UNIVERSITY OF CALIFORNIA, MERCED

Exponential time integration and chemical combustion simulations

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

in

Applied Mathematics

by

Jared James Stewart

Committee in charge:

Mayya Tokman, Chair
Fabrizio Bisetti
Changho Kim
Maxime Theillard

2024

The dissertation of Jared James Stewart is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

(Fabrizio Bisetti)

(Changho Kim)

(Maxime Theillard)

(Mayya Tokman, Chair)

University of California, Merced

2024

DEDICATION

I must acknowledge that a portion of every milestone is owed to others. Therefore, this thesis is dedicated to the following people:

First, I dedicate much of this work to my mother, father, and grandmother. Their hard work and dedication offered me a brighter future: a future they knew was beyond their reach. I would not be who and where I am today without them. They all saw me begin this journey, but sadly, only my father will see me complete it.

Secondly, a thesis is not just a culmination of personal efforts but a testament to the guidance and support of a mentor. Over the course of these seven years, which were marked by the challenges of COVID-19 and personal loss, Mayya has been a beacon of unwavering support, lifting me up and equipping me with the tools to become who I am today. I am eternally grateful for her and her Herculean effort.

My partner has been my emotional rock in these trying times. Thanks to her support, understanding, compassion, and love, I have endured. At the threshold of the end, we look to begin a new chapter together: The beginning of our future.

Finally, I want to thank my fellow students in our department. Because of you, I have enjoyed my time at UC Merced. Walking this long path with you has taught me the strength of comradery and community. I cannot wait to see what awaits us on the other side of this long journey.

EPIGRAPH

Est scientia illumit mundus.

Fiat lux.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

VITA

2008	B. S. in Sociology, California State University, Northridge
2017	M. S. in Applied Mathematics, California State University, Northridge
2024	Ph. D. in Applied Mathematics, University of California, Merced

ABSTRACT OF THE DISSERTATION

Exponential time integration and chemical combustion simulations

by

Jared James Stewart
Doctor of Philosophy in Applied Mathematics
University of California Merced, 2024
Mayya Tokman, Chair

Computational combustion plays a major role in engineering applications. However, solving the large systems of differential equations that model combustive processes is challenging. One of the major computational difficulties with the numerical modeling of combustion lies in the widely varying time scales present in the model equations that describe chemical interactions of species with the fluid and thermodynamic transport phenomena. The resulting dramatic stiffness of the equations demands the development of more efficient temporal integrators to enable efficient and accurate combustion simulation.

Traditionally, the stiffness of computational combustion models has been addressed using implicit methods. However, the performance of the implicit schemes depends highly on the availability of an efficient preconditioner to alleviate stiffness constraints. Additionally, due to the complexity of the coupling between chemistry, fluid, and transport phenomena, splitting is often used to simplify the time integration of the model equations. Splitting, in turn, reduces the accuracy of the approximation. While constructing a preconditioner for a portion of the source terms, such as chemical reactions, is feasible, this task becomes more complicated if one considers the full source term of the equations.

Exponential integrators have recently emerged as an efficient alternative to implicit methods for solving large-scale stiff systems, particularly when no effective preconditioner is available. In this thesis, we explore whether exponential methods

can be used for combustion simulations and study the computational advantages of such schemes.

We first explore how to tackle the stiff nature of chemical kinetics in a model that forgoes transport phenomena to isolate the chemical kinetics terms and forms the core of various more complex combustion models. A novel time adaptive exponential integrator is presented and then applied to this zero-dimensional (Zero-D) combustion problem. We demonstrate that the new method can perform comparably to well-established implicit-Krylov time integration methods. We study the performance of the exponential integration methods and demonstrate how they are affected by the spectrum of the problem.

We then extend the chemical kinetics core to include transport phenomena and develop an exponential integration-based numerical approach to the propagating flame front model in one dimension. In addition to the embedded homogeneous reactor problem, advection and diffusion terms are added, and continuity is considered, which generates a highly coupled system of PDEs. In these problems, operator splitting is typically used to separate the stiff chemical reaction source terms from the much slower transport phenomena. Effectively, this reduces the equations to a Zero-D problem and a transport problem, which needs to be solved at each time step. While effective, this introduces unfavorable splitting errors; we demonstrate that this split can be avoided when using an exponential time integration scheme. We study the performance of the new exponential integration approach for the model using several different chemical mechanisms and compare the performance of the new integrators to the state-of-the-art NGA code, which uses implicit methods.

Our computational models and numerical experiments indicate that exponential methods offer a promising approach to modeling combustion and highlight directions for further studies that can help to develop more efficient time integrators for computational combustion problems.

Chapter 1

Introduction

Many dynamic processes in science and engineering are modeled as initial value problems for either systems of ordinary (ODEs) or partial differential equations (PDEs). With the method of lines and an appropriate spatial discretization, PDEs may be rewritten as a system of ODEs. However, analytic solutions to these commonly large-scale, nonlinear systems of ODEs are infeasible, and the primary tools to solve these systems are numerical simulations. This, together with the growing demand for more efficient and accurate numerical algorithms and techniques, demonstrates the increasingly important role of scientific computing in science and engineering.

Combustion technology is ubiquitous in the transportation and energy industries. The common goal of these fields is to leverage the properties of controlled exothermic chemical reactions to produce hot gases that are then expanded or used for heating. Some practical applications are jet engines, which provide means of propulsion for airplanes; internal combustion engines (ICEs) in vehicles; and rockets for space exploration and satellite delivery. Before numerical simulations, the only way to study this type of chemistry was through experiments, theory, and creating reduced models of chemistry that would yield analytic solutions. Numerical simulations allow for the use of realistic, complex, high-fidelity models that couple evolution equations together in non-trivial geometries for zero, one, two, or three spatial dimensions. For example, airplane jet engines, with their complex sets of moving parts interacting with hot combusting jet fuel, can be simulated with high

fidelity, allowing access to data and information about these systems that would otherwise not be available. Thus, computer simulations significantly speed up the research and development of new combustion technologies.

The combustion process involves a complex series of branching chemical reactions occurring over a wide range of temporal scales. The broad spectrum of scales manifests in the wide spread of eigenvalues of the Jacobian derived from the chemical source terms appearing in the mass conservation equations. A majority of these eigenvalues are negative, but positive eigenvalues associated with the explosive behavior of the system may also be present once combustion commences [1, 2, 3]. Numerical simulation of combustion is complicated by the stiffness of the combustion models, characterized by the large ratio of the smallest and largest magnitude eigenvalues.

Combustion is modeled by a system of PDEs, which couples chemical kinetics with both thermodynamic processes and fluid transport. In order to address the complexity introduced by this highly nonlinear coupling, splitting methods are often leveraged to break the larger problem into a series of more manageable subsystems solved one after the other. A common splitting strategy is to integrate the chemical reaction terms separately from the transport phenomena [4]. This integration is typically handled implicitly due to the problem's stiffness, commonly with one of the Backward Differentiation Formula (BDF) methods [5, 6]. A number of scientific computing software packages, such as the Sundials suite [7], implement such algorithms. If an efficient preconditioner is chosen, the BDF methods are computationally quick when applied to spatially homogeneous chemical kinetics problems with large numbers of species [8, 9].

Additionally, splitting introduces a significant numerical error in the solution. It may be desirable to avoid splitting and integrating the forcing terms that include both the chemistry and transport terms together. It is reasonable to ask whether it is possible to develop a more efficient time integrator that can allow for large time step sizes akin to implicit methods while still developing computational savings. Over the past decades, exponential time integrators, which work well and often outperform implicit methods for large-scale stiff systems, have been developed

[10, 11, 12, 13]. In this thesis, we explore the applicability of exponential schemes to combustion problems and demonstrate that the exponential integration approach is indeed promising in this field.

The core concept of exponential time integrators is to express the approximate solution in terms of matrix-vector products of exponential and exponential-like functions of the Jacobian or its estimate. Exponential schemes result from choosing efficient algorithms to estimate these products in conjunction with an appropriate quadrature rule. Similar to a rational function of the Jacobian used in an implicit method, exponentials of this matrix allow integration with a time step size that far exceeds the stability constraints suffered by explicit integrators. Simultaneously to the introduction of Krylov-type algorithms [14], which compute matrix-vector products for exponential or exponential-like matrix functions, the computational savings delivered by these computations became clear when compared to using the iterative algorithms that estimate rational functions of a matrix. Over the past several decades, many classes of exponential methods have been constructed with these ideas, and they demonstrably work well for several applications [10, 11, 12, 13]. In particular, Exponential Propagation Iterative (EPI) methods have been proposed for general large-scale stiff systems of differential equations [13, 10], and when applied, provide computational savings compared to commonly used implicit and explicit integrators for problems from plasma physics [15], fluid dynamics [16, 17] and other fields [18]. Combining EPI methods with the KIOPS algorithm to compute linear combinations of exponential-like φ_k functions of matrices with vectors delivered efficient schemes for problems where constant time stepping is used [19]. These results inspire our investigation into whether similar exponential methods can be effectively applied to problems in computational combustion. The main focus of this thesis is to employ ideas and advances in exponential integration and construct methods that can deliver computational savings for combustion modeling.

The thesis is organized as follows. In Chapter 2, we describe the set of combustion problems that will serve as a test suite for our new methodology. We present a detailed description of these problems, discuss the numerical challenges

they present, and describe the practical implementation of these systems. Also, the overall construction of an exponential integrator and the numerical linear algebra algorithms needed to evaluate the matrix functions-vector products as part of the exponential schemes are presented.

We describe two approaches to applying exponential integration to combustion models. The first is found in Chapter 3, where we consider the so-called Zero-D problem. We isolate the chemistry and explore how exponential integration performs for homogeneous reactor problems. Such an application of the exponential integrators could potentially be implemented within a splitting approach by speeding up simulations of the chemical kinetics integration step. A novel third-order in time exponential time integrator is presented and demonstrated to perform comparably to implicit methods.

In Chapter 4, we consider the second approach, which integrates a more complex problem that couples chemical kinetics and transport processes in one dimension. We formulate a 1D propagating flame front problem and study the performance of exponential methods in application to this model. We compare exponential schemes with a split implicit-explicit method [5] and discuss the relative performance of these two numerical schemes and the advantages of the new approach.

Finally, in Chapter 5, we summarize the results, discuss the broader implications of the work, and outline research directions for future work and development.

Chapter 2

Preliminaries

This chapter will discuss the preliminary material that provides the necessary context and information for the subsequent chapters. We describe the numerical characteristics of computational combustion modeling that motivate this work. In addition, we provide a brief historical overview of exponential integrators and describe the numerical algorithms we used for combustion simulations.

2.1 The simulation of reactive flows

Mathematical models of chemically reactive flows, such as those encountered in combustion applications, consist of a set of conservation equations for mass, momentum, and energy, which take the form of a system of coupled non-linear partial differential equations (PDEs) for unsteady and spatially inhomogeneous scalar and vector fields. Within the confines of the ideal gas model, the kinematic and thermo-chemical states of the reactive gas are uniquely identified by the density of each of the molecules and atoms in the gaseous mixture, the mass-averaged velocity of the gas, and its pressure and temperature (or alternatively its specific enthalpy or internal energy). Thus, coupled and non-linear PDEs for each field that appears in the state vector must be solved.

The PDEs are complemented by so-called *closures*, such as the equation of state for a mixture of ideal gases, a constitutive law for the stress tensor (typically the one for Newtonian fluids), and algebraic expressions for transport properties (e.g.,

viscosity, thermal conductivity, and species diffusion coefficients) and chemical reaction rates.

As a matter of example, each of the governing equations for the density of species i of M reads

$$\frac{\partial \rho_i}{\partial t} + \nabla \cdot (\rho_i \boldsymbol{u}) = \nabla \cdot (\rho \mathcal{D}_i \nabla Y_i) + \rho \omega_i, \quad i = 1, \ldots, M, \tag{2.1}$$

where ρ_i is the density of the species, t is time, $\boldsymbol{u}$ is the fluid velocity, ρ is the mixture density, $\mathcal{D}_i$ the species mass diffusion coefficient, Y_i its mass fraction (i.e., the mass of species i per unit mass of the gaseous mixture), and ω_i the rate of production of mass of species i per unit mass of the mixture. In the order in which they appear, the terms on the left-hand-side are: unsteady and convective terms. The terms on the right-hand-side are the diffusive and chemical source terms.

In most combustion applications, the fuel consists of hundreds of individual species [20, 21, 22, 23] (often hydrocarbons, but recently also carbon-free fuels such as hydrogen and ammonia), which undergo thousands of chemical reactions as they oxidize to products (e.g., carbon dioxide and water) through complex chemical pathways involving oxygen and many intermediate species and radicals [23, 24]. Accurate modeling of the heat release rate associated with fuel oxidation and pollutant emissions such as oxides of nitrogen requires detailed chemical models, which feature extensive networks of chemical reactions and intermediate species [9, 20].

Such detailed combustion mechanisms have important computational implications since they result in a large number of chemical species and chemical reactions. Since the inclusion of a species implies that a PDE for its density field must be solved, it is apparent how detailed combustion mechanisms result in large systems of PDEs. Combustion mechanisms featuring $\mathcal{O}(100-1000)$ species are not unusual when simulating combustion in realistic configurations. However, most simulations attempt to reduce the number of species by adopting reduced order combustion models [21, 25]. In fact, recent trends in combustion mechanisms show a persisting trend whereby the number of species has increased with time so that the largest combustion mechanisms now feature $\mathcal{O}(10^4)$ species [23].

At times, reduction of the size of the combustion mechanism is not advisable

due to the chemical complexity of large hydrocarbon fuel molecules, which requires that many intermediate species be included in the quantitative description of its oxidation [23, 26, 27]. Such is the case for mechanisms tasked with modeling quantitatively the combustion of so-called transportation fuels such as gasoline [28], diesel [29], and jet engine fuels [24, 30], including the newly proposed sustainable aviation fuels [31] (SAF) as those fuels feature hundreds of components and significant chemical complexity. Furthermore, the requirement that the combustion mechanism be able to predict emissions of harmful pollutants such as nitrous oxides, polycyclic aromatic hydrocarbons (PAH), and soot (i.e., carbon particulate matter) brings about additional complexity, increasing the dimensionality of the models.

Beyond the obvious challenge of solving for $\mathcal{O}(100 - 1000)$ PDEs in addition to those for the momentum and energy of the reactive mixture, the network of species and chemical reactions introduces significant stiffness to the system. By *stiffness*, we mean that the Jacobian, defined as the derivative of the chemical source terms with respect to the thermo-chemical state (i.e., species densities and temperature for isobaric applications) has a large spectrum [4]. In other words, the eigenvalues of the chemical Jacobian are broadly distributed. In particular, the spectrum features real negative eigenvalues of large magnitude that represent time scales that are much smaller than those of interest. It is then clear that explicit time integration methods are computationally inefficient, while implicit methods are preferable for the simulation of chemical kinetics as widely accepted [5, 6, 32]. The chemical Jacobian typically features positive real eigenvalues of relatively small magnitude as well, which represent so-called *explosive modes*. Those positive eigenvalues are associated with sudden ignition and reflect the significant nonlinearity of the chemical kinetics model as explosive modes *exhaust* or disappear once the gaseous system reaches equilibrium.

Once large and stiff chemical kinetics mechanisms are adopted for the simulation of unsteady and spatially inhomogeneous reactive flow configurations, discretization of the governing PDEs results in very large systems with the vector of unknowns being of high dimensionality. As a matter of example, consider the

simulation of a laminar, steady, non-premixed coflow flame, which is a canonical problem in numerical combustion. Such a flame features a fast moving central jet of fuel issuing into a slowly moving coflow of oxidizing air, so that a diffusion (or non-premixed) flame forms at the interface between fuel and air. Finite difference mesh discretizations for axisymmetric coflow flames (cylindrical coordinate systems with radial and axial coordinates are common) typically feature about $N = 10^5$ grid points. If a combustion mechanism featuring M species is considered, the number of unknowns is $(M + 5) \times N$, since one solves for the three components of velocity, hydrodynamic pressure, and an energy variable, e.g., temperature, in addition to the M species density equations (see Eq. (2.1)). Even considering a rather "small" mechanism with $M = 10$ species, the problem features $(M + 5) \times N \approx 15 \times 10^5 \sim 1.5$ M unknowns.

The challenge is exacerbated further in the case of turbulent flows, which bring about very stringent spatial and temporal resolution requirements to resolve all pertinent scales of turbulence, which is broadly distributed with a wide dynamic range [4]. For simulations of turbulent reactive flows, which require unsteady simulations in three dimensions, the number of grid points increases quickly upwards of 100 M for the lowest and most modest Reynolds numbers, with finite difference meshes featuring $\mathcal{O}(10^9 - 10^{10})$ grid points being state of the art at the time of this writing.

The governing equations contain terms that represent physical processes such as convection, diffusion, and reactions; numerical methods tailored to the time-integration of those terms have been developed and are widely used [5, 6]. For example, variable density reactive flows at low speeds, i.e., flows for which the magnitude of the velocity of the fluid is smaller than the speed of sound, are modeled within the *low Mach number* restriction of the Navier-Stokes equations [33]. Within the framework of this simplification, density is assumed to be solely a function of temperature, composition, and constant background pressure, but not of the spatially inhomogeneous hydrodynamic pressure appearing in the momentum equation. Such simplification is critical to computational efficiency since it removes acoustic waves from the solution, thereby improving time step size restrictions for

explicit time integration. Yet, coupling between reactive scalars, density, and momentum persists as fluid regions with intense exothermic chemical reactions are characterized by large values of the velocity divergence and such non-linear coupling between the momentum and thermo-chemical state of the gaseous mixture requires specialized algorithms and methods. Specifically, the low Mach number restriction of the Navier-Stokes equations is often solved with fractional step methods that advance momentum first, followed by the species densities and the energy variable, and require the solution of a Poisson equation for the hydrodynamic pressure at each time step [34, 35]. Segregation of the momentum equation from those of the species densities and temperature allows replacing one large problem with several smaller ones.

