OPTIMAL ENERGY STORAGE STRATEGIES IN MICROGRIDS

by

Arnab Bhattacharya

B. Tech, Indian Institute of Technology, Kharagpur, 2011

M. Tech, Indian Institute of Technology, Kharagpur, 2012

Submitted to the Graduate Faculty of
the Swanson School of Engineering in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2017

UNIVERSITY OF PITTSBURGH

SWANSON SCHOOL OF ENGINEERING

This dissertation was presented

by

Arnab Bhattacharya

It was defended on

October 9, 2017

and approved by

Dr. Jeffrey P. Kharoufeh, Professor, Department of Industrial Engineering

Dr. Daniel R. Jiang, Assistant Professor, Department of Industrial Engineering

Dr. Bo Zeng, Assistant Professor, Department of Industrial Engineering

Dr. Zhi-Hong Mao, Associate Professor, Department of Electrical and Computer Engineering

Dissertation Director: Dr. Jeffrey P. Kharoufeh, Professor, Department of Industrial Engineering

OPTIMAL ENERGY STORAGE STRATEGIES IN MICROGRIDS

Arnab Bhattacharya, Ph.D.

University of Pittsburgh, 2017

Microgrids are small-scale distribution networks that provide a template for large-scale deployment of renewable energy sources, such as wind and solar power, in close proximity to demand. However, the inherent variability and intermittency of these sources can have a significant impact on power generation and scheduling decisions. Distributed energy resources, such as energy storage systems, can be used to decouple the times of energy consumption and generation, thereby enabling microgrid operators to improve scheduling decisions and exploit arbitrage opportunities in energy markets. The integration of renewable energy sources into the nation's power grid, by way of microgrids, holds great promise for sustainable energy production and delivery; however, operators and consumers both lack effective strategies for optimally using stored energy that is generated by renewable energy sources.

This dissertation presents a comprehensive stochastic optimization framework to prescribe optimal strategies for effectively managing stored energy in microgrids, subject to the inherent uncertainty of renewable resources, local demand and electricity prices. First, a Markov decision process model is created to characterize and illustrate structural properties of an optimal storage strategy and to assess the economic value of sharing stored energy between heterogeneous, demand-side entities. Second, a multistage stochastic programming (MSP) model is formulated and solved to determine the optimal storage, procurement, selling and energy flow decisions in a microgrid, subject to storage inefficiencies, distribution line losses and line capacity constraints. Additionally, the well-known stochastic dual dynamic programming (SDDP) algorithm is customized and improved to drastically reduce the computation time and significantly improve solution quality when approximately solving this MSP model. Finally, and more generally, a novel nonconvex regularization scheme is developed to improve the computational performance of the SDDP algorithm for solving high-dimensional MSP models. Specifically, it is shown that these nonconvex regularization problems can be reformulated as mixed-integer programming problems with provable convergence guarantees. The benefits of this regularization scheme are illustrated by way of a computational study that reveals significant improvements in the convergence rate and solution quality over the standard SDDP algorithm and other regularization schemes.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

I am indebted to so many individuals who have positively influenced my intellectual, professional, and personal growth during my time at Pitt. First, and foremost, I would like to sincerely thank Dr. Jeff Kharoufeh for being an exceptional advisor during the past four and a half years. Dr. Kharoufeh's attention to detail is unparalleled, and his commitment to academic excellence and integrity is infectious. I am indebted to him for providing me with an exciting research topic very early in my Ph.D. studies and for inspiring me to pursue excellence, even during the most challenging phases of graduate student life. I am especially grateful for his commitment to my professional development by providing funding for my research, tirelessly honing my writing skills, providing opportunities for me to participate in professional conferences, and motivating me to apply for prestigious awards. A special token of appreciation is due to his wonderful family, who always made me feel welcomed at their home for delicious Thanksgiving and Christmas dinners.

Dr. Bo Zeng has effectively served as a second advisor to me. Dr. Zeng's technical suggestions and insightful feedback significantly improved the methodological contributions of my dissertation. He always provided answers to my challenging optimization questions, which made my life so much easier. Dr. Zeng is a wonderful role model for maintaining work-life balance in academia and was kind enough to offer me wide-ranging advice to survive the grind of graduate school. Special thanks are also due to Dr. Daniel Jiang for his invaluable suggestions that improved many of the technical results in Chapters 2 and 4, and to Dr. Zhi-Hong Mao, of the Department of Electrical and Computer Engineering, for his constructive comments on several realistic aspects of the microgrid problem. This research would not have been possible without the support of three seed grants from the Mascaro Center of Sustainable Innovation at the University of Pittsburgh.

I am also grateful to the outstanding trio of David Abdul-Malak, Juan Borrero, and Ruichen Sun for being my best pals in the department. I will always remember our enlightening discussions over lunch about research and every other topic imaginable; thanks for your excellent insights and constructive suggestions. I thoroughly enjoyed my collaboration with Ashley Anhalt on a number of challenging healthcare and energy problems. Mohammad Hossein, Moataz Abdulhafez, Shadi Sanoubar, Ibrahim El Sharr, and Tarik Bilgic were also wonderful departmental colleagues. Outside of Pitt IE, I would like to thank Manaal Faruqui, Piyush Panigrahi, Amit Behera, and Gaurav Agarwal for their well wishes and support amidst the vagaries of life as a doctoral student.

Finally, I am eternally grateful to my parents, Leena and Kallol, and my younger brother, Aritra, for their unwavering love, support, and trust in me. I am forever indebted to my parents for teaching me the importance of hard work and dedication since I was a child, and for helping me achieve my dreams at the expense of their own. I know that my family, more than anyone else, is happy to see me earn a Ph.D.

1.0 INTRODUCTION

1.1 BACKGROUND AND MOTIVATION

Global population growth and the proliferation of electronic household appliances and devices are driving unprecedented demand for electricity in the United States (U.S.) and abroad. Despite energy efficiency improvements and targeted efforts to reduce residential energy consumption, the U.S. Energy Information Administration (EIA) estimates a 24% increase in U.S. residential electricity demand by 2040 and a 13% price increase over the same period [1]. Aging infrastructure, the risk of blackouts, security concerns, and an aversion to carbon-based fuel sources have placed the nation's electric power grid under enormous pressure to produce reliable, efficient, and sustainable energy. But the demand for energy has far outpaced investment in critical infrastructure, resulting in dire consequences. For instance, the infamous Northeast blackout of August 14, 2003 affected roughly 50 million people and cost an estimated 6 billion dollars [46]. Moreover, capacity constraints and significant power losses over transmission lines (15 to 20 percent) have contributed to systemic inefficiencies and excessive operational costs in the distribution network [2, 144].

These concerns, and others, have spawned immense national interest in so-called *smart grid* initiatives that promote wholesale modernization of the nation's electric power grid. In fact, the U.S. Congress passed the landmark Energy Independence and Security Act of 2007 [142], advocating the establishment of a grid that will usher in a new era of technological sophistication and efficiency. Central to the grid paradigm is the integration of renewable energy sources (RES), such as wind and solar power, in close proximity to consumers. The integration of RES is now mandated by the U.S. government [142, 159], and the penetration of renewables in the overall energy production portfolio is forecasted to reach at least 22%

by 2030 [34]. However, before these ambitious goals can be realized, several technological and operational challenges related to integration must be overcome.

Microgrids have emerged as a potential solution to integrating renewables locally using distributed generation (DG) and energy storage systems (ESS). A microgrid is a small-scale version of a centralized power grid that generates, distributes, and regulates electricity flow to local consumers using distributed generation [27, 65, 91, 96]. Microgrids aim to integrate grid equipment, metering technologies, devices, and software to improve local power distribution. Moreover, they provide a template for future large-scale deployment of RES using DG and energy storage located near consumer demand [8, 91, 108, 144]. Indeed, it is estimated that total microgrid generation capacity in North America will expand to 5.9 Gigawatts (GW), or roughly 64% of global capacity, by the year 2020 [9]. Figure 1 depicts the main components of a microgrid connected to the main grid.

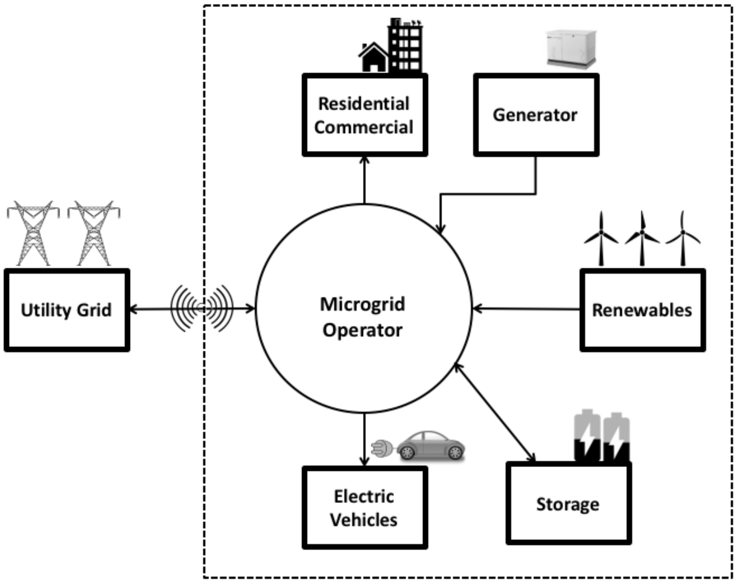


Figure 1: A microgrid with distributed generation and energy storage.

Typically, a consumer in a microgrid has access to local RES that are used to satisfy his demand requirements. Any additional demand requirements are met by procuring electricity at real-time prices from the main grid. However, sophisticated bidirectional technologies in microgrids will engender a new class of consumers, called the *prosumers*, who can actively procure, as well as, sell electricity to real-time or day-ahead energy markets to minimize their overall electricity costs [60, 61]. Further, such bidirectional technologies will allow prosumers to dynamically adjust their demand in response to time-varying, real-time electricity prices, and thereby, improve the overall utilization of RES in a microgrid. Moreover, prosumers with higher demand requirements will have on-site wind and/or solar installations to offset costs due to distribution losses in a microgrid. This will create a network of self-sustaining prosumers that will be primarily powered by renewable sources in a microgrid.

However, deeper penetration of renewables in the overall energy portfolio poses a new set of challenges. Renewable energy sources are intermittent due to inherent variability in the source itself, and because future supplies are difficult to predict. Moreover, because renewable sources, such as wind and power, are not dispatchable, i.e., their power output cannot be directly controlled by operators within a microgrid, their volatility can dramatically affect production scheduling decisions [105]. Further, DG sources in microgrids have limited generating capacity, typically in the range 10-1000 kilowatts (kW), making them susceptible to potential inefficiencies like real-time power shortages and power reliability issues. Due to the intermittent and variable nature of renewable energy sources, distributed energy storage systems, such as batteries and flywheels, can be used to shift local energy consumption from high demand to low demand periods [141, 165]. Although the main grid is more reliable and predictable, distributed storage offers a substantially less expensive alternative, especially during peak demand periods when prices are also typically high. The deregulation of energy markets and the advent of bidirectional communication technologies in microgrids [108, 144, 145] now allows consumers to exploit time-of-use and dynamic pricing rates in the electricity markets by storing energy during off-peak periods for use during peak demand periods [48, 80, 154]. This temporal shifting of demand has the potential to reduce consumer electricity costs and curtail the dependence on polluting ancillary generating units (e.g., diesel generators) in a microgrid [82]. While the integration of RES into microgrids holds great promise for sustainable energy creation and delivery, there is a critical need to devise optimal strategies for demand-side entities using energy storage.

Battery systems have emerged as one of the leading distributed storage technologies because of their long shelf lives, low maintenance costs, and high energy-density values [75, 108, 148]. However, major technological and economical barriers in batteries pose a major challenge to their integration in microgrids. A key drawback is that batteries are highly inefficient – nearly 20 to 40 percent of the total energy is lost during one complete charge-discharge cycle of a battery [47, 48]. Moreover, battery electrolytes have high internal resistances that cause continuous dissipation of the stored electrical energy as heat [48, 148]. The per-unit cost of battery capacity is typically high, which makes them economically unattractive for large storage installations [79, 103]. Additionally, each battery type has its own set of constraints related to their charging and discharging regimes. For example, nickel-cadmium batteries have to be completely discharged before charging, while lead-acid batteries should never be fully discharged [113]. The aforementioned budgetary and technological constraints of a battery make it significantly hard to derive an optimal storage policy for a demand-side entity in a microgrid.

1.2 PROBLEM STATEMENT AND RESEARCH OBJECTIVES

This dissertation addresses the problems related to reducing the total electricity costs of consumers in a microgrid with access to distributed storage and renewable energy sources. The aim is to develop a comprehensive stochastic optimization framework to prescribe optimal strategies for effectively managing stored energy in microgrids under uncertainty of renewable generation, local demand, and electricity prices. The primary research objectives are as follows:

1. To formulate and solve multistage stochastic optimization models that prescribe optimal storage, procurement, and selling strategies in a microgrid under randomly evolving renewable supply, demand, and prices;
2. To characterize structural properties of the optimal storage strategy in a microgrid, and assess the economic value of sharing stored energy under storage capacity and power flow restrictions;

3. To develop scalable approaches to solve large-scale, high-dimensional multistage stochastic optimization models arising in energy via mathematical programming techniques.

1.3 DISSERTATION OUTLINE AND CONTRIBUTIONS

Chapter 2 presents the problem of dynamically controlling a 2-bus energy distribution network with energy storage capabilities. A microgrid operator seeks to dynamically adjust the amount of energy to charge to, or discharge from, energy storage devices in response to randomly-evolving renewable supply, demand and prices. The objective is to minimize the expected total discounted costs incurred within the network over a finite planning horizon. Formulated is a Markov decision process model that prescribes the optimal amount of energy to charge or discharge, and transmit between the two buses during each stage of the planning horizon. Established are the multimodularity of the value function, and the monotonicity of the optimal policy in the storage levels. It is also shown that the optimal operational cost is convex and monotone in the storage capacities. Furthermore, we establish bounds on the optimal cost by analyzing comparable single-storage systems with pooled and decentralized storage configurations, respectively. These results extend to more general multi-bus network topologies. Numerical examples illustrate the main results and highlight the significance of interacting demand-side entities.

Chapter 3 introduces a multistage stochastic programming model whose objective is to minimize the expected total energy costs incurred within a microgrid over a finite planning horizon. The model prescribes the amount of energy to procure, store and discharge in each decision stage of the horizon. However, for even a moderate number of stages, the model is computationally intractable; therefore, we customize the stochastic dual dynamic programming (SDDP) algorithm to obtain high-quality approximate solutions. Computation times and optimization gaps are significantly reduced by implementing a dynamic cut selection procedure and a lower bound improvement scheme within the SDDP framework. An extensive computational study reveals significant cost savings as compared to myopic and non-storage policies, as well as policies obtained using a two-stage SP model.

Chapter 4 considers a novel regularization scheme to improve the computational performance of the stochastic dual dynamic programming (SDDP) algorithm for solving multistage stochastic linear programs. Specifically, we employed a class of nonconvex regularization functions, called the folded-concave penalty, to enhance the quality of outer approximations obtained via SDDP. A mixed-integer linear programming (MILP) strategy is used to solve the nonconvex regularization problem to global optimality. The proposed regularization scheme is numerically stable and facilitates the use of state-of-the-art optimization MILP solvers within the SDDP framework. We establish provable convergence guarantees of our regularized SDDP algorithm under mild regularity conditions. Furthermore, we empirically demonstrate the potential benefits of our regularization scheme for two large-scale stochastic optimization problems that arise in energy and finance. Our results reveal significant improvements in the convergence rate and solution quality of SDDP, especially for high-dimensional problems.

2.0 STRUCTURED STORAGE POLICIES FOR ENERGY DISTRIBUTION NETWORKS

This chapter considers the problem of dynamically controlling a 2-bus energy distribution network with energy storage capabilities. A microgrid operator seeks to dynamically adjust the amount of energy to charge to, or discharge from, energy storage devices in response to randomly-evolving demand, renewable supply and prices. The objective is to minimize the expected total discounted costs incurred within the network over a finite planning horizon. Formulated is a Markov decision process (MDP) model that prescribes the optimal amount of energy to charge or discharge, and transmit between the two buses during each stage of the planning horizon. Established are the multimodularity of the value function, and the monotonicity of the optimal policy in the energy storage levels. It is also shown that the optimal operational cost is convex and monotone in the storage capacities. Furthermore, bounds on the optimal cost are established by analyzing comparable single-storage systems with pooled and decentralized storage configurations, respectively. These results extend to more general multi-bus network topologies. Numerical examples illustrate the main results and highlight the significance of interacting demand-side entities.

2.1 SUMMARY OF RELEVANT LITERATURE

Energy storage, as a means by which to integrate renewable sources in the power grid, has spawned significant interest in the energy systems modeling literature. The storage problem bears some resemblance to classical inventory and asset management problems (cf. [43, 171]), except that the operator is faced with multiple sources of uncertainty, storage and line inefficiencies, as well as network energy balance constraints. Most of the relevant work in this area focuses on devising an optimal storage policy for a single consumer (or supplier) with access to renewable energy and finite-capacity storage. A linear programming approach was employed to solve the consumer's storage problem under deterministic price, demand and renewable supply levels in [3, 74]. Bar-Noy et al. [13, 14] developed efficient online algorithms to reduce a consumer's peak demand costs by optimally procuring and storing energy when demand is uncertain. More recently, MDP models have been used extensively to analyze the single consumer problem under exogenous uncertainty. Using an infinite-horizon MDP model, Van de Ven et al. [153] proved the existence of an optimal dual-threshold storage policy for a consumer with uncertain demands, subject to deterministic time-of-use electricity prices. Harsha and Dahleh [64] derived a similar dual-threshold optimal storage policy for a finite-horizon problem under uncertain prices, demand and renewable supply. Furthermore, they analytically characterize a consumer's optimal storage capacity for the case when prices are fixed. Similar single-storage MDP models have been employed for a supplier's storage management problem, which involves optimizing the bidding strategies of renewable suppliers that participate in day-ahead or real-time energy markets to maximize profits by deploying energy storage (cf. [22, 26, 57, 84, 85, 86, 107, 131, 167]). However, single-storage models do not account for network constraints and interactions between different network entities with storage, rendering them unrealistic for our setting. Alternatively, stochastic programming (SP) models have been devised to solve network storage problems with continuous actions and high-dimensional state spaces. Some representative examples of such models include [17, 57, 92, 110, 160]. Although SP models allow for the incorporation of network constraints, the number of possible scenarios in these models can be prohibitively large. Additionally, solutions to SP models can be difficult to interpret, as they provide little insight into the structure of the optimal policy.

The model presented here is distinguished from existing single-storage models in that it considers the perspective of multiple demand-side entities, each with energy storage capabilities, in a distribution network. Specifically, we first examine a 2-bus network model in which decisions are made under randomly-evolving demand, renewable supply and real-time electricity prices. This model captures the salient features of distributed energy storage operations by considering the impact of renewable generation, storage inefficiencies, supply-demand imbalances, distribution energy losses, and constrained power-line capacities on the optimal storage decisions. We focus first on the 2-bus system in light of the fact that network reduction methods can be used to analyze more complicated multi-bus networks as equivalent 2-bus networks for power system planning and operational problems [115, 124, 158]. A unique feature of our model is the fact that the buses can transmit energy to one another – a feature that is shown to significantly impact the optimal decisions and operational costs. Our main results can be summarized as follows. First, we establish the monotonicity and convexity of our MDP model's value function in the storage levels for each fixed exogenous state. Next, we prove that the value function is multimodular in the storage levels, the optimal policy is monotone and the optimal storage decisions in each stage exhibit bounded sensitivities. We also establish bounds that compare the cost of the 2-bus network to those of two comparable systems with pooled and decentralized storage configurations, respectively, and the main results are extended to more general multi-bus network topologies. To illustrate the structural properties, we present numerical examples that use real renewable generation and pricing data obtained from open sources. These examples help quantify the benefits of using the network model in lieu of simpler, single-storage models that fail to account for interactions between demand-side entities in a distribution network.

The remainder of the chapter is organized as follows. The next section describes the 2-bus distribution network and introduces notation and nomenclature of the mathematical model. In Section 2.3, we present the main results, which include structural properties of the value function, optimal policy and the optimal operational cost. Section 2.5 provides numerical examples that illustrate the main structural results and highlight the importance of interactions between network entities.

2.2 MARKOV DECISION PROCESS MODEL FOR A 2-BUS NETWORK

Consider a 2-bus network connected to the main grid through a reference bus (or *feeder*) as depicted in Figure 2. The feeder is not connected to any distributed energy storage system or renewable energy sources; however, the other two buses (the *load buses*) are connected to finite-capacity storage systems and renewable generators that satisfy the demand realized at these buses. Any unmet demand can be satisfied by procuring energy from the grid and/or by receiving energy transmitted from the other bus. Similarly, any surplus energy generated at a load bus can be sold to the main grid and/or transmitted to the other load bus. However, energy flow between the buses is constrained by the capacity of the power line (hereafter the *line*) connecting them, as well as supply-demand balance constraints at each bus. Additionally, storage capacity limitations restrict the amount of energy that can be charged to, or discharged from, the storage devices. We assume that a central network operator (or controller) is responsible for all energy flow and storage decisions within the distribution network.

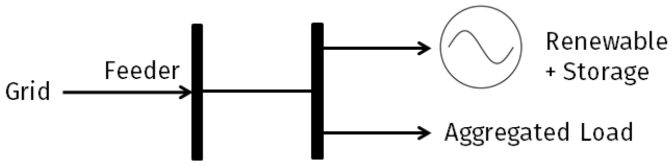


Figure 2: Graphical depiction of a 2-bus distribution network.

The distribution network incurs three types of costs: (i) the explicit cost of procuring energy from, or selling energy to, the grid at real-time prices; (ii) the implicit cost of lost energy due to line losses stemming from resistive overheating [6]; and (iii) costs associated with storage inefficiencies. While transmitting energy between the load buses helps to offset the cost of procuring energy from the grid, only a limited amount of energy can be transmitted due to a line capacity constraint between the buses. Moreover, transmitting stored energy to another bus is a lost opportunity to procure and store surplus energy from the grid for future use when prices are high. Therefore, an obvious tradeoff exists between the amount of energy to buy or sell, and the amount that is transmitted between the load buses. The operator's objective is to minimize the expected total discounted costs incurred over a finite planning horizon by making a sequence of operational decisions. For each stage of the planning horizon, the operator must decide the amount of energy to: (*i*) buy from, or sell to, the main grid; (*ii*) charge to, or discharge from, the energy storage devices; and (*iii*) transmit between the two load buses. These decisions are made under randomly-varying demand, renewable generation and real-time prices.

We formulate the operator's sequential decision problem using a finite-horizon Markov decision process (MDP) model. Specifically, consider a planning horizon of length Υ and partition the time interval $[0, \Upsilon)$ so that

$$[0, \Upsilon) = \bigcup_{t=1}^{N} [\varepsilon_{t-1}, \varepsilon_t),$$

where N is the number of time intervals (or stages) and ε_t is the tth decision epoch with $\varepsilon_0 \equiv 0$ and $\varepsilon_N \equiv \Upsilon$. The discrete time horizon is denoted by $T = \{1, 2, \ldots, N\}$, where $t \in T$ is the index of the tth stage, namely the interval $[\varepsilon_{t-1}, \varepsilon_t)$. It is assumed that no decisions are made at stage N. For future use, let $T' \equiv T \setminus \{N\}$. Let $C = \{0, 1, 2\}$ be the set of buses in the network, where bus 0 is the feeder, and bus $i \in C' \equiv \{1, 2\}$ denotes the ith load bus. The set of all lines in the network is denoted by $A = \{(i, j) : i, j \in C\}$, where (i, j) is the line connecting bus i to bus j.

The physical parameters of the network are described as follows. Let α_i ($\alpha_i < \infty$) denote the capacity of the storage device located at bus $i \in C'$. The parameters ρ_c^i and ρ_d^i denote the charging and discharging efficiencies of the storage device at bus i, where $\rho_c^i, \rho_d^i \in (0, 1]$. The round-trip efficiency of the storage device at bus i is defined as $\rho_i \equiv \rho_c^i \rho_d^i$. The quantities τ_c^i and τ_d^i denote the maximum charging and discharging rates of the storage device at bus i, respectively. Gather the parameters α_i, τ_c^i and τ_d^i in the vectors $\boldsymbol{\alpha}, \boldsymbol{\tau}_c$ and $\boldsymbol{\tau}_d$, respectively. Let β be the capacity of the line connecting the load buses. Finally, let ν denote the per-unit cost of line losses, while φ is the per-unit cost of charging energy to, or discharging energy from, the storage devices.

The model contains several sources of uncertainty that we now describe in detail. All random variables are defined on a common and complete probability space $(\Omega, \mathscr{A}, \mathbb{P})$ with natural filtration $\{\mathscr{A}_t : t \in \mathcal{T}\}$, i.e., $\mathscr{A}_t$ contains the information available up to stage t. Any random quantity with subscript t is assumed to be $\mathscr{A}_t$-measurable. Let D_t^i denote the random net demand (demand minus renewable supply) at bus $i \in C'$ with countable support $\mathcal{D}_t^i \subset \mathbb{R}$, and let P_t be the random real-time price at the start of stage t with countable support $\mathcal{P}_t \subset \mathbb{R}_+$. Let $\boldsymbol{W}_t = (P_t, D_t^1, D_t^2)$ denote the *exogenous* information available at the start of stage t, and let $\mathcal{W}_t \equiv \mathcal{P}_t \times \mathcal{D}_t^1 \times \mathcal{D}_t^2$ be the support of $\boldsymbol{W}_t$. A realization of $\boldsymbol{W}_t$ is denoted by $\boldsymbol{w}_t \equiv (p_t, d_t^1, d_t^2)$, where p_t, d_t^1 and d_t^2 are realizations of P_t, D_t^1 and D_t^2, respectively. This information is exogenous in the sense that the evolution of $\boldsymbol{W}_t$ is independent of the operator's decisions over the planning horizon. The set of all sample paths of $\boldsymbol{W} = \{\boldsymbol{W}_t : t \in T\}$ is denoted by $\mathcal{W} \equiv \mathcal{W}_1 \times \cdots \times \mathcal{W}_N$, and it is assumed that $\boldsymbol{W}$ possesses the Markov property, i.e., for any $t \in T$,

$$\mathbb{P}(\boldsymbol{W}_t = \boldsymbol{w}_t | \boldsymbol{W}_{t-1}, \ldots, \boldsymbol{W}_1; \mathscr{A}_{t-1}) = \mathbb{P}(\boldsymbol{W}_t = \boldsymbol{w}_t | \boldsymbol{W}_{t-1}; \mathscr{A}_{t-1}), \quad \boldsymbol{w}_t \in \mathcal{W}_t.$$

At the start of stage t, let the (random) storage level at bus i be denoted by Y_t^i, define $\boldsymbol{Y}_t = (Y_t^1, Y_t^2)$ and let $\mathcal{Y} \equiv [0, \alpha_1] \times [0, \alpha_2]$ be the set of all possible storage levels. Note that $\mathcal{Y}$ is time-invariant, as the storage capacities α_1 and α_2 are fixed *a priori*. In contrast to the exogenous variables, the endogenous component $\boldsymbol{Y}_t$ is influenced by the operator's actions up to stage $t-1$. The random state of the process at the start of stage t is a vector $\boldsymbol{S}_t = (\boldsymbol{W}_t, \boldsymbol{Y}_t)$ whose state space is $\mathcal{S}_t \equiv \mathcal{W}_t \times \mathcal{Y}$. A realization of $\boldsymbol{S}_t$ is denoted by $\boldsymbol{s}_t = (\boldsymbol{w}_t, \boldsymbol{y}_t)$ for $\boldsymbol{w}_t \in \mathcal{W}_t$ and $\boldsymbol{y}_t \in \mathcal{Y}$, and we assume that the initial state, $\boldsymbol{S}_1$, is known with certainty.

The decision process evolves as follows. At the start of each stage, the operator observes the exogenous state and the current storage levels at the load buses. Then, the operator makes the operational decisions to procure or sell, to charge or discharge, and whether to transmit energy between the load buses. The operator makes no decisions in the final stage and incurs a terminal cost. It is noted that all of the decisions are made simultaneously because, unlike other commodities, energy cannot be backlogged and needs to be consumed immediately. Let $\boldsymbol{x}_t(\boldsymbol{s}_t)$ be the decision vector at the start of stage t when state $\boldsymbol{s}_t$ is realized; henceforth, the dependence of $\boldsymbol{x}_t$ on $\boldsymbol{s}_t$ is suppressed for notational brevity. The decision vector assumes the form $\boldsymbol{x}_t = (\boldsymbol{u}_t, q_t)$, where the vector $\boldsymbol{u}_t$ contains the charge/discharge decisions at each bus, and q_t is the amount of energy to transmit between the buses. Note that the buy/sell decisions are not explicitly included in the decision vector, as these decisions are auxiliary to the charge/discharge decisions. These quantities are further elucidated in what follows, along with the feasibility set of $\boldsymbol{x}_t$.

Let $\boldsymbol{u}_t = (u_t^i : i \in C')$ be the vector of charging/discharging decisions at stage t described as follows. For each stage t: (i) if $u_t^i > 0$, then u_t^i units of energy are charged to the storage device at bus i; (ii) if $u_t^i < 0$, then $-u_t^i$ units of energy are discharged from the storage device at bus i; and (iii) if $u_t^i = 0$, then energy is neither charged to, nor discharged from, the storage device at bus i. The charging/discharging decisions are constrained by the storage capacities and the charging/discharging rates of the storage devices. That is,

$$-\min\{\boldsymbol{y}_t, \boldsymbol{\tau}_d\} \leq \boldsymbol{u}_t \leq \min\{\boldsymbol{\alpha} - \boldsymbol{y}_t, \boldsymbol{\tau}_c\}, \quad t \in T', \tag{2.1}$$

where all inequalities involving vectors are understood to hold component-wise.

The energy flow between the buses at stage t, namely q_t, is described as follows: if $q_t > 0$, then q_t units of energy flow from bus 1 to bus 2; if $q_t < 0$, then $-q_t$ units of energy flow from bus 2 to bus 1; and if $q_t = 0$, then no energy flows between the two buses. These variables are constrained by the line capacity via

$$-\beta \leq q_t \leq \beta, \quad t \in T'. \tag{2.2}$$

Similarly, let g_t^1 and g_t^2 be the energy flow in the lines connecting the feeder to buses 1 and 2 (the *feeder lines*), respectively. Here, if $g_t^1, g_t^2 > 0$, then energy flows from the feeder to the load buses, and if $g_t^1, g_t^2 < 0$, energy flows from the buses to the feeder. If these quantities are zero, then no energy is bought from, or sold to, the grid. For each $t \in T'$ and $i \in C'$, define the variables θ_t^i as follows:

$$\theta_t^i = \begin{cases} 1/\rho_c^i, & u_t^i \geq 0, \\ \rho_d^i, & u_t^i < 0. \end{cases}$$

Then the supply-demand balance equations at the load buses are

$$g_t^1 = d_t^1 + \theta_t^1 u_t^1 + q_t, \quad t \in T', \tag{2.3}$$

$$g_t^2 = d_t^2 + \theta_t^2 u_t^2 - q_t, \quad t \in T'. \tag{2.4}$$

As noted earlier, g_t^1 and g_t^2 are auxiliary variables that depend on $(\boldsymbol{u}_t, q_t)$ and represent the buying/selling decisions. Because the feeder lines serve as the main connection between the distribution system and the main grid, they typically possess sufficient, reliable capacity and incur only minor line losses. Therefore, we assume that energy flows in the feeder lines are not restricted by line capacities and do not incur line losses. Next, define the stage t feasibility set (or action space) by $\mathcal{X}_t(\boldsymbol{y}_t)$, which is characterized by the linear constraints (2.1) and (3.11); note that $\mathcal{X}_t(\boldsymbol{y}_t)$ is a bounded polyhedron. A feasible policy is a vector $\boldsymbol{\pi} = (\boldsymbol{x}_t : t \in T') \in \Pi$, where Π denotes the set of all feasible Markov deterministic (MD) policies.

Following the actions taken by the operator in the current stage, the process next transitions randomly to another state in the next stage. The endogenous storage levels evolve as a deterministic function of the current storage levels and the charge/discharge decisions via

$$\boldsymbol{y}_{t+1} = \boldsymbol{y}_t + \boldsymbol{u}_t, \quad t \in T', \tag{2.5}$$

while the exogenous variables evolve according to the non-stationary, conditional probability distribution $\mathbb{P}_t(\boldsymbol{w}_{t+1}|\boldsymbol{w}_t)$. The transition probabilities of the induced Markov chain are

$$\mathbb{P}_t^{\boldsymbol{\pi}}(\boldsymbol{s}_{t+1}|\boldsymbol{s}_t) = \psi(\boldsymbol{y}_{t+1} - \boldsymbol{y}_t - \boldsymbol{u}_t)\mathbb{P}_t(\boldsymbol{w}_{t+1}|\boldsymbol{w}_t), \quad t \in T',\ (\boldsymbol{s}_t, \boldsymbol{s}_{t+1}) \in \mathcal{S}_t \times \mathcal{S}_{t+1}, \tag{2.6}$$

where $\psi(\mathbf{a})$ is the Kronecker-delta function, i.e., for $\mathbf{a} \in \mathbb{R}^n$, $\psi(\mathbf{a}) = 1$ when $\mathbf{a} = \mathbf{0}$, and $\psi(\mathbf{a}) = 0$ otherwise.

Next, we describe the objective function which is the cost to be minimized. The one-step cost incurred in stage t is

$$c_t(\boldsymbol{s}_t, \boldsymbol{x}_t) = p_t(g_t^1 + g_t^2) + \varphi(|u_t^1| + |u_t^2|) + \nu\xi(\boldsymbol{u}_t, q_t), \quad t \in T', \tag{2.7}$$

and $c_N(\boldsymbol{s}_N) = 0$ without loss of generality. The first term on the right-hand side (r.h.s.) of (2.7) is the total cost of procuring or selling energy, the second term is the total cost of charging or discharging energy, and the third term is the cost of resistive line losses, where ξ, the resistive line-loss function, is non-negative, separable and convex in $(\boldsymbol{u}_t, q_t)$ (cf. [17, 143]). It is assumed that $|c_t(\boldsymbol{s}_t, \boldsymbol{x}_t)| < \infty$. For an *a priori* storage configuration $\boldsymbol{\alpha}$, the operator seeks an optimal policy $\boldsymbol{\pi}^* \in \Pi$ that minimizes the expected total discounted costs over the planning horizon as follows:

$$z_{\boldsymbol{\alpha}} = \min_{\boldsymbol{\pi} \in \Pi} \mathbb{E}_{\boldsymbol{s}}^{\boldsymbol{\pi}} \left(\sum_{t \in T'} \delta^{t-1} c_t(\boldsymbol{s}_t, \boldsymbol{x}_t) \middle| \boldsymbol{S}_1 = \boldsymbol{s}; \boldsymbol{\alpha} \right), \tag{2.8}$$

where $\delta \in [0,1]$ is a discount factor. Henceforth, $z_{\boldsymbol{\alpha}}$ is called the optimal operational cost. Bellman's optimality equations are then

$$V_t(\boldsymbol{s}_t) = \min_{\boldsymbol{x}_t \in \mathcal{X}_t(\boldsymbol{y}_t)} c_t(\boldsymbol{s}_t, \boldsymbol{x}_t) + \delta \mathbb{E}(V_{t+1}(\boldsymbol{s}_{t+1}) | \boldsymbol{s}_t, \boldsymbol{x}_t), \quad t \in T', \tag{2.9}$$

with $V_N(\boldsymbol{s}_N) = 0$.

2.3 STRUCTURAL RESULTS

In this section, we examine important structural properties of the value function, V_t, the optimal policy $\boldsymbol{\pi}^*$ and the optimal cost $z_{\boldsymbol{\alpha}}$. First, we examine properties of V_t that depend on the endogenous storage levels $\boldsymbol{y}_t$. In what follows, a fixed exogenous state is denoted by $\bar{\boldsymbol{w}}_t$.

2.3.1 Structural Properties of the Value Function

Proposition 2.1 asserts that, for a fixed exogenous state, the expected future cost at each stage is jointly convex in the storage levels. Stated more clearly, the marginal cost of using storage increases with increasing storage levels.

Proposition 2.1. *For each $t \in T$, $V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t)$ is convex in $\boldsymbol{y}_t \in \mathcal{Y}$.*

Proof. The result is proved using backward induction on $V_t(\boldsymbol{w}_t, \boldsymbol{y}_t)$ for a fixed $\boldsymbol{w}_t$. By assumption, $V_N(\boldsymbol{s}_N) = 0$ for all $\boldsymbol{s}_N \in \mathcal{S}_N$, so the result clearly holds at stage N. For the induction hypothesis, suppose $V_{t+1}(\boldsymbol{w}_{t+1}, \boldsymbol{y}_{t+1})$ is convex in $\boldsymbol{y}_{t+1}$, given a fixed $\boldsymbol{w}_{t+1}$, for $t+1 < N$. Note that the expectation in (2.9) is taken with respect to (w.r.t.) the conditional probability distribution $\mathbb{P}_t(\boldsymbol{w}_{t+1} | \bar{\boldsymbol{w}}_t)$. Moreover, $\boldsymbol{y}_{t+1}$ is a linear, deterministic function of $\boldsymbol{y}_t$ and $\boldsymbol{u}_t$ by (2.5). Therefore, the expectation in (2.9) can be expressed around the post-decision state $\boldsymbol{s}_t^x = (\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t + \boldsymbol{u}_t)$ (see [120] for additional details), so that

$$V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t) = \min_{\boldsymbol{x}_t \in \mathcal{X}_t(\boldsymbol{y}_t)} c_t(\bar{\boldsymbol{w}}_t, \boldsymbol{x}_t) + V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t + \boldsymbol{u}_t), \tag{2.10}$$

where $V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t + \boldsymbol{u}_t) \equiv \delta\mathbb{E}(V_{t+1}(\boldsymbol{s}_{t+1}|\boldsymbol{s}_t^x))$ is called the post-decision value function. As the expectation operator preserves convexity, and compositions of convex and affine functions are also convex (see Proposition 2.1.3 (b) of [140]), $V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t + \boldsymbol{u}_t)$ is convex in $(\boldsymbol{y}_t, \boldsymbol{u}_t)$. Note that an optimal solution to (2.10) exists because $\mathcal{X}_t(\boldsymbol{y}_t)$ is a bounded polyhedron. Moreover, $c_t(\bar{\boldsymbol{w}}_t, \boldsymbol{x}_t)$ is piecewise-convex in $\boldsymbol{x}_t$. As the sum of two convex functions is convex, the objective function of (2.10) is jointly convex in $(\boldsymbol{y}_t, \boldsymbol{x}_t)$. As convexity is preserved under partial minimization (see Section 3.2.5 in [25]), we conclude that $V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t)$ is convex in $\boldsymbol{y}_t$. □

Intuitively, Proposition 2.1 implies that the flexibility to store surplus generation decreases with an increase in the current storage level. Consequently, the operator must either sell the excess energy, possibly at a lower price, or transmit a portion of it to the other bus, possibly incurring line-loss costs. Note that, if the functional form of V_t is known for a fixed $\bar{\boldsymbol{w}}_t$, the optimality equations (2.9) can be solved efficiently using convex optimization algorithms. Unfortunately, characterizing the expectation in (2.9) is nontrivial due to the multidimensional nature of $\mathcal{S}_t$ and $\mathcal{X}_t(\boldsymbol{y}_t)$. The next result, Proposition 2.2, asserts that the expected future cost at each stage is monotone decreasing in the storage levels.

Proposition 2.2. *For each $t \in T$, $V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t)$ is monotone decreasing in $\boldsymbol{y}_t \in \mathcal{Y}$.*

Proof. The proposition is proved via backward induction on $V_t(\boldsymbol{w}_t, \boldsymbol{y}_t)$ for a fixed $\boldsymbol{w}_t$. Clearly, the result holds at stage N. For the induction hypothesis, suppose $V_{t+1}(\boldsymbol{w}_{t+1}, \boldsymbol{y}_{t+1})$ is monotone decreasing in $\boldsymbol{y}_{t+1}$ for $t+1 < N$ for a fixed $\boldsymbol{w}_{t+1}$. As the expectation operator preserves monotonicity, the function $V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t + \boldsymbol{u}_t)$ in (2.10) is decreasing in $\boldsymbol{y}_t$ and $\boldsymbol{u}_t$. Next, consider two states $\boldsymbol{s}_t^a = (\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^a)$ and $\boldsymbol{s}_t^b = (\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^b)$, such that $0 \leq \boldsymbol{y}_t^a < \boldsymbol{y}_t^b \leq \boldsymbol{\alpha}$. We seek to show that $V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^a) \geq V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^b)$. To this end, let $\boldsymbol{x}_t^a = (\boldsymbol{u}_t^a, q_t^a)$ and $\boldsymbol{x}_t^b = (\boldsymbol{u}_t^b, q_t^b)$ be the optimal solutions of (2.10) for states $\boldsymbol{s}_t^a$ and $\boldsymbol{s}_t^b$, respectively. Then,

$$V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^a) = c_t(\bar{\boldsymbol{w}}_t, \boldsymbol{u}_t^a, q_t^a) + V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^a + \boldsymbol{u}_t^a),$$
$$V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^b) = c_t(\bar{\boldsymbol{w}}_t, \boldsymbol{u}_t^b, q_t^b) + V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^b + \boldsymbol{u}_t^b).$$

Consider the following two cases:

Case 1: Suppose $\boldsymbol{x}_t^a \in \mathcal{X}_t(\boldsymbol{y}_t^b)$. As $\boldsymbol{x}_t^a$ is feasible for problem (2.10) in state $\boldsymbol{s}_t^b$, the optimal value $V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^b)$ is at most equal to the objective value at $\boldsymbol{x}_t^a$, i.e.,

$$c_t(\bar{\boldsymbol{w}}_t, \boldsymbol{u}_t^a, q_t^a) + V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^b + \boldsymbol{u}_t^a) \geq V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^b). \tag{2.11}$$

As $\boldsymbol{y}_t^a < \boldsymbol{y}_t^b$, the following inequality holds by the induction hypothesis on V_t^x:

$$V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^a + \boldsymbol{u}_t^a) \geq V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^b + \boldsymbol{u}_t^a). \tag{2.12}$$

Adding $c_t(\bar{\boldsymbol{w}}_t, \boldsymbol{u}_t^a, q_t^a)$ to both sides of (2.12) and combining it with (2.11) yields

$$c_t(\bar{\boldsymbol{w}}_t, \boldsymbol{u}_t^a, q_t^a) + V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^a + \boldsymbol{u}_t^a) \geq c_t(\bar{\boldsymbol{w}}_t, \boldsymbol{u}_t^a, q_t^a) + V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^b + \boldsymbol{u}_t^a) \geq V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^b), \tag{2.13}$$

where the left-most expression in (2.13) equals $V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^a)$ by definition. Hence, $V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^a) \geq V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^b)$.

Case 2: Suppose $\boldsymbol{x}_t^a \notin \mathcal{X}_t(\boldsymbol{y}_t^b)$. A sufficient condition for $\boldsymbol{x}_t^a \notin \mathcal{X}_t(\boldsymbol{y}_t^b)$ is $\boldsymbol{u}_t^a \in (\boldsymbol{\alpha} - \boldsymbol{y}_t^b, \min\{\boldsymbol{\tau}_c, \boldsymbol{\alpha} - \boldsymbol{y}_t^a\}]$. Construct a feasible solution $\bar{\boldsymbol{x}}_t^b = (\bar{\boldsymbol{u}}_t^b, \bar{q}_t^b) \in \mathcal{X}_t(\boldsymbol{y}_t^b)$ such that $\bar{\boldsymbol{u}}_t^b = \boldsymbol{\alpha} - \boldsymbol{y}_t^b < \boldsymbol{u}_t^a$ and $\bar{q}_t^b = q_t^a$. For such a case,

$$\begin{aligned}
V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^a) = c_t(\bar{\boldsymbol{w}}_t, \boldsymbol{u}_t^a, q_t^a) + V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^a + \boldsymbol{u}_t^a) &\geq c_t(\bar{\boldsymbol{w}}_t, \bar{\boldsymbol{u}}_t^b, \bar{q}_t^b) + V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^a + \boldsymbol{u}_t^a) \\
&\geq c_t(\bar{\boldsymbol{w}}_t, \bar{\boldsymbol{u}}_t^b, \bar{q}_t^b) + V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{\alpha}) \\
&\geq c_t(\bar{\boldsymbol{w}}_t, \boldsymbol{u}_t^b, \boldsymbol{q}_t^b) + V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^b + \boldsymbol{u}_t^b) \\
&= V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^b).
\end{aligned}$$

The first inequality holds because $c_t(\bar{\boldsymbol{w}}_t, \boldsymbol{u}_t^a, q_t^a) \geq c_t(\bar{\boldsymbol{w}}_t, \bar{\boldsymbol{u}}_t^b, \bar{q}_t^b)$ for $\boldsymbol{u}_t^a > \bar{\boldsymbol{u}}_t^b$ and $q_t^a = \bar{q}_t^b$. The second inequality holds by the induction hypothesis on V_t^x. The third inequality holds because $(\bar{\boldsymbol{u}}_t^b, \bar{q}_t^b)$ is a feasible, but not necessarily optimal, solution to problem (2.10) for state $\boldsymbol{s}_t^b$. Hence, we conclude that $V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^a) \geq V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t^b)$, and the proof is complete. □

Proposition 2.2 suggests that stored energy tends to reduce the expected operational costs. When the storage levels are high, a larger fraction of the demand is satisfied by using stored energy, thereby reducing the overall operational cost. Moreover, higher storage levels allow the operator to satisfy demand and sell any excess energy back to the grid. This is especially useful during the peak-price, peak-demand periods.

Next, we present a result for the special case in which the load buses have similar operational characteristics. The load buses are called homogenous if: (i) $\alpha_1 = \alpha_2$, and (ii) the (conditional) joint cumulative distribution function (c.d.f.) of the net demands at each stage is *symmetric*, i.e., for $t \in T$, $k \in \{t, \ldots, N\}$ and any $a_1, a_2 \in \mathbb{R}$,

$$\mathbb{P}_k(D^1_{k+1} \leq a_1,\, D^2_{k+1} \leq a_2 | \boldsymbol{D}_k; \mathscr{A}_k) = \mathbb{P}_k(D^1_{k+1} \leq a_2,\, D^2_{k+1} \leq a_1 | \boldsymbol{D}_k; \mathscr{A}_k). \tag{2.14}$$

Condition (2.14) is indicative of a joint distribution function in $\mathbb{R}^2$ that is symmetric along the line $a_1 = a_2$. Proposition 2.3 asserts that, for a pair of homogenous buses, allocating the total stored energy equally between the two buses minimizes the expected future cost at each stage. For ease of exposition, let $\mathcal{Y}^{\Theta}_t \equiv \{\boldsymbol{y}_t \in \mathcal{Y} : y^1_t + y^2_t = \Theta\}$ denote the set of feasible storage-level allocations at stage t when the total stored energy in the network is $\Theta \in [0, \alpha_1 + \alpha_2]$.

Proposition 2.3. *For each $t \in T$ and a fixed Θ,*

$$V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t) \geq V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{\Theta}/2), \quad \forall \boldsymbol{y}_t \in \mathcal{Y}^{\Theta}_t.$$

Proof. The result obviously holds at stage N. For $t \in T'$, consider two feasible storage-level vectors $\boldsymbol{y}^a_t$ and $\boldsymbol{y}^b_t$, such that $\boldsymbol{y}^a_t, \boldsymbol{y}^b_t \in \mathcal{Y}^{\Theta}_t$. By definition, $(\boldsymbol{y}^a_t + \boldsymbol{y}^b_t)/2 = (\Theta/2, \Theta/2) \equiv \boldsymbol{\Theta}/2$. For a pair of homogenous buses that satisfy the conditions $\alpha_1 = \alpha_2$ and (2.14), it follows directly that V_t is symmetric w.r.t. $\boldsymbol{y}_t$ for a fixed $\bar{\boldsymbol{w}}_t$, i.e., $V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}^a_t) = V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}^b_t)$. Using Jensen's inequality for the convex function V_t at the points $\boldsymbol{y}_t = \boldsymbol{y}^a_t$ and $\boldsymbol{y}_t = \boldsymbol{y}^b_t$, we obtain

$$V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}^a_t) = \frac{1}{2}\left(V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}^a_t) + V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}^b_t)\right) \geq V_t(\bar{\boldsymbol{w}}_t, (\boldsymbol{y}^a_t + \boldsymbol{y}^b_t)/2) = V_t(\bar{\boldsymbol{w}}_t, \Theta/2, \Theta/2).$$

As $\boldsymbol{y}^a_t$ (or $\boldsymbol{y}^b_t$) is any feasible element in $\mathcal{Y}^{\Theta}_t$, we conclude that $V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t) \geq V_t(\bar{\boldsymbol{w}}_t, \Theta/2, \Theta/2)$ for all $\boldsymbol{y}_t \in \mathcal{Y}^{\Theta}_t$. □

2.3.2 Behavior of the Optimal Policy

Here we examine structural properties of the optimal policy $\boldsymbol{\pi}^*$. For a fixed $\bar{\boldsymbol{w}}_t$, the optimality equations (2.9) are a collection of parameterized optimization problems in which the objective function and the constraints depend on the storage levels $\boldsymbol{y}_t$. For such a class of problems, *monotone comparative statics* [150] can be used to characterize the monotone behavior of optimal decisions with respect to the state variables. Moreover, monotone comparative statics are useful for problems in which the value function is non-differentiable [106] and are closely linked to the concept of *substitutability.* Two variables are called economic substitutes if an increase in one variable increases the marginal cost of the other variable [140]. The property of multimodularity [5, 63, 112] is known to imply substitutability, and is inherently related to the concepts of supermodularity and increasing differences that arise frequently in lattice theory [150]. In what follows, we show that, for a fixed $\bar{\boldsymbol{w}}_t$, V_t is multimodular in $\boldsymbol{y}_t$, and the optimal decisions $\boldsymbol{x}_t^*$ are not only monotone, but also economic substitutes of $\boldsymbol{y}_t$. We first review some needed elements of lattice theory.

Consider $A \subset \mathbb{R}^n$ with the standard component-wise order $\leq$; that is, for any $\mathbf{a}, \mathbf{a}' \in A$, $\mathbf{a} \leq \mathbf{a}'$ if and only if $a_i \leq a_i'$ for each $i = 1, \ldots, n$. Any subset of $\mathbb{R}^n$ is a partially-ordered set (or poset) by definition (see Section 2.2 of [150]). A special poset, namely a lattice, is a group-algebraic structure as next defined.

Definition 2.1. *A poset $(A, \leq)$ is called a lattice if and only if for any $\mathbf{a}, \mathbf{a}' \in A$,*

$$\mathbf{a} \vee \mathbf{a}' \equiv (\sup\{a_1, a_1'\}, \ldots, \sup\{a_n, a_n'\}) \in A,$$
$$\mathbf{a} \wedge \mathbf{a}' \equiv (\inf\{a_1, a_1'\}, \ldots, \inf\{a_n, a_n'\}) \in A.$$

In words, a lattice is a poset whose nonempty, finite subsets possess a supremum and an infimum. Given a lattice $(A, \leq)$, any $S \subseteq A$ is called a sublattice of A if S is itself a lattice. Note that $\mathbb{R}^n$ is a lattice by definition. Next, we review important properties of functions defined on lattices.

Definition 2.2. *A mapping $f : A \to \mathbb{R}$ is supermodular on A if for any $\mathbf{a}, \mathbf{a}' \in A$,*

$$f(\mathbf{a}) + f(\mathbf{a}') \leq f(\mathbf{a} \vee \mathbf{a}') + f(\mathbf{a} \wedge \mathbf{a}').$$

The function f is said to be submodular on A if $-f$ is supermodular on A.

Supermodular functions exhibit the more intuitive increasing differences property (see Theorem 2.2.2 in [140]). Given two posets $(M, \leq)$ and $(N, \leq)$, a mapping $f : M \times N \to \mathbb{R}$ has increasing differences if for any $n, n' \in N$ with $n \leq n'$, $f(m, n') - f(m, n)$ is increasing in $m \in M$. Clearly, increasing differences, and therefore supermodularity, imply substitutability. However, supermodularity is not preserved under minimization [97, 140]. By contrast, multimodularity, which is next defined, is preserved under minimization [97, 170].

Definition 2.3. *Let $A = \{(\boldsymbol{v}, b) \in \mathbb{R}^{n+1} : (v_1 - b, v_2 - v_1, \ldots, v_n - v_{n-1}) \in U \subseteq \mathbb{R}^n, b \in \mathbb{R}\}$ be a lattice characterized by the posets $(U, \leq)$ and $(\mathbb{R}, \leq)$. A mapping $f : U \to \mathbb{R}$ is said to be multimodular on U if $\Psi(\boldsymbol{v}, b) \equiv f(v_1 - b, v_2 - v_1, \ldots, v_n - v_{n-1})$ is submodular on A.*

To establish the multimodularity of V_t, we first recast the sets $\mathcal{Y}$, $\mathcal{X}_t(\boldsymbol{y}_t)$ and $\mathcal{U} \equiv \mathcal{Y} \times \cup_{\boldsymbol{y}_t \in \mathcal{Y}} \mathcal{X}_t(\boldsymbol{y}_t)$ as lattices by employing the following change of variables: $y_t^1 = v_1 - b$, $y_t^2 = v_2 - v_1$, $u_t^1 = r_1 - v_1$, $u_t^2 = r_2 - v_2$, and $q_t = r_3 - r_2$, where $(\boldsymbol{v}, b) \equiv (v_1, v_2, b)$ and $\boldsymbol{r} \equiv (r_1, r_2, r_3)$. Redefining the sets $\mathcal{Y}$, $\mathcal{U}$ and $\mathcal{X}_t(\boldsymbol{y}_t)$, respectively, we obtain

$$\mathcal{V} \equiv \left\{(\boldsymbol{v}, b) \in \mathbb{R}^3 : v_1 - b \in [0, \alpha_1], v_2 - v_1 \in [0, \alpha_2]\right\}, \tag{2.15}$$

$$\mathcal{L} \equiv \left\{(\boldsymbol{v}, b, \boldsymbol{r}) \in \mathbb{R}^6 : (\boldsymbol{v}, b) \in \mathcal{V}, (r_1 - v_1, r_2 - v_2, r_3 - r_2) \in \mathcal{X}_t(v_1 - b, v_2 - v_1)\right\}, \tag{2.16}$$

$$\mathcal{L}(\boldsymbol{v}, b) \equiv \left\{\boldsymbol{r} \in \mathbb{R}^3 : (r_1 - v_1, r_2 - v_2, r_3 - r_2) \in \mathcal{X}_t(v_1 - b, v_2 - v_1)\right\}. \tag{2.17}$$

The set $\mathcal{U}$ is the set of all feasible state-action pairs in stage t for a fixed $\bar{\boldsymbol{w}}_t$, and for any $(\boldsymbol{v}, b) \in \mathcal{V}$, $\mathcal{L}(\boldsymbol{v}, b)$ is called a *section* of $\mathcal{L}$ at $(\boldsymbol{v}, b)$ (see page 16 in [150]). Henceforth, we assume that $\boldsymbol{\tau}_c, \boldsymbol{\tau}_d > \boldsymbol{\alpha}$ to simplify the analysis. Proposition 2.4 asserts that the posets $\mathcal{V}$, $\mathcal{L}$ and $\mathcal{L}(\boldsymbol{v}, b)$ are lattices.

Proposition 2.4. *The sets $\mathcal{V}$, $\mathcal{L}$ and $\mathcal{L}(\boldsymbol{v}, b)$ are lattices.*

Proof. We first consider the set $\mathcal{L}$ that is characterized by the following constraints:

$$0 \leq r_1 - b \leq \alpha_1, \tag{2.18a}$$

$$0 \leq r_2 - v_1 \leq \alpha_2, \tag{2.18b}$$

$$-\beta \leq r_3 - r_2 \leq \beta, \tag{2.18c}$$

$$0 \leq v_1 - b \leq \alpha_1, \tag{2.18d}$$

$$0 \leq v_2 - v_1 \leq \alpha_2. \tag{2.18e}$$

Define $\mathbf{a} \equiv (\boldsymbol{v}, b, \boldsymbol{r}) \in \mathcal{L}$. It is noted that each constraint in $\mathcal{L}$ has exactly two variables with the coefficients $+1$ and -1, while the remaining coefficients are equal to zero. Thus, each constraint in $\mathcal{L}$ defines an affine half-space of the form $A_{i,j} = \{\mathbf{a} \in \mathbb{R}^6 : a_i - a_j \leq h, a_k = 0, i \neq j, k \neq i, j\}$. We show that each such $A_{i,j}$ is a lattice. Consider two points $\mathbf{a}, \mathbf{a}' \in A_{i,j}$. For the cases $\mathbf{a} \leq \mathbf{a}'$ and $\mathbf{a}' \leq \mathbf{a}$, it is easy to verify that $\mathbf{a} \vee \mathbf{a}' \in A_{i,j}$ and $\mathbf{a} \wedge \mathbf{a}' \in A_{i,j}$. Next, consider the case when $a_i \geq a'_i$, $a'_j \geq a_j$, and $a_k = a'_k = 0$ for all $k \neq i, j$. Then, the ith and jth components of $\mathbf{a} \wedge \mathbf{a}'$ and $\mathbf{a} \vee \mathbf{a}'$ are (a'_i, a_j) and (a_i, a'_j), respectively. As $\mathbf{a} \in A_{i,j}$, $a_i \leq a_j + h$. But $a'_i \leq a_i$, and therefore, $a'_i \leq a_j + h \Rightarrow a'_i - a_j \leq h$, which implies that $\mathbf{a} \wedge \mathbf{a}' \in A_{i,j}$. Similarly, $a_i \leq h + a_j \leq h + a'_j \Rightarrow a_i - a'_j \leq h$ as $\mathbf{a}' \in A_{i,j}$ and $a'_j \geq a_j$, and therefore, $\mathbf{a} \vee \mathbf{a}' \in A_{i,j}$. Similar arguments are valid when $a_i \leq a'_i$, $a'_j \leq a_j$ and $a_k = a'_k = 0$ for all $k \neq i, j$; therefore, $A_{i,j}$ is a lattice. As a finite intersection of lattices is also a lattice (see Lemma 2.2.2 of [150]), $\mathcal{L}$ is a lattice. Following similar lines of reasoning, we can show that $\mathcal{V}$, defined by constraints (2.18d)–(2.18e), is also a lattice. For a fixed $(\boldsymbol{v}, b) \in \mathcal{V}$, $\mathcal{L}(\boldsymbol{v}, b)$ is defined by the constraints (2.18a)–(2.18c). Because $\mathcal{L}(\boldsymbol{v}, b)$ is a section of $\mathcal{L}$ at $(\boldsymbol{v}, b)$, it is also a lattice by Lemma 2.2.3 of [150]. □

Next, we present our main result, Theorem 2.1, which asserts the multimodularity of V_t, and the monotonicity of $\boldsymbol{x}_t^*$, with respect to $\boldsymbol{y}_t$. With a slight abuse of notation, let $\Delta_i f(\bar{\boldsymbol{w}}_t, \mathbf{a})$ denote both the forward and backward finite differences of a function $f(\bar{\boldsymbol{w}}_t, \mathbf{a})$ with respect to dimension i of $\mathbf{a}$. Specifically, for some $\epsilon > 0$, the forward difference of f is

$$\Delta_i f(\bar{\boldsymbol{w}}_t, \mathbf{a}) = f(\bar{\boldsymbol{w}}_t, \mathbf{a} + \epsilon \boldsymbol{e}_i) - f(\bar{\boldsymbol{w}}_t, \mathbf{a}),$$

and the backward difference is $f(\bar{\boldsymbol{w}}_t, \mathbf{a}) - f(\bar{\boldsymbol{w}}_t, \mathbf{a} - \epsilon \boldsymbol{e}_i)$, where $\boldsymbol{e}_i$ is the ith unit vector. Similarly, let $\Delta_{i,j} f(\bar{\boldsymbol{w}}_t, \mathbf{a})$ be the second-order finite difference of $f(\bar{\boldsymbol{w}}_t, \mathbf{a})$, with respect to dimensions i and j of $\mathbf{a}$, defined by

$$\Delta_{i,j} f(\bar{\boldsymbol{w}}_t, \mathbf{a}) = \Delta_j \left(\Delta_i f(\bar{\boldsymbol{w}}_t, \mathbf{a})\right).$$

Theorem 2.1. *For each $t \in T$ and a fixed exogenous state $\bar{\boldsymbol{w}}_t$,*

(i) $V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t)$ is multimodular in $\boldsymbol{y}_t \in \mathcal{Y}$;

(ii) $V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t)$ *has increasing differences and is component-wise convex in* $\boldsymbol{y}_t \in \mathcal{Y}$*. That is,*

$$\Delta_{1,1} V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t) \geq \Delta_{1,2} V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t) \geq 0,$$
$$\Delta_{2,2} V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t) \geq \Delta_{2,1} V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t) \geq 0;$$

(iii) $\boldsymbol{x}_t^* = (\boldsymbol{u}_t^*, q_t^*)$ *is monotone decreasing in* $\boldsymbol{y}_t \in \mathcal{Y}$*. Furthermore, if* q_t *is fixed, then*

$$-1 \leq \Delta_1 \, u_t^{1*}(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t) \leq \Delta_2 \, u_t^{1*}(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t) \leq 0,$$
$$-1 \leq \Delta_2 \, u_t^{2*}(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t) \leq \Delta_1 \, u_t^{2*}(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t) \leq 0.$$

Proof. To prove part (i), we use backward induction on $V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t)$. The result clearly holds for stage N. For the induction hypothesis, suppose $V_{t+1}(\boldsymbol{w}_{t+1}, \boldsymbol{y}_{t+1})$ is multimodular in $\boldsymbol{y}_{t+1} \in \mathcal{Y}$ for any $\boldsymbol{w}_{t+1} \in \mathcal{W}_{t+1}$. We seek to show that $V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t)$ is multimodular in $\boldsymbol{y}_t \in \mathcal{Y}$. This is equivalent to showing that the function

$$\begin{aligned}\Psi(\bar{\boldsymbol{w}}_t, \boldsymbol{v}, b) &= V_t(\bar{\boldsymbol{w}}_t, v_1 - b, v_2 - v_1), \\ &= \min_{\boldsymbol{u}_t, q_t} \left\{ c_t(\bar{\boldsymbol{w}}_t, \boldsymbol{u}_t, q_t) + \delta \mathbb{E}(V_{t+1}(\boldsymbol{W}_{t+1}, v_1 - b + u_t^1, v_2 - v_1 + u_t^2)) \right\}, \\ &= \min_{\boldsymbol{r}} \left\{ c_t(\bar{\boldsymbol{w}}_t, r_1 - v_1, r_2 - v_2, r_3 - r_2) + \delta \mathbb{E}\left(V_{t+1}(\boldsymbol{W}_{t+1}, r_1 - b, r_2 - v_1)\right) \right\},\end{aligned} \tag{2.19}$$

is submodular in $(\boldsymbol{v}, b) \in \mathcal{V}$, such that $(r_1 - v_1, r_2 - v_2, r_3 - r_2) \in \mathcal{X}_t(v_1 - b, v_2 - v_1)$ and $\boldsymbol{r} \in \mathbb{R}^3$. First, we establish that the objective function of (2.19) is submodular in $(\boldsymbol{v}, b, \boldsymbol{r}) \in \boldsymbol{\mathcal{L}}$, where $\boldsymbol{\mathcal{L}}$ is defined in (2.16). The post-decision value function $V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_{t+1})$ is

$$V_t^x(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_{t+1}) = \sum_{\boldsymbol{w}_{t+1} \in \mathcal{W}_{t+1}} \delta \mathbb{P}^{\boldsymbol{\pi}^*}(\boldsymbol{w}_{t+1} | \bar{\boldsymbol{w}}_t) V_{t+1}(\boldsymbol{w}_{t+1}, \boldsymbol{y}_{t+1}).$$

As $\delta \mathbb{P}^{\boldsymbol{\pi}^*}(\boldsymbol{w}_{t+1} | \bar{\boldsymbol{w}}_t) \geq 0$, and a non-negative affine combination of multimodular functions is multimodular by Lemma 2 (i) of [97], V_t^x is multimodular in $\boldsymbol{y}_{t+1}$. Note that $\boldsymbol{y}_{t+1} = \boldsymbol{y}_t + \boldsymbol{u}_t = (u_t^1 + y_t^1, y_t^2 + u_t^2)$. Using Lemma 2 (vii) in [97], we conclude that V_t^x is multimodular in $(u_t^1, y_t^1, y_t^2, u_t^2)$, or equivalently that V_t^x is submodular in $(\boldsymbol{v}, b, \boldsymbol{r})$. The one-step cost in (2.19) is

$$\begin{aligned}c_t(\bar{\boldsymbol{w}}_t, r_1 - v_1, r_2 - v_2, r_3 - r_2) = k + \theta_t^1(r_1 - v_1) &+ \theta_t^2(r_2 - v_2) + \varphi(|r_1 - v_1| + |r_2 - v_2|) \\ &+ \nu \xi(r_1 - v_1, r_2 - v_2, r_3 - r_2),\end{aligned} \tag{2.20}$$

where k is a constant that depends on $\bar{\boldsymbol{w}}_t$. As the absolute value function and ξ are convex, by Theorem 2.2.6 (b) of [140], we have that the terms $|r_1-v_1|$, $|r_2-v_2|$ and $\xi(r_1-v_1, r_2-v_2, r_3-r_2)$ in (2.20) are submodular in $(\boldsymbol{v}, b, \boldsymbol{r})$. Moreover, the linear terms in (2.20) are submodular in $(\boldsymbol{v}, b, \boldsymbol{r})$ by Lemma 2.2.3 in [140]. As the sum of two submodular functions is submodular by Lemma 2.6.1 in [150], the objective function of (2.19) is also submodular in $(\boldsymbol{v}, b, \boldsymbol{r}) \in \mathcal{L}$. It is noted that problem (2.19) involves minimizing a submodular function in $(\boldsymbol{v}, b, \boldsymbol{r})$ along a section $\mathcal{L}(\boldsymbol{v}, b)$ of $\mathcal{L}$ at some $(\boldsymbol{v}, b) \in \mathcal{V}$. Also, $\Psi(\bar{\boldsymbol{w}}_t, \boldsymbol{v}, b) > -\infty$ because $\mathcal{L}(\boldsymbol{v}, b)$ is a polyhedron. Then, using Theorem 2.7.6 in [150], we establish that $\Psi(\bar{\boldsymbol{w}}_t, \boldsymbol{v}, b)$ is submodular in $(\boldsymbol{v}, b) \in \mathcal{V}$. Therefore, $V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t)$ is multimodular in $\boldsymbol{y}_t \in \mathcal{Y}$.

To prove Theorem 2.1 (ii), the dependence of V_t and Ψ on $\bar{\boldsymbol{w}}_t$ is suppressed to simplify notation. Let $\epsilon > 0$ and note that

$$\begin{aligned}\Delta_{2,3}\Psi(\boldsymbol{v}, b) &= \Psi(v_1, v_2+\epsilon, b+\epsilon) - \Psi(v_1, v_2+\epsilon, b) - \Psi(v_1, v_2, b+\epsilon) + \Psi(v_1, v_2, b),\\ &= V_t(y_t^1-\epsilon, y_t^2+\epsilon) - V_t(y_t^1, y_t^2+\epsilon) - V_t(y_t^1-\epsilon, y_t^2) + V_t(y_t^1, y_t^2),\\ &= -\Delta_{1,2}V_t(\boldsymbol{y}_t), \end{aligned} \tag{2.21}$$

where the last equality stems from successive forward and backward finite difference operations on V_t w.r.t. y_t^2 and y_t^1, respectively. As submodularity implies decreasing differences by Theorem 2.2.2 in [140], we have $\Delta_{2,3}\Psi(\boldsymbol{v}, b) \le 0 \Rightarrow \Delta_{1,2}V_t(\boldsymbol{y}_t) \ge 0$ by equation (2.21). Next, we show that $\Delta_{2,2}V_t(\boldsymbol{y}_t) \ge \Delta_{1,2}V_t(\boldsymbol{y}_t)$. By definition,

$$\begin{aligned}\Delta_{1,3}\Psi(\boldsymbol{v}, b) &= \Psi(v_1+\epsilon, v_2, b+\epsilon) - \Psi(v_1+\epsilon, v_2, b) - \Psi(v_1, v_2, b+\epsilon) + \Psi(v_1, v_2, b),\\ &= V_t(y_t^1, y_t^2-\epsilon) - V_t(y_t^1+\epsilon, y_t^2-\epsilon) - V_t(y_t^1-\epsilon, y_t^2) + V_t(y_t^1, y_t^2). \end{aligned} \tag{2.22}$$

Similarly, successive forward and backward finite difference operations on V_t yield

$$\Delta_{1,2}V_t(\boldsymbol{y}_t) = V_t(y_t^1+\epsilon, y_t^2) - V_t(y_t^1+\epsilon, y_t^2-\epsilon) - V_t(y_t^1, y_t^2) + V_t(y_t^1, y_t^2-\epsilon), \tag{2.23}$$

$$\Delta_{1,1}V_t(\boldsymbol{y}_t) = V_t(y_t^1+\epsilon, y_t^2) - 2V_t(y_t^1, y_t^2) + V_t(y_t^1-\epsilon, y_t^2). \tag{2.24}$$

Subtracting (2.24) from (2.23), we see that $\Delta_{1,2}V_t(\boldsymbol{y}_t) - \Delta_{1,1}V_t(\boldsymbol{y}_t) = \Delta_{1,3}\Psi(\boldsymbol{v}, b)$ by (2.22). But $\Delta_{1,3}\Psi(\boldsymbol{v}, b) \le 0$, as Ψ is a submodular function. Therefore,

$$0 \le \Delta_{1,2}V_t(\boldsymbol{y}_t) \le \Delta_{1,1}V_t(\boldsymbol{y}_t). \tag{2.25}$$

Likewise, it can be shown that $0 \leq \Delta_{2,1}V_t(\boldsymbol{y}_t) \leq \Delta_{2,2}V_t(\boldsymbol{y}_t)$. Therefore, V_t has increasing differences and component-wise convexity.

To prove Theorem 2.1 (*iii*), we note that $\mathcal{L}$ is a sublattice of $\mathcal{V} \times \mathbb{R}^3$, both of which are lattices. Hence, for any $(\boldsymbol{v}, b) \in \mathcal{V}$, the section $\mathcal{L}(\boldsymbol{v}, b)$ is also a sublattice by Lemma 2.2.3 (a) of [150]. Let $\sqsubseteq$ denote the *strong set order* defined for subsets of a lattice, where for any $A', A'' \subseteq A$, $A' \sqsubseteq A'' \Rightarrow \mathbf{a}' \wedge \mathbf{a}'' \in A'$ and $\mathbf{a}' \vee \mathbf{a}'' \in A''$ for all $\mathbf{a}' \in A'$ and $\mathbf{a}'' \in A''$ (see Section 2.4 of [150]). As $\mathcal{L}(\boldsymbol{v}, b)$ is a sublattice, it is also an increasing set function in $(\boldsymbol{v}, b) \in \mathcal{V}$ with respect to $\sqsubseteq$ by Theorem 2.4.5(a) of [150]. It is easy to verify that $\mathcal{L}(\boldsymbol{v}, b)$ is nonempty for any $(\boldsymbol{v}, b) \in \mathcal{V}$. Therefore, we conclude that $\boldsymbol{x}_t^*$ is monotone decreasing in $\boldsymbol{y}_t \in \mathcal{Y}$ by using Theorem 2.8.2 in [150]. Next, we derive the bounds on $\Delta_1 u_t^{1*}$ and $\Delta_2 u_t^{1*}$ (their dependence on $(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t)$ is suppressed for simplicity). Let $J_t(u_t^1, y_t^1, y_t^2, u_t^2)$ be the objective function of (2.9) for a fixed $q_t \in [-\beta, \beta]$. By definition,

$$V_t(y_t^1, y_t^2) = \min_{u_t^1, u_t^2} \left\{ J_t(u_t^1, y_t^1, y_t^2, u_t^2) : 0 \leq u_t^1 + y_t^1 \leq \alpha_1, 0 \leq u_t^2 + y_t^2 \leq \alpha_2 \right\}.$$

Next, J_t is minimized sequentially with respect to u_t^2 and u_t^1, respectively. Minimizing J_t over the set of feasible u_t^2 values, and using arguments similar to those in the proof of part (*i*), we obtain

$$\tilde{J}_t(u_t^1, y_t^1, y_t^2) = \min_{u_t^2} \left\{ J_t(u_t^1, y_t^1, y_t^2, u_t^2) : 0 \leq u_t^1 + y_t^1 \leq \alpha_1, 0 \leq u_t^2 + y_t^2 \leq \alpha_2 \right\},$$

which is multimodular in (u_t^1, y_t^1, y_t^2). Next, minimizing $\tilde{J}_t$ over the set of feasible u_t^1 values gives

$$V_t(y_t^1, y_t^2) = \min_{u_t^1} \left\{ \tilde{J}_t(u_t^1, y_t^1, y_t^2) : 0 \leq u_t^1 + y_t^1 \leq \alpha_1 \right\},$$

which is multimodular in (y_t^1, y_t^2). Applying Corollary 1 (*ii*) of [97], we obtain $-1 \leq \Delta_1 u_t^{1*} \leq \Delta_2 u_t^{1*} \leq 0$. Similar bounds on $\Delta_1 u_t^{2*}$ and $\Delta_2 u_t^{2*}$ are obtained by minimizing J_t over u_t^1, followed by u_t^2, and applying Theorem 1 (*ii*) of [97]. □

The multimodularity asserted in Theorem 2.1 (*i*) directly implies result (*ii*), which is often called the *diagonal-dominance* property (cf. [112, 172]) in the inventory literature. Multimodular value functions imply that the storage levels at the load buses are economic substitutes of each other. Theorem 2.1 (*ii*) implies result (*iii*). The monotonicity result in part (*iii*) asserts that, when the storage levels are high, it is more profitable to discharge and sell excess energy to the grid, rather than procuring energy and storing it. However, more insightful is the fact that the optimal storage decisions exhibit bounded sensitivities, as seen in the two inequalities of part (*iii*). That is, for each bus, using the optimal policy, a unit increase in the amount of stored energy yields less than a unit decrease in optimal charge/discharge decision. Furthermore, this marginal decrease is more sensitive to a local increase in the storage level, as opposed to an increase in the storage level at the other bus. The bounded sensitivities property shows that it need not be optimal to fully charge, or fully discharge, the storage devices at each stage, even when $\boldsymbol{\tau}_c, \boldsymbol{\tau}_d > \boldsymbol{\alpha}$. That is, the optimal storage policy is not necessarily of the so-called "bang-bang" type, which is optimal in single-storage models that assume batteries with fast-charging capabilities (cf. [64, 131]). Thus, the bounded sensitivities property is indicative of a stable operating regime for the network and highlights the economic benefit of sharing stored energy under line capacity constraints.

2.3.3 Behavior of the Optimal Operational Cost

Here, we examine the behavior of the optimal operational cost $z_{\boldsymbol{\alpha}}$. This examination is motivated by the operator's desire to determine the appropriate storage capacity at each bus prior to making any operational decisions. This determination is further warranted by the significant costs associated with storage investment in distribution networks. The operator's storage allocation problem is formulated as follows:

$$\min_{\boldsymbol{\alpha}} \ \kappa \sum_{i \in C'} \alpha_i + z_{\boldsymbol{\alpha}}, \tag{2.26a}$$

$$\text{s.t.} \ \mathbf{0} \leq \boldsymbol{\alpha} \leq \bar{\boldsymbol{\alpha}}, \tag{2.26b}$$

where κ is the per-unit cost of storage capacity, and $\bar{\boldsymbol{\alpha}}$ is a budget vector of the maximum storage capacity allowed at each bus. Proposition 2.5 asserts that the optimal operational cost is convex and monotone decreasing in the storage capacities.

Proposition 2.5. *The optimal operational cost $z_{\boldsymbol{\alpha}}$ is convex and monotone decreasing in $\boldsymbol{\alpha}$.*

Proof. Consider three storage capacity vectors $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2$ and $\boldsymbol{\alpha}_3$, such that $\boldsymbol{\alpha}_1 < \boldsymbol{\alpha}_2 < \boldsymbol{\alpha}_3$ and $\boldsymbol{\alpha}_2 = \eta\boldsymbol{\alpha}_1 + (1-\eta)\boldsymbol{\alpha}_3$, where $\eta \in [0,1]$. We seek to show that $z_{\boldsymbol{\alpha}_2} \leq \eta z_{\boldsymbol{\alpha}_1} + (1-\eta) z_{\boldsymbol{\alpha}_3}$. Consider a state $\boldsymbol{s}_t = (\boldsymbol{w}_t, \boldsymbol{y}_t) \in \mathcal{S}_t$. Let $\boldsymbol{x}_t^{k*} = (\boldsymbol{u}_t^{k*}, q_t^{k*})$ be the optimal solution vector of (2.8) for $\boldsymbol{\alpha} = \boldsymbol{\alpha}_k$, where $k \in \{1,2,3\}$. Construct a policy $\tilde{\boldsymbol{\pi}} = (\tilde{\boldsymbol{x}}_t : t \in T')$, such that $\tilde{\boldsymbol{u}}_t = \eta\boldsymbol{u}_t^{1*} + (1-\eta)\boldsymbol{u}_t^{3*}$ and $\tilde{q}_t = \eta q_t^{1*} + (1-\eta) q_t^{3*}$. Multiply both sides of the constraints in $\mathcal{X}_t(\boldsymbol{y}_t;\boldsymbol{\alpha}_1)$ by η and those in $\mathcal{X}_t(\boldsymbol{y}_t;\boldsymbol{\alpha}_3)$ by $(1-\eta)$, and add the corresponding constraints to obtain the inequalities

$$\begin{aligned} -\min\{\boldsymbol{\tau}_d, \boldsymbol{y}_t\} \leq \eta\boldsymbol{u}_t^{1*} + (1-\eta)\boldsymbol{u}_t^{3*} &\leq \min\{\boldsymbol{\tau}_c, \boldsymbol{\alpha}_2 - \boldsymbol{y}_t\}, \\ -\beta \leq \eta q_t^{1*} + (1-\eta) q_t^{3*} &\leq \beta, \end{aligned}$$

which shows that $\tilde{\boldsymbol{\pi}}$ is a feasible, but not necessarily optimal, policy of (2.8) for $\boldsymbol{\alpha} = \boldsymbol{\alpha}_2$. Furthermore, it can be verified that

$$c_t(\boldsymbol{s}_t, \tilde{\boldsymbol{x}}_t) = \eta c_t(\boldsymbol{s}_t, \boldsymbol{x}_t^{1*}) + (1-\eta) c_t(\boldsymbol{s}_t, \boldsymbol{x}_t^{3*}), \quad t \in T'. \tag{2.27}$$

Summing the one-step costs in (2.27) and taking the expectation of this sum gives

$$z_{\boldsymbol{\alpha}_2} \leq \mathbb{E}_{\tilde{\boldsymbol{\pi}}}\left(\sum_{t\in T'} c_t(\boldsymbol{s}_t, \tilde{\boldsymbol{x}}_t)\right) = \eta z_{\boldsymbol{\alpha}_1} + (1-\eta) z_{\boldsymbol{\alpha}_3},$$

which completes the proof of convexity.

Next, to show monotonicity, consider two storage capacity vectors $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$, such that $\boldsymbol{\alpha}_2 = \boldsymbol{\alpha}_1 + \boldsymbol{\Gamma}$ where $\boldsymbol{\Gamma} > \mathbf{0}$. Let $\boldsymbol{w} = (\boldsymbol{w}_t : t \in T)$ be a realization of the exogenous process, and $\boldsymbol{\pi}_1$ and $\boldsymbol{\pi}_2$ denote the optimal policies of (2.8) for $\boldsymbol{\alpha} = \boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha} = \boldsymbol{\alpha}_2$, respectively. Let $z_{\boldsymbol{\alpha}_1}(\boldsymbol{w})$ and $z_{\boldsymbol{\alpha}_2}(\boldsymbol{w})$ be the total costs incurred by along the trajectory $\boldsymbol{w}$ using $\boldsymbol{\pi}_1$ and $\boldsymbol{\pi}_2$, respectively. Without loss of generality, suppose $\boldsymbol{y}_1 = \mathbf{0}$ for both policies. Let $\boldsymbol{s}_t^1 = (\boldsymbol{w}_t, \boldsymbol{y}_t^1)$ be the state of the process at stage t under $\boldsymbol{\pi}_1$. Then, $\boldsymbol{\pi}_1$ is a feasible policy of (2.8) when

$\boldsymbol{\alpha} = \boldsymbol{\alpha}_2$, as $\boldsymbol{u}_t^1 \leq \boldsymbol{\alpha}_1 - \boldsymbol{y}_t^1 \Rightarrow \boldsymbol{u}_t^1 \leq \boldsymbol{\alpha}_1 + \boldsymbol{\Gamma} - \boldsymbol{y}_t^1 = \boldsymbol{\alpha}_2 - \boldsymbol{y}_t^1$ as $\boldsymbol{\Gamma} > \boldsymbol{0}$ and $q_t^1 \in [-\beta, \beta]$; therefore, $z_{\boldsymbol{\alpha}_1}(\boldsymbol{w}) \geq z_{\boldsymbol{\alpha}_2}(\boldsymbol{w})$. As $\boldsymbol{w}$ is any feasible realization in $\mathcal{W}$, we can conclude that

$$z_{\boldsymbol{\alpha}_1} = \sum_{\boldsymbol{w} \in \mathcal{W}} z_{\boldsymbol{\alpha}_1}(\boldsymbol{w}) \mathbb{P}(\boldsymbol{W} = \boldsymbol{w}) \geq \sum_{\boldsymbol{w} \in \mathcal{W}} z_{\boldsymbol{\alpha}_2}(\boldsymbol{w}) \mathbb{P}(\boldsymbol{W} = \boldsymbol{w}) = z_{\boldsymbol{\alpha}_2},$$

which completes the proof. □

It is seen that additional storage capacity leads to lower costs, but with decreasing marginal benefit.

Next, we compare the operational cost of the 2-bus network to those of two comparable networks having distinct storage configurations. The 2-bus network has a *coupled storage* (CS) configuration, in which the two buses can transmit stored energy between them. By contrast, energy cannot be transmitted between the buses in a *decentralized storage* (DS) setting; thus, there is no interaction between the buses, and the operational cost is the sum of the operational costs incurred at each of the buses. Finally, a *pooled storage* (PS) configuration consists of a centralized storage facility that satisfies the collective energy demand in the network. Figure 3 depicts these three networks and their storage configurations. It is assumed that the total storage capacity in each network is equal to $\alpha_1 + \alpha_2$. Additionally, it is assumed that both the charging and the discharging efficiencies in each network are equal to ρ for a fixed $\rho \in (0, 1]$.

Theorem 2.2 asserts that the network with pooled storage has the lowest operational cost, followed by the one with coupled storage, which in turn is less than the cost in the decentralized storage network. For ease of exposition, let z_P, z_C and z_D denote the optimal operational costs of the PS, CS and DS network configurations, respectively. The next result shows how these costs compare to one another.

Theorem 2.2. *The optimal operational costs of PS, CS and DS configurations are ordered such that $z_P \leq z_C \leq z_D$.*

Proof. As $q_t = 0 \in [-\beta, \beta]$ at each stage $t \in T'$ in the DS network, it is clear that any optimal policy for the DS network is a feasible, but not necessarily optimal, policy for the CS network; therefore, we have $z_C \leq z_D$. Next, we show that $z_P \leq z_C$. Assume that the

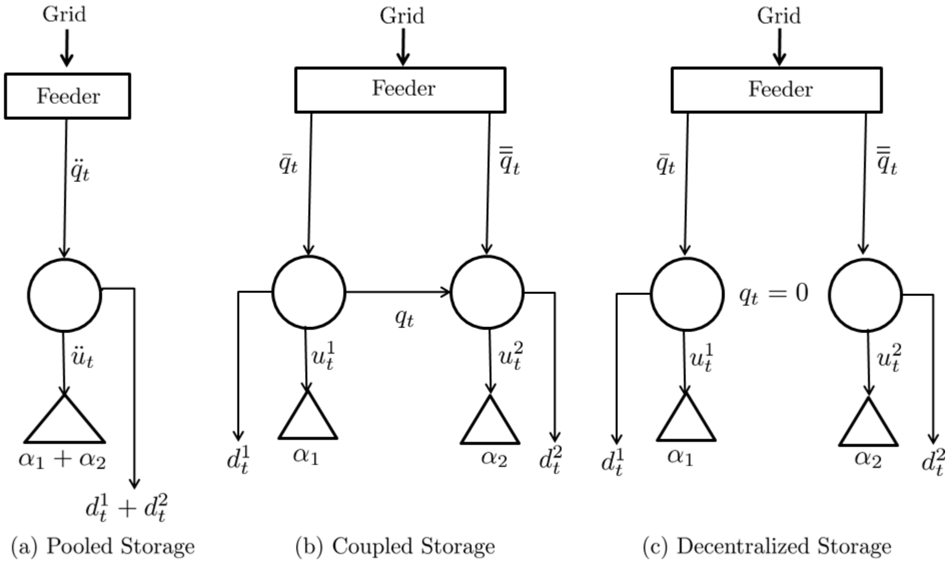


Figure 3: The pooled, coupled and decentralized storage configurations.

initial storage levels in the CS and PS networks are zero without loss of generality. Let $\boldsymbol{u}_t^* = (u_t^{1*}, u_t^{2*})$ be the optimal charge/discharge decisions in the CS network at stage t. Construct a storage policy for the PS network, denoted by $\ddot{\boldsymbol{\pi}} = (\ddot{u}_t : t \in T')$, such that $\ddot{u}_t = u_t^{1*} + u_t^{2*}$ for each $t \in T'$. Let $\ddot{y}_t$ be the associated storage level realized at stage t under policy $\ddot{\boldsymbol{\pi}}$, where $\ddot{y}_{t+1} = \ddot{y}_t + \ddot{u}_t$. Then, starting from $\ddot{y}_t = 0$, it is easy to verify that $0 \leq \ddot{y}_t + \ddot{u}_t \leq \alpha_1 + \alpha_2$ for each $t \in T'$, which implies that $\ddot{\boldsymbol{\pi}}$ is a feasible policy for the PS network. Next, define the variables $(\ddot{\theta}_t : t \in T')$, such that

$$\ddot{\theta}_t = \begin{cases} 1/\rho, & \ddot{u}_t \geq 0, \\ \rho, & \ddot{u}_t < 0. \end{cases}$$

Consider a realization $\boldsymbol{w} = (\boldsymbol{w}_t : t \in T)$ of the exogenous process $\boldsymbol{W}$. The one-step cost incurred at stage t in the PS network, when the state $(\boldsymbol{w}_t, \ddot{y}_t)$ is realized, is

$$\ddot{c}_t(\boldsymbol{w}_t, \ddot{u}_t) = p_t(d_t^1 + d_t^2 + \ddot{\theta}_t \ddot{u}_t) + \varphi|\ddot{u}_t|,$$

while the corresponding one-step cost in the CS network for state $(\boldsymbol{w}_t, \boldsymbol{y}_t^*)$ is

$$c_t(\boldsymbol{w}_t, \boldsymbol{u}_t^*, q_t^*) = p_t(d_t^1 + d_t^2 + \theta_t^1 u_t^{1*} + \theta_t^2 u_t^{2*}) + \varphi(|u_t^{1*}| + |u_t^{2*}|) + \nu\xi(\boldsymbol{u}_t^*, q_t^*).$$

Define $\ddot{a}_t \equiv \ddot{\theta}_t \ddot{u}_t$ and $a_t \equiv \theta_t^1 u_t^{1*} + \theta_t^2 u_t^{2*}$. Next, compare the terms $\ddot{c}_t(\boldsymbol{w}_t, \ddot{u}_t)$ and $c_t(\boldsymbol{w}_t, \boldsymbol{u}_t^*, q_t^*)$. To this end, consider the following six cases involving u_t^{1*}, u_t^{2*} and $\ddot{u}_t \equiv u_t^{1*} + u_t^{2*}$:

Case 1: $u_t^{1*} \geq 0$, $u_t^{2*} \geq 0$ and $u_t^{1*} + u_t^{2*} \geq 0$. Then $\theta_t^1 = \theta_t^2 = \ddot{\theta}_t = 1/\rho$, and $a_t = \ddot{a}_t = (u_t^{1*} + u_t^{2*})/\rho$.

Case 2: $u_t^{1*} \geq 0$, $u_t^{2*} < 0$ and $u_t^{1*} + u_t^{2*} \geq 0$. Then $\theta_t^1 = \ddot{\theta}_t = 1/\rho$, $\theta_t^2 = \rho$, and $a_t = u_t^{1*}/\rho + \rho u_t^{2*} \geq (u_t^{1*} + u_t^{2*})/\rho = \ddot{a}_t$.

Case 3: $u_t^{1*} < 0$, $u_t^{2*} \geq 0$ and $u_t^{1*} + u_t^{2*} \geq 0$. Then $\theta_t^1 = \rho$, $\theta_t^2 = \ddot{\theta}_t = 1/\rho$, and $a_t = \rho u_t^{1*} + u_t^{2*}/\rho \geq (u_t^{1*} + u_t^{2*})/\rho = \ddot{a}_t$.

Case 4: $u_t^{1*} < 0$, $u_t^{2*} < 0$ and $u_t^{1*} + u_t^{2*} < 0$. Then $\theta_t^1 = \theta_t^2 = \ddot{\theta}_t = \rho$, and $a_t = \ddot{a}_t = \rho(u_t^{1*} + u_t^{2*})$.

Case 5: $u_t^{1*} < 0$, $u_t^{2*} \geq 0$ and $u_t^{1*} + u_t^{2*} < 0$. Then $\theta_t^1 = \ddot{\theta}_t = \rho$, $\theta_t^2 = 1/\rho$, and $a_t = \rho u_t^{1*} + u_t^{2*}/\rho \geq \rho(u_t^{1*} + u_t^{2*}) = \ddot{a}_t$.

Case 6: $u_t^{1*} \geq 0$, $u_t^{2*} < 0$ and $u_t^{1*} + u_t^{2*} < 0$. Then $\theta_t^1 = 1/\rho$, $\theta_t^2 = \ddot{\theta}_t = \rho$, and $a_t = u_t^{1*}/\rho + \rho u_t^{2*} \geq \rho(u_t^{1*} + u_t^{2*}) = \ddot{a}_t$.

Clearly, $a_t \geq \ddot{a}_t \Rightarrow \theta_t^1 u_t^{1*} + \theta_t^2 u_t^{2*} \geq \ddot{\theta}_t \ddot{u}_t$ in all of the above cases. Also, $|\ddot{u}_t| = |u_t^{1*} + u_t^{2*}| \leq |u_t^{1*}| + |u_t^{2*}|$ by the triangle inequality, and $\nu\xi(u_t^*, q_t^*) \geq 0$ by definition. Therefore, $c_t(\boldsymbol{w}_t, \boldsymbol{u}_t^*, q_t^*) \geq \ddot{c}_t(\boldsymbol{w}_t, \ddot{u}_t)$ for each $t \in T'$. Adding the one-step costs over the decision stages and taking expectation of these sums, we obtain

$$\mathbb{E}_{\ddot{\boldsymbol{\pi}}}\left(\sum_{t \in T'} \ddot{c}_t(\boldsymbol{w}_t, \ddot{u}_t)\right) \leq \mathbb{E}_{\boldsymbol{\pi}^*}\left(\sum_{t \in T'} c_t(\boldsymbol{w}_t, \boldsymbol{u}_t^*, q_t^*)\right) = z_C. \tag{2.28}$$

However, the l.h.s. of (2.28) is greater than or equal to z_P because $\ddot{\boldsymbol{\pi}}$ is a feasible, but not necessarily optimal, policy for the PS network. This concludes the proof. □

To the authors' knowledge, Theorem 2.2 is the first result to establish theoretical bounds on the optimal operational cost (z_C) incurred in a 2-bus distribution network with storage; however, the upper and lower bounds of z_C (z_D and z_P, respectively) need not be tight in general. Specifically, these bounds are the optimal costs of simplified, single-storage models that do not account for energy flow constraints in a 2-bus network. It is well-known that optimal storage policies for single-storage models exhibit a dual-threshold structure (cf. [64, 84, 132, 153, 160]), allowing such models to be solved efficiently using specialized backward induction algorithms (see Section 4.7.6 in [122]). It is instructive that the quantity $z_C - z_P$ represents the cost savings achieved by pooling stored energy in a centralized facility, while the quantity $z_D - z_C$ can be interpreted as the opportunity cost of prohibiting the transmission of stored energy between the buses. Consequently, the ratio $(z_D - z_P)/z_D$ can be viewed as the marginal benefit of centralizing the storage operations of two decentralized storage systems.

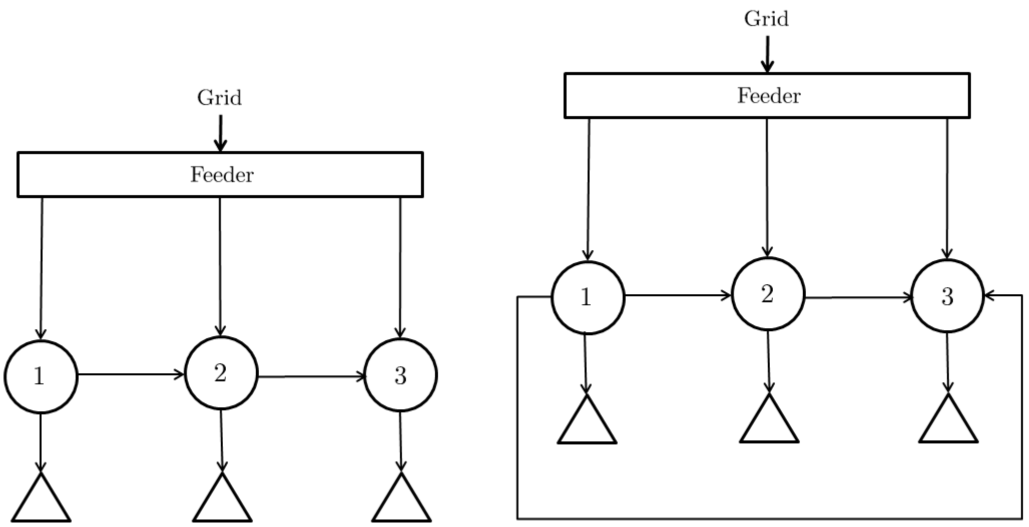
2.4 EXTENSION TO MULTI-BUS NETWORKS

In this section, we extend the results of Theorems 2.1 and 2.2 to networks with more than two buses. Specifically, we consider *loop* and *mesh* network configurations, which are common for distribution networks [28]. Figure 4 depicts these two configurations, which differ primarily in the number of line connections between the buses. The mesh configuration is a fully-connected network topology in which each pair of buses is connected by a line. By contrast, a loop network is a simply connected network.

Let G denote the total number of load buses in the network, and let $C' \equiv \{1, \ldots, G\}$ denote the set of such buses. The set of lines in the loop and the mesh networks are denoted by A_L and A_M, respectively. For both networks, let $\bar{q}_t^i$ be the energy flow in the feeder line connected to bus i, and let $q_t(i,j)$ denote the energy flow in line (i,j), where $q_t(i,j) \geq 0$ if energy flows from bus i to bus j, and $q_t(i,j) < 0$ otherwise. For notational convenience, denote $q_t(G, G+1) \equiv q_t(G, 1)$. For each $t \in T'$ and $i \in C'$, the supply-demand balance constraints in these two multi-bus networks are

$$\begin{aligned}
&\text{(Loop)} \quad \bar{q}_t^i = d_t^i + \theta_t^i u_t^i + q_t(i, i+1) - q_t(i-1, i), \\
&\text{(Mesh)} \quad \bar{q}_t^i = d_t^i + \theta_t^i u_t^i + \sum_{(i,j) \in A_M : j > i} q_t(i,j) - \sum_{(j,i) \in A_M : j < i} q_t(j,i).
\end{aligned}$$

Let $\boldsymbol{q}_t^L = (q_t(i,j) : (i,j) \in A_L)$ and $\boldsymbol{q}_t^M = (q_t(i,j) : (i,j) \in A_M)$ be the vector of energy flows between the load buses in the loop and the mesh networks, respectively. Then, one-step



(a) Loop network.

(b) Mesh network.

Figure 4: Depiction of networks with the loop and the mesh configurations.

costs incurred at stage t in the two multi-bus networks are

$$\text{(Loop)} \quad c_t^L(\boldsymbol{s}_t, \boldsymbol{x}_t) = p_t \sum_{i \in C'} (d_t^i + \theta_t^i u_t^i) + \varphi \sum_{i \in C'} |u_t^i| + \nu \xi(\boldsymbol{u}_t, \boldsymbol{q}_t^L),$$

$$\text{(Mesh)} \quad c_t^M(\boldsymbol{s}_t, \boldsymbol{x}_t) = p_t \sum_{i \in C'} (d_t^i + \theta_t^i u_t^i) + \varphi \sum_{i \in C'} |u_t^i| + \nu \xi(\boldsymbol{u}_t, \boldsymbol{q}_t^M).$$

The functions c_t^L and c_t^M are of the same form as that of c_t in (2.7). Moreover, the storage level and line capacity constraints in the multi-bus networks mirror those in the 2-bus network; hence, the lattice structure of the feasibility sets is conserved, despite the fact that the number of constraints is significantly higher for the multi-bus configurations. This leads us to the next result in Theorem 2.3 that holds for both the multi-bus networks and is stated without proof.

Theorem 2.3. *For each $t \in T$,*

(i) $V_t(\bar{\boldsymbol{w}}_t, \boldsymbol{y}_t)$ is multimodular in $\boldsymbol{y}_t \in \mathcal{Y}$.

(ii) $\boldsymbol{x}_t^*$ is monotone decreasing in $\boldsymbol{y}_t \in \mathcal{Y}$.

The next result, Theorem 2.4, establishes a sequence of bounds involving the optimal costs of the pooled (z_P), decentralized (z_D), loop (z_L) and mesh (z_M) networks.

Theorem 2.4. *The optimal operational costs are ordered such that* $z_P \leq z_M \leq z_L \leq z_D$.

Proof. Note that, for the DS network, $q_t(i,j) = 0$ for all $(i,j) \in A_L$; therefore, it is clear that any optimal policy for the DS network is a feasible, but not necessarily optimal, policy for the loop network. Therefore, $z_L \leq z_D$. Similarly, $q_t(i,j) = 0$ for all $(i,j) \in A_M \setminus A_L$ in the loop network. Using a similar feasibility-optimality argument for the optimal costs of the loop and mesh networks, we conclude that $z_M \leq z_L$. Next, we show that $z_P \leq z_M$. Assume that the initial storage levels in the mesh and PS networks are zero without loss of generality. Let $\boldsymbol{u}_t^* = (u_t^{i*} : i \in C')$ be the optimal charge/discharge decisions in the mesh network at stage t. Construct a storage policy for the PS network, denoted by $\ddot{\boldsymbol{\pi}} = (\ddot{u}_t : t \in T')$, such that $\ddot{u}_t = \sum_{i \in C'} u_t^{i*}$ for each $t \in T'$. Let $\ddot{y}_t$ be the storage level realized at stage t under policy $\ddot{\boldsymbol{\pi}}$, where $\ddot{y}_{t+1} = \ddot{y}_t + \ddot{u}_t$. Then, starting from $\ddot{y}_t = 0$, it is easy to verify that $0 \leq \ddot{y}_t + \ddot{u}_t \leq \sum_{i \in C'} \alpha_i$, which implies that $\ddot{\boldsymbol{\pi}}$ is a feasible policy for the PS network. Next, define the variables $(\ddot{\theta}_t : t \in T')$, such that

$$\ddot{\theta}_t = \begin{cases} 1/\rho, & \ddot{u}_t \geq 0, \\ \rho, & \ddot{u}_t < 0. \end{cases}$$

Next, we show that $\sum_{i \in C'} \theta_t^i u_t^{i*} \geq \ddot{\theta}_t \ddot{u}_t$ by using a simple induction argument. For $k \in C' \setminus \{1\}$, define $\tilde{u}_t^k \equiv \sum_{i=1}^k u_t^{i*}$, and let $\tilde{\theta}_t^k = 1/\rho$ if $\tilde{u}_t^k \geq 0$, and $\tilde{\theta}_t^k = \rho$ otherwise. From the proof of Theorem 2.2, we know that $\sum_{i=1}^k \theta_t^i u_t^{i*} \geq \tilde{\theta}_t^k \tilde{u}_t^k$ for $k = 2$. For the induction hypothesis, suppose $\sum_{i=1}^k \theta_t^i u_t^{i*} \geq \tilde{\theta}_t^k \tilde{u}_t^k$ for some $k > 2$. Adding the term $\theta_t^{k+1} u_t^{(k+1)*}$ to both sides, we obtain $\sum_{i=1}^{k+1} \theta_t^i u_t^{i*} \geq \tilde{\theta}_t^k \tilde{u}_t^k + \theta_t^{k+1} u_t^{(k+1)*}$. Define $a_t \equiv \tilde{\theta}_t^k \tilde{u}_t^k + \theta_t^{k+1} u_t^{(k+1)*}$ and $\tilde{a}_t = \tilde{\theta}_t^{k+1} \tilde{u}_t^{k+1}$. Next, we compare the terms a_t and $\tilde{a}_t$ for the following six cases involving the terms $\tilde{u}_t^k, \tilde{u}_t^{k+1}$ and $u_t^{(k+1)*}$:

Case 1: $\tilde{u}_t^k \geq 0$, $u_t^{(k+1)*} \geq 0$ and $\tilde{u}_t^{k+1} \geq 0$. Then $\tilde{\theta}_t^k = \tilde{\theta}_t^{k+1} = \theta_t^{(k+1)*} = 1/\rho$, and $a_t = \tilde{a}_t = \tilde{u}_t^{k+1}/\rho$.

Case 2: $\tilde{u}_t^k \geq 0$, $u_t^{(k+1)*} < 0$, and $\tilde{u}_t^{k+1} \geq 0$. Then $\tilde{\theta}_t^k = \tilde{\theta}_t^{k+1} = 1/\rho$, $\theta_t^{(k+1)*} = \rho$ and $a_t = \tilde{u}_t^k/\rho + \rho u_t^{(k+1)*} \geq \tilde{u}_t^{k+1}/\rho = \tilde{a}_t$.

Case 3: $\tilde{u}_t^k < 0$, $u_t^{(k+1)*} \geq 0$, and $\tilde{u}_t^{k+1} \geq 0$. Then $\tilde{\theta}_t^k = \rho$, $\theta_t^{(k+1)*} = \tilde{\theta}_t^{k+1} = 1/\rho$ and $a_t = \rho\tilde{u}_t^k + u_t^{(k+1)*}/\rho \geq \tilde{u}_t^{k+1}/\rho = \tilde{a}_t$.

Case 4: $\tilde{u}_t^k < 0$, $u_t^{(k+1)*} < 0$, and $\tilde{u}_t^{k+1} < 0$. Then $\tilde{\theta}_t^k = \theta_t^{(k+1)*} = \tilde{\theta}_t^{k+1} = \rho$ and $a_t = \tilde{a}_t = \rho\tilde{u}_t^{k+1}$.

Case 5: $\tilde{u}_t^k < 0$, $u_t^{(k+1)*} \geq 0$, and $\tilde{u}_t^{k+1} < 0$. Then $\tilde{\theta}_t^k = \tilde{\theta}_t^{k+1} = \rho$, $\theta_t^{(k+1)*} = 1/\rho$ and $a_t = \rho\tilde{u}_t^k + u_t^{(k+1)*}/\rho \geq \rho\tilde{u}_t^{k+1} = \tilde{a}_t$.

Case 6: $\tilde{u}_t^k \geq 0$, $u_t^{(k+1)*} < 0$, and $\tilde{u}_t^{k+1} < 0$. Then $\tilde{\theta}_t^k = 1/\rho$, $\theta_t^{(k+1)*} = \tilde{\theta}_t^{k+1} = \rho$ and $a_t = \tilde{u}_t^k/\rho + \rho u_t^{(k+1)*} \geq \rho\tilde{u}_t^{k+1} = \tilde{a}_t$.

Clearly, $a_t \geq \tilde{a}_t \Rightarrow \tilde{\theta}_t^k\tilde{u}_t^k + \theta_t^{k+1}u_t^{(k+1)*} \geq \tilde{\theta}_t^{k+1}\tilde{u}_t^{k+1}$. Finally, by the induction hypothesis, we obtain $\sum_{i=1}^{k+1} \theta_t^i u_t^{i*} \geq \tilde{\theta}_t^{k+1}\tilde{u}_t^{k+1}$, which proves that our induction hypothesis is true. For $k = M$, this is equivalent to $\sum_{i \in C'} \theta_t^i u_t^{i*} \geq \ddot{\theta}_t\ddot{u}_t$. The rest of the proof is similar to that of Theorem 2.2, from which we conclude that $z_D \leq z_M$. □

2.5 NUMERICAL EXAMPLES

In this section, we present numerical examples to illustrate the structural properties of the value function (V_t), the optimal policy ($\boldsymbol{\pi}^*$) and the optimal operational cost ($z_{\boldsymbol{\alpha}}$) for a 2-bus network using real renewable generation and pricing data. Before presenting these examples, the source data, solution methodology and computational study are described in greater detail.

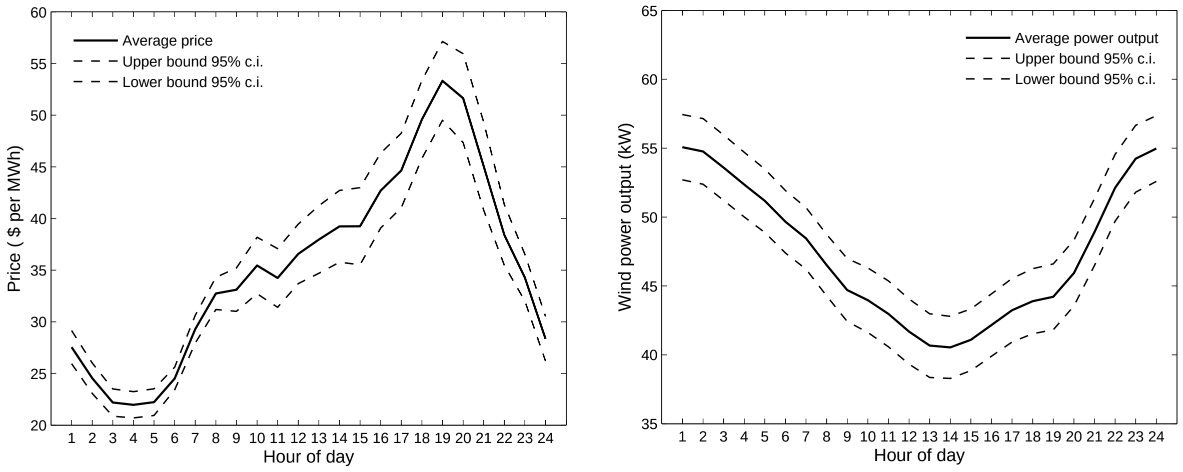
2.5.1 Data Description

Hourly wind speed and real-time electricity pricing data for calendar year 2012 were obtained from the NREL (National Renewable Energy Laboratory; http://www.nrel.gov) and PJM (Pennsylvania-Jersey-Maryland Interconnection; http://www.pjm.com), respectively. Let v_t and P_t be the wind speed and price in hour $t \in \{1, \ldots, 24\}$, respectively. Due to seasonality effects, we partitioned both data sets into 24 segments, each one hour in duration, and fit separate probability density functions to each segment. The hourly prices were fit using truncated normal (TN) distributions of the form $P_t \sim TN(\hat{p}_t, \hat{\sigma}_t^2)$, where $\hat{p}_t$ and $\hat{\sigma}_t^2$ are the (estimated) mean and variance of the price level in hour t, respectively. As was done in [29, 133], we fit the hourly wind speeds using Weibull distributions, i.e., $v_t \sim \text{Weibull}\,(\hat{\ell}_t, \hat{n}_t)$, where $\hat{\ell}_t$ and $\hat{n}_t$ are the (estimated) shape and scale parameters, respectively. Each of the distribution parameters were estimated from the real data using maximum likelihood estimation (MLE), and the values are presented in Table 1. Next, we determined the wind generation levels at the two buses. Let R_t^i denote the wind generation in hour t at bus i. It was assumed that the Evance R9000 wind turbine models is installed at both buses. The turbine at bus 1 has a power rating of $\bar{R}^1 = 50$ kW, while the turbine at bus 2 has a power rating of $\bar{R}^2 = 25$ kW. Both turbines have a cut-in speed of $v_c = 3$ meter per second (m/s), a cut-off speed of $v_f = 60$ m/s, and a rated wind speed of $\bar{v} = 12$ m/s. The following deterministic model (see page 547 of [102]) was used to compute the hourly wind generation level at each bus i:

$$R_t^i = \begin{cases} \bar{R}^i \left(\frac{\bar{v}-v_t}{\bar{v}-v_c}\right), & v_c \leq v_t \leq \bar{v}, \\ \bar{R}^i, & \bar{v} \leq v_t \leq v_f, \\ 0, & \text{otherwise.} \end{cases}$$

For the analysis that follows, P_t, R_t^1, and R_t^2 are assumed to be mutually independent random variables. Figure 5 depicts the average hourly wind generation at bus 1 and price levels (and associated 95% confidence intervals) for a 24-hour period. In this figure, hour 1 is midnight to 0100, hour 2 is 0100–0200, hour 3 is 0200–0300, and so forth. Examining Figure 5(a) closely, it is seen that the evening hours (hours 17 to 21) are the peak-price periods, while the off-peak price periods span the late night and early morning hours (hours 1 to 7). The variability in the hourly prices exhibits a similar trend. By contrast, as seen in Figure 5(b), wind power output is highest during the late night and early morning hours and is lowest in the afternoon (hours 12 to 16).

Next, we impose assumptions about the wind generation levels at the two buses. Let $\mathcal{R}_t^1$, $\mathcal{R}_t^2$ and $\mathcal{P}_t$ denote the (bounded) supports of R_t^1, R_t^2 and P_t, respectively. The lower and upper limits of these sets correspond to their respective minimum and maximum values observed during 2012. In order to numerically compute the optimal policy, we assume finite



(a) Real-time hourly electricity prices.

(b) Hourly wind generation levels at bus 1.

Figure 5: Average price and wind generation levels in the year 2012.

Table 1: MLE estimates of the wind energy and price distribution parameters.

t	1	2	3	4	5	6	7	8	9	10	11	12
$\hat{\ell}_t$	4.51	4.49	4.41	4.36	4.33	4.29	4.22	4.08	3.81	3.61	3.56	3.59
$\hat{n}_t$	2.09	2.08	2.06	2.02	1.99	1.96	1.89	1.79	1.59	1.46	1.43	1.43
$\hat{p}_t$	27.26	25.65	23.48	23.39	23.75	25.65	28.47	32.46	33.67	36.82	34.95	37.64
$\hat{\sigma}_t^2$	6.50	9.45	10.42	12.91	14.56	15.18	15.43	18.21	19.36	21.59	20.07	24.31

t	13	14	15	16	17	18	19	20	21	22	23	24
$\hat{\ell}_t$	3.62	3.71	3.78	3.85	3.87	3.83	3.78	3.86	4.07	4.31	4.46	4.52
$\hat{n}_t$	1.43	1.44	1.67	1.49	1.50	1.52	1.55	1.59	1.72	1.87	1.98	2.03
$\hat{p}_t$	38.54	39.94	40.67	41.52	44.85	46.64	50.33	52.78	51.08	38.85	34.71	28.25
$\hat{\sigma}_t^2$	25.74	26.25	28.53	28.21	29.03	30.33	32.74	36.59	31.28	26.65	18.19	14.09

supports for the exogenous variables. Theorem 6.10.11 of [122] provides an error bound for finite-state approximations to countable-state MDP models. These supports were therefore discretized as follows: $\mathcal{R}_t^1 = \mathcal{R}_t^2 = \{0, 1, \ldots, 9\}$, and $\mathcal{P}_t = \{5n : n = 0, 1, \ldots, 12\}$. Let ϕ_t, ϑ_t^1 and ϑ_t^2 be the probability density functions of P_t, R_t^1 and R_t^2, respectively. For ease of computation, we fix the hourly demand levels at their mean values (obtained from PJM demand data). Therefore, the (random) exogenous state $\boldsymbol{W}_t$ consists of the price and wind generation levels only, i.e., $\boldsymbol{W}_t = (P_t, R_t^1, R_t^2)$, so that $\mathcal{W}_t = \mathcal{P}_t \times \mathcal{R}_t^1 \times \mathcal{R}_t^2$. Let g_t be the joint probability mass function of $\boldsymbol{W}_t$. Then, due to the independence assumption, the exogenous process transitions from state $\boldsymbol{w}_{t-1} \in \mathcal{W}_{t-1}$ to another state $\boldsymbol{w}_t = (p, r_1, r_2) \in \mathcal{W}_t$ with probability

$$\mathbb{P}_{t-1}(\boldsymbol{w}_t|\boldsymbol{w}_{t-1}) = g_t(\boldsymbol{w}_t) = \frac{\phi_t(p)}{\sum\limits_{\tilde{p}\in\mathcal{P}_t} \phi_t(\tilde{p})} \times \frac{\vartheta_t^1(r_1)}{\sum\limits_{\tilde{r}_1\in\mathcal{R}_t^1} \vartheta_t^1(\tilde{r}_1)} \times \frac{\vartheta_t^2(r_2)}{\sum\limits_{\tilde{r}_2\in\mathcal{R}_t^2} \vartheta_t^2(\tilde{r}_2)}. \tag{2.29}$$

For the problem instances that follow, we used the parameter values listed in Table 2. It was assumed that both buses have identical energy storage parameters $\alpha_i, \rho_c^i, \rho_d^i, \tau_c^i$ and τ_d^i. Moreover, the storage devices were assumed to have a shelf-life exceeding one year.

Table 2: Summary of parameter values for the problem instances.

Parameters	Parameter descriptions	Value(s)
(α_1, α_2)	Storage capacities at buses 1 and 2 (in kW-h)	(10,10)
(τ_c^i, τ_d^i)	Storage charging and discharging rates at buses 1 and 2 (in kW)	(4,4)
(ρ_c^i, ρ_d^i)	Storage charging and discharging efficiencies at buses 1 and 2	(0.90,0.85)
β	Line capacity (in kW-h)	3.5
φ	Per-unit cost of charging or discharging energy ($ per kW-h)	10
ν	Per-unit cost of line losses ($ per kW-h)	20

The parameter φ can be viewed as the implicit cost of degradation per unit of energy charged or discharged from the battery and can be determined using a life-cycle cost analysis that accounts for several factors affecting battery performance, such as temperature, state-of-charge profile, and depth-of-discharge limits (see [11, 71, 72] below). Similarly, the quantity ν can be estimated by using a life-cycle cost model (e.g., Equation (6) of [169]), which uses the resistance per unit length of the power lines, per unit electricity prices, and the maximum allowable current in the power lines.

2.5.2 Solving the MDP Model

For the computational experiments, we considered a 24-hour (or 25-stage) planning horizon, i.e., $T = \{1, \dots, 25\}$, in January 2012. It is assumed that the decisions are made at the start of each hour (or stage). Moreover, the state space in each stage was assumed to be time invariant, i.e., $\mathcal{S}_t = \mathcal{S}$ for all $t \in T$. We discretized the storage levels, Y_t^1 and Y_t^2, to have support $\mathcal{Y} = \{0, 1, \dots, 10\} \times \{0, 1, \dots, 10\}$. Hence, there are $13 \times 10^2 \times 11^2 = 157,300$

possible states in each stage. We employ the linear programming (LP) approach devised in [16] to solve the non-stationary, finite-horizon model (2.8) and begin by introducing its primal LP formulation. For notational convenience, denote $\mathcal{X}_t(\boldsymbol{s})$ simply as $\mathcal{X}_t$. Let $(\lambda_t(\boldsymbol{s}) : t \in T', \boldsymbol{s} \in \mathcal{S})$ be the vector of primal LP variables, and $(\gamma_t(\boldsymbol{s}) : t \in T', \boldsymbol{s} \in \mathcal{S})$ be the vector of cost coefficients such that $\gamma_t(\boldsymbol{s}) \in (0, \infty)$ for each $t \in T'$ and $\boldsymbol{s} \in \mathcal{S}$. The primal LP formulation is

$$\max \quad \sum_{t \in T'} \sum_{\boldsymbol{s} \in \mathcal{S}} \gamma_t(\boldsymbol{s}) \lambda_t(\boldsymbol{s}) \tag{2.30a}$$

$$\text{s.t.} \quad \lambda_t(\boldsymbol{s}) \leq c_t(\boldsymbol{s}, \boldsymbol{x}) + \delta \sum_{\boldsymbol{s}' \in \mathcal{S}} \mathbb{P}_t(\boldsymbol{s}' | \boldsymbol{s}, \boldsymbol{x}) \lambda_{t+1}(\boldsymbol{s}'), \quad \forall t \in T',\, \boldsymbol{s} \in \mathcal{S},\, \boldsymbol{x} \in \mathcal{X}_t, \tag{2.30b}$$

$$\lambda_t(\boldsymbol{s}) \in \mathbb{R}. \tag{2.30c}$$

Let $(\lambda_t^*(\boldsymbol{s}) : t \in T', \boldsymbol{s} \in \mathcal{S})$ be the vector of optimal solutions of (2.30). As $\gamma_t(\boldsymbol{s}) > 0$, it must be the case that the constraints (2.30b) hold with equality at optimality, i.e.,

$$\lambda_t^*(\boldsymbol{s}) = c_t(\boldsymbol{s}, \boldsymbol{x}) + \delta \sum_{\boldsymbol{s}' \in \mathcal{S}} \mathbb{P}_t(\boldsymbol{s}' | \boldsymbol{s}, \boldsymbol{x}) \lambda_{t+1}^*(\boldsymbol{s}'), \quad \forall t \in T',\, \boldsymbol{s} \in \mathcal{S},\, \boldsymbol{x} \in \mathcal{X}_t,$$

which implies that $\lambda_t^*(\boldsymbol{s}) = V_t(\boldsymbol{s})$ by Bellman's optimality principle. Thus, we can recover the value functions of (2.8) by solving model (2.30). Note if we choose $\gamma_1(\boldsymbol{s}) = \Pr(\boldsymbol{S}_1 = \boldsymbol{s})$, such that $\sum_{\boldsymbol{s} \in \mathcal{S}} \gamma_1(\boldsymbol{s}) = 1$, we can express the optimal value of (2.8) according to

$$z_{\boldsymbol{\alpha}} = \sum_{\boldsymbol{s} \in \mathcal{S}} \gamma_1(\boldsymbol{s}) \lambda_1^*(\boldsymbol{s}) = \sum_{\boldsymbol{s} \in \mathcal{S}} \gamma_1(\boldsymbol{s}) V_1(\boldsymbol{s}).$$

Unfortunately, the number of constraints in formulation (2.30) is prohibitively large for the problem instances considered in our numerical examples; hence, we solve the dual of (2.30), which has significantly fewer constraints. Let $(\mu_t(\boldsymbol{s}, \boldsymbol{x}) : t \in T',\, \boldsymbol{s} \in \mathcal{S},\, \boldsymbol{x} \in \mathcal{X}_t)$ be the vector of dual variables associated with constraints (2.30b). Define $T'' \equiv \{1, \dots, N-2\}$. Then, the dual LP formulation of (2.8) is

$$
\begin{aligned}
\min \quad & \sum_{t \in T'} \sum_{\boldsymbol{s} \in \mathcal{S}} \sum_{\boldsymbol{x} \in \mathcal{X}_t} c_t(\boldsymbol{s}, \boldsymbol{x}) \mu_t(\boldsymbol{s}, \boldsymbol{x}) && && (2.31a) \\
\text{s.t.} \quad & \sum_{\boldsymbol{x} \in \mathcal{X}_1} \mu_1(\boldsymbol{s}, \boldsymbol{x}) = \gamma_1(\boldsymbol{s}), && \forall \boldsymbol{s} \in \mathcal{S}, && (2.31b) \\
& \sum_{\boldsymbol{x} \in \mathcal{X}_t} \mu_{t+1}(\boldsymbol{s}, \boldsymbol{x}) - \delta \sum_{\boldsymbol{s}' \in \mathcal{S}} \sum_{\boldsymbol{x} \in \mathcal{X}_t} \mathbb{P}_t(\boldsymbol{s} | \boldsymbol{s}', \boldsymbol{x}) \mu_t(\boldsymbol{s}', \boldsymbol{x}) = \gamma_{t+1}(\boldsymbol{s}), && \forall t \in T'', \ \boldsymbol{s} \in \mathcal{S}, && (2.31c) \\
& \mu_t(\boldsymbol{s}, \boldsymbol{x}) \geq 0, && \forall t \in T', \ \boldsymbol{s} \in \mathcal{S}, \ \boldsymbol{x} \in \mathcal{X}_t. && (2.31d)
\end{aligned}
$$

It can be shown that the optimal dual solutions of (2.31) has a one-to-one correspondence with the optimal policy of (2.8). That is, for each $\boldsymbol{s} \in \mathcal{S}_t$, $\mu_t^*(\boldsymbol{s}, \boldsymbol{x}) > 0$ when $\boldsymbol{x} = \boldsymbol{x}_t^*$, and $\mu_t^*(\boldsymbol{s}, \boldsymbol{x}) = 0$ otherwise (see the discussion in [16]). Therefore, the optimal policy $\boldsymbol{\pi}^*$ can be directly recovered from the optimal solutions of model (2.31).

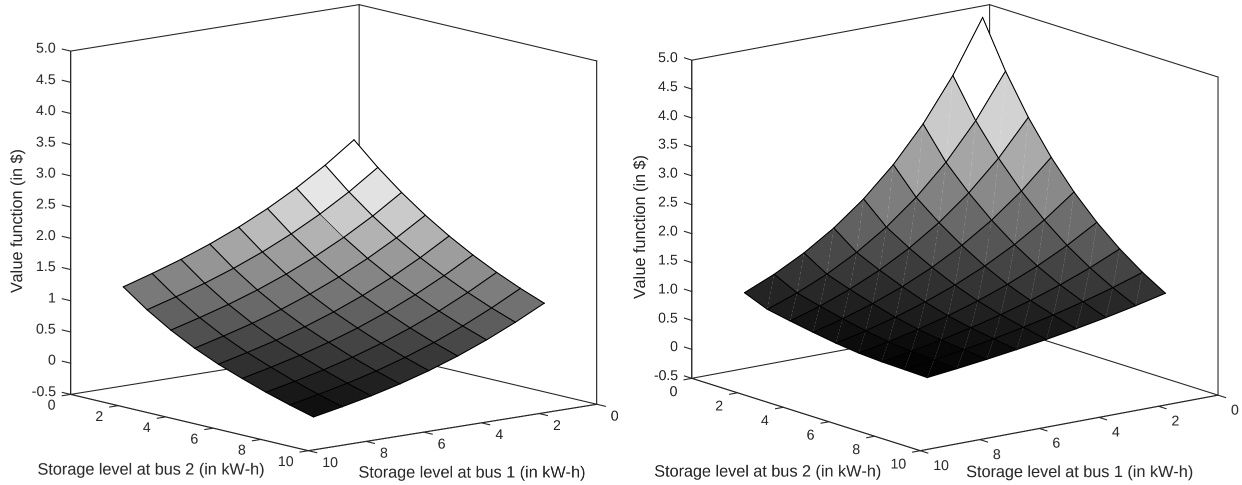
Model (2.31) was coded in Python 2.7 and solved using Gurobi 6.5. The discount factor δ was set to 0.99. All problem instances were executed on a Windows-based 64-bit, 4th generation, Intel® Core™ i7, 64 GB, 2.9 GHz Windows machine.

2.5.3 Results and Discussion

First, we illustrate the behavior of the value functions with respect to the storage levels. For a given storage level vector $\boldsymbol{y}_t$, the *average value function*, denoted by $\bar{V}_t(\boldsymbol{y}_t)$, is

$$
\bar{V}_t(\boldsymbol{y}_t) = \sum_{\boldsymbol{w}_t \in \mathcal{W}_t} \mathbb{P}(\boldsymbol{W}_t = \boldsymbol{w}_t) V_t(\boldsymbol{w}_t, \boldsymbol{y}_t), \quad \boldsymbol{y}_t \in \mathcal{Y}.
$$

Figure 6 depicts the average value functions at stages 1 and 17 as functions of the storage levels. Clearly, $\bar{V}_t(\boldsymbol{y}_t)$ is monotone decreasing and convex in $\boldsymbol{y}_t$. This implies that the expected future cost decreases with increasing storage levels but with decreasing marginal benefit. Similar trends were observed at all other stages in the planning horizon. Interestingly, the surface plot in Figure 6(b) has a steeper slope than the one in Figure 6(a), particularly at lower storage levels. This is because stage 17 marks the onset of the peak-price periods – characterized by high price variability – in which procurement costs rise rapidly when stored energy is in short supply. Note that the average value of the function $\bar{V}_1(\boldsymbol{y}_1)$ over its domain $\mathcal{Y}$ represents the daily, optimal operational cost of using a storage system with total capacity $\bar{\alpha} = \alpha_1 + \alpha_2$. Then, the marginal benefit (or marginal value) of using storage can be defined as the difference between the operational costs at capacities 0 and $\bar{\alpha}$. For instance, the marginal benefit of using storage was equal to \$26.75 for the problem instance used here ($\bar{\alpha} = 20$ kW-h).

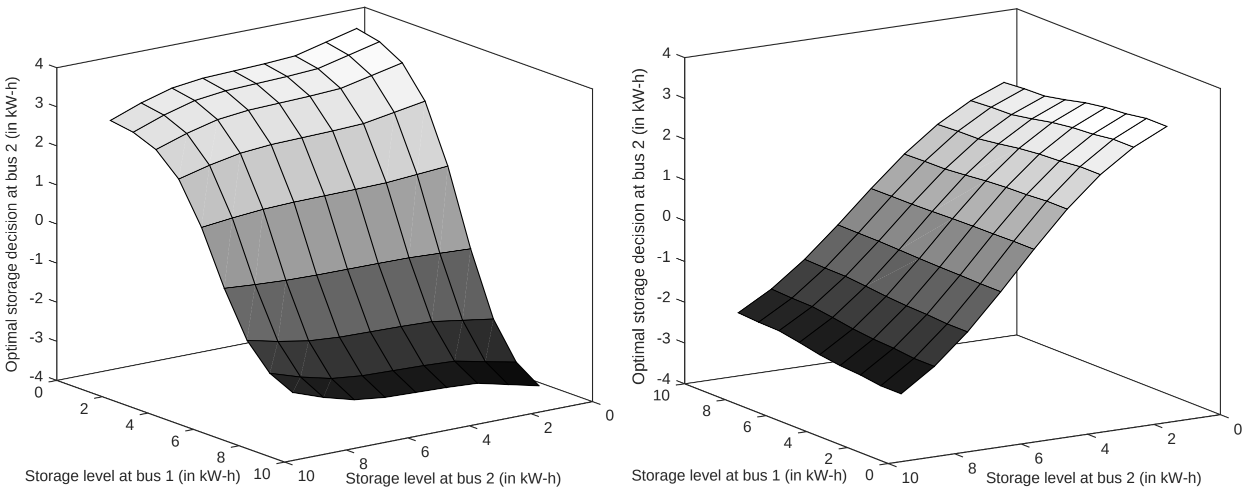


(a) Stage 1 average value function $\bar{V}_1(\boldsymbol{y}_1)$.

(b) Stage 17 average value function $\bar{V}_{17}(\boldsymbol{y}_{17})$.

Figure 6: Average value functions in stages 1 and 17.

Next, to illustrate the main results of Theorem 2.1, Figure 7 depicts the optimal charge or discharge decisions in stage $t = 17$ as functions of the storage levels for a fixed exogenous state $\boldsymbol{w}_{17} = (30, 4.5, 4.5)$. We note that the optimal storage decisions are monotone decreasing in the storage levels. Moreover, the optimal storage decision at each bus exhibits greater sensitivity to a marginal change in the storage level at that bus, as opposed to the storage level at the other bus. To illustrate this point, when y_2 is fixed at 10 kW-h, and y_1 increases from 0 to 10 kW-h, the optimal charge/discharge decision (u_t^{1*}) decreases from +2.97 kW-h to -3.03 kW-h as (i.e., $\Delta_1 u_t^{1*} = -0.6$). On the other hand, when y_1 is fixed at 10 kW-h, and y_2 increases from 0 to 10 kW-h, u_t^{1*} decreases from -2.95 kW-h to -3.32 kW-h (i.e., $\Delta_2 u_t^{1*} = -0.037$). Similar trends were observed in u_t^{2*}, and more generally, for all of the storage decisions at each stage $t \in T'$.



(a) Optimal storage decisions at bus 1 (u_t^{1*}).

(b) Optimal storage decisions at bus 2 (u_t^{2*}).

Figure 7: Optimal storage decisions in stage 17.

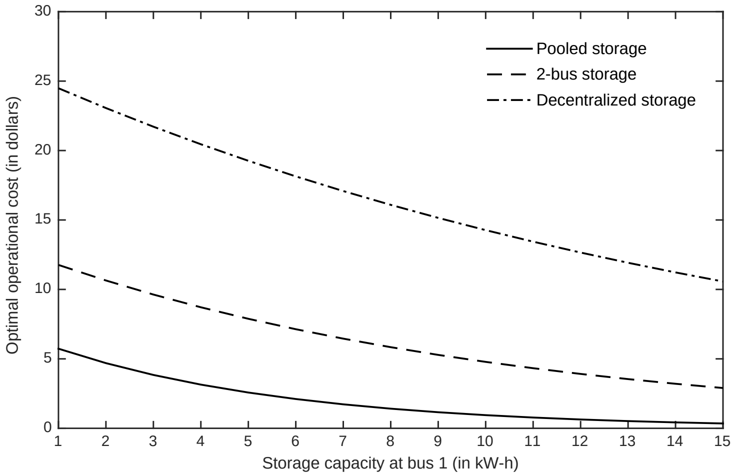


Figure 8: Comparison of the optimal operational costs in PS, CS, and DS networks.

Finally, we compared the operational costs of the pooled storage (PS), 2-bus storage (CS), and decentralized storage (DS) networks for different storage capacities. For the sake of comparison, we fixed $\alpha_2 = 10$ kW-h and varied α_1 from 1 to 15 kW-h in intervals of

1 kW-h. Figure 8 depicts the relative magnitudes of the optimal operational costs of the three network configurations as functions α_1, illustrating Theorem 2.2. As the optimal policies of single-storage models exhibit a dual-threshold structure [64, 153], the pooled and the decentralized storage models were solved using the monotone value iteration algorithm (see [122]). It is evident from Figure 8 that, for all three networks, the operational cost decreases with increasing storage capacity but with decreasing marginal benefit. Moreover, the differences between the operational costs of each decreases rapidly as the storage capacity at bus 1 increases. This behavior stems from the increased flexibility of using stored energy to satisfy demand locally at each bus without having to transmit much energy between them. Although Theorem 2.2 establishes the relative magnitudes of these operational costs, it is clear from Figure 8 that the bounds are not tight, especially at low storage capacities. This is indicative of non-negligible line-loss costs in energy networks with capacitated lines and storage systems. Hence, optimal policies derived from single-storage models are not suitable for distribution networks, as they do not adequately capture interactions between distinct storage devices in a networked environment.

3.0 ENERGY STORAGE MANAGEMENT IN MICROGRIDS VIA STOCHASTIC PROGRAMMING

In this chapter, a multistage stochastic programming (SP) model is formulated whose objective is to minimize the expected total energy costs incurred within a microgrid under randomly evolving renewable supply, demand, and pricing information. The model prescribes the amount of energy to procure, store, and discharge in each decision stage of a finite planning horizon. However, for even a moderate number of stages, the model is computationally intractable; therefore, the stochastic dual dynamic programming (SDDP) algorithm is customized to obtain high-quality approximate solutions. Computation times and optimization gaps are significantly reduced by implementing a dynamic cut selection procedure and a lower bound improvement scheme within the SDDP framework. An extensive computational study reveals significant cost savings as compared to myopic and non-storage policies, as well as policies obtained using a two-stage SP model.

3.1 SUMMARY OF RELEVANT LITERATURE

The literature related to optimal demand-side energy storage strategies is relatively small but is developing rapidly. Most prevalent are models that devise optimal demand-response schemes for consumers with elastic loads [10, 59]. However, such schemes have exhibited only a minor shift in consumer demand to match prices [98]. More recently, residential storage has emerged as a key facilitator of demand response on the consumer side [44, 157]. Earlier formulations of the demand-side storage management problem have employed a linear programming approach to minimize finite-horizon electricity costs assuming *a priori* knowledge of prices [3, 74]. Van de Ven et al. [153] examined a demand response problem under real-time pricing uncertainty using a finite-horizon Markov decision process (MDP) model. Proved was the existence of a dual-threshold, cost-minimizing optimal storage policy for a residential consumer with finite energy storage capacity. Hill et al. [69] developed simple threshold policies for grid-scale energy storage to mitigate negative impacts of solar energy integration while improving the overall real-time frequency response and voltage control capabilities of the grid. Koutsopoulos et al. [86] analyzed an optimal storage control problem under price and supply uncertainty. Using an infinite-horizon MDP model, they derived an optimal threshold policy for the online problem and proved its asymptotic optimality (as the storage capacity approaches infinity). However, none of these MDP models account for simultaneous uncertainty in demand, supply, and pricing, and they use relatively few scenarios to keep the problem dimension low. Furthermore, they are single-consumer models that do not consider network constraints and interactions between different microgrid entities.

Consequently, stochastic programming (SP) has emerged as a viable alternative to MDP models for problems with continuous actions and high-dimensional state spaces (cf. [21] for additional details). Lee et al. [92] formulated a two-stage linear SP model to minimize investment and ancillary-generation costs in a power network with high penetration of renewable sources and energy storage. They employed the well-known L-shaped algorithm [21] to solve the model in a day-ahead setting; however, a relatively small number of scenarios (≈ 100) were considered. Ji et al. [78] proposed a two-stage stochastic, mixed-integer, quadratic programming (SMIQP) model to jointly optimize the day-ahead operations of renewable sources and storage systems in a microgrid, also using a small number of scenarios. Xi et al. [160] addressed the problem of co-optimizing the real-time scheduling of storage usage for multiple applications, such as energy arbitrage and regulation services, while accounting for price and renewable uncertainty. They proposed a two-stage, stochastic mixed-integer programming (SMIP) model to obtain piecewise-linear value function approximations for a MDP model with continuous states and actions. Other representative two-stage SP models for similar problems can be found in references [53, 110, 146, 168]. Generally speaking, most realistic SP models are NP-hard; however, their prevalence in the energy literature stems from the fact that several decomposition algorithms are available to solve such models efficiently [4].

By comparison, the literature is relatively sparse for multistage SP models that allow for recourse decisions at multiple stages of decision making. A key feature of multistage models is that they yield solutions that are *non-anticipative*, i.e., decisions made in any stage depend only on the information available up to that stage. By contrast, decisions from two-stage models are anticipative and may result in suboptimal strategies. However, multistage SP models are significantly harder to solve, and are intractable for even a moderate number of stages [20].

The main contributions of this chapter can be summarized as follows. First, a novel, multistage SP model is considered to obtain viable procurement and storage operation strategies in a grid-connected microgrid. In this model, the initial (or first-stage) decisions are the storage capacities of local energy storage systems (e.g., batteries), subject to a budget constraint. During each subsequent stage of the planning horizon, multiple recourse decisions are made, including the total active and reactive power procured from the grid, the active power charged to, or discharged from, local storage systems, and the active and reactive power flowing in the lines based on a rigorous power flow model. The objective is to minimize the expected total energy costs incurred within the microgrid over a finite planning horizon, subject to storage capacity, line capacity, and other physical constraints. The model is distinguished from existing two-stage SP models in that non-anticipative procurement and storage decisions must be made in the face of multiple sources of uncertainty: renewable supply, demand, and prices are all modeled as random variables. Second, to overcome the computational challenges associated with multistage SP models, we customize the stochastic dual dynamic programming (SDDP) algorithm [136] to obtain high-quality solutions for a 24-hour planning horizon. The algorithm is enhanced by implementing a dynamic cut selection (DCS) heuristic [40] to significantly reduce the SDDP computation time. Moreover, the SDDP algorithm is remarkably improved by employing a novel, yet pragmatic, lower bound enhancement procedure using Jensen's inequality. This refinement drastically reduces computation time and significantly improves solution quality, and it facilitates the use of a large number of potential scenarios. The computational study demonstrates that very tight solution bounds are attainable within a reasonable amount of time. The results also suggest the scalability of our customized SDDP algorithm to problems of larger scale. Finally, the computational results reveal significant economic advantages as compared to myopic and non-storage policies, as well as policies obtained using a two-stage SP model.

3.2 MODEL DESCRIPTION

This section describes the multistage SP formulation to prescribe viable energy procurement and storage strategies for microgrid entities over a finite planning horizon. As in [6, 139, 163], consider a grid-connected microgrid with a radial topology – a tree-like network of interconnected buses and power lines emanating from a reference bus (the *feeder*), which is connected to the main grid. The feeder is often connected to a distribution substation and delivers power procured from the main grid to other microgrid buses. Some or all of those buses have access to distributed storage systems. The operators use both distributed generation (e.g., wind and/or solar) and electricity procured from the main grid to satisfy the net demand and power flow constraints in each stage of the planning horizon. Surplus energy can be stored in finite-capacity storage systems for future use. The decision makers, who make procurement and storage decisions at the start of each stage, have access to real-time pricing information from an electricity spot market. However, the amount of energy that can be stored or made available for current or future stages is constrained by the capacity of the storage systems and power lines. Moreover, these decisions are subject to renewable supply, demand, and pricing uncertainty. The objective is to minimize the expected total energy costs incurred within the microgrid over the finite planning horizon.

Consider a planning horizon of length Υ, and partition the time interval $[0, \Upsilon)$ so that

$$[0, \Upsilon) = \bigcup_{t=1}^{N} [\varepsilon_{t-1}, \varepsilon_t),$$

where N is the number of time intervals (or stages) and ε_t is the tth decision epoch with $\varepsilon_0 \equiv 0$ and $\varepsilon_N \equiv \Upsilon$. Therefore, the discrete time horizon is denoted by $T = \{1, \ldots, N\}$, where $t \in T$ is the index of the tth stage, namely $[\varepsilon_{t-1}, \varepsilon_t)$. Let $\delta_t \equiv (\varepsilon_t - \varepsilon_{t-1})$ denote the duration of the tth stage. Let $C = \{0, 1, \ldots, K\}$ be the finite set of buses in the microgrid, where bus 0 denotes the feeder connected to the main grid, and bus $i \in C \setminus \{0\}$ denotes the ith microgrid bus. It is assumed that the feeder is not connected to any load, renewable generator or storage device, and has a fixed voltage level. For notational convenience, define $T' \equiv T \setminus \{1\}$ and $C' \equiv C \setminus \{0\}$. The set of all lines in the microgrid is denoted by $A = \{(i,k) : i,k \in C\}$, where $(i,k) \in A$ is a power line connecting bus i to bus k.

Next, the physical parameters of the microgrid are described. Let α be the maximum storage capacity of the microgrid, and $\beta_a(i,k)$ and $\beta_r(i,k)$ be the active and reactive power capacities (also known as nameplate capacities) of line (i,k). Let $\varphi(i,k)$ denote the impedance of line (i,k) such that

$$\varphi(i,k) = \lambda(i,k) + j\vartheta(i,k), \quad (i,k) \in A,$$

where $\lambda(i,k)$ and $\vartheta(i,k)$ denote the resistance and reactance of line (i,k), respectively, and $j = \sqrt{-1}$ is the unit imaginary number. Let ν denote the average per-unit cost of power lost due to resistive heating in any line $(i,k) \in A$. The quantity η_i denotes the average cost per unit energy charged to, or discharged from, the battery at bus i, while κ_i is the per-unit cost of battery capacity at bus i. The parameters ρ_c^i and ρ_d^i represent the charging and discharging efficiencies of the battery at bus i, respectively. The round-trip efficiency of the battery at bus i is defined as $\rho_i \equiv \rho_c^i \rho_d^i$ – a value that usually lies in the interval $[0.7, 0.9]$. The quantities τ_c^i and τ_d^i denote the maximum charging and discharging rates of the battery at bus i, respectively. Let $\gamma_{\min}^i$ and $\gamma_{\max}^i$ be the minimum and maximum proportions of battery capacity that can store energy, where $\gamma_{\min}^i, \gamma_{\max}^i \in (0,1)$. It is assumed that the batteries cannot self-discharge, i.e., energy is not dissipated when the batteries are not in use.

The uncertain variables in the model are described next. All random variables are defined on a common and complete probability space $(\Omega, \mathscr{A}, \mathbb{P})$. Let d_t^i denote the net-demand (demand minus renewable supply) realized per unit-time at bus i at the start of stage t, such that

$$d_t^i = r_t^i + jw_t^i, \quad (i,t) \in C \times T',$$

where r_t^i and w_t^i denote the active and reactive power components of d_t^i, respectively. Collect the net demand realizations in the vector $d_t \equiv ((r_t^i, w_t^i) : i \in C')$. Let p_t denote the real-time price realized at the start of stage t. Then for each $t \in T'$, the bounded vector $\omega_t \equiv (d_t, p_t) \in \Omega_t$ denotes the stage t realization of the random vector $\widetilde{\omega}_t$. Assume $|\Omega_t| < \infty$

for all $t \in T'$. It is noted that $\Omega_t \subseteq \mathbb{R}^M$, where $M \leq 2K - 1$. Henceforth, we refer to a scenario $\omega \in \Omega \equiv \Omega_2 \times \cdots \times \Omega_N$ as a realization (or sample path) of the stochastic process $\{\widetilde{\omega}_t : t \in T'\}$.

In the following subsection, the decision variables and constraints of the model are defined, and a multistage SP model is formulated using the DistFlow equations for radial networks.

3.3 MULTISTAGE STOCHASTIC PROGRAMMING MODEL

Let x_1^i be the stage 1 battery capacity decision at bus i that incurs a cost $\kappa_i \, x_1^i$. The capacity decisions are made before any of the random variables are realized. Collect the stage 1 decisions in the vector $x_1 \equiv (x_1^i : i \in C')$. Let $c_1 \equiv (\kappa_i : i \in C')$ denote the cost vector in stage 1 so that the total cost in this stage is

$$c_1' x_1 = \sum_{i \in C'} \kappa_i \, x_1^i. \tag{3.1}$$

However, the capacity decisions are constrained by the maximum storage capacity of the microgrid as follows:

$$0 \leq \sum_{i \in C'} x_1^i \leq \alpha. \tag{3.2}$$

Unique to this model is the fact that, starting from stage 2 and moving forward in time, microgrid operators make recourse decisions at the start of each stage. The stage t recourse decisions, when ω_t is realized, are collected in the vector $x_t(\omega_t)$. Henceforth, for notational convenience, $x_t(\omega_t)$ is simply written as x_t. For each $i \in C'$ and $(i, k) \in A$, the decision vector for stage $t \in T'$ is defined as $x_t \equiv (y_t^i, m_t^i, n_t^i, v_t^i, q_t(i,k), s_t(i,k), a_t, b_t)$, whose elements are as follows:

- y_t^i: energy storage level at bus i at the start of stage t;
- m_t^i: active power charged into the battery at bus i;
- n_t^i: active power discharged from the battery at bus i;
- v_t^i: voltage level at bus i

- $q_t(i,k)$: active power flow in line (i,k);
- $s_t(i,k)$: reactive power flow in line (i,k);
- a_t: active power procured from main grid at the feeder;
- b_t: reactive power procured from main grid at the feeder.

For each bus $i \in C'$, the battery levels in successive stages are coupled via

$$y^i_{t+1} = y^i_t + \delta_t(\rho^i_c m^i_t - n^i_t/\rho^i_d), \quad t \in T'. \tag{3.3}$$

An interpretation of (3.3) is that the energy stored in a battery equals the stored energy at the start of the current stage, minus (plus) the amount of energy discharged from (charged to) the battery in the current stage, scaled by the discharging (charging) efficiency parameter. Note that the power quantities m^i_t and n^i_t are multiplied by the factor δ_t to convert them into units of energy. It is assumed that the batteries are capable of charging or discharging active power only and not reactive power (cf. [6, 54, 162]). Because demand, renewable supply and prices exhibit diurnal seasonality [152, 161], storage operations are optimized over a planning horizon that covers at least one complete cycle of the seasonal variables. Therefore, the terminal storage levels of the batteries are set to their initial levels [12, 76]. Specifically, for all $i \in C'$,

$$y^i_1 = y^i_N. \tag{3.4}$$

The energy that is charged to, or discharged from, the storage device is constrained by the current storage level, as well as the charging and discharging rates of the battery. Therefore, for each $i \in C'$,

$$0 \leq m^i_t \leq \min\{\tau^i_c, \delta^{-1}_t(x^i_1 - y^i_t)/\rho^i_c\}, \qquad t \in T', \tag{3.5}$$

$$0 \leq n^i_t \leq \min\{\tau^i_d, \delta^{-1}_t\, \rho^i_d y^i_t\}, \qquad t \in T'. \tag{3.6}$$

As battery life can be reduced due to excessive charging or discharging, for each $i \in C'$, battery levels in each stage are limited by the following state-of-charge (SOC) constraints:

$$\gamma^i_{\min}\, x^i_1 \leq y^i_t \leq \gamma^i_{\max}\, x^i_1, \quad t \in T'. \tag{3.7}$$

Next, described are the constraints related to power flow in the lines. In contrast to transmission systems, where power flows are characterized using DC optimal power flow approximations, the DistFlow model is often adopted for distribution networks to calculate the complex power flow and voltage profiles. Several recent papers have justified using the DistFlow equations for microgrids [6, 31, 147, 156, 163]. Because power flow is directional, assume that $q_t(i,k) \geq 0$ when active power flows from bus i to bus k, and $q_t(i,k) < 0$ if it flows from bus k to i; similar notation is adopted for the reactive component $s_t(i,k)$. For a given bus $i \in C'$, let Λ_i and Θ_i denote the parent bus and the adjoining children buses connected to bus i, respectively. The fixed voltage level at the feeder is denoted by v_0. Then for each $i \in C'$ and $t \in T'$, the DistFlow equations are

$$q_t(\Lambda_i, i) = r_t^i + m_t^i - n_t^i + \sum_{k \in \Theta_i} q_t(i,k), \tag{3.8a}$$

$$s_t(\Lambda_i, i) = w_t^i + \sum_{k \in \Theta_i} s_t(i,k), \tag{3.8b}$$

$$v_t^i = v_t^{\Lambda_i} - \frac{\lambda(\Lambda_i, i) q_t(\Lambda_i, i) + \vartheta(\Lambda_i, i) s_t(\Lambda_i, i)}{v_0}. \tag{3.8c}$$

The left-hand side of (3.8a) represents the active power that flows into a bus from its parent, while the right-hand side is the net active power that flows out of the bus to its children, after accounting for the local active demand and battery power flows. Equation (3.8b) can be similarly interpreted for reactive power flows. Equation (3.8c) is used to compute the voltage level of bus i.

Power procured from the main grid is delivered to the microgrid via the feeder (bus 0). The DistFlow equations at the feeder are

$$a_t = \sum_{k \in \Theta_0} q_t(0,k), \qquad t \in T', \tag{3.9a}$$

$$b_t = \sum_{k \in \Theta_0} s_t(0,k), \qquad t \in T', \tag{3.9b}$$

where $a_t + jb_t$ is the net power injected into the microgrid via the feeder at stage t. To avoid reverse power flows at the feeder that can negatively affect operation of voltage regulators and protective devices [6, 155], the following non-negativity constraints are imposed:

$$a_t \geq 0, \qquad t \in T'. \tag{3.10}$$

For any line $(i,k) \in A$, the active and reactive power flows are constrained by the nameplate capacities via

$$|q_t(i,k)| \le \beta_a(i,k), \qquad t \in T', \tag{3.11}$$

$$|s_t(i,k)| \le \beta_r(i,k), \qquad t \in T', \tag{3.12}$$

ensuring that power lines are not damaged due to resistive overheating [6, 50]. Additionally, distribution networks typically require the nominal voltage level at each bus to be maintained within a tolerance band [139, 163]. Therefore, the voltage level at each bus $i \in C'$ is constrained by the inequality

$$v^i_{\text{min}} \le v^i_t \le v^i_{\text{max}}, \quad t \in T', \tag{3.13}$$

where v^i_{min} and v^i_{max} denote the minimum and maximum voltage levels allowed at bus i, respectively.

It is noted that the decisions made at a current stage depend on the decisions made up to the previous stage via the temporal linking constraints (3.3). Thereby, the set of feasible decisions x_t in each stage $t \in T'$ is denoted by $\mathcal{X}_t(x_{t-1}, \omega_t)$, and this set is defined by the constraints (3.3) – (3.13) for each $(\omega, t) \in \Omega \times T'$.

Finally, the objective function, which is to be minimized, is described. Let c_t denote the cost vector in stage $t \in T'$ so that the total cost incurred in this stage is

$$c_t' x_t = p_t a_t \delta_t + \sum_{i \in C'} \eta_i (m^i_t + n^i_t)\delta_t + \sum_{(i,k)\in A} \nu \ell_t(i,k)\delta_t, \tag{3.14}$$

where $\ell_t(i,k)$ is the resistive power loss in line (i,k). The first term on the right-hand side of (3.14) is the total grid procurement cost, the second term is the total battery charge-discharge cost, and the third term represents the total cost incurred due to power-line losses in stage t. Each term on the right-hand side of (3.14) is multiplied by δ_t to convert power units to energy units. The battery cost rate η_i is assumed to be equal for both charging and discharging; however, this assumption can be relaxed. Using the DistFlow equations, the resistive power loss in line $(i,k) \in A$ can be closely represented by the quadratic function (see [143, 162])

$$\ell_t(i,k) = \lambda(i,k) \left(\frac{q_t(i,k)^2 + s_t(i,k)^2}{v_0^2} \right), \quad t \in T', \tag{3.15}$$

so the per-unit cost incurred due to power losses in line (i, k) at stage t is equal to $\nu \ell_t(i, k)$. Note that, because $\ell_t(i, k)$ is quadratic and convex, it can be readily approximated using piecewise-linear functions (cf. [166] in the context of the unit-commitment problem). However, note that the bilinear term $p_t a_t$ makes the cost function nonconvex in (ω_t, x_t). To make $c_t' x_t$ convex in (x_t, ω_t), we use the McCormick approximation [104] to replace the bilinear term by its convex envelope. Similar convex approximations were used by [30, 119] in the context of the hydrothermal operation planning problem. The multistage SP model can now be formulated and represented in the nested form

$$
\begin{aligned}
z = \min_{x_1} \;\; & c_1' x_1 + \mathbb{E}_{\widetilde{\omega}_2}\left[\min_{x_2} c_2' x_2 + \mathbb{E}_{\widetilde{\omega}_3|\widetilde{\omega}_2}\left[\min_{x_3} c_3' x_3 + \cdots + \mathbb{E}_{\widetilde{\omega}_N|\widetilde{\omega}_{N-1}}\left[\min_{x_N} c_N' x_N\right] \ldots\right]\right] \\
\text{s.t.} \;\; & x_t \in \mathcal{X}_t(x_{t-1}, \omega_t), \;\; \forall (\omega, t) \in \Omega \times T', && \text{(3.16a)} \\
& 0 \leq \sum_{i \in C'} x_1^i \leq \alpha, && \text{(3.16b)}
\end{aligned}
$$

where $\mathbb{E}_{\widetilde{\omega}_i|\widetilde{\omega}_j}$ denotes the expectation taken with respect to the conditional probability measure $\mathbb{P}(\widetilde{\omega}_i|\widetilde{\omega}_j)$. Note that the nested structure of model (4.2) is a direct consequence of the multiple recourse opportunities available to the decision maker as information is progressively revealed over the planning horizon. This distinguishes model (4.2) from two-stage SP models that allow only a singular recourse opportunity under uncertainty [89, 110]. Unfortunately, model (4.2) is computationally intractable, even when the number of stages N is moderate (see [20, 136, 138] for additional details). However, in Section 3.4, we describe how the stochastic dual dynamic programming (SDDP) algorithm can be used to obtain high-quality, approximate solutions to model (4.2).

3.4 STOCHASTIC DUAL DYNAMIC PROGRAMMING (SDDP) ALGORITHM

SDDP is a well-known decomposition procedure that can be used to solve multistage, stochastic programs with a large number of stages [117, 136]. The SDDP algorithm builds piecewise-linear outer approximations of the cost-to-go functions at each stage by randomly sampling from a finite set of scenarios. The algorithm iteratively updates the lower and upper bounds of the optimal value z of model (4.2) using a two-step procedure – a forward pass and a backward pass – and assumes stage-wise independence of the random variables. The algorithm terminates once the bounds satisfy a prescribed convergence criterion.

To customize the SDDP algorithm, model (4.2) is first reformulated as an N-stage stochastic dynamic program. The first-stage (or stage 1) problem is defined as

$$\begin{aligned} z = \min_{x_1} \; & c_1' x_1 + \mathbb{E}[V_2(x_1, \widetilde{\omega}_2)] \\ \text{s.t.} \;\; & 0 \leq \sum_{i \in C'} x_1^i \leq \alpha, \end{aligned} \tag{3.17}$$

where $V_2(x_1, \omega_2)$ is the total future cost incurred under decision x_1 and realization $\omega_2 \in \Omega_2$. For $t \in T'$, the stage t problem is defined as

$$\begin{aligned} V_t(x_{t-1}, \omega_t) = \min_{x_t} \; & c_t' x_t + \mathbb{E}[V_{t+1}(x_t, \widetilde{\omega}_{t+1})] \\ \text{s.t.} \;\; & F_t x_t = h_t(\omega_t) - G_t x_{t-1}. \end{aligned} \tag{3.18}$$

In formulation (3.18), matrices F_t and G_t, and the vector $h_t(\omega_t)$, are obtained by reformulating constraints (3.3) – (3.13) as equality constraints. Let $\pi_t(\omega_t)$ denote the optimal dual vector associated with (3.18). In dynamic programming parlance, $V_t(x_{t-1}, \omega_t)$ is the stage t cost-to-go (or value) function. Without loss of generality, assume $\mathbb{E}[V_{N+1}(x_N, \widetilde{\omega}_{N+1})] = 0$ in the stage N problem; however, any continuous, convex function can be assumed for $\mathbb{E}[V_{N+1}(x_N, \widetilde{\omega}_{N+1})]$.

To implement the SDDP algorithm, the scenarios of model (4.2) are stored as a finite scenario tree with N stages. A scenario tree originates at a *root* node that stores the first-stage decision x_1 and progressively branches to other child nodes that are defined by the number of possible stage t realizations $|\Omega_t|$ in stage $t \in T'$. The nodes in stage N are called the *leaf nodes*. The total number of leaf nodes equals the number of scenarios of (4.2). Figure 9 depicts a scenario tree with three stages and six scenarios. Because the number of scenarios, $|\Omega_2 \times \cdots \times \Omega_N|$, grows exponentially with N, problem (4.2) must be solved approximately to accommodate a planning horizon of 24 (or more) decision stages. To solve (4.2) approximately, a finite number of scenarios are sampled from the scenario tree

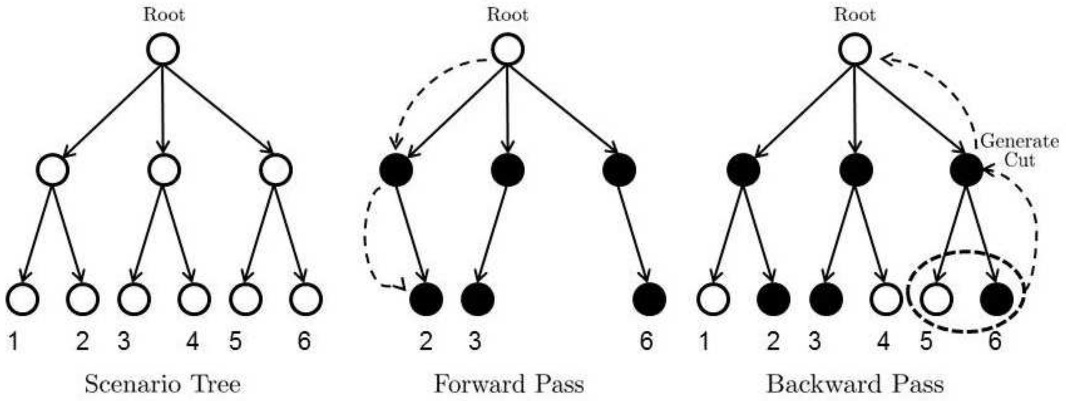


Figure 9: Depiction of a scenario tree with three stages and six scenarios.

to develop a piecewise-linear outer approximation of $\mathbb{E}[V_{t+1}(x_t, \widetilde{\omega}_{t+1})]$ in (3.18) for each stage t problem. The approximate stage t cost-to-go function is denoted by $\widehat{V}_t(x_{t-1}, \omega_t)$. The outer approximations are developed by generating Bender's cuts

$$\theta_{t+1} \geq \bar{h}_{t+1,k} - \bar{\pi}'_{t+1,k} G_{t+1} x_t, \qquad k \in \mathcal{K}. \tag{3.19}$$

In (3.19), the set $\mathcal{K}$ is the index set for all Bender's cuts added to each stage t problem (3.18). Here, column vector $\bar{\pi}_{t+1,k} \equiv \mathbb{E}[\pi_{t+1}(\widetilde{\omega}_{t+1})]$ defines the gradient and $\bar{h}_{t+1,k}$ is the intercept term for cut $k \in \mathcal{K}$, where $\bar{h}_{t+1,k} \equiv \mathbb{E}[\widehat{V}_{t+1}(x_t^k, \widetilde{\omega}_{t+1})] + \bar{\pi}'_{t+1,k} G_{t+1} x_t^k$, and x_t^k is a feasible stage t solution. Thus, the approximate stage t problem has the form

$$\begin{aligned} \widehat{V}_t(x_{t-1}, \omega_t) = \min_{x_t} \ & c'_t x_t + \theta_{t+1} \\ \text{s.t.} \ & F_t x_t = h_t(\omega_t) - G_t x_{t-1}, \\ & \theta_{t+1} \geq \bar{h}_{t+1,k} - \bar{\pi}'_{t+1,k} G_{t+1} x_t, \ k \in \mathcal{K}, \end{aligned} \tag{3.20}$$

while the approximate stage 1 problem is

$$\begin{aligned} \hat{z} = \min_{x_1} \ & c'_1 x_1 + \theta_2 \\ \text{s.t.} \ & \sum_{i \in C'} x_1^i \leq \alpha, \\ & \theta_2 \geq \bar{h}_{2,k} - \bar{\pi}'_{2,k} G_2 x_1, \ k \in \mathcal{K}, \\ & x_1 \geq 0. \end{aligned} \tag{3.21}$$

Let $\hat{x}(\omega) \equiv (\hat{x}_t(\omega))_{t\in T}$ be an approximate policy obtained by solving problems (3.20) and (3.21) for scenario $\omega \in \Omega_2 \times \cdots \times \Omega_N$. In the forward pass of the SDDP algorithm, S distinct scenarios are sampled uniformly from the scenario tree using the well-known Monte-Carlo method (see [40, 136, 138]). Subsequently, the stage t problems are solved approximately for each of the sampled scenarios, starting from the first stage and moving forward to the final stage. At the completion of the forward pass, an upper bound to z is calculated, and a convergence criterion is tested. If the criterion is satisfied, the algorithm terminates; otherwise, the current optimal policy is amended by adding $|\Omega_t|$ Bender's cuts to each of the stage t problems associated with the sampled scenarios, starting from the last stage and working backwards to the first stage. Figure 9 depicts the forward and backward passes of the SDDP algorithm for a given scenario tree. Let $\Omega_S \subset \Omega$ be a finite set of S distinct scenarios ω sampled from Ω. The steps of the algorithm are summarized as follows:

1. **Sampling**

   Sample S distinct scenarios ω from Ω to form Ω_S.

2. **Forward Pass**

   2a) For $t = 1$, solve (3.21) and save $\hat{x}_1$ and $\hat{z}$;

   2b) For $t = 2, \ldots, N$ and $\omega \in \Omega_S$, solve (3.20) and store $\hat{x}_t(\omega)$ and $\widehat{V}_t(\hat{x}_{t-1}(\omega), \omega_t)$, where ω_t is the $(t-1)$th component of ω.

3. **Convergence Test** (at the 95% confidence level)

   3a) Compute an upper bound of z by

   $$z_u = c_1'\hat{x}_1 + (1/S) \sum_{\omega\in\Omega_S} \sum_{t=2}^{N} c_t'\hat{x}_t(\omega),$$

   and assign lower bound $z_\ell := \hat{z}$ by solving (3.21);

   3b) Terminate the algorithm if (see [136])

   $$z_u + \left(1.96\,\hat{\sigma}_u/\sqrt{S}\right) - z_\ell \leq \epsilon,$$

   where ϵ is a prescribed accuracy level, and $\hat{\sigma}_u$ is the sample standard deviation of the observations $\{z_\omega : \omega \in \Omega_S\}$ with

   $$z_\omega = c_1'\hat{x}_1 + \sum_{t=2}^{N} c_t'\hat{x}_t(\omega);$$

   Else go to Step 4.

4. **Backward Pass**

   4a) For $t = 2, \ldots, N$, $\omega \in \Omega_S$, and for each $\omega_t \in \Omega_t$, solve (3.20) using $\hat{x}_{t-1}(\omega)$, and save $\hat{\pi}_t(\omega_t)$ and $\widehat{V}_t(\hat{x}_{t-1}(\omega), \omega_t)$; Generate a Bender's cut (3.19) and add it to all subproblems at stage $t-1$;

   4b) Go to step 1.

3.5 IMPROVING THE PERFORMANCE OF SDDP

The standard SDDP algorithm of Section 3.4 involves visiting S scenarios in the forward pass, and then a backward pass is performed to build $|\Omega_t|$ cuts for each stage t problem. This procedure yields an increasing number of Bender's cuts for each stage t problem, not all of which are active at each iteration of the algorithm. To reduce the computational burden of solving problems (3.20) and (3.21), a more sensible approach is to select cuts for the current iteration from a collection of cuts that have been generated in prior iterations. While there exist several classes of cut selection procedures in the stochastic programming literature (cf. [21, 138]), an effective dynamic cut selection (DCS) procedure due to de Matos et al. [40] was implemented to reduce the computation time of the standard SDDP algorithm.

3.5.1 Dynamic Cut Selection (DCS) Heuristic

In the DCS procedure, cuts are added iteratively rather than adding all cuts at once. At each iteration of the SDDP algorithm, a sequence of values $Q_k \equiv (\bar{h}_{t+1,k} - \bar{\pi}'_{t+1,k} G_{t+1} \hat{x}_t)$ are computed for all $k \in \mathcal{K}$ (the index set of all cuts generated in prior iterations) and $\hat{x}_t$ is the current optimal solution at stage t. If the cut $k^* = \text{argmax}_k\{Q_k\}$ has not been added to (3.20) yet, then k^* is added to (3.20) and re-solved. Moreover, at each stage $t \in T'$, the cuts that were generated for the stage t problems, associated with the scenarios visited in prior iterations, can be accessed to determine the set of active cuts for the stage t problems in the current iteration. Consequently, a broad set of cuts are retained that are likely to be important to all the subproblems at a given stage. Note that at the start of a new iteration, cuts of all the stage problems are cleared. However, the cuts are not discarded because one cannot ensure that a currently inactive cut will remain inactive for other scenarios visited at a later iteration. Thus, the algorithm has access to all the cuts generated in earlier iterations, and only adds the active cuts to the stage t problems during the forward and backward passes of the current iteration. The steps of the algorithm are summarized as follows:

1. **Remove cuts**

   Remove the cuts from all stage t problems (3.20) and the stage 1 problem (3.21).

2. **Sampling**

   Sample S distinct scenarios ω from Ω to form Ω_S.

3. **Forward Pass**

   3a) For $t = 1$, solve (3.21) and save $\hat{x}_1$ and $\hat{z}$;

   3b) For $t = 2, \ldots, N$ and $\omega \in \Omega_S$, solve (3.20) and store $\hat{x}_t(\omega)$ and $\widehat{V}_t(\hat{x}_{t-1}(\omega), \omega_t)$, where ω_t is the $(t-1)$th component of ω; if $k^* = \text{argmax}_k\{\bar{h}_{t+1,k} - \bar{\pi}'_{t+1,k} G_{t+1} \hat{x}_t(\omega)\}$ is not in (3.20), then add cut k^* and re-solve (3.20).

4. **Convergence Test**

   Identical to Step 3 of the standard SDDP algorithm.

5. **Backward Pass**

   5a) For $t = 2, \ldots, N$,$\omega \in \Omega_S$, and for each $\omega_t \in \Omega_t$, solve (3.20) using $\hat{x}_{t-1}(\omega)$, and save $\hat{\pi}_t(\omega_t)$ and $\widehat{V}_t(\hat{x}_{t-1}(\omega), \omega_t)$; If $k^* = \text{argmax}_k\{\bar{h}_{t+1,k} - \bar{\pi}'_{t+1,k} G_{t+1} \hat{x}_t(\omega)\}$ is not in (3.20), then add cut k^* and re-solve (3.20); Generate a Bender's cut (3.19) and add it to all subproblems at stage $t - 1$;

   5b) Go to step 1.

3.5.2 Lower Bound Improvement via Jensen's Inequality

The DCS heuristic reduces the number of cuts that are added at each stage; however, it cannot guarantee the strength of these cuts. The standard SDDP algorithm exhibits slow convergence because the lower bounds obtained from the approximate stage t problems – which do not exploit strong valid inequalities – are relatively weak (cf. [15, 149]). To address this shortcoming, we propose a lower bound improvement scheme that makes use of Jensen's inequality (see [125], p. 188). The idea is to generate a set of strong valid inequalities during the backward pass of the SDDP algorithm. First, the elements of an artificial scenario, $\bar{\omega} \equiv (\bar{\omega}_2, \ldots, \bar{\omega}_N)$, are obtained by

$$\bar{\omega}_t = \sum_{\omega_t \in \Omega_t} \omega_t / |\Omega_t|, \quad t \in T'.$$

We call $\bar{\omega}$ the average scenario and note that it may not necessarily belong to Ω. Next, during a backward pass of the SDDP algorithm, a valid inequality of the form

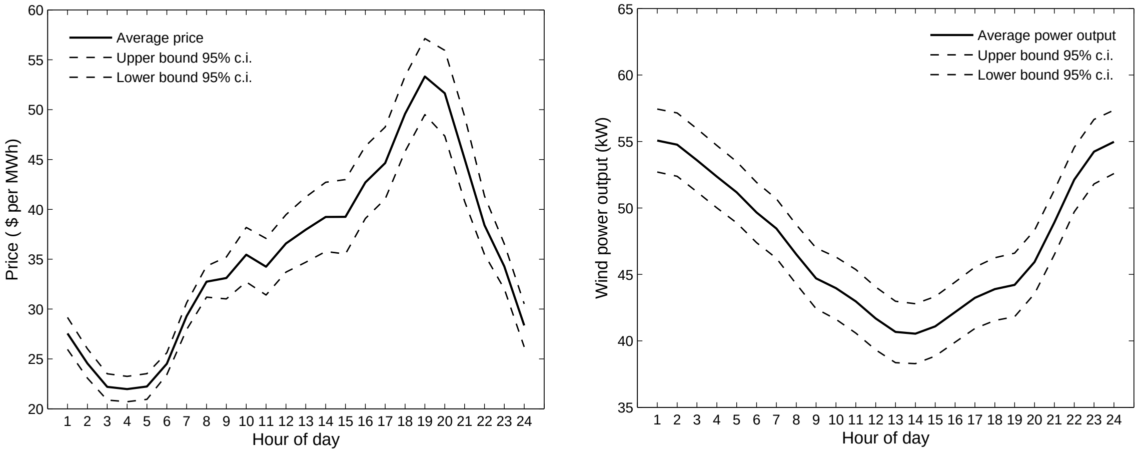
$$\theta_{t+1} \geq V_{t+1}(x_t, \bar{\omega}_{t+1}), \quad t \in T, \tag{3.22}$$

is added to each approximate stage t problem. The right-hand side of inequality (3.22) is evaluated by setting $x_t = \hat{x}_t$, where $\hat{x}_t$ is the current optimal solution at stage t. Because $\bar{\omega}$ is computed *a priori*, and $\hat{x}_t$ is known for each $t \in T'$ from the forward pass, adding cut (3.22) does not impose any additional computational burden. Note that, for a given feasible solution x_t, the inequality $\theta_{t+1} \geq \mathbb{E}[V_{t+1}(x_t, \widetilde{\omega}_{t+1})]$ holds because problem (3.20) is a relaxation of problem (3.18). Furthermore, we have that $\mathbb{E}[V_{t+1}(x_t, \widetilde{\omega}_{t+1})] \geq V_{t+1}(x_t, \bar{\omega}_{t+1})$ by Jensen's inequality. Therefore, the cuts (3.22) are valid inequalities for the approximate stage t problems. The computational results of Section 3.6 reveal that the addition of these valid inequalities significantly improves solution quality and drastically reduces computation time, as compared to the standard SDDP algorithm.

3.6 COMPUTATIONAL RESULTS

This section presents computational results illustrating procurement and storage policies obtained by solving problem (4.2) approximately using standard SDDP, its DCS variant (SDDP+DCS), and the lower bound improvement scheme integrated within SDDP+DCS (SDDP+DCS+J), as described in Sections 3.4, 3.5.1 and 3.5.2, respectively. First, detailed descriptions of the source data, microgrid configuration and computational study are provided.

Hourly demand and real-time electricity pricing data were obtained from PJM Interconnection (http://pjm.com/), while hourly wind speed data were obtained from the National Renewable Energy Laboratory (http://nrel.gov/) for the years 2012 and 2013. Wind speeds were converted to active wind-power outputs for a small-scale wind turbine by applying equation (13) of [29]. Moreover, reactive wind-power outputs were obtained from the PQ characteristic curve of a small wind turbine; for more details, see [36]. As price and wind levels are highly seasonal, the original data were partitioned into two disjoint sets, each spanning one year. As an aid to data visualization, for each data set, 95% confidence intervals (c.i.) were constructed for hourly demand, wind-generation and price levels by fitting truncated normal distributions whose parameters were estimated via maximum likelihood estimation (MLE). Figure 10 depicts the average electricity prices and wind-generation levels over a 24-hour period for the year 2012. Note that midnight is 0000 so that hour 1 corresponds to 0000 to 0100, hour 2 is 0100 to 0200, and so forth.



(a) Real-time hourly electricity prices.

(b) Hourly wind-generation levels.

Figure 10: Average price and wind-generation levels for the year 2012.

3.6.1 Microgrid Configuration

We consider a 4-bus microgrid configuration as depicted in Figure 11. The 4-bus system is powered by the main grid and a small-scale wind turbine that is connected to a single local storage device located at bus 3, which implies that $x_1^i = 0$ for all $i \in C' \setminus \{3\}$. In what follows, let $\bar{r}_t^i$ and $\bar{w}_t^i$ denote the sample means of the active power ($\tilde{r}_t^i$) and reactive power ($\tilde{w}_t^i$) components of net demand, respectively, in stage $t \in T'$ at bus $i \in C'$. Denote the associated sample variances by $\hat{r}_t^i$ and $\hat{w}_t^i$, respectively. The net demand components at bus $i \in C'$ are assumed to follow truncated normal distributions, i.e.,

$$\tilde{r}_t^i \sim TN(\bar{r}_t^i, \hat{r}_t^i), \qquad t \in T', \tag{3.23}$$

$$\tilde{w}_t^i \sim TN(\bar{w}_t^i, \hat{w}_t^i), \quad t \in T'. \tag{3.24}$$

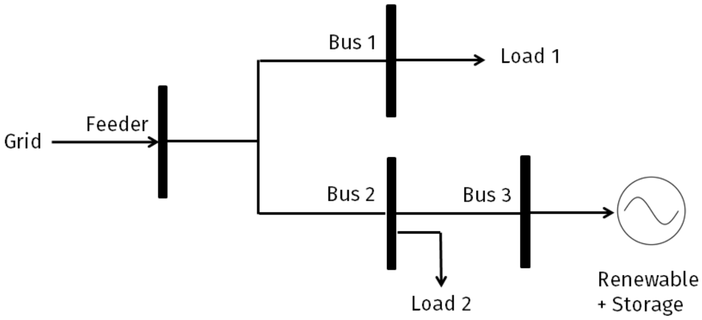


Figure 11: A 4-bus, grid-connected microgrid.

For each stage t, the (finite) supports of the random variables $\tilde{r}_t^i$ and $\tilde{w}_t^i$ were determined using the maximum and minimum levels of net demand obtained from the PJM demand and NREL wind data. The loads connected to buses 1 and 2 are assumed to be homogenous with identical active power distributions, i.e.,

$$\tilde{r}_t^1 \overset{d}{=} \tilde{r}_t^2, \quad t \in T'.$$

Moreover, both loads are assumed to have high power factors (the ratio of active to apparent power) and, therefore, consume negligible reactive power. Thus,

$$\tilde{w}_t^1 = 0 \quad \text{and} \quad \tilde{w}_t^2 = 0, \quad t \in T'.$$

The wind turbine connected to bus 3, on the other hand, generates both active and reactive power in each stage t.

The microgrid operators are assumed to be price-takers with no demand-response capabilities. However, simple demand-response schemes can be integrated into the model without imposing additional computational burden. Microgrid operators have access to the main grid at all times and can procure electricity at spot prices that also follow truncated normal distributions of the form

$$\tilde{p}_t \sim TN(\bar{p}_t, \hat{p}_t), \quad t \in T', \tag{3.25}$$

where $\bar{p}_t$ and $\hat{p}_t$ are the sample mean and variance of the price at stage t, respectively.

The power lines are all assumed to have identical line capacities (β_a, β_r), resistance (λ) and reactance (ϑ) values. A deep-cycle, lead-acid battery with a shelf-life of over 24 hours is used as the storage device at bus 3. The battery parameters $(\kappa, \eta, \rho_c, \rho_d, \tau_c, \tau_d, \gamma_{\min}, \gamma_{\max})$ are selected based on information provided in [93]. It is assumed that the battery is charged up to its maximum SOC level at the start of the planning horizon, i.e., $y_1 = \gamma_{\max} x_1$, and therefore, the terminal battery level is $y_N = y_1 = \gamma_{\max} x_1$ by (3.4). The voltage limits $(v_{\min}, v_{\max})$ at each bus were set to ±5% of the feeder voltage v_0 (see [6, 139]). Table 3 summarizes the parameter values used in the computational experiments.

3.6.2 Description of Experiments

For the computational experiments, a 24-hour planning horizon in the year 2012 was considered, i.e., $T = \{1, \ldots, 25\}$, where hour 1 is stage 2, hour 2 is stage 3 and so forth; hence, $\delta_t = 1$ for each t. The standard SDDP, SDDP+DCS and SDDP+DCS+J algorithms were coded in Python 2.7 and solved using the Gurobi 6.5 solver. The piecewise-linear approximation to (3.15) was created using the default settings of Gurobi, and the dual-simplex method was selected as the default linear programming (LP) solver. The algorithms were implemented on a 64-bit, 4th generation Intel® Core™ i7, 64 GB, 2.9 GHz Windows machine.

We considered different combinations of the number of sampled scenarios (S) in the forward pass, and the number of stage t realizations $(|\Omega_t|)$ for the backward pass, holding $|\Omega_t|$ constant for all $t \in T'$. The parameter S was varied from 50 to 250 in increments of 50, while $|\Omega_t|$ was varied from 5 to 20 in increments of 5. For each $t \in T'$, a set of $|\Omega_t|$ samples

Table 3: Parameter values for the problem instances.

Parameter	Parameter description	Value
α	Maximum battery capacity (kWh)	60.00
κ	Per-unit cost of battery capacity (\$/kWh)	50.00
η	Per-unit cost of charging/discharging (\$/kWh)	2.00
τ_c	Charging rate of the battery (kW)	25.00
τ_d	Discharging rate of the battery (kW)	25.00
ρ_c	Charging efficiency of the battery	0.95
ρ_d	Discharging efficiency of the battery	0.90
$\gamma_{\min}$	Minimum battery SOC fraction	0.10
$\gamma_{\max}$	Maximum battery SOC fraction	0.90
β_a	Line capacity for active power (kW)	60.00
β_r	Line capacity for reactive power (kVAR)	60.00
λ	Line resistance (Ohm)	0.009
ϑ	Line reactance (Ohm)	0.009
υ_0	Fixed voltage level at the feeder (kV)	10.00
$\upsilon_{\min}$	Minimum bus voltage level (kV)	9.50
$\upsilon_{\max}$	Maximum bus voltage level (kV)	10.50

were first generated to construct a scenario tree before running any of the three procedures. To generate the samples $\omega_t \equiv (r_t^i, w_t^i, p_t : i \in C')$ at each stage t, a multivariate truncated normal distribution was used in which each marginal distribution is also truncated normal. The sampling procedure was further simplified by assuming that the random variables at each stage are all mutually independent, i.e., for each $t \in T'$

$$\mathcal{F}_t(\omega_t) = \mathcal{P}_t(p_t) \prod_{i=1}^{3} \mathcal{R}_t^i(r_t^i)\mathcal{W}_t^i(w_t^i), \quad \forall \omega_t \in \Omega_t,$$

where $\mathcal{F}_t$ is the joint probability density function (p.d.f.), and $\mathcal{P}_t$, $\mathcal{R}_t^i$, and $\mathcal{W}_t^i$ are the stage t marginal p.d.f.s of price (3.25), active power (3.23) and reactive power (3.24) components of net demand at bus i, respectively. Alternative sampling procedures can be used, such as those described in [87]; however, we chose this sampling procedure to satisfy the stage-wise independence criterion of the SDDP algorithm. For comparison purposes, the same $|\Omega_t|$ samples were used to generate scenario trees for all three procedures.

Once a scenario tree is generated, S distinct forward-pass scenarios $\omega \equiv (\omega_t : t \in T')$ are uniformly sampled from the scenario tree in each iteration of the three procedures. It is worth noting that empirical forecast distributions, based on forward bootstrapping techniques [116], can also be used to sample scenarios; however, our main purpose here is to illustrate the usefulness of the SP model and its solutions. The quality of solutions obtained using standard SDDP, SDDP+DCS, and SDDP +DCS+J algorithms was assessed by computing the approximate gap percentage

$$\text{Gap } (\%) = \frac{z_u - z_\ell}{z_\ell} \times 100. \tag{3.26}$$

All three procedures were terminated if either the SDDP convergence criterion was satisfied (with $\epsilon = 10^{-5}$), or 500 iterations were completed, whichever occurred first.

3.6.3 Results and Discussion

The computational results for the standard SDDP procedure are provided in Table 4. It is noted that for a fixed number of scenarios S, the gap percentage decreases sharply as $|\Omega_t|$ increases; however, as one might expect, the computation times increase. Specifically, if additional state information is used at each stage to develop Bender's cuts, better value-function approximations are obtained; however, a far greater number of stage t problems must be solved. Similarly, for a fixed value of $|\Omega_t|$, the gap percentages decrease, and the computation time increases (but only moderately) as S increases. This trend stems from the fact that the lower bounds progressively improve, albeit slowly, as more scenarios are sampled, but a larger number of stage t problems must be solved in each iteration. However, the reported gap percentages indicate that the bounds are not tight. For example, when $S = 250$ and $|\Omega_t| = 20$, solving the model for over 6.7 hours reduced the gap to only 6.96%.

Table 5 summarizes the results when using the standard SDDP algorithm supplemented with DCS. The table reveals that both solution quality and computation time improved,

Table 4: Results using the standard SDDP algorithm.

S	$\lvert\Omega_t\rvert$	z_ℓ	z_u	Gap %	Time (mins)
50	5	62.46	78.03	24.93	48.76
	10	66.28	77.64	17.14	64.85
	15	68.89	77.21	12.08	98.33
	20	69.8	77.02	10.34	132.65
100	5	63.42	77.97	22.94	61.28
	10	67.24	77.28	14.93	92.37
	15	69.72	77.15	10.66	131.54
	20	70.23	77.01	9.65	189.95
150	5	64.45	77.56	20.34	73.89
	10	67.93	77.15	13.57	129.76
	15	69.96	76.97	10.03	186.21
	20	70.82	76.58	8.13	238.86
200	5	64.98	77.39	18.96	84.21
	10	68.02	76.93	13.10	148.49
	15	70.37	76.62	8.88	235.72
	20	70.95	76.51	7.84	325.39
250	5	65.35	77.12	18.01	97.65
	10	68.63	76.75	11.83	171.26
	15	70.92	76.58	7.98	272.74
	20	71.43	76.21	6.96	403.64

relative to the results for standard SDDP. For instance, when $S = 250$ and $|\Omega_t| = 20$, SDDP+DCS reduces the gap percentage by a factor of over 2.3 (from 6.96% to 3%). Additionally, the computation time is reduced by a factor of nearly 1.3 (from 403.64 min to 311.37 min). However, the gap percentages for SDDP+DCS are still high in absolute terms, which points to a slow rate of convergence of the lower bounds z_ℓ. In fact, for both the SDDP and SDDP+DCS procedures, the lower bounds converged rapidly in the first few iterations, and converged very slowly thereafter. For example, when $S = 250$ and $|\Omega_t| = 20$, the SDDP +DCS lower bound improved from $-\infty$ to 71.43 (final value of standard SDDP lower bound) in the first 60 iterations, and increased to only 73.98 in the next 440 iterations. This slow convergence may be attributed to weak Bender's cuts that are generated in the backward pass. Such weak cuts often lead to relaxations of the stage t problems that are not tight – a common trend in Bender's decomposition-based algorithms [149]. The numerical results indicate that DCS alone may not significantly improve computation time and the quality of solutions obtained by the SDDP algorithm.

To improve the convergence rate of the lower bounds, and further reduce the computation time, valid inequalities of the form (3.22) were added to each stage t problem in the backward pass (see Section 3.5.2). Table 6 summarizes the results for the SDDP+DCS+J procedure. The table reveals a dramatic improvement in solution quality as well as computation time. For instance, when $S = 250$ and $|\Omega_t| = 20$, SDDP+DCS+J reduces the gap percentage by factors of 21.75 and 9.36, respectively, as compared to standard SDDP (from 6.96% to 0.32%) and SDDP+DCS (from 3% to 0.32%). These results are highly significant in that a realistic scenario tree with 25 stages yields very high quality solutions (gap of 0.32%) within a reasonable amount of time (206.73 minutes). Figures 12 and 13 illustrate the gap and computation time reductions achieved by using SDDP+DCS+J for different values of $|\Omega_t|$.

Moreover, the numerical experiments suggest a significant improvement in the convergence rate of SDDP+DCS+J. For example, when $S = 250$ and $|\Omega_t| = 20$, the SDDP+DCS+J lower bound attained the final value of the SDDP+DCS lower bound ($z_\ell = 73.98$) in only 20 iterations. Furthermore, the lower bounds increased by only 5×10^{-4} in the final 300 iterations of SDDP+DCS+J, indicating convergence of the lower bounds. It is noteworthy that a mere 2.7% improvement (73.98 to 75.99) in the lower bound of SDDP+DCS+J, over that of the SDDP+DCS, caused the gap percentage to drop by nearly 2.7% (3% to 0.32%). The comparisons are even more stark between the standard SDDP and SDDP+DCS+J. A mere 6.38% (71.43 to 75.99) increase in z_ℓ reduced the gap from 6.96% to 0.32%.

Table 5: Results using the SDDP+DCS algorithm.

S	$\lvert\Omega_t\rvert$	z_ℓ	z_u	Gap %	Time (mins)
50	5	64.03	78.05	21.89	33.67
	10	68.25	77.61	13.71	49.29
	15	70.04	77.18	10.19	66.27
	20	71.26	77.01	8.08	87.25
100	5	65.57	77.93	18.85	48.58
	10	69.72	77.34	10.93	73.46
	15	71.23	77.16	8.30	101.28
	20	72.61	76.92	5.97	148.65
150	5	67.03	77.39	15.46	60.09
	10	70.04	77.19	10.04	102.25
	15	72.27	76.89	6.43	156.43
	20	73.06	76.45	4.61	195.23
200	5	68.17	77.41	13.55	73.21
	10	70.92	76.90	8.43	112.28
	15	72.89	76.69	5.21	179.38
	20	73.21	76.42	4.45	235.29
250	5	68.87	77.41	11.83	86.46
	10	71.11	76.98	7.68	147.76
	15	73.57	76.81	3.96	202.41
	20	73.98	76.15	3.00	311.37

Figure 14 depicts box plots of the lower bounds obtained for the SDDP, SDDP+DCS, and SDDP+DCS+J algorithms. It is noted that the lower bounds obtained by SDDP+DCS+J are not only stronger, but also less variable, as compared to the those of standard SDDP and SDDP+DCS. This is because the Jensen's inequality-based Bender's cuts lead to stronger

Table 6: Results using the SDDP+DCS+J algorithm.

S	$\lvert\Omega_t\rvert$	z_ℓ	z_u	Gap %	Time (mins)
50	5	70.12	77.98	11.21	16.65
	10	72.93	77.46	6.24	24.28
	15	74.17	77.19	4.07	36.69
	20	75.03	76.97	2.23	49.63
100	5	71.47	77.81	8.87	25.97
	10	73.54	77.34	5.13	37.80
	15	74.97	77.15	2.91	54.32
	20	75.45	76.82	1.84	71.21
150	5	72.12	77.72	7.77	38.57
	10	74.32	77.29	4.56	68.43
	15	75.59	76.83	1.64	101.36
	20	75.67	76.29	0.89	150.26
200	5	72.96	77.43	6.13	50.45
	10	75.01	77.16	2.95	83.53
	15	75.71	76.86	1.52	121.62
	20	75.98	76.28	0.45	179.97
250	5	73.21	77.41	6.03	64.36
	10	75.06	77.01	2.83	98.14
	15	75.89	76.85	1.34	141.42
	20	75.99	76.19	0.32	206.73

relaxations and, therefore, tighter lower bounds. That is, as the number of iterations increases, the relaxations become progressively stronger as a large number of high-quality Bender's cuts are added. Furthermore, the computation time decreases because DCS retains only the strong, active cuts from prior iterations. Consequently, the lower bounds converge

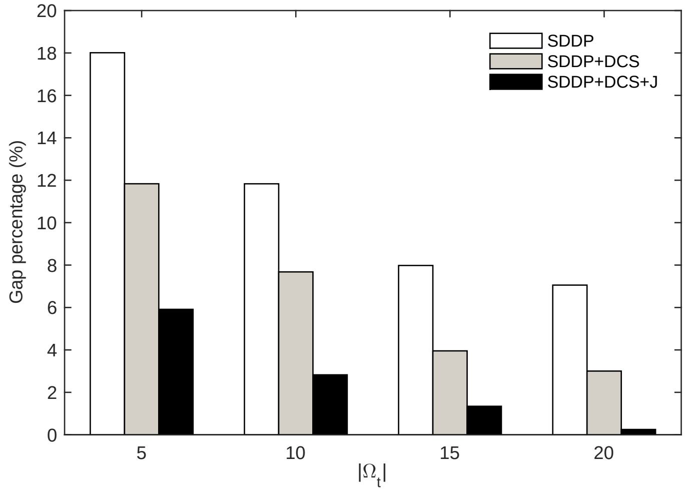


Figure 12: Gap percentage for different values of $|\Omega_t|$ when $S = 250$.

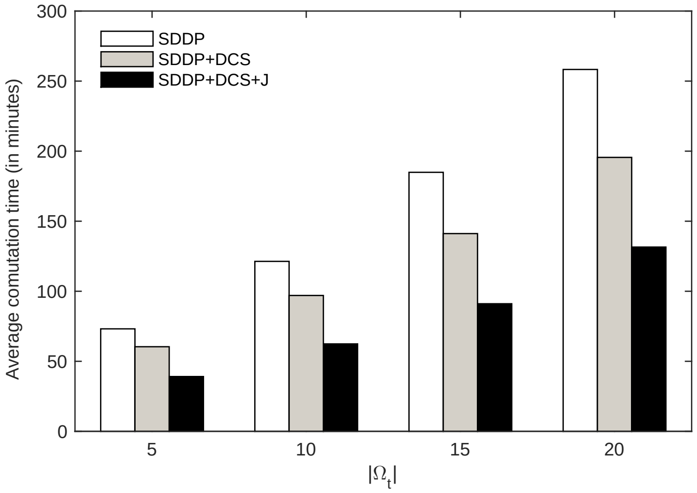


Figure 13: Average computation time for different values of $|\Omega_t|$.

rapidly, and the computation time decreases significantly using SDDP+DCS+J. By contrast, the variability of lower bounds obtained by standard SDDP and SDDP+DCS are of the same order because the DCS heuristic does not generate stronger cuts; it simply reduces the number of cuts that are retained from prior iterations during the current iteration. Figure 14 confirms our conjecture that SDDP+DCS+J generates much tighter lower bounds.

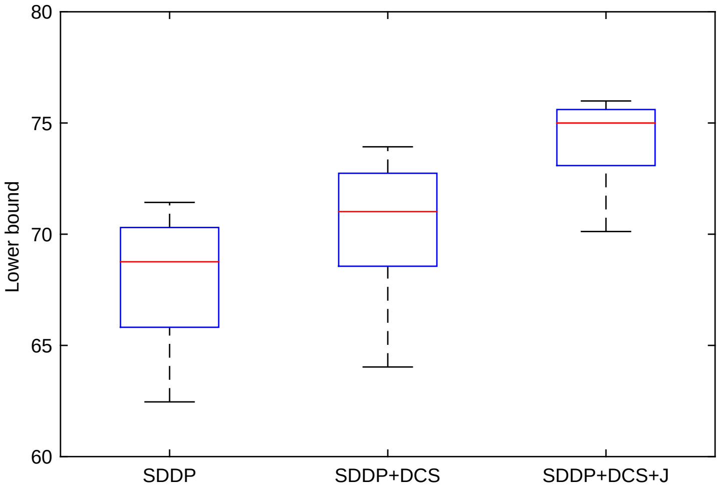


Figure 14: Box plots of the lower bounds obtained via SDDP and its two variants.

Next, we compared the approximate solutions of SDDP+DCS+J to solutions obtained using a myopic, price-based threshold (MPT) policy that ignores the future impact of current storage and power-flow decisions. The MPT policy maximizes the charging or discharging quantities at each stage, subject to the feasibility constraints (3.3)–(3.13), depending on whether the price realized at that stage is below or above a fixed price threshold, respectively. That is, for a given scenario $\omega \in \Omega$, the MPT policy, denoted by $x^{\phi}(\omega) \equiv (x_t^{\phi}(\omega))_{t \in T'}$, was obtained by solving a sequence of stage t problems

$$\begin{aligned} x_t^{\phi}(\omega) = \operatorname{argmax} \; & \sum_{i \in C'} m_t^i \mathbb{I}_{[p_t < \bar{p}]} + n_t^i \mathbb{I}_{[p_t \geq \bar{p}]} \\ \text{s.t.} \qquad & F_t x_t = h_t(\omega_t) - G_t x_{t-1}^{\phi}(\omega), \end{aligned}$$

where $x_{t-1}^{\phi}(\omega)$ is the MPT decision vector at stage $t-1$, $\bar{p}$ is a known price threshold and $\mathbb{I}_A$ denotes the indicator function of event A. In these experiments, the threshold $\bar{p}$ was set to the sample mean of the prices in the PJM pricing data. Specifically,

$$\bar{p} = \frac{1}{24} \sum_{t=2}^{25} \bar{p}_t \approx \$35.79/\text{MWh}.$$

The one-step cost at stage t using the MPT policy is denoted by $c_t' x_t^{\phi}(\omega)$ and is calculated via (3.14). Then, the total cost over the horizon for scenario ω is

$$z_{\phi}(\omega) = c_1' x_1^{\phi} + \sum_{t \in T'} c_t' x_t^{\phi}(\omega).$$

Let z_{ϕ} denote the average MPT policy cost of S distinct forward-pass scenarios used in SDDP+DCS+J.The expected cumulative costs incurred over the planning horizon were compared using SDDP+DCS+J, the MPT policy, and the corresponding model when no energy storage is available (for the case $S = 250$ and $|\Omega_t| = 20$). The expected cumulative cost at hour t is the sum of accumulated costs up to that hour, so the expected cost at hour 24 is the expected total cost incurred over the planning horizon. In the absence of storage capacity, we set $x_1 = 0$; therefore, there are no charging or discharging decisions in each of the subsequent stages. Denote the cost of the no-storage policy by z_N. Figure 15 reveals that the SDDP+DCS+J policy significantly reduces cumulative costs in each stage. Specifically, the total horizon costs are reduced by 25.65% (from 102.47 to 76.19) as compared to the MPT policy, and by 48.68% (from 143.67 to 76.19) as compared to the policy that does not use energy storage.

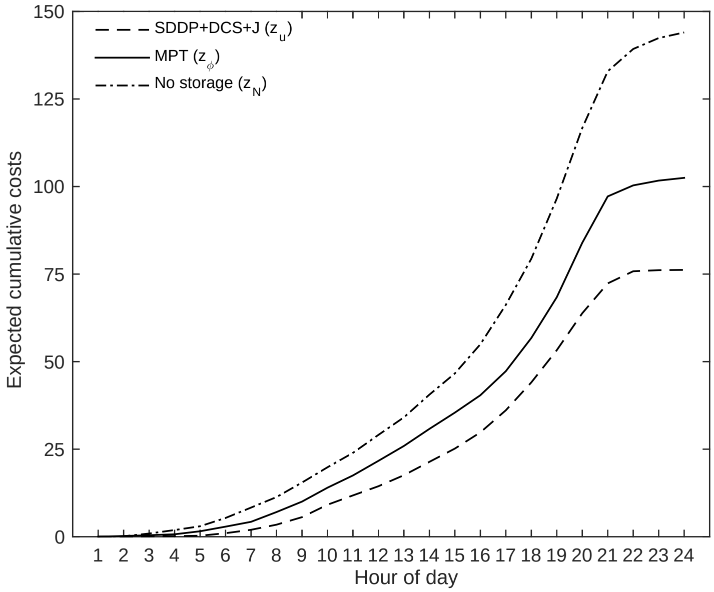


Figure 15: Expected cumulative costs over the planning horizon.

Figure 16 depicts the average battery levels for the SDDP+DCS+J and MPT policies and the average electricity price in each hour of the day. The battery can be charged up to its maximum SOC level ($\gamma_{\max} x_1$) at the start of the planning horizon. The data revealed that hours 1 to 10, on average, had low price and high wind-generation levels. Therefore, the battery retains most of its initial charge during hours 1 to 10 under the SDDP+DCS+J policy. The MPT policy does not discharge energy (on average) during hours 1 to 10 because the prices (on average) are less than the price threshold $\bar{p}$ in these periods. As prices increase and wind generation decreases in subsequent periods, energy is discharged from the battery to satisfy the demand under both of these policies. Not surprisingly, the battery is discharged to its minimum SOC level ($\gamma_{\min} x_1$) during the peak-price periods (hours 18 to 21) in both cases.

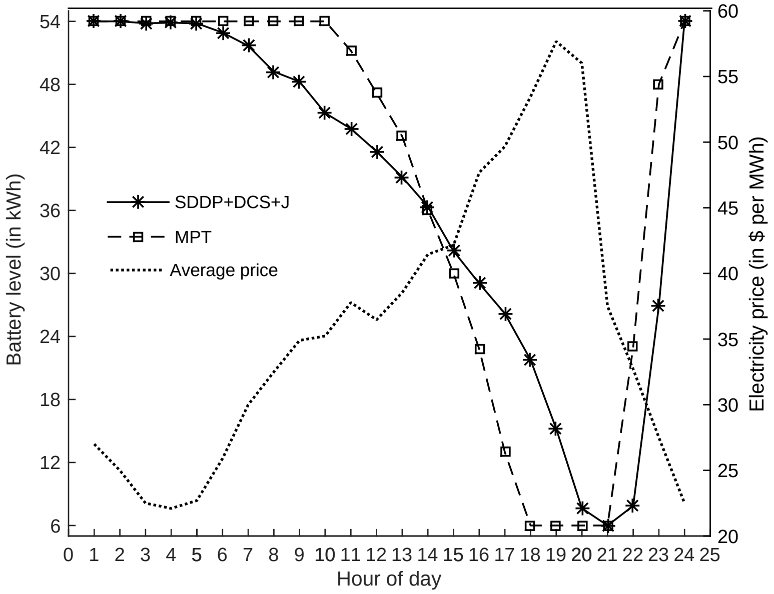


Figure 16: Average battery level and average price for each hour.

Although the initiation of discharging is earlier in SDDP+DCS+J (hour 6) compared to the MPT policy (hour 11), the MPT battery levels fall dramatically once discharge is initiated in hour 11. This is because the MPT policy maximizes the energy discharged from the battery between hours 11 to 21, when prices, on average, are greater than the threshold level $\bar{p}$. It is noted that for the MPT policy, the battery level reaches the minimum SOC level as early as hour 18, thereby forcing procurement of electricity during hours 18 to 21 when the (average) prices are highest. This behavior is reflected in Figure 15 by the steep slope of the cumulative-cost curve of the MPT policy between hours 18 to 21. Thus, the MPT policy effectively diminishes the advantage of using stored energy – namely to reduce peak-period costs. By contrast, the policy obtained via SDDP+DCS+J is intelligent in that it prescribes the use of stored energy during peak-price periods, thereby reducing the overall expected costs over the horizon. Figure 16 highlights the benefits of time-shifting energy consumption via storage in a microgrid. Although the two policies exhibit similar behavior over time, the SDDP+DCS+J policy accounts for the future impact of current decisions, yielding better operational decisions that lead to significant cost savings.

Next, we compared the solutions of the multistage SP model to those obtained by solving an associated two-stage SP model in which the non-anticipativity condition is relaxed. The two-stage model allows for only a single recourse opportunity in stage 2. The stage 2 recourse decisions for a realized scenario $\omega \in \Omega$ are collected in the vector $x(\omega) \equiv (x_t(\omega) : t \in T')$. For notational convenience, we drop the dependence of x on ω and simply write x. Let $c \equiv (c_t : t \in T')$ be the corresponding second-stage cost vector (also a function of ω). The stage 1 problem is

$$\check{z} = \min_{x_1} c_1' x_1 + \mathbb{E}[V(x_1, \omega)] \tag{3.27a}$$

$$\text{s.t. } 0 \leq \sum_{i \in C'} x_1^i \leq \alpha, \tag{3.27b}$$

and the stage 2 problem is

$$V(x_1, \omega) = \min_{x} c'x \tag{3.28a}$$

$$\text{s.t. } Fx = h(\omega) - Gx_1. \tag{3.28b}$$

The matrices F and G and vector $h(\omega)$ were obtained by reformulating constraints (3.3) – (3.13) together for all stages $t \in T'$. The SDDP+DCS+J algorithm was used to solve model (3.27) approximately. The lower and upper bounds of the optimal value $\check{z}$ are denoted by $\check{z}_\ell$ and $\check{z}_u$, respectively. Table 7 summarizes the results of the two-stage SP model.

Table 7: Two-stage results using the SDDP+DCS+J algorithm.

S	$\|\Omega\|$	$\check{z}_\ell$	$\check{z}_u$	Gap %	Time (mins)
50	10^4	80.02	82.86	3.55	7.57
	3×10^4	80.93	82.54	1.99	20.26
100	10^4	80.21	82.69	3.09	15.48
	3×10^4	81.61	82.43	1.00	46.23
150	10^4	80.52	82.48	2.45	24.86
	3×10^4	81.88	82.37	0.59	73.39
200	10^4	80.73	82.38	2.04	37.12
	3×10^4	81.96	82.37	0.50	101.72
250	10^4	81.01	82.36	1.67	56.75
	3×10^4	82.13	82.36	0.28	148.27

Similar to the results of the multistage model, the gap percentage for the two-stage model decreases as both S and $|\Omega|$ increase, while the computation time increases. Of far greater interest, however, is the comparison between the solution bounds of the multistage and two-stage models. Let $\check{z}_u^*$ and z_u^* denote the best upper bound obtained for the two-stage and multistage models, respectively, for a particular number of forward-pass scenarios S. We compared these upper bounds for each $S \in \{50, 100, 150, 200, 250\}$ using

$$\Delta_S(\%) = \frac{\check{z}_u^* - z_u^*}{\check{z}_u^*} \times 100,$$

where Δ_S denotes the cost savings over the planning horizon (as a percentage) when multiple recourse opportunities are available. Table 8 provides strong evidence that substantial cost savings are achieved by using the multistage SP model.

Finally, to examine scalability issues associated with our solution procedure, we solved a simplified two-bus system and compared its results with the 4-bus system. As depicted in Figure 17, the two-bus system is configured by aggregating the loads of the 4-bus system

Table 8: Two-stage versus multistage SP upper bounds.

S	$\check{z}_u^*$	z_u^*	$\Delta_S(\%)$
100	82.54	76.97	6.75
200	82.43	76.82	6.81
300	82.37	76.29	7.38
400	82.37	76.28	7.39
500	82.36	76.19	7.49

into a single bus that is connected to the wind turbine and a single storage device. The aggregated net demand components in the two-bus system are assumed to follow truncated normal distributions

$$\begin{aligned} \tilde{r}_t &\sim TN(\bar{r}_t, \hat{r}_t), \qquad t \in T', \\ \tilde{w}_t &\sim TN(\bar{w}_t, \hat{w}_t), \qquad t \in T', \end{aligned}$$

where the corresponding means and variances are as follows:

$$\begin{aligned} \bar{r}_t &= \sum_{i \in C'} \bar{r}_t^i; \qquad \hat{r}_t = \sum_{i \in C'} \hat{r}_t^i; \\ \bar{w}_t &= \sum_{i \in C'} \bar{w}_t^i; \qquad \hat{w}_t = \sum_{i \in C'} \hat{w}_t^i. \end{aligned}$$

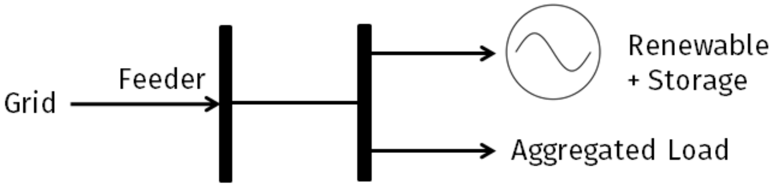


Figure 17: A two-bus, grid-connected microgrid.

Table 9: Results using SDDP+DCS+J for the two-bus microgrid.

$\|\Omega_t\|$	z_ℓ	z_u	Gap %	Time (mins)
5	54.62	56.15	2.90	17.36
10	55.06	55.54	0.87	35.32
15	55.32	55.52	0.36	58.54
20	55.41	55.51	0.18	84.75

The price distributions are identical to those of the 4-bus system, and the two-bus system has a single line constraint. The model parameters for the two-bus system were set to the values given in Table 3. The model was solved using the SDDP+DCS+J procedure. Table 9 summarizes the results for the case $S = 250$, as it provides the best gap percentages.

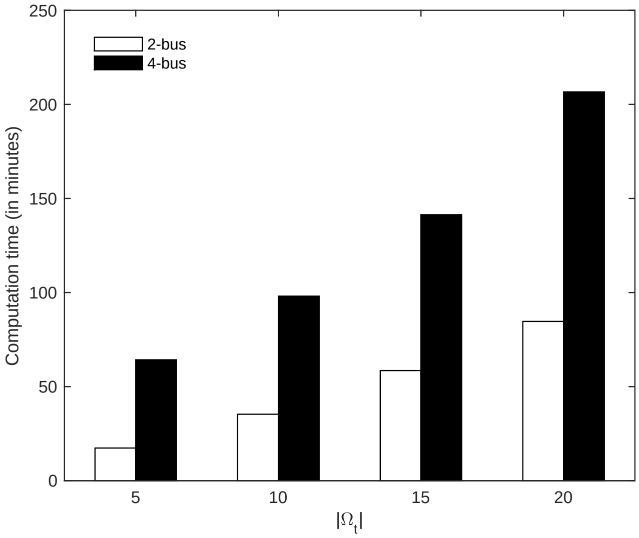


Figure 18: Computation time for different values of $|\Omega_t|$ when $S = 250$.

It is noted that gaps below 1% were obtained for all values of $|\Omega_t| \geq 10$. Moreover, the computation times are strikingly smaller. For example, a 25-stage scenario-tree with $|\Omega_t| = 20$ was solved in less than 1.5 hours with a gap of only 0.18%. Figure 18 compares the computation time between the two systems for different values of $|\Omega_t|$. The results are intuitive due to dimensionality reduction of the two-bus system. However, it is interesting to note that the computation time does not appear to scale exponentially with problem size. This provides affirmative evidence of the scalability of the SDDP+DCS+J algorithm to problems with multiple scales.

4.0 NONCONVEX REGULARIZATION FOR THE SDDP ALGORITHM

This chapter introduces a novel regularization scheme to improve the computational performance of the stochastic dual dynamic programming (SDDP) algorithm. Specifically, a class of nonconvex regularization functions, called the *folded-concave penalty*, is employed in the forward pass of SDDP to enhance the quality of outer approximations. The nonconvex forward-pass problems are solved using a mixed-integer linear programming (MILP) strategy. Established are almost sure convergence guarantees of the proposed regularization scheme under some mild regularity conditions. The potential benefits of the regularization scheme are demonstrated by way of two large-scale stochastic programming models arising in energy and finance. The results reveal significant improvements in the convergence rate and solution quality of SDDP, especially for high-dimensional problems.

4.1 SUMMARY OF RELEVANT LITERATURE

Multistage stochastic linear programming is a popular modeling framework for decision-making problems that allow multiple recourse opportunities under uncertainty. The underlying uncertainty is usually modeled as a general stochastic process. Such models find applications in a wide variety of domains, including energy ([17, 39]), finance ([23, 37]), supply-chain management ([67]) and transportation ([51, 121]). Except in rare cases, a multistage stochastic linear program (MSLP) is computationally intractable when the stochastic process has an uncountable number of realizations. Solution approaches for such problems are based on approximating the stochastic process by another process with finitely many realizations that can be represented using a scenario tree. One such approach is the sample-average approximation (SAA) procedure, which involves sampling a finite number of realizations and approximating the true value functions by sample-average functions. The resulting SAA instance is then solved using a deterministic optimization procedure (see [20, 126]), and the procedure is repeated until a termination criterion is satisfied.

However, the SAA approach is computationally intractable for high-dimensional problems. [135] showed that the number of scenarios required to obtain high-quality solutions grows exponentially with the number of decision stages, which results in prohibitively large scenario trees. The computational burden can be somewhat reduced by constructing smaller yet representative scenario trees using probability metrics (see [45, 66]). However, sampling-based decomposition (SBD) procedures provide the main route to solving large-scale SAA instances (see [33, 70, 118]).

The most popular among the SBD algorithms is the stochastic dual dynamic programming (SDDP) procedure of [117]. Specifically, the SDDP procedure constructs piecewise-linear outer approximations of the sample-average functions by iteratively sampling a finite number of scenarios from the scenario tree. Each iteration begins with a forward pass that generates a sequence of feasible decisions, called trial solutions, and computes a statistical upper bound of the optimal value of a SAA instance (henceforth, called the optimal SAA value). Each iteration ends with a backward pass that uses the trial solutions to update the outer approximations and produces a valid lower bound of the optimal SAA value. The algorithm terminates once the upper and lower bounds satisfy a prescribed convergence criterion. The essential role of sampling is to reduce the number of scenarios traversed in each iteration, which makes the SDDP procedure computationally appealing. It is well-known that SDDP is a finitely-convergent algorithm with provable performance guarantees (see [100, 118]). Moreover, when the stochastic process is stage-wise independent, SDDP promotes sharing of cuts among different nodes in a scenario tree, which is known to significantly speