onset of strict quarantine measures. In both cases (quarantine and no quarantine), we use $\rho(\tau; n = 8, \lambda = 1.25)$ (see Eq. (9.2.7) in the main text) to describe the initial distribution of infection times τ. As time progresses, more of the distribution of τ moves towards smaller values until quarantine measures take effect (see Fig. A.2(c) and (d)).

A.5 Appendix for Chapter 10

A.5.1 Basic reproduction number

In this appendix, we analytically derive the basic reproduction number $\mathcal{R}_0$ for uncorrelated networks and compare the resulting values with those obtained using Eqs. (10.2.7) and (10.2.8). As a starting point, we note that the conditional degree distribution $P(\ell|k)$ can be expressed in terms of a symmetric (for undirected networks) joint degree distribution $P(\ell, k)$, the probability that a randomly chosen edge connects two nodes with degrees ℓ and k. Marginalizing $P(\ell, k)$ over ℓ yields the distribution over edge ends [WP07] $P_{\rm e}(\ell) \equiv \sum_k P(\ell, k) = \ell P(\ell)/\langle k \rangle$, where $\langle k \rangle = \sum_k k P(k)$ is the mean degree. The conditional degree distribution is related to the joint distribution via

$$P(\ell|k) = \frac{P(\ell, k)}{P_{\rm e}(k)} = \frac{\langle k \rangle P(\ell, k)}{k P(k)} = \frac{E_{\ell,k}}{k P(k) N}, \tag{A.5.1}$$

which can be further simplified in the uncorrelated network limit where $P(\ell, k) \approx P_{\rm e}(k) P_{\rm e}(\ell)$:

$$P(\ell|k) \approx \frac{\ell P(\ell)}{\langle k \rangle}. \tag{A.5.2}$$

Eqs. (A.5.1) or (A.5.2) can be used as a simpler replacement for $P(\ell|k)$ in Eqs. (10.2.2) and (10.2.3) if $E_{\ell,k}/(kN_k)$ is not directly accessible. For example, for an uncorrelated network

(*i.e.*, for $P(\ell|k) = \ell P(\ell)/\langle k\rangle$), we find

$$\frac{\mathrm{d}i_k^{\mathrm{u}}(t)}{\mathrm{d}t} = \beta^{\mathrm{u}} \frac{k s_k(t)}{\langle k\rangle} \sum_{\ell} \ell i_\ell^{\mathrm{u}}(t) - \gamma^{\mathrm{u}} i_k^{\mathrm{u}}(t), \tag{A.5.3}$$

where we have set testing rates $f_k(0) = 0$ at the start of the infection. According to [KGK06], we define

$$I^{\mathrm{u}}(t) \coloneqq \sum_k i_k^{\mathrm{u}}(t), \quad J^{\mathrm{u}}(t) \coloneqq \sum_k k i_k^{\mathrm{u}}(t) \tag{A.5.4}$$

and obtain

$$\begin{aligned} \frac{\mathrm{d}I^{\mathrm{u}}(t)}{\mathrm{d}t} &= \beta^{\mathrm{u}} J^{\mathrm{u}}(t) - \gamma^{\mathrm{u}} I^{\mathrm{u}}(t), \\ \frac{\mathrm{d}J^{\mathrm{u}}(t)}{\mathrm{d}t} &= \beta^{\mathrm{u}} \frac{\langle k^2\rangle}{\langle k\rangle} J(t) - \gamma^{\mathrm{u}} J^{\mathrm{u}}(t). \end{aligned} \tag{A.5.5}$$

We perform a linear stability analysis around the disease-free state $(I^*, J^*) = (0, 0)$ and find the eigenvalues to Eqs. (A.5.5):

$$\lambda_\pm = -\gamma^{\mathrm{u}} \pm \beta^{\mathrm{u}} \frac{\langle k^2\rangle}{\langle k\rangle}. \tag{A.5.6}$$

The transition from negative to positive eigenvalues occurs for $-\gamma^{\mathrm{u}} + \beta^{\mathrm{u}}\langle k^2\rangle/\langle k\rangle = 0$. Hence, the basic reproduction number is

$$\mathcal{R}_0 = \frac{\beta^{\mathrm{u}}}{\gamma^{\mathrm{u}}} \frac{\langle k^2\rangle}{\langle k\rangle} = \frac{\beta^{\mathrm{u}}}{\gamma^{\mathrm{u}}} \left(\langle k\rangle + \frac{\mathrm{Var}[k^2]}{\langle k\rangle} \right). \tag{A.5.7}$$

If we use the conditional degree distribution $P(\ell|k) = (\ell - 1)P(\ell)/\langle k\rangle$ proposed by Kiss *et al.* [KGK06] to account for a reduction in neighboring susceptible vertices, the corresponding basic reproduction number is modified to

$$\mathcal{R}_0^{\mathrm{Kiss}} = \frac{\beta^{\mathrm{u}}}{\gamma^{\mathrm{u}}} \left(\langle k\rangle - 1 + \frac{\mathrm{Var}[k^2]}{\langle k\rangle} \right). \tag{A.5.8}$$

The mean degrees of the BA and SBM networks are 3.77 and 23.14, and the variances for the BA and SBM networks are 20.40 and 36.62, respectively. Using the values $\gamma^{\mathrm{u}} = 14^{-1}/\mathrm{d}ay$

and $\beta^{\mathrm{u}} = 0.0411/\mathrm{day}$ for the BA network, we find that the basic reproduction numbers $\mathcal{R}_0 = 5.361$ and $\mathcal{R}_0^{\mathrm{Kiss}} = 4.777$ are larger than 4.5, the value we used to determine β^{u} according to the next-generation matrix method (Eqs. (10.2.7) and (10.2.8)). The observed approximation errors in Eqs. (A.5.7) and (A.5.8) are a consequence of the assumption that the underlying network is uncorrelated. For the SBM network, we find $\mathcal{R}_0 = 4.499$ and $\mathcal{R}_0^{\mathrm{Kiss}} = 4.317$, close to the 4.5 value used to find $\beta^{\mathrm{u}} = 0.0130$ using Eqs. (10.2.7) and (10.2.8).

To summarize, our comparison shows that in the SBM model where the degrees of neighbors are uncorrelated, Eqs. (A.5.7) and (A.5.8) give close approximations of the actual reproduction number calculated from the next-generation matrix method (10.2.7). For the BA network, degree correlations make Eqs. (A.5.7) and (A.5.8) overestimate the actual reproduction number. Therefore, we recommend using the next-generation matrix method to numerically determine the basic reproduction number unless degree correlations are weak and Eqs. (A.5.7) and (A.5.8) can provide accurate estimates of $\mathcal{R}_0$.

A.5.2 Optimal testing and vaccination algorithms

Below, we explicitly give the pseudo-code for the testing and quarantine model based on Pontryagin's maximum principle.

A.5.3 Reinforcement-learning strategy

To identify effective testing and vaccination strategies, we also investigated reinforcement-learning (RL) approaches. RL explores the space of all possible actions and directly optimizes the loss functions for testing and vaccination defined in Eqs. (10.3.3) and (10.4.6). Here, we use an RL approach with experience replay to learn both the optimal testing strategy in Eqs. (10.2.2)–(10.2.5) and the optimal vaccination strategy in Eqs. (10.4.1)–(10.4.3).

Typically, applying a policy-gradient method to a continuous action space will usually yield poor results due to the inability of such methods to explore the whole space. However,

Algorithm 7 Pseudo-code for determining optimal testing strategies based on Pontryagin's maximum principle.

Initialize $t = 0$, $s_k(0), i_k^{\mathrm{u}}(0), i_k^*(0)$, Δt, $T = n\Delta t$, $\beta^{\mathrm{u}}, \beta^*$, $\gamma^{\mathrm{u}}, \gamma^*$, δ, initial strategy $F(k\Delta t), k$, $f_{\max}$, $f_{\min}$, ϵ, $iter_{\max}$
for $k = 0 : n-1$ **do**
    Calculate $s_k(t), i_k^*(t), i_k^{\mathrm{u}}(t)$ under the strategy $F(k\Delta t)$ from Eqs. (10.2.2)–(10.2.4)
end for
Set $\lambda_k^s, \lambda_k^{\mathrm{u}}, \lambda_k^* = 0, k = n$
Calculate the loss function L_1 in Eq. (10.3.3)
for $k = n-1 : 0$ **do**
    Calculate $\lambda_k^s, \lambda_k^{\mathrm{u}}, \lambda_k^*$ under the strategy $F(k\Delta t)$ from Eqs. (10.3.5)–(10.3.7)
end for
for $k = 0 : n-1$ **do**
    First renew the strategy $F(k\Delta t)$, then calculate $s_k, i_k^{\mathrm{u}}, i_k^*$ under the strategy $F(k\Delta t)$ from Eqs. (10.2.2)–(10.2.4)
end for
Calculate the loss function L_2 in Eq. (10.3.3)
$i \leftarrow 1$
while $|L_1 - L_2| > \epsilon$ && $i < iter_{\max}$ **do**
    $i \leftarrow i + 1$
    $L_1 \leftarrow L_2$
    Set $k = n, \lambda_k^s, \lambda_k^{\mathrm{u}}, \lambda_k^* = 0$
    for $k = n-1 : 0$ **do**
        Calculate $\lambda_k^s, \lambda_k^{\mathrm{u}}, \lambda_k^*$ under the strategy $F(k\Delta t)$ from Eqs. (10.3.5)–(10.3.7)
    end for
    for $k = 0 : n-1$ **do**
        First renew the strategy $F(k\Delta t)$, then calculate $s_k, i_k^{\mathrm{u}}, i_k^*$ under the strategy $F(k\Delta t)$ from Eqs. (10.2.2)–(10.2.4)
    end for
    Calculate the Loss function L_2 in Eq. (10.3.3)
end while

using our previous results based on PMP, we know that the optimal strategy is always obtained by maximizing the testing and vaccination rates for subpopulations presumed to be at a higher risk.

Therefore, we do not need to explore the whole space of all possible actions. Instead, from Eqs. (10.3.8), (10.4.7), we can restrict our strategy space to the extreme points[1] of the set

$$\{(f_k)|_{k=1}^K | \sum_{k=1}^K f_k = F(t), f_{\min} \leq \frac{f_k}{N_k} \leq f_{\max}\} \tag{A.5.9}$$

for determining the testing-resource allocation and the extreme points of the set

$$\{(v_k)|_{k=1}^K | \sum_{k=1}^K v_k = V(t), v_{\min} \leq \frac{v_k}{N s_k(t)} \leq v_{\max}\} \tag{A.5.10}$$

for determining vaccination resource allocation at each step. The set of extreme points represents all strategies that maximize the testing/vaccination rates for some groups and minimize them for other groups. Such strategies also cannot be written as nontrivial convex combinations of other strategies. By confining ourselves to extreme points, the possible action space is reduced to a finite set on which we perform RL.

Since the curse of dimensionality increases the number of all possible strategies exponentially with K, we further restrict our RL approach to networks with degree cutoff $K = 20$. This additional constraint allows us to perform RL with a computation time of about 30 days for the testing model on the BA network, 3 days for the testing model on the SBM network, 6 hours for the vaccination model on the BA network, and 2 hours for the vaccination model on an SBM network. All computations are performed using Python 3.8.10 on a laptop with a 4-core Intel(R) Core(TM) i7-8550U CPU @ 1.80 GHz.

To identify effective testing and vaccination strategies, we use the reward functions

[1]Extreme points are points in a set that cannot be written as a nontrivial convex linear combination of any other points in the same set.

Algorithm 8 Pseudo-code of Q-Learning in testing resource allocation.

Initialize $F, \delta, C,\ i_k^u(0), i_k^*(0), \beta^u, \beta^*, \gamma^u, \gamma^*, M, \epsilon$
Initialize replay memory D
Randomly initialize the hyperparameter set $\Theta^- \leftarrow \Theta$ for evaluating the action value function $Q^*(\mathcal{S}, \mathcal{A}; \Theta)$
for episode $\ell = 1 : M$ **do**
    Initialize $\mathcal{S}_0$
    for $t = 0 : T_{\max} - 1$ **do**
        With probability ϵ, randomly select an action a_i
        otherwise select $\mathcal{A}_t = \text{argmax}_{\mathcal{A}} Q(\mathcal{S}_t, \mathcal{A}; \Theta)$
        Execute action $\mathcal{A}_t$ and observe reward R_t and state $\mathcal{S}_{t+1}$
        Store transition $(\mathcal{S}_t, \mathcal{A}_t, R_t, \mathcal{S}_{t+1})$ in D
        Sample random minibatch of transitions $(\mathcal{S}_j, \mathcal{A}_j, R_j, \mathcal{S}_{j+1})$ from D
        if $j = T_{\max} - 1$ **then**
            Set $y_j = R_j$
        else
            Set $y_j = R_j + \delta \max_{\mathcal{A}'} \hat{Q}(\mathcal{S}_{j+1}, \mathcal{A}'; \Theta^-)$
            Perform a gradient descent step on the minibatch $\sum_j [y_j - Q(\mathcal{S}_j, \mathcal{A}_j; \Theta)]^2$ with respect to the network hyperparameter set Θ
        end if
    end for
    Every C steps reset $\Theta^- \leftarrow \Theta$
end for

(10.3.3) and (10.4.6). We define the reward at time $t_i = i\Delta t$ as

$$R(\mathcal{S}_i, \mathcal{A}_i, i) = \sum_{k=1}^{K} \left[s_k(t_{i+1}) - s_k(t_i) \right], \tag{A.5.11}$$

the "negative" of the number of total infections during the time period $[t_i, t_{i+1})$. Here, the state $\mathcal{S}_i$ and action $\mathcal{A}_i$ are

$$\begin{aligned} \mathcal{S}_i =& (s_1(t_i), \ldots, s_K(t_i), i_1^{\mathrm{u}}(t_i), \ldots, i_K^{\mathrm{u}}(t_i), \\ & i_1^*(t_i), \ldots, i_K^*(t_i)) \in \mathbb{R}^{3K}, \\ \mathcal{A}_i =& (f_1(t_i), \ldots, f_K(t_i)) \in \mathbb{R}^K \end{aligned} \tag{A.5.12}$$

for the testing model Eqs. (10.2.2)–(10.2.5) and

$$\begin{aligned} \mathcal{S}_i &= (s_1(t_i), \ldots, s_K(t_i), i_1(t_i), \ldots, i_K(t_i)) \in \mathbb{R}^{2K}, \\ \mathcal{A}_i &= (v_1(t_i), \ldots, v_K(t_i)) \in \mathbb{R}^K \end{aligned} \tag{A.5.13}$$

for the vaccination model Eqs. (10.4.1)–(10.4.3). We recursively define the state-value function under a certain policy π to be

$$V^{\pi}(\mathcal{S}_i, i) = \begin{cases} V^{\pi}(\mathcal{S}_{i+1})\delta + R(\mathcal{S}_i, \pi(\mathcal{S}_i)), & t_i < T_{\max}, \\ 0, & t_i = T_{\max}, \end{cases} \tag{A.5.14}$$

where $\pi(\mathcal{S}_i)$ is the action determined under policy π given $\mathcal{S}_i$ and $\delta \in (0, 1]$ is a discount factor. We also define the action-value function to be

$$Q^{\pi}(\mathcal{S}_i, \mathcal{A}_i, i) = \begin{cases} V^{\pi}(\mathcal{S}_{i+1})\delta + R(\mathcal{S}_i, \mathcal{A}_i, i), & t_i < T_{\max} - 1, \\ R(\mathcal{S}_i, \mathcal{A}_i, i), & t_i = T_{\max} - 1. \end{cases} \tag{A.5.15}$$

We use Q^* and V^* to denote the action-value and state-value functions, respectively, under the best policy and apply the deep Q-learning algorithm, which has been used to find the

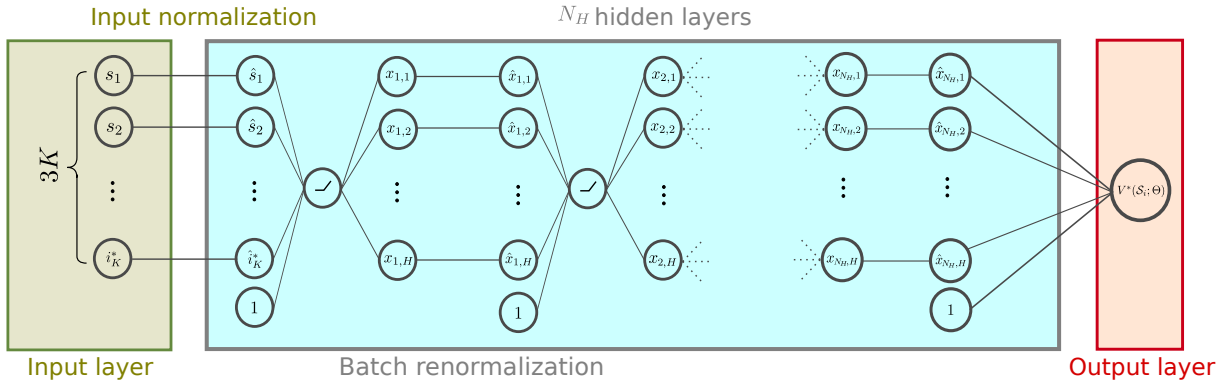

Figure A.3: Illustration of the neural network used to identify effective testing and vaccination strategies. The inputs of the input layer are $(s_1(t_i), \ldots, s_K(t_i), i_1^{\mathrm{u}}(t_i), \ldots, i_K^{\mathrm{u}}(t_i), i_1^*(t_i), \ldots, i_K^*(t_i)) \in \mathbb{R}^{3K}$. For each hidden layer i $(1 \leq i \leq N_H)$, we normalize the corresponding outputs $x_{i,j}$ for all samples in a minibatch such that the resulting values $\hat{x}_{i,j}$ have zero mean and unit variance. These values are used as inputs to a rectified linear unit (ReLU) activation function in the next hidden layer. Neurons labeled 1 are bias terms. The output $V^*(\mathcal{S}_i; \Theta)$ is an estimate of the state-value function under the optimal policy (see Eq. (A.5.14)), where Θ denotes the set of hyperparameters.

RL strategies that can approximate optimal strategies of certain Atari 2600 games [MKS15]. Here, we use a neural network with a hyperparameter set Θ, representing neural-network weights and biases to estimate the action-value function under the best policy $Q^*(\mathcal{S}, \mathcal{A}; \Theta)$, which is improved over epochs by Alg. 8.

An illustration of the neural network, its layers, and activation functions, is shown in Fig. A.3. We use another neural network with a hyperparameter set Θ^- updated every $C = 4$ steps to match Θ. The neural network contains $N_H = 4$ hidden layers with $H = 30$ neurons in each layer. The input data is the state at the i^{th} step $\mathcal{S}_i$, and the output is $V^*(\mathcal{S}_i; \Theta)$, the prediction for the optimal state-value function generated by the neural network. In each layer, the batch normalization technique is used before a rectified linear unit (ReLU) function is applied as an activation function. We compare the optimal strategies based on the PMP approach from Alg. 7 with the RL strategies that are based on Alg. 8. We set $T = 100$ and $\Delta t = 1$ so that the strategy is updated every day. Here, we use $f_{\min} = 0.002/\text{day}, f_{\max} = 0.4/\text{day}$. We use Eq. (10.2.7) with $\gamma^{\mathrm{u}} = (1/14)/\text{day}$ to calculate $\beta^{\mathrm{u}} = 0.0703/\text{day}$ for the $K = 20$ BA network and $\beta^{\mathrm{u}} = 0.0632/\text{day}$ for the $K = 20$ SBM network. Both PMP and RL strategies are also compared to the uniform testing strategy

(10.3.11). For RL, we train the underlying neural network for $M = 100$ epochs using Alg. 8.

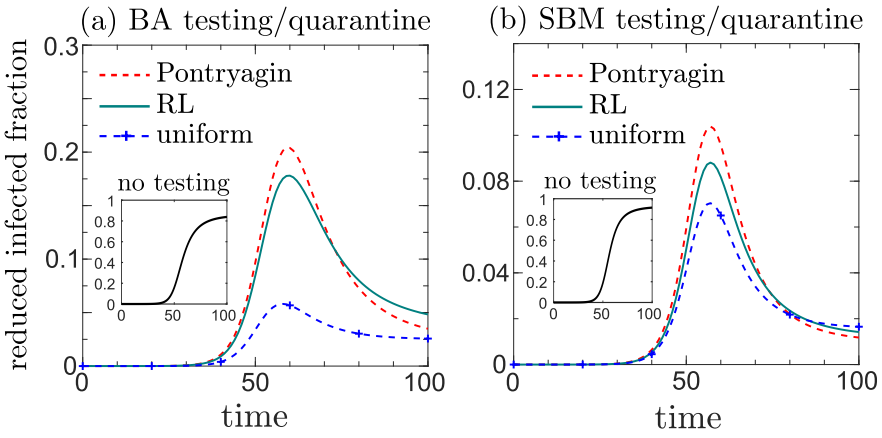

Figure A.4: Reduction in fractions of infected individuals calculated as the difference between the fractions infected obtained with testing and without testing for the BA network is shown in (a) and for the SBM network is shown in (b). The optimal control approach based on PMP reduces early infections the most. RL outperforms uniform testing in reducing the number of early-stage infections. Additionally, the effect of the optimal strategy is more striking in the BA network because it has a more heterogeneous node degree distribution.

Figure A.4 shows the differences between the infected fractions in simulations with and without testing. The PMP-based optimal control reduces early infections the most for both BA and SBM networks. Early infections contribute more to the loss function (10.3.3) since we set the discount factor to $\delta = 0.95$. We also observe that RL-based testing strategies outperform uniform testing in reducing early-stage infections. Comparing Fig. A.4(a,b), the effect of the optimal vaccination strategy in the BA network is more pronounced than that in the SBM network. In the BA network, node degrees are more heterogeneous and most nodes have small degrees, indicating that epidemic spreading can be controlled effectively as long as the few high-degree nodes are monitored and tested. Finally, comparing the result of the optimal-control approach in Fig. A.4 with Fig. 10.2, we observe that with a smaller K in the SBM network, the effect of the optimal vaccination strategy is less apparent because node degrees are more homogeneous.

Next, we compared the PMP approach with the RL approach for the optimal vaccination

strategy model Eqs. (10.4.1)–(10.4.3). Here, we set $v_{\min} = 0.0001, v_{\max} = 1$. For both

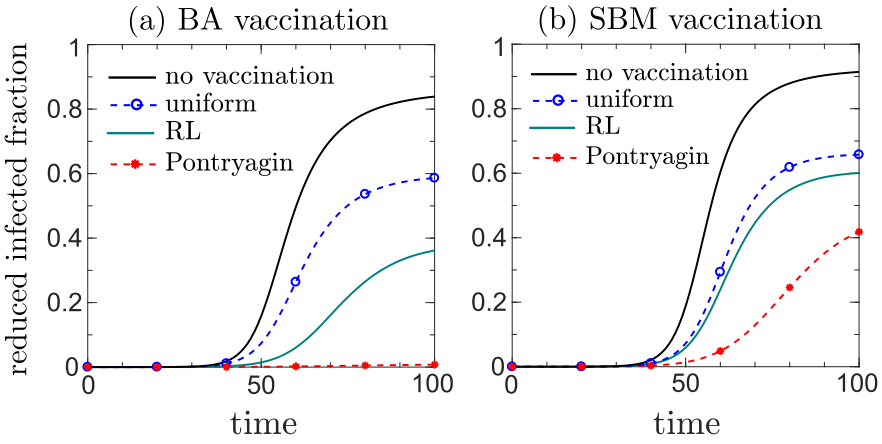

Figure A.5: Reduction in fractions of infected individuals calculated as the difference between the fractions infected obtained with vaccination and without vaccination for the BA network is shown in (a) and the SBM network is shown in (b). The optimal control approach using PMP can most effectively reduce infections for both networks and successfully suppress the spreading of the disease in the BA network. On the other hand, although not as good as the PMP-optimal strategies, the strategies obtained by the RL algorithm Alg. 8 can obviously reduce infections compared to the uniform vaccination rate strategy. As with testing, we observe that the effect of optimal vaccination is more pronounced in the BA network than in the SBM network.

networks, the optimal vaccination strategy obtained using PMP can most effectively reduce the initial infections because early infections have a higher weight in the loss function (10.4.6). Reinforcement-learning-based vaccination policies can also reduce initial infections, but the reduction is less than that of the PMP approach. Comparing Fig. A.5(a,b), we again observe that the effect of the optimal vaccination strategy for the BA network is more pronounced than that of the SBM network because the BA network has a more heterogeneous degree and is dominated by small-degree nodes.

To summarize, the controls derived from PMP are more effective than those based on RL. One limitation of RL-based interventions is that the possible action space that needs to be explored is usually large. However, based on our PMP results, we can constrain the action space before the learning process. Such PMP-informed constraints allow us to explore just the extreme points of the whole action space and thus make the training more efficient. Yet, the total number of possible actions grows exponentially with the maximal degree K and

the strategy obtained by the RL approach will probably be only locally optimal, violating the PMP condition and thus underperforming PMP. Nonetheless, RL could be useful if a procedure for computing an explicit solution cannot be formulated.

A.5.4 Simulations of corresponding stochastic models

We impose the optimal testing and vaccination strategies derived from applying PMP to the ODE system Eqs. (10.2.2)–(10.2.5) and Eqs. (10.4.1)–(10.4.3) on a simple discrete stochastic model and compare the resulting total infections. The corresponding optimal testing or vaccination is implemented by probabilistically testing or vaccinating each selected subpopulation. For example, in the testing model, we can employ a rejection-free event-based Monte-Carlo (MC) algorithm [Gil77] that implements a testing strategy.

For initial conditions, we randomly choose two nodes with a degree $k = 10$ to be infected. Correspondingly, for the deterministic ODE models, we set $s_k(0) = p(k) - \frac{2}{N}\mathbb{1}_{k,10}, i_k^{\mathrm{u}}(0) = \frac{2}{N}\mathbb{1}_{k,10}, i_k^*(0) = 0$ for the testing model and $s_k(0) = p(k) - \frac{2}{N}\mathbb{1}_{k,10}, i_k(0) = \frac{2}{N}\mathbb{1}_{k,10}$ for the vaccination model. We set the recovery rates $\gamma = \gamma^{\mathrm{u}} = \gamma^* = (14)^{-1}/\mathrm{day}$ and use the same reproduction number $\mathcal{R}_0(\beta^{\mathrm{u}}) = 4.5$ to calculate the unconstrained infection rates for the two networks from Eqs. (10.2.7). The loss functions defined for the testing and vaccination models in Eqs. (10.3.3) and Eqs.(10.4.6) are plotted below.

From Fig. A.6, the deterministic ODE models tend to overestimate the loss functions since all subpopulations are well mixed by the conditional degree distribution function $P(\ell|k)$ and therefore a single infected node could have an impact on the whole system. This difference arises because in a fully discrete realization of a BA or SBM network, each node can be in only one of three or four states and the disease may never arrive at certain critical nodes, significantly delaying its spread and allowing the overall infection to dissipate before ever reaching portions of the network. In contrast, the mass-action ODE model allows all nodes to be partially infected, allowing continuous transmission of the disease. Therefore, more network measures may be needed to accurately quantify the dynamics of disease spread

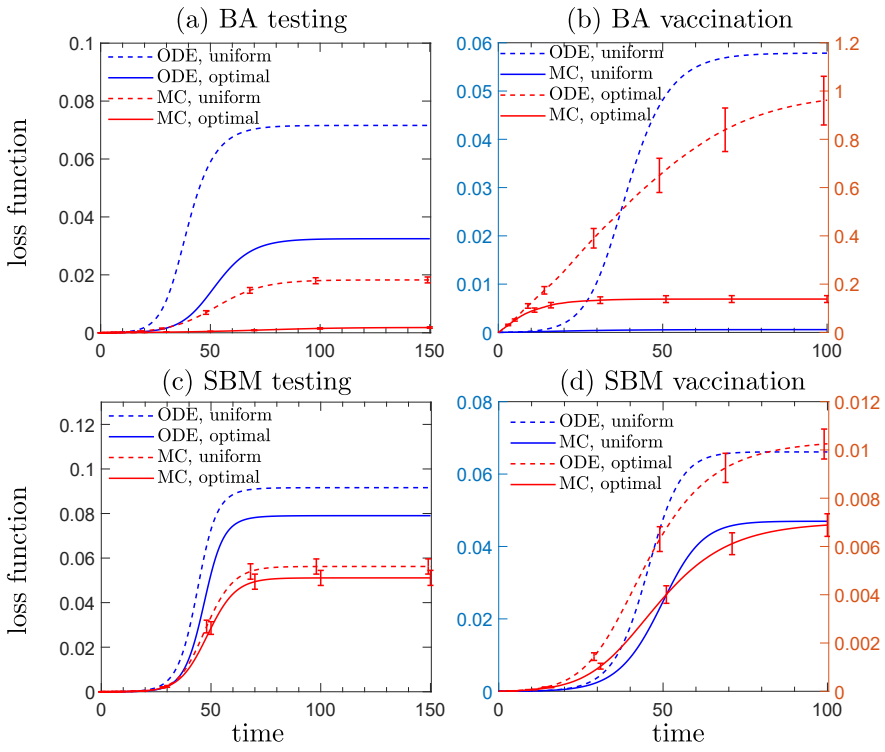

Figure A.6: Loss functions associated with the deterministic ODE models Eqs. (10.2.2)–(10.2.5) and (10.4.1)–(10.4.3), and the corresponding stochastic models. We apply PMP-based (solid lines) and uniform (dashed lines) testing and vaccination protocols. Panels (a) and (b) show the loss functions (10.3.3) and (10.4.6) associated with testing and vaccination interventions in a BA network. Results from the ODE models are shown in blue while the loss functions derived from the simulated stochastic model are shown in red. Panels (c) and (d) show loss functions for the testing and vaccination models in the SBM network. Note the different scales for the ODE (blue, left) and the MC (red, right) results. The loss functions of the discrete stochastic models are obtained by averaging over 100 trajectories with the standard error of the mean (standard deviation of means divided by $\sqrt{N}$) indicated by the error bars. For both networks, the deterministic ODE models yield larger losses than those obtained from averaging MC trajectories. For both deterministic ODEs and stochastic systems, the loss functions during optimal testing and vaccination are much smaller than when testing and vaccination are uniformly applied.

across discrete agent-based network models. Higher-order interactions beyond the pairwise conditional degree distribution [IPB19, BCI20, LXP22] could help explain the discrepancy between deterministic ODE and stochastic models and in estimating optimal policies in the fully stochastic context.

Nonetheless, Figure A.6 shows that the PMP-based interventions that we derived in the main text are also more effective than uniform testing and vaccination strategies in the stochastic agent-based model. This loss function reduction arises for both the BA and SBM networks. Thus, the optimal testing and vaccination strategies obtained from the deterministic model outperforms uniform testing and vaccination strategies even when applied on discrete stochastic network models, representing a reasonable starting point for approximating optimal strategies within agent-based discrete systems.

References

[20120] nCoV 2019 Data Working Group. "Epidemiological data from the nCoV-2019 outbreak: early descriptions from publicly available data." http://virological.org/t/epidemiological-data-from-the-ncov-2019-outbreak-early-descriptions-from-publicly-available-data/337, 2020. Accessed: 2020-02-26.

[AAB08] X. Antoine, A. Arnold, C. Besse, M. Ehrhardt, and A. Schädle. "A review of transparent and artificial boundary conditions techniques for linear and nonlinear Schrödinger equations." *Communications in Computational Physics*, **4**:729–796, 2008.

[AB02] R. Albert and A.-L. Barabási. "Statistical mechanics of complex networks." *Reviews of Modern Physics*, **74**(1):47, 2002.

[ABA22a] T. Asikis, L. Böttcher, and N. Antulov-Fantulin. "Neural Ordinary Differential Equation Control of Dynamics on Graphs." *Physical Review Research*, **4**:013221, 2022.

[ABA22b] Thomas Asikis, Lucas Böttcher, and Nino Antulov-Fantulin. "Neural ordinary differential equation control of dynamics on graphs." *Physical Review Research*, **4**(1):013221, 2022.

[ABM20] B. Abdalhamid, C. R. Bilder, E. L. McCutchen, S. H. Hinrichs, S. A. Koepsell, and P. C. Iwen. "Assessment of specimen pooling to conserve SARS-CoV-2 testing resources." *American Journal of Clinical Pathology*, **153**(6):715–718, 2020.

[ACH18] S. Arora, N. Cohen, and E. Hazan. "On the optimization of deep networks: Implicit acceleration by overparameterization." In *International Conference on Machine Learning*, pp. 244–253. PMLR, 2018.

[ACV19] P. Antonietti, C. Canuto, and M. Verani. "An adaptive hp–DG–FE method for elliptic problems: convergence and optimality in the 1D case." *Communications on Applied Mathematics and Computation*, **1**:309–331, 2019.

[ADB21] M.A. Acuna-Zegarra, S. Diaz-Infante, D. Baca-Carrasco, and D. Olmos-Liceaga. "COVID-19 optimal vaccination policies: A modeling study on efficacy, natural and vaccine-induced immunity responses." *Mathematical Biosciences*, **337**:108614, 2021.

[AK07] S. M. Aseev and A. V. Kryazhimskii. "The Pontryagin maximum principle and optimal economic growth problems." *Proceedings of the Steklov institute of mathematics*, **257**(1):1–255, 2007.

[AMR08] P. Auger, P. Magal, and S. Ruan. *Structured Population Models in Biology and Epidemiology*, volume 1936. Springer, 2008.

[APV20] A. N. Angelopoulos, R. Pathak, R. Varma, and M. I. Jordan. "On identifying and mitigating bias in the estimation of the COVID-19 case fatality rate." 2020.

[AWN15] K. E. Atkins, N. S. Wenzel, M. Ndeffo-Mbah, F. L. Altice, J. P. Townsend, and A. P. Galvani. "Under-reporting and case fatality estimates for emerging epidemics." *BMJ*, **350**:h1115, 2015.

[BA99] A.-L. Barabási and R. Albert. "Emergence of scaling in random networks." *Science*, **286**(5439):509–512, 1999.

[BA20] L. Böttcher and N. Antulov-Fantulin. "Unifying susceptible-infected-recovered processes on networks." *arXiv preprint arXiv:2002.11765*, 2020.

[BA22] L. Böttcher and T. Asikis. "Near-optimal control of dynamical systems with neural ordinary differential equations." *Machine Learning: Science and Technology*, **3**(4):045004, 2022.

[BAA22] L. Böttcher, N. Antulov-Fantulin, and T. Asikis. "AI Pontryagin or how artificial neural networks learn to control dynamical systems." *Nature communications*, **13**(1):333, 2022.

[BBD19] L. Bolzoni, E. Bonacini, R. Della Marca, and M. Groppi. "Optimal control of epidemic size and duration with limited resources." *Mathematical Biosciences*, **315**:108232, 2019.

[BBP04] M. Barthélemy, A. Barrat, R. Pastor-Satorras, and A. Vespignani. "Velocity and hierarchical spread of epidemic outbreaks in scale-free networks." *Physical Review Letters*, **92**(17):178701, 2004.

[BBP05] M. Barthélemy, A. Barrat, R. Pastor-Satorras, and A. Vespignani. "Dynamical patterns of epidemic outbreaks in complex heterogeneous networks." *Journal of Theoretical Biology*, **235**(2):275–288, 2005.

[BC19] N. Boussaid and A. Comech. *Nonlinear Dirac Equation: Spectral Stability of Solitary Waves*. American Mathematical Society, 2019.

[BCI20] F. Battiston, G. Cencetti, I. Iacopini, V. Latora, M. Lucas, A. Patania, J.-G. Young, and G. Petri. "Networks beyond pairwise interactions: structure and dynamics." *Physics Reports*, **874**:1–92, 2020.

[BDC21] L. Böttcher, M. R. D'Orsogna, and T. Chou. "Using excess deaths and testing statistics to determine COVID-19 mortalities." *European Journal of Epidemiology*, pp. 1–14, 2021.

[BDC22] L. Böttcher, M. R. D'Orsogna, and T. Chou. "A statistical model of COVID-19 testing in populations: effects of sampling bias and testing errors." *Philosophical Transactions of the Royal Society A*, **380**(2214):20210121, 2022.

[BDG19] E. Bernard, M. Doumic, and P. Gabriel. "Cyclic asymptotic behaviour of a population reproducing by fission into two equal parts." *Kinetic and Related Models*, **12**(3):551–571, 2019.

[BE22] H. Bararnia and M. Esmaeilpour. "On the application of physics informed neural networks (PINN) to solve boundary layer thermal-fluid problems." *International Communications in Heat and Mass Transfer*, **132**:105890, 2022.

[Bez74] N. Y. Beznoshchenko. "On finding a coefficient in a parabolic equation." *Differential Equations*, **10**:24–35, 1974.

[BFH12] I. Babuska, J. E. Flaherty, W. D. Henshaw, J. E. Hopcroft, J. E. Oliger, and T. Tezduyar. *Modeling, Mesh generation, and Adaptive Numerical Methods for Partial Differential Equations*, volume 75. Springer Science & Business Media, 2012.

[BH21] L. Böttcher and H. J. Herrmann. *Computational Statistical Physics*. Cambridge University Press, 2021.

[BK04] J. Betschinger and J. A. Knoblich. "Dare to be different: asymmetric cell division in Drosophila, C. elegans and vertebrates." *Current Biology*, **14**(16):R674–R685, 2004.

[BMA23] C. Bajaj, L. McLennan, T. Andeen, and A. Roy. "Recipes for when physics fails: recovering robust learning of physics informed neural networks." *Machine Learning: Science and Technology*, **4**:015013, 2023.

[BN21] L. Böttcher and J. Nagler. "Decisive conditions for strategic vaccination against SARS-CoV-2." *Chaos: An Interdisciplinary Journal of Nonlinear Science*, **31**(10):101105, 2021.

[BNM13] C. Brown, A. Noulas, C. Mascolo, and V. Blondel. "A place-focused model for social networks in cities." In *2013 International Conference on Social Computing*, pp. 75–80. IEEE, 2013.

[BOY11] J. Barré, A. Olivetti, and Y. Y. Yamaguchi. "Algebraic damping in the one-dimensional Vlasov equation." *Journal of Physics A: Mathematical and Theoretical*, **44**(40):405502, 2011.

[BPR18] A. G. Baydin, B. A. Pearlmutter, A. A. Radul, and J. M. Siskind. "Automatic differentiation in machine learning: a survey." *Journal of Marchine Learning Research*, **18**:1–43, 2018.

[BPV03] M. Boguná, R. Pastor-Satorras, and A. Vespignani. "Absence of epidemic threshold in scale-free networks with degree correlations." *Physical Review Letters*, **90**(2):028701, 2003.

[BRB20] M. P. Barman, T. Rahman, K. Bora, and C. Borgohain. "COVID-19 pandemic and its recovery time of patients in India: A pilot study." *Diabetes & Metabolic Syndrome: Clinical Research & Reviews*, **14**(5):1205–1211, 2020.

[BVO20] K. J. Burns, G. M. Vasil, J. S. Oishi, D. Lecoanet, and B. P. Brown. "Dedalus: A flexible framework for numerical simulations with spectral methods." *Physical Review Research*, **2**(2):023068, 2020.

[BWA15] L. Böttcher, O. Woolley-Meza, N. A. Araújo, H. J. Herrmann, and D. Helbing. "Disease-induced resource constraints can trigger explosive epidemics." *Scientific Report*, **5**(1):1–11, 2015.

[BWG16] L. Böttcher, O. Woolley-Meza, E. Goles, D. Helbing, and H. J. Herrmann. "Connectivity disruption sparks explosive epidemic spreading." *Physics Review E*, **93**(4):042315, 2016.

[BWW21] J. Brandstetter, D. Worrall, and M. Welling. "Message passing neural PDE solvers." In *International Conference on Learning Representations*, 2021.

[BXC20] L. Böttcher, M. Xia, and T. Chou. "Why case fatality ratios can be misleading: individual-and population-based mortality estimates and factors influencing them." *Physical Biology*, **17**(6):065003, 2020.

[BZA11] S. Bhattacharya, Q. Zhang, and M. E. Andersen. "A deterministic map of Waddington's epigenetic landscape for cell fate specification." *BMC Systems Biology*, **5**(1):1–12, 2011.

[Can68] J. R. Cannon. "Determination of an unknown heat source from overspecified boundary data." *SIAM Journal on Numerical Analysis*, **5**(2):275–286, 1968.

[Cas05] H. Caswell. "Sensitivity analysis of the stochastic growth rate: three extensions." *Australian & New Zealand Journal of Statistics*, **47**(1):75–85, 2005.

[CCZ20] Z. Chen, Y. Cao, D. Zou, and Q. Gu. "How much over-parameterization is sufficient to learn deep ReLU networks?" In *International Conference on Learning Representations*, 2020.

[CDC21] CDC. "New COVID-19 Variants.", 2021, accessed: January 15, 2021.

[CFK90] O. Coulaud, D. Funaro, and O. Kavian. "Laguerre spectral approximation of elliptic problems in exterior domains." *Computer Methods in Applied Mechanics and Engineering*, **80**:451–458, 1990.

[CG16] T. Chou and C. D. Greenman. "A hierarchical kinetic theory of birth, death and fission in age-structured interacting populations." *Journal of Statistical Physics*, **164**:49–76, 2016.

[CHS20] M. Coomer, L. Ham, and M. P. H. Stumpf. "Shaping the epigenetic landscape: Complexities and consequences." *bioRxiv*, 2020.

[CHS22] M. A. Coomer, L. Ham, and M. P. H. Stumpf. "Noise distorts the epigenetic landscape and shapes cell-fate decisions." *Cell Systems*, **13**(1):83–102, 2022.

[CKS16] J. Cuevas–Maraver, P. G. Kevrekidis, A. Saxena, A. Comech, and R. Lan. "Stability of solitary waves and vortices in a 2D nonlinear Dirac model." *Physics Review Letters*, **116**:214101, 2016.

[CMG18] C. Cadart, S. Monnier, J. Grilli, P.J. Sáez, N. Srivastava, R. Attia, E. Terriac, B. Baum, M. Cosentino-Lagomarsino, and M. Piel. "Size control in mammalian cells involves modulation of both growth rate and cell cycle duration." *Nature Communications*, **9**:3275, 2018.

[CNS17] C. Canuto, R. H. Nochetto, R. Stevenson, and M. Verani. "On p-robust saturation for hp-AFEM." *Computers & Mathematics with Applications*, **73**(9):2004–2022, 2017.

[cor20] "COVID-19 statistics." https://www.worldometers.info/coronavirus/, 2020. Accessed: 2020-02-26.

[COV20] SG COVID19. "Dashboard of the COVID-19 virus outbreak in Singapore." https://co.vid19.sg/cases, 2020. Accessed: 2020-04-04.

[CS21] W. Choi and E. Shim. "Optimal strategies for social distancing and testing to control COVID-19." *Journal of Theoretical Biology*, **512**:110568, 2021.

[CSW17] D. Chandler-Brown, K.M. Schmoller, Y. Winetraub, and J.M. Skotheim. "The Adder Phenomenon Emerges from Independent Control of Pre- and Post-Start Phases of the Budding Yeast Cell Cycle." *Current Biology*, **27**:2774–2783, 2017.

[CSX23] T. Chou, S. Shao, and M. Xia. "Adaptive Hermite spectral methods in unbounded domains." *Applied Numerical Mathematics*, **183**:201–220, 2023.

[DC] European Centre for Disease Prevention and Control. "Field epidemiology manual Wiki.".

[DC20] M. D'Arienzo and A. Coniglio. "Assessment of the SARS-CoV-2 basic reproduction number, R0, based on the early phase of COVID-19 outbreak in Italy." *Biosafety and Health*, **2**(2):57–59, 2020.

[DDG20] E. Dong, H. Du, and L. Gardner. "An interactive web-based dashboard to track COVID-19 in real time." *The Lancet Infectious Diseases*, 2020.

[DG10] M. Doumic Jauffret and P. Gabriel. "Eigenelements of a general aggeregation-fragmentation model." *Mathematical Models and Methods in Applied Sciences*, **20**:757–783, 2010.

[DHK15] M. Doumic, M. Hoffmann, N. Krell, and L. Robert. "Statistical estimation of a growth-fragmentation model observed on a genealogical tree." *Bernoulli*, **21**:1760–1799, 2015.

[DHM90] O. Diekmann, J.A.P. Heesterbeek, and J.A. Metz. "On the definition and the computation of the basic reproduction ratio R_0 in models for infectious diseases in heterogeneous populations." *Journal of Mathematical Biology*, **28**(4):365–382, 1990.

[DKT07] M. Dumbser, M. Käser, and E. F. Toro. "An arbitrary high-order Discontinuous Galerkin method for elastic waves on unstructured meshes-V. Local time stepping and p-adaptivity." *Geophysical Journal International*, **171**(2):695–717, 2007.

[DL92] R. Dautray and J.-L. Lions. *Mathematical Analysis and Numerical Methods for Science and Technology: Volume 5 Evolution Problems I.* Springer, 1992.

[DPZ09] M. Doumic, B. Perthame, and J. P. Zubelli. "Numerical solution of an inverse problem in size-structured population dynamics." *Inverse Problems*, **25**:045008, 2009.

[Dur19] R. Durrett. *Probability: Theory and Examples*, volume 49. Cambridge U Press, 2019.

[DW02] P. Van den Driessche and J. Watmough. "Reproduction numbers and subthreshold endemic equilibria for compartmental models of disease transmission." *Mathematical Biosciences*, **180**(1-2):29–48, 2002.

[DWH17] M. Delarue, D. Weissman, and O. Hallatschek. "A simple molecular mechanism explains multiple patterns of cell-size regulation." *PLoS One*, **12**(8):e0182633, 2017.

[Fam20] M. Famulare. "2019-nCoV: preliminary estimates of the confirmed-case-fatality-ratio and infection-fatality-ratio, and initial pandemic risk assessment." *Institute for Disease Modeling*, **19**, 2020.

[Foe59] H. von Foerster. "Some remarks on changing populations." *The Kinetics of Cellular Proliferation, Grune and Stratton*, pp. 382–407, 1959.

[FZ19] Z. Fang and J. Zhan. "A physics-informed neural network framework for PDEs on 3D surfaces: Time independent problems." *IEEE Access*, **8**:26328–26335, 2019.

[Gar09] C. Gardiner. *Stochastic Methods: A Handbook for the Natural and Social Sciences.* Springer., 2009.

[GC16] C. D. Greenman and T. Chou. "Kinetic theory of age-structured stochastic birth-death processes." *Physical Review E*, **93**(1):012112, 2016.

[GDC05] A.C. Ghani, C.A. Donnelly, D.R. Cox, J.T. Griffin, C. Fraser, T.H. Lam, L.M. Ho, R.M. Chan, W.S.and Anderson, A.J. Hedley, and G.M. Leung. "Methods for estimating the case fatality ratio for a novel, emerging infectious disease." *American Journal of Epidemiology*, **162**(5):479–486, 2005.

[Gil77] D. T. Gillespie. "Exact stochastic simulation of coupled chemical reactions." *The Journal of Physical Chemistry*, **81**(25):2340–2361, 1977.

[GJJ96] M. Guo, L. Y. Jan, and Y. N. Jan. "Control of daughter cell fates during asymmetric division: interaction of numb and notch." *Neuron*, **17**(1):27–41, 1996.

[GKC17] G. Gul Zaman, Y. H. Kang, G. Cho, and I. H. Jung. "Optimal strategy of vaccination & treatment in an SIR epidemic model." *Mathematics and Computers in Simulation*, **136**:63–77, 2017.

[GLD09] T. Garske, J. Legrand, C.A. Donnelly, H. Ward, S. Cauchemez, C. Fraser, N.M. Ferguson, and A.C. Ghani. "Assessing the severity of the novel influenza A/H1N1 pandemic." *BMJ*, **339**:b2840, 2009.

[Gre17] C. D. Greenman. "A path integral approach to age dependent branching processes." *Journal of Statistical Mechanics*, **2017**:033101, 2017.

[GSP20] G. Gorin, V. Svensson, and L. Pachter. "Protein velocity and acceleration from single-cell multiomics experiments." *Genome Biology*, **21**(1):1–6, 2020.

[GW01] B.-Y. Guo and L.-L. Wang. "Jacobi interpolation approximations and their applications to singular differential equations." *Advances in Computational Mathematics*, **14**:227–276, 2001.

[GWW06] B. Y. GUO, L.-L. Wang, and Z. Q. Wang. "Generalized Laguerre interpolation and pseudospectral method for unbounded domains." *SIAM Journal on Numerical Analysis*, **43**:2567–2589, 2006.

[HD81] O. Huisman and R. D'Ari. "An inducible DNA replication–cell division coupling mechanism in E. coli." *Nature*, **290**(5809):797–799, 1981.

[HDO12] R. V. Hugli, G. Duff, B. O'Conchuir, E. Mengotti, A. F. Rodriguez, F. Nolting, L. J. Heyderman, and H. B. Braun. "Artificial Kagomé spin ice: dimensional reduction, avalanche control and emergent magnetic monopoles." *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **370**(1981):5767–5782, 2012.

[HH92] H. R. Horvitz and I Herskowitz. "Mechanisms of asymmetric cell division: two Bs or not two Bs, that is the question." *Cell*, **68**(2):237–255, 1992.

[HJW05] H. Han, J. Jin, and X. Wu. "A finite-difference method for the one-dimensional time-dependent Schrödinger equation on unbounded domain." *Computer & Mathematics with Applications*, **50**(8-9):1345–1362, 2005.

[HL07] T. Y. Hou and R. Li. "Computing nearly singular solutions using pseudo-spectral methods." *Journal of Computational Physics*, **226**:379–397, 2007.

[HLA18] P.-Y. Ho, J. Lin, and A. Amir. "Modeling cell size regulation: from single-cell-level statistics to molecular mechanisms and population-level effects." *Annual Review of Biophysics*, **47**(1):251–271, 2018.

[HLL83] P. W. Holland, K. B. Laskey, and S. Leinhardt. "Stochastic blockmodels: First steps." *Social networks*, **5**(2):109–137, 1983.

[HLW20] X. He, E. H. Lau, P. Wu, X. Deng, J. Wang, X. Hao, Y.C. Lau, J. Y. Wong, Y. Guan, X. Tan, X. Mo, Y. Chen, B. Liao, W. Chen, F. Hu, Q. Zhang, M. Zhong, Y. Wu, L. Zhao, F. Zhang, B. J. Cowling, F. Li, and G. M. Leung. "Temporal dynamics in viral shedding and transmissibility of COVID-19." *Nature Medicine*, **26**:672–675, 2020.

[HM22] A. Hyafil and D. Moriña. "Analysis of the impact of lockdown on the reproduction number of the SARS-Cov-2 in Spain." *Gaceta Sanitaria*, **35**:453–458, 2022.

[Hor91] K. Hornik. "Approximation capabilities of multilayer feedforward networks." *Neural Networks*, **4**(2):251–257, 1991.

[HP14] A. Hasanov and B. Pekta**c**c. "A unified approach to identifying an unknown spacewise dependent source in a variable coefficient parabolic equation from final and integral overdeterminations." *Applied Numerical Mathematics*, **78**:49–67, 2014.

[HSW89] K. Hornik, M. Stinchcombe, and H. White. "Multilayer feedforward networks are universal approximators." *Neural Networks*, **2**(5):359–366, 1989.

[Hun21] M. J. Huntul. "Identification of the timewise thermal conductivity in a 2D heat equation from local heat flux conditions." *Inverse Problems in Science and Engineering*, **29**(7):903–919, 2021.

[Ian95] M. Iannelli. "Mathematical theory of age-structured population dynamics." *Giardini Editori E Stampatori in Pisa*, 1995.

[IKS19] A. Iserles, K. Kropielnicka, and P. Singh. "Solving Schrödinger equation in semi-classical regime with highly oscillatory time-dependent potentials." *Journal of Computational Physics*, **376**:564–584, 2019.

[IPB19] I. Iacopini, G. Petri, A. Barrat, and V. Latora. "Simplicial models of social contagion." *Nature Communications*, **10**(1):2485, 2019.

[IS15] S. Ioffe and C. Szegedy. "Batch normalization: accelerating deep network training by reducing internal covariate shift." In *International Conference on Machine Learning*, pp. 448–456. PMLR, 2015.

[Iva93] N. I. Ivanchov. "Inverse problems for the heat-conduction equation with nonlocal boundary conditions." *Ukrainian Mathematical Journal*, **45**(8):1186–1192, 1993.

[JAH20] S.M. Jung, A.R. Akhmetzhanov, K. Hayashi, N.M. Linton, Y. Yang, B. Yuan, T. Kobayashi, R. Kinoshita, and H. Nishiura. "Real-time estimation of the risk of death from novel coronavirus (COVID-19) infection: inference using exported cases." *Journal of Clinical Medicine*, **9**(2):523, 2020.

[JK20] A. D. Jagtap and G. E. Karniadakis. "Extended physics-informed neural networks (xpinns): A generalized space-time domain decomposition based deep learning framework for nonlinear partial differential equations." *Communications in Computational Physics*, **28**(5):2002–2041, 2020.

[JL07] B. T. Johansson and D. Lesnic. "A variational method for identifying a spacewise-dependent heat source." *IMA Journal of Applied Mathematics*, **72**(6):748–760, 2007.

[Jon62] B. F. Jones Jr. "The Determination of a Coefficient in a Parabolic Differential Equation: Part I. Existence and Uniqueness." *Journal of Mathematics and Mechanics*, **11**:907–918, 1962.

[KB17] D. A. Kessler and S. Burov. "Stochastic maps, continuous approximation, and stable distribution." *Physical Review E*, **96**:042139, Oct 2017.

[KB18] D. A. Kessler and S. Burov. "Effective Potential for Cellular Size Control." *Bulletin of the American Physical Society*, **63**, 2018.

[KC13] H. Kelly and B.J. Cowling. "Case Fatality: Rate, Ratio, or Risk?" *Epidemiology*, **24**(4):622–623, 2013.

[KGC17] Jan Kukačka, Vladimir Golkov, and Daniel Cremers. "Regularization for deep learning: A taxonomy." *arXiv preprint arXiv:1710.10686*, 2017.

[KGK06] I. Z. Kiss, D. M. Green, and R. R. Kao. "The effect of contact heterogeneity and multiple routes of transmission on final epidemic size." *Mathematical Biosciences*, **203**(1):124–136, 2006.

[KKL21] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang. "Physics-informed machine learning." *Nature Reviews Physics*, **3**(6):422–440, 2021.

[KMB20] G.G. Katul, A. Mrad, S. Bonetti, G. Manoli, and A.J. Parolari. "Global convergence of COVID-19 basic reproduction number and estimation from early-time SIR dynamics." *PLoS One*, **15**(9):e0239800, 2020.

[KR11] M. J. Keeling and P. Rohani. *Modeling Infectious Diseases in Humans and Animals*. Princeton University Press, 2011.

[KX01] B. L. Karihaloo and Q. Z. Xiao. "Accurate determination of the coefficients of elastic crack tip asymptotic field by a hybrid crack element with p-adaptivity." *Engineering Fracture Mechanics*, **68**(15):1609–1630, 2001.

[KZK19] E. Kharazmi, Z. Zhang, and G. E. Karniadakis. "Variational physics-informed neural networks for solving partial differential equations." *arXiv preprint arXiv:1912.00873*, 2019.

[LA17] J. Lin and A. Amir. "The effects of stochasticity at the single-cell level and cell size control on the population growth." *Cell Systems*, **5**(4):358–367, 2017.

[LB20] F. Liu and M. Buss. "Optimal control for heterogeneous node-based information epidemics over social networks." *IEEE Transactions on Control of Network Systems*, **7**(3):1115–1126, 2020.

[LDF15] M. Lipsitch, C. A. Donnelly, C. Fraser, Isobel M. Blake, A. Cori, I. Dorigatti, N. M. Ferguson, T. Garske, H. L. Mills, S. Riley, Maria D. Van K., and M.A Hernán. "Potential biases in estimating absolute and relative case-fatality risks during outbreaks." *PLOS Neglected Tropical Diseases*, **9**(7):e0003846, 07 2015.

[Lin76] S. Linnainmaa. "Taylor expansion of the accumulated rounding error." *BIT Numerical Mathematics*, **16**(2):146–160, 1976.

[LJP21] L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis. "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators." *Nature Machine Intelligence*, **3**(3):218–229, 2021.

[LJY20] F. W. Lewis, S. Jagannathan, and A. Yesildirak. *Neural Network Control of Robot Manipulators and Nonlinear Systems*. CRC Press, 2020.

[LKA20] Z. Li, N. B. Kovachki, K. Azizzadenesheli, K. Bhattacharya, A. Stuart, and A. Anandkumar. "Fourier neural operator for parametric partial differential equations." In *International Conference on Learning Representations*, 2020.

[LLM02] R. Li, W. Liu, H. Ma, and T. Tang. "Adaptive finite element approximation for distributed elliptic optimal control problems." *SIAM Journal on Control and Optimization*, **41**(5):1321–1349, 2002.

[LLM18] Z. Long, Y. Lu, X. Ma, and B. Dong. "PDE-net: Learning PDEs from data." In *International Conference on Machine Learning*, pp. 3208–3216. PMLR, 2018.

[LLS20] M. Liu, L. Liang, and W. Sun. "A generic physics-informed neural network-based constitutive model for soft biological tissues." *Computer methods in applied mechanics and engineering*, **372**:113402, 2020.

[LMD11] J. Lindquist, J. Ma, P. Van den Driessche, and F. H. Willeboordse. "Effective degree network disease models." *Journal of Mathematical Biology*, **62**(2):143–164, 2011.

[LOR15] A. Lecavil, N. Oudjane, and F. Russo. "Probabilistic representation of a class of non conservative nonlinear partial differential equations." *arXiv preprint arXiv:1504.03882*, 2015.

[LPC20] R. Li, S. Pei, B. Chen, Y. Song, T. Zhang, W. Yang, and J. Shaman. "Substantial undocumented infection facilitates the rapid dissemination of novel coronavirus (SARS-CoV-2)." *Science*, **368**(6490):489–493, 2020.

[LRP19] M. Lutter, C. Ritter, and J. Peters. "Deep Lagrangian networks: using physics as model prior for deep learning." In *International Conference on Learning Representations*. OpenReview.net, 2019.

[LSK20] C.C. Lai, T.P. Shih, W.C. Ko, H.J. Tang, and P.R. Hsueh. "Severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) and corona virus disease-2019 (COVID-19): the epidemic and the challenges." *International Journal of Antimicrobial Agents*, **55**:105924, 2020.

[LSZ18] G. La Manno, R. Soldatov, A. Zeisel, E. Braun, H. Hochgerner, V. Petukhov, K. Lidschreiber, M. E. Kastriti, P. Lönnerberg, A. Furlan, et al. "RNA velocity of single cells." *Nature*, **560**(7719):494–498, 2018.

[LTZ01] R. Li, T. Tang, and P. Zhang. "Moving mesh methods in multiple dimensions based on harmonic maps." *Journal of Computational Physics*, **170**(2):562–588, 2001.

[LXP22] W. Li, X. Xue, L. Pan, T. Lin, and W. Wang. "Competing spreading dynamics in simplicial complex." *Applied Mathematics and Computation*, **412**:126595, 2022.

[LZK21] Z. Li, H. Zheng, N. Kovachki, D. Jin, H. Chen, B. Liu, K. Azizzadenesheli, and A. Anandkumar. "Physics-informed neural operator for learning partial differential equations." *arXiv preprint arXiv:2111.03794*, 2021.

[LZZ18] B. Li, J. Zhang, and C. Zheng. "Stability and error analysis for a second-order fast approximation of the one-dimensional Schrödinger equation under absorbing boundary conditions." *SIAM Journal on Scientific Computing*, **40**(6):A4083–A4104, 2018.

[M98] J. Müller. "Optimal vaccination patterns in age-structured populations." *SIAM Journal on Applied Mathematics*, **59**(1):222–241, 1998.

[MA88] R. M. May and R. M. Anderson. "The transmission dynamics of human immunodeficiency virus (HIV)." *Philosophical Transactions of the Royal Society of London. B, Biological Sciences*, **321**(1207):565–607, 1988.

[Mah20] E. Mahase. "Coronavirus: COVID-19 has killed more people than SARS and MERS combined, despite lower case fatality rate.", 2020.

[MC20] K. Mizumoto and G. Chowell. "Estimating risk for death from 2019 novel coronavirus disease, China, January-February 2020." 2020.

[McK26] A. G. McKendrick. "Applications of mathematics to medical problems." *Proceedings of the Edinburgh Mathematical Society*, **44**:98–130, 1926.

[MD86] J. A. J. Metz and O. Diekmann. *The Dynamics of Physiologically Structured Populations*. Springer, 1986.

[MHD21] S. Moore, E. M. Hill, L. Dyson, M. J. Tildesley, and M. J. Keeling. "Modelling optimal vaccination strategy for SARS-CoV-2 in the UK." *PLoS Computational Biology*, **17**:e1008849, 2021.

[MHR11] E. Mengotti, L. J. Heyderman, A. F. Rodriguez, F. Nolting, R. V. Hugli, and H.-B. Braun. "Real-space observation of emergent magnetic monopoles and associated Dirac strings in artificial Kagomé spin ice." *Nature Physics*, **7**(1):68–74, 2011.

[MJK20] Z. Mao, A. D. Jagtap, and G. E. Karniadakis. "Physics-informed neural networks for high-speed flows." *Computer Methods in Applied Mechanics and Engineering*, **360**:112789, 2020.

[MKS15] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, and S. Peterson. "Human-level control through deep reinforcement learning." *Nature*, **518**(7540):529–533, 2015.

[MRO21] E. Mathieu, H. Ritchie, E. Ortiz-Ospina, M. Roser, J. Hasell, C. Appel, C. Giattino, and L. Rodés-Guirao. "A global database of COVID-19 vaccinations." *Nature Human Behaviour*, **5**:947–953, 2021.

[MST05] H. Ma, W. Sun, and T. Tang. "Hermite spectral methods with a time-dependent scaling for parabolic equations in unbounded domains." *SIAM Journal on Numerical Analysis*, **43**:58–75, 2005.

[MV78] C. Moler and C. Van Loan. "Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later." *SIAM Review*, **20**(4):801–836, 1978.

[MVC20] G. S. Misyris, A. Venzke, and S. Chatzivasileiadis. "Physics-informed neural networks for power systems." In *2020 IEEE Power & Energy Society General Meeting (PESGM)*, pp. 1–5. IEEE, 2020.

[MVG17] S. Modi, C.A. Vargas-Garcia, K.R. Ghusinga, and A. Singh. "Analysis of noise mechanisms in cell-size control." *Biophysical Journal*, **112**:2408–2418, 2017.

[NDF16] C. D. Nadell, K. Drescher, and K. R. Foster. "Spatial structure, cooperation and competition in biofilms." *Nature Reviews Microbiology*, **14**(9):589–600, 2016.

[New18] M. Newman. *Networks*. Oxford University Press, 2018.

[NLV21] S. Nowak, P. N. de Lima, and R. Vardavas. "Should we mitigate or suppress the next pandemic? Time-horizons and costs shape optimal social distancing strategies." *medRxiv*, 2021.

[NVP21] C. Nieto, C. Vargas-Garcia, and J. M. Pedraza. "Continuous rate modeling of bacterial stochastic size dynamics." *Physical Review E*, **104**(4):044415, 2021.

[OCK21] G. Ódor, D. Czifra, J. Komjáthy, L. Lovász, and M. Karsai. "Switchover phenomenon induced by epidemic seeding on geometric networks." *Proceedings of the National Academy of Sciences*, **118**(41):e2112607118, 2021.

[OH20] J. Oke and C. Heneghan. "Global Covid-19 case fatality rates: Oxford COVID-19 Evidence Service.", 2020. Accessed: 2020-03-27.

[OM00] P. Ogren and C. F. Martin. "Optimal vaccination strategies for the control of epidemics in highly mobile populations." In *Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No. 00CH37187)*, volume 2, pp. 1782–1787. IEEE, 2000.

[ORB20] G. Onder, G. Rezza, and S. Brusaferro. "Case-fatality rate and characteristics of patients dying in relation to COVID-19 in Italy." *Jama*, **323**:1775–1776, 2020.

[Org20a] World Health Organization. "Cumulative number of reported probable cases of severe acute respiratory syndrome (SARS)." https://www.who.int/csr/sars/country/en/, 2020. Accessed: 2020-03-30.

[Org20b] World Health Organization. "WHO director-general's opening remarks at the media briefing on COVID-19 - 24 February 2020." https://www.who.int/dg/speeches/detail/who-director-general-s-opening-remarks-at-the-media-briefing-on-covid-19—24-february-2020, 2020. Accessed: 2020-02-28.

[Org20c] World Health Organization. "WHO Director-General's opening remarks at the media briefing on COVID-19 - 3 March 2020." https://www.who.int/dg/speeches/detail/who-director-general-s-opening-remarks-at-the-media-briefing-on-covid-19—3-march-2020, 2020. Accessed: 2020-03-05.

[Ors71] S. A. Orszag. "On the elimination of aliasing in finite-difference schemes by filtering high-wavenumber components." *Journal of the Atmospheric Sciences*, **28**:1074–1074, 1971.

[Per08] B. Perthame. "Introduction to structured equations in biology." *CNA Summer School Lecture Notes*, 2008.

[PG21] R. Prieto Curiel and H. Gonzalez Ramirez. "Vaccination strategies against COVID-19 and the diffusion of anti-vaccination views." *Scientific Reports*, **11**:6626, 2021.

[PGC17] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. "Automatic differentiation in PyTorch." 2017.

[PPH13] M. Piraveenan, M. Prokopenko, and L. Hossain. "Percolation centrality: Quantifying graph-theoretic impact of nodes during percolation in networks." *PLoS One*, **8**(1):e53095, 2013.

[PRK21] Nathan Peiffer-Smadja, Sacha Rozencwajg, Yousra Kherabi, Yazdan Yazdanpanah, and Philippe Montravers. "COVID-19 vaccines: a race against time." *Anaesthesia, Critical Care & Pain Medicine*, **40**(2):100848, 2021.

[PS18] D. M. Popescu and S. X. Sun. "Building the space elevator: lessons from biological design." *Journal of The Royal Society Interface*, **15**(147):20180086, 2018.

[PSK20] R. Porcheddu, C. Serra, D. Kelvin, N. Kelvin, and S. Rubino. "Similarity in case fatality rates (CFR) of COVID-19/SARS-COV-2 in Italy and China." *The Journal of Infection in Developing Countries*, **14**:125–128, 2020.

[PSR20] N.C. Peeri, N. Shrestha, M.S. Rahman, R. Zaki, Z. Tan, S. Bibi, M. Baghbanzadeh, N. Aghamohammadi, W. Zhang, and U. Haque. "The SARS, MERS and novel coronavirus (COVID-19) epidemics, the newest and biggest global health threats: what lessons have we learned?" *International Journal of Epidemiology*, 02 2020.

[PTV07] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, 2007.

[PV01a] R. Pastor-Satorras and A. Vespignani. "Epidemic dynamics and endemic states in complex networks." *Physical Review E*, **63**(6):066117, 2001.

[PV01b] R. Pastor-Satorras and A. Vespignani. "Epidemic spreading in scale-free networks." *Physical Review Letters*, **86**(14):3200, 2001.

[PYL20] S. Park, C. Yun, J. Lee, and J. Shin. "Minimum width for universal approximation." In *International Conference on Learning Representations*, 2020.

[PZN23] M. Penwarden, S. Zhe, A. Narayan, and R. M. Kirby. "A metalearning approach for physics-informed neural networks (PINNs): Application to parameterized PDEs." *Journal of Computational Physics*, p. 111912, 2023.

[QCH21] B. J. Quilty, S. Clifford, J. Hellewell, T. W. Russell, A. J. Kucharski, S. Flasche, W. J. Edmunds, K. E. Atkins, A. M. Foss, N. R. Waterlow, and K. Abbas. "Quarantine and testing strategies in contact tracing for SARS-CoV-2: a modelling study." *The Lancet Public Health*, **6**(3):e175–e183, 2021.

[QZM22] X. Qiu, Y. Zhang, J. D. Martin-Rufino, C. Weng, S. Hosseinzadeh, D. Yang, A. N. Pogson, M. Y. Hein, K. H. J. Min, L. Wang, et al. "Mapping transcriptomic vector fields of single cells." *Cell*, **185**(4):690–711, 2022.

[Rai18] M. Raissi. "Deep hidden physics models: Deep learning of nonlinear partial differential equations." *Journal of Machine Learning Research*, **19**(1):932–955, 2018.

[RHC20] J. Riou, A. Hauser, M. J. Counotte, and C. L. Althaus. "Adjusted age-specific case fatality ratio during the COVID-19 epidemic in Hubei, China, January and February 2020." *medRxiv*, 2020.

[RHK14] L. Robert, M. Hoffmann, N. Krell, S. Aymerich, J. Robert, and M. Doumic. "Division in Escherichia coli is triggered by a size-sensing rather than a timing mechanism." *BMC Biology*, **12**:1–10, 2014.

[RPK19] M. Raissi, P. Perdikaris, and G. E. Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations." *Journal of Computational Physics*, **378**:686–707, 2019.

[RRB20] M. A. Roehrl, T. A. Runkler, V. Brandtstetter, M. Tokic, and S. Obermayer. "Modeling system dynamics with physics-informed neural networks based on Lagrangian mechanics." *IFAC-PapersOnLine*, **53**(2):9195–9200, 2020.

[Rua20] S. Ruan. "Likelihood of survival of coronavirus disease 2019." *The Lancet Infectious Diseases*, **20**(6):630–631, 2020.

[RW00] W. Ren and X.-P. Wang. "An iterative grid redistribution method for singular problems in multiple dimensions." *Journal of Computational Physics*, **159**(2):246–273, 2000.

[SB19] S. H. Strub and L. Böttcher. "Modeling deformed transmission lines for continuous strain sensing applications." *Measurement Science and Technology*, **31**(3):035109, 2019.

[SBK20] P. Spychalski, A. Blażyńska-Spychalska, and J. Kobiela. "Estimating case fatality rates of COVID-19." *The Lancet Infectious Diseases*, **20**(7):774–775, 2020.

[SDW22] T. Schneider, O. R. Dunbar, J. Wu, L. Böttcher, D. Burov, A. Garbuno-Inigo, G. L. Wagner, S. Pei, C. Daraio, R. Ferrari, and J. Shaman. "Epidemic management and control through risk-dependent individual contact interventions." *PLOS Computational Biology*, **18**(6):e1010171, 2022.

[Seo21] B. Seoane. "A scaling approach to estimate the COVID-19 infection fatality ratio from incomplete data." *PLoS One*, **16**:e0246831, 2021.

[SLC11] S. Shao, T. Lu, and W. Cai. "Adaptive conservative cell average spectral element methods for transient Wigner equation in quantum transport." *Communications in Computational Physics*, **9**(03):711–739, 2011.

[SLS19] F. Si, G. Le Treut, J. T. Sauls, S. Vadia, P. A. Levin, and S Jun. "Mechanistic origin of cell-size control and homeostasis in bacteria." *Current Biology*, **29**(11):1760–1770, 2019.

[SM73] L. Sompayrac and O. Maaloe. "Autorepressor model for control of DNA replication." *Nature New Biology*, **241**(109):133–135, 1973.

[SMK58] M. Schaechter, O. Maaloe, and N. O. Kjeldgaard. "Dependency on medium and temperature of cell size and chemical composition during balanced grown of Salmonella typhimurium." *Journal of General Microbiology*, **19**(3):592–606, 1958.

[SQM14] S. Shao, N. R. Quintero, F. G. Mertens, F. Cooper, A. Khare, and A. Saxena. "Stability of solitary waves in the nonlinear Dirac equation with arbitrary nonlinearity." *Physics Review E*, **90**:032915, 2014.

[SS67] J. W. Sinko and W. Streifer. "A New Model for Age-size Structure of a Population." *Ecology*, **48**(6):910–918, 1967.

[ST05] S. Shao and H. Tang. "Interaction for the solitary waves of a nonlinear Dirac model." *Physics Letter A*, **345**(1-3):119–128, 2005.

[STH11] D. Steinsaltz, S. Tuljapurkar, and C. Horvitz. "Derivatives of the stochastic growth rate." *Theoretical Population Biology*, **80**(1):1–15, 2011.

[STW11] J. Shen, T. Tang, and L.-L. Wang. *Spectral Methods: Algorithms, Analysis and Applications*. Springer Science & Business Media, New York, 2011.

[STW20] J. Sheng, C.and Shen, T. Tang, L.-L. Wang, and H. Yuan. "Fast Fourier-like mapped Chebyshev spectral-Galerkin methods for PDEs with integral fractional Laplacian in unbounded domains." *SIAM Journal on Numerical Analysis*, **58**(5):2435–2464, 2020.

[SW09] J. Shen and L. L. Wang. "Some recent advances on spectral methods for unbounded domains." *Communications in Computational Physics*, **5**:195–241, 2009.

[SW10a] J. Shen and L.-L. Wang. "Sparse spectral approximations of high-dimensional problems based on hyperbolic cross." *SIAM Journal on Numerical Analysis*, **48**(3):1087–1109, 2010.

[SW10b] J. Shen and L.-L. Wang. "Sparse spectral approximations of high-dimensional problems based on hyperbolic cross." *SIAM Journal on Numerical Analysis*, **48**(3):1087–1109, 2010.

[SYP20] F. Sahli Costabal, Y. Yang, P. Perdikaris, D. E. Hurtado, and E. Kuhl. "Physics-informed neural networks for cardiac activation mapping." *Frontiers in Physics*, **8**:42, 2020.

[TA84] T.R. Taha and M.I. Ablowitz. "Analytical and numerical aspects of certain nonlinear evolution equations. II. Numerical, nonlinear Schrödinger equation." *Journal of Computational Physics*, **44**(2):203–230, 1984.

[Tan93] T. Tang. "The Hermite spectral method for Gaussian-type functions." *SIAM Journal on Scientific Computing*, **14**:594–606, 1993.

[TBS15] S. Taheri-Araghi, S. Bradde, J. T. Sauls, N. S. Hill, P. A. Levin, J. Paulsson, M. Vergassola, and S Jun. "Cell-size control and homeostasis in bacteria." *Current Biology*, **25**(3):385–391, 2015.

[tes20a] "CDC viral test for COVID-19." https://www.cdc.gov/coronavirus/2019-ncov/php/testing.html, 2020. Accessed: 2020-05-13.

[tes20b] "Research use only 2019-novel coronavirus (2019-nCoV) real-time RT-PCR primer and probe information." https://www.cdc.gov/coronavirus/2019-ncov/lab/rt-pcr-panel-primer-probes.html, 2020. Accessed: 2020-05-13.

[TMN22] P. Thanasutives, T. Morita, M. Numao, and K. i. Fukui. "Noise-aware physics-informed machine learning for robust PDE discovery." *Machine Learning: Science and Technology*, **4**:015009, 2022.

[TNF21] P. Thanasutives, M. Numao, and K. Fukui. "Adversarial multi-task learning enhanced physics-informed neural networks for solving partial differential equations." In *2021 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–9. IEEE, 2021.

[Tre00] L. N. Trefethen. *Spectral Methods in MATLAB*. SIAM, 2000.

[Tsy98] S. V. Tsynkov. "Numerical solution of problems on unbounded domains. A review." *Applied Numerical Mathematics*, **27**(4):465–532, 1998.

[TT03] Huazhong Tang and Tang Tao. "Adaptive mesh methods for one- and two-dimensional hyperbolic conservation laws." *SIAM Journal on Numerical Analysis*, **41**:487–515, 2003.

[TWY20] T. Tang, L.-L. Wang, H. Yuan, and T. Zhou. "Rational spectral methods for PDEs involving fractional Laplacian in unbounded domains." *SIAM Journal on Scientific Computing*, **42**(2):A585–A611, 2020.

[TYZ18a] T. Tang, H. Yuan, and T. Zhou. "Hermite spectral collocation methods for fractional PDEs in unbounded domain." *Communications in Computational Physics*, **24**:1143–1168, 2018.

[TYZ18b] T. Tang, H. Yuan, and T. Zhou. "Hermite spectral collocation methods for fractional PDEs in unbounded domains." *Communications in Computational Physics*, **24**:1143–11468, 2018.

[UGA23] Z. Uddin, S. Ganga, R. Asthana, and W. Ibrahim. "Wavelets based physics informed neural networks to solve non-linear differential equations." *Scientific Reports*, **13**(1):1–19, 2023.

[US19] Y. Ueda and N. Saito. "The inf-sup condition and error estimates of the Nitsche method for evolutionary diffusion–advection-reaction equations." *Japan Journal of Industrial and Applied Mathematics*, **36**(1):209–238, 2019.

[VKF93] W. J. Voorn, L. J. Koppes, and N. B. Frover. "Mathematics of cell division in Escherichia coli cell division: comparison between sloppy-size and incremental-size kinetics." *Current Topics in Molecular Genetics*, **1**:187–194, 1993.

[VOD20] R. Verity, L.C. Okell, I. Dorigatti, P. Winskill, C. Whittaker, N. Imai, G. Cuomo-Dannenburg, H. Thompson, P.G. Walker, H. Fu, and A. Dighe. "Estimates of the severity of coronavirus disease 2019: a model-based analysis." *The Lancet Infectious Diseases*, **20**(6):669–677, 2020.

[VSS16] C. A. Vargas-Garcia, M. Soltani, and A. Singh. "Conditions for cell size homeostasis: a stochastic hybrid system approach." *IEEE Life Sciences Letters*, **2**(4):47–50, 2016.

[Web08] G.F. Webb. "Population models structured by age, size, and spatial position." In *Structured population models in biology and epidemiology*, pp. 1–49. Springer, 2008.

[Wes94] J. G. H. Wessels. "Developmental regulation of fungal cell wall formation." *Annual Review of Phytopathology*, **32**(1):413–437, 1994.

[WFL16] M. Wallden, D. Fange, E. G. Lundius, Ö Baltekin, and E. Johan. "The synchronization of replication and division cycles in individual E. coli cells." *Cell*, **166**(3):729–739, 2016.

[WLB20] J.T. Wu, K. Leung, M. Bushman, N. Kishore, R. Niehus, P.M. de Salazar, B.J. Cowling, M. Lipsitch, and G.M. Leung. "Estimating clinical severity of COVID-19 from the transmission dynamics in Wuhan, China." *Nature Medicine*, **26**:506–610, 2020.

[WM20] Z. Wu and J. M. McGoogan. "Characteristics of and important lessons from the coronavirus disease 2019 (COVID-19) outbreak in China: summary of a report of 72 314 cases from the Chinese Center for Disease Control and Prevention." *Jama*, **323**(13):1239–1242, 2020.

[WP07] S. Weber and M. Porto. "Generation of arbitrarily two-point-correlated random networks." *Physical Review E*, **76**(4):046111, 2007.

[WSD20] B. Wang, Y. Sun, T. Q. Duong, L. D. Nguyen, and L. Hanzo. "Risk-aware identification of highly suspected COVID-19 cases in Social IoT: A joint graph theory and reinforcement learning approach." *IEEE Access*, **8**:115655–115661, 2020.

[WSJ20] C. Wilasang, C. Sararat, N.C. Jitsuk, N. Yolai, P. Thammawijaya, P. Auewarakul, and C. Modchang. "Reduction in effective reproduction number of COVID-19 is higher in countries employing active case detection with prompt isolation." *Journal of Travel Medicine*, **27**(5):1–3, 2020.

[WT19] T. Wu and M. Tegmark. "Toward an artificial intelligence physicist for unsupervised learning." *Physical Review E*, **100**(3):033311, 2019.

[WWP21] S. Wang, H. Wang, and P. Perdikaris. "Learning the solution operator of parametric partial differential equations with physics-informed deeponets." *Science Advances*, **7**(40):eabi8605, 2021.

[WXC21] Z. Wang, C. Xia, Z. Chen, and G. Chen. "Epidemic propagation with positive and negative preventive information in multiplex networks." *IEEE Transactions on Cybernetics*, **51**(3):1454–1462, 2021.

[WYW20] Y. Wang, X. Y. You, Y. J. Wang, L. P Peng, Z. C. Du, S. Gilmour, D. Yoneoka, J. Gu, C. Hao, Y. T. Hao, and J. H. Li. "Estimating the basic reproduction number of COVID-19 in Wuhan, China." *Chinese Journal of Epidemiology*, **41**(4):476–479, 2020.

[WZX11] J. Wang, K. Zhang, L. Xu, and E. Wang. "Quantifying the Waddington landscape and biological paths for development and differentiation." *Proceedings of the National Academy of Sciences*, **108**(20):8257–8262, 2011.

[XBC22] M. Xia, L. Böttcher, and T. Chou. "Controlling epidemics through optimal allocation of test kits and vaccine doses across networks." *IEEE Transactions on Network Science and Engineering*, **9**(3):1422–1436, 2022.

[XBC23] M. Xia, L. Böttcher, and T. Chou. "Spectrally adapted physics-informed neural networks for solving unbounded domain problems." *Machine Learning: Science and Technology*, **4**(2):025024, 2023.

[XC21] M. Xia and T. Chou. "Kinetic theory for structured populations: application to stochastic sizer-timer models of cell proliferation." *Journal of Physics A: Mathematical and Theoretical*, **54**(38):385601, sep 2021.

[XG22] Y. Xiong and X. Guo. "A short-memory operator splitting scheme for constant-Q viscoelastic wave equation." *Journal of Computational Physics*, **449**:110796, 2022.

[XGC20] M. Xia, C. D. Greenman, and T. Chou. "PDE models of adder mechanisms in cellular proliferation." *SIAM Journal on Applied Mathematics*, **80**(3):1307–1335, 2020.

[XLT20] Z. Xu, S. Li, S. Tian, H. Li, and L.-Q. Kong. "Full spectrum of COVID-19 severity still being depicted." *The Lancet*, **395**(10228):947–948, 2020.

[XSC21a] M. Xia, S. Shao, and T. Chou. "Efficient scaling and moving techniques for spectral methods in unbounded domains." *SIAM Journal on Scientific Computing*, **43**(5):A3244–A3268, 2021.

[XSC21b] M. Xia, S. Shao, and T. Chou. "A frequency-dependent p-adaptive technique for spectral methods." *Journal of Computational Physics*, **446**:110627, 2021.

[XST15] J. Xu, S. Shao, H. Tang, and D. Wei. "Multi-hump solitary waves of a nonlinear Dirac equation." *Communications in Mathematical Sciences*, **13**(5):1219–1242, 2015.

[XSW20] Z. Xu, L. Shi, Y. Wang, J. Zhang, L. Huang, C. Zhang, S. Liu, P. Zhao, H. Liu, L. Zhu, and Y. Tai. "Pathological findings of COVID-19 associated with acute respiratory distress syndrome." *The Lancet respiratory medicine*, **8**(4):420–422, 2020.

[YAT20] I. Yelin, N. Aharony, E. S. Tamar, A. Argoetti, E. Messer, D. Berenbaum, E. Shafran, A. Kuzli, N. Gandali, O. Shkedi, and T. Hashimshony. "Evaluation of COVID-19 RT-qPCR test in multi sample pools." *Clinical Infectious Diseases*, **71**(16):2073–2078, 2020.

[YDY11] L. Yang, M. Dehghan, J.-N. Yu, and G.-W. Luo. "Inverse problem of time-dependent heat sources numerical reconstruction." *Mathematics and Computers in Simulation*, **81**(8):1656–1672, 2011.

[YF10] F. Yang and C.-L. Fu. "A simplified Tikhonov regularization method for determining the heat source." *Applied Mathematical Modelling*, **34**(11):3286–3299, 2010.

[YLL05a] P.S. Yip, K.F. Lam, E.H. Lau, P.H. Chau, K.W. Tsang, and A. Chao. "A comparison study of realtime fatality rates: severe acute respiratory syndrome in Hong Kong, Singapore, Taiwan, Toronto and Beijing, China." *Journal of the Royal Statistical Society*, **168**(1):233–243, 2005.

[YLL05b] P.S. Yip, E.H. Lau, K.F. Lam, and R.M. Huggins. "A chain multinomial model for estimating the real-time fatality rate of a disease, with an application to severe acute respiratory syndrome." *American Journal of Epidemiology*, **161**(7):700–706, 2005.

[YMK21] L. Yang, X. Meng, and G. E. Karniadakis. "B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data." *Journal of Computational Physics*, **425**:109913, 2021.

[Yua21] H. Yuan. "An efficient spectral-Galerkin method for fractional reaction-diffusion equations in unbounded domains." *Journal of Computational Physics*, **428**:110083, 2021.

[YYF09] L. Yan, F.-L. Yang, and C.-L. Fu. "A meshless method for solving an inverse spacewise-dependent heat source problem." *Journal of Computational Physics*, **228**(1):123–136, 2009.

[YZ14] X. Yang and J. Zhang. "Computation of the Schrödinger equation in the semiclassical regime on an unbounded domain." *SIAM Journal on Numerical Analysis*, **52**(2):808–831, 2014.

[ZDC19] Y. D. Zhong, B. Dey, and A. Chakraborty. "Symplectic ODE-net: learning Hamiltonian dynamics with control." In *International Conference on Learning Representations*, 2019.

[ZLZ12] Y. Zhao, E. Levina, and J. Zhu. "Consistency of community detection in networks under degree-corrected stochastic block models." *The Annals of Statistics*, **40**(4):2266–2292, 2012.

[ZNL21] Z.Y. Zhao, Y. Niu, L. Luo, Q.Q. Hu, T.L. Yang, M.J. Chu, Q.P. Chen, Z. Lei, J. Rui, C.L. Song, and S.N. Lin. "The optimal vaccination strategy to control COVID-19: a modeling study based on the transmission scenario in Wuhan City, China." *Infectious diseases of poverty*, **10**:48–73, 2021.

[ZZ17] L. Zheng and X. Zhang. *Modeling and Analysis of Modern Fluid Problems*. Academic Press, 2017.