The temporal advancement of the governing equations for the species mass densities and the energy variable (the so-called *reactive scalars*) is also challenging since the discrete approximations to the convective and diffusive terms bring coupling across nearby grid points for each reactive scalar, while chemical sources bring coupling across all scalars for each grid point. Such coupling is problematic for implicit time-integration methods because the overall size of the problem is equal to $(M + 1) \times N$, where M is the number of species and N is the number of grid points. As discussed, fully implicit methods applied to the simulations of a modest axisymmetric flame with $M = 10$ and $N = 10^5$ require the solution to a system of 1 M (non-linear) algebraic equations in 1 M unknowns at each time step. Because of such computational challenge, most reactive flow solvers split the time-integration of transport (i.e., convective and diffusive terms) from that of reactions (i.e., chemical source terms) via *operator splitting* [34, 35]. By splitting the terms on the right-hand side of the system of equations, methods that are most appropriate to each task are employed. For example, the chemical source terms are typically integrated with fully implicit Backward Differentiation Formulas (BDF) methods implemented in available packages such as SUNDIALS. At each step, N such integrations are performed in the form of initial value problems (IVPs), i.e. one independent IVP per grid point.

While computationally affordable and efficient, operator splitting does intro-

duce splitting errors [36], which limit the size of the time steps that can be taken prior to compromising accuracy. Depending on the splitting method, the temporal order of convergence might degrade easily to first order, despite the order of the methods used for the integration of the segregated terms being second order. Consequently, implicit time-integration methods that avoid splitting errors while retaining performance and accuracy for large problems are highly desirable.

Semi-implicit iterative methods blend implicit, explicit, and sometimes splitting techniques to efficiently integrate the coupled system of reacting flow equations. These methods can recover the stability properties of implicit methods by iteration over each time step [36, 37, 38, 39].

These methods are varied, but they share a few common elements. First, they employ temporal staggering, where the *scalar* quantities of species mass fractions, pressure, density, and energy are stored at a half time step $t_{n+\frac{1}{2}}$ and the *vector* velocity is stored at the whole time step t_n. For example, the popular method of Savard et al. [36] advances the variables in the following order to construct an iterate. First, the scalar variables are advanced, possibly with a decoupling between the transport and chemistry source terms with different solvers, then the density is updated, and then finally, the velocity with a predictor-corrector method to enforce continuity. Multiple iterates of this procedure may be necessary for accuracy and, when sufficiently converged, will ensure the true overall temporal order of convergence is preserved despite the decoupling [36, 37, 38, 39].

Semi-implicit iterative methods also posses *monolithic* approaches which aim to advance the *scalar* variables without the commonly used decoupling between the transport and chemistry source terms [37]. This is accomplished by factoring the scalar Jacobian source term into the exact chemical and second-order approximation of the transport source term. This practice was found to be computationally efficient compared to splitting strategies and allowed for generally larger overall time steps [37]. However, in order to maintain the overall second order of convergence with a time step of Δt, the residuals on the sub-iterates must be less than $\mathcal{O}(\Delta t^2)$. Thus, a balancing act between the costly sub-iterating process, the desire for the higher temporal order of convergence, and accuracy must be made.

2.1.1 Summary of challenges

In summary, the challenges associated with the numerical simulation of chemically reactive flows, such as those pertinent to combustion applications, are as follows.

1. **Problem size.** Once discretized on a computational mesh, the number of unknowns increases linearly with the number of chemical species included in the kinetics mechanism and linearly with the number of grid points. This is especially problematic for detailed combustion mechanisms featuring a large number of species and for simulations of turbulent reactive flows, which require governing equations in three-dimensions.

2. **Chemical stiffness.** Kinetics mechanisms for combustion applications feature significant stiffness in that the Jacobian associated with the chemical source terms has a broad spectrum with real negative eigenvalues of large magnitude. Such occurrences make explicit time integration methods problematic and inefficient since the time scales of interest are much longer than those associated with the largest negative eigenvalues. While desirable for accuracy and stability, the monolithic implicit time-integration of the semi-discrete form of the PDEs requires addressing the large problem size of the fully coupled system of equations with efficient and robust methods.

3. **Non-linearity of the equations.** The mathematical models for simulating unsteady and spatially inhomogeneous chemically reactive flows are non-linear. Important non-linearities originate from the convective terms in the governing transport equations, the product of transport coefficients (which are themselves non-linear functions of the state), and the fields and their gradients. Additional critical non-linearities are present in the functional form of the chemical source terms, which are proportional to the exponential of the mixture temperature (the so-called Arrhenius rate coefficients). Somewhat more benign non-linearities in the chemical source terms pertain to the product of the concentrations of the chemical species. The presence of non-linear

terms in the mathematical models complicates the task of time integration with implicit methods, which require the solution of large non-linear systems of algebraic equations at each time step.

4. **Density-velocity coupling and splitting errors.** In the low-Mach number limit of the reactive Navier-Stokes equations, velocity is coupled to the local thermo-chemical state of the mixture through the density of the gaseous mixture and its transport properties, which depend on the species concentrations and temperature. Such coupling is usually addressed by the fractional step method, which segregates the time integration of the momentum equation from that of the reactive scalars, i.e., species densities and energy variable – either the temperature or the enthalpy of the mixture. Fractional step methods do require the solution of a Poisson equation for the hydrodynamic pressure, which can be computationally demanding. Moreover, the reactive scalar fields are coupled locally to each other by chemical reactions and across grid points by gradient operators. While operator splitting methods are commonly used to reduce the size of the problems, they introduce temporal errors that might easily degrade temporal convergence to first order.

Given these challenges, the questions we ask in this work are whether it is possible to develop numerical time integrators that (i) allow for more efficient integration of the chemical kinetics portion of the problem and/or (ii) enable a *monolithic* update of the scalar variables without the use of semi-implicit iterative methods that will (iii) perform comparably to the state-of-the-art. In the subsequent chapters, we consider these two challenges and propose approaches to temporal integration based on exponential propagation ideas, as described below.

2.2 Exponential Time Integrators

Over the past several decades, exponential integrators emerged as an efficient alternative to explicit and implicit methods in solving initial value problems for

certain classes of large-scale stiff systems of ODEs, such as

$$\frac{dy}{dt} = f(y(t)), \quad y(t_0) = y_0. \tag{2.2}$$

Original papers introducing exponential integrators date back to the 1960's [40]. Initially, exponential schemes have not been widely adopted for large-scale problems since the algorithms to approximate products of exponential-like functions of matrices and vectors were not sufficiently efficient [41]. However, with the advent of Krylov-projection-based methods to approximate these products [14], the efficiency of the exponential methods compared to implicit techniques became notable and spurred more research into the construction of schemes of this type [12, 13].

The main idea behind exponential methods is that the solution of a given IVP can be expressed in terms of exponential-like functions of a Jacobian matrix. Let $t_n = t_0 + nh$ be a discretization of the time interval over which the solution has to be computed, and $y_n = y(t_n)$ be a solution of the IVP at time t_n. Using the Taylor expansion, the right-hand-side of (2.2) around the time t_n, the problem can be written as

$$y' = f(y) = f(y_n) + f'(y_n)(y - y_n) + R(y), \tag{2.3}$$

where $R(y) = f(y) - f(y_n) - f'(y_n)(y - y_n)$ is the nonlinear remainder function. Defining the Jacobian of $f(y)$ evaluated at time t_n as $J_n = f'(y_n)$ and using an integrating factor $e^{-J_n t}$ equation (2.3) can be written in integral form as

$$y(t_n + h) = y_n + (hJ_n)^{-1}\big(e^{J_n h} - 1\big)hf_n + \int_{t_n}^{t_n+h} e^{J_n(t_n+h-t)}R(y(t))dt. \tag{2.4}$$

Changing the integration variable t to s with $t = t_n + sh$ and defining $\varphi_1(z) = \frac{e^z-1}{z}$ (2.4) can be written as

$$y(t_n + h) = y_n + \varphi_1(hJ_n)hf_n + \int_0^1 e^{J_n h(1-s)}hR(y(t_n + sh))ds. \tag{2.5}$$

Equation (2.5) usually serves as a starting point for the derivation of an exponential integrator. To construct a particular exponential method, one has to develop an approximation of the nonlinear integral in (2.5) and to choose an algorithm to estimate the product of an exponential-like function φ_1 of a matrix and a vector.

Higher order exponential integrators are typically constructed by approximating the nonlinear remainder $R(y(t_n + sh))$ with a polynomial in s, which results in the approximate solution $y(t_n + h)$ being expressed as a linear combination of φ-functions defined by

$$\varphi_0(z) = e^z, \quad \varphi_1(z) = \frac{e^z - 1}{z}, \quad \varphi_2(z) = \frac{e^z - z - 1}{z^2}, \tag{2.6}$$

and, in general,

$$\varphi_k(z) = \int_0^1 e^{z(1-\theta)} \frac{\theta^{k-1}}{(k-1)!} d\theta. \tag{2.7}$$

Over the past several decades, a number of exponential integrators have been constructed, analyzed, and used [10, 11, 16, 15, 17, 18, 19]. Based on the literature review and our experience with using exponential integrators for various applications including combustion [18, 42, 43] in this work, we focus on exploring the use of Exponential Propagation Iterative (EPI) schemes with Krylov-projection-based estimators for products exponential matrix functions and vectors for combustion modeling.

EPI methods have been specifically designed to reduce the overall computational cost of an exponential integrator for large-scale stiff problems, particularly when used with Krylov-projection based algorithms. Krylov-projection algorithms were first proposed as a method to estimate a general function of a matrix-vector product by Van der Vorst [14] and since then were developed further and shown to be the most efficient general method to estimate products of type $F(A)v$ where A is a large stiff matrix. The basic idea of a Krylov algorithm is to compute an estimate of $F(A)v$ using projections onto the Krylov subspace

$$K_m = \text{span}\Big\{v, Av, ..., A^{m-1}b\Big\}. \tag{2.8}$$

Since the Krylov vectors $A^k v$ do not constitute a well-conditioned basis of the Krylov subspace, an Arnoldi iteration is typically used to construct an orthonormal basis $v_1, v_2, ..., v_m$ which spans K_m. If $V_m = [v_1 v_2, ..., v_m]$ has orthonormal vectors v_k as columns, then matrix $P = V_m V_m^T$ is a projector onto the Krylov space. Arnoldi orthogonalization algorithm can be written in a matrix form as

$$AV_m = V_m H_m + h_{m+1,m} v_{m+1} e_m^T \tag{2.9}$$

where e_m the mth unit vector in $\mathbf{R}^m$ and H_m is an upper Hessenberg matrix which can be calculated using the orthogonality of vectors v_k by $H_m = V_m^T A V_m$. Approximating $F(A)v$ is then accomplished with the following estimates:

$$F(A)v \approx V_m V_m^T F(A) V_m V_m^T v \approx V_m F(H_m)||v||_2 e_1. \tag{2.10}$$

Since H_m is then a small $m \times m$ matrix $F(H_m)$ is calculated using Pade approximation with scaling and squaring procedure [44].

This classic version of the Krylov projection algorithm has been further refined in a series of publications [45, 46, 47] to improve its efficiency with the use of substepping and algorithms to estimate the parameters of the algorithm. As a result, the KIOPS algorithm has been developed and shown to be very efficient for a wide range of application problems. KIOPS estimates linear combinations

$$\varphi_0(A)b_0 + \varphi_1(A)b_1 + ... + \varphi_k(A)b_k. \tag{2.11}$$

Using a reformulation of the problem to a single exponential of an augmented matrix and an adaptive substepping procedure. We give a brief overview of KIOPS in section 3.2 and refer the reader to [47] for details of the algorithm.

Chapter 3

Numerical modeling of homogeneous batch reactors.

The simulation of chemically reactive systems is challenging due to their wide range of spatial and temporal scales. Furthermore, transport of mass, momentum, and energy are tightly coupled to chemical reactions at the molecular scale. For many problems in the low Mach number regime, chemical reaction rates are significantly faster than those of transport processes. When systems like these are solved numerically, it is a common practice to use temporal integration methods with Strang splitting, which advances chemistry and transport separately. However, integration of the chemical source terms is difficult because it involves a large number of reactions occurring with widely-ranging reaction rates. In other words, while Strang splitting addresses the global stiffness of the problem, integration of the chemistry is still stiff. The development of efficient time integrators for the chemical source terms appearing in the transport equations for the species concentrations is a critical task in computational combustion. In this paper, we investigate whether exponential time integration, which has proven efficient in other fields, offers advantages over more established approaches for chemical kinetics problems in combustion.

The stiffness of systems of ordinary differential equations that describe the evolution of reactive species and temperature in a homogeneous (Zero-D) reactor makes explicit time integration methods impractical since stability constraints on

time-step sizes are too severe. Instead, implicit schemes are typically used. Commonly employed methods include backward differentiation formulas (BDF)-based integrators [4, 5, 6, 48]. These algorithms are typically used in conjunction with a modified Newton solver and Krylov-projection-type iterative methods to solve its embedded linear systems [5, 6]. Whether the performance of such implicit methods is satisfactory or not is often predicated on whether an efficient preconditioner can be constructed to accelerate the linear solves. Also, the functional form of the chemical source terms is complex (these source terms and their Jacobians are most often evaluated by software packages such as TCHEM [49], Cantera [50], or Chemkin[51]), so it is often quite challenging to construct an effective preconditioner [4, 8], particularly one that is general enough to be effective across chemical mechanisms. Recently, exponential methods emerged as an efficient alternative to implicit integrators for problems for which an effective preconditioner is not available [10, 13, 32].

Herein, we apply a new time-adaptive exponential integrator to the simulation of the temporal evolution of chemically reactive and spatially homogeneous systems, i.e. chemical reactors that are described by a system of ordinary differential equations (ODEs). It is found that the novel exponential time integrator will accurately resolve all three chosen chemical mechanisms. Chapter 3 is organized as follows. Section 3.1 describes the governing differential equations. In Section 3.2, we discuss the time integration method. Section 4.3 presents the results and includes a discussion of the comparative performance of legacy implicit and novel exponential methods. The last Section 4.4 outlines the conclusions of our study and future directions.

3.1 Governing equations

We consider a spatially homogeneous chemically reactive system consisting of an ideal gaseous mixture undergoing chemical reactions at constant pressure. The mass fractions of each chemical species and temperature uniquely identify the thermo-chemical state of the mixture.

Chemical species react with each other according to several reactions. For reaction j, the forward reaction rate constant is given in Arrhenius form by

$$f_j = A_j T^{\alpha_j} \exp\Big(\frac{-E_j}{RT}\Big), \tag{3.1}$$

where A_j, and α_j are pre-exponential and exponential constants, R is the universal gas constant, and E_j is the activation energy. If reaction j is reversible, then the backwards reaction constant, b_j, is not zero and is given directly in Arrhenius form (3.1) or leveraging the equilibrium constant K_j:

$$b_j = f_j K_j. \tag{3.2}$$

The net rate of reaction, representing the number of times that the reaction occurs in the forward direction per unit time per unit volume, is the difference between forward and backward rate constants multiplied by the molar concentration of the species participating in the reaction raised to their stoichiometric coefficients, i.e.

$$\mathcal{R}_j = f_j \prod_{i=1}^{\mathbf{K}} \chi_i^{v'_{ji}} - b_j \prod_{i=1}^{\mathbf{K}} \chi_i^{v''_{ji}}, \tag{3.3}$$

where v'_{ji} and v''_{ji} are the reactant and product stoichiometric coefficients respectively for species i in reaction j, and $\mathbf{K}$ is the number of chemical species.

Production or loss of a species because of reaction is equal to the difference between the forward and backward stoichiometric coefficients times the rate of reaction. We denote the number of reactions as $\mathcal{N}$, and the total amount of species produced is found by summing over all reactions

$$\dot{\omega}_i = \sum_{j=1}^{\mathcal{N}} (v''_{ji} - v'_{ji}) \mathcal{R}_j. \tag{3.4}$$

The rate of change in temperature is given by the total volumetric heat production divided by the thermal capacity of the mixture per unit volume. The negative sign in equation (3.5a) exists because energy is understood with respect to the chemical bonds, not the gas; if these bonds lose energy, the gas' energy increases. The change in the species mass fractions with respect to time is the net rate of production normalized to mass fractions. The resulting system of ODEs models the evolution of the temperature and species mass fractions:

$$\frac{dT}{dt} = -\frac{1}{\rho c_p}\sum_{k=1}^{\mathcal{N}} \dot{\omega}_k H_k W_k, \tag{3.5a}$$

$$\frac{dY_i}{dt} = \frac{1}{\rho}\dot{\omega}_i W_i, \quad i = 1, ..., \mathbf{K}. \tag{3.5b}$$

The above system of equations includes the gas density, ρ, the heat capacity at a constant pressure, c_p, the molar rate of production, $\dot{\omega}_k$, the specific enthalpy, H_k, and the species' molar mass, W_k. The system of equations requires a closure for density ρ according to the equation of state for an ideal gas.

Equations (3.5a-3.5b) is a general model for a spatially homogeneous isobaric reactive mixture. To finalize the model a specific list of species and reaction parameters must be provided. These are defined through carefully assembled kinetic mechanisms that describe the species present in the gas solution, the chemical reactions that occur, and the species' thermodynamic properties. The resulting model determines the evolution of the specific gas.

We will use gases consisting of hydrocarbon fuels (C_xH_y) and oxygen (O_2), with additional species such nitrogen (N_2) and argon (Ar) accounting for most of the mass [24, 25, 28].

In order to model the complexity of chemical reactions and the numerous branching pathways typical of large hydrocarbon oxidation, a large number of species and reactions are required. We study three hydrocarbon-based kinetic mechanisms: GRI3.0 [52], *n*-butane [53], and *n*-dodecane [54], which model the combustion of methane, butane, and dodecane, respectively. Methane, CH_4, while the smallest hydrocarbon we study, is found in a wide variety of fuels and is the main component of natural gas. The methane mechanism is detailed and contains 53 species and 325 reactions. The second mechanism is for butane, C_4H_{10}, and contains 154 species and has 680 reactions. Butane behaves similarly to more complex practical fuels [53] and is a component of gasoline [28]. The last mechanism is *n*-dodecane, which is the largest hydrocarbon, $C_{12}H_{26}$, we study. It is a component of kerosene and some jet fuels [55]; it contains 105 species and 420 reactions.

3.2 Methods & Implementation

For ease of notation, we organize the thermo-chemical state variables in a vector ordered with temperature followed by species in the same order as they appear in the chemical mechanism with N_s being the number of species:

$$y(t) = [T(t), \;\; Y_1(t), \;\; \ldots \;\; , \;\; Y_{N_s}(t)]^T. \tag{3.6}$$

Denoting the initial gas state of the mixture $y(t_0) = y_0$, one obtains the initial value problem

$$\frac{dy(t)}{dt} = F(y(t)), \tag{3.7a}$$

$$y(t_0) = y_0. \tag{3.7b}$$

Time is discretized as $[t_0, t_1, \ldots, t_m]$, where $t_{i+1} = t_i + h_i$, and h_i is the time step size. Approximations of the state, right-hand-side function and Jacobian at time t_n are denoted as $y_n \approx y(t_n)$, $F_n \approx F(y(t_n))$, $J_n \approx J(y(t_n))$, respectively.

We employ TCHEM [49] in order to compute the chemical source terms and Jacobians. Given the mass fractions, temperature, and gas pressure, TCHEM returns both F_n and J_n. The Jacobian can be computed either via finite differences or analytically. We use the analytical version to avoid the inaccuracies associated with approximations. Moreover, it is well known that computation of the Jacobian via finite differences is inefficient and computationally expensive for large mechanisms [56].

The simulation of combustion processes requires the solution of nonlinear and stiff systems of ordinary differential equations, which can be large in size depending on the chemical mechanism. Exponential propagation iterative methods of Runge–Kutta type (EPIRK) have been shown to perform efficiently for a range of large-scale stiff systems [47], including reaction-diffusion models [10]. Because of the record of success of EPIRK methods and the requirement that the time step size be adaptive in order to simulate ignition problems, we extend the EPIRK framework to a novel time-adaptive third-order EPIRK method with an embedded second-order scheme for error estimation. The details of the derivation of order

conditions and their solution for constructing a particular scheme can be found in [10]. The same approach is used to formulate the following EPIRK integrator, EPI3V:

$$Y_1 = y_n + \varphi_1\Big(\tfrac{3}{4}h_n J_n\Big)h_n F_n, \tag{3.8a}$$

$$R(z) = f(z) - F_n - J_n(z - y_n), \tag{3.8b}$$

$$y_{n+1} = y_n + \varphi_1(h_n J_n)h_n F_n + \varphi_3(h_n J_n)2h_n R(Y_1). \tag{3.8c}$$

The φ-functions are

$$\varphi_1(z) = \frac{e^z - 1}{z}, \quad \varphi_3(z) = \frac{e^z - \frac{1}{2}z^2 - z - 1}{z^3}. \tag{3.9}$$

The method above uses matrix arguments for the φ-functions; computing approximations of the product of exponential-like matrix functions and vectors of type $\varphi_k(A)v$ is the largest computational expense of exponential integrators. Systems of N ordinary differential equations that model realistic physical processes result in large exponential matrices of size $N \times N$ that make the evaluation of the φ-functions prohibitively expensive with traditional approximations like Padé [44] or Taylor expansions. Our EPIRK method was designed to leverage KIOPS, an adaptive Krylov-projection algorithm designed to estimate φ-functions [47]. In KIOPS, an augmented matrix $\tilde{A}$ is used to express the linear combination of φ-functions as:

$$w(\tau) = \sum_{j=0}^{p} \tau^j \varphi_j(\tau A) b_j = e^{\tau \tilde{A}} v. \tag{3.10}$$

A sub-stepping procedure is then employed to estimate the successive products of matrix exponentials and vectors by iteratively letting $\tau = \tau_1 + \tau_2 + ... + \tau_M$:

$$e^{\tau \tilde{A}} v = e^{(\tau_0 + \tau_1 + ... + \tau_M)\tilde{A}} v = e^{\tau_0 \tilde{A}} e^{\tau_1 \tilde{A}} ... e^{\tau_M \tilde{A}} v. \tag{3.11}$$

Each product $e^{(\tau_l \tilde{A})} v_{\tau_l}$ is approximated with a Krylov projection in the KIOPS algorithm. With the length of Krylov basis being m, V an $N \times m$ matrix with Krylov basis vectors v_{τ_i} as its columns, and H an $m \times m$ matrix so that $H_{ij} = (\tilde{A} v_{\tau_{r,j}})^T v_{\tau_{r,i}}$ we have the projection

$$e^{\tau_i \tilde{A}} v_{\tau_i} \approx V e^{\tau_i H} V^T v_{\tau_i}, \tag{3.12}$$

where $e^{\tau_l H}$ is approximated using Padé with a squaring and scaling algorithm [44].

We use the Exponential Propagation Integrators Collection (EPIC) C++ package [57], which includes implementation of EPIRK methods and KIOPS and allows for easy implementation of new methods. EPIC provides a linear combination of products of φ-functions and vectors or a single product estimated at various scalar multiples of the φ-function's arguments. In order to obtain a linear combination of φ-function vector products, the user provides a matrix A and vectors b_i. EPIC then uses KIOPS to approximate

$$\varphi_0(A)b_0 + \varphi_1(A)b_1 + ... + \varphi_p(A)b_p. \tag{3.13}$$

The user can also provide a single b_i and a set of intermediate time points

$$[T_1, \ldots, T_M], \quad T_{j+1} > T_j, \quad T_j \in (0, 1). \tag{3.14}$$

The KIOPS algorithm allows the time integrator to stop at each T_j and save the values:

$$\varphi_i(T_1 A)b_i, \quad \varphi_i(T_2 A)b_i, \quad \ldots \quad , \quad \varphi_i(T_M A)b_i, \quad \varphi_i(A)b_i. \tag{3.15}$$

We accomplish all φ-function approximations with two calls to KIOPS. The first call estimates both

$$\varphi_1\left(\tfrac{3}{4}h_n J_n\right)h_n F_n, \text{ and } \varphi_1(hJ_n)h_n F_n. \tag{3.16}$$

The second call estimates

$$\varphi_3(h_n J_n)2h_n R(Y_1). \tag{3.17}$$

Two separate calls are necessary in order to obtain the local truncation error in support of the adaptive time step size selection algorithm. The lower order exponential Euler method is:

$$y_{n+1} = y_n + h_n \varphi_1(hJ_n)F_n. \tag{3.18}$$

Thus, by subtracting the right hand side of equation (3.18) from the right hand side of equation (3.8c), we obtain a local truncation error estimate:

$$\varphi_3(h_n J_n)2h_n R(Y_1). \tag{3.19}$$

This quantity is obtained by the second call to the KIOPS algorithm (3.17).

We implemented a standard adaptive controller from Wanner et al. [58] to create a time-adaptive method. After a step is calculated, the local truncation error (3.17) is compared with the controller's tolerance. If not within tolerance, and the step is rejected, the time step is adjusted and the process is repeated. Once the tolerance is achieved, the step is accepted, the time step is adjusted, and the iteration proceeds to the next step. We also implement slight modifications due to the specific features of an ignition process. Step sizes change dramatically during the chain-branching phase of ignition and shortly thereafter. The following constraints limit the change between step sizes: h_{old}, h_{new}, and the estimated new step size $\hat{h}$

$$\begin{cases} h_{new} = 2\hat{h}, & \hat{h} > 100 h_{old} \\ h_{new} = \frac{1}{100}\hat{h}, & \hat{h} < 1000 h_{old}. \end{cases} \tag{3.20}$$

Implicit methods have shown to be effective for solving stiff systems of equations arising from modeling homogeneous chemically reacting systems for over thirty years [5, 6]. The CVODE package from Lawrence Livermore National Laboratory is widely used for solving general systems of ODEs. This C++ package implements a variable-coefficient ODE (VODE) solver offering adaptivity of the step size [7]. The user can select linear and non-linear solvers from a list of available options. Because the proposed EPI3V method uses Krylov projection methods, we configure CVODE to allow for the most informative comparison possible. We choose the BDF non-linear solver and the Scaled Preconditioned Generalized Minimized Residual (SPGMR) linear solver. Additionally, we force the Jacobian to be evaluated once each time step and fix the maximum order to 3.

The EPIC package uses NVector data structures from SUNDIALS for vector operations. The current version of EPIC is compatible with SUNDIALS v5.7.

3.3 Numerical experiments

Our novel time integration method is compared to that implemented in CVODE by simulating the ignition of mixtures of air and hydrocarbons. Ignition is a funda-

mental process in combustion, whereby the gaseous species in the mixture undergo accelerating exothermic chemical reactions. As the energy in the chemical bonds of the fuel is converted into sensible enthalpy, it contributes to an increase in the mixture temperature, which in turn leads to an acceleration of the reaction rate. Thus, ignition is characterized by a sudden and abrupt exponential increase in temperature and rate of chemical reactions, which are accompanied by a corresponding depletion of fuel and oxidizer and the formation of combustion products, i.e., water and carbon dioxide. Once either the fuel or oxidizer is exhausted, the mixture reaches an elevated equilibrium temperature and composition, which no longer vary in time.

In this work, we hasten the onset of ignition by setting the initial temperature at or above 1000 Kelvin, which is sufficient to induce the thermal decomposition of molecular oxygen into its O atoms, which commence ignition by attacking the fuel molecule.

For our numerical experiments we ensure that the parameter values for simulations, including temperature, pressure and initial mass fractions, are sufficient for auto-ignition and lean fuel mixtures. The lean mixtures mean that stoichiometrically there is more oxygen than fuel which will ensure the combustion terminates when the fuel is consumed, preventing reactions between products of combustion with the fuel hydrocarbons. Experiments are run through the initial buildup phase into the ignition phase until the steady state is achieved. Table 4.1 shows the temperature, mass fractions and final simulation times for each kinetic mechanism.

Data sets are generated by executing simulations of ignition and storing the state vector at the end of the time integration interval. The same simulations are run repeatedly with different absolute and relative tolerances in order to obtain a set of solutions of increasing accuracy for each kinetic mechanism. The error at the final time is computed with respect to a reference solution generated using CVODE with tight relative and absolute tolerances.

We compare the performance of the EPI3V method against that of CVODE by plotting precision diagrams (CPU time versus a measure of accuracy, here

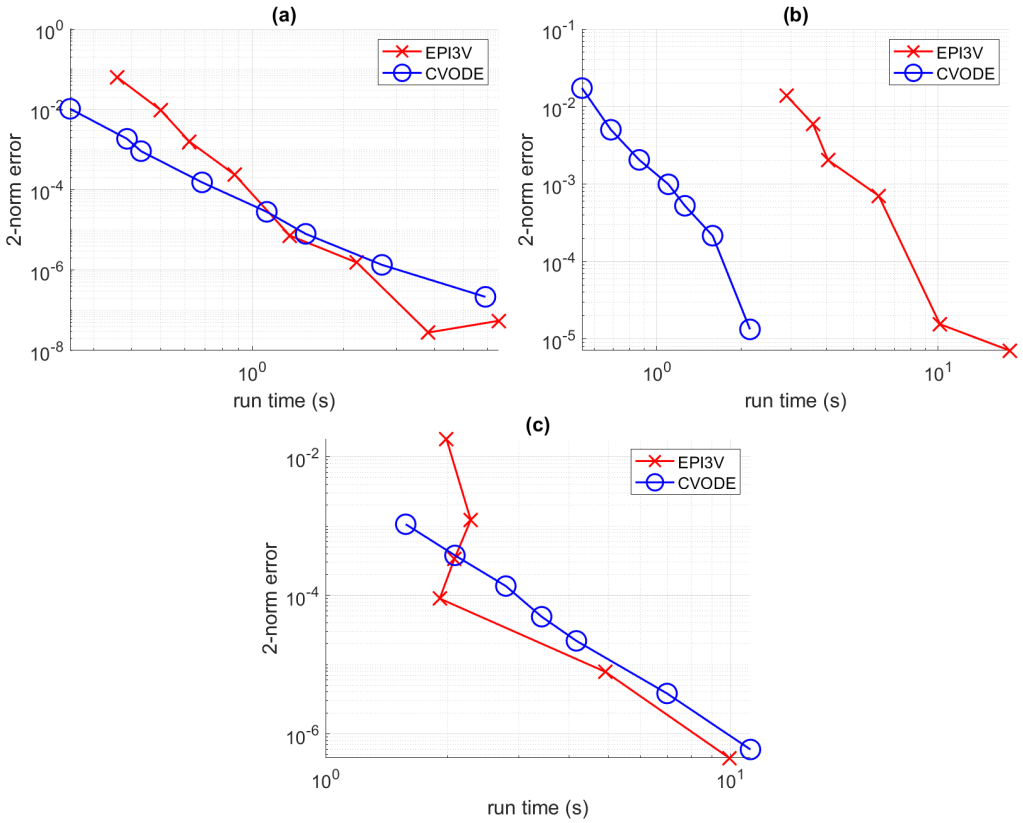

Figure 3.1: Precision diagrams comparing the CPU run time against the 2-norm error of EPI3V versus CVODE. The plots show GRI3.0 (a), *n*-dodecane (b), and *n*-butane (c) respectively.

the 2-norm of the error vector) for both methods in figure 3.1. Tolerances were chosen in order to generate error values for the two methods. Table 3.2 contains the tolerances selected for each numerical experiment. Table 3.3 contains the tolerances which generate each experiment's CVODE reference solution.

For the GRI3.0 mechanism, we see a modest advantage in performance for CVODE if loose tolerances are used. However, at tighter tolerances that yield errors below 10^{-5} the EPI3V method outperforms CVODE. With the butane mechanism, we observe a similar relative performance between the two methods. For loose tolerances, the computational time spent integrating the method is insensitive to accuracy, although the wall-clock time is larger than with CVODE until the error is approximately 10^{-5}. For errors lower than 10^{-5}, the EPI3V method

Table 3.1: Experiment configurations.

GRI 3.0 Mechanism initial values	
Item	Value
Kelvin	1000
CH_4	0.0548
O_2	0.2187
Ar	0.0126
N_2	0.7137

n-butane Mechanism initial values	
Item	Value
Kelvin	1200
O_2	0.2173
C_4H_{10}	0.0607
Ar	0.0125
N_2	0.7092

n-dodecane Mechanism initial values	
Item	Value
Kelvin	1200
O_2	0.2169
$C_{12}H_{26}$	0.0624
N_2	0.7080

Mechanism final times	
Mechanism	Final time (s)
GRI 3.0	1.2
n-butane	$2 \cdot 10^{-3}$
n-dodecane	$5 \cdot 10^{-4}$

becomes slightly faster than CVODE. It is important to note that while CVODE is an established code with decades of optimization, our EPI3V implementation is rather new; both software and algorithmic optimizations are ongoing and improvements are expected. For example, significant computational savings were obtained for EPIRK methods recently as the exponential matrix functions evaluations transitioned from straightforward Krylov projection to adaptive Krylov method *phipm* [46] and later to KIOPS [47]. In fact, in addition to improvements of the methods' parameters new algorithms may be beneficial for approximating exponential matrix functions for select problems. Our third test problem, *n*-dodecane, illustrates the importance of research in this direction.

Unlike the other two mechanisms, simulation of ignition with the *n*-dodecane mechanism presented a challenge for the EPI3V method. Like in the other experiments, we verified the EPI3V method generated the correct solution. However, unlike for the two previous cases, EPI3V was consistently slower than CVODE by an order of magnitude. To explore potential causes we considered the spectrum of the Jacobian matrix for all cases.

Table 3.2: Absolute and relative tolerances used to generate precision diagrams in figure 3.1.

GRI3.0	
Method	(Absolute Tolerance, Relative Tolerance)
EPI3V	$(10^{-10}, 2{\cdot}10^{-2})$, $(10^{-10}, 3{\cdot}10^{-3})$, $(10^{-10}, 5{\cdot}10^{-4})$, $(10^{-10}, 10^{-4})$, $(10^{-10}, 10^{-5})$, $(10^{-11}, 5{\cdot}10^{-6})$, $(10^{-12}, 2{\cdot}10^{-6})$, $(10^{-13}, 8{\cdot}10^{-7})$.
CVODE	$(10^{-7}, 10^{-5})$, $(10^{-8}, 10^{-6})$, $(10^{-8}, 10^{-7})$, $(10^{-9}, 10^{-8})$, $(10^{-10}, 10^{-10})$, $(10^{-10}, 10^{-11})$, $(10^{-11}, 10^{-11})$, $(10^{-11}, 10^{-12})$.

n-butane	
Method	(Absolute Tolerance, Relative Tolerance)
EPI3V	$(10^{-7}, 4 \cdot 10^{-4})$, $(4 \cdot 10^{-8}, 2 \cdot 10^{-4})$, $(2 \cdot 10^{-8}, 10^{-4})$, $(10^{-8}, 10^{-4})$, $(10^{-10}, 10^{-5})$, $(10^{-11}, 10^{-6})$.
CVODE	$(10^{-10}, 10^{-5})$, $(10^{-10}, 10^{-6})$, $(10^{-10}, 10^{-7})$, $(10^{-10}, 10^{-9})$, $(10^{-11}, 10^{-10})$, $(10^{-12}, 10^{-11})$.

n-dodecane	
Method	(Absolute Tolerance, Relative Tolerance)
EPI3V	$(10^{-5}, 5 \cdot 10^{-4})$, $(2 \cdot 10^{-6}, 2 \cdot 10^{-4})$, $(10^{-7}, 5 \cdot 10^{-5})$, $(10^{-8}, 10^{-5})$, $(5 \cdot 10^{-9}, 5 \cdot 10^{-6})$, $(10^{-10}, 10^{-6})$, $(10^{-11}, 10^{-7})$.
CVODE	$(10^{-8}, 10^{-3})$, $(10^{-8}, 10^{-4})$, $(10^{-8}, 10^{-5})$, $(10^{-8}, 10^{-6})$, $(10^{-8}, 10^{-7})$, $(10^{-8}, 10^{-9})$, $(10^{-8}, 10^{-10})$, $(10^{-8}, 10^{-11})$.

For each experiment, we computed the eigenvalues of the exact Jacobian provided by TCHEM using MATLAB's `eig` function. The tightest set of tolerances from Table (3.2) were chosen, and the three experiments were carried out using

Table 3.3: Tolerances used to generate the reference solutions in figure 3.1.

Reference tolerances		
Mechanism name	Absolute tolerance	Relative tolerance
GRI 3.0	10^{-13}	10^{-13}
NButane	10^{-12}	10^{-12}
NDodecane	10^{-10}	10^{-10}

the EPI3V time integration scheme, storing the Jacobian calculated by TCHEM at each step. In order to provide a measure of the size of the spectrum of the Jacobian, we define the area Ω of the smallest rectangle with sides aligned with the coordinate axes such that it encloses all the eigenvalues in the complex plane, i.e. if $\lambda_j = a_j + \mathbf{i}\, b_j$ $(j = 1, ..., N)$ are the eigenvalues, the sides of the rectangle are $\alpha = \max_j(a_j) - \min_j(a_j)$ and $\beta = \max_j(b_j) - \min_j(b_j)$ so that $\Omega = \alpha\beta$. Figure 3.2 shows evolution of α and Ω during ignition, along with a normalized step cost, defined as the CPU time spent computing a step divided by the time interval stepped.

It is apparent in figure 3.2 that the *n*-dodecane mechanism has both the largest real spread α and Ω in all cases. The KIOPS algorithm is based on projections onto the Krylov subspace and the estimation of exponentials of approximate eigenvalues. If the problem's spectrum contains large positive real eigenvalues with a large Ω, computing exponentials of these augmented systems is problematic; the adaptive time stepping procedure in KIOPS will reduce the time step size significantly to accommodate the user designated tolerance. However, in the case of the *n*-dodecane mechanism, this time step reduction penalizes performance of EPIRK methods compared to the implicit scheme implemented in CVODE. This increased cost is also reflected in the normalized CPU time in figure 3.2.

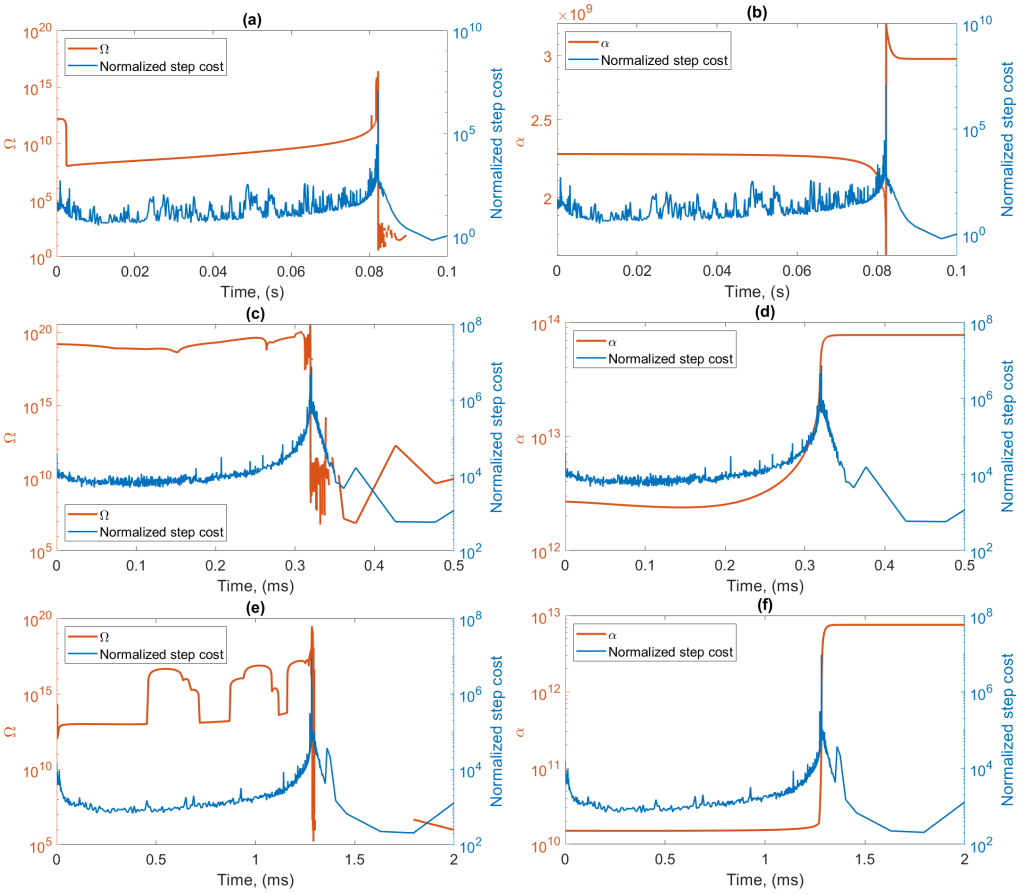

Figure 3.2: Plots vizualizing Ω, which measures the area of the spectrum, and α, measuring the real spread of the specturm, versus normalized step cost (the time spent integrating a step divided by the time step used). Plots (a) and (b) display information for the GRI mechanism. Plots (c) and (d) show the results for *n*-dodecane, while (e) and (f) demonstrate *n*-butane data. Ω and α scales are set on the left axes, while the normalized step costs scales are on the right axes of the plots.

3.4 Conclusion

In our work, we investigated the performance of the novel EPI3V variable time stepping exponential integrator for the simulation of chemically reactive and spatially homogeneous systems, i.e. chemical reactors. We compared the performance of our EPI3V method to that of CVODE, which uses a modified Newton solver and

Krylov-projection-type iterative method. Numerical expleriments were conducted for three chemical kinetics mechanisms of increasing complexity. We found that the exponential method performed favorably for certain mechanisms, but not for others. Comparable CPU time and accuracy were observed for both the GRI3.0 and *n*-butane mechanism. However, for the *n*-dodecane mechanism the CPU time for the EPI3V method required to obtain similar errors to the CVODE was an order of magnitude higher than that for CVODE at the same value of the error norm.

We found the performance degradation of EPI3V method for *n*-dodecane stems from a combination of a wide spectrum of the Jacobian and the presence of very large positive real eigenvalues. Because KIOPS is based on Krylov-iteration approximation of matrix exponential, its performance degrades in the presence of such spectra. This finding points to a promising research direction to explore alternatives to Krylov-based algorithms for estimating products of matrix exponentials with vectors. In the future we plan to investigate whether contour integration and quadrature-based methods will yield better performance [59]. We are also currently extending our study to combustion problems that include transport, in particular to modeling flame front propagation.

Chapter 4

Numerical modeling of propagating flame front.

Combustion plays a major role in many applications, so modeling, understanding, and predicting properties of combustive processes are necessary components of modern engineering. Experimental verification of theoretical predictions of combustive behavior can be very expensive in terms of time, effort, and cost. Computer simulations of combustion offer a relatively cheap and safe alternative to experimental verification and engineering design [21, 26, 27]. While computer simulations do not fully replace physical experiments, they have become an integral part of the engineering workflow [21, 26].

Numerical simulation of combustion involves solving a large system of PDEs wherein the velocity, temperature, and species are strongly coupled [26, 27, 60]. The primary challenge in multidimensional large-scale chemical combustion problems is their stiffness caused by the widely varying time-scales. The temperature, species mass fractions, and velocity equations include advective, diffusive, and reactive source terms. In low Mach number regimes, reaction term time-scales are significantly faster than those of the transport phenomena. Splitting techniques that separate the fast chemistry source term from the relatively slow transport properties are commonly used to overcome this challenge. However, this approach incurs a splitting error penalty [34, 35, 36, 60, 61]. We employ exponential integrators and show that the performance of these methods is comparable to the

state-of-the-art implicit techniques with the added benefit of not requiring an operator splitting between transport and reaction phenomena in the temperature and chemical species equations.

4.1 Model

We perform simulations to model the propagation of a flame front inside a gaseous mixture. Accurately modeling this combustion process necessitates solving a coupled system of PDEs that govern the evolution of the thermo-chemical state and the fluid dynamics. Specifically, the governing PDEs consist of equations for energy, species mass fractions, and continuity, which include source terms that account for chemical reaction rates and species production/consumption.

The reaction terms describe how chemical species interact with one another via a list of $\mathcal{N}$ many global reactions. For reaction j, the forward reaction rate constant is in Arrhenius form:

$$f_j = A_j T^{\alpha_j} \exp\Big(\frac{-E_j}{RT}\Big), \tag{4.1}$$

where A_j and α_j are pre-exponential and exponential constants, R is the universal gas constant, E_j is the activation energy, and T is the temperature. This Arrhenius expression encapsulates the non-linear dependence of the reaction rate on the temperature and accounts for the energy barrier that must be overcome so that the reaction can proceed.

If reaction j is reversible, then the backward reaction constant, b_j, is not zero and is given directly in Arrhenius form (4.1) or leveraging the equilibrium constant K_j:

$$b_j = f_j K_j. \tag{4.2}$$

How many times reaction j occurs in the forward direction per unit time per unit volume is denoted by the variable $\mathcal{R}_j$. This is the difference between forward and backward rate constants multiplied by the molar participating species concentration in the reaction raised to their stoichiometric coefficients,

$$\mathcal{R}_j = f_j \prod_{i=1}^{N_s} \chi_i^{\nu'_{ji}} - b_j \prod_{i=1}^{N_s} \chi_i^{\nu''_{ji}}, \tag{4.3}$$

where v'_{ji} and v''_{ji} are the reactant and product stoichiometric coefficients for species i in reaction j, and N_s is the number of chemical species.

Production or loss amount of a species due to reaction is equal to the difference between the forward and backward stoichiometric coefficients times the rate of reaction. Let the number of reactions as $\mathcal{N}$, then by summing all reactions the total species produced is

$$\dot{\omega}_i = \sum_{j=1}^{\mathcal{N}} (v''_{ji} - v'_{ji}) \mathcal{R}_j. \tag{4.4}$$

The equation for the total change in the amount of species Y_i mass fraction is

$$\frac{\partial Y_i}{\partial t} = -v \cdot \nabla Y_i + \frac{1}{\rho} \nabla \cdot (\rho D_i \nabla Y_i) + \frac{\dot{\omega}_i}{\rho}, \quad i \in \{1, 2, ..., N_s\}, \tag{4.5}$$

where ρ is the density of the gas, D_i is the diffusion coefficient for species i, v is the gas velocity, and c_p is the heat capacity of the gas at a constant pressure. Temperature, denoted T, in units of Kelvin, measures the kinetic energy of the gas, which is affected by the total heat enthalpy of formation. Summing over all reactions and using the enthalpy of formation of the reaction, h_i, we obtain

$$\dot{\omega}_T = \sum_{i=1}^{\mathcal{N}} h_i \dot{\omega}_i W_i. \tag{4.6}$$

Using (4.6) the governing equation for the temperature variable becomes

$$\frac{\partial T}{\partial t} = -v \cdot \nabla T + \frac{1}{\rho c_p} \nabla \cdot (\lambda \nabla T) + \frac{\dot{\omega}_T}{\rho c_p} + \frac{1}{\rho c_p} \frac{dp}{dt}, \tag{4.7}$$

where λ is the thermal diffusion coefficent. Under our assumption of the constant and uniform pressure the final term for pressure vanishes, leaving

$$\frac{\partial T}{\partial t} = -v \cdot \nabla T + \frac{1}{\rho c_p} \nabla \cdot (\lambda \nabla T) + \frac{\dot{\omega}_T}{\rho c_p}. \tag{4.8}$$

The equation for velocity divergence is derived from the continuity equation:

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho v) = -(\nabla \rho) \cdot v - \rho (\nabla \cdot v). \tag{4.9}$$

Rearranging equation (4.9) with the aid of the material derivative, $\frac{D\rho}{Dt} = \frac{\partial \rho}{\partial t} + v \cdot \nabla \rho$ we get

$$\frac{1}{\rho} \frac{D\rho}{Dt} = -\nabla \cdot v. \tag{4.10}$$

Then by applying the material derivative to the equation of state, $p = \rho RT$, where $R = \frac{\mathcal{R}}{W}$ we obtain

$$-\frac{1}{\rho}\frac{D\rho}{Dt} = \frac{1}{T}\frac{DT}{Dt} + \frac{1}{R}\frac{DR}{Dt} - \frac{1}{p}\frac{Dp}{Dt}, \tag{4.11}$$

which can be substituted into equation (4.10), yielding

$$\nabla \cdot v = \frac{1}{T}\frac{DT}{Dt} + \frac{1}{R}\frac{DR}{Dt} - \frac{1}{p}\frac{Dp}{Dt}. \tag{4.12}$$

Working out the dependence between R and W we find that:

$$\frac{DR}{Dt} = W\frac{DW^{-1}}{Dt} = \sum_{i=1}^{M}\frac{W}{W_i}\frac{DY_i}{Dt}. \tag{4.13}$$

Using (4.13) in the velocity divergence (4.12) we obtain the final form of the velocity divergence term:

$$\nabla \cdot v = \frac{1}{T}\frac{DT}{Dt} + \sum_{i=1}^{M}\frac{W}{W_i}\frac{DY_i}{Dt} - \frac{1}{p}\frac{Dp}{Dt}. \tag{4.14}$$

Finally, we apply the definition of the material derivatives to the velocity divergence equation and use the fact that pressure is held constant. These equations give the general model for a spatially inhomogeneous isobaric reactive mixture:

$$\frac{dT}{dt} = -v \cdot \nabla T + \frac{1}{\rho c_p}\nabla \cdot (\lambda \nabla T) - \frac{\dot{\omega}_T}{\rho c_p}, \tag{4.15a}$$

$$\frac{dY_i}{dt} = -v \cdot \nabla Y_i + \frac{1}{\rho}\nabla \cdot (\rho D_i \nabla Y_i) + \frac{\dot{\omega}_i}{\rho}, \quad i = 1, ..., N_s, \tag{4.15b}$$

$$\nabla \cdot v = \frac{D_T}{T}\nabla^2 T + \frac{1}{\rho c_p T}\nabla\lambda \cdot \nabla T + \frac{\dot{\omega_T}}{\rho c_p T} \tag{4.15c}$$

$$+ \sum_{i=1}^{N_s}\frac{W}{W_i}(D_i\nabla^2 T + \frac{1}{\rho}\nabla(\rho D_i)\cdot\nabla Y_i + \frac{\dot{\omega}_i}{\rho}). \tag{4.15d}$$

The governing equations are augmented with three chemical mechanisms representing fuels of practical and experimental interest in combustion. The first mechanism models the combustion of hydrogen (H_2) fuel molecules [62]. The relatively small chemical model consists of 10 species and 21 reactions and is attractive as a potential low-emission alternative to hydrocarbon fuels. Second, we consider the ammonia (NH_3) mechanism that includes 31 species and 203 reactions [63]. Ammonia is also an appealing hydrocarbon alternative because it can be used

Table 4.1: Initial gas configurations.

GRI 3.0 mass fractions, T=800K, P= 4 atm			
ϕ	CH_4	O_2	N_2
0.8	0.0446	0.2225	0.7328
1.0	0.0551	0.2200	0.7247
1.2	0.0654	0.2176	0.7168
Hydrogen mass fractions, T=445.8 K, P=4 atm			
ϕ	H_2	O_2	N_2
0.8	0.0229	0.2275	0.7494
1.0	0.0285	0.2263	0.7452
1.2	0.0340	0.2250	0.7410
Ammonia mass fractions, T=800 K, P = 1 atm			
ϕ	NH_3	O_2	N_2
0.8	0.1167	0.2057	0.6775
1.0	0.1418	0.1998	0.6582
1.2	0.1655	0.1943	0.64012

with hydrogen and is safer to store and transport than pure hydrogen fuels. Finally, we study the methane hydrocarbon fuels (CH_4) mechanism, comprising 54 species and 680 reactions [52]. In all these cases, the fuel is combined with oxygen (O_2) along with additional species such as nitrogen (N_2) accounting for air which forms most of the mass [24, 25].

4.2 Methods & Implementation

This subsection details the combustion simulation's methods and specifics. It includes a discussion of the spatial operators and their finite difference approximation, the geometry and the imposed boundary conditions, the sub-problems generated by splitting, and the methods used to solve these parts.

Computational simulations of a propagating flame front within a radially sym-

metric tube will be performed. Due to the radial symmetry, the tube's three-dimensional computational domain can be reduced to a one-dimensional line segment. For a variable X and the grid spacing of Δx, the operators are approximated by:

$$\nabla \cdot X \approx \frac{X_{i+1} - X_{i-1}}{2\Delta x}, \quad \nabla X \approx \frac{X_{i+1} - X_{i-1}}{2\Delta x}, \quad \nabla^2 X \approx \frac{X_{i-1} - 2X_i + X_{i+1}}{\Delta x^2}.$$

A Marker-and-Cell (MAC) method [64] is used for fluid flow stability; here, the faces of the computational cells store the velocity, and the centers store the pressure and other scalar quantities. In one spatial dimension, the cells reduce to two staggered grids, where the velocity grid is offset by one-half the step-size, e.g., the spacing between neighboring velocity and scalar spatial points is $\frac{\Delta x}{2}$. Due to staggering, an equation will often require data stored on one grid rather than the other. These scenarios are handled in one of two ways. If a derivative quantity is needed, it is possible to obtain these estimations with second-order accuracy via the equations with the step size of $\frac{\Delta x}{2}$ via the estimation

$$X'_{\textbf{staggered}} = \frac{X_{i+\frac{1}{2}} - X_{i-\frac{1}{2}}}{\Delta x}. \tag{4.16}$$

Otherwise, adjacent cell averaging estimation is used to obtain the needed quantities.

The imposed boundary conditions are such that one end of the tube is open while the other is closed. The right end of the tube is designated as the open end, and the left end is closed. A homogeneous Neumann boundary condition is applied at the open end, whereas the closed end is subject to a fixed, non-homogeneous Dirichlet boundary condition. In the MAC configuration, the boundary locations coincide with the velocity grid. Thus, non-homogeneous Dirichlet scalar boundary quantities cannot be stored. Near the closed boundary, when a stencil requires information outside the computational domain, the stenciling assumes the data at the needed point has the same value as the non-homogeneous Dirichlet boundary condition, which is enforced to be to the gas's uniform initial composition as seen in the table (4.1).

Splitting methods are a commonly used strategy to overcome challenges associated with combustion in state-of-the-art combustion simulation strategies. A

standard approach as developed in Tomboulides and Orzag [60] tackles the stiff system of PDEs with a two-stage, three sub-step splitting approach. Splitting between the vector equation (4.15d) and the scalar equations (4.15a-4.15b) is made, and then a further splitting between the transport terms and the chemical reaction terms inside equations (4.15a-4.15b) is used. The first two sub-steps will hold the velocity constant to advance the temperature and species variables. The slower transport phenomena are first advanced with a semi-implicit scheme; then, the faster reaction term is integrated with an implicit BDF method. The advancement of the velocity field is last and is typically done with a pressure corrector method.

To numerically solve the governing system of PDEs given by equations (4.15a-4.15d), a method of lines approach is employed. The spatial derivatives are approximated using a second-order finite difference scheme with uniform grid spacing Δx, reducing the PDEs to a system of coupled ODEs. The scalar quantities of temperature and species mass fractions each have an ODE for each computational point. We denote the number of spatial points as M and then the state vector y can be represented as a vector of length $(N_s + 1) \times M$:

$$y = [T_1, ..., T_M, Y_{1,1}, ..., Y_{1,M}, Y_{2,1}, ..., Y_{2,M}, ..., Y_{N_s,1}, ..., Y_{N_s,M}]'. \tag{4.17}$$

Additionally, in order to calculate the quantities once per time step all the thermo-

chemical and thermodynamics properties are stored for each computational point

$$\begin{aligned}
\rho &= [\rho_1, ..., \rho_M], \\
\lambda &= [\lambda_1, ..., \lambda_M], \\
c_p &= [c_{p1}, ..., c_{pM}], \\
\dot{\omega}_T &= [\dot{\omega}_{T1}, ..., \dot{\omega}_{TM}], \\
\dot{\omega}_1 &= [\dot{\omega}_{1,1}, ..., \dot{\omega}_{1,M}], \\
&\vdots \\
\dot{\omega}_{N_s} &= [\dot{\omega}_{N_s,1}, ..., \dot{\omega}_{N_s,M}], \\
D_1 &= [D_{1,1}, ..., D_{1,M}], \\
&\vdots \\
D_{N_s} &= [D_{N_s,1}, ..., D_{N_s,M}], \\
\lambda_{N_s} &= [\lambda_{N_s,1}, ..., \lambda_{N_s,M}].
\end{aligned}$$

By applying finite difference approximations to the scalar equations (4.15a-4.15b) we generate a system of $(N_s + 1) \times M$ ODEs that advance the temperature and species mass fractions. For the i-th chemical species, the first spatial point requires the Dirichlet boundary condition, and the ODE is written as:

$$\frac{dY_{i,1}}{dt} = -\frac{v_1 + v_2}{2} \cdot \frac{Y_{i,2} - Y_{i,\text{inlet}}}{2\Delta x} + \frac{1}{\rho_1} \cdot \Big(\frac{\rho_2 - \rho_{\text{inlet}}}{2\Delta x} \cdot D_{i,1} \cdot \frac{Y_{i,2} - Y_{i,\text{inlet}}}{2\Delta x} + \rho_1 \cdot \frac{D_{i,2} - D_{1,\text{inlet}}}{2\Delta x} \cdot \frac{Y_{i,2} - Y_{i,\text{inlet}}}{2\Delta x} + \rho_1 \cdot D_{i,1} \cdot \frac{Y_{i,2} - 2Y_{i,1} + Y_{i,\text{inlet}}}{\Delta x^2} \Big) + \frac{\dot{\omega}_i}{\rho_1}.$$

The variables that include the "inlet" subscript denote the scalar non-homogeneous Dirichlet boundary conditions. Any interior j-th point will have the usual two and three-point stenciling

$$\frac{dY_{i,j}}{dt} = -\frac{v_j + v_{j+1}}{2} \cdot \frac{Y_{i,j+1} - Y_{i,j-1}}{2\Delta x} + \frac{1}{\rho_j} \cdot \Big(\frac{\rho_{\text{j+1}} - \rho_{\text{j-1}}}{2\Delta x} D_{i,j} \cdot \frac{Y_{i,\text{j+1}} - Y_{i,\text{j-1}}}{2\Delta x} + \rho_j \cdot \frac{D_{1,\text{j+1}} - D_{1,\text{j-1}}}{2\Delta x} \cdot \frac{Y_{i,\text{j+1}} - Y_{i,\text{j-1}}}{2\Delta x} + \rho_j \cdot D_{i,j} \cdot \frac{Y_{i,j+1} - 2Y_{i,j} + Y_{i,\text{j-1}}}{\Delta x^2} \Big) + \frac{\dot{\omega}_{i,j}}{\rho_j}.$$

Then, for the species ODE associated with the point M nearest to the homogeneous

Neumann boundary condition we have:

$$\frac{dY_{i,M}}{dt} = -\frac{v_M + v_{M+1}}{2} \cdot \frac{Y_{1,M} - Y_{1,M-1}}{2\Delta x} + \frac{1}{\rho_M} \cdot \Big(\frac{\rho_{\mathrm{M}} - \rho_{\mathrm{M\text{-}1}}}{2\Delta x} \cdot D_{1,M} \frac{Y_{\mathrm{M}} - Y_{\mathrm{M\text{-}1}}}{2\Delta x} + \rho_M \cdot \frac{D_{i,\mathrm{M}} - D_{i,\mathrm{M\text{-}1}}}{\Delta x} \cdot \frac{Y_{i,\mathrm{M}} - Y_{i,\mathrm{M\text{-}1}}}{2\Delta x} + \rho_M \cdot D_{i,M} \cdot \frac{Y_{i,\mathrm{M\text{-}1}} - Y_{i,M}}{\Delta x^2} \Big) + \frac{\dot{\omega}_{i,M}}{\rho_M}.$$

The boundary conditions for the temperature ODEs are applied in a similar way; for the first computational point, we have the following temperature ODE

$$\frac{dT_1}{dt} = -\frac{v_1 + v_2}{2} \cdot \frac{T_2 - T_{\mathrm{inlet}}}{2\Delta x} + \frac{1}{c_{p1} \cdot \rho_1} \cdot \Big(\frac{\lambda_2 - \lambda_{\mathrm{inlet}}}{2\Delta x} \cdot \frac{T_2 - T_{\mathrm{inlet}}}{2\Delta x} + \lambda_1 \cdot \frac{T_2 - 2T_1 + T_{\mathrm{inlet}}}{\Delta x^2} \Big) + \frac{\dot{\omega_{T1}}}{c_{p1} \cdot \rho_1}.$$

The interior points have ODEs which are

$$\frac{dT_j}{dt} = -\frac{v_j + v_{j+1}}{2} \cdot \frac{T_{j+1} - T_{j-1}}{2\Delta x} + \frac{1}{c_{pj} \cdot \rho_j} \cdot \Big(\frac{\lambda_{j+1} - \lambda_{j-1}}{2\Delta x} \cdot \frac{T_{\mathrm{j+1}} - T_{j-1}}{2\Delta x} + \lambda_j \cdot \frac{T_{j+1} - 2T_j + T_{j-1}}{\Delta x^2} \Big) + \frac{\dot{\omega_{Tj}}}{c_{pj} \cdot \rho_j}.$$

And the point nearest the homogeneous Neumann boundary condition have the following ODE

$$\frac{dT_M}{dt} = -\frac{v_M + v_{M+1}}{2} \cdot \frac{T_M - T_{M-1}}{2\Delta x} + \frac{1}{c_{pj} \cdot \rho_j} \cdot \Big(\frac{\lambda_M - \lambda_{M-1}}{2\Delta x} \cdot \frac{T_{\mathrm{M}} - T_{M-1}}{2\Delta x} + \lambda_M \cdot \frac{T_{M-1} - T_M}{\Delta x^2} \Big) + \frac{\dot{\omega_{TM}}}{c_{pM} \cdot \rho_M}.$$

All these ODEs are altogether compactly written as the vector $F(y)$:

$$F(y) = \Big[\frac{dT_1}{dt}, .., \frac{dT_M}{dt}, \frac{dY_{1,1}}{dt}, ..., \frac{dY_{1,M}}{dt}, \frac{dY_{2,1}}{dt}, ..., \frac{dY_{2,M}}{dt}, ..., \frac{dY_{N_s,1}}{dt}, ..., \frac{dY_{N_s,M}}{dt} \Big]'. \tag{4.18}$$

Exponential integrators, described in the next subsection, offer an advantage compared to the implicit methods of advancing the large scalar temperature and species ODEs (4.18) without a splitting method between the transport and reaction terms. Advancing this large system of ODEs is the first of the two sub-steps employed to advance the whole equation. This computation is then followed by the advancement of velocity.

For higher dimensional problems the velocity advancement is typically done using a pressure corrector method. However, since we have a one dimensional model we can integrate of the velocity divergence equation directly (4.15d). Applying spatial discretization to the divergence of the velocity on the scalar grids gives a series of M equations.

$$\begin{aligned}
\nabla \cdot v_1 = {} & \frac{D_{T1}}{T_1} \cdot \frac{T_2 - 2T_1 + T_{\text{inlet}}}{\Delta x^2} + \frac{1}{\rho_1 \cdot c_{p1} \cdot T_1} \cdot \frac{\lambda_2 - \lambda_{\text{inlet}}}{2\Delta x} \cdot \frac{T_2 - T_{\text{inlet}}}{2\Delta x} + \\
& \sum_{i=1}^{N_s} \frac{W_1}{W_i} \cdot \Big(D_{i,1} \cdot \frac{T_2 - 2T_1 + T_{\text{inlet}}}{\Delta x^2} + \frac{D_{i,1}}{\rho_1} \cdot \frac{\rho_2 - \rho_{\text{inlet}}}{\Delta x} \cdot \frac{Y_{i,2} - Y_{\text{inlet}}}{\Delta x} + \\
& \frac{D_{i,2} - D_{i,\text{inlet}}}{\Delta x} \cdot \frac{Y_{i,2} - Y_{\text{inlet}}}{\Delta x} + \frac{\dot{\omega}_{i,1}}{\rho_1} \Big), \\
& \vdots \\
\nabla \cdot v_j = {} & \frac{D_{Tj}}{T_j} \cdot \frac{T_{j+1} - 2T_j + T_{j-1}}{\Delta x^2} + \frac{1}{\rho_j \cdot c_{pj} \cdot T_j} \cdot \frac{\lambda_{j+1} - \lambda_{j-1}}{2\Delta x} \cdot \frac{T_{j+1} - T_{j-1}}{2\Delta x} + \\
& \sum_{i=1}^{N_s} \frac{W_j}{W_i} \cdot \Big(D_{i,j} \cdot \frac{T_{j+1} - 2T_j + T_{j-1}}{\Delta x^2} + \frac{D_{i,j}}{\rho_j} \cdot \frac{\rho_{j+1} - \rho_{j-1}}{\Delta x} \cdot \frac{Y_{i,j+1} - Y_{j-1}}{\Delta x} + \\
& \frac{D_{i,j+1} - D_{i,j-1}}{\Delta x} \cdot \frac{Y_{i,j+1} - Y_{i,j-1}}{\Delta x} + \frac{\dot{\omega}_{i,j}}{\rho_j} \Big), \\
& \vdots \\
\nabla \cdot v_M = {} & \frac{D_{T_M}}{T_M} \cdot \frac{T_{M-1} - T_M}{\Delta x^2} + \frac{1}{\rho_M \cdot c_{pM} \cdot T_M} \cdot \frac{\lambda_M - \lambda_{M-1}}{2\Delta x} \cdot \frac{T_M - T_{M-1}}{2\Delta x} + \\
& \sum_{i=1}^{N_s} \frac{W_M}{W_i} \cdot \Big(D_{i,M} \cdot \frac{T_M - T_M}{\Delta x^2} + \frac{D_{i,M}}{\rho_M} \cdot \frac{\rho_M - \rho_{M-1}}{\Delta x} \cdot \frac{Y_{i,M+1} - Y_{M-1}}{\Delta x} + \\
& \frac{D_{i,M} - D_{i,M-1}}{\Delta x} \cdot \frac{Y_{i,M} - Y_{i,M-1}}{\Delta x} + \frac{\dot{\omega}_{i,M}}{\rho_M} \Big).
\end{aligned}$$

Solving for an updated velocity is done by approximating the integral

$$\int_0^x \nabla \cdot v(s) dx = \sum_{n=1}^{N} \int_{a_n}^{b_n} \nabla \cdot v(x) dx. \tag{4.19}$$

Each a_i and b_i represent a cell with two velocity points and a single scalar point. The approximation for the integral of a cell is given by:

$$\int_a^b \nabla \cdot v(x) dx \approx \frac{b-a}{6} \cdot \Big[\nabla \cdot v(a) + 4 \nabla \cdot v\Big(\frac{a+b}{2}\Big) + \nabla \cdot v(b) \Big]. \tag{4.20}$$

Because a and b are chosen to be on the cell boundaries, the quantity of $\nabla \cdot v$ cannot be computed directly at both a and b; however, the center $\frac{a+b}{2}$ can as that is on the scalar grid. Thus, the edges are approximated with averaging. Once the velocity field has been advanced by this method, the time step is complete.

In our implementation, we employ third-party software libraries and packages to aid in constructing the source term. Utilizing third-party software allows our implementation to leverage validated and optimized routines, and this modular approach allows for flexibility in incorporating different chemical mechanisms or thermodynamics models without requiring extensive development and testing.

TCHEM suite [65] is an open-source software package written in C++ by Sandia National Laboratories. TCHEM implements a variety of different combustion problems, and the one pertinent to the propagating flame front problem is the homogeneous isobaric batch reactor problem. The reactive terms written in equations (4.15a-4.15b) by themselves are equivalent to a homogeneous batch reactor problem. Thus, TCHEM is used to obtain these required source terms for the larger propagating flame front problem. At the beginning of each integration substep, TCHEM is called once for each computational point. TCHEM is given the thermo-chemical state and the pressure for that point and outputs the source term which is then stored for use as a component of the total source terms in equations (4.15a-4.15b).

Currently TCHEM does not support the generation of transport terms or their necessary transport coefficients. For this task, Cantera [50] was chosen, which is well-known open-source software for use in Python, C++, and MATLAB combustion codes. Our propagating flame front simulation leverages Cantera to obtain the thermal diffusion coefficient for temperature and species, D_T and D_i, respectively, and the thermal conductivity, λ. Cantera uses YAML files to initiate gas phase objects. At the start of each update step, we provide the thermo-chemical state to Cantera via its C++ interface for each grid point. Cantera then calculates the required transport coefficients, which are returned and saved as vectors.

A comparable version of the simulation using the NGA code framework is used to compare and validate the results of the exponential methods. NGA is

a structured, finite-difference code that is staggered in time and space and has been developed for accurate direct numerical simulation (DNS) and large eddy simulation (LES) computations in one, two, or three spatial dimensions [39]. This framework is flexible and allows development for a wide variety of CFD simulations, including but not limited to low mach turbulent flows [39], solid gas suspensions [66], and particle collision dynamics [67]. This code will represent and implement the standard semi-implicit solutions for the scalar transport, implicit treatment of reaction, and fractional-step pressure corrector methods for the velocity update. Additionally, ChemJac produces analytic versions of the chemical Jacobians [68].

4.2.1 Exponential time integrators

Exponential time integrators are a class of methods designed for systems of ODEs of the form

$$\frac{dy}{dt} = F(y), \quad y(t_0) = y_0. \tag{4.21}$$

On large-scale stiff problems they have demonstrated favorable stability and computational time costs when compared to implicit time integration methods [13, 12, 11, 10].

To numerically integrate the system of ODEs, the time variable is represented discretely as a set of equally spaced points $[t_0, t_1, \ldots, t_m]$. where the time difference between any two consecutive points is Δt, i.e., $t_{n+1} = t_n + \Delta t$. Assuming that the source term F is differentiable, we approximate F and its derivative, F', at time t_n. We denote their approximations by $y_n \approx y(t_n)$, $F_n \approx F(y(t_n))$, and $F'_n \approx F'(y(t_n))$ respectively. Using this notation the second order in time exponential Euler scheme can be written as

$$y_{n+1} = y_n + \varphi_1(h_n J_n)\Delta t F_n, \quad \varphi_1(z) = \frac{e^z - 1}{z}. \tag{4.22}$$

The propagating flame front problem and the three chemical mechanisms present a system of ODEs comprising thousands of equations. Obviously, for large dimensional problems, the exact evaluation of the matrix-vector product $\varphi_1(h_n J_n)\Delta t F_n$ is prohibitively computationally expensive, and such products require approximation. The evaluations of these products of exponential-like matrix functions and

Simulation detail	NGA	exponential
Strang Split	Yes	Yes
Lie Split	Yes	No
Velocity update	Pressure corrector	2nd order integrator
Scalar integrator	BDF	EPI2
Jacobians	Analytic Jacobians by ChemJac	Finite difference approximation

Table 4.2: Table of code differences

vectors that constitute an exponential integrator's main computational cost, thus, must be done as efficiently as possible. Some of the most effective and general methods to estimate such products are based on Krylov subspace projections. As discussed above, KIOPS has been shown to be one of the most efficient general algorithms, and we employ it here [47]. However, similar to what was seen in the case of homogeneous batch reactor problems, we need to use KIOPS with full orthogonalization of the Krylov basis since partial orthogonalization does not provide sufficient accuracy. We use EPIC (Exponential Propagation Integrators Collection) software package (https://faculty.ucmerced.edu/mtokman/software), which implements several exponential integrators along with KIOPS method to compute the matrix functions-vector products.

4.3 Numerical experiments

For our experiments we model 4 millimeters (mm) long tube filled with a uniform premixed gas mixture. The tube is represented using a uniform one-dimensional grid of 400 scalar and 401 velocity grid points, spaced equidistantly. The spatial resolution is 10 micrometers (μm).

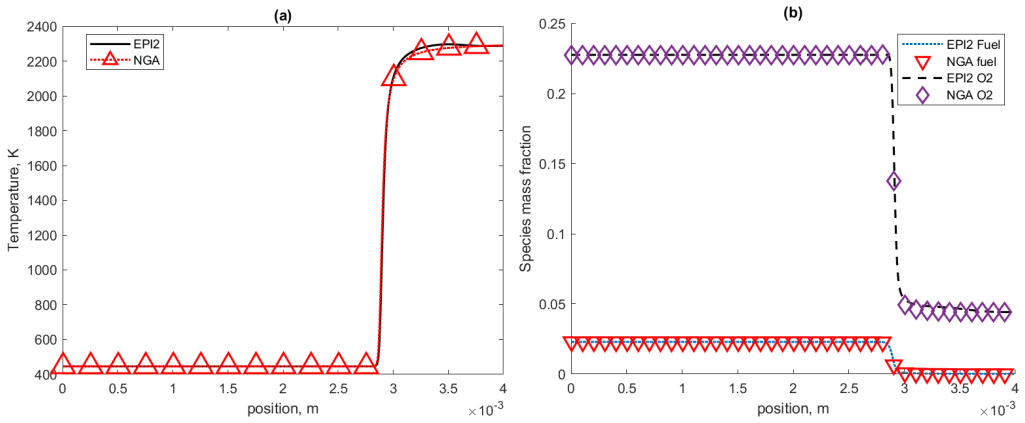

Figure 4.1: Comparison of NGA and exponential methods solutions of the temperature (a) and fuel/oxidizer (b) variables with an equivalence ratio of 0.8 for the hydrogen mechanism.

Initially the premixed gas mixture in the tube is at a temperature below its auto-ignition point. A localized Gaussian heat source is then introduced near the open end of the tube to initiate flame formation and propagation. This heat source takes the form:

$$H(x) = 5 \cdot 10^{10} \cdot \exp\Big(\frac{(x-c)^2}{10^8}\Big), \tag{4.23}$$

where x is the spatial coordinate variable and c is a centering parameter set to 90% of the tube's length as measured from the closed end. The concentrated localized heating from the Gaussian profile raises the gas temperature beyond auto-ignition in the region around $x = c$.

After applying the heat source for 100 microseconds (μs), the external heating element is deactivated. The flame front is then allowed to propagate freely for an additional 100 μs, after which the simulation terminates. At termination, the full thermo-chemical state (species, temperature) and velocity field data are saved by the simulation code.

This saved data provides the initial conditions to restart the simulation and perform precision comparisons between two different numerical integration techniques. The precision study is conducted over three equivalence ratios of lean, stoichiometric, and rich mixtures per the initial configuration data summarized in Table (4.1). For each case, the time period required for the flame front to propagate

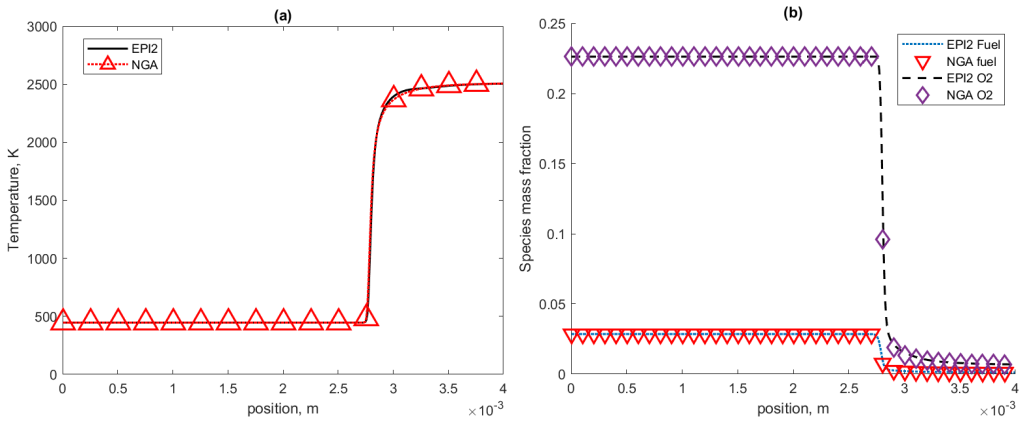

Figure 4.2: Comparison of NGA and exponential methods solutions of the temperature (a) and fuel/oxidizer (b) variables with an equivalence ratio of 1.0 for the hydrogen mechanism.

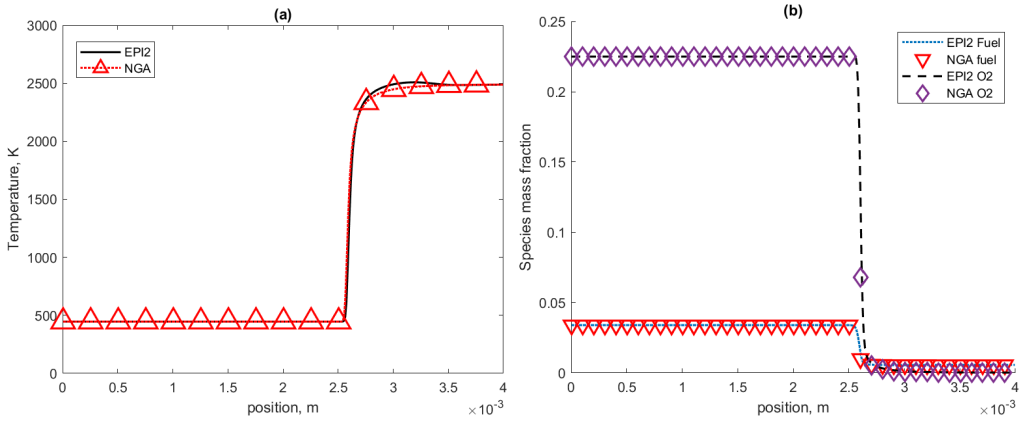

Figure 4.3: Comparison of NGA and exponential methods solutions of the temperature (a) and fuel/oxidizer (b) variables with an equivalence ratio of 1.2 for the hydrogen mechanism.

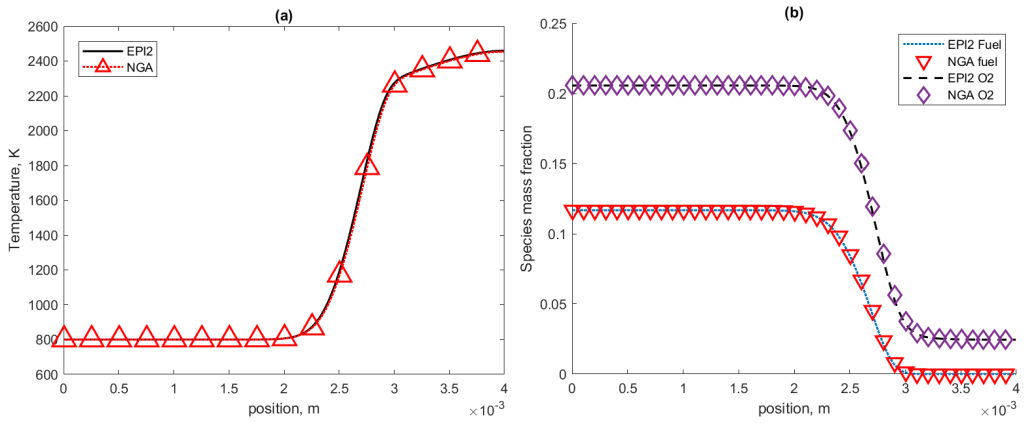

Figure 4.4: Comparison of NGA and exponential methods solutions of the temperature (a) and fuel/oxidizer (b) variables with an equivalence ratio of 0.8 for the ammonia mechanism.

one flame length is first computed using the formula:

$$\frac{\max(\nabla T)}{\Delta T}. \tag{4.24}$$

A temporal refinement is then conducted by performing a sequence of simulations with successively smaller time step sizes spanning this propagation time period.

Upon completion of each simulation, the final thermo-chemical state (species mass fractions, temperature) and velocity field are recorded. It is important to note that due to significant algorithmic differences between the implicit and the exponential time integration solvers, as outlined in Table (4.2), the two schemes are not expected to converge to an identical solution as the time-step approaches zero. However, in the study, we demonstrate that both numerical methods capture the qualitative behavior and profiles of combustion consistently.

Figures (4.1-4.9) show spatial profiles of the temperature, fuel, and oxidizer variables saved at the last step of the numerical experiments for both integration methods across all mechanisms for different values of time steps. The oxidizer is oxygen across all experiments, but the fuels are hydrogen, ammonia, and methane for the hydrogen, ammonia, and GRI mechanisms respectively. Separate plots for temperature and species mass fractions are presented due to the significant difference in scale between mass fractions and temperature.

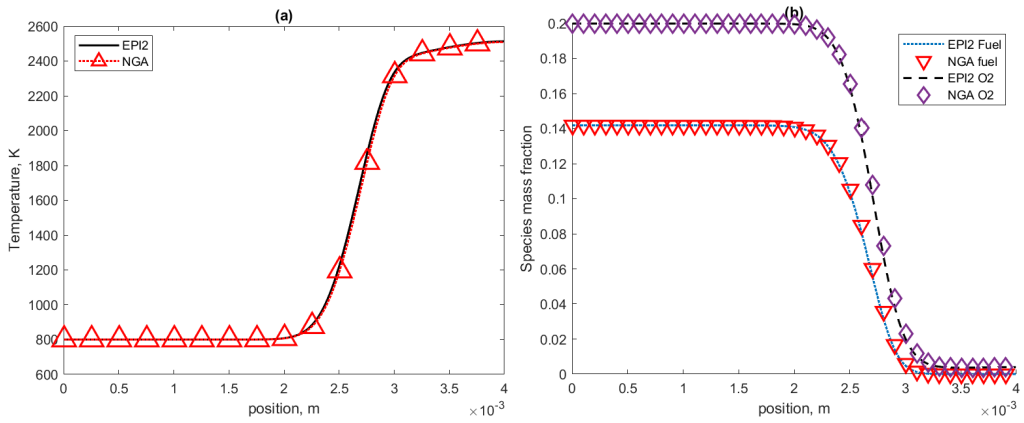

Figure 4.5: Comparison of NGA and exponential methods solutions of the temperature (a) and fuel/oxidizer (b) variables with an equivalence ratio of 1.0 for the ammonia mechanism.

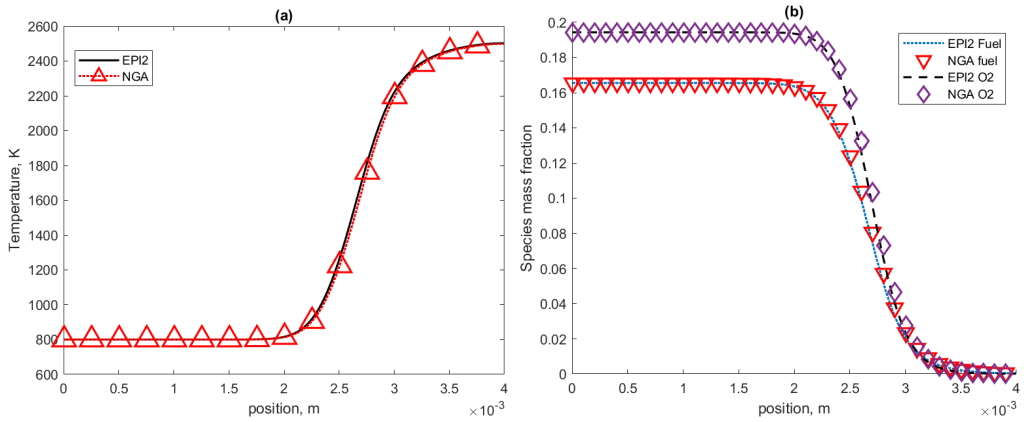

Figure 4.6: Comparison of NGA and exponential methods solutions of the temperature (a) and fuel/oxidizer (b) variables with an equivalence ratio of 1.2 for the ammonia mechanism.

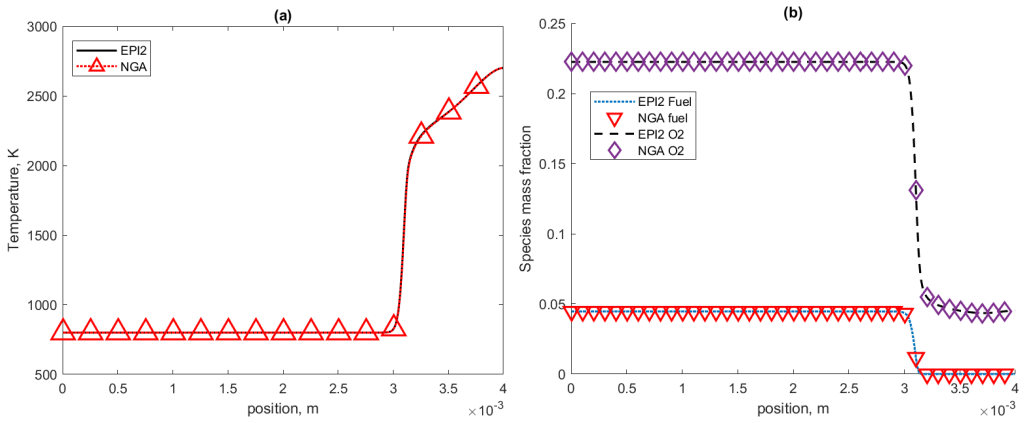

Figure 4.7: Comparison of NGA and exponential methods solutions of the temperature (a) and fuel/oxidizer (b) variables with an equivalence ratio of 0.8 for the GRI mechanism.

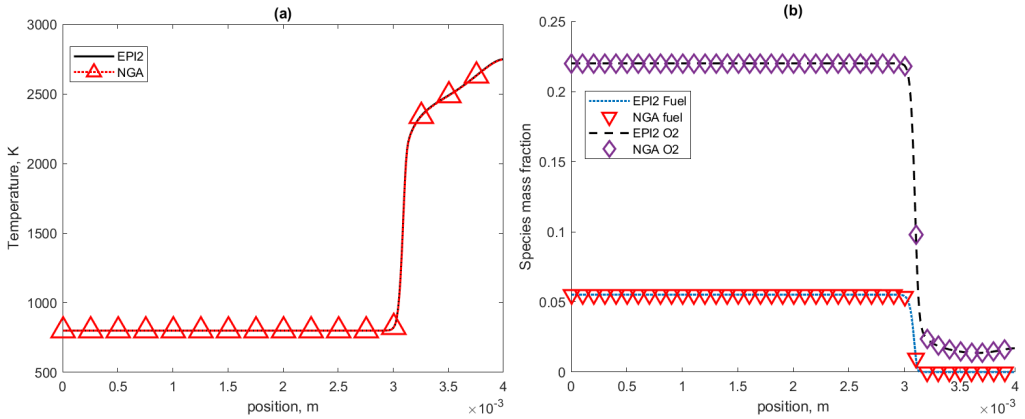

Figure 4.8: Comparison of NGA and exponential methods solutions of the temperature (a) and fuel/oxidizer (b) variables with an equivalence ratio of 1.0 for the GRI mechanism.

With three species and three equivalence ratios, nine figures are presented. These are grouped by mechanism first and within the group, sorted in ascended equivalence ratio order (0.8, 1.0, 1.2). So figures (4.1-4.3) detail the hydrogen mechanism experiment's data, figures (4.4-4.6) plot the ammonia data, and finally figures (4.7-4.9) show the GRI mechanism's temperature, fuel and oxidizer information.

In order to highlight the more subtle quantitative differences between the two

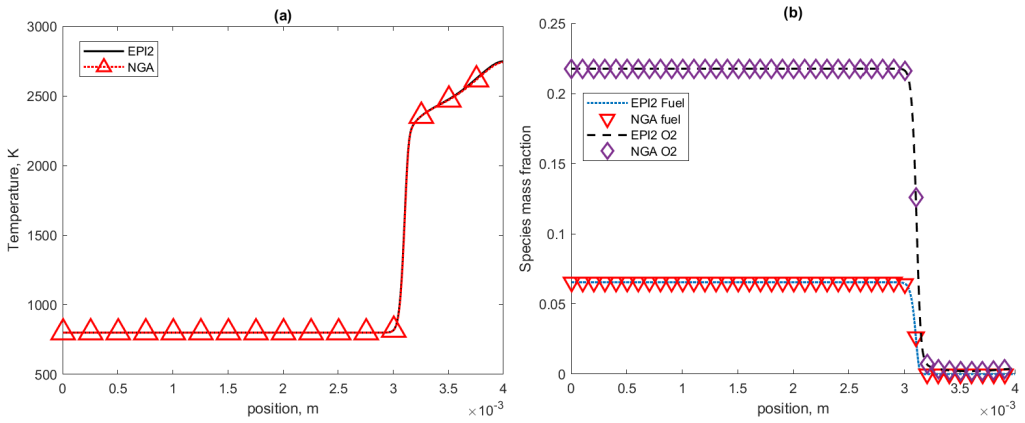

Figure 4.9: Comparison of NGA and exponential methods solutions of the temperature (a) and fuel/oxidizer (b) variables with an equivalence ratio of 1.2 for the GRI mechanism.

integration methods not seen in figures (4.1-4.9), figures (4.10-4.12) plots the absolute point-wise difference in log scale of temperature, fuel and oxidizer variables saved by the precision experiments. The figures are organized in the order of hydrogen, ammonia, and GRI. Each figure contains three plots of the absolute difference across the equivalence ratios of 0.8, 1.0, 1.2.

The spatial profile exhibits three distinct regions for a one-dimensional propagating flame front. The region upstream of the flame front, where the gas mixture is still unburnt, has a uniform temperature and species mass fraction distributions. The reaction zone is where exothermic chemical reactions driving the combustion process occur and is characterized by steep gradients in temperature, fuel, and oxidizer mass fractions. Downstream of the reaction zone lies the post-reaction region, where the combustion products begin to reach equilibrium in the wake of the propagating flame front. In this region, temperature and species mass fraction profiles tend to approach plateaued values, as the chemical reactions have largely completed and convective and diffusive transport processes dominate.

Figures (4.1-4.9) illustrate that both the NGA and exponential methods capture very similar qualitative behavior in the combustion profiles. However, it is noteworthy that there are minor quantitative differences between the two methods. Although the results appear visually similar, a closer examination of figures

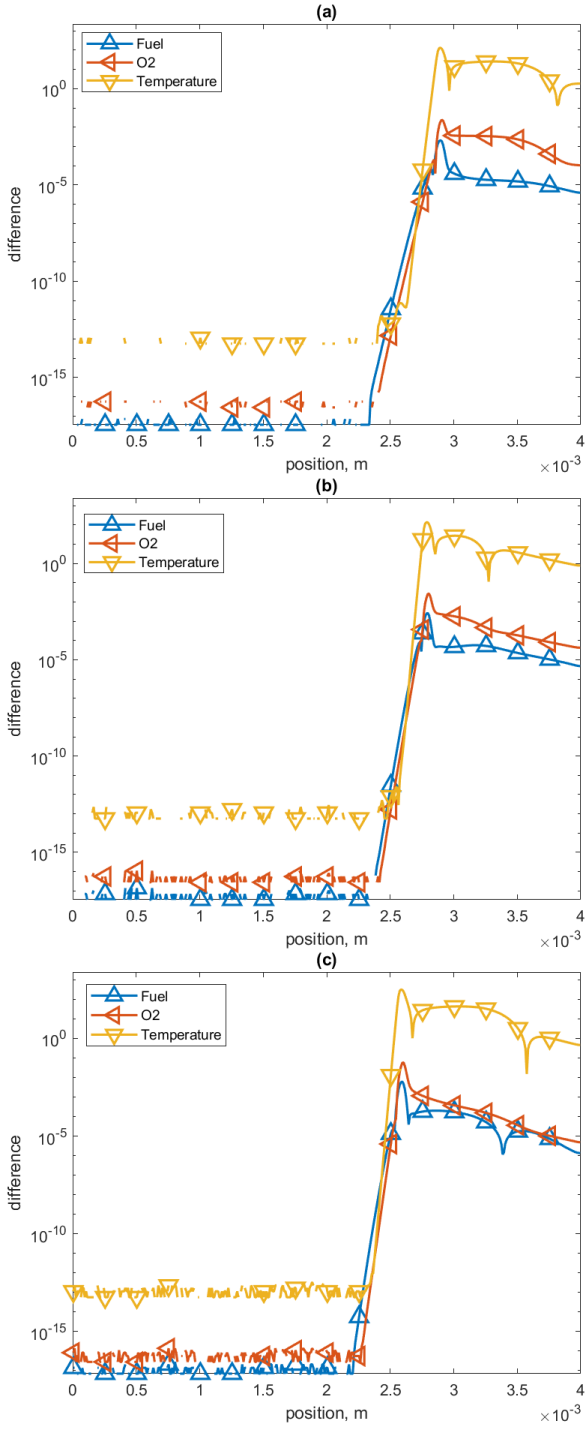

Figure 4.10: The hydrogen mechanism's final time-step comparison of the EPI2 and NGA integrator's hydrogen fuel, oxygen, and temperature values at each spatial point for the equivalence ratios of 0.8 (a), 1.0 (b), and 1.2 (c). Differences are given in absolute value.

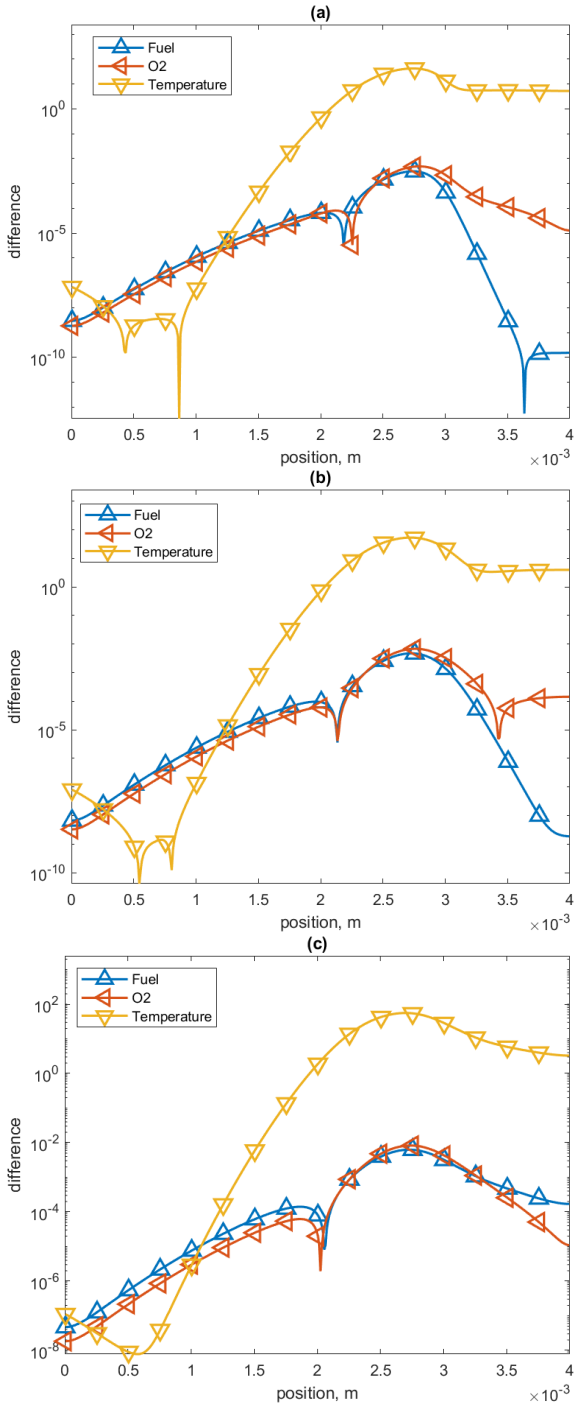

Figure 4.11: The ammonia mechanism's final time-step comparison of the EPI2 and NGA integrator's fuel (NH_3), oxygen, and temperature values at each spatial point for the equivalence ratios of 0.8 (a), 1.0 (b), and 1.2 (c). Differences are given in absolute value.

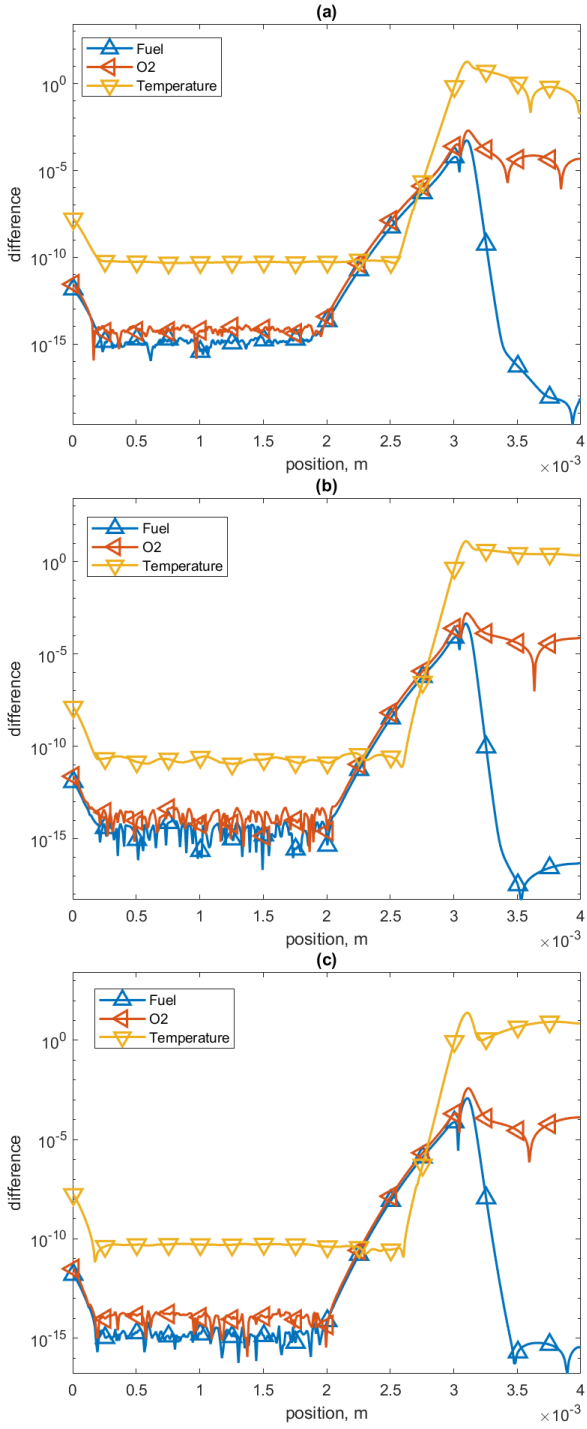

Figure 4.12: The GRI3.0 mechanism's final time-step comparison of the EPI2 and NGA integrator's fuel, oxygen, and temperature values at each spatial point for the equivalence ratios of 0.8 (a), 1.0 (b), and 1.2 (c). Differences are given in absolute value.

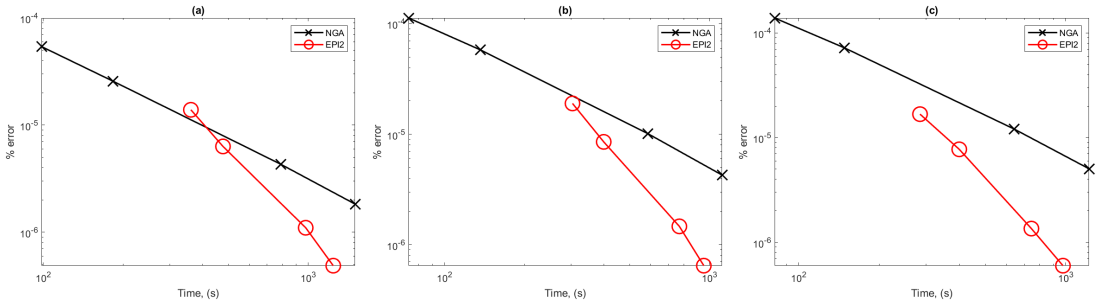

Figure 4.13: Precision diagram for NGA and exponential methods for the GRI methane mechanism with 54 species and the equivalence ratio of (a) 0.8, (b) 1.0, and (c) 1.2. The y axis shows the square mean percentage absolute error (SMAPE) normalized for the vector length of the fuel, temperature, and oxidizer variables compared to the highly refined reference solution.

(4.10-4.12) reveals quantitative discrepancies.

A notable feature observed in the results for the GRI mechanism is the presence of a remnant from the initial flame kernel used to initiate flame propagation within the tube. This remnant manifests as an additional peak in the temperature profiles in the post-flame region, as evident in Figures (4.7-4.9). This artifact would have been convected out of the computational domain with sufficient simulation time due to the transport phenomena, resulting in a more uniform, plateau-like temperature profile in the post-reaction region.

For the hydrogen mechanism, as seen in figures (4.1-4.3) and (4.10), the exponential methods show minor discrepancies in temperature in the initial portion of the post-reaction region. However, as we progress further downstream, these deviations between the two numerical solutions gradually diminish. Aside from the hydrogen case, there are small orders of 10 differences or less in temperature in the reaction and the initial regions of the reaction and post-reaction zones. This is a small relative difference, given the temperature values are in the order of thousands.

The relative performance of the NGA and exponential time integration methods is plotted in figure (4.15). This figure shows the CPU time versus the error for both methods over one characteristic flame length. The x-axis represents the CPU

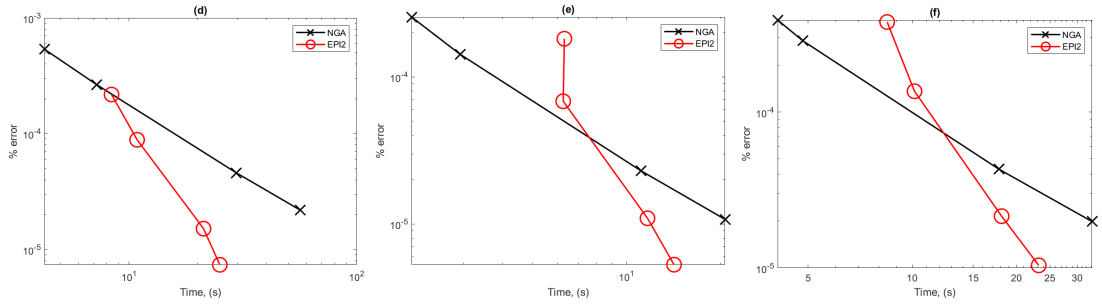

Figure 4.14: Precision diagram for NGA and exponential methods for the hydrogen mechanism of 10 species and the equivalence ratio of (a) 0.8, (b) 1.0, and (c) 1.2. The y axis shows the square mean percentage absolute error (SMAPE) normalized for the vector length of the fuel, temperature, and oxidizer variables compared to the highly refined reference solution.

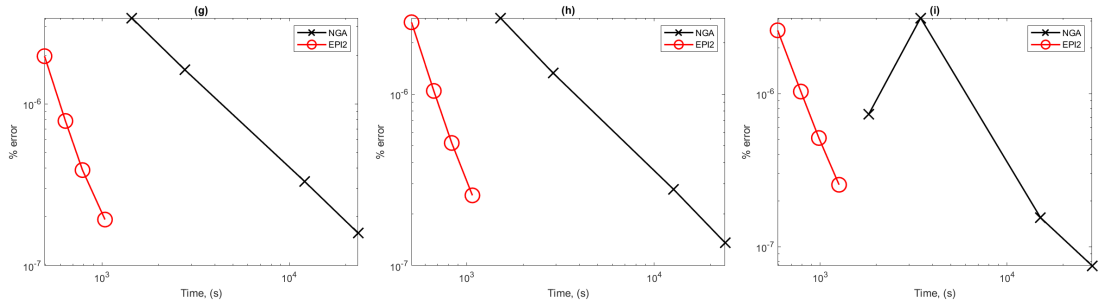

Figure 4.15: Precision diagram for NGA and exponential methods for the ammonia mechanism with 31 species and the equivalence ratio of (a) 0.8, (b) 1.0, and (c) 1.2. The y axis shows the square mean percantage absolute error (SMAPE) normalized for the length of vector for the fuel, temperature, and oxidizer variables compared to the highly refined reference solution.

time spent integrating the scalar update step, while the y-axis depicts the error.

The chosen error norm is the Square Mean Absolute Percentage Error (SMAPE). SMAPE is beneficial as it smooths out single points with high relative errors by averaging over the entire computational domain. Without using SMAPE, specific points in the reaction and post-reaction zone corresponding to vanishing species exhibited sizeable relative errors. Additionally, the error calculation considers the same subset of the thermo-chemical state variables of the temperature, fuel, and oxidizer mass fractions. Comparing the results using the entire thermo-chemical state, including all species, did not reveal any qualitative behavioral differences.

The precision diagrams in figure (4.15) demonstrate that the exponential method performs comparably to the implicit method when updating the scalar equations. Notably, when the time step size is small, the exponential method becomes a more efficient choice compared to the NGA method.

4.4 Conclusion

The propagating flame front experiments demonstrate that exponential time integrators can reliably obtain numerical solutions that exhibit qualitatively similar behavior to implicit methods, which are widely considered state-of-the-art for stiff combustion problems. Furthermore, the computational efficiency of the exponential approach, as measured by the CPU run time vs error, is comparable to that of implicit methods. A notable advantage of exponential integrators is their ability to advance the scalar equations without a splitting between the transport and reactive terms, leading to improved accuracy.

The results are promising as a proof of concept for the application of exponential time integrator methods to both propagating flame front and more general combustion simulations. The current implementation of the exponential propagating flame front is penalized in its CPU time performance. The method relies on finite difference approximations for the Jacobian terms, which are comparatively slow to analytic methods for problems of moderate scale or larger. Because the scalar advancement step is handled without employing a Lie splitting technique,

the derivatives of both the transport and reaction terms are treated together as a finite difference approximation. The transport component of the Jacobian is particularly complex since the transport coefficients are interdependent on the thermo-chemical state, making it challenging to derive an analytic closed-form expression. Future work should focus on developing methods to address this issue. One potential approach is to use a partially analytic Jacobian, where the reactive component is provided analytically by TCHEM, and the transport component is approximated using finite differences. Another avenue worth exploring is the use of automatic differentiation techniques, where software packages such as SACADO can be employed to calculate the combined source term, including both transport and chemical contributions.

Chapter 5

Conclusion

Numerical modeling of combustion is challenging due to its multi-physics and multi-scale nature. The complex interaction of fluid mechanics, thermodynamics, and chemical kinetics processes in combustion results in a strongly coupled set of PDEs that describe evolution occurring over a wide range of spatial and temporal scales. Consequently, the numerical solution of this system in time requires careful choice of temporal integration methods that can handle the severe stiffness of the problem.

Traditionally, splitting methods have addressed computational challenges present in computational combustion. In particular, the spatially discretized PDEs usually are evolved with a three-stage time advancement. First, the scalar quantities are evolved, and then the velocity field is advanced. Of the two scalar substeps, the one that contains the chemical kinetics terms is formulated as a series of homogeneous batch reactors to advance the chemistry. The remaining sub-steps advance the transport and velocity. While this approach has advantages, splitting always introduces additional computational errors and numerical remedies, such as extra iterations, which have to be used to increase the accuracy of the solution. Thus, it is desirable to explore whether other temporal integrators exist that can ensure an accurate solution while not sacrificing the efficiency of the calculation.

In this thesis, we have explored whether exponential integration can deliver computational savings in numerical combustion modeling. We studied two approaches to designing efficient exponential integration schemes for combustion

simulations. First, we applied exponential integration to evolve only the chemical kinetics portion of the dynamics. The second approach was to exponentially integrate chemical kinetics and thermodynamics jointly to avoid one of the splits in the standard integration methods.

Our first study (Chapter 3) applied a novel time-adaptive EPIRK method, EPI3V, to constant-pressure homogeneous batch reactor problems involving various chemical mechanisms representing hydrocarbon combustion. The performance of EPI3V was evaluated and found to be comparable to the widely adopted CVODE software suite, which implements an implicit variable time-stepping BDF implicit method. The results seen in homogeneous batch reactors indicated that exponential time integration methods may serve as a viable alternative to implicit methods within larger multi-stage splitting schemes commonly employed in combustion simulations. Specifically, exponential integrators could potentially replace the implicit solvers used for the sub-step dedicated to advancing the stiff chemical kinetics while leveraging their favorable stability and accuracy properties.

In the second part of the thesis (Chapter 4), we developed a computational model of the propagating flame front that involved exponentially advancing the temperature and mass fractions scalar equations without using the traditional splitting approach. The model consisted of a one-dimensional tube simulation and was solved with an exponential Euler method coupled with a Krylov-projection based algorithm to evaluate exponential-like matrix functions. Testing this numerical approach for three different chemical mechanisms and comparing it with a proven implicit-methods-based code, we found that exponential methods deliver the same qualitative behavior of the solution as the implicit method with comparable efficiency while avoiding the chemistry-thermodynamics splitting error.

By demonstrating the applicability and competitiveness of exponential time integration techniques for homogeneous reactors and propagating flame front problems, this thesis paves the way for further exploration and adoption of these methods in the broader domain of computational combustion. The findings contribute to the growing body of evidence supporting the potential benefits of exponential integrators for efficiently and accurately simulating large-scale stiff systems, such

as those arising in combustion processes.

Several future research directions are motivated by the results of this thesis. Our studies suggest that both the homogeneous batch reactor and the propagating flame front problem would benefit from using novel higher-order exponential methods coupled with advanced Krylov-projection-based algorithms. In the case of homogeneous batch reactors, higher-order adaptive time-stepping methods would be particularly beneficial due to the necessity of variable time-stepping. KIOPS offers the ability to calculate linear combinations of φ matrix-vector products, making adaptive methods relatively computationally inexpensive. The new high-order time adaptive exponential methods may offer the ability to take fewer overall time steps, thus offering considerable computational savings over lower-order methods and can be competitive with widely used implicit integrators. While adaptive time-stepping is less critical for propagating flame front problems, applying or deriving efficient high-order methods may yield improved accuracy and efficiency. Further development of adaptivity in exponential time integration of combustion models is also a potentially promising research direction. Our tests indicated that improvements can be made in the efficiency of the calculations if the adaptive algorithms are improved to minimize the number of rejected time steps, thus optimizing temporal integration.

The new exponential integration-based approach has to be tested for higher-dimensional combustion models. A simple extension of the propagating flame front problem would be to extend the model to two and three spatial dimensions and explore additional chemical mechanisms. Recent advances have been made in improving the Krylov-projection-based algorithms, particularly for massively parallel computations using low-synchronization algorithms. It would be valuable to explore whether such approaches can tackle the combustion models' high dimensionality and stiffness.

In summary, we proposed new exponential integration-based numerical methods for numerical combustion modeling. Our studies demonstrated this exponential approach is promising from the perspectives of both accuracy and efficiency improvements, and suggests new promising research directions which may deliver

further computational savings.

Bibliography

[1] C. Xu, J. Park, C. S. Yoo, J. H. Chen, T. Lu, *Identification of Premixed Flame Propagation Modes Using Chemical Explosive Mode Analysis*, Proceedings of the Combustion Institute", Volume 37, Issue 2, 2019, Pages 2407-2415, ISSN 1540-7489, https://doi.org/10.1016/j.proci.2018.07.069.

[2] C Xu, A. Y. Poludnenko, X. Zhao, H. Wang, T. Lu, *Structure of Strongly Turbulent Premixed N-Dodecane–Air Flames: Direct Numerical Simulations and Chemical Explosive Mode Analysis*, Combustion and Flame, Volume 209, 2019, Pages 27-40, ISSN 0010-2180, https://doi.org/10.1016/j.combustflame.2019.07.027.

[3] T. F. Lu, C. S. Yoo, J. H. Chen, and C. K. Law, *Three-Dimensional Direct Numerical Simulation of a Turbulent Lifted Hydrogen Jet Flame in Heated Coflow: a Chemical Explosive Mode Analysis*, Journal of Fluid Mechanics, vol. 652, pp. 45–64, 2010. doi:10.1017/S002211201000039X

[4] A. D Gomez, N. Deak, F. Bisetti, *Jacobian-free Newton–Krylov Method for the Simulation of Non-Thermal Plasma Discharges with High-Order Time Integration and Physics-Based Preconditioning*, Journal of Computational Physics, Vol 480, 2023.

[5] P.N. Brown, G.D. Byrne, and A.C. Hindmarsh, *VODE: A Variable Coefficient ODE Solver*, SIAM J. Sci. Statist. Comput. 10(5) (1989), pp. 1038–1051.

[6] G.D. Byrne and A.M. Dean, *The Numerical Solution of Some Kinetics Models with VODE and CHEMKIN II*, Comput. Chem. 17(3) (1993), pp. 297–302.

[7] D. J. Gardner, D. R. Reynolds, C. S. Woodward, and C. J. Balos. *Enabling new flexibility in the SUNDIALS suite of nonlinear and differential/algebraic equation solvers.*, ACM Transactions on Mathematical Software (TOMS) 48, no. 3 (2022): 1-24.

[8] M. J. McNenly, R. A. Whitesides, D. L. Flowers *Faster Solvers for Large Kinetic Mechanisms Using Adaptive Preconditioners*, Proceedings of the combustion institute, Volume 35, Issue 1, 2015.

[9] S. Lapointe, S. Mondal, R A. Whitesides, *Data-Driven Selection of Stiff Chemistry ODE Solver in Operator-Splitting Schemes*, Combustion and Flame,Volume 220,2020,Pages 133-143.

[10] M. Tokman, *A New Class of Exponential Propagation Iterative Methods of Runge–Kutta Type (EPIRK)*, Journal of Computational Physics Volume 230, Issue 24, 1 October 2011, Pages 8762-8778

[11] J. Loffeld, M. Tokman, *Comparative Performance of Exponential, Implicit, and Explicit Integrators for Stiff Systems of ODEs*, Journal of Computational and Applied Mathematics, Volume 241, 2013, Pages 45-67, ISSN 0377-0427, https://doi.org/10.1016/j.cam.2012.09.038.

[12] M. Narayanamurthi, P. Tranquilli, A. Sandu, M. Tokman, *EPIRK-W and EPIRK-K Time Discretization Methods*, J Sci Comput 78, 167–201 (2019). https://doi.org/10.1007/s10915-018-0761-3.

[13] M. Hochbruck, A. Ostermann, *Exponential Runge–Kutta methods for parabolic problems*, Applied Numerical Mathematics, Volume 53, Issues 2–4, 2005, Pages 323-339,ISSN 0168-9274,https://doi.org/10.1016/j.apnum.2004.08.005.

[14] H. A. van der Vorst. *Iterative Krylov Methods for Large Linear Systems*. Cambridge University Press; 2003.

[15] L. Einkemmer, M. Tokman, J. Loffeld, *On the Performance of Exponential Integrators for Problems in Magnetohydrodynamics*, Journal of Computational Physics, Volume 330, 2017, Pages 550-565, ISSN 0021-9991, https://doi.org/10.1016/j.jcp.2016.11.027.(https://www.sciencedirect.com/science/article/pii/S0021999116306131)

[16] S. Gaudreault, M. Charron, V. Dallerit, M. Tokman, *High-Order Numerical Solutions to the Shallow-Water Equations on the Rotated Cubed-Sphere Grid*. J. Comput. Phys. 449: 110792 (2022)

[17] S. Gaudreault, J. A. Pudykiewicz, *An Efficient Exponential Time Integration Method for the Numerical Solution of the Shallow Water Equations on the Sphere*. J. Comput. Phys. 322: 827-848 (2016)

[18] D. L. Michels, V. T. Luan, and M. Tokman. *A Stiffly Accurate Integrator for Elastodynamic Problems*. ACM Trans. Graph. 36, 4, Article 116, 2017. https://doi.org/10.1145/3072959.3073706

[19] V. Dallerit, M. Tokman, I. Joseph, *Exponential Integrators for Non-Linear Diffusion*, 2022 CoRR, https://doi.org/10.48550/arXiv.2207.02439 db/journals/corr/corr2207.html#abs-2207-02439

[20] T. Lu, C.K. Law *Toward Accommodating Realistic Fuel Chemistry in Large-Scale Computations*, Prog. Energy Combust. Sci., 35 (2009), pp. 192-215

[21] Z. X. Chen, I. Langella, and N. Swaminathan, *The Role of CFD in Modern Jet Engine Combustor Design*, Environmental Impact of Aviation and Sustainable Solutions. IntechOpen. 2020 doi:10.5772/intechopen.88267.

[22] R. Reitz,*Directions in Internal Combustion Engine Research*, Combustion and Flame,Volume 160, Issue 1,2013,Pages 1-8,ISSN 0010-2180

[23] T. Lu, C. K. Law, *Toward Accommodating Realistic Fuel Chemistry in Large-Scale Computations*, Progress in Energy and Combustion Science, Volume 35, Issue 2, 2009, Pages 192-215, ISSN 0360-1285,https://doi.org/10.1016/j.pecs.2008.10.002.

[24] D. Kim, J. Martz, A.Violi, *A Surrogate for Emulating the Physical and Chemical Properties of Conventional Jet Fuel*, Combustion and Flame, Volume 161, Issue 6, June 2014, Pages 1489-1498.

[25] T Kathrotia, P Oßwald, C Naumann, S Richter, M Köhler, *Combustion Kinetics of Alternative Jet Fuels, Part-II: Reaction Model for Fuel Surrogate*, Fuel, Volume 302, 2021, 120736, ISSN 0016-2361, https://doi.org/10.1016/j.fuel.2021.120736.

[26] C. Westbrook, Y. Mizobuchi, T. Poinsot, P. Smith, J. Warnatz, *Computational Combustion*, Proceedings of the Combustion Institute 30 (2005) 125–157.

[27] C. K. Law, *Combustion and a Crossroads: Status and Prospects*, Proc. Combust. Inst. 31 1–29 2006.

[28] W. Dabelstein, A. Reglitzky, A. Schütze and K. Reders *Automotive Fuels*, Ullmann's Encyclopedia of Industrial Chemistry 2007, Wiley-VCH, Weinheim.

[29] C.K. Westbrook, W.J. Pitz, M. Mehl, H.J. Curran, *Detailed Chemical Kinetic Reaction Mechanisms for Primary Reference Fuels for Diesel Cetane Number and Spark-Ignition Octane Number*, Proceedings of the Combustion Institute, Volume 33, Issue 1, 2011, Pages 185-192, ISSN 1540-7489, https://doi.org/10.1016/j.proci.2010.05.087.

[30] J. Yu, G. Tang, J. Yu, *Detailed Combustion Chemical Mechanism for Surrogates of Representative Jet Fuels*, Journal of the Energy Institute, Volume 93, Issue 6, 2020, Pages 2421-2434,ISSN 1743-9671, https://doi.org/10.1016/j.joei.2020.07.017. (https://www.sciencedirect.com/science/article/pii/S174396712030138)

[31] P. Oßwald, J. Zinsmeister, T. Kathrotia, M. Alves-Fortunato, V. Burger, R. van der Westhuizen, C. Viljoen, K. Lehto, R. Sallinen, K. Sandberg, M. Aigner, P. L. Clercq, M. Köhler, *Combustion Kinetics of Alternative Jet Fuels, Part-I: Experimental Flow Reactor Study*, Fuel, Volume 302, 2021, 120735, ISSN 0016-2361, https://doi.org/10.1016/j.fuel.2021.120735.

[32] F. Bisetti, *Integration of Large Chemical Kinetic Mechanisms via Exponential Methods with Krylov Approximations to Jacobian Matrix Functions*, Combustion theory and modeling, Vol 16, Number 3, 2012.

[33] J. Huang, R. Li, Y. He, Z. Qu, *Solutions for Variable Density Low Mach Number Flows with a Compressible Pressure-Based Algorithm*, Applied Thermal Engineering, Volume 27, Issues 11–12, 2007, Pages 2104-2112, ISSN 1359-4311, https://doi.org/10.1016/j.applthermaleng.2006.11.010. (https://www.sciencedirect.com/science/article/pii/S135943110600408X)

[34] A. A. Lukassen, M. Kiehl, *Operator Splitting for Chemical Reaction Systems with Fast Chemistry*, Journal of Computational and Applied Mathematics, 344, (2018) 495–511 https://doi.org/10.1016/j.cam.2018.06.001.

[35] S. Descombes, M. Duarte, M. Massot, *Operator Splitting Methods with Error Estimator and Adaptive Time-Stepping. Application to the Simulation of Combustion Phenomena*, In: R. Glowinski, , S. Osher, W. Yin, (eds) "Splitting Methods in Communication, Imaging, Science, and Engineering Scientific Computation.", pp 627–641, Springer, Cham. https://doi.org/10.1007/978-3-319-41589-5_19

[36] B. Savard, Y. Xuan, B. Bobbitt, G. Blanquart, *A Computationally-Efficient, Semi-Implicit, Iterative Method for the Time-Integration of Reacting Flows with Stiff Chemistry*, Journal of Computational Physics, Volume 295, 2015, Pages 740-769, ISSN 0021-9991, https://doi.org/10.1016/j.jcp.2015.04.018.

[37] Jonathan F. MacArt, Michael E. Mueller, *Semi-Implicit Iterative Methods for Low Mach Number Turbulent Reacting Flows: Operator Splitting Versus Approximate Factorization*, Journal of Computational Physics, Volume 326, 2016, Pages 569-595, ISSN 0021-9991, https://doi.org/10.1016/j.jcp.2016.09.016.

[38] C. D. Pierce, *Progress-Variable Approach for Large-Eddy Simulation of Turbulent Combustion*, Ph.D dissertation, Stanford University, 2001.

[39] Olivier Desjardins, Guillaume Blanquart, Guillaume Balarac, Heinz Pitsch, *High Order Conservative Finite Difference Scheme for Variable Density Low Mach Number Turbulent Flows*, Journal of Computational Physics, Volume 227, Issue 15, 2008, Pages 7125-7159, ISSN 0021-9991, https://doi.org/10.1016/j.jcp.2008.03.027.

[40] D.A. Pope, *An Exponential Method of Numerical Integration of Ordinary Differential Equations* Commun. ACM, 6 (1963), pp. 491-493

[41] M. Cleve and L. Charles, *Nineteen Dubious Ways to Compute the Exponential of a Matrix, Twenty-Five Years Later*, SIAM Review 2003 45:1, 3-49.

[42] M. Tokman, P.M. Bellan *Three-Dimensional Model of the Structure and Evolution of Coronal Mass Ejections* The Astrophysical Journal, 567:1202-1210, 2002 March 10

[43] J. Stewart, M. Tokman, V. Dallerit, F. Bisetti, O. Diaz-Ibarra. *Variable Time-stepping Exponential Integrators for Chemical Reactors with Analytical Jacobians*. Applied and Computational Mathematics, 13(2), 29-37, 2024. https://doi.org/10.11648/j.acm.20241302.11

[44] N. J. Higham. *The Scaling and Squaring Method for the Matrix Exponential Revisited*. SIAM J. Matrix Anal. Appl. 26, 4 (2005), 1179–1193. https://doi.org/10.1137/04061101X

[45] R. B. Sidje. *Expokit: A Software Package for Computing Matrix Exponentials*, ACM Trans. Math. Softw., 24(1):130-156, 1998.

[46] J. Niesen, W. M. Wright *Algorithm 919: A Krylov subspace algorithm for evaluating the φ-functions appearing in exponential integrators.*, ACM Transactions on Mathematical Software (TOMS), 38(3), 1-19, 2012.

[47] S. Gaudreault, G. Rainwater, M. Tokman, *KIOPS: A fast adaptive Krylov subspace solver for exponential integrators*, Journal of Computational Physics, Volume 372, 2018, Pages 236-255, ISSN 0021-9991, https://doi.org/10.1016/j.jcp.2018.06.026.

[48] F. Bisetti, *High-order Methods for the Simulation of Unsteady Counterflow Flames Subject to Stochastic Forcing of Large Amplitude*, Combustion Theory and Modelling, 2023 DOI: 10.1080/13647830.2023.2218621.

[49] https://github.com/sandialabs/TChem

[50] D. G. Goodwin, H. K. Moffat, I. Schoegl, R. L. Speth, B. W. Weber. *Cantera: An Object-Oriented Software Toolkit for Chemical Kinetics, Thermodynamics, and Transport Processes.*, https://www.cantera.org, 2023. Version 3.0.0. doi:10.5281/zenodo.8137090

[51] R. J. Kee, F. M. Rupley, and J. A. Miller, *Chemkin-II: A Fortran Chemical Kinetics Package for the Analysis of Gas-Phase Chemical Kinetics*, Sandia National Laboratories, 1989, SAND-89-8009.

[52] G. P. Smith, D. M. Golden, M. Frenklach, N. W. Moriarty, B. Eiteneer, M. Goldenberg, C. T. Bowman, R. K. Hanson, S. Song, W. C. Gardiner, Jr., V. V. Lissianski, Z. Qin http://www.me.berkeley.edu/gri_mech/

[53] N.M Marinov, W.J. Pitz, C.K. Westbrook, A. M. Vincitore, M.J. Castaldi, S. M. Senkan, *Aromatic and Polycyclic Aromatic Hydrocarbon Formation in a Laminar Premixed n-Butane Flame*, Combustion and Flame 114 192-213 (1998).

[54] Z. Luo., S. Som , S.M. Sarathy, M. Plomer, W.J. Pitz, D.E. Longman, T.F. Lu , *Development and Validation of an n-Dodecane Skeletal Mechanism for Diesel Spray-Combustion Applications*, Combust. Theory Model., DOI: 10.1080/13647830.2013.872807, 2014

[55] S. C. Moldoveanu, *Pyrolysis of Organic Molecules (Second Edition)*, Elsevier,2019,ISBN 9780444640000.

[56] K. Kim, O. Diaz-Ibarra, H. Najm, J. Zádor, and C. Safta. *TChem: A Performance Portable Parallel Software Toolkit for Complex Kinetic Mechanisms*, Computer Physics Communications, Vol 285, 2023, DOI: https://doi.org/10.1016/j.cpc.2022.108628.

[57] J. Loffeld, and M. Tokman. *Implementation of Parallel Adaptive-Krylov Exponential Solvers for Large Scale Stiff Problems*. SIAM Journal on Scientific Computing, Vol 36, Iss 5, 2014.

[58] G. Wanner, E. Hairer, S. P. Norsett, *Solving Ordinary Differential Equations I: Nonstiff Problems.*, Germany: Springer Berlin Heidelberg, 2013.

[59] A. Kassam, L. N.Trefethen, *Fourth-Order Time-Stepping for Stiff PDEs*, SIAM Journal on Scientific Computing, 26(4),pages 1214-1233, 2005.

[60] A. G. Tomboulides, J. C. Y. Lee, S. A. Orszag, *Numerical Simulation of Low Mach Number Reactive Flows.*, Journal of Scientific Computing 12, 139–167 (1997). https://doi.org/10.1023/A:1025669715376

[61] R. L. Speth, W. H. Green, S. Macmara, G. Strang, *Balanced Splitting and Rebalanced Splitting*. SIAM Journal on Numerical Analysis, 51(6), 2013, 3084–3105. http://www.jstor.org/stable/24511532

[62] R.A. Yetter, F.L. Dryer, and H. Rabitz, *A comprehensive reaction mechanism for carbon monoxide/hydrogen/oxygen kinetics*, Comb. Sci. & Tech., vol. 79, pp. 97-128 (1991)

[63] A. Stagni, C. Cavallotti, S. Arunthanayothin, Y. Song, O. Herbinet, F. Battin-Leclerc, T. Faravelli, *An Experimental, Theoretical and Kinetic-Modeling Study of the Gas-Phase Oxidation of Ammonia*, Reaction Chemistry and Engineering, 2020.

[64] J. E. Welch, F.H. Harlow, J. P. Shannon, B.J. Daly, B J.*The Mac Method-A Computing Technique for Solving Viscous, Incompressible, Transient Fluid-Flow Problems Involving Free Surfaces*. Los Alamos National Laboratory, 1965. United States. https://doi.org/10.2172/4563173. https://www.osti.gov/servlets/purl/4563173.

[65] K. Kim, O.. Diaz-Ibarra, C. Safta, H. Najm, *TChem v3.0 - A Software Toolkit for the Analysis of Complex Kinetic Models*, Sandia National Laboratories, SAND 2021-14064, 2021.

[66] M. H. Kasbaoui, D. L. Koch, G. Subramanian, and O. Desjardins, *Preferential Concentration Driven Instability of Sheared Gas–Solid Suspensions*, Journal of Fluid Mechanics, vol. 770, pp. 85–123, 2015. doi:10.1017/jfm.2015.136

[67] J. Capecelatro, O. Desjardins, and R. O. Fox, *Numerical Study of Collisional Particle Dynamics in Cluster-Induced Turbulence*, Journal of Fluid Mechanics, vol. 747, p. R2, 2014. doi:10.1017/jfm.2014.194

[68] F. Bisetti, "Chemjac 1.0", 2022, https://doi.org/10.18738/T8/700QAK, Texas Data Repository, V1.