UNIVERSITY OF CALIFORNIA

Los Angeles

Provably correct optimization and estimation: continuous, discrete, and dynamical

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Electrical and Computer Engineering

by

Jonathan Michael Bunton

2023

ABSTRACT OF THE DISSERTATION

Provably correct optimization and estimation:
continuous, discrete, and dynamical

by

Jonathan Michael Bunton
Doctor of Philosophy in Electrical and Computer Engineering
University of California, Los Angeles, 2023
Professor Paulo Tabuada, Chair

Many engineering tasks are, at their core, a series of large, high-stakes decision-making problems. Generally we are faced with some issue to resolve, then asked to incorporate all of the relevant information, context, and data to produce the best solution possible. Often we can only determine if we made an optimal choice in hindsight, but we generally take comfort in knowing that every decision we have made so far is the best possible given all of our current information.

When synthesizing the information, context, and data into a decision-making problem, we are implicitly creating an optimization problem, asking ourselves to determine the option that satisfies our specifications while performing the best, according to some chosen metric. While many tasks may be framed this way, the resulting optimization problems are not necessarily computationally tractable.

The first part of this thesis considers one such class of typically intractable optimization problems: problems with joint continuous and discrete decision variables. This class of problems is NP-Hard to solve in general, but we show that by leveraging *submodularity*,

a property of functions over partially-ordered sets, we can identify a new special subset for which we provide provably exact algorithms that run in polynomial time. In the larger, decision-making context, this result should provide the trepidatious engineer with confidence that, given all the data, constraints, and information they have at the current moment, they have selected the best possible solution.

Many optimization problems still fall outside of this special subset, where the functions involved are not submodular. We address part of this issue by showing how some problems outside this class–in particular, quadratic optimization problems with combinatorial regularizers–may be approximately solved by instead solving a suitably chosen surrogate problem from within our previously identified subset. The suboptimality of this approach is then naturally bounded by the distance between the original optimization problem and our class of submodular ones.

The second part of this thesis considers nonlinear state estimation. In this scenario, we collect measurements from a nonlinear system (e.g., a mobile robot), and from the knowledge of the system's dynamics and these measurements are asked to estimate the system's state as accurately as possible. In this dynamic estimation context, we are faced with a sequence of these optimization problems (select the best choice of state), each closely related to the previous.

Feedback control typically relies on such an estimate of the system state provided by an estimation scheme. These estimates, however, are always affected by errors that have non-negligible impacts on control performance. Various stabilizing and safety-critical control frameworks address this issue, but all require some characterization of the current estimation error to determine when to apply more or less conservative control inputs. Current methods of bounding these errors either take a very coarse worst-case bound or employ computationally expensive time-varying set-valued methods.

To tackle this problem, we turn to a state estimation scheme based on polynomial least-squares, termed *Savitzky-Golay* filtering. This scheme relies on approximating the output

of the system and its derivatives via polynomial least-squares, then using information about the system dynamics to convert these derivatives into an estimate of the system state. Our analysis presents a new, *online* error bound that highlights the connection between the suboptimality of the optimization problem's solution and the quality of the state estimate. In our analysis, we show several intuitive properties of these bounds, with the main intuition that when the system dynamics are well-behaved and the measurements are noiseless, the function approximation task becomes easier and the guarantees tighten.

Further, these error bounds provide an online, deterministic measure of uncertainty, which a downstream control algorithm can use to adapt its levels of robustness in real-time. This particular interaction appeals to the simple intuition that a robot should only make aggressive maneuvers when it is highly confident in its current position. The frameworks of measurement-robust control barrier functions and robust control Lyapunov functions in particular are immediate candidates for this type of interface, as they would naturally accommodate the estimation error while maintaining safety and stability guarantees.

The dissertation of Jonathan Michael Bunton is approved.

Lieven Vandenberghe

Bahman Gharesifard

Christina Fragouli

Paulo Tabuada, Committee Chair

University of California, Los Angeles

2023

To my family and the friends close enough to call family:

both I and this thesis owe you our lives.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

This thesis would be incomplete if I did not take a moment to thank the countless people who made it possible. Perhaps surprisingly, this section has been the hardest for me to put into words, mostly because I can't seem to find the right words to express to quite enough people in muy life how thankful I am for all of their help and support. I can only hope the meager attempt here conveys some fraction of my gratitude.

First and foremost, I have to thank my advisor, Professor Paulo Tabuada. I don't know if I can ever thank you enough for taking a chance on the awkward physics graduate from Tennessee all those years ago. You changed my life, and I am the scientist and researcher I am today because of your encouragement, instruction, and deep patience for my nonsense. You have a way of attacking problems that cuts quick to the core that I've been trying to mimic since meeting you. Thank you, from the bottom of my heart, for helping me become who I am today.

I must also acknowledge my absolutely wonderful committee. To Professor Vandenberghe, you may not know, but I owe you my research trajectory: it was you, during my very first year, that provided me with the first references that launched me into my study of joint continuous and discrete optimization. I enjoyed your classes so much that I was branded our lab's "optimization guy" when there was a need. Professor Gharesifard, it's a running joke that you play the role of "cool uncle" to CyPhy Lab, and we mean this in the most heartfelt way possible. You've been there for me both to mold me into a better researcher and to listen as I agonize over the future. Professor Fragouli, you have always been an incredibly gentle, thoughtful, and kind voice and I can't think of a person I would rather have bookend my graduate studies: both at the very beginning, as the instructor of my first graduate school course, and the end, as a member of this committee.

I could write an entire six-page paper (with references, IEEE format) on the incredible group of students I've been able to work with during my five and a half years in CyPhy

Lab. It was my goal and mission when starting graduate school to make sure we were more than co-workers, but friends! Miraculously, everyone made this mission incredibly easy by already being an incredibly warm, welcoming, and tight-knit group of friends to begin with. Thank you to each and every one of you: Alimzhan Sultangazin and Tzanis Anevlavis, for providing a flawless example of how to be an outstanding researcher while still having fun; Lucas Fraile, for re-introducing me to D&D and being a source of "reckless confidence"; Yanwen Mao, for your simple and straightforward friendship; Luigi Pannochi, for the years of work putting together a great testbed; Yskandar Gas, for teaching us all to relax and enjoy our time together; and Matteo Marchi, for being the best collaborator and friend I could have asked for. Again, I could write essays singing each of your praises, but just know that I could not have asked for a better group of lab-mates and friends during these years. I love each of you, and I look forward to more years to come.

I need to thank my family for completing the arduous task of raising me into the type of person who gets a PhD. To my dad, thank you for always pushing me to try hard and to not shy away from hard work. To my mom, thank you for teaching me to love reading, learning, and maybe most of all, living. To both of you, thanks for allowing me to spend the last 25 years of my life in school. Sammy, I am and always will be proud to be your brother. And to all the family–I'm always so thankful for your constant love and support.

Lastly, I want to thank Clarissa Morales. Both I and this thesis simply would not be here today if not for you. You have been there for every high, low, and in-between experience of this process, and I am so grateful for the patient understanding and the necessary kicks in the rear along the way. You have put me back together when I needed it most. I am so glad to have you in my life, and I am a better person for having known you.

VITA

2018	B.S. Physics, Applied Mathematics (double major), Austin Peay State University, Clarksville, TN, USA.
2019	M.S. Electrical and Computer Engineering University of California, Los Angeles, CA, USA.
2019-2023	PhD Candidate, Department of Electrical and Computer Engineering, University of California, Los Angeles, CA, USA.

PUBLICATIONS

M. Marchi, **J. Bunton**, Y. Gas, B. Gharesifard, and P. Tabuada, "Sharp performance bounds for PASTA," *Control Systems Letters (L-CSS)*, to appear, 2023.

J. Bunton and P. Tabuada, "Joint continuous and discrete model selection via submodularity," *Journal of Machine Learning Research*, vol. 23, no. 329, pp. 1-42, 2022.

J. Bunton and P. Tabuada, "IoBT resource allocation via mixed discrete and continuous optimization," *IoT for Defense and National Security*, pp. 39-57, 2022.

J. Bunton and P. Tabuada, "Give the problem a lift: solving quadratic programs with combinatorial costs," *61st Conference on Decision and Control (CDC)*, pp. 6941-6946, 2022.

M. Marchi, **J. Bunton**, B. Gharesifard, and P. Tabuada, "Lidar point cloud registration with formal guarantees," *61st Conference on Decision and Control (CDC)*, pp. 3462-3467, 2022.

M. Marchi, **J. Bunton**, B. Gharesifard, and P. Tabuada, "Safety and stability guarantees for control loops with deep learning perception," *Control Systems Letters (L-CSS)*, pp. 1286-1291, 2022.

J. Bunton, T. Anevlavis, G. Verma, and P. Tabuada, "Split to win: near-optimal sensor network synthesis via path-greedy subproblems," *IEEE Military Communications Conference (MILCOM)*, pp. 789-794, 2021.

P. Ghosh, **J. Bunton**, D. Pylorof, M. Vieira, K. Chan, R. Govindan, G. Suhatme, P. Tabuada, and G. Verma, "Synthesis of Large-Scale Instant IoT Networks," *Transactions on Mobile Computing (TMC)*, 2021.

P. Ghosh, **J. Bunton**, D. Pylorof, M. Vieira, K. Chan, R. Govindan, G. Suhatme, P. Tabuada, and G. Verma, "Rapid Top-Down Synthesis of Large-Scale IoT Networks," *International Conference on Computer Communications and Networks (ICCCN)*, pp. 1-9, 2020.

J. Bunton and P. Tabuada, "Why not both? Exact continuous and discrete optimization with submodularity," *59th Conference on Decision and Control (CDC)*, pp. 4842-4847, 2020.

T. Anevlavis, **J. Bunton** A. Parayil, J. George, and P. Tabuada, "To beam or not to beam? Beamforming with submodularity-inspired group sparsity," *59th Conference on Decision and Control (CDC)*, pp. 390-395, 2020.

Part I

Optimization

CHAPTER 1

Summary

In model selection problems for machine learning, the desire for a well-performing model with meaningful structure is typically expressed through a regularized optimization problem. In many scenarios, however, the meaningful structure is specified in some discrete space, leading to difficult nonconvex optimization problems. In this part of the thesis, we connect the model selection problem with structure-promoting regularizers to submodular function minimization with continuous and discrete arguments. In particular, we leverage the theory of submodular functions to identify a class of these problems that can be solved exactly and efficiently with an agnostic combination of discrete and continuous optimization routines. We show how simple continuous or discrete constraints can also be handled for certain problem classes, and extend these ideas to a robust optimization framework. We also show how some problems outside of this class can be embedded within the class, further extending the class of problems our framework can accommodate. Finally, we numerically validate our theoretical results with several proof-of-concept examples with synthetic and real-world data, comparing against state-of-the-art algorithms.

CHAPTER 2

Introduction

In many machine learning tasks, we require a model that not only performs a specified task well, but also has some meaningful structure. Models with meaningful structure can, for example, be easier to understand and implement. The desire for both accuracy and meaningful structure is usually expressed in a regularized optimization problem:

$$\underset{\mathbf{x}\in\mathcal{X}}{\text{minimize}}\ f(\mathbf{x}) + \lambda g(\mathbf{x}). \tag{2.0.1}$$

In this problem, $\mathbf{x}$ is a choice of model parameters from a parameter space $\mathcal{X}$, $f : \mathcal{X} \to \mathbb{R}$ is a function that describes the misfit of the model with the selected parameters to the given task (e.g., empirical risk), $g : \mathcal{X} \to \mathbb{R}$ is a function that expresses the deviation of our selected model parameters from some desired structure, and $\lambda \in \mathbb{R}_{\geq 0}$ is a tradeoff parameter.

Problem (2.0.1) becomes difficult when the desired model structure is an inherently discrete property, but the model parameters are continuous values $\mathbf{x}$ from a continuum $\mathcal{X}$. A prime example of this issue arises in feature selection for sparse regression, where we seek a linear predictor $\mathbf{x}^* \in \mathcal{X} \subseteq \mathbb{R}^n$ such that:

$$\mathbf{x}^* \in \underset{\mathbf{x}\in\mathcal{X}}{\text{argmin}}\ \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_0, \tag{2.0.2}$$

for some $\mathbf{A} \in \mathbb{R}^{m\times n}$ and $\mathbf{b} \in \mathbb{R}^m$, with $\|\mathbf{x}\|_2$ the standard Euclidean norm on $\mathbb{R}^m$, and $\|\mathbf{x}\|_0$ the ℓ_0 pseudo-norm that counts the number of nonzero entries in the predictor $\mathbf{x}$. The desired structure, in this case, is a sparse predictor $\mathbf{x} \in \mathcal{X}$. Sparsity, however, only depends on the combinatorial choice of zero entries in the model parameters $\mathbf{x}$, whereas the model also requires a choice of continuous values for $\mathbf{x} \in \mathcal{X}$.

Problems with this mixed dependence on both continuous and discrete properties of the model parameters such as (2.0.2) are notoriously difficult, and even NP-Hard in general [Rau10]. A typical workaround is to replace the function describing model structure, g in problem (2.0.1), with a continuous relaxation that is more amenable to optimization. One of the more celebrated instances of this approach is the relaxation of the ℓ_0 pseudo-norm in (2.0.2) to the convex ℓ_1 norm $\|\mathbf{x}\|_1$, which instead sums the absolute values of the vector $\mathbf{x}$. While this relaxation still encourages the intended structure, the minimizer for the relaxed problem does not necessarily correspond to the minimizer for the initially specified problem [BJM12]. Moreover, the well-known conditions for sparse recovery in regression problems, such as Restricted Isometry Properties [CT05], Null Space Properties [Rau10], and Irrepresentability Conditions [ZY06], are not applicable to more general discrete functions g.

In contrast, in this work we identify conditions that allow us to directly solve the originally posed regularized model-fitting problem (2.0.1) exactly and efficiently. To derive our new conditions, we leverage submodularity, a property of functions that defines a boundary between easy and hard optimization problems. Our approach stands in stark contrast to existing methods, which either focus on submodularity in purely one domain [Bac19] or relies on restricted isometry or strong convexity constants that are NP-Hard to compute [EKD18, EJ20].

Traditionally, submodularity is defined for functions on bounded discrete sets, where arbitrary function minimization is NP-Hard. When a function is submodular, however, it can be minimized exactly in polynomial time [Sch03]. The definition of submodularity extends to continuous functions as well, and recently the associated optimization guarantees have also been extended [Bac19, BLK17]. In particular, if a continuous function is submodular, it can also be minimized exactly in polynomial time.

The natural next question–which is addressed in this work–to ask is if submodularity still defines a boundary between easy and hard *mixed* optimization problems such as (2.0.1),

where the function f in (2.0.1) is continuous, but the function g has a discrete co-domain. Our work explores this boundary and identifies sufficient conditions, based on the submodularity of both functions, under which the exact solution of problem (2.0.1) can be efficiently computed.

Exploiting submodularity in these mixed scenarios is not a new idea, given its utility in discrete optimization problems. Notable uses include establishing approximation guarantees for greedy algorithms applied to sparsity-constrained optimization [EKD18], or in producing tight convex relaxations for set-function descriptions of desired sparsity patterns [BJM12].

As highlighted above, [Bac19] shows that if a continuous function is submodular, it can be *discretized* into a discrete submodular function, which can then be minimized exactly in polynomial time. However, this discretization is only valid for compact subsets of continuous spaces and necessarily introduces discretization error into the produced solution.

In a line of work similar to this one, authors in [EJ20] propose converting the mixed problem to a purely discrete one without discretizing. They then advocate using a specific submodular set function minimization algorithm for solving the discrete problem, and give approximation guarantees under the assumption that the functions are nearly submodular. Our proposed approach is similar, but our work instead focuses on finding conditions under which an *arbitrary choice* (of potentially more efficient) algorithms produce *exact* results, which leads to their choice as a special case.

The sufficient conditions we require may be violated in practice. Traditionally, violations of submodularity are handled by suitably relaxing the definition with an additive or multiplicative constant and propagating the constant through a particular algorithm [EJ20, EKD18]. Alternatively, in this work we find a sub-class of optimization problems that we can always lift into problems that satisfy our assumptions. Moreover, we prove that the solution of the lifted problem gives a near-optimal solution to the original. Our lifting approach stands in stark contrast to existing methods, as it is algorithm-independent with a guarantee that is easy to compute rather than tied to a specific algorithm and dependent

on constants that are NP-Hard to compute [EJ20, EKD18].

We make several technical contributions, namely:

(i) We identify new sufficient conditions, based on submodularity, under which the regularized model selection problem (2.0.1) can be solved efficiently and exactly;

(ii) We extend this theory to accommodate simple continuous and discrete constraints on the model parameter for some problem classes;

(iii) We highlight the utility of exact solutions for robust optimization scenarios;

(iv) We show that problems violating our sufficient conditions can be lifted to problems that do satisfy them, and whose solutions correspond to optimal or near-optimal solutions of the original problem;

(v) We numerically validate the correctness of our theory with examples from sparse regression and retail price optimization.

2.1 Submodular Functions on Lattices

In this work, we consider optimization problems defined on two sets: an uncountably infinite set, typically $\mathbb{R}^n$ or a subset thereof referred to as a *continuous set*, and a countable set, typically finite and referred to as a *discrete set*. Because we would like to efficiently solve optimization problems defined on both continuous and discrete sets, we study a structure that can allow efficient optimization in both cases: submodularity.

Submodularity is typically defined as a property of set functions, which are functions that map any subset of a finite set V to a real number, i.e., $f : 2^V \to \mathbb{R}$. More generally, however, submodularity is a property of functions on *lattices* which can be continuous or discrete sets.

Let $\mathcal{X}$ be a set equipped with a partial order of its elements, denoted by $\preceq$. For any two elements $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$ we define their least upper bound, or *join*, as:

$$\mathbf{x} \curlyvee \mathbf{x}' = \inf\{\mathbf{y} \in \mathcal{X} \ : \ \mathbf{x} \leq \mathbf{y}, \ \mathbf{x}' \leq \mathbf{y}\}. \tag{2.1.1}$$

Dually, we define their greatest lower bound, or *meet*, as:

$$\mathbf{x} \curlywedge \mathbf{x}' = \sup\{\mathbf{y} \in \mathcal{X} \ : \ \mathbf{y} \leq \mathbf{x}, \ \mathbf{y} \leq \mathbf{x}'\}. \tag{2.1.2}$$

If for any two elements $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$, their join, $\mathbf{x} \curlyvee \mathbf{x}'$, and their meet, $\mathbf{x} \curlywedge \mathbf{x}'$, exist and are in $\mathcal{X}$, then the set $\mathcal{X}$ and its order define a *lattice*. We write the lattice and its partial order together as $(\mathcal{X}, \preceq)$, but will often write just $\mathcal{X}$ when the order is clear from context. If a subset $\mathcal{S} \subseteq \mathcal{X}$ is such that for any two of its elements $\mathbf{x}, \mathbf{x}' \in \mathcal{S}$, both their join, $\mathbf{x} \curlyvee \mathbf{x}'$, and their meet, $\mathbf{x} \curlywedge \mathbf{x}'$, are in $\mathcal{S}$, the subset $\mathcal{S}$ is called a *sublattice* of $\mathcal{X}$ [DP02].

As an example, consider a finite set of elements V. Then its power set, 2^V (the set of all its possible subsets), forms a lattice when ordered by set inclusion, $\subseteq$. Under this order, the join of any two elements $X, X' \subseteq V$ is their set union, $X \cup X' \subseteq V$, and dually, their meet is their set intersection $X \cap X' \subseteq V$.

We can also endow continuous sets with partial orders that define lattices. Recent work has brought attention to $\mathbb{R}^n$ equipped with the partial order $\preceq$, defined as:

$$\mathbf{x} \preceq \mathbf{x}' \quad \Leftrightarrow \quad \mathbf{x}_i \leq \mathbf{x}'_i \quad \text{for all } i = 1, 2, ..., n, \tag{2.1.3}$$

where $\leq$ denotes the usual order on $\mathbb{R}$.

Under this order, the join and meet operation for any two elements $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^n$ are element-wise maximum and minimum, respectively, meaning:

$$(\mathbf{x} \curlyvee \mathbf{x}')_i = \max\{\mathbf{x}_i, \mathbf{x}'_i\}, \ \text{for all } \ i = 1, 2, ..., n, \tag{2.1.4}$$

$$(\mathbf{x} \curlywedge \mathbf{x}')_i = \min\{\mathbf{x}_i, \mathbf{x}'_i\}, \ \text{for all } \ i = 1, 2, ..., n. \tag{2.1.5}$$

Given a lattice $\mathcal{X}$, consider a function $f : \mathcal{X} \to \mathbb{R}$. The function f is *submodular* on the lattice $\mathcal{X}$ when the following inequality holds for all $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$:

$$f(\mathbf{x}) + f(\mathbf{x}') \geq f(\mathbf{x} \curlyvee \mathbf{x}') + f(\mathbf{x} \curlywedge \mathbf{x}'). \tag{2.1.6}$$

The function f is *monotone* when it satisfies:

$$\mathbf{x} \preceq \mathbf{x}' \quad \implies \quad f(\mathbf{x}) \leq f(\mathbf{x}'). \tag{2.1.7}$$

When working with the lattice $(2^V, \subseteq)$, the submodular inequality (2.1.6) becomes:

$$f(A) + f(B) \geq f(A \cup B) + f(A \cap B) \quad \text{for all } A, B \subseteq V. \tag{2.1.8}$$

Similarly, the monotonicity implication (2.1.7) becomes:

$$A \subseteq B \quad \implies \quad f(A) \leq f(B). \tag{2.1.9}$$

Minimizing or maximizing an arbitrary set function is NP-Hard in general. If the set function is submodular, however, it can be exactly minimized and approximately maximized (up to a constant-factor approximation ratio) in polynomial time [Sch03, NWF78]. The computational tractability of submodular optimization for set functions has a variety of applications in countless fields such as sparse regression, summarization, and sensor placement [EKD18, LB11, KGG06].

When working with the lattice $(\mathbb{R}^n, \preceq)$, a function $f : \mathbb{R}^n \to \mathbb{R}$ is submodular when:

$$f(\mathbf{x}) + f(\mathbf{x}') \geq f(\max\{\mathbf{x}, \mathbf{x}'\}) + f(\min\{\mathbf{x}, \mathbf{x}'\}) \quad \text{for all } \mathbf{x}, \mathbf{x}' \in \mathbb{R}^n, \tag{2.1.10}$$

where the maximum and minimum operations are performed element-wise, as expressed in (2.1.4) and (2.1.5). When f is twice differentiable, submodularity on $\mathbb{R}^n$ is equivalent (see [Top98, Bac19]) to the condition:

$$\frac{\partial^2 f}{\partial \mathbf{x}_i \partial \mathbf{x}_j} \leq 0 \quad \text{for all } i \neq j. \tag{2.1.11}$$

Perhaps surprisingly, the guarantees associated with submodular set function optimization extend to functions that are submodular on $\mathbb{R}^n$. In particular, submodular functions on $\mathbb{R}^n$ can be minimized over a bounded sublattice in polynomial time (see [Bac19]), and can be approximately maximized with constant-factor approximation ratios [BMB16, BLK17].

2.2 Problem Formulation

In this section, we bridge continuous and discrete submodular function minimization in one unified problem statement. We do this by drawing inspiration from the field of structured sparsity, where the choice of zero entries in real-valued decision variables is viewed as a coupled discrete and continuous problem [Bac13, Bac11].

To highlight the connection with structured sparsity problems, for $n \in \mathbb{Z}_{>0}$, we denote by $[n]$ the set $\{1, 2, ..., n\}$, and by $2^{[n]}$ the set of all possible subsets of $[n]$. Define the map $\text{supp} : \mathbb{R}^n \to 2^{[n]}$ as:

$$\text{supp}\,(\mathbf{x}) = \{i \in [n] \mid \mathbf{x}_i \neq 0\}. \tag{2.2.1}$$

In words, supp returns the set of indices where the vector $\mathbf{x}$ is nonzero. Consider arbitrary functions $f : \mathbb{R}^n \to \mathbb{R}$ and $g : 2^{[n]} \to \mathbb{R}$. Problems of the form:

$$\underset{\mathbf{x}\in\mathbb{R}^n}{\text{minimize}} \quad f(\mathbf{x}) + g(\text{supp}\,(\mathbf{x})), \tag{2.2.2}$$

often arise in structured sparse optimization, where the preferences in discrete selections (the zero entries of $\mathbf{x}$) are expressed through the function g. As a special case, if we let $f(\mathbf{x}) = \|\mathbf{D}\mathbf{x} - \mathbf{b}\|_2^2$ with $\mathbf{D} \in \mathbb{R}^{m\times n}$ and $\mathbf{b} \in \mathbb{R}^m$ and define $g(A) = |A|$ as the cardinality of the set A, (2.2.2) becomes:

$$\underset{\mathbf{x}\in\mathbb{R}^n}{\text{minimize}} \quad \|\mathbf{D}\mathbf{x} - \mathbf{b}\|_2^2 + \|\mathbf{x}\|_0, \tag{CS}$$

where $\|\cdot\|_0$ denotes the ℓ_0 pseudo-norm. The problem (CS) is a form of the well-studied compressed sensing problem, which is NP-Hard in general [Rau10].

Generalizing the idea of making continuous decisions through the choice of $\mathbf{x}$ in (2.2.2), and discrete decisions through the choice of the zero entries of $\mathbf{x}$, we consider two lattices, $(\mathcal{X}, \preceq)$ and $(\mathcal{Y}, \sqsubseteq)$, related by a map $\eta : \mathcal{X} \to \mathcal{Y}$. We let $f : \mathcal{X} \to \mathbb{R}$ be a function describing the cost of assignments of variables in $\mathcal{X}$, and similarly let $g : \mathcal{Y} \to \mathbb{R}$ describe the associated cost of choices in $\mathcal{Y}$. Then, we seek the optimal point $\mathbf{x}^* \in \mathcal{X}$ in the problem:

$$\underset{\mathbf{x} \in \mathcal{X}}{\text{minimize}} \quad f(\mathbf{x}) + g(\eta(\mathbf{x})). \tag{P}$$

Although we will eventually let $\mathcal{X}$ describe continuous choices and $\mathcal{Y}$ describe associated discrete ones, our theoretical results do not rely on the cardinality of the lattices $\mathcal{X}$ and $\mathcal{Y}$.

Intuitively, problem (P) asks for the element $\mathbf{x} \in \mathcal{X}$ which incurs minimum cost in $\mathcal{X}$, as measured by $f(\mathbf{x})$, and in $\mathcal{Y}$, as measured by $g(\eta(\mathbf{x}))$. Given that the special case of (CS) is already hard in general, with no additional structure on f, g and η, this problem is hopelessly difficult. To provide the necessary structure, we make the following assumptions.

Assumptions. *Consider the lattices $(\mathcal{X}, \preceq)$ and $(\mathcal{Y}, \sqsubseteq)$ and the maps $\eta : \mathcal{X} \to \mathcal{Y}$, $f : \mathcal{X} \to \mathbb{R}$, and $g : \mathcal{Y} \to \mathbb{R}$. We make the following assumptions:*

1. *The functions f and g are submodular on the lattices $\mathcal{X}$ and $\mathcal{Y}$, respectively,*
2. *The function g is monotone on $\mathcal{Y}$,*
3. *For all $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$:*

$$\eta(\mathbf{x} \curlyvee \mathbf{x}') \sqsubseteq \eta(\mathbf{x}) \sqcup \eta(\mathbf{x}'), \quad \eta(\mathbf{x} \curlywedge \mathbf{x}') \sqsubseteq \eta(\mathbf{x}) \sqcap \eta(\mathbf{x}').$$

Remark 1. If the map $\eta : \mathcal{X} \to \mathcal{Y}$ satisfies Assumption 3, it is an order-preserving join-homomorphism, meaning it maintains the order and joins of elements in $\mathcal{X}$. (Prop. 2.19 in [DP02]) Explicitly, Assumption 3 is equivalent to the condition that for any $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$:

$$\mathbf{x} \preceq \mathbf{x}' \Rightarrow \eta(\mathbf{x}) \sqsubseteq \eta(\mathbf{x}'),$$
$$\eta(\mathbf{x} \curlyvee \mathbf{x}') = \eta(\mathbf{x}) \sqcup \eta(\mathbf{x}').$$

Despite this equivalence, we leave Assumption 3 as written above for clarity in future proofs.

We highlighted the lattices $(\mathbb{R}^n, \preceq)$ and $(2^{[n]}, \subseteq)$, but for the map supp : $\mathbb{R}^n \to 2^{[n]}$ to satisfy Assumption 3, we must restrict the domain of f to only only the first orthant, $(\mathbb{R}^n_{\geq 0}, \preceq)$. As mentioned by [BLK17], this issue can often be resolved by considering an appropriate *orthant conic lattice*, which views $\mathbb{R}^n$ as a product of n copies of $\mathbb{R}$ and selects a different order for each copy. Alternatively, any least-squares problem such as (CS) can be lifted to a non-negative least-squares problem, allowing us to satisfy Assumption 3 with the map supp, but potentially no longer satisfying Assumption 1 (see Appendix A).

Assumption 1, which requires f and g to be submodular can be restrictive in practice. To mitigate this, in Section 6 we show how some specific problem instances that do not satisfy Assumption 1–in particular when f is quadratic–can be lifted to a new optimization problem that satisfies all the required assumptions. We then derive conditions under which solving the new, lifted problem still provides a solution to the original problem that violated Assumption 1. In contrast, the more typical way of handling non-submodular f involves relaxing the definition of submodularity (2.1.6) to include an additive or multiplicative constant and propagating it through a chosen algorithm to give near-optimality guarantees. [EJ20, EKD18] Our suggested lifting, however, sidesteps the need for a particular algorithm while still providing optimality or near-optimality guarantees.

CHAPTER 3

Solving an Equivalent Problem

In this section, we outline our approach for solving the problem (P) by defining a related optimization problem on a single lattice. We then prove that this related problem is a submodular function minimization problem, and that by solving it we recover a solution to (P). Finally, we highlight some conditions under which solving this related problem is a polynomial time operation.

3.1 The Equivalent Submodular Minimization Problem

As expressed above, the problem (P) asks for the a choice of $\mathbf{x} \in \mathcal{X}$ and associated $\eta(\mathbf{x}) \in \mathcal{Y}$. Our key observation is that we could instead ask for a choice of $\mathbf{y} \in \mathcal{Y}$ and best associated $\mathbf{x} \in \mathcal{X}$, leading to the problem:

$$\underset{\mathbf{y} \in \mathcal{Y}}{\text{minimize}} \;\; g(\mathbf{y}) + \min_{\substack{\mathbf{x} \in \mathcal{X} \\ \eta(\mathbf{x}) = \mathbf{y}}} f(\mathbf{x}).$$

In the special case of (CS) explored earlier, this equivalent problem becomes:

$$\underset{S \in 2^{[n]}}{\text{minimize}} \;\; |S| + \min_{\substack{\mathbf{x} \in \mathbb{R}^n_{\geq 0} \\ \text{supp}(\mathbf{x}) = S}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2.$$

While this new problem is clearly the same as (CS), the innermost minimization is over the set of $\mathbf{x} \in \mathbb{R}^n_{\geq 0}$ such that $\text{supp}\,(\mathbf{x}) = S$, or equivalently, $\mathbf{x}_i \neq 0$ for all $i \in S$, and $\mathbf{x}_i = 0$ for all $i \notin S$. This feasible set is not a closed subset of $\mathbb{R}^n_{\geq 0}$, and thus the corresponding minimizer of this innermost problem may not exist [BL06].

With this issue in mind, we instead consider a slight relaxation of the above problem:

$$\underset{\mathbf{y}\in\mathcal{Y}}{\text{minimize}} \;\; g(\mathbf{y}) + H(\mathbf{y}), \tag{P-R}$$

where we have defined the function $H : \mathcal{Y} \to \mathbb{R}$ as:

$$H(\mathbf{y}) = \min_{\substack{\mathbf{x}\in\mathcal{X} \\ \eta(\mathbf{x}) \sqsubseteq \mathbf{y}}} f(\mathbf{x}). \tag{3.1.1}$$

In the special case of (CS), this relaxation produces the problem:

$$\underset{S\in 2^{[n]}}{\text{minimize}} \;\; |S| + \min_{\substack{\mathbf{x}\in\mathbb{R}^n_{\geq 0} \\ \text{supp}(\mathbf{x})\subseteq S}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2, \tag{CS-R}$$

where the innermost minimization is instead over the set of $\mathbf{x} \in \mathbb{R}^n_{\geq 0}$ such that $\mathbf{x}_i = 0$ for all $i \notin S$, which is a closed subset of $\mathbb{R}^n_{\geq 0}$.

We now prove that under Assumptions 1-3, the relaxed problem (P-R) is a submodular minimization problem, and that by solving it we can recover the corresponding minimizer for (P). As established above, minimizing functions on finitely presentable distributive lattices is efficient when the functions are submodular, so we show that the relaxed problem (P-R) is a submodular function minimization problem on $\mathcal{Y}$.

Theorem 2. *Under Assumptions 1-3, the function $g + H : \mathcal{Y} \to \mathbb{R}$ is submodular on $\mathcal{Y}$, and therefore the relaxed problem* (P-R) *is a submodular function minimization problem over $\mathcal{Y}$. Moreover, let $\mathbf{y}^* \in \mathcal{Y}$ be the minimizer for the problem* (P-R)*, and let $\mathbf{x}^* \in \mathcal{X}$ be such that:*

$$\mathbf{x}^* \in \underset{\substack{\mathbf{x}\in\mathcal{X} \\ \eta(\mathbf{x}) \sqsubseteq \mathbf{y}^*}}{\text{argmin}} \; f(\mathbf{x}).$$

Then $\mathbf{x}^*$ is a minimizer for the problem (P)*.*

To prove this result, we require a few technical lemmas.

Lemma 1. *Let $(\mathcal{X}, \preceq)$ and $(\mathcal{Y}, \sqsubseteq)$ be lattices with the map $\eta : \mathcal{X} \to \mathcal{Y}$ satisfying Assumption 3. Then the set:*

$$\mathcal{D} = \{(\mathbf{x}, \mathbf{y}) \in \mathcal{X} \times \mathcal{Y} \mid \eta(\mathbf{x}) \sqsubseteq \mathbf{y}\}, \tag{3.1.2}$$

is a sublattice of the product lattice, $\mathcal{X} \times \mathcal{Y}$.

Proof. On the product lattice, the join of any two elements $(\mathbf{x}, \mathbf{y}), (\mathbf{x}', \mathbf{y}') \in \mathcal{D}$ is denoted by $\vee_{\mathcal{D}}$, and defined as:

$$(\mathbf{x}, \mathbf{y}) \vee_{\mathcal{D}} (\mathbf{x}', \mathbf{y}') = (\mathbf{x} \curlyvee \mathbf{x}', \mathbf{y} \sqcup \mathbf{y}').$$

Then, we note that for this same $(\mathbf{x}, \mathbf{y}), (\mathbf{x}', \mathbf{y}') \in \mathcal{D}$:

$$\eta(\mathbf{x} \curlyvee \mathbf{x}') \sqsubseteq \eta(\mathbf{x}) \sqcup \eta(\mathbf{x}') \sqsubseteq \mathbf{y} \sqcup \mathbf{y}',$$

where we first used Assumption 3, then the fact that $(\mathbf{x}, \mathbf{y}), (\mathbf{x}', \mathbf{y}') \in \mathcal{D}$. Therefore, the pair $(\mathbf{x} \curlyvee \mathbf{x}', \mathbf{y} \sqcup \mathbf{y}')$ is also in D.

Because $(\mathbf{x}, \mathbf{y})$ and $(\mathbf{x}', \mathbf{y}')$ were arbitrary, this holds for all of $\mathcal{D}$. A dual analysis follows for the meet operation. □

The sublattice $\mathcal{D}$ is useful as the only pairs of $(\mathbf{x}, \mathbf{y}) \in \mathcal{X} \times \mathcal{Y}$ considered in the problem (P-R) are those that are in $\mathcal{D}$. The following theorem then uses this sublattice to prove that H is submodular. The result is a simple application of an established theorem in literature, but we include its proof here for completeness.

Theorem 3. *(Application of Theorem 2.7.6 in [Top98]) Let $f : \mathcal{X} \to \mathbb{R}$, $g : \mathcal{Y} \to \mathbb{R}$, and $\eta : \mathcal{X} \to \mathcal{Y}$ be maps satisfying Assumptions 1 and 3. Then the function $g + H : \mathcal{Y} \to \mathbb{R}$, with H defined as in* (3.1.1)*, is submodular on $\mathcal{Y}$.*

Proof. To prove this statement, we take two points $\mathbf{y}, \mathbf{y}' \in \mathcal{Y}$ and compare the values of the function $g + H$, verifying the submodular inequality (2.1.6). We note that for any $\mathbf{y}, \mathbf{y}' \in \mathcal{Y}$, there are corresponding $\mathbf{z}, \mathbf{z}' \in \mathcal{X}$ such that:

$$\begin{aligned} \mathbf{z} \in \underset{\substack{\mathbf{x} \in \mathcal{X} \\ \eta(\mathbf{x}) \sqsubseteq \mathbf{y}}}{\operatorname{argmin}} \; f(\mathbf{x}) \quad &\Rightarrow \quad H(\mathbf{y}) = f(\mathbf{z}), \\ \mathbf{z}' \in \underset{\substack{\mathbf{x} \in \mathcal{X} \\ \eta(\mathbf{x}) \sqsubseteq \mathbf{y}'}}{\operatorname{argmin}} \; f(\mathbf{x}) \quad &\Rightarrow \quad H(\mathbf{y}') = f(\mathbf{z}'). \end{aligned} \tag{3.1.3}$$

By definition, $(\mathbf{z}, \mathbf{y})$ and $(\mathbf{z}', \mathbf{y}')$ are both in the subset $\mathcal{D}$ as defined in (3.1.2). Then, it follows:

$$\begin{aligned} g(\mathbf{y}) + H(\mathbf{y}) + g(\mathbf{y}') + H(\mathbf{y}') &= g(\mathbf{y}) + f(\mathbf{z}) + g(\mathbf{y}') + f(\mathbf{z}') \\ &\geq g(\mathbf{y} \sqcup \mathbf{y}') + g(\mathbf{y} \sqcap \mathbf{y}') + f(\mathbf{z} \curlyvee \mathbf{z}') + f(\mathbf{z} \curlywedge \mathbf{z}'), \end{aligned}$$

where we first used (3.1.3) and then the submodularity of f and g.

By Lemma 1, $\mathcal{D}$ is a sublattice of $\mathcal{X} \times \mathcal{Y}$, and so the pairs $(\mathbf{z} \curlyvee \mathbf{z}', \mathbf{y} \sqcup \mathbf{y}')$ and $(\mathbf{z} \curlywedge \mathbf{z}', \mathbf{y} \sqcap \mathbf{y}')$ are also in $\mathcal{D}$, meaning:

$$\begin{aligned} \eta(\mathbf{z} \curlyvee \mathbf{z}') &\sqsubseteq \mathbf{y} \sqcup \mathbf{y}', \\ \eta(\mathbf{z} \curlywedge \mathbf{z}') &\sqsubseteq \mathbf{y} \sqcap \mathbf{y}'. \end{aligned}$$

Therefore $\mathbf{z} \curlyvee \mathbf{z}'$ and $\mathbf{x} \curlywedge \mathbf{x}'$ are feasible points in the minimization defining $H(\mathbf{y} \sqcup \mathbf{y}')$ and $H(\mathbf{y} \sqcap \mathbf{y}')$, respectively, in (3.1.1). We then have, as desired:

$$\begin{aligned} g(\mathbf{y}) + H(\mathbf{y}) + g(\mathbf{y}') + H(\mathbf{y}') &\geq g(\mathbf{y} \sqcup \mathbf{y}') + g(\mathbf{y} \sqcap \mathbf{y}') + f(\mathbf{z} \curlyvee \mathbf{z}') + f(\mathbf{z} \curlywedge \mathbf{z}') \\ &\geq g(\mathbf{y} \sqcup \mathbf{y}') + g(\mathbf{y} \sqcap \mathbf{y}') + \min_{\substack{\mathbf{x} \in \mathcal{X} \\ \eta(\mathbf{x}) \sqsubseteq \mathbf{y} \sqcup \mathbf{y}'}} f(\mathbf{x}) + \min_{\substack{\mathbf{x} \in \mathcal{X} \\ \eta(\mathbf{x}) \sqsubseteq \mathbf{y} \sqcap \mathbf{y}'}} f(\mathbf{x}) \\ &= g(\mathbf{y} \sqcup \mathbf{y}') + H(\mathbf{y} \sqcup \mathbf{y}') + g(\mathbf{y} \sqcap \mathbf{y}') + H(\mathbf{y} \sqcap \mathbf{y}'). \end{aligned}$$

□

Because $g + H$ is submodular on $\mathcal{Y}$, solving (P-R), is an instance of submodular function minimization. What remains is to show that solving this relaxed problem allows us to also solve to the original problem, (P).

Lemma 2. *Let $\mathbf{y}^* \in \mathcal{Y}$ be a minimizer for the relaxed problem* (P-R)*, and let $\mathbf{x}^* \in \mathcal{X}$ be such that:*

$$\mathbf{x}^* \in \operatorname*{argmin}_{\substack{\mathbf{x} \in \mathcal{X} \\ \eta(\mathbf{x}) \sqsubseteq \mathbf{y}^*}} f(\mathbf{x}).$$

If g satisfies Assumption 2, then $\mathbf{x}^*$ is a minimizer for the problem (P)*.*

Proof. To prove this lemma, we consider an optimal $\mathbf{z}^* \in \mathcal{X}$ for problem (P) and verify that the proposed minimizer, $\mathbf{x}^* \in \mathcal{X}$, has the same cost.

We first note that by the optimality of $\mathbf{z}^*$ in problem (P):

$$f(\mathbf{z}^*) + g(\eta(\mathbf{z}^*)) \leq f(\mathbf{x}^*) + g(\eta(\mathbf{x}^*)). \tag{3.1.4}$$

Additionally, we have:

$$\begin{aligned}
f(\mathbf{z}^*) + g(\eta(\mathbf{z}^*)) &\geq \min_{\substack{\mathbf{x}\in\mathcal{X} \\ \eta(\mathbf{x}) \sqsubseteq \eta(\mathbf{z}^*)}} f(\mathbf{x}) + g(\eta(\mathbf{z}^*)) && \text{(minimizing, as } \mathbf{z}^* \text{ is feasible)} \\
&= H(\eta(\mathbf{z}^*)) + g(\eta(\mathbf{z}^*)) && \text{(definition of } H\text{)} \\
&\geq H(\mathbf{y}^*) + g(\mathbf{y}^*) && \text{(optimality of } \mathbf{y}^* \text{ in P-R)} \\
&= f(\mathbf{x}^*) + g(\mathbf{y}^*) && \text{(definition of } \mathbf{x}^*\text{).}
\end{aligned}$$

This sequence of inequalities implies:

$$f(\mathbf{z}^*) + g(\eta(\mathbf{z}^*)) \geq f(\mathbf{x}^*) + g(\mathbf{y}^*). \tag{3.1.5}$$

By construction, there are exactly two possible relationships between $\mathbf{x}^*$ and $\mathbf{y}^*$.

Case 1: $\eta(\mathbf{x}^*) = \mathbf{y}^*$. In this case, (3.1.5) becomes:

$$f(\mathbf{z}^*) + g(\eta(\mathbf{z}^*)) \geq f(\mathbf{x}^*) + g(\mathbf{y}^*) = f(\mathbf{x}^*) + g(\eta(\mathbf{x}^*)).$$

Then, combining inequality (3.1) with (3.1.4), we have that:

$$\begin{aligned}
f(\mathbf{z}^*) + g(\eta(\mathbf{z}^*)) \geq f(\mathbf{x}^*) + g(\eta(\mathbf{x}^*)) \geq f(\mathbf{z}^*) + g(\eta(\mathbf{z}^*)) \\
\Rightarrow f(\mathbf{z}^*) + g(\eta(\mathbf{z}^*)) = f(\mathbf{x}^*) + g(\eta(\mathbf{x}^*)),
\end{aligned}$$

and therefore $\mathbf{x}^*$ is also a minimizer for problem (P).

Case 2: $\eta(\mathbf{x}^*) \sqsubset \mathbf{y}^*$. In this case, because g is monotone, $g(\mathbf{y}^*) \geq g(\eta(\mathbf{x}^*))$. Using this fact, we can lower bound the right-hand side of (3.1.5):

$$\begin{aligned}
f(\mathbf{z}^*) + g(\eta(\mathbf{z}^*)) &\geq f(\mathbf{x}^*) + g(\mathbf{y}^*) \\
&\geq f(\mathbf{x}^*) + g(\eta(\mathbf{x}^*)).
\end{aligned}$$

At this point, we have obtained (3.1), and we can follow the argument used in Case 1. □

This series of results gives rise to Theorem 2, which provides sufficient conditions under which we can transform problem (P), an optimization problem on two lattices, into problem (P-R), a submodular function minimization problem on a single lattice.

Proof. (Theorem 2)
Under Assumptions 1 and 3, Theorem 3 states that the function $g+H : \mathcal{Y} \to \mathbb{R}$ is submodular on the lattice $\mathcal{Y}$. Therefore, solving (P-R) is a submodular function minimization problem over $\mathcal{Y}$, and the first part of the theorem is proved.

Under Assumption 2, by Lemma 2, given the minimizer $\mathbf{y}^*$ of (P-R), the point $\mathbf{x}^* \in \mathcal{X}$ defined by:

$$\mathbf{x}^* \in \operatorname*{argmin}_{\substack{\mathbf{x}\in\mathcal{X} \\ \eta(\mathbf{x}) \sqsubseteq \mathbf{y}^*}} f(\mathbf{x}),$$

is a minimizer in the original problem (P). □

3.2 Solving (P-R) in Polynomial Time

Despite the submodular structure of the functions, we can only truly solve (P-R) in polynomial time if $\mathcal{Y}$ is a finitely presentable distributive lattice and evaluating H is a polynomial time operation. The function H, however, is implicitly defined through an optimization problem on a subset of $\mathcal{X}$ (3.1.1). Solving the problem (P-R) in polynomial time then requires solving these smaller optimization problems defining H efficiently.

We are particularly interested in joint continuous and discrete optimization, such as when $(\mathcal{X}, \preceq) = (\mathbb{R}^n_{\geq 0}, \sqsubseteq)$ and $(\mathcal{Y}, \preceq) = (2^{[n]}, \subseteq)$ connected by the map $\operatorname{supp} : \mathbb{R}^n_{\geq 0} \to 2^{[n]}$ as expressed in (2.2.1). In this case, evaluating H requires solving the optimization problem:

$$\operatorname*{minimize}_{\substack{\mathbf{x}\in\mathbb{R}^n_{\geq 0} \\ \operatorname{supp}(\mathbf{x})\subseteq A}} f(\mathbf{x}), \tag{3.2.1}$$

for any $A \in 2^{[n]}$. While (3.2.1) is a continuous submodular minimization problem, the set $\mathbb{R}_{\geq 0}$ is not a bounded sublattice. Moreover, algorithms for solving the submodular set

function minimization (P-R) require an accurate oracle model for $g+H$. As discussed above, continuous submodular minimization algorithms introduce discretization error, thus limiting the accuracy of the evaluations of H. Continuous submodularity alone appears limited in this way, hence, we may consider an alternative problem structure that allows for algorithms to produce efficient and arbitrarily accurate solutions of (3.2.1): convexity.

Note that in the sub-problem (3.2.1), for any $A \in 2^{[n]}$, the feasible set is a convex subset of $\mathbb{R}^n_{\geq 0}$. If the function $f : \mathbb{R}^n_{\geq 0} \to \mathbb{R}$ was convex, then we could use any generic convex optimization routine to solve (3.2.1). We already assumed that f is submodular on $\mathbb{R}^n_{\geq 0}$, but submodular functions are neither a subset nor a superset of convex functions, so we can also require that f is convex. For example, any separable convex function f satisfies this assumption, as do convex quadratic functions with non-positive off-diagonal entries, or functions on $\mathbb{R}^n$ that can be identified as the Lovász extension of submodular *set* functions. Under this assumption, evaluating H, and by extension solving (P), is efficient.

Corollary 1. *Let $f : \mathbb{R}^n_{\geq 0} \to \mathbb{R}$ be a submodular and convex function on $(\mathbb{R}^n_{\geq 0}, \preceq)$, $\mathcal{Y}$ be a finitely presentable distributive or diamond modular lattice, $g : \mathcal{Y} \to \mathbb{R}$ be a monotone submodular set function, and let $\eta : \mathbb{R}^n_{\geq 0} \to \mathcal{Y}$ satisfy Assumption 3. Further assume that for every $\mathbf{y} \in \mathcal{Y}$, the set of $\mathbf{x} \in \mathcal{X}$ such that $\eta(\mathbf{x}) \sqsubseteq \mathbf{y}$ is a convex subset of $\mathbb{R}^n_{\geq 0}$. Then the problem:*

$$\underset{\mathbf{x} \in \mathbb{R}^n_{\geq 0}}{\textit{minimize}} \quad f(\mathbf{x}) + g(\eta(\mathbf{x})),$$

can be solved in polynomial time.

Proof. Assumptions 1, 2, and 3 are satisfied by the lattices $(\mathbb{R}^n_{\geq 0}, \leq)$, $(\mathcal{Y}, \sqsubseteq)$, and the functions $\eta : \mathbb{R}^n_{\geq 0} \to \mathcal{Y}$, $f : \mathbb{R}^n_{\geq 0} \to \mathbb{R}$ and $g : \mathcal{Y} \to \mathbb{R}$. By Theorem 2, we can solve the problem (P) by instead minimizing the submodular function $g+H$ over $\mathcal{Y}$, i.e., solving problem (P-R).

Submodular function minimization over finitely presentable distributive and diamond modular lattices has polynomial complexity in the size of its representation–which is finite

by assumption–and the number of function evaluations of H. Because f is convex, and for any $\mathbf{y} \in \mathcal{Y}$ the minimization defining $H(\mathbf{y})$ is a convex set, evaluating $H(\mathbf{y})$ is a convex optimization problem. Convex optimization is a polynomial time operation, therefore evaluating H is also a polynomial time operation, and the total complexity of solving (P-R) using this oracle for H is polynomial in both n and the size of the representation of $\mathcal{Y}$. □

Our theory is agnostic to the choice of subroutines both for evaluating H and solving the set function minimization problem. If we assume f is convex, evaluate it through convex optimization, and use projected subgradient descent on the Lovàsz extension of $g + H$ as the algorithm for solving the set function minimization, we recover exactly the approach proposed by [EJ20].

Convexity of f is not the only assumption that leads to tractable evaluations of H. As an alternative, we could consider a nonconvex quadratic form for $f : \mathbb{R}^n_{\geq 0} \to \mathbb{R}$:

$$f(\mathbf{x}) = \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{p}^T \mathbf{x}, \tag{3.2.2}$$

with $\mathbf{Q} \in \mathbb{R}^{n \times n}$ and $\mathbf{p} \in \mathbb{R}^n$. The assumption that this quadratic function is submodular on $\mathbb{R}^n_{\geq 0}$ is equivalent to the condition:

$$\frac{\partial^2 f}{\partial \mathbf{x}_i \partial \mathbf{x}_j} = \mathbf{Q}_{ij} \leq 0, \quad \text{for all } i \neq j.$$

Moreover, for a given $A \in 2^{[n]}$, our sub-problem instance (3.2.1) is a constrained, nonconvex quadratic program:

$$\begin{aligned} \underset{\mathbf{x} \in \mathbb{R}^n}{\text{minimize}} \quad & \mathbf{x}^T \mathbf{Q} \mathbf{x} + 2\mathbf{p}^T \mathbf{x} \\ \text{subject to} \quad & \mathbf{x} \geq 0 \\ & \mathbf{x}_i = 0, \ i \notin A. \end{aligned} \tag{3.2.3}$$

Researchers [KK03] have established that nonconvex quadratic programs satisfying submodularity admit tight semidefinite program relaxations. In particular, we have the following theorem:

Theorem 4. *(Theorem 3.1 in [KK03]) Let* $\mathbf{Q} \in \mathbb{R}^{n \times n}$ *have nonpositive off-diagonal entries. Let* $\mathrm{tr} : \mathbb{R}^{n \times n} \to \mathbb{R}$ *denote the trace of a matrix,* $\mathrm{diag} : \mathbb{R}^{n \times n} \to \mathbb{R}^n$ *denote the diagonal entries of the matrix, and let* $\succeq$ *indicate the positive semidefiniteness of a symmetric matrix. Further, for any* $A \in 2^{[n]}$*, let* $\mathbf{Z}_{A^c}$ *denote the rows and columns of* $\mathbf{Z}$ *with indices not in the set* A*. Consider the semi-definite program:*

$$\begin{aligned} \underset{\substack{\mathbf{z} \in \mathbb{R}^n \\ \mathbf{Z} \in \mathbb{S}^n}}{\textit{minimize}} \quad & \mathrm{tr}\left(\mathbf{QZ}\right) + 2\mathbf{p}^T\mathbf{z} \\ \textit{subject to} \quad & \mathrm{tr}\left(\mathbf{Z}_{A^c}\right) \leq 0 \\ & \mathrm{diag}\left(\mathbf{Z}\right) \geq 0 \\ & \begin{bmatrix} 1 & \mathbf{z}^T \\ \mathbf{z} & \mathbf{Z} \end{bmatrix} \succeq 0, \end{aligned}$$

Given the solution $(\mathbf{Z}^*, \mathbf{z}^*)$ *to this SDP, the vector* $\mathbf{x}_i^* = \sqrt{\mathbf{Z}_{ii}^*}$*,* $i = 1, ..., n$ *is a minimizer for the non-convex quadratic program* (3.2.3)*.*

Because semi-definite programs can be solved in polynomial time, we could use this relaxation to evaluate H for any subset $A \in 2^{[n]}$ in polynomial time. As before, this ability would produce an identical statement to Corollary 1, but for functions f of the form (3.2.2) that satisfy submodularity.

CHAPTER 4

Constrained Optimization

In this and the following sections, we extend our framework both theoretically and algorithmically for the specific case of the lattices $(\mathbb{R}^n_{\geq 0}, \preceq)$ and $(2^{[n]}, \subseteq)$, connected by the support map $\text{supp} : \mathbb{R}^n_{\geq 0} \to 2^{[n]}$.

In many problems, we may be interested in optimization over a feasible strict subset $C \subset \mathbb{R}^n_{\geq 0}$. Unfortunately, submodular function minimization and maximization subject to constraints is NP-Hard in general [FI11]. This difficulty arises because arbitrary subsets of a lattice rarely define sublattices.

One simple class of problems whose feasible sets are not sublattices are problems with *budget constraints*:

$$\begin{aligned} \underset{\mathbf{x} \in \mathbb{R}^n_{\geq 0}}{\text{minimize}} \quad & f(\mathbf{x}) + g(\text{supp}\,(\mathbf{x})) \\ \text{subject to} \quad & \textstyle\sum_{i=1}^n W_i(\mathbf{x}_i) \leq B, \end{aligned} \tag{4.0.1}$$

with $W_i : \mathbb{R}_{\geq 0} \to \mathbb{R}$ strictly increasing functions for $i = 1, 2, ..., n$ and $B \in \mathbb{R}_{>0}$ a "budget".

When confronted with constrained optimization problems such as (4.0.1), one common approach is to add a Lagrange multiplier $\mu \in \mathbb{R}_{\geq 0}$ and instead solve the unconstrained problem:

$$\underset{\mathbf{x} \in \mathbb{R}^n_{\geq 0}}{\text{minimize}} \quad f(\mathbf{x}) + g(\text{supp}\,(\mathbf{x})) + \mu \sum_{i=1}^n W_i(\mathbf{x}_i). \tag{4.0.2}$$

For the correct choice of $\mu \in \mathbb{R}_{\geq 0}$, solving the regularized problem (4.0.2) can be equivalent to solving the constrained problem (4.0.1) [NKA11, SJ19]. Because (4.0.1) is non-convex,

identifying when this approach is valid requires some careful detail. When possible, however, determining the μ that renders the two problems equivalent is typically a difficult task.

Our work in this section relies on the following result that relates parameterized families of submodular set function minimization problems to a single convex optimization problem.

Theorem 5. *(Proposition 8.4 in [Bac13]) Let $h : 2^{[n]} \to \mathbb{R}$ be a submodular set function, and $h_L : \mathbb{R}^n \to \mathbb{R}$ its Lovàsz extension (which is therefore convex). If, for some $\epsilon > 0$, $\psi_i : \mathbb{R}_{\geq \epsilon} \to \mathbb{R}$ is a strictly increasing function on its domain for all $i = 1, 2, ..., n$, then the minimizer $\mathbf{u}^* \in \mathbb{R}^n_{\geq 0}$ of the convex optimization problem:*

$$\underset{\mathbf{u} \in \mathbb{R}^n_{\geq 0}}{\text{minimize}} \;\; h_L(\mathbf{u}) + \sum_{i=1}^{n} \int_{\epsilon}^{\epsilon + \mathbf{u}_i} \psi_i(\mu) d\mu, \tag{4.0.3}$$

is such that the set $A^\mu = \{i \in [n] \;:\; \mathbf{u}^*_i > \mu\}$ is the minimizer with smallest cardinality for the submodular set function minimization problem:

$$\underset{A \in 2^{[n]}}{\text{minimize}} \;\; h(A) + \sum_{i \in A} \psi_i(\mu), \tag{4.0.4}$$

for any $\mu \in \mathbb{R}_{\geq \epsilon}$.

In the following subsections we identify classes of problems that allow the regularized problem (4.0.2) to be expressed in the form given by (4.0.4). Theorem 5 then provides a single convex optimization problem we can solve to recover the solution to (4.0.2) for all possible values of the regularization strength μ. In prior work, this same theory was applied to purely discrete submodular minimization problems [FI11], and purely continuous submodular minimization problems [SJ19], but our work lies between these two extremes.

4.1 Support Knapsack Constraints

We first consider a knapsack constraint, meaning the function W has the form:

$$W(\mathbf{x}) = \sum_{j \in \text{supp}(\mathbf{x})} \mathbf{w}_j,$$

for some $\mathbf{w} \in \mathbb{R}^n_{>0}$. The regularized problem (4.0.2) in this case is:

$$\underset{\mathbf{x}\in\mathbb{R}^n_{\geq 0}}{\text{minimize}} \quad f(\mathbf{x}) + g(\text{supp}\,(\mathbf{x})) + \mu \sum_{j\in\text{supp}(\mathbf{x})} \mathbf{w}_j.$$

Because W is a set function in this case, the relaxed problem (P-R) becomes:

$$\underset{A\in 2^{[n]}}{\text{minimize}} \quad g(A) + H(A) + \sum_{j\in A} \psi_j(\mu), \tag{4.1.1}$$

where we have defined $\psi_j(\mu) = \mu\mathbf{w}_j$ for each $j = 1, 2, ..., n$. Because $\mathbf{w}_j > 0$ for all j, these functions are strictly increasing, and we have a problem in the form (4.0.4). By Theorem 5, we can solve the convex optimization problem:

$$\underset{\mathbf{u}\in\mathbb{R}_{\geq\epsilon}}{\text{minimize}} \quad g_L(\mathbf{u}) + H_L(\mathbf{u}) + \frac{1}{2}\sum_{j=1}^{n} \mathbf{w}_j\mathbf{u}_j^2,$$

then appropriately threshold the solution to recover the solution to (4.1.1) for all possible values of $\mu \in \mathbb{R}_{\geq\epsilon}$. Because ψ_j is finite and strictly increasing on all of $\mathbb{R}$, we can simply select $\epsilon = 0$.

Given the solutions to the regularized problem A^μ specified by Theorem 5, we select the set A^μ with smallest $\mu \in \mathbb{R}$ such that the constraint $W(\mathbf{x}) \leq B$ is satisfied. Note however, that we only recover the solution for *any* given $B \in \mathbb{R}_{\geq 0}$ if the elements of $\mathbf{u}^*$ are unique [Bac13]. Otherwise, we only recover the solutions for a few particular values of B. If these elements are unique, however, we can use the result of Theorem 2 to compute the minimizer in the original optimization problem over $\mathbb{R}^n_{\geq 0}$. Moreover, by the same argument as in [NKA11], this solution corresponds to the solution of the original constrained problem.

4.2 Continuous Budget Constraints

As shown above, the Lovàsz extension lets us handle problems with discrete budget constraints, so a natural next step is to consider continuous budget constraints, meaning con-

tinuous functions $W : \mathbb{R}^n_{\geq 0} \to \mathbb{R}$, such that:

$$W(\mathbf{x}) = \sum_{i=1}^{n} W_i(\mathbf{x}_i),$$

with each $W_i : \mathbb{R}_{\geq 0} \to \mathbb{R}$ a strictly increasing function. With this particular W, the regularized optimization problem (4.0.2) with Lagrange multiplier $\mu \in \mathbb{R}_{\geq 0}$ becomes:

$$\underset{\mathbf{x} \in \mathbb{R}^n_{\geq 0}}{\text{minimize}} \quad f(\mathbf{x}) + g(\text{supp}\,(\mathbf{x})) + \mu \sum_{i=1}^{n} W_i(\mathbf{x}_i).$$

To recover the problem form (4.0.4) specified by Theorem 5, we further assume that $f : \mathbb{R}^n_{\geq 0} \to \mathbb{R}$ is separable, i.e., $f(\mathbf{x}) = \sum_{i=1}^{n} f_i(\mathbf{x}_i)$. In this case, the relaxed optimization problem (P-R) is:

$$\underset{A \in 2^{[n]}}{\text{minimize}} \quad g(A) + \sum_{i \in A} H_i(\mu), \tag{4.2.1}$$

where we defined $H_i : \mathbb{R}_{>0} \to \mathbb{R}$ as the function:

$$H_i(\mu) = \min_{\mathbf{z} \geq 0} \; f_i(\mathbf{z}) + \mu W_i(\mathbf{z}), \quad i = 1, 2, ..., n, \tag{4.2.2}$$

and assumed (without loss of generality) that $W_i(0) = f_i(0) = 0$.

To apply Theorem 5, we need $H_i : \mathbb{R}_{>0} \to \mathbb{R}$ to be strictly increasing on its domain. We verify this property in the following proposition, whose proof we detail in Appendix B.

Proposition 1. The function $H_i : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\leq 0}$ defined in (4.2.2) is monotone in μ for all $i = 1, 2..., n$. It is strictly increasing for all $\mu \in [0, c]$, where $c \in \mathbb{R}_{\geq 0}$ is the smallest constant such that $H_i(c) = 0$. In addition, H_j is constant and zero on the interval $[c, \infty[$.

Because the only point at which H_i is not strictly increasing occurs when its value is exactly zero (implying that allowing the element $\mathbf{x}_i$ to be nonzero provides no decrease in continuous cost), the desired result from Theorem 5 still holds with only a minor modification, the details of which we also defer to Appendix B.

It then follows from Theorem 5 that by solving the single convex optimization problem:

$$\underset{\mathbf{u} \in \mathbb{R}^n_{\geq 0}}{\text{minimize}} \quad g_L(\mathbf{u}) + \sum_{i=1}^{n} \int_{\epsilon}^{\epsilon + \mathbf{u}_i} H_i(\mu) d\mu, \tag{4.2.3}$$

we can recover the solution to a family of regularized optimization problems (4.2.1). As before, we select the set A^{μ} with the largest $\mu \in \mathbb{R}_{\geq \epsilon}$ such that the budget constraint $W(\mathbf{x}) \leq B$ is satisfied. As discussed above, we only recover the solution for *all* $B \in \mathbb{R}_{\geq 0}$ if the elements of $\mathbf{u}^*$ are all unique. Within each choice of support, simple convex duality–which we can apply when f_i and W_i are convex functions–guarantees the existence of a $\mu \in \mathbb{R}_{\geq 0}$ that renders the constrained problem and the regularized problem equivalent.

CHAPTER 5

Robust Optimization

Joint continuous and discrete optimization problems can easily arise as sub-problems in larger contexts. For example, in *robust optimization*, we seek to solve an optimization problem while remaining resilient to worst-case problem instances.

5.1 Motivating Example from Multiple Domain Learning

Recent work by [QZT19] highlighted the concept of *multiple domain learning*, where a single machine learning model is trained on sets of data from K different domains. By training against worst-case distributions of the data in these domains, they show that the resulting machine learning model often achieves lower generalization and worst-case testing errors.

In particular, let the training data for a learning model be $S = \{S_1, S_2, ..., S_K\}$ with S_i the data from domain i. We also let $f_i : W \to \mathbb{R}$ for $i = 1, 2, ..., K$ be the empirical risk of the model on the data from each domain i, given parameters in some convex subset $W \subseteq \mathbb{R}^n$. The proposed robust optimization problem is then:

$$\underset{\mathbf{w}\in W}{\text{minimize}} \ \max_{\mathbf{p}\in C} \ \sum_{i=1}^{K} \mathbf{p}_i f_i(\mathbf{w}),$$

with $C = \{\mathbf{p} \in \mathbb{R}^K_{\geq 0} \mid \sum_{i=1}^{K} \mathbf{p}_i \leq 1\}$, the simplex. If we additionally reward the use of data from domain i (or equivalently, penalize the worst-case distribution of data for including domain i), then we form the robust continuous and discrete optimization problem:

$$\underset{\mathbf{w}\in W}{\text{minimize}} \ \max_{\mathbf{p}\in C} \ \sum_{i=1}^{K} \mathbf{p}_i f_i(\mathbf{w}) - g(\text{supp}\,(\mathbf{p})),$$

with $g : 2^{[K]} \to \mathbb{R}$ a monotone submodular set function. By considering a penalty on the set of nonzero entries of the worst-case distribution, we encode some prioritization of which domains are more or less relevant to us in our application. Then by Theorem 5, we can solve the inner maximization problem (with an appropriate change of signs) by adding a Lagrange multiplier μ and solving a related convex problem.

5.2 General Results

More generally, robust optimization problems can often be expressed as a min-max saddle point optimization problem of a function $q : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$:

$$\underset{\mathbf{x}\in\mathcal{X}}{\text{maximize}} \min_{\mathbf{y}\in\mathcal{Y}} q(\mathbf{x}, \mathbf{y}). \tag{5.2.1}$$

This problem is interpreted as maximizing the function $q(\mathbf{x}, \mathbf{y})$ with respect to our available parameters $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^n$, under the worst case choice of additional problem parameters $\mathbf{y} \in \mathcal{Y} \subseteq \mathbb{R}^m$ [BEN09].

Given some appropriate structure for the function q, the min-max problem (5.2.1) is surprisingly tractable. If we define $Q : \mathcal{X} \to \mathbb{R}$ as:

$$Q(\mathbf{x}) = \min_{\mathbf{y}\in\mathcal{Y}} q(\mathbf{x}, \mathbf{y}),$$

we can express the saddle-point problem (5.2.1) as:

$$\underset{\mathbf{x}\in\mathcal{X}}{\text{maximize}} \;\; Q(\mathbf{x}). \tag{5.2.2}$$

If the function $q(\mathbf{x}, \mathbf{y})$ is concave in $\mathbf{x}$ for any fixed $\mathbf{y} \in \mathcal{Y}$, then the function Q is also concave in $\mathbf{x}$ [BL06]. Moreover, we can compute a subgradient of Q at any $\mathbf{x}_0 \in \mathcal{X}$ as:

$$\nabla_{\mathbf{x}} Q(\mathbf{x}_0) = \nabla_{\mathbf{x}} q(\mathbf{x}_0, \mathbf{y}^*),$$

$$\mathbf{y}^* \in \underset{\mathbf{y}\in\mathcal{Y}}{\text{argmin}} \;\; q(\mathbf{x}_0, \mathbf{y}).$$

In other words, efficiently solving the minimization problem defining Q for an $\mathbf{x}_0 \in \mathcal{X}$ also gives a subgradient of Q. Because Q is concave in $\mathbf{x}$, even a straightforward algorithm such as projected subgradient ascent in the problem (5.2.2) will converge to a global optimum.

In this work, we showed that minimization problems in the form of (2.2.2) with functions satisfying Assumptions 1-3 can be solved efficiently. Suppose then, that the function $q : \mathcal{X} \times \mathcal{Y}$ is of the form:

$$q(\mathbf{x}, \mathbf{y}) = f(\mathbf{x}, \mathbf{y}) + g(\eta(\mathbf{y}))$$

with $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ concave in $\mathbf{x}$ for any fixed $\mathbf{y}$ and also convex and submodular on $\mathcal{Y} \subseteq \mathbb{R}^n_{\geq 0}$ in $\mathbf{y}$ for any fixed $\mathbf{x}$. If $\eta : \mathcal{Y} \to \mathcal{L}$ satisfies Assumption 3, $g : \mathcal{L} \to \mathbb{R}$ is monotone and submodular, and we assume the set of $\mathbf{y} \in \mathcal{Y}$ such that $\eta(\mathbf{y}) \sqsubseteq \ell$ is a convex subset for any $\ell \in \mathcal{L}$, then the robust optimization problem (5.2.1) becomes:

$$\underset{\mathbf{x} \in \mathbb{R}^n}{\text{maximize}} \ \min_{\mathbf{y} \in \mathcal{Y}} f(\mathbf{x}, \mathbf{y}) + g(\eta(\mathbf{y})). \tag{5.2.3}$$

For a given $\mathbf{x}_0 \in \mathbb{R}^n$, we view the selection of $\mathbf{y} \in \mathcal{Y}$ as a worst-case, or "adversarial" choice of parameters for the function f. The penalty on $\eta(\mathbf{y})$ suggests that the adversarial parameters are selected while considering some preferred structure, such as sparsity. Submodularity here, implies that this adversary pays diminishing prices as it increases the number of parameters it uses.

In addition, Q becomes:

$$Q(\mathbf{x}) = \min_{\mathbf{y} \in \mathcal{Y}} f(\mathbf{x}, \mathbf{y}) + g(\eta(\mathbf{y})),$$

which is still the minimum of a family of concave functions, and therefore amenable to subgradient ascent methods as discussed above. A subgradient of Q can easily be computed as:

$$\nabla_{\mathbf{x}} Q(\mathbf{x}_0) = \nabla_{\mathbf{x}} q(\mathbf{x}_0, \mathbf{y}^*) = \nabla_{\mathbf{x}} f(\mathbf{x}_0, \mathbf{y}^*),$$

$$\mathbf{y}^* \in \underset{\mathbf{y} \in \mathcal{Y}}{\text{argmin}} \ f(\mathbf{x}_0, \mathbf{y}) + g(\eta(\mathbf{y})).$$

We collect these ideas into the following theorem.

Theorem 6. *Consider the robust optimization problem* (5.2.3). *Assume* $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ *is concave in* $\mathbf{x} \in \mathcal{X}$ *for any fixed* $\mathbf{y} \in \mathcal{Y}$, *and also convex and submodular in* $\mathbf{y} \in \mathcal{Y}$ *for any fixed* $\mathbf{x} \in \mathcal{X}$. *Let* $\eta : \mathcal{Y} \to \mathcal{L}$ *satisfy Assumption 3,* $g : \mathcal{L} \to \mathbb{R}$ *be a monotone submodular function and assume that for a given* $\ell \in \mathcal{L}$, *the set of* $\mathbf{y} \in \mathcal{Y}$ *such that* $\eta(\mathbf{y}) \sqsubseteq \ell$ *is a convex subset of* $\mathcal{Y}$. *Moreover, let* $\mathcal{Y}$ *be a finitely presentable distributive lattice. For any* $\epsilon \in \mathbb{R}_{>0}$, *let* $T \in \mathbb{Z}_{>0}$ *be of order* $O(\frac{1}{\epsilon^2})$, *meaning as* T *tends to infinity, there exists a constant* $M \in \mathbb{R}_{>0}$ *such that* $T \leq \frac{M}{\epsilon^2}$. *Then* T *iterations of projected subgradient ascent using step lengths* $\eta_i = \frac{1}{\sqrt{T}}$ *produces, in polynomial time, iterates* $\mathbf{x}^{(i)} \in \mathcal{X}$ *for* $i = 1, 2, ..., T$ *such that* $\frac{1}{T} \sum_{i=1}^{T} Q(\mathbf{x}^{(i)}) \leq Q(\mathbf{x}^*) + \epsilon$.

The computational complexity of this approach may be high, as projected subgradient ascent can be slow in practice. However, each sub-problem instance involves a mixed continuous and discrete optimization problem, so this complexity is warranted.

CHAPTER 6

Relaxing Submodularity

For the results of Theorem 2 and therefore Corollary 1 and its extensions to apply, Assumptions 1-3 must be met. There are, however, situations where these assumptions may not hold. For example, consider again a quadratic form for $f : \mathbb{R}^n_{\geq 0} \to \mathbb{R}$:

$$f(\mathbf{x}) = \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{p}^T \mathbf{x}, \tag{6.0.1}$$

and a monotone and submodular set function $g : 2^{[n]} \to \mathbb{R}$. Then the general lattice optimization problem (P) becomes:

$$\underset{\mathbf{x} \in \mathbb{R}^n_{\geq 0}}{\text{minimize}} \quad \ell(\mathbf{x}) := \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{p}^T \mathbf{x} + g(\text{supp}\,(\mathbf{x})). \tag{6.0.2}$$

The assumption that f is submodular on $(\mathbb{R}^n_{\geq 0}, \preceq)$ is equivalent to:

$$\frac{\partial^2 f}{\partial \mathbf{x}_i \partial \mathbf{x}_j} = \mathbf{Q}_{ij} \leq 0, \quad \text{for all } i \neq j.$$

Moreover, for Corollary 1 to apply, we also need the matrix $\mathbf{Q}$ to be positive semidefinite. These two assumptions are unlikely to both be met by quadratic forms resulting from real data.

Typically, violations of submodularity are handled by suitably relaxing the definition of submodularity with an additive or multiplicative constant [EKD18, DK18]. This constant is then propagated through the particular algorithm choice, providing a similarly relaxed optimality guarantee [EJ20].

Alternatively, our work focuses on finding exact solutions to these joint problems in an algorithm-agnostic and efficient way. In this spirit, we show in this section how quadratic

problems such as (6.0.2) can be embedded in another optimization problem satisfying Assumptions 1-3. We then prove conditions under which the solutions to this *lifted* optimization problem–which can be efficiently found, since Assumptions 1-3 are now satisfied–correspond to an exact solution of the original quadratic problem (6.0.2).

6.1 Lifting Non-submodular Quadratics

Given the quadratic form for f as in (6.0.1), we can decompose the matrix $\mathbf{Q} \in \mathbb{R}^{n \times n}$ into its submodular and non-submodular parts additively:

$$\mathbf{Q} = \mathbf{Q}^- + \mathbf{Q}^+, \tag{6.1.1}$$

$$\mathbf{Q}^-_{ij} = \begin{cases} \mathbf{Q}_{ij}, & i = j \text{ or } \mathbf{Q}_{ij} \leq 0, \\ 0, & \text{otherwise}, \end{cases} \qquad \mathbf{Q}^+_{ij} = \begin{cases} \mathbf{Q}_{ij}, & i \neq j \text{ and } \mathbf{Q}_{ij} > 0 \\ 0, & \text{otherwise}. \end{cases} \tag{6.1.2}$$

Then, we define a new, lifted quadratic function $\tilde{f} : \mathbb{R}^n_{\geq 0} \times \mathbb{R}^n_{\geq 0} \to \mathbb{R}$ as:

$$\tilde{f}(\mathbf{z}, \mathbf{w}) = \frac{1}{2} \begin{bmatrix} \mathbf{z} \\ \mathbf{w} \end{bmatrix}^T \begin{bmatrix} \mathbf{Q}^- & \mathbf{Q}^+ \\ \mathbf{Q}^+ & \mathbf{Q}^- \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \mathbf{w} \end{bmatrix} + \frac{1}{2} \begin{bmatrix} \mathbf{q} \\ \mathbf{q} \end{bmatrix}^T \begin{bmatrix} \mathbf{z} \\ \mathbf{w} \end{bmatrix}. \tag{6.1.3}$$

The lifted function $\tilde{f}$ also has some nice properties that we can use to our advantage.

Lemma 3. *The function $\tilde{f} : \mathbb{R}^n_{\geq 0} \times \mathbb{R}^n_{\geq 0} \to \mathbb{R}$ defined in* (6.1.3) *is such that for all $(\mathbf{z}, \mathbf{w}) \in \mathbb{R}^n_{\geq 0} \times \mathbb{R}^n_{\geq 0}$:*

$$\tilde{f}(\mathbf{z}, \mathbf{w}) = \tilde{f}(\mathbf{w}, \mathbf{z}), \tag{6.1.4}$$

and for all $\mathbf{x} \in \mathbb{R}^n_{\geq 0}$:

$$\tilde{f}(\mathbf{x}, \mathbf{x}) = f(\mathbf{x}). \tag{6.1.5}$$

We can similarly lift the function $g : 2^{[n]} \to \mathbb{R}$ to the function $\tilde{g} : 2^{[n]} \times 2^{[n]} \to \mathbb{R}$, defined simply as:

$$\tilde{g}(S, T) = \frac{1}{2} \left(g(S) + g(T) \right). \tag{6.1.6}$$

The lifted function $\tilde{g}$ satisfies the same symmetry and embedding properties as the lifted function $\tilde{f}$.

Lemma 4. *The function $\tilde{g}$ defined in* (6.1.6) *is such that for all $(S,T) \in 2^{[n]} \times 2^{[n]}$:*

$$\tilde{g}(S,T) = \tilde{g}(T,S), \tag{6.1.7}$$

and for all $A \in 2^{[n]}$:

$$\tilde{g}(A,A) = g(A). \tag{6.1.8}$$

With the lifted functions $\tilde{f}$ and $\tilde{g}$ in hand, we define a lifted version of the original quadratic optimization problem (6.0.2):

$$\underset{(\mathbf{z},\mathbf{w})\in\mathbb{R}^n_{\geq 0}\times\mathbb{R}^n_{\geq 0}}{\text{minimize}} \quad \tilde{\ell}(\mathbf{z},\mathbf{w}) := \tilde{f}(\mathbf{z},\mathbf{w}) + \tilde{g}\left(\text{supp}\left(\mathbf{z}\right), \text{supp}\left(\mathbf{w}\right)\right). \tag{6.1.9}$$

If we were to solve this lifted problem and find a solution on the diagonal, i.e., a solution $(\mathbf{z}^*, \mathbf{w}^*)$ such that $\mathbf{z}^* = \mathbf{w}^*$, we immediately recover the solution to the original quadratic problem (6.0.2).

Lemma 5. *If the solution to the lifted problem* (6.1.9)*, denoted $(\mathbf{z}^*, \mathbf{w}^*) \in \mathbb{R}^n_{\geq 0} \times \mathbb{R}^n_{\geq 0}$ is such that $\mathbf{z}^* = \mathbf{w}^*$, then the point $\mathbf{x}^* = \mathbf{z}^* = \mathbf{w}^*$ is an optimal solution to the original quadratic problem* (6.0.2).

Proof. By Lemmas 3 and 4, we know that:

$$\tilde{\ell}(\mathbf{z}^*, \mathbf{w}^*) = \ell(\mathbf{z}^*) = \ell(\mathbf{w}^*).$$

Further, by the optimality of $(\mathbf{z}^*, \mathbf{w}^*)$ and by shrinking the feasible set, we have:

$$\tilde{\ell}(\mathbf{z}^*, \mathbf{w}^*) = \ell(\mathbf{z}^*) \leq \min_{\mathbf{z},\mathbf{w}\in\mathbb{R}^n_{\geq 0}} \tilde{\ell}(\mathbf{z},\mathbf{w}) \leq \min_{\substack{\mathbf{z},\mathbf{w}\in\mathbb{R}^n_{\geq 0} \\ \mathbf{z}=\mathbf{w}}} \tilde{\ell}(\mathbf{z},\mathbf{w}) = \min_{\mathbf{x}\in\mathbb{R}^n_{\geq 0}} \ell(\mathbf{x}).$$

Therefore, the points $\mathbf{z}^*$ and $\mathbf{w}^*$ are also minimizers of the original problem (6.0.2). □

By Lemma 5, the solution to our initial quadratic problem is embedded in the new lifted problem (6.1.9). To use this result, however, we need two key ingredients: the ability to solve the lifted problem exactly and efficiently, and a way to easily produce solutions on the diagonal.

6.2 Efficiently solving the lifted problem

The lifted quadratic problem (6.1.9) has a nearly identical form to the original problem (6.0.2), but now satisfies Assumptions 1-3, as we prove next. As a result, we can use the approach outlined in Section 3.2 to solve the lifted problem.

To discuss Assumption 1 and submodularity, we define a partial order and lattice on the lifted space $\mathbb{R}^{\mathrm{n}}_{\geq 0} \times \mathbb{R}^{\mathrm{n}}_{\geq 0}$ so that we can discuss submodularity. In particular, we consider the partial order $\ll$, defined as:

$$(\mathbf{z}, \mathbf{w}) \ll (\mathbf{z}', \mathbf{w}') \qquad \Leftrightarrow \qquad \mathbf{z} \preceq \mathbf{z}' \text{ and } \mathbf{w} \succeq \mathbf{w}', \tag{6.2.1}$$

where $\preceq$ denotes the partial order on $\mathbb{R}^n$ previously defined in (2.1.3). In words, we order the first part of each pair of vectors in the typical fashion, but reverse the order for the second part. This choice of partial order also defines the join and meet operations:

$$(\mathbf{z}, \mathbf{w}) \veebar (\mathbf{z}', \mathbf{w}') = (\mathbf{z} \curlyvee \mathbf{z}', \mathbf{w} \curlywedge \mathbf{w}') \tag{6.2.2}$$

$$(\mathbf{z}, \mathbf{w}) \barwedge (\mathbf{z}', \mathbf{w}') = (\mathbf{z} \curlywedge \mathbf{z}', \mathbf{w} \curlyvee \mathbf{w}'), \tag{6.2.3}$$

where $\curlyvee$ and $\curlywedge$ are the join and meet operations on $(\mathbb{R}^n, \preceq)$ defined in (2.1.4) and (2.1.5).

By construction, then, the lifted quadratic function $\tilde{f}$ is submodular on this lattice. Moreover, since it is a quadratic form, simple conditions guarantee its convexity. We pursue convexity here to leverage faster exact algorithms for solving the problem, rather than the more general approach for continuous submodular minimization. Applying the continuous submodular minimization algorithm to this lifted problem while using arbitrarily fine discretization may be of future independent interest.

Lemma 6. *The function $\tilde{f} : \mathbb{R}^n \to \mathbb{R}^n$ defined in* (6.1.3) *is submodular on the lattice* $(\mathbb{R}^n \times \mathbb{R}^n, \ll)$. *Further, $\tilde{f}$ is convex if and only if both $\mathbf{Q}$ and $\mathbf{Q}^+ - \mathbf{Q}^-$ are positive semidefinite.*

Proof. We first note that the lattice $(\mathbb{R}^n \times \mathbb{R}^n, \ll)$ is an *orthant conic lattice*, as defined by [BLK17]. Therefore, by Proposition 2 of [BLK17], $\tilde{f}$ is submodular on this lattice if and only if:

$$\frac{\partial^2 \tilde{f}}{\partial \mathbf{x}_i \partial \mathbf{x}_j} \leq 0, \tag{6.2.4}$$

for all $i, j = 1, 2, ..., n$ or $i, j = n+1, n+2, ..., 2n$ with $i \neq j$ and:

$$\frac{\partial^2 \tilde{f}}{\partial \mathbf{x}_i \partial \mathbf{x}_j} \geq 0, \tag{6.2.5}$$

for all $i = 1, 2, ..., n$ and $j = n+1, n+2, ..., 2n$. For our lifted function $\tilde{f}$, its Hessian matrix is exactly:

$$\frac{\partial^2 \tilde{f}}{\partial \mathbf{x}^2} = \begin{bmatrix} \mathbf{Q}^- & \mathbf{Q}^+ \\ \mathbf{Q}^+ & \mathbf{Q}^- \end{bmatrix}.$$

By their construction, the matrices $\mathbf{Q}^+$ and $\mathbf{Q}^-$ satisfy both (6.2.4) and (6.2.5), and $\tilde{f}$ is submodular on $(\mathbb{R}^n \times \mathbb{R}^n, \ll)$.

For convexity, we note that the Hessian matrix must be positive semidefinite. By the matrix similarity:

$$\frac{1}{2} \begin{bmatrix} \mathbf{I} & -\mathbf{I} \\ \mathbf{I} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{Q}^+ & \mathbf{Q}^- \\ \mathbf{Q}^- & \mathbf{Q}+ \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{I} \\ -\mathbf{I} & \mathbf{I} \end{bmatrix} = \begin{bmatrix} \mathbf{Q}^+ - \mathbf{Q}^- & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}^+ + \mathbf{Q}^- \end{bmatrix},$$

this holds only when $\mathbf{Q} = \mathbf{Q}^+ + \mathbf{Q}^-$ and $\mathbf{Q}^+ - \mathbf{Q}^-$ are positive semidefinite. □

Similarly, we define a lattice in the lifted discrete space $2^{[n]} \times 2^{[n]}$ using the partial order $\Subset$ defined as:

$$(S, T) \Subset (S', T') \qquad \Leftrightarrow S \subseteq S' \text{ and } T \supseteq T'.$$

The join and meet operations on $(2^{[n]} \times 2^{[n]}, \Subset)$, denoted by $\Cup$ and $\Cap$ respectively, are:

$$(S,T) \Cup (S',T') = (S \cup S', T \cap T')$$
$$(S,T) \Cap (S',T') = (S \cap S', T \cup T').$$

We can then easily establish that the lifted function $\tilde{g}$ is submodular on the lifted discrete lattice.

Lemma 7. *If the function $g : 2^{[n]} \to \mathbb{R}$ is monotone and submodular, then the lifted function $\tilde{g}$ defined in* (6.1.6) *is submodular on the lattice $(2^{[n]} \times 2^{[n]}, \Subset)$. Moreover, it is monotone and submodular on the product lattice, $(2^{[n]} \times 2^{[n]}, \subseteq)$.*

Proof. Take a set $(S,T) \in 2^{[n]} \times 2^{[n]}$ and another set $(S',T') \in 2^{[n]} \times 2^{[n]}$. Then by definition, we have:

$$\begin{aligned}
\tilde{g}(S,T) + \tilde{g}(S',T') &= \frac{1}{2}\left(g(S) + g(T) + g(S') + g(T')\right) \\
&\geq \frac{1}{2}\left(g(S \cap S') + g(S \cup S') + g(T \cap T') + g(T \cup T')\right) \\
&= \tilde{g}\left((S,T) \Cup (S',T')\right) + \tilde{g}\left((S,T) \Cap (S',T')\right),
\end{aligned}$$

where the inequality follows from the submodularity of g, with $\Cup$ and $\Cap$ the join and meet operations associated with the partial order $\Subset$ on $2^{[n]} \times 2^{[n]}$. By grouping terms differently, we also see that $\tilde{g}$ is also monotone and submodular on the more typical product lattice $(2^{[n]} \times 2^{[n]}, \subseteq)$. □

Because $\tilde{g}$ is monotone on the product lattice and $\tilde{h}$ is submodular on $(\mathbb{R}^n_{\geq 0} \times \mathbb{R}^n_{\geq 0}, \ll)$, Lemma 2 applies, and we can define the parameterized function $\tilde{h} : 2^{[n]} \times 2^{[n]} \to \mathbb{R}$:

$$\tilde{h}(S,T) = \min_{\substack{\mathbf{z},\mathbf{w} \in \mathbb{R}^n_{\geq 0} \\ \operatorname{supp}(\mathbf{z}) \subseteq S \\ \operatorname{supp}(\mathbf{w}) \subseteq T}} \tilde{f}(\mathbf{z}, \mathbf{w}), \tag{6.2.6}$$

and then the solution to:

$$\underset{S,T \in 2^{[n]} \times 2^{[n]}}{\text{minimize}} \quad \tilde{g}(S,T) + \tilde{h}(S,T) \tag{6.2.7}$$

corresponds to a solution of the lifted problem (6.1.9).

Finally, note that Assumptions 1 and 3 are satisfied by $\tilde{f}$, $\tilde{g}$, the lattices $(2^{[n]} \times 2^{[n]}, \Subset)$ and $(\mathbb{R}^n_{\geq 0} \times \mathbb{R}^n_{\geq 0}, \ll)$, and the mapping $\text{supp} : \mathbb{R}^n_{\geq 0} \times \mathbb{R}^n_{\geq 0} \to 2^{[n]} \times 2^{[n]}$. Therefore, we have the following direct corollary of Theorem 2.

Corollary 2. *The function $\tilde{h} : 2^{[n]} \times 2^{[n]}$ is submodular on the lattice $(2^{[n]} \times 2^{[n]}, \Subset)$.*

Finally, if the non-submodular contribution to the quadratic form is not too large, particularly if $\mathbf{Q}^+ - \mathbf{Q}^-$ is positive semidefinite, then by Lemma 6 $\tilde{f}$ is also convex. Under this assumption, Corollary 1 applies, so we can solve the lifted optimization problem exactly in polynomial time.

Corollary 3. *Under the same assumptions as Corollary 1, if $\mathbf{Q}$ and $\mathbf{Q}^+ - \mathbf{Q}^-$ are both positive semidefinite matrices and $g : 2^{[n]} \to \mathbb{R}$ is monotone and submodular, then the lifted quadratic optimization problem* (6.1.9) *can be solved exactly in polynomial time.*

6.3 Guarantees

Corollary 3 in the previous subsection showed that a quadratic problem that does not satisfy Assumptions 1-3 can be lifted to another quadratic problem that does. Moreover, under mild assumptions on the problem data, the lifted problem can be solved exactly in polynomial time. The question then arises: is this lifted problem's solution useful?

Lemma 5 stated that if we are lucky enough to compute a minimizer to the lifted problem on the diagonal, then it is also necessarily a minimizer of the original quadratic problem. If we are unlucky, however, we would like to still to construct a minimizer of the original problem using the solution we found. The following result shows that this is indeed possible.

Lemma 8. *Let $(\mathbf{z}^*, \mathbf{w}^*) \in \mathbb{R}^n_{\geq 0} \times \mathbb{R}^n_{\geq 0}$ be a solution to the lifted quadratic optimization problem* (6.1.9)*. If:*

$$(\mathbf{z}^* - \mathbf{w}^*)^T \mathbf{Q}^- (\mathbf{z}^* - \mathbf{w}^*) \leq 0, \tag{6.3.1}$$

then both $(\mathbf{z}^*, \mathbf{z}^*)$ *and* $(\mathbf{w}^*, \mathbf{w}^*)$ *are also minimizers of the lifted problem. By extension,* $\mathbf{z}^*$ *and* $\mathbf{w}^*$ *are minimizers of the original quadratic problem* (6.0.2)*.*

Proof. By Proposition 6 (in the appendix), we have that:

$$\tilde{\ell}(\mathbf{z}^*, \mathbf{z}^*) + \tilde{\ell}(\mathbf{w}^*, \mathbf{w}^*) = 2\tilde{\ell}(\mathbf{z}^*, \mathbf{w}^*) + (\mathbf{z}^* - \mathbf{w}^*)^T \mathbf{Q}^- (\mathbf{z}^* - \mathbf{w}^*).$$

Re-arranging, and applying the optimality of $(\mathbf{z}^*, \mathbf{w}^*)$, it follows that:

$$(\mathbf{z}^* - \mathbf{w}^*)^T \mathbf{Q}^- (\mathbf{z}^* - \mathbf{w}^*) = \underbrace{\tilde{\ell}(\mathbf{z}^*, \mathbf{z}^*) - \tilde{\ell}(\mathbf{z}^*, \mathbf{w}^*)}_{\geq 0} + \underbrace{\tilde{\ell}(\mathbf{w}^*, \mathbf{w}^*) - \tilde{\ell}(\mathbf{z}^*, \mathbf{w}^*)}_{\geq 0} \geq 0.$$

Next, by assumption, $(\mathbf{z}^* - \mathbf{w}^*)^T \mathbf{Q}^- (\mathbf{z}^* - \mathbf{w}^*) \leq 0$, and therefore:

$$\tilde{\ell}(\mathbf{z}^*, \mathbf{z}^*) - \tilde{\ell}(\mathbf{z}^*, \mathbf{w}^*) + \tilde{\ell}(\mathbf{w}^*, \mathbf{w}^*) - \tilde{\ell}(\mathbf{z}^*, \mathbf{w}^*) = 0.$$

If we again re-arrange and apply the optimality of $(\mathbf{z}^*, \mathbf{w}^*)$, we find:

$$0 \leq \tilde{\ell}(\mathbf{z}^*, \mathbf{z}^*) - \tilde{\ell}(\mathbf{z}^*, \mathbf{w}^*) = \tilde{\ell}(\mathbf{z}^*, \mathbf{w}^*) - \tilde{\ell}(\mathbf{w}^*, \mathbf{w}^*) \leq 0,$$

and therefore we have:

$$\tilde{\ell}(\mathbf{z}^*, \mathbf{z}^*) = \tilde{\ell}(\mathbf{z}^*, \mathbf{w}^*) = \tilde{\ell}(\mathbf{w}^*, \mathbf{w}^*),$$

and by Lemma 5 the points $\mathbf{z}^*$ and $\mathbf{w}^*$ are both minimizers of the original quadratic problem (6.0.2). □

Note then that for any minimizer $(\mathbf{z}^*, \mathbf{w}^*)$ of the lifted problem (6.1.9), by the submodularity of $\tilde{f}$ and $\tilde{g}$ and the definition of the lattice $(\mathbb{R}^n_{\geq 0} \times \mathbb{R}^n_{\geq 0}, \ll)$, we can also construct the minimizer $(\mathbf{z}^* \curlyvee \mathbf{w}^*, \mathbf{z}^* \curlywedge \mathbf{w}^*)$ and its counterpart, $(\mathbf{z}^* \curlywedge \mathbf{w}^*, \mathbf{z}^* \curlyvee \mathbf{w}^*)$. If *any* of these minimizers satisfy the criteria of Lemma 8, then we immediately recover an optimal solution of the original quadratic problem.

The conditions required by Lemma 8 are in fact not only sufficient, but necessary. In particular, any two solutions that are on the diagonal must satisfy them. We defer its proof to the appendix because of its similarity to the proof of Lemma 8.

Lemma 9. *If* $(\mathbf{z}^*, \mathbf{z}^*)$ *and* $(\mathbf{w}^*, \mathbf{w}^*)$ *are minimizers of the lifted problem* (6.1.9)*, then:*

$$(\mathbf{z}^* - \mathbf{w}^*)^T \mathbf{Q}^- (\mathbf{z}^* - \mathbf{w}^*) \leq 0.$$

Lemmas 8 and 9 show that the easily verified quadratic form condition on the solutions to the lifted problem are both necessary and sufficient. In practice, we can simply solve the lifted problem and then check if the condition holds.

What might happen if the conditions of Lemma 8 are not satisfied, but we use its suggested minimizer anyways? It turns out that these solutions are still nearly optimal, with the distance from optimality measured using the same necessary and sufficient condition in Lemmas 8 and 9.

Lemma 10. *Let* $\mathbf{x}^* \in \mathbb{R}^n_{\geq 0}$ *be a minimizer of the original quadratic problem* (6.0.2)*, and* $(\mathbf{z}^*, \mathbf{w}^*) \in \mathbb{R}^n_{\geq 0} \times \mathbb{R}^n_{\geq 0}$ *be a minimizer of the lifted quadratic problem* (6.1.9)*. Then:*

$$\min\{\ell(\mathbf{z}^*), \ell(\mathbf{w}^*)\} \leq \ell(\mathbf{x}^*) + (\mathbf{z}^* - \mathbf{w}^*)^T \mathbf{Q}^- (\mathbf{z}^* - \mathbf{w}^*).$$

Proof. Again applying Proposition 6, we have:

$$\tilde{\ell}(\mathbf{z}^*, \mathbf{z}^*) + \tilde{\ell}(\mathbf{w}^*, \mathbf{w}^*) = 2\tilde{\ell}(\mathbf{z}^*, \mathbf{w}^*) + (\mathbf{z}^* - \mathbf{w}^*)^T \mathbf{Q}^- (\mathbf{z}^* - \mathbf{w}^*).$$

Then, applying the optimality of $(\mathbf{z}^*, \mathbf{w}^*)$, we upper bound the right hand side:

$$\begin{aligned}\tilde{\ell}(\mathbf{z}^*, \mathbf{z}^*) + \tilde{\ell}(\mathbf{w}^*, \mathbf{w}^*) &= 2\tilde{\ell}(\mathbf{z}^*, \mathbf{w}^*) + (\mathbf{z}^* - \mathbf{w}^*)^T \mathbf{Q}^- (\mathbf{z}^* - \mathbf{w}^*) \\ &\leq 2\tilde{\ell}(\mathbf{x}^*, \mathbf{x}^*) + (\mathbf{z}^* - \mathbf{w}^*)^T \mathbf{Q}^- (\mathbf{z}^* - \mathbf{w}^*).\end{aligned}$$

If we divide by two note that the minimum is less than the average, we have:

$$\begin{aligned}&\tilde{\ell}(\mathbf{z}^*, \mathbf{w}^*) \leq 2\tilde{\ell}(\mathbf{x}^*, \mathbf{x}^*) + (\mathbf{z}^* - \mathbf{w}^*)^T \mathbf{Q}^- (\mathbf{z}^* - \mathbf{w}^*) \\ \Rightarrow &\min\{\tilde{\ell}(\mathbf{z}^*, \mathbf{z}^*), \tilde{\ell}(\mathbf{w}^*, \mathbf{w}^*)\} \leq \tilde{\ell}(\mathbf{x}^*, \mathbf{x}^*) + \frac{1}{2}(\mathbf{z}^* - \mathbf{w}^*)^T \mathbf{Q}^- (\mathbf{z}^* - \mathbf{w}^*).\end{aligned}$$

Then, by Lemmas 3 and 4, this implies the result:

$$\min\{\tilde{\ell}(\mathbf{z}^*, \mathbf{z}^*), \tilde{\ell}(\mathbf{w}^*, \mathbf{w}^*)\} = \min\{\ell(\mathbf{z}^*), \ell(\mathbf{w}^*)\} \leq \ell(\mathbf{x}^*) + \frac{1}{2}(\mathbf{z}^* - \mathbf{w}^*)^T \mathbf{Q}^- (\mathbf{z}^* - \mathbf{w}^*).$$

□

This series of results suggests the following approach for quadratic problems that violate Assumption 1: lift the problem to a higher-dimensional one satisfying all the required assumptions, solve the new lifted problem, then check if the conditions for Lemma 8 are satisfied. If so, then construct the associated minimizer of the original problem. If the conditions are not satisfied, the value we computed immediately gives an additive bound on the suboptimality of the result.

CHAPTER 7

Examples and Computational Evaluation

In this section, we illustrate the proposed theoretical results on several numerical examples involving optimization on the lattices $\mathbb{R}^n_{\geq 0}$ and $2^{[n]}$. We compare against two state-of-the-art techniques: a direct application of the continuous submodular function minimization algorithms outlined by [Bac19], and the projected subgradient descent method proposed in [EJ20].

The algorithms for continuous submodular function minimization operate by discretizing the domain $\mathbb{R}^n_{\geq 0}$ into k discrete points in each dimension, converting the continuous optimization problem into a submodular minimization problem over a bounded integer lattice. In our examples, we consider the domain $[0, 1]^n \subseteq \mathbb{R}^n_{\geq 0}$ and set the discretization level to $k = 51$ unless otherwise specified. The algorithms for continuous submodular function minimization then solve an equivalent convex optimization problem (defined using a generalized Lovász extension for the integer lattice) using projected subgradient or Frank-Wolfe techniques. In our implementation, we use the Pairwise Frank-Wolfe algorithm to solve this convex problem, with all relevant results plotted in blue and labeled *Cont Submodular*.

The projected subgradient method is known to provide approximation guarantees even in the non-submodular case [EJ20], but as shown in Section 3.2, amounts to a specific choice of algorithms in our theory. The algorithm operates by solving an equivalent convex optimization problem–in particular, minimizing the Lovász extension of $g + H$ over $[0, 1]^n$–using projected subgradient descent. To implement this approach, we use IBM's CPLEX 12.8 constrained quadratic program solver in MATLAB to evaluate the function H (as expressed

in (3.1.1)) and use Polyak's rule for updating the step size. The relevant results are plotted in red, and labeled *PGD + CPLEX* in figures.

Our approach is agnostic to the choice of convex optimization and submodular set function minimization routines, so we also use CPLEX to evaluate H. To highlight the utility of an algorithm-agnostic approach, we also implement an active-set method for fast non-negative quadratic programming to evaluate H [BD97]. For the submodular set function minimization algorithm, we use the minimum-norm point algorithm from [FI11] as implemented in MATLAB by [Kra10], coupled with the semi-gradient lattice pruning strategy proposed by [IJB13] which has quadratic complexity and drastically reduces the problem size. Our results are plotted in black, and labeled *MNP + CPLEX* and *MNP + FNNQP* in figures.

The various methods are given identical cost functions to minimize, and are run until either convergence to suboptimality below 10^{-4} or a maximum of 100 iterations. The experiments were all run on a laptop with an AMD Ryzen 9 4900HS CPU and 16GB of RAM.

7.1 Regularized Sparse Regression

We first examine a regularized sparse regression problem, similar in spirit to (CS). Consider some $\mathbf{x} \in \mathbb{R}^n_{\geq 0}$, $\mathbf{D} \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^m$, and define the function $f : \mathbb{R}^n_{\geq 0} \to \mathbb{R}$ as:

$$f(\mathbf{x}) = \|\mathbf{D}\mathbf{x} - \mathbf{b}\|_2^2. \tag{7.1.1}$$

Then define the monotone submodular set function $g : 2^{[n]} \to \mathbb{R}$ as:

$$g(A) = \begin{cases} \lambda \left[(n-1) + \max(A) - \min(A) + |A| \right], & A \neq \emptyset, \\ 0 & A = \emptyset, \end{cases} \tag{7.1.2}$$

with $\lambda \in \mathbb{R}_{\geq 0}$, and $\max(A)$ and $\min(A)$ denoting the largest and smallest index element, respectively, in the set of indices A. This choice of g in the sparse regression problem (P)

places a high penalty on large sets of nonzero entries in the vector $\mathbf{x} \in \mathbb{R}^n_{\geq 0}$ that are far apart in index.

We generate a series of random problem instances with $m = n$ satisfying the assumption of submodularity on $\mathbb{R}^n_{\geq 0}$ and also the convexity condition of Corollary 1. Let chol : $\mathbb{R}^{n \times n} \to \mathbb{R}^{n \times n}$ denote a Cholesky decomposition of a positive semidefinite matrix, and construct the matrix $\mathbf{D}$ in (7.1.1) as:

$$\mathbf{D} = \text{chol}\left(\frac{1}{2}(\mathbf{C} + \mathbf{C}^T) + n\mathbf{I}\right), \quad \mathbf{C}_{ij} \sim \text{unif}(-1, 0), \text{ for all } i, j = 1, 2, ..., n.$$

This construction guarantees that the function f in (7.1.1) is both convex and submodular on $\mathbb{R}^n_{\geq 0}$, satisfying the conditions for Corollary 1. For the parameter $\mathbf{b} \in \mathbb{R}^m$, we use the signal in the top plot of Figure 7.1, and we set the regularization strength to $\lambda = 0.05$ so that both the functions f and g play nontrivial roles in the combined objective function.

We plot the results from each algorithm in Figure 7.1. Because the minimizer of the optimization problem is a representation of $\mathbf{b}$ using structured sparse columns of $\mathbf{D}$, we show the the reconstructed vector $\mathbf{Dx}$ produced by each algorithm in the second, third, and fourth plots of Figure 7.1. Because there is no reliance on discretization, both the projected subgradient descent and minimum-norm point algorithms produce a much smoother result, as expected.

In the bottom left plot of Figure 7.1, we show the cost achieved over iterations of each algorithm. The minimum-norm point converges almost immediately to the globally optimal cost, while the projected subgradient descent method takes longer to achieve the same cost. In contrast, the discretization error associated with the continuous submodular function minimization approach prevents it from ever achieving the true optimal cost, by a small amount.

Finally, over a small window of problem sizes, we show the running times of each algorithm in the bottom right plot of Figure 7.1. Interestingly, our approach presents a compromise between the slow optimality of the projected subgradient descent method and the

fast but inexact continuous submodular function minimization algorithm. Moreover, when we take advantage of the extra problem structure to use specialized algorithms, we achieve comparable running times to the continuous submodular minimization algorithm.

7.2 Signal Denoising

We next study a simple denoising example, where we consider a signal $\mathbf{x} \in \mathbb{R}^n_{\geq 0}$, which is corrupted by some additive disturbance $\mathbf{w} \in \mathbb{R}^n$, with $\mathbf{w} \sim \mathcal{N}(0, 0.1\mathbf{I})$. We would like to recover the signal $\mathbf{x}$ from the noisy measurements $\mathbf{y} = \mathbf{x} + \mathbf{w}$, under the assumption that the true signal $\mathbf{x}$ is smooth (meaning variations between adjacent entries ought to be small), and that the meaningful content arrived in a small number of contiguous sets of entries.

We can express the desire to match the noisy signal $\mathbf{y}$ with a smooth one with the convex and submodular function $f : \mathbb{R}^n \to \mathbb{R}$ defined as:

$$f(\mathbf{x}) = \frac{1}{2}\|\mathbf{x} - \mathbf{y}\| + \mu \sum_{i=1}^{n-1} \left(\mathbf{x}_i - \mathbf{x}_{i+1}\right)^2 . \tag{7.2.1}$$

The first term promotes matching the slightly corrupted signal, while the quadratic penalty on adjacent entries of $\mathbf{x} \in \mathbb{R}^n_{\geq 0}$ promotes smoothness.

Similarly, we can express the knowledge of a small and contiguous set of nonzero entries in the vector $\mathbf{x}$ with the monotone submodular set function $g : 2^{[n]} \to \mathbb{R}$ defined by:

$$g(A) = \lambda \left(|A| + \#\text{int}(A)\right), \tag{7.2.2}$$

where $\lambda \in \mathbb{R}_{\geq 0}$, and the function $\#\text{int}(A)$ counts the number of sets of contiguous indices in the set A. This set function is smallest on subsets with a small number of entries that are adjacent in index.

For experiments, we use the signal $\mathbf{x} \in \mathbb{R}^n_{\geq 0}$ shown in the top plot of Figure 7.2, with the noise-corrupted measurements $\mathbf{x}+\mathbf{w} = \mathbf{y} \in \mathbb{R}^n$ with an example shown in dotted orange. We then let $\mu = 0.8$ in (7.2.1) and $\lambda = 0.05$ in (7.2.2) so that the overall problem's cost function

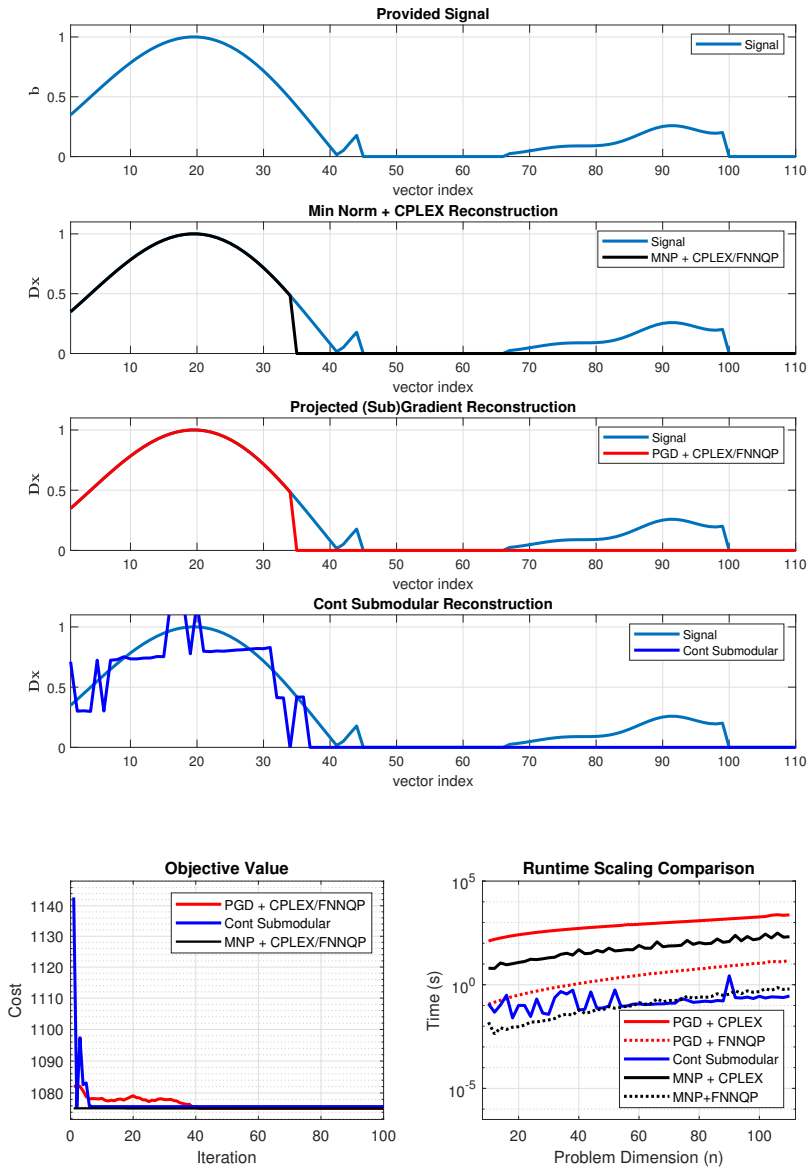

Figure 7.1: Results from the sparse regression problem simulations. The reconstructed signal representations using columns of $\mathbf{D}$ created by each algorithm are shown in the second, third, and fourth plot. Note the solutions produced by projected subgradient and the minimum-norm point algorithm are identical. We plot the cost function value over each algorithm's iterations in the bottom left, while in the bottom right we compare the running times of the algorithms over a small window of problem dimensions.

has nontrivial contributions from both the smoothness-promoting function and the sparsity-inducing regularizer. In this case, for the continuous submodular algorithm we discretize the compact set $[0, 1]^n \subseteq \mathbb{R}^n$ into $k = 51$ distinct values per index.

We show the resulting denoised signals in the second, third, and fourth plots in Figure 7.2, with the running time comparison over a small window of problem dimensions in the bottom right. The discretization of the domain in the continuous submodular function minimization approach produces artifacts in the reconstructed signal, whereas the result of the projected subgradient and minimum-norm point algorithms are smoother with smaller sets of nonzero entries. We see once more that our proposed minimum-norm point algorithm poses a compromise between speed and accuracy, providing guaranteed global optimality without the high running time of projected subgradient descent. Moreover, when we use more specialized algorithms for each sub-problem, we achieve competitive performance with the continuous submodular minimization algorithm.

We also compare the objective value achieved during the iterations of each algorithm for a single instance in the bottom left plot of Figure 7.2 with $n = 100$. Again, the minimum-norm point algorithm converges almost immediately to the minimum alongside the projected subgradient method, while the continuous submodular function minimization approach's discretization error prevents it from achieving full global optimality.

7.3 Price optimization with start-up costs

In price optimization problems, we are asked to determine prices for a set of products that maximizes the expected profit while considering any inter-product demand effects caused by these prices [IF16, IF17]. Usually this process relies on a simple predictive model for the relationship between the price of an item and its demand, which we can easily derive with a regression technique. Given a predictive model of the pricing-demand relationship and a characterization of our cost for each product, we want to determine the optimal pricing

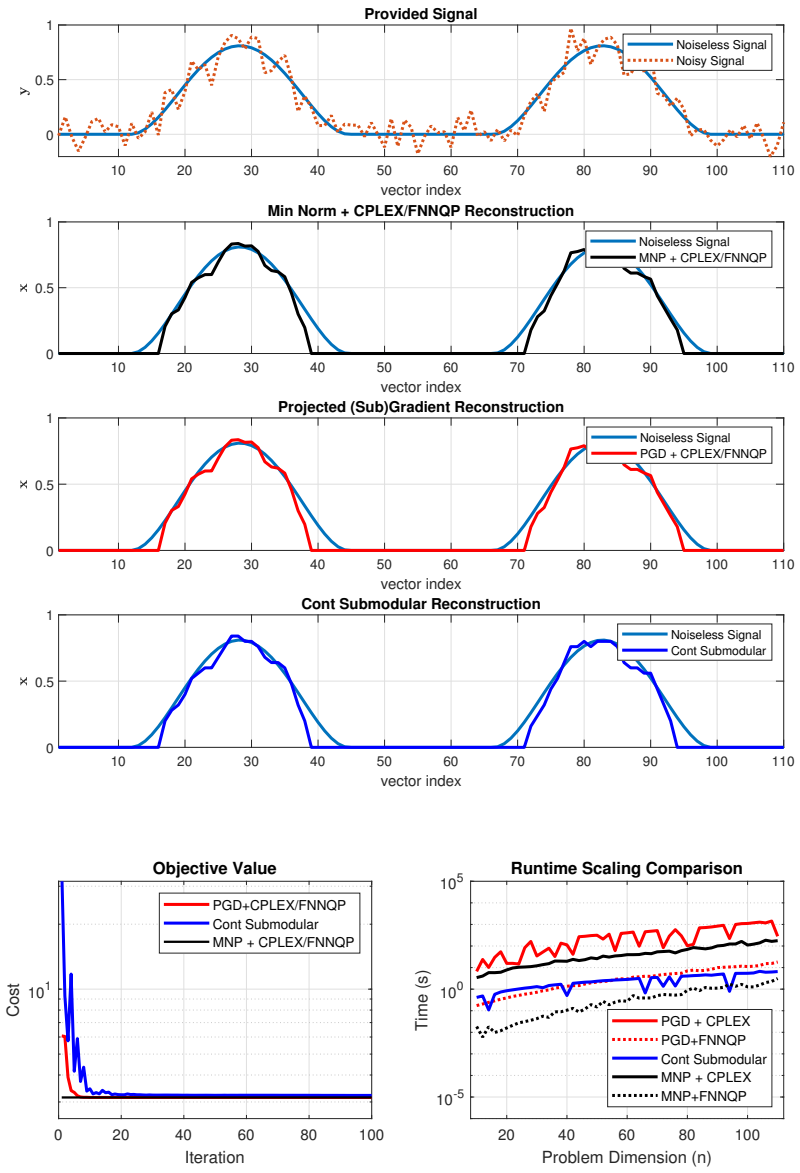

Figure 7.2: Results of the denoising problem simulations. The true signal and its noisy counterpart are shown in the top plot. The second, third, and fourth plots show the denoised signals produced by each of the three algorithms. Note that the results from the minimum-norm point algorithm and the projected subgradient descent method are identical. The bottom left plot shows the objective value across iterations for $n = 100$, and bottom right shows the running times of each algorithm for a window of problem dimensions.

strategy that maximizes our profit.

Let $\mathbf{c}_i \in \mathbb{R}_{\geq 0}$ and $\mathbf{p}_i \in \mathbb{R}_{\geq 0}$ denote the cost and retail price per unit, respectively, of each item of each item $i = 1, 2, ..., n$. Let the function $d : \mathbb{R}^{\mathrm{n}}_{\geq 0} \to \mathbb{R}^{\mathrm{n}}_{\geq 0}$ be the predictive demand model, meaning that given a set of prices $\mathbf{p}$ it estimates the number of sales (or demand) of the products. The estimated total profit of a pricing $\mathbf{p}$ can then be described by the function:

$$f(\mathbf{p}) = \sum_{i=1}^{n} (\mathbf{p}_i - \mathbf{c}_i) d(\mathbf{p})_i. \tag{7.3.1}$$

Without loss of generality, we assume there is a minimum loss we are willing to accept for each item, meaning there is a lower bound $\underline{\mathbf{p}} \in \mathbb{R}^{\mathrm{n}}_{\geq 0}$, and that if $\mathbf{p}_i = \underline{\mathbf{p}}_i$, we will not sell product i.

While the expression for profit (7.3.1) includes the cost of each item, it does not account for any start-up costs associated with providing them. In particular, to provide an item, we may have to order it from a supplier and have it shipped to our facilities, paying various logistical fees to do so. We pay these fees regardless of the *quantity* of products, meaning they are a function purely of which items we choose to stock. Moreover, in many cases these logistical costs are lumped together between items, such as when sourcing multiple products from the same supplier.

More mathematically, assume we have $k \in \mathbb{Z}_{>0}$ groups of products with shared start-up costs, with each group represented as a subset $G_i \subseteq [n]$, each with some start-up cost $\mathbf{w}_i$. Then the total incurred start-up costs of a subset of provided products S can be expressed with a set function $g : 2^{[\mathrm{n}]} \to \mathbb{R}$:

$$g(S) = \sum_{\substack{k \in [n] \\ S \cap G_k \neq \emptyset}} \mathbf{w}_i. \tag{7.3.2}$$

We apply this set function to the set of products we choose to sell, $\operatorname{supp}\left(\mathbf{p} - \underline{\mathbf{p}}\right) \subseteq [n]$. In this work, without loss of generality we let $\underline{\mathbf{p}} = 0$, which implies that an item priced at

$\mathbf{p}_i = \underline{\mathbf{p}}_i$ earns no reward and also has no impact on the demand of the other products. By carefully defining the demand model d and costs c, we can enforce this property for any desired minimum price $\underline{\mathbf{p}}$.

The true underlying demand model d is unknown in practice. In a small time window, however, we can use historical data to build a local linear approximation for it, $\hat{d} : \mathbb{R}^{\mathrm{n}}_{\geq 0} \to \mathbb{R}^{\mathrm{n}}_{\geq 0}$:

$$\hat{d}(\mathbf{p}) = \boldsymbol{\beta}\,\mathbf{p} + \boldsymbol{\alpha},$$

with $\boldsymbol{\beta} \in \mathbb{R}^{n \times n}$ and $\boldsymbol{\alpha} \in \mathbb{R}^n$. The entries $\boldsymbol{\beta}_{ij}$ describe the impact that the price of product i has on the demand for product j, sometimes referred to as the *elasticity of demands*[IF16, IF17]. Using this model, the estimated expected profit (7.3.1) is a quadratic function:

$$f(\mathbf{p}) = \sum_{i=1}^{n} (\mathbf{p}_i - \mathbf{c}_i)\hat{d}(\mathbf{p})_i = \mathbf{p}^T\,\boldsymbol{\beta}\,\mathbf{p} + \mathbf{p}^T(\boldsymbol{\alpha} - \boldsymbol{\beta}^T\,\mathbf{c}) - \mathbf{c}^T\,\boldsymbol{\alpha}\,.$$

Combining the expected profits with the start-up costs, we are faced with the optimization problem:

$$\begin{aligned} \underset{\mathbf{p}}{\text{minimize}} \quad & -\mathbf{p}^T\,\boldsymbol{\beta}\,\mathbf{p} - \mathbf{p}^T(\boldsymbol{\alpha} - \boldsymbol{\beta}^T\,\mathbf{c}) + \mathbf{c}^T\,\boldsymbol{\alpha} + g\left(\operatorname{supp}\left(\mathbf{p} - \underline{\mathbf{p}}\right)\right) \\ \text{subject to} \quad & \mathbf{p} \geq \underline{\mathbf{p}}. \end{aligned} \tag{7.3.3}$$

We create this scenario with real retail sales data collected from a UK-based online retail store available in the UCI Machine Learning Repository [DG17, CSG12]. We use this data to estimate the matrix $\boldsymbol{\beta} \in \mathbb{R}^{n \times n}$ and vector $\boldsymbol{\alpha} \in \mathbb{R}^n$ with simple ridge regression. To make the pricing problem (7.3.3) well-posed, we also enforce a weak diagonal dominance constraint on $\boldsymbol{\beta}$. In addition to making the problem well-posed, this constraint enforces the intuition that the most relevant factor in each product's demand is its own prices.

Even with a diagonal dominance constraint, the cross-terms $\boldsymbol{\beta}_{ij}$ with $i \neq j$ can easily be either positive or negative, depending on the demand and price relationships of the products. As a result, we cannot directly apply our parameterization method. We can, however, use the

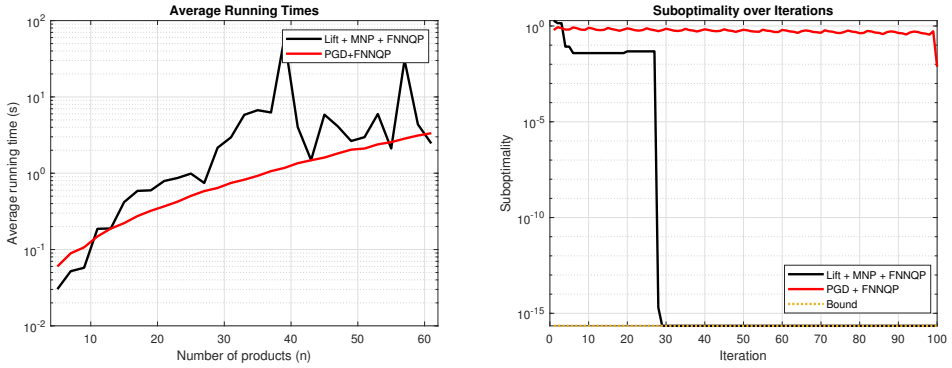

Figure 7.3: Results of the price optimization problem simulations. We show the running times of each algorithm for various problem sizes (left) and the achieved cost across iterations of the algorithms for a problem of size $n = 20$ (right). The dotted line below indicates the guaranteed lower bound on the optimal solution provided by our lift.

quadratic structure of (7.3.3) and follow the results of Section 6 to lift the pricing problem into a new quadratic problem amenable to our parameterization approach.

We compare our parameterization approach to solving (7.3.3) against the projected subgradient descent method applied directly to the original quadratic program for 100 iterations. This algorithm gives near-optimality guarantees, but explicitly computing the associated bound is NP-Hard. Alternatively, our quadratic lifting approach gives an easily computable additive suboptimality guarantee in Lemma 10 at the cost of solving a larger problem instance. This trade-off is highlighted in the plot of running times across varying problem sizes and the achieved cost across over iterations of each algorithm for an instance of $n = 20$ in Fig. 7.3.

We could also, in principle, use the continuous submodular minimization algorithm to solve the lifted quadratic problem. However, this approach will still suffer inaccuracy from the discretization step, and further, runs slower than the other algorithms that take advantage of the quadratic problem structure.

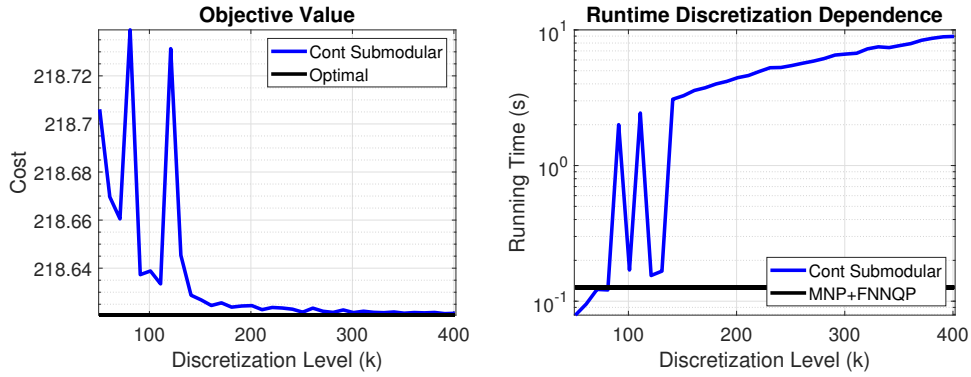

Figure 7.4: Results highlighting the role of the discretization resolution k on the continuous submodular algorithm's optimality (left) and running times (right) in an instance of the sparse regression problem with $n = 100$.

7.4 Discretization Error Dependence

In this section, we explore the relationship between the continuous submodular function minimization algorithm's discretization error and its running time. To this end, we ran instances of the sparse regression example with the modified range function penalty, using a discretization resolution in each dimension ranging from $k = 50$ to $k = 400$.

The minimum cost achieved at each discretization level k is shown in the left plot of Figure 7.4. Similarly, the associated running times of the algorithm are shown in the right-hand plot of Figure 7.4. Interestingly, near the value of $k = 250$, the achieved cost becomes effectively optimal, but the running time increases by an order of magnitude.

To give a coarse estimate on the origin of higher running times for projected subgradient descent and the minimum-norm point algorithms, we note that the computational cost of each iteration is dominated by the cost of computing the Lovàsz extension of H. This computation has time complexity $O(n \log n + nEO)$, where EO is the complexity of evaluating H. If H is evaluated through convex optimization, many generic interior-point methods have time complexity that is approximately $EO = O(n^3)$. Therefore, each iteration of the

minimum-norm point algorithm and the projected subgradient descent algorithm might have complexity on the order of $O(n \log n + n^4)$. When using the fast non-negative quadratic programming algorithm, however, each evaluation operation is typically much lower than the generic $O(n^3)$. Moreover, the lattice reduction technique of [IJB13] runs in approximately $O(n^2)$, and reduces the problem size drastically in many problems, as seen above.

CHAPTER 8

Conclusions

In this work, we showed that model-fitting problems with structure-promoting regularizers could be expressed as optimization problems defined over two connected lattices. Using submodularity theory, we derived conditions on these functions and their domains under which we can directly solve these problems exactly and efficiently. We focused on continuous and Boolean lattices, and derived conditions under which an agnostic combination of submodular set function minimization and convex optimization algorithms can compute the exact solution in polynomial time.

We then extended this theory to handle optimization problems with simple continuous or discrete budget constraints on the model parameters. We did this by naively adding the constraint to the cost with a Lagrange multiplier, but then used submodular function theory to solve for all possible Lagrange multiplier values with a single convex optimization problem. We also highlighted robust or adversarial optimization scenarios, where our exact solutions could provide subgradients to be used in globally convergent ascent methods.

Finally, we acknowledged there may be scenarios where our sufficient conditions are violated, and sought a way to weaken them without sacrificing our algorithm-agnostic approach. To do so, we identified a class of quadratic programming problems that can be lifted to problems satisfying our conditions. We then proved that the solutions of the lifted problem–which can then be found in polynomial time using our previously developed techniques–give provably optimal or near-optimal solutions to the original problem. Moreover, the additive approximation bound we provide is simple to compute, unlike existing guarantees in

literature that involve constants that are NP-Hard to compute.

Part II

Estimation

CHAPTER 9

Summary

Feedback control typically relies on an estimate of the system state provided by an estimation scheme. These estimates, however, are always affected by errors that have non-negligible impacts on control performance. Various stabilizing and safety-critical control frameworks address this issue, but all require some characterization of the current estimation error to determine when to apply more or less conservative control inputs. Current methods of bounding these errors either take a very coarse worst-case bound or employ computationally expensive time-varying set-valued methods.

This part of the thesis fills the missing gap in these works, presenting new deterministic worst-case error bounds for a state estimation scheme for generic nonlinear systems. Crucially, these error bounds can be efficiently computed in real-time and shrink or grow depending on the current system behavior and the current measurement quality. These new, lightweight, "online" error bounds can directly interface with the aforementioned measurement-robust control frameworks, resulting in less conservative control actions while retaining safety and stability guarantees.

CHAPTER 10

Introduction

In feedback control, one typically builds a full or partial-state feedback control law to accomplish the desired control task. This is particularly true in safety-critical scenarios, where one must prioritize the system's safety above all else. Almost all of these techniques for nonlinear control–particularly in safety-critical control–rely on knowledge of the system state. In practice, this means that the state feedback control law is designed first, and then implemented using, not the true system state, but an estimate from a separately designed state estimator.

While theoretically justified in some cases, the choice of estimation scheme can have major impacts on the overall control performance. It is well-known, for example, that stabilizing control laws for nonlinear systems may catastrophically fail when instead given a state *estimate* [Kha15].

Many modern nonlinear control techniques have been developed that accommodate the inherent imperfect knowledge of the state in a measurement-robust or uncertainty-aware framework. For example, the robust control Lyapunov and Barrier function frameworks have both been adapted to handle uncertainty in estimation [Fre96, CST21, AP23]. These frameworks, however, must assume some bound on the state estimation error and employ more conservative control actions depending on the magnitude of the error bound. In safety-critical control, for example, the measurement-robust barrier function framework effectively "inflates" the unsafe set and attempts to maintain a harsher safety criterion [CST21]. Beyond conservative control inputs, loose bounds can also lead to issues where a guaranteed safe control input does not exist.

What these existing measurement-robust frameworks lack is an estimation scheme that comes with error guarantees that *vary with time* in order to be less conservative, as noted in [AP23]. When equipped with such an estimation method, the measurement-robust control frameworks can adapt to be more or less conservative as the estimation error bounds grow or shrink. This adaptation may even be necessary in order to properly guarantee the safety or stability of the closed-loop system.

Many nonlinear estimation methods–even classic algorithms such as the Extended Kalman Filter (EKF)–already have some form of time-varying error guarantee [Kha15]. When these guarantees exist, however, they typically include a fixed inflation to accommodate the *worst-case* measurement noise or disturbances in the system, in a manner akin to input-to-state stability (ISS) bounds [SL16]. These bounds are then always "inflated" regardless of the *actually experienced* measurement noise or disturbances, even if the observer itself may be performing better in some periods than others.

Alternatively, there exist *set-valued observers* that hold on to *tight, time-varying* error bounds that may shrink or grow depending on the exact sequence of system outputs. Set-valued observers, however, are typically only available for highly structured or linear systems [KY22]. Even when available, these methods are often extremely computationally demanding, limiting their practical utility [ST99].

In this work, we present an estimation scheme based on numerical differentiation that directly targets these issues: it possesses deterministic, time-varying bounds that adapt online to the *experienced* measurement noise and system disturbances. These new guarantees can directly be handed to any measurement-robust control framework, where their time-varying nature permits more aggressive control actions when the estimation method is more confident. Moreover, since these guarantees are deterministic worst-case bounds, any measurement-robust control law based on these values will yield deterministic worst-case correctness proofs.

CHAPTER 11

Problem setup and background

We consider nonlinear control systems of the form:

$$\begin{aligned} \dot{x} &= f(x,u) \\ y &= h(x,u), \end{aligned} \tag{11.0.1}$$

where for all $t \in \mathbb{R}_{\geq 0}$, $x(t) \in \mathbb{R}^n$ is the state, $u(t) \in \mathbb{R}^m$ is the input, and $y(t) \in \mathbb{R}^p$ is the output. We assume that the map f is sufficiently regular for solutions of (11.0.1) to exist and be unique for all $t \geq 0$ and $x(0) \in \mathbb{R}^n$.

We are interested in estimating the state $x(t)$ of the system (11.0.1) at some time $t \in [t_0, t_N]$ given (possibly noise-corrupted) $N+1$ sampled-data measurements of $y(t)$ at a window of times $\{t_0, t_1, ..., t_N\}$. For this problem to be well-posed, we make the following standing assumption.

Assumption 1. *The control system* (11.0.1) *is* differentially observable of order d. *In particular, there exists a (possibly time-dependent) continuous function $\mathcal{L}$ that maps the d derivatives of the output $y(t)$ and input $u(t)$ to the state $x(t)$. More explicitly, $\mathcal{L}$ is such that:*

$$(y(t), \dot{y}(t), ..., y^{(d)}, u(t), \dot{u}(t), ..., u^{(d)}(t)) \xmapsto{\mathcal{L}} x(t).$$

As described in [Ber19, DFG01], places where the map $\mathcal{L}$ fails to exist are called singular *observations.*

11.1 Related work

A number of frameworks have addressed the interface between uncertainty in the system state and control in the context of stability and safety. We only discuss non-stochastic methods here, as they are closer to our work and its deterministic guarantees. From the perspective of stability, there are characterizations of ISS with respect to estimation errors, which guarantee that bounded errors cannot unboundedly destroy stability [KSW01]. In practice, however, there is no method for creating controllers that enforce this particular form of ISS for arbitrary nonlinear systems [Fre95].

Other works developed notions of robustness to estimation errors, leading to the concepts of *robust* control Lyapunov functions (CLFs) and measurement-robust control barrier functions (CBFs) [Fre96, CST21, AP23]. Both of these schools of thought, however, rely on a characterization of the estimation error that is valid at any instant of time. Loose offline characterizations of this uncertainty can lead to overly conservative controls, or worse, issues of feasibility from a lack of "guaranteed safe/stable" control actions.

On the estimation side of this problem, there are many methods for state estimation that possess a time-varying error bound. Perhaps the most straightforward to understand are set-valued observers, wherein a tight approximation (polyhedral, ellipsoidal, or a hyper-rectangle) of the possible states of the system is propagated through the dynamics at each step [ST99]. This tighter representation is less conservative, but comes at the cost of limited applicability and often prohibitively large amounts of computation and memory.

More familiar observers possess asymptotic guarantees, and even ISS-like guarantees are often established [SL16]. Even these ISS guarantees, however, rely on some *a priori estimate* of the worst-case measurement noise for all time, then inflate the estimation guarantees *for all time* accordingly. Some promising and recent notions of ISS observers with "fading memory" exist through the use of input-to-state *dynamical* stability, but the error bounds provided by these estimators are not always available in real-time to mitigate the issues in

measurement-robust control [DN15].

While developing observers for systems such as (11.0.1), one natural thought is to directly leverage Assumption 1 and consider derivative estimation equivalent to state estimation [Ber19]. This idea is not new, and many existing works connected numerical differentiation techniques to state estimation dating back to the 1990s, proving that these estimation techniques can produce globally bounded error [Dio07]. The existing guarantees, however, are strictly *offline*: given some estimate of the output's nonlinearity and magnitude of the noise, a single static error bound is provided for all time.

In this chapter, we show that these offline guarantees can be *significantly tightened* and made *online*. In particular, we prove a time-varying estimation error bound for Savitzky-Golay filtering that can be computed online with a simple multiplication by a fixed precomputed matrix. Moreover, we show in experiments that these online bounds are *orders of magnitude tighter* than the previously established offline bounds.

CHAPTER 12

Savitzky-Golay filtering

The differential observability condition effectively equates estimating the *state* of the control system with estimating its output and derivatives. As such, we will construct a method for estimating d derivatives of the output from sampled-data measurements that possesses the online error bounds we seek.

We propose a state estimation framework built on a classical scheme for numerical differentiation: polynomial least-squares, or *Savitzky-Golay filtering* [SG64].

In Savitzky-Golay filtering, we build a local (in time) approximation of the output signal y by fitting a window of $N+1$ samples in some interval $[t_i, t_f] \subseteq \mathbb{R}$ with a degree-d polynomial. We then estimate the d derivatives of y at some time in this window $\tau \in [t_i, t_f]$ by differentiating the polynomial approximation to y at τ. Finally, we apply the map $\mathcal{L}$ from Assumption 1 to produce a state estimate $\hat{x}(\tau)$. Notably, if the samples are uniformly spaced, this entire process becomes a single matrix multiplication with a fixed matrix computed offline.

In this section, we appeal to the following intuition: for a well-posed fitting procedure, the *residuals* from the least-squares regression should naturally measure fit quality. By residuals, we mean the misfit between the polynomial p and the output y at the sampled outputs. These residuals may be high or low depending on the *actual* measurement noise and nonlinearities at any given time, rather than being fixed a priori. Our main results formalize this intuition by connecting the online residuals to a deterministic worst-case error bound on the derivative estimation error during Savitzky-Golay filtering.

CHAPTER 13

Online error bounds

We assume that the output function h for the control system (11.0.1) is such that the output y is continuous and $d+1$-times differentiable. For simplicity of analysis, we discuss only scalar outputs ($m = 1$), as the generalization to higher dimensions is straightforward.

We then approximate the output locally with a degree-d polynomial $p : \mathbb{R} \to \mathbb{R}$ of the form:

$$p(t) = a_0 + a_1 t + \cdots + a_{d-1} t^{d-1} + a_d t^d. \tag{13.0.1}$$

Given $N+1$ measurements of the output y at times $\{t_0, t_1, ..., t_N\} \subseteq \mathbb{R}$, each corrupted by some noise signal $e(t_i)$, $t_i \in \{t_0, t_1, ..., t_N\}$, we determine the polynomial p by minimizing the squared error in the following optimization problem:

$$\underset{a \in \mathbb{R}^{d+1}}{\text{minimize}} \quad \sum_{i=0}^{N} \|[y(t_i) + e(t_i)] - p(t_i)\|_2^2. \tag{13.0.2}$$

Note that we do not have access to $y(t_i)$, only its noisy measurements $y(t_i) + e(t_i)$ at each sampled time.

13.1 Error bounds on derivatives

First, we state our main result which holds with equality.

Theorem 7. *Choose any subset of sample times $\mathcal{D} := \{s_0, s_1, ..., s_d\} \subseteq \{t_0, t_1, ..., t_N\}$ with cardinality $|\mathcal{D}| = d + 1 \leq N + 1$, and let $p : \mathbb{R} \to \mathbb{R}$ be any degree-d polynomial. Define*

the degree-d polynomial "residual interpolant" $r_{\mathcal{D}}$ associated with p, i.e., the polynomial such that:

$$y(s_i) - p(s_i) = r_{\mathcal{D}}(s_i) \quad \textit{for all } s_i \in \mathcal{D}. \tag{13.1.1}$$

Then for any $t \in [s_0, s_d]$, it holds that:

$$y^{(k)}(t) - p^{(k)}(t) = r_{\mathcal{D}}^{(k)}(t) + \frac{y^{(d+1)}(\xi)}{(d-k+1)!} \prod_{i=0}^{d-k} (t - \nu_i), \tag{13.1.2}$$

where $s_i \leq \nu_i \leq s_{i+k}$ for each $i = 0, 1, ..., d-k$ and $\xi \in [s_0, s_d]$.

Proof. Define the auxiliary function $Q : \mathbb{R} \to \mathbb{R}$ as:

$$Q(t) = y(t) - p(t) - r_{\mathcal{D}}(t).$$

By construction, Q is continuous and at least $d+1$-times differentiable with at least $d+1$ zeroes in the interval $[s_0, s_d]$. In particular, its zeros are each of the $s_i \in \mathcal{D}$. By repeated applications of Rolle's Theorem, $Q^{(k)}$ has at least $d-k$ zeros, each denoted by ν_i, with $\nu_i \in [s_i, s_{i+k}]$.

Consider another function $H : \mathbb{R} \to \mathbb{R}$ defined as:

$$H(z) = Q^{(k)}(z) - \alpha \prod_{i=0}^{d-k} (z - \nu_i), \tag{13.1.3}$$

for some $\alpha \in \mathbb{R}$. Note that for any chosen $t \in [s_0, s_d]$ with $t \neq \nu_i$ for all $i = 0, 1, ..., d-k$, there exists a choice of $\alpha \in \mathbb{R}$ such that $H(t) = 0$. We will derive an explicit expression for this α in terms of $y^{(d+1)}$.

Because $H(t) = 0$ for $t \in [s_0, s_d]$, then H is $d-k+1$ times differentiable with at least $d-k+2$ zeros in the interval $[s_0, s_d]$. In particular, $H(z) = 0$ when $z = \nu_i$, with $i = 0, 1, ..., d-k$, and also at the prescribed $z = t$. Again using repeated applications of Rolle's Theorem, the $d-k+1$ derivative of H then has at least one zero in the interval

$[s_0, s_d]$, meaning there exists some $\xi \in [s_0, s_d]$ (depending on t) such that:

$$\begin{aligned} H^{(d-k+1)}(\xi) &= Q^{(k+(d-k+1))}(\xi) - \alpha(d-k+1)! \\ 0 &= Q^{(d+1)}(\xi) - \alpha(d-k+1)! \\ &= y^{(d+1)}(\xi) - \alpha(d-k+1)! \\ \Rightarrow \ \alpha &= \frac{y^{(d+1)}(\xi)}{(d-k+1)!}, \end{aligned}$$

where in the third equality we abused the fact that p, $r_{\mathcal{D}}$, and $e_{\mathcal{D}}$ are degree-d polynomials.

Simply plugging this value for α into (13.1.3) and re-arranging, we find:

$$\begin{aligned} H(t) = 0 = Q^{(k)}(t) &- \frac{y^{(d+1)}(\xi)}{(d-k+1)!} \prod_{i=0}^{d-k}(t-\nu_i) \\ = y^{(k)} - p^{(k)} &- r_{\mathcal{D}}^{(k)}(t) \\ &- \frac{y^{(d+1)}(\xi)}{(d-k+1)!} \prod_{i=0}^{d-k}(t-\nu_i) \\ \Rightarrow \ y^{(k)} - p^{(k)} &= r_{\mathcal{D}}^{(k)}(t) + \frac{y^{(d+1)}(\xi)}{(d-k+1)!} \prod_{i=0}^{d-k}(t-\nu_i), \end{aligned}$$

for all $t \in [s_0, s_d]$ as desired. □

Note that Theorem 7 is an *equality*, meaning there is no tighter bound for a given polynomial p. Our choice of polynomial p, however, will change the values (and derivatives) of the residual interpolant $r_{\mathcal{D}}$, suggesting we choose p that minimizes its impact (e.g., least-squares).

The equality (13.1.2) also behaves in expected ways for specific cases. If there is no measurement error $e(t_i) = 0$ and the function y *is* a polynomial of degree at most d, then the interpolating polynomial p has zero residuals, $y^{(d+1)}$ is uniformly zero, and therefore (13.1.2) guarantees zero misfit everywhere in the interval. Similarly, when the number of points and degree of the polynomial are equal ($d = N$), (13.1.2) immediately recovers the guarantee associated with *interpolating* polynomials.

Despite its tightness, Theorem 7 relies on knowledge of parameters that we do not have access to in reality: the noise-free values of $y^{(d+1)}(\xi)$, the times ν_i, and the underlying

true misfit $y(t_i) - p(t_i)$. In practice, we only have access to the measured (noise-impacted) residuals, $y(t_i) + e(t_i) - p(t_i)$, and perhaps a uniform bound on the noise and value of $y^{(d+1)}(\xi)$. In the following corollary, we loosen the equality in (13.1.2) by only relying on these assumptions.

Corollary 4. *Assume that $|y^{(d+1)}(\xi)| \leq M$ for all $\xi \in [s_0, s_d]$, and that the measurement noise is uniformly bounded by $|e(s_i)| \leq E$ for all $s_i \in \mathcal{D}$. If the subset $\mathcal{D}$ has maximal inter-sample spacing $s_{i+1} - s_i \leq \delta$, then:*

$$\begin{aligned} |y^{(k)}(t) - p^{(k)}(t)| \leq \sum_{s_i \in \mathcal{D}} \left| l_i^{(k)}(t) \left(y(s_i) + e(s_i) - p(s_i) \right) \right| \\ + E \sum_{s_i \in \mathcal{D}} |l_i^{(k)}(t)| + M\delta^{d-k+1}, \end{aligned} \tag{13.1.4}$$

where $l_i : \mathbb{R} \to \mathbb{R}$ with $i = 0, 1, ..., d$ are the Lagrange basis polynomials for $\mathcal{D}$:

$$l_i(t) = \prod_{s_j \in \mathcal{D} \setminus \{s_i\}} \frac{t - s_j}{s_i - s_j}. \tag{13.1.5}$$

Proof. We begin by simply applying the triangle inequality to the right-hand side of (13.1.2):

$$\begin{aligned} |y^{(k)}(t) - p^{(k)}(t) &\leq |r_{\mathcal{D}}^{(k)}(t)| + \frac{|y^{(d+1)}(\xi)|}{(d-k+1)!} \prod_{i=0}^{d-k} |t - \nu_i| \\ &\leq |r_{\mathcal{D}}^{(k)}(t)| + \frac{M}{(d-k+1)!} \prod_{i=0}^{d-k} |t - \nu_i|. \end{aligned}$$

Then note that the interpolating polynomial $r_{\mathcal{D}}$ can be explicitly written as a function of its interpolation sites using the Lagrange basis for $\mathcal{D}$, as defined in (13.1.5):

$$\begin{aligned} \left| \frac{d^k}{dt^k} r_{\mathcal{D}}(t) \right| &= \left| \frac{d^k}{dt^k} \sum_{s_i \in \mathcal{D}} l_i(t) \left(y(s_i) - p(s_i) \right) \right| \\ &= \left| \sum_{s_i \in \mathcal{D}} l_i^{(k)}(t) \left(y(s_i) + e(s_i) - p(s_i) \right) - l_i^{(k)}(t) e(s_i) \right| \\ &\leq \sum_{s_i \in \mathcal{D}} \left| l_i^{(k)}(t) \left(y(s_i) + e(s_i) - p(s_i) \right) \right| + \left| l_i^{(k)}(t) e(s_i) \right| \\ &\leq \sum_{s_i \in \mathcal{D}} \left| l_i^{(k)}(t) \left(y(s_i) + e(s_i) - p(s_i) \right) \right| + E \left| l_i^{(k)}(t) \right|. \end{aligned}$$

Finally, we note that each of the ν_i is in the interval $[s_i, s_{i+k}]$. By assumption, each of these intervals is at most size $k\delta$. The product term is upper bounded by a choice of t that is at one end of the polynomial, which we can use as a lazy bound:

$$\begin{aligned}\frac{M}{(d-k+1)!}\prod_{i=0}^{d-k}|t-\nu_i| &\leq \frac{M}{(d-k+1)!}\prod_{i=0}^{d-k}(i+1)\cdot k\delta \\ &= M\delta^{d-k+1}.\end{aligned}$$

While this bound is valid, it can *easily* be sharpened by characterizing this product for the specific choice of t where estimation is relevant. Combining these terms, we recover the desired result. □

We have now removed any unknown quantities from Theorem 7, meaning Corollary 4 presents an *online-computable* bound characterizing the error in derivative estimation. Interestingly, this bound may vary in time with the fit *residuals* $y(t_i) + e(t_i) - p(t_i)$, which formalizes the intuition that "good polynomial fits" should produce better estimates, regardless of the standing assumptions on the system.

While the bounds in Corollary 4 are in principle "online computable", their practical value only holds if they are also computationally lightweight. Implementing both Savitzky-Golay filters *and evaluating Corollary 4's bounds* are computationally efficient. The filtering itself is a simple matrix multiplication of the current window of outputs by a fixed $N+1 \times N+1$ fitting matrix. The bounds require the measured residuals (one more matrix multiply and a vector subtraction) followed by a simple inner product with a (fixed, offline-computable) vector of $l_i^{(k)}(t)$ evaluations at the estimation time of interest $t \in [t_0, t_N]$.

Both Theorem 7 and Corollary 4 hold for an arbitrary degree-d polynomial p and its measured residuals. In practice, the Savitzky-Golay scheme uses the *least-squares* polynomial, which is clearly useful because it indirectly minimizes the individual measured residuals in the bound (13.1.4). Notably, if we are able to use an *interpolating* polynomial, then all residuals are zero and the guarantee given by Corollary 4 collapses to the usual guarantee

given for polynomial interpolation.

One of the main motivations for using least-squares over interpolation is the ability to "smooth out" the impact measurement noise. This property is implicit in (13.1.4), where we can reduce the magnitude of the terms involving measurement noise by shrinking the values of the Lagrange basis polynomials $l_i^{(k)}(t)$ associated with the subset $\mathcal{D}$. To shrink these values, we must select a subset of times $\mathcal{D} \subseteq \{t_0, t_1, ..., t_N\}$ that is spaced as far apart as possible. If the polynomial p was selected with least-squares, then the term associated to its residuals maintains the same uniform bound regardless of the subset of fitting points $\mathcal{D}$. This property holds as the ℓ_2 norm always upper bounds the ℓ_∞ norm (which we could choose to optimize instead while still applying Corollary 4). As we select a subset $\mathcal{D}$ with larger inter-sample times, however, the final term in (13.1.4) representing the output's deviation from polynomial grows. Choosing the best subset $\mathcal{D} \subseteq \{t_0, t_1, ..., t_N\}$ optimizes the trade off between *smoothing* and *accuracy*.

In principle, we could solve the combinatorial problem of choosing the subset $\mathcal{D} \subseteq \{t_0, t_1, ..., t_N\}$ with cardinality $|\mathcal{D}| = d + 1$ that minimizes the bound (13.1.4) each time we make a derivative estimate. This approach is clearly intractable, but we can easily approximate it by choosing a small family of different subsets (e.g., by parameterizing the subset by several choices of inter-sample spacing δ) and evaluating (13.1.4) for each subset choice online, always claiming the tightest guarantee achieved by this family. Similarly, we could select the subset $\mathcal{D}$ *a priori* by assuming some fixed maximum values for the measured residuals and optimizing (13.1.4) over $\mathcal{D}$, but this reduces the dynamic properties of the bound.

13.2 From derivatives to state

Up to this point, we have discussed only online error bounds for derivative estimation. We can transfer these error bounds from the space of derivative estimates to state estimates in

a number of ways, such as interval analysis techniques. In practice, we may use whichever method provides the tightest guarantees, but for completeness, we state a naive but immediate result for the special case of Lipschitz continuous observability maps.

Theorem 8. *Assume that the function $\mathcal{L}$ in Assumption 1 is globally Lipschitz continuous with Lipschitz constant L, and let $\hat{x}(t)$ denote the result of composing $\mathcal{L}$ with estimates of y and its d derivatives from a degree-d polynomial $p : \mathbb{R} \to \mathbb{R}$. Under the same setting as Corollary 4, there exists a nondecreasing function $\alpha : \mathbb{R} \to \mathbb{R}$ from the measured polynomial fit residuals to the estimation error:*

$$|x(t) - \hat{x}(t)| \leq \alpha \left(\sum_{s_i \in \mathcal{D}} |y(s_i) + e(s_i) - p(s_i)| \right).$$

If an explicit expression for the observation map $\mathcal{L}$ is not known, we could follow the steps proposed in [GM90] and solve the observation equations (or the equations governing the derivatives) for the state via Newton's method. The error bounds would then propagate through the convergence guarantees of this method.

The process outline above produces a state estimate for the time $t \in [t_0, t_N]$ where the derivative estimation takes place. Depending on when this time is chosen, there is necessarily a delay in the state estimate. We could choose to estimate derivatives at the most recent time $t_N \in [t_0, t_N]$, but differentiating fitting polynomials at their endpoints is notoriously inaccurate [BT04]. This difficulty is also reflected in the bounds given from Corollary 4, which are maximized when evaluated at t_0 and t_N.

We could also counteract the estimation delay by evolving the estimate from Theorem 8 forward with the differential equation model (11.0.1). We could even use an Extended Kalman Filter (EKF) initialized at the delayed estimate $\hat{x}(t)$ to both remove the delay and tighten the bounds from Theorem 8 when the EKF's local exponential convergence can be guaranteed [DFG01, Dio07].

CHAPTER 14

Experiments and evaluation

In this section, we validate the theoretical bounds from Corollary 4 in a couple simple examples. In each case, we show that our online error bounds are orders of magnitude tighter than more standard offline bounds, and vary with time depending on the system dynamics.

14.1 Lorenz Attractor System

We consider the Lorenz attractor system dynamics with a single output:

$$\begin{aligned} \dot{x}_1 &= \sigma \cdot (x_2 - x_1) \\ \dot{x}_2 &= x_1 \cdot (\rho - x_3) - x_2 \ , \qquad y = x_1, \\ \dot{x}_3 &= x_1 x_2 - \beta x_3 \end{aligned}$$

where we set the parameters $\sigma = 10$, $\rho = 28$, and $\beta = \frac{8}{3}$. We use a sampling frequency of 100Hz (inter-sample time $\delta = 0.01$ seconds) and apply a Savitzky-Golay filter to fit a degree $d = 3$ polynomial to sliding windows of 20 measurements. We differentiate this polynomial at the midpoint, and in our comparisons we use the delayed value of the system output and state (meaning we do not consider the effects of estimation lag). We derive the error bounds on the state estimate by performing interval analysis on an explicit expression for the map from Assumption 1.

To highlight the *online* nature of our bounds, we add bounded ($E = 0.5$) measurement errors to the system *only during times* $t \in [1.6, 3.3]$, and otherwise we have zero noise.

Crucially, we supply the bounds in Corollary 4 with the same value of $E = 0.5$ at all times, meaning we are *always* theoretically accommodating these measurement errors, even when none are present in the system. We also provide the bounds with a uniform bound $|y^{(d+1)}| \leq 96733$, which is valid for all time.

For comparison, we also plot some naive offline bounds derivable via Taylor series analysis, identical to those given in [Dio07]. We omit the derivation of these bounds here for brevity, but the interested reader may find them in Appendix D. In 14.1, we show the error in the estimated output derivatives on a log-scale plot, highlighting that our new online bounds are *orders of magnitude tighter*. In addition to always being tighter, these bounds may adapt naturally the noise in the measurements. Furthermore, the measurement errors are bounded $E = 0.5$ and so the output's value (i.e., the estimate of the state x_1) cannot be more accurate than this fundamental limit. Similarly, the output's derivative estimation error is always lower bounded by $\frac{2E}{\delta_{max}} \approx 20$, and $\frac{4E}{\delta_{max}^2} \approx 800$ for the second derivative, where δ_{max} is the largest possible inter-sample spacing. Our error bounds in Fig. 14.1 show that our method is provably near these fundamental limits in the noiseless regime.

This same phenomenon is apparent in Fig. 14.2, where we plot the state of the system (blue) alongside the state estimate (dashed red) with error bounds (red shading). The bounds naturally accommodate the extreme noise levels, but immediately tighten when no measurement errors are present. Moreover, the the map $\mathcal{L}$ from Assumption 1 naturally incorporates the system dynamics, which is why at some particular times the bounds increase, despite no measurement errors being added.

14.2 Ackerman Steering Model

We also consider a more physical system for a two-axle Ackerman steering model with "GPS" position outputs, whose states and dynamics are illustrated in Fig. 14.3.

We set the axle separation ℓ to 0.5, and use a sampling frequency of 100Hz ($\delta = 0.01$),

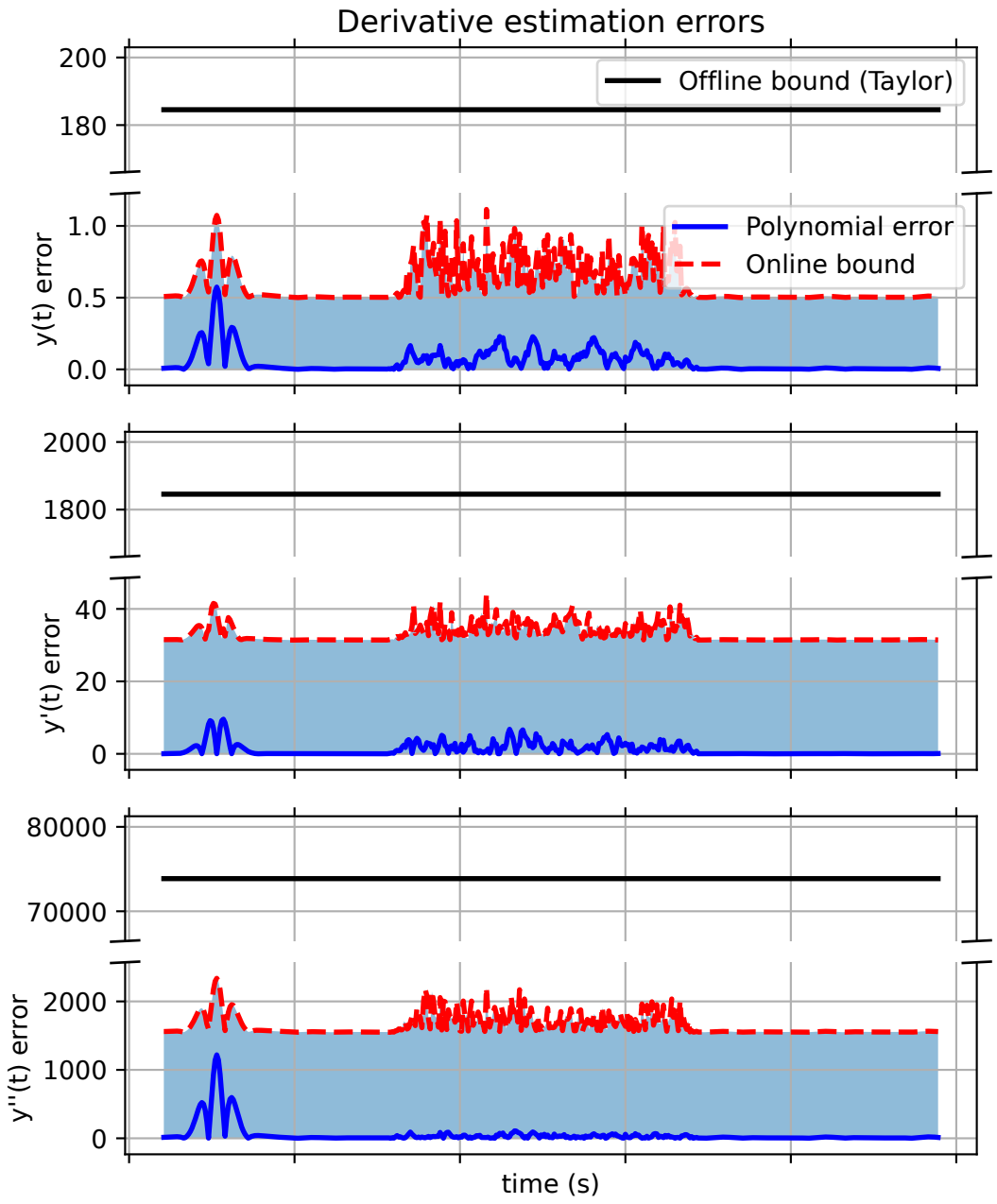

Figure 14.1: Error in the the derivative estimation for the Lorenz system. The true estimation error is shown in blue, with dashed red lines and shading indicating the online error bounds of Corollary 4. The solid black lines denote offline bounds.

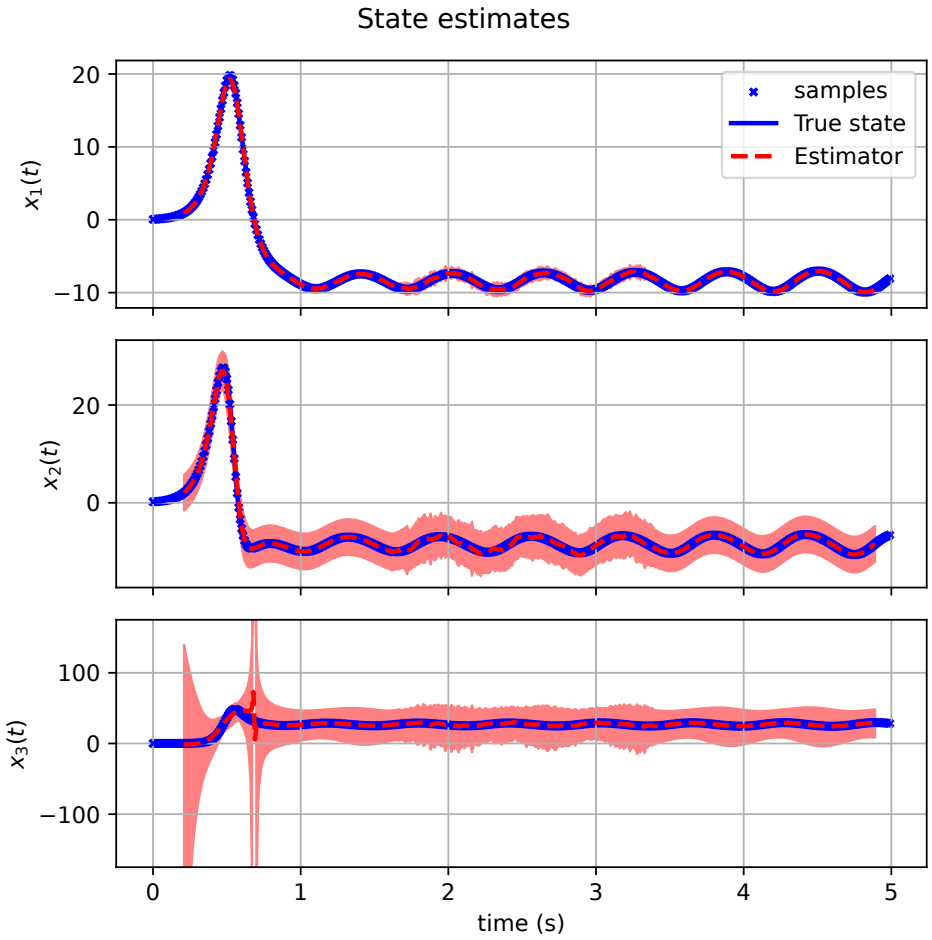

Figure 14.2: State estimates for the Lorenz system. The true state is shown in blue, with dashed red lines and red shading indicating the state estimate and online error bounds of Corollary 4. Note that the system produces a *singular measurement* around $t = 0.5$.

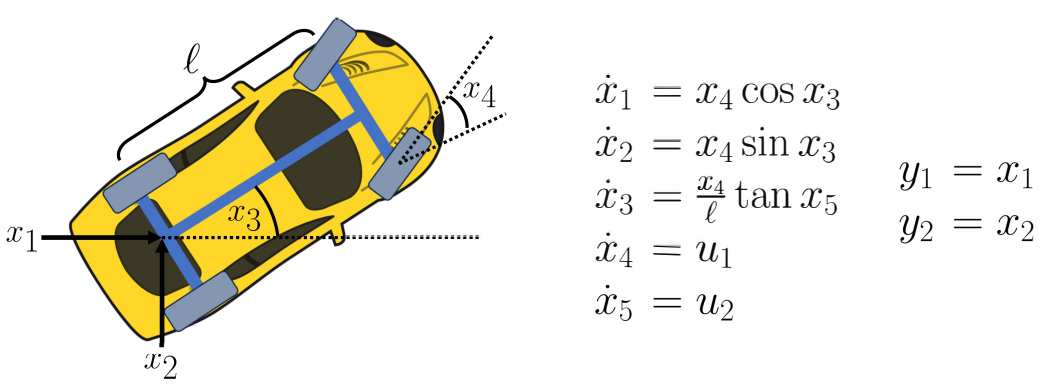

Figure 14.3: A diagram illustrating the states of the Ackerman steering model.

fitting a degree $d = 5$ polynomial to the data using a sliding window of 50 measurements. We inject bounded measurement errors with magnitude $E = 0.025$ only in the interval $t \in [4.9, 9.8]$.

Here we show the state estimates (with error bounds) in Fig. 14.4. Notably, in the interval where there were measurement errors, the bounds naturally inflate slightly and become less "smooth". The local dynamics of the vehicle, however, impact precisely how much inflation occurs.

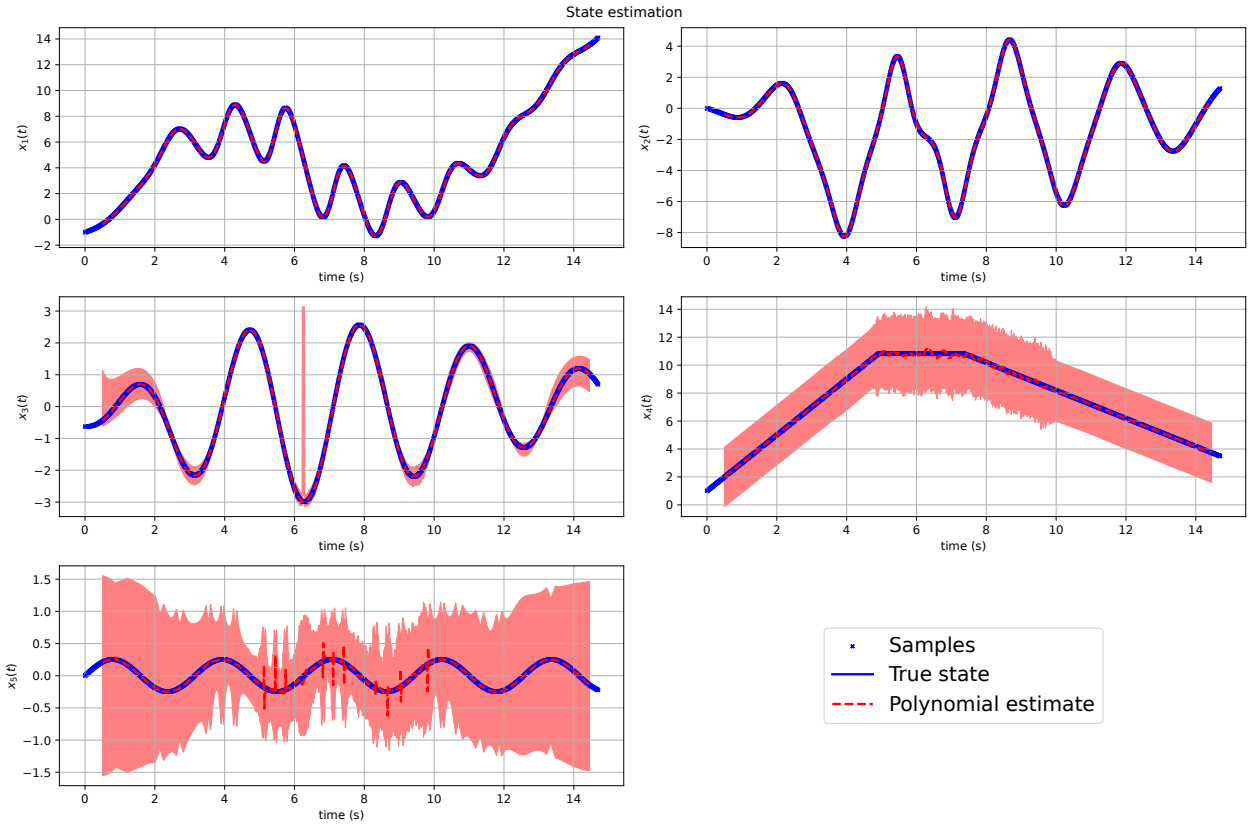

Figure 14.4: State estimates for the Ackerman model. The true state is shown in blue, with dashed red lines and red shading indicating the state estimate and online error bounds of Corollary 4. The spike in x_3 at $t \approx 6$ is caused by numerical angle wrapping artifacts, since x_3 lies on the manifold $\mathbb{S}$. Spikes in x_5, however, *are* caused by singular measurements, as estimating x_5 effectively computes the *curvature* of the vehicle path.

CHAPTER 15

Conclusions

In this chapter, we presented new deterministic worst-case error guarantees for a nonlinear state estimation scheme. Most importantly, our error bounds are easy to compute online and shrink or grow depending on the system behavior. These new "online" error bounds can easily interface with existing measurement-robust control frameworks, reducing the inherently conservative nature of these methods. A promising direction of future work would study the interaction of these observers with the control law, effectively ensuring not measurement-robust safety, which requires strong assumptions on the available control actions, but *certifiable* safety.

We validated this estimator and its guarantees with two different nonlinear systems, verifying their performance and tightness. In the future, this principle of relating fitting "residuals" to estimation errors could be extended to more classical estimators, proving new "online" error bounds for other families of nonlinear observers.

Part III

Appendices

APPENDIX A

Submodularity, Lattice Morphisms, and Least Squares

There is a massive body of work that identifies conditions under which compressed sensing problems of the form:

$$\underset{\mathbf{x}\in\mathbb{R}^n}{\text{minimize}} \quad \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + |\text{supp}\,(\mathbf{x})\,|, \tag{A.0.1}$$

for $\mathbf{A}\in\mathbb{R}^{m\times n}$ (with normalized unit norm columns, without loss of generality) and $\mathbf{b}\in\mathbb{R}^m$ can be efficiently solved by a convex relaxation of the ℓ_0 pseudo-norm to the ℓ_1 norm:

$$\underset{\mathbf{x}\in\mathbb{R}^n}{\text{minimize}} \quad \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \|\mathbf{x}\|_1,$$

with $\|\mathbf{x}\|_1 = \sum_{i=1}^n |\mathbf{x}_i|$. The majority of these conditions rely on the matrix $\mathbf{A}$ being "close to an isometry", or "nearly orthogonal". In this appendix, we highlight how these near-orthogonality conditions on the matrix $\mathbf{A}$ can be related to the assumptions made in this work.

Interestingly, any least-squares problem in the form of (A.0.1) can be written as a least-squares problem over $\mathbb{R}^n_{\geq 0}$, by considering auxiliary variables:

$$\mathbf{x} = \mathbf{x}^+ - \mathbf{x}^-, \quad \mathbf{x}^+, \mathbf{x}^- \in \mathbb{R}^n_{\geq 0}.$$

Using these new variables, the least squares problem (A.0.1) becomes:

$$\underset{\mathbf{x}^+,\mathbf{x}^-\in\mathbb{R}^n_{\geq 0}}{\text{minimize}} \quad \left\| \begin{bmatrix} \mathbf{A} & -\mathbf{A} \end{bmatrix} \begin{bmatrix} \mathbf{x}^+ \\ \mathbf{x}^- \end{bmatrix} - \mathbf{b} \right\|_2^2 + |\text{supp}\left(\mathbf{x}^+ - \mathbf{x}^-\right)|.$$

If we assume (without loss of generality) that at most one of $\mathbf{x}_i^+$ or $\mathbf{x}_i^-$ are nonzero for each $i = 1, 2, ..., n$, then we can equivalently write:

$$\underset{\mathbf{x}^+,\mathbf{x}^- \in \mathbb{R}^n_{\geq 0}}{\text{minimize}} \quad \begin{bmatrix} \mathbf{x}^+ \\ \mathbf{x}^- \end{bmatrix}^T \begin{bmatrix} \mathbf{A}^T\mathbf{A} & -\mathbf{A}^T\mathbf{A} \\ -\mathbf{A}^T\mathbf{A} & \mathbf{A}^T\mathbf{A} \end{bmatrix} \begin{bmatrix} \mathbf{x}^+ \\ \mathbf{x}^- \end{bmatrix} - 2\mathbf{b}^T \begin{bmatrix} \mathbf{A} & -\mathbf{A} \end{bmatrix} \begin{bmatrix} \mathbf{x}^+ \\ \mathbf{x}^- \end{bmatrix}$$
$$+ |\text{supp}\left(\mathbf{x}^+\right)| + |\text{supp}\left(\mathbf{x}^-\right)|.$$

In this lifted problem, Assumption 1 states that the cost function must be submodular on $\mathbb{R}^n_{\geq 0} \times \mathbb{R}^n_{\geq 0}$. For our lifted problem's cost function, this assumption is equivalent to the condition:

$$\begin{aligned} \left(\mathbf{A}^T\mathbf{A}\right)_{ij} &\leq 0, \quad \text{for all } i \neq j \\ -\left(\mathbf{A}^T\mathbf{A}\right)_{ij} &\leq 0, \quad \text{for all } i, j. \end{aligned}$$

This set of conditions in turn implies that $\left(\mathbf{A}^T\mathbf{A}\right)_{ii} \geq 0$ for all i, which is always satisfied, but also that $\left(\mathbf{A}^T\mathbf{A}\right)_{ij} = 0$ for all $i \neq j$.

By this analysis, any arbitrary least-squares problem with a monotone subset penalty can be converted to a nonnegative least-squares problem satisfying Assumptions 1-3 and the required convexity for Theorem 2 if $\mathbf{A}$ is orthogonal. The nearness of the matrix $\mathbf{A}$ to satisfying this condition is often measured with the notion of its *coherence*:

$$\max_{i \neq j} \left(\mathbf{A}^T\mathbf{A}\right)_{ij},$$

which is commonly used to identify well-structured instances of least-squares problems [Rau10].

APPENDIX B

Continuous Budget Constraints

In this appendix, we prove the relevant results for continuous budget constraints. We let $f_i : \mathbb{R}_{\geq 0} \to \mathbb{R}$ and $W_i : \mathbb{R}_{\geq 0} \to \mathbb{R}$ be continuous functions such that $f_i(0) = W_i(0) = 0$ for all $i = 1, 2, ..., n$. We further assume that each W_i is strictly increasing for each i. Then define the function $H_i : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\leq 0}$:

$$H_i(\alpha) = \min_{\mathbf{z} \geq 0} f_i(\mathbf{z}) + \alpha W_i(\mathbf{z}). \tag{B.0.1}$$

We first note that H_i is monotone in α.

Proposition 2. The function $H_i : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\leq 0}$ is monotone in α for all $i = 1, 2, ..., n$. It is strictly increasing for all $\alpha \in [0, c]$, where $c \in \mathbb{R}_{\geq 0}$ is the smallest constant such that $H_i(c) = 0$. Additionally, H_i is constant and zero on the interval $[c, \infty[$.

Proof. Consider $\alpha, \beta \in \mathbb{R}_{\geq 0}$, with $\alpha \leq \beta$, and define the points $\mathbf{z}^\alpha \in \mathbb{R}_{\geq 0}$ and $\mathbf{z}^\beta \in \mathbb{R}_{\geq 0}$ as:

$$\mathbf{z}^\alpha \in \operatorname*{argmin}_{\mathbf{z} \geq 0} f_i(\mathbf{z}) + \alpha W_i(\mathbf{z}),$$
$$\mathbf{z}^\beta \in \operatorname*{argmin}_{\mathbf{z} \geq 0} f_i(\mathbf{z}) + \beta W_i(\mathbf{z}).$$

Note that for any $\alpha \in \mathbb{R}_{\geq 0}$, because $\mathbf{z} = 0$ is a feasible point in the minimization defined in (B.0.1):

$$\begin{aligned} H_i(\alpha) &= \min_{\mathbf{z} \geq 0} f_i(\mathbf{z}) + \alpha W_i(\mathbf{z}) \\ &\leq f_i(0) + \alpha W_i(0) = 0, \end{aligned}$$

thus H_i is bounded above by zero. Moreover, observe that by optimality of $\mathbf{z}^\alpha$:

$$H_i(\alpha) = f_i(\mathbf{z}^\alpha) + \alpha W_i(\mathbf{z}^\alpha) \leq f_i(\mathbf{z}) + \alpha W_i(\mathbf{z}), \quad \text{for all } \mathbf{z} \geq 0.$$

Moreover, because $W_i(0) = 0$ and W_i is increasing, $W_i(\mathbf{z}) \geq 0$. Then, because $\alpha \leq \beta$:

$$\begin{aligned} H_i(\alpha) &= f_i(\mathbf{z}^\alpha) + \alpha W_i(\mathbf{z}^\alpha) \\ &\leq f_i(\mathbf{z}) + \alpha W_i(\mathbf{z}) \\ &\leq f_i(\mathbf{z}) + \beta W_i(\mathbf{z}), \quad \text{for all } \mathbf{z} \geq 0. \end{aligned}$$

This inequality is strict when $\alpha < \beta$ and $W_i(\mathbf{z}^\alpha) \neq 0$, or equivalently $H_i(\alpha) < 0$. In particular, because $\mathbf{z}^\beta \geq 0$:

$$H_i(\alpha) \leq f_i(\mathbf{z}^\beta) + \beta W_i(\mathbf{z}^\beta) = H_i(\beta),$$

with strict inequality when $H_i(\alpha) < 0$. Therefore H_i is monotone and strictly increasing for all $\alpha \in \mathbb{R}_{\geq 0}$ such that $H_i(\alpha) < 0$. Because it is also bounded above by zero, monotonicity implies that once $H_i(c) = 0$ for some $c \in \mathbb{R}_{\geq 0}$, it is zero for all $\beta \geq c$. □

Let $g : 2^{[n]} \to \mathbb{R}$ be a monotone submodular set function, and consider a family of optimization problems parameterized by $\mu \in \mathbb{R}_{\geq 0}$:

$$\underset{A \in 2^{[n]}}{\text{minimize}} \quad g(A) + \sum_{i \in A} H_i(\mu). \tag{B.0.2}$$

Given Proposition 2, we know that $H_i(0) \leq 0$ for all $i = 1, 2, ..., n$. If there exists an $i \in [n]$ such that $H_i(0) = 0$, Proposition 2 further states that $H_i(\alpha)$ is also zero for all $\alpha \geq 0$. Moreover, because g is monotone, we know:

$$\begin{aligned} g(A) + \sum_{i \in A} H_i(\alpha) &= g(A) + \sum_{i \in A \setminus \{j\}} H_i(\alpha) \\ &\geq g(A \setminus \{j\}) + \sum_{i \in A \setminus \{j\}} H_i(\alpha). \end{aligned}$$

In words, because g is monotone and $H_i(\alpha)$ is zero for all α, we can always reduce the cost of a subset by removing i. Equivalently, we can simply remove i from the ground set of elements.

We then follow the analysis in [Bac13], generalizing as needed to accommodate for the non-strict monotonicity of H_i.

Proposition 3. (Proposition 8.2 in [Bac13]) Let A^α and A^β be minimal (i.e., smallest in size) minimizers for (B.0.2) with respective parameters α and β, with $\alpha < \beta$. Then $A^\beta \subseteq A^\alpha$.

Proof. By the optimality of A^α and A^β, we have:

$$g(A^\alpha) + \sum_{i \in A^\alpha} H_i(\alpha) \leq g(A^\alpha \cup A^\beta) + \sum_{i \in A^\alpha \cup A^\beta} H_i(\alpha) \tag{B.0.3}$$

$$g(A^\beta) + \sum_{i \in A^\beta} H_i(\beta) \leq g(A^\alpha \cap A^\beta) + \sum_{i \in A^\alpha \cap A^\beta} H_i(\beta). \tag{B.0.4}$$

If we sum these inequalities and apply the submodularity of g, we have:

$$\begin{aligned} g(A^\alpha \cup A^\beta) + g(A^\alpha \cap A^\beta) + &\sum_{i \in A^\alpha \cup A^\beta} H_i(\alpha) + \sum_{i \in A^\alpha \cap A^\beta} H_i(\beta) \\ &\geq g(A^\alpha) + g(A^\beta) + \sum_{i \in A^\alpha} H_i(\alpha) + \sum_{i \in A^\beta} H_i(\beta) \\ &\geq g(A^\alpha \cup A^\beta) + g(A^\alpha \cap A^\beta) + \sum_{i \in A^\alpha} H_i(\alpha) + \sum_{i \in A^\beta} H_i(\beta). \end{aligned} \tag{B.0.5}$$

Subtracting equations (B.0.3) and (B.0.4) from (B.0.5), we have:

$$\begin{aligned} \sum_{i \in A^\alpha \cup A^\beta} H_i(\alpha) + \sum_{i \in A^\alpha \cap A^\beta} H_i(\beta) &\geq \sum_{i \in A^\alpha} H_i(\alpha) + \sum_{i \in A^\beta} H_i(\beta) \\ \Rightarrow \sum_{i \in A^\beta \setminus A^\alpha} [H_i(\beta) - H_i(\alpha)] &\leq 0. \end{aligned} \tag{B.0.6}$$

By Proposition 2, as $\alpha < \beta$, each $H_i(\beta) - H_i(\alpha)$ in the summation (B.0.6) is strictly positive, or $H_i(\alpha) = H_i(\beta) = 0$. But if $H_i(\alpha) = H_i(\beta) = 0$, as g is monotone, we may remove i from both A^α and A^β and decrease the cost in (B.0.2), contradicting the minimality of A^α and A^β.

By this argument, the left-hand side of inequality (B.0.6) is the sum of strictly positive terms. However, it is bounded above by zero, so it must therefore be the empty summation, i.e., $A^\beta \setminus A^\alpha = \emptyset$, and therefore $A^\beta \subseteq A^\alpha$. □

We now identify a related convex optimization problem:

$$\underset{\mathbf{u}\in\mathbb{R}^n_{\geq 0}}{\text{minimize}} \;\; g_L(\mathbf{u}) + \sum_{i=1}^{n} \int_{\epsilon}^{\epsilon+\mathbf{u}_i} H_i(\alpha) d\alpha. \tag{B.0.7}$$

A classical result in submodular function theory establishes that the Lovàsz extension g_L is convex if and only if g is submodular [Lov83]. Moreover, $\int_{\epsilon}^{\epsilon+\mathbf{u}_i} H_i(\alpha) d\alpha$ is convex if and only if H_i is monotone in α, which is true by Proposition 2. Therefore, problem (B.0.7) is a convex optimization problem.

We now establish a relationship between the parameterized family of set function minimization problems (B.0.2) and the convex optimization problem (B.0.7).

Proposition 4. (Proposition 8.3 in [Bac13]) Given the (minimal) solutions A^α to the set function minimization problem (B.0.2) for all values of the parameter $\alpha \geq \epsilon$, define the vector $\mathbf{u}^* \in \mathbb{R}^n_{\geq 0}$ defined by:

$$\mathbf{u}^*_i = \sup\left(\{\alpha \in \mathbb{R}_{\geq 0} \mid i \in A^\alpha\}\right).$$

Then the vector $\mathbf{u}^*$ is the minimizer of the convex optimization problem (B.0.7).

Proof. For $\alpha \geq 0$ small enough (as, without loss of generality, $H_i(0) < 0$ for all i), we have $H_i(\alpha) < 0$ for all $i = 1, 2, ..., n$. Because g is monotone, for this α, the optimal A^α is equal to $\{1, 2, ..., n\}$, and thus $\mathbf{u}$ is well defined for all $i = 1, 2, ..., n$.

For simplicity, we use the notation $\{\mathbf{u} \geq \mu\}$ to denote the set:

$$\{\mathbf{u} \geq \mu\} = \{i \in \{1, 2, ..., n\} \mid \mathbf{u}_i \geq \mu\},$$

for any $\mathbf{u} \in \mathbb{R}^n$ and $\mu \in \mathbb{R}$. Then for any $\mu \geq 0$, we have:

$$\begin{aligned} g_L(\mathbf{u}) + \sum_{i=1}^{n} \int_{\epsilon}^{\epsilon+\mathbf{u}_i} H_i(\mu)d\mu &= g_L(\mathbf{u} + \mathbf{1}\epsilon) - \epsilon g(\{1, 2, .., n\}) + \sum_{i=1}^{n} \int_{\epsilon}^{\epsilon+\mathbf{u}_i} H_i(\alpha)d\alpha \\ &= \int_{0}^{\infty} g(\{\mathbf{u} + \mathbf{1}\epsilon \geq \mu\})d\mu + \sum_{i=1}^{n} \int_{\epsilon}^{\epsilon+\mathbf{u}_i} H_i(\alpha)d\alpha - \epsilon g(\{1, 2, ..., n\}) \\ &= \int_{\epsilon}^{\infty} \left[g(\{\mathbf{u} + \mathbf{1}\epsilon \geq \mu\}) + \sum_{i=1}^{n} \mathbb{1}_{\{\mathbf{u_i}+\epsilon \geq \mu\}} H_i(\mu) \right] d\mu, \end{aligned} \tag{B.0.8}$$

where we used the indicator function defined as:

$$\mathbb{1}_{\{\mathbf{u_i}^* + \epsilon \geq \mu\}} = \begin{cases} 1, & \mathbf{u}_i^* + \epsilon \geq \mu \\ 0, & \text{otherwise.} \end{cases}$$

In the right-hand side of (B.0.8), every $\mu \geq \epsilon$ in the integral defines a set function minimization for which the optimal subset is A^μ. Because we constructed $\mathbf{u}^*$ as the minimizer to each of these optimal subsets, the value at $\mathbf{u}^*$ must be lower than all other $\mathbf{u}$, leading to the inequality:

$$\begin{aligned} g_L(\mathbf{u}^*) + \sum_{i=1}^{n} \int_{\epsilon}^{\epsilon+\mathbf{u}_i^*} H_i(\mu)d\mu &\leq \int_{\epsilon}^{\infty} \left[g(\{\mathbf{u} + \mathbf{1}\epsilon \geq \mu\}) + \sum_{i=1}^{n} \mathbb{1}_{\{\mathbf{u_j}+\epsilon \geq \mu\}} H_j(\mu) \right] d\mu \\ &= g_L(\mathbf{u}) + \sum_{i=1}^{n} \int_{\epsilon}^{\epsilon+\mathbf{u}_i^*} H_i(\mu)d\mu, \end{aligned}$$

for all other $\mathbf{u} \in \mathbb{R}^n_{\geq 0}$, and therefore $\mathbf{u}^*$ is optimal for (B.0.7). □

Proposition 4 establishes the relationship between the parameterized family of optimization problems (B.0.2) and the convex optimization problem (B.0.7). We state the next theorem without proof, as it requires no special modifications for our conditions.

Proposition 5. (Proposition 8.4 in [Bac13]) If $\mathbf{u}^*$ is the minimizer for the convex optimization problem (B.0.7), then for all $\mu \geq \epsilon$, the minimal minimizer of the corresponding set function minimization in (B.0.2) is:

$$A^\mu = \{i \in \{1, 2, ..., n\} \mid \mathbf{u}_i^* > \mu\}.$$

This sequence of propositions ultimately abuses the interpretation of the Lovàsz extension as an integral, and states that optimizing over the integral itself (the convex problem) and optimizing over the integrated functions for all integration variables (the set functions) is equivalent.

A noteworthy addendum is that in the definition of H_i, we could equivalently perform scalar minimization over a closed subset of $\mathbb{R}_{\geq 0}$, and the analysis would still follow through. This alteration would result in effectively "capping" the H_i functions from below, which retains the monotonicity properties necessary for the proofs.

APPENDIX C

A useful symmetry property

The lifted quadratic cost function $\tilde{c} : \mathbb{R}^{\mathrm{n}}_{\geq 0} \times \mathbb{R}^{\mathrm{n}}_{\geq 0} \to \mathbb{R}$ satisfies a convenient property that we abuse to prove several results. We prove it here.

Proposition 6. Let $\tilde{\ell}$ be defined as in (6.1.9). Then for any $(\mathbf{z}, \mathbf{w}) \in \mathbb{R}^{\mathrm{n}}_{\geq 0} \times \mathbb{R}^{\mathrm{n}}_{\geq 0}$, we have:

$$\tilde{\ell}(\mathbf{z}, \mathbf{z}) + \tilde{\ell}(\mathbf{w}, \mathbf{w}) = 2\tilde{\ell}(\mathbf{z}, \mathbf{w}) + (\mathbf{z} - \mathbf{w})^T \mathbf{Q}^- (\mathbf{z} - \mathbf{w}). \tag{C.0.1}$$

Proof. We proceed by directly computing:

$$\begin{aligned}
\tilde{\ell}(\mathbf{z}, \mathbf{z}) + \tilde{\ell}(\mathbf{w}, \mathbf{w}) &= f(\mathbf{z}) + g(\operatorname{supp}(\mathbf{z})) + f(\mathbf{w}) + g(\operatorname{supp}(\mathbf{w})) \\
&= \mathbf{z}^T \mathbf{Q}^+ \mathbf{z} + \mathbf{z}^T \mathbf{Q}^- \mathbf{z} + \mathbf{z}^T \mathbf{p} + \mathbf{w}^T \mathbf{Q}^+ \mathbf{w} + \mathbf{w}^T \mathbf{Q}^- \mathbf{w} + \mathbf{w}^T \mathbf{p} \\
&\quad + g(\operatorname{supp}(\mathbf{z})) + g(\operatorname{supp}(\mathbf{w})).
\end{aligned}$$

Then, adding and subtracting the missing cross term, we have:

$$\begin{aligned}
\tilde{\ell}(\mathbf{z}, \mathbf{z}) + \tilde{\ell}(\mathbf{w}, \mathbf{w}) &= \mathbf{z}^T \mathbf{Q}^+ \mathbf{z} + \mathbf{w}^T \mathbf{Q}^+ \mathbf{w} + \mathbf{z}^T \mathbf{p} + \mathbf{w}^T \mathbf{p} + g(\operatorname{supp}(\mathbf{z})) + g(\operatorname{supp}(\mathbf{w})) \\
&\quad + \mathbf{z}^T \mathbf{Q}^- \mathbf{z} + \mathbf{w}^T \mathbf{Q}^- \mathbf{w} \\
&= 2\tilde{\ell}(\mathbf{z}, \mathbf{w}) + \mathbf{z}^T \mathbf{Q}^- \mathbf{z} - 2\mathbf{z}^T \mathbf{Q}^- \mathbf{w} + \mathbf{w}^T \mathbf{Q}^- \mathbf{w} \\
&= 2\tilde{\ell}(\mathbf{z}, \mathbf{w}) + (\mathbf{z} - \mathbf{w}) \mathbf{Q}^- (\mathbf{z} - \mathbf{w})
\end{aligned}$$

□

We also provide a proof that the condition on the minimizers of the lifted problem is not only sufficient, but necessary.

Lemma 11. *If* $(\mathbf{z}^*, \mathbf{z}^*)$ *and* $(\mathbf{w}^*, \mathbf{w}^*)$ *are minimizers of the lifted problem* (6.1.9)*, then:*

$$(\mathbf{z}^* - \mathbf{w}^*)^T \mathbf{Q}^- (\mathbf{z}^* - \mathbf{w}^*) \leq 0.$$

Proof. Note that by the submodularity of $\tilde{\ell}$, if $(\mathbf{z}^*, \mathbf{z}^*)$ and $(\mathbf{w}^*, \mathbf{w}^*)$ are minimizers of the lifted problem (6.1.9), then their join and meet, $(\mathbf{z}^* \curlyvee \mathbf{w}^*, \mathbf{z}^* \curlywedge \mathbf{w}^*)$ and $(\mathbf{z}^* \curlywedge \mathbf{w}^*, \mathbf{z}^* \curlyvee \mathbf{w}^*)$, respectively, are also minimizers. Then, working through the proof of Lemma 8 backwards proves the result. □

APPENDIX D

Offline guarantees

We show here how the guarantee given by Corollary 4 can be immediately used to derive a fully *offline* guarantee for estimation. Recalling Corollary 4 here for clarity:

Corollary 5. *Assume that* $|y^{(d+1)}(\xi)| \leq M$ *for all* $\xi \in [s_0, s_d]$*, and that the measurement noise is uniformly bounded by* $|e(s_i)| \leq E$ *for all* $s_i \in \mathcal{D}$*. If the subset* $\mathcal{D}$ *has maximal inter-sample spacing* $s_{i+1} - s_i \leq \delta$*, then:*

$$\begin{aligned} |y^{(k)}(t) - p^{(k)}(t)| \leq \sum_{s_i \in \mathcal{D}} \left| l_i^{(k)}(t) \left(y(s_i) + e(s_i) - p(s_i) \right) \right| \\ + E \sum_{s_i \in \mathcal{D}} |l_i^{(k)}(t)| + M\delta^{d-k+1}, \end{aligned} \tag{D.0.1}$$

where $l_i : \mathbb{R} \to \mathbb{R}$ *with* $i = 0, 1, ..., d$ *are the Lagrange basis polynomials for* $\mathcal{D}$*:*

$$l_i(t) = \prod_{s_j \in \mathcal{D} \setminus \{s_i\}} \frac{t - s_j}{s_i - s_j}. \tag{D.0.2}$$

D.1 Explicitly bounding residuals

In order to transition the guarantee given by (D.0.1) from one that depends on the online residuals to a fully offline bound, we simply need to bound the worst-case values of these residuals. By assuming a global limit $||y^{(d+1)}(t)| \leq M$ for all $t \in \mathbb{R}_{\geq 0}$, we can immediately derive this bound with a simple application of the Taylor Remainder Theorem.

Corollary 6. *Assume* $|y^{(d+1)}| \leq M$ *for all* $t \in \mathbb{R}$*. Then for any time* $t \in \mathbb{R}$*, the least-squares*

polynomial $p_{LS} : \mathbb{R} \to \mathbb{R}$ fit to the window of $N+1$ data points satisfies the bound:

$$|y^{(k)}(t) - p_{LS}^{(k)}(t)| \leq M \left(\frac{L^{(k)}(t)\delta^{d+1}}{(d+1)!} \sqrt{2S_{2(d+1)}\left(\left\lceil \frac{N+1}{2} \right\rceil \right)} + \delta^{d-k+1} \right) + EL^{(k)}(t) \left(\sqrt{N+1} + 1 \right),$$

where we have defined:

$$L^{(k)}(t) = \sum_{s_i \in \mathcal{D}} |l_i^{(k)}(t)|.$$

Proof. First, consider any sequence of $N+1$ measurements, from which we have constructed the least-squares polynomial p_{LS}. Then (D.0.1) states:

$$\begin{aligned} |y^{(k)}(t) - p_{LS}^{(k)}(t)| &\leq \sum_{s_i \in \mathcal{D}} |l_i^{(k)}(t) \left(y(s_i) + e(s_i) - p_{LS}(s_i) \right)| + E \sum_{s_i \in \mathcal{D}} |l_i^{(k)}(t)| + M\delta^{d-k+1} \\ &\leq \sum_{s_i \in \mathcal{D}} |l_i^{(k)}(t)| \cdot |(y(s_i) + e(s_i) - p_{LS}(s_i)| + E \sum_{s_i \in \mathcal{D}} |l_i^{(k)}(t)| + M\delta^{d-k+1}. \end{aligned} \tag{D.1.1}$$

Now we also consider the degree d Taylor approximation to y, denoted $p_T : \mathbb{R} \to \mathbb{R}$, expanded about some time $t_0 \in [s_0, s_d]$. By construction, the Taylor approximation satisfies:

$$y(t) - P_T(t) = \frac{y^{(d+1)}(c)}{(d+1)!}(t - t_0)^{d+1}, \tag{D.1.2}$$

for some $c \in [s_0, s_d]$.

To finish our conversion to an offline bound, we need to characterize the weighted sum of the residual errors in (D.1.1). Consider any single residual from this summation and note:

$$|y(s_i) + e(s_i) - p_{LS}(s_i)| \leq \max_{s_i \in \mathcal{D}} |y(s_i) + e(s_i) - p_{LS}(s_i)| \tag{D.1.3}$$

$$\leq \max_{i=0,1,\ldots,N} |y(t_i) + e(t_i) - p_{LS}(t_i)| \tag{D.1.4}$$

$$\leq \sqrt{\sum_{i=0}^{N} |y(t_i) + e(t_i) - p_{LS}(t_i)|^2}, \tag{D.1.5}$$

where the final inequality arises by the fact that the ℓ_2 norm always upper bounds the ℓ_∞ norm. Next we recall that the least-squares polynomial achieves minimal ℓ_2 norm of its residuals, and further that the degree d Taylor polynomial p_T is *feasible* in the associated optimization problem. Therefore, we have the bound:

$$|y(s_i) + e(s_i) - p_{LS}(s_i)| \leq \sqrt{\sum_{i=0}^{N} |y(t_i) + e(t_i) - p_T(t_i)|^2}. \tag{D.1.6}$$

Next, by directly applying (D.1.2) and the triangle inequality for the ℓ_2 norm, we have:

$$|y(s_i) + e(s_i) - p_{LS}(s_i)| \leq \sqrt{\sum_{i=0}^{N} \left| \frac{y^{(d+1)}(c)}{(d+1)!}(t_i - t_0)^{d+1} + e(t_i) \right|^2} \tag{D.1.7}$$

$$\leq \sqrt{\sum_{i=0}^{N} \left| \frac{y^{(d+1)}(c)}{(d+1)!}(t_i - t_0)^{d+1} \right|^2} + \sqrt{\sum_{i=0}^{N} |e(t_i)|^2} \tag{D.1.8}$$

$$\leq \sqrt{\left(\frac{|y^{(d+1)}(c)|}{(d+1)!} \right)^2 \sum_{i=0}^{N} |t_i - t_0|^{2(d+1)}} + \sqrt{\sum_{i=0}^{N} |e(t_i)|^2} \tag{D.1.9}$$

$$= \frac{|y^{(d+1)}(c)|}{(d+1)!} \sqrt{\sum_{i=0}^{N} |t_i - t_0|^{2(d+1)}} + \sqrt{\sum_{i=0}^{N} |e(t_i)|^2}. \tag{D.1.10}$$

Next, we recall our global bounds, in particular that $|y^{(d+1)}(t)| \leq M$ for all $t \in \mathbb{R}$, and also that the measurement errors are bounded, $|e(t_i)| \leq E$ for all t_i. We immediately recover the following:

$$|y(s_i) + e(s_i) - p_{LS}(s_i)| \leq \frac{M}{(d+1)!} \sqrt{\sum_{i=0}^{N} |t_i - t_0|^{2(d+1)}} + \sqrt{(N+1)E^2} \tag{D.1.11}$$

$$= \frac{M}{(d+1)!} \sqrt{\sum_{i=0}^{N} |t_i - t_0|^{2(d+1)}} + E\sqrt{N+1}. \tag{D.1.12}$$

Lastly, we note that we may only claim the inequality given by Corollary 4 when considering a $t \in [s_0, s_d]$. We may place the expansion point for our Taylor approximation t_0 anywhere in our window of interest, and by lazily selecting t_0 to be the median in the window of $N+1$

points, we can bound:

$$\sqrt{\sum_{i=0}^{N} |t_i - t_0|^{2(d+1)}} \leq \sqrt{\sum_{i=1}^{\lceil \frac{N+1}{2} \rceil} 2\left(i \cdot \delta\right)^{2(d+1)}} \tag{D.1.13}$$

$$\leq \delta^{d+1} \sqrt{2 \sum_{i=1}^{\lceil \frac{N+1}{2} \rceil} i^{2(d+1)}}. \tag{D.1.14}$$

since the maximal inter-sample spacing is δ. We note here that the final summation is a *power sum*, typically denoted by:

$$S_{2(d+1)}\left(\left\lceil \frac{N+1}{2} \right\rceil\right) = \sum_{i=1}^{\lceil \frac{N+1}{2} \rceil} i^{2(d+1)}, \tag{D.1.15}$$

for which various closed-form expressions exist. A more naive (and definitively looser) bound for these terms could easily be derived by bounding every term $|t_i - t_0|$ by the maximal window length, instead deriving the bound:

$$\sqrt{\sum_{i=0}^{N} |t_i - t_0|^{2(d+1)}} \leq \sqrt{\sum_{i=0}^{N} (N\delta)^{2(d+1)}} \tag{D.1.16}$$

$$= \sqrt{N+1}(N\delta)^{(d+1)}. \tag{D.1.17}$$

Finally, we return to our bound for a single term (D.1.12), applying the power sum bound (D.1.15) and find:

$$|y(s_i) + e(s_i) - p_{LS}(s_i)| \leq \frac{M}{(d+1)!} \sqrt{\sum_{i=0}^{N} |t_i - t_0|^{2(d+1)}} + E\sqrt{N+1} \tag{D.1.18}$$

$$\leq \frac{M\delta^{d+1}}{(d+1)!} \sqrt{2 S_{2(d+1)}\left(\left\lceil \frac{N+1}{2} \right\rceil\right)} + E\sqrt{N+1}. \tag{D.1.19}$$

Or alternatively, choosing the looser bound (D.1.17):

$$|y(s_i) + e(s_i) - p_{LS}(s_i)| \leq \frac{M\sqrt{N+1}(N\delta)^{d+1}}{(d+1)!} + E\sqrt{N+1}. \tag{D.1.20}$$

Finally, we can return to the full weighted sum (D.1.1) to find:

$$|y^{(k)}(t) - p_{LS}^{(k)}(t)| \leq \sum_{s_i \in \mathcal{D}} |l_i^{(k)}(t)| \cdot \left[\frac{M\delta^{d+1}}{(d+1)!} \sqrt{2S_{2(d+1)} \left(\left\lceil \frac{N+1}{2} \right\rceil \right)} + E\sqrt{N+1} \right] \quad \text{(D.1.21)}$$

$$+ E \sum_{s_i \in \mathcal{D}} |l_i^{(k)}(t)| + M\delta^{d-k+1}. \quad \text{(D.1.22)}$$

For notational cleanliness, we define $L^{(k)}(t) = \sum_{s_i \in \mathcal{D}} |l_i^{(k)}(t)|$, and re-arrange terms to see:

$$|y^{(k)}(t) - p_{LS}^{(k)}(t)| \leq M \left(\frac{L^{(k)}(t)\delta^{d+1}}{(d+1)!} \sqrt{2S_{2(d+1)} \left(\left\lceil \frac{N+1}{2} \right\rceil \right)} + \delta^{d-k+1} \right) \quad \text{(D.1.23)}$$

$$+ EL^{(k)}(t) \left(\sqrt{N+1} + 1 \right). \quad \text{(D.1.24)}$$

□

The exact algebraic form of the bound in Corollary 6 is complicated, but it does exhibit some key behaviors we would expect. For example, hgiher derivative orders have weaker guarantees. In particular, the value of δ^{d-k+1} increases and the values of $L^{(k)}(t)$ increase monotonically with increasing k. Additionally, as the sampling time decreases (δ shrinks), the guarantees become tighter.

One key behavior exhibited in the bound from Corollary 6 is that it consists of two terms, one which is proportional to the measurement error bound E, and one which is proportional to the "ill-conditioning" of the target function M. This type of bound not only makes sense, but has its theoretical roots in ill-posed inverse problem theory [Dio07, Kir11].

D.2 Directly using least-squares

A final observation is that we could have derived the offline guarantee in Corollary 6 through the lens of a pure least-squares analysis.

In particular, Savitzky-Golay filtering solves the least squares problem:

$$\underset{a \in \mathbb{R}^{d+1}}{\text{minimize}} \quad \|Y + Z - Fa\|_2^2, \quad \text{(D.2.1)}$$

where $Y \in \mathbb{R}^{N+1}$ and $Z \in \mathbb{R}^{N+1}$ denote the measurement and noise vectors, and $F \in \mathbb{R}^{N+1\times d+1}$ is the relevant *Vandermonde* matrix of polynomial regression coefficients. Explicitly, these matrices are:

$$Y = \begin{pmatrix} y(t_0) \\ y(t_1) \\ \cdots \\ y(t_N) \end{pmatrix}, \quad Z = \begin{pmatrix} e(t_0) \\ e(t_1) \\ \cdots \\ e(t_N) \end{pmatrix}, \quad F = \begin{pmatrix} 1 & t_0 & t_0^2 & \cdots & t_0^d \\ 1 & t_1 & t_1^2 & \cdots & t_1^d \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & t_N & t_N^2 & \cdots & t_N^d \end{pmatrix}. \tag{D.2.2}$$

The solution of this optimization problem is a set of coefficients $a_{LS} \in \mathbb{R}^{d+1}$ for the least-squares polynomial. To evaluate it and its derivatives at a point, we may consider the "evaluation matrix" $B(t) \in \mathbb{R}^{d\times d+1}$:

$$B(t) = \begin{pmatrix} 1 & t & t^2 & \cdots & t^d \\ 0 & 1 & 2t & \cdots & dt^{d-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \cdots & d! \end{pmatrix}, \tag{D.2.3}$$

which is built such that $B(t)a_{LS} \in \mathbb{R}^d$ produces a vector containing the Savitzky-Golay filter estimates of the d derivatives of y at the time t.

As we did in the proof of Corollary 6, we note that the degree d Taylor approximating polynomial expanded about some t_0 is also feasible for the optimization problem (D.2.1). The Taylor approximation corresponds to a set of coefficients $a_T \in \mathbb{R}^{d+1}$, and in particular we know (via the Taylor Remainder Theorem) that:

$$Y = Fa_T + R_T, \tag{D.2.4}$$

where the errors R_T are derived from (D.1.2). We actually know from our work in Corollary 6 that these bounds will consist of two terms, one relating to the $d+1$ derivative bound M and one relating to the measurement noise bound E, with powers of the inter-sample spacing δ.

Using this simple observation, we can directly derive a simple offline bound.

Lemma 12. *Let the vector of remainders for a degree d Taylor approximation of y about $t \in \mathbb{R}$ be defined as R_T (as we did in* (D.2.4)*). Then the least-squares estimator has error bounded by:*

$$|y^{(k)}(t) - \hat{y}^{(k)}(t)| = |y^{(k)}(t) - [B(t)a_{LS}]_k| \leq |B_k(t)F^\dagger|_\infty \left(|R_T|_\infty + E\right), \tag{D.2.5}$$

where $F^\dagger \in \mathbb{R}^{d+1 \times N+1}$ denotes the Moore-Penrose pseudo-inverse of the matrix F.

Proof. Directly computing:

$$|y^{(k)}(t) - [B(t)a_{LS}]_k| = |y^{(k)}(t) - [B(t)a_T]_k + [B(t)a_T]_k - [B(t)a_{LS}]_k|. \tag{D.2.6}$$

Because we have considered the Taylor series expanded about the time t, the error in its approximation of $y^{(k)}(t)$ is exactly zero for all orders $k \leq d$. Then, using our definition of the residual vector in (D.2.4) and the closed form expression for the least-squares solution to the optimization problem (D.2.1), we have:

$$\begin{aligned}
|y^{(k)}(t) - [B(t)a_{LS}]_k| &= |[B(t)F^\dagger(Y + R_T)]_k - [B(t)F^\dagger(Y + Z)]_k| && \text{(D.2.7)}\\
&\leq |[B(t)F^\dagger(R_T - Z)]_k| && \text{(D.2.8)}\\
&\leq \|B(t)F^\dagger(R_T - Z)\|_\infty && \text{(D.2.9)}\\
&\leq \|B(t)F^\dagger\|_\infty \|R_T - Z\|_\infty && \text{(D.2.10)}\\
&\leq \|B(t)F^\dagger\|_\infty \left(\|R_T\|_\infty + \|Z\|_\infty\right) && \text{(D.2.11)}\\
&\leq \|B(t)F^\dagger\|_\infty \left(\|R_T\|_\infty + E\right), && \text{(D.2.12)}
\end{aligned}$$

where we slightly abused notation to use $\|\cdot\|_\infty$ to denote the ℓ_∞ vector norm and also the operator infinity norm of a matrix. □

These guarantees, while very direct, obfuscate the roles of the relevant design parameters. In particular, the smoothing properties of least-squares and the roles of the residual are all hidden within the complicated pseudo-inverse of the Vandermonde matrix F. We opted for the form o guarantees in the main body of this work because they provide a more explicit characterization of the tradeoffs between various design parameters in the estimator.

REFERENCES

[AP23] D. Agrawal and D. Panagou. "Safe and Robust Observer-Controller Synthesis Using Control Barrier Functions." *IEEE Control Systems Letters*, **7**:127–132, 2023.

[Bac11] F. Bach. "Shaping level sets with submodular functions." In *Advances in Neural Information Processing Systems*, pp. 10–18, 2011.

[Bac13] F. Bach. "Learning with submodular functions: A convex optimization perspective." *Foundations and Trends® in Machine Learning*, **6**(2-3):145–373, 2013.

[Bac19] F. Bach. "Submodular functions: from discrete to continuous domains." *Mathematical Programming*, **175**(1-2):419–459, 2019.

[BD97] R. Bro and S. De Jong. "A fast non-negativity-constrained least squares algorithm." *Journal of Chemometrics*, **11**(5):393–401, 1997.

[BEN09] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust optimization*, volume 28. Princeton University Press, 2009.

[Ber19] P. Bernard. *Observer design for nonlinear systems*, volume 479. Springer, 2019.

[BJM12] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. "Structured sparsity through convex optimization." *Statistical Science*, **27**(4):450–468, 2012.

[BL06] J. Borwein and A. Lewis. *Convex Analysis and Nonlinear Optimization : Theory and Examples*. Number 2 in CMS Books in Mathematics. Springer-Verlag New York, 2006.

[BLK17] A. Bian, K. Levy, A. Krause, and J.M. Buhmann. "Non-monotone Continuous DR-submodular Maximization: Structure and Algorithms." *CoRR*, **abs/1711.02515**, 2017.

[BMB16] A.A. Bian, B. Mirzasoleiman, J.M. Buhmann, and A. Krause. "Guaranteed Non-convex Optimization: Submodular Maximization over Continuous Domains." *CoRR*, **abs/1606.05615**, 2016.

[BT04] J.P. Berrut and L.N. Trefethen. "Barycentric Lagrange Interpolation." *SIAM Review*, **46**(3):501–517, 2004.

[CSG12] D. Chen, S.L. Sain, and K. Guo. "Data mining for the online retail industry: A case study of RFM model-based customer segmentation using data mining." *Journal of Database Marketing & Customer Strategy Management*, **19**:197–208, 2012.

[CST21] R. Cosner, A. Singletary, A. Taylor, T. Molnar, K. Bouman, and A. Ames. "Measurement-Robust Control Barrier Functions: Certainty in Safety with Uncertainty in State." In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 6286–6291, 2021.

[CT05] E. J. Candes and T. Tao. "Decoding by linear programming." *IEEE Transactions on Information Theory*, **51**(12):4203–4215, 2005.

[DFG01] S. Diop, V. Fromion, and J. W. Grizzle. "A resettable Kalman filter based on numerical differentiation." In *2001 European Control Conference*, pp. 1239–1244, 2001.

[DG17] D. Dua and C. Graff. "UCI Machine Learning Repository.", 2017.

[Dio07] S. Diop. "Observers for sampled data nonlinear systems via numerical differentiation." In *2007 European Control Conference*, pp. 1179–1184, 2007.

[DK18] A. Das and D. Kempe. "Approximate submodularity and its applications: Subset selection, sparse approximation and dictionary selection." *The Journal of Machine Learning Research*, **19**(1):74–107, 2018.

[DN15] S. Dashkoviskiy and L. Naujok. "Quasi-ISS/ISDS observers for interconnected systems and applications." *Systems & Control Letters*, **77**:11–21, 2015.

[DP02] B.A. Davey and H.A. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, 2002.

[EJ20] M. El Halabi and S. Jegelka. "Optimal approximation for unconstrained non-submodular minimization." *International Conference on Machine Learning (ICML)*, 2020.

[EKD18] E.R. Elenberg, R. Khanna, A.G. Dimakis, and S. Negahban. "Restricted strong convexity implies weak submodularity." *The Annals of Statistics*, **46**(6B):3539–3568, 2018.

[FI11] S. Fujishige and S. Isotani. "A submodular function minimization algorithm based on the minimum-norm base." *Pacific Journal of Optimization*, **7**(1):3–17, 2011.

[Fre95] R. Freeman. "Global internal stabilizability does not imply global external stabilizability for small sensor disturbances." *IEEE Transactions on Automatic Control*, **40**(12):2119–2122, 1995.

[Fre96] P. Freeman, R. Kokotović. *Robust Nonlinear Control Design: State Space and Lyapunov Techniques*. Modern Birkhäuser Classics. Birkhäuser, Boston, MA, 1 edition, 1996.

[GM90] J.W. Grizzle and P.E. Moraal. "Newton, observers and nonlinear discrete-time control." In *29th IEEE Conference on Decision and Control*, pp. 760–767 vol.2, 1990.

[IF16] S. Ito and R. Fujimaki. "Large-Scale Price Optimization via Network Flow." In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[IF17] S. Ito and R. Fujimaki. "Optimization Beyond Prediction: Prescriptive Price Optimization." In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, p. 1833–1841, New York, NY, USA, 2017. Association for Computing Machinery.

[IJB13] R. Iyer, S. Jegelka, and J. Bilmes. "Fast semidifferential-based submodular function optimization: Extended version." In *International Conference on Machine Learning (ICML)*, 2013.

[KGG06] A. Krause, C. Guestrin, A. Gupta, and J. Kleinberg. "Near-optimal sensor placements: Maximizing information while minimizing communication cost." In *International Conference on Information Processing in Sensor Networks*, pp. 2–10, 2006.

[Kha15] H.K. Khalil. *Nonlinear Control*. Pearson Education, 1 edition, 2015.

[Kir11] Andreas Kirsch et al. *An introduction to the mathematical theory of inverse problems*, volume 120. Springer, 2011.

[KK03] S. Kim and M. Kojima. "Exact Solutions of Some Nonconvex Quadratic Optimization Problems via SDP and SOCP Relaxations." *Computational Optimization and Applications*, **26**(2):143–154, 2003.

[Kra10] A. Krause. "SFO: A toolbox for submodular function optimization." *Journal of Machine Learning Research (JMLR)*, **11**(Mar):1141–1144, 2010.

[KSW01] M. Krichman, E. D. Sontag, and Y. Wang. "Input-Output-to-State Stability." *SIAM Journal on Control and Optimization*, **39**(6):1874–1928, 2001.

[KY22] M. Khajenejad and S.Z. Yong. "Hinf-Optimal Interval Observer Synthesis for Uncertain Nonlinear Dynamical Systems via Mixed-Monotone Decompositions." *IEEE Control Systems Letters*, **6**:3008–3013, 2022.

[LB11] H. Lin and J. Bilmes. "A Class of Submodular Functions for Document Summarization." In *Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pp. 510–520, USA, 2011. Association for Computational Linguistics.

[Lov83] L. Lovász. "Submodular functions and convexity." In *Mathematical Programming The State of the Art*, pp. 235–257. Springer, 1983.

[NKA11] K. Nagano, Y. Kawahara, and K. Aihara. "Size-Constrained Submodular Minimization through Minimum Norm Base." In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, p. 977–984, Madison, WI, USA, 2011. Omnipress.

[NWF78] G.L. Nemhauser, L.A. Wolsey, and M.L. Fisher. "An analysis of approximations for maximizing submodular set functions—I." *Mathematical programming*, **14**(1):265–294, 1978.

[QZT19] A. Qian, S. Zhu, J. Tang, R. Jin, B. Sun, and H. Li. "Robust optimization over multiple domains." In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 4739–4746, 2019.

[Rau10] H. Rauhut. "Compressive sensing and structured random matrices." *Theoretical Foundations and Numerical Methods for Sparse Recovery*, **9**:1–92, 2010.

[Sch03] A. Schrijver. *Combinatorial optimization: polyhedra and efficiency*, volume 24. Springer Science & Business Media, 2003.

[SG64] A. Savitzky and M. J. E. Golay. "Smoothing and Differentiation of Data by Simplified Least Squares Procedures." *Analytical Chemistry*, **36**(8):1627–1639, 1964.

[SJ19] M. Staib and S. Jegelka. "Robust Budget Allocation Via Continuous Submodular Functions." *Applied Mathematics & Optimization*, 2019.

[SL16] H. Shim and D. Liberzon. "Nonlinear Observers Robust to Measurement Disturbances in an ISS Sense." *IEEE Transactions on Automatic Control*, **61**(1):48–61, 2016.

[ST99] J.S. Shamma and K.Y. Tu. "Set-valued observers and optimal disturbance rejection." *IEEE Transactions on Automatic Control*, **44**(2):253–264, 1999.

[Top98] D.M. Topkis. *Supermodularity and complementarity*. Princeton University Press, 1998.

[ZY06] P. Zhao and B. Yu. "On Model Selection Consistency of Lasso." *Journal of Machine Learning Research (JMLR)*, **7**(Nov):2541–2563, 2006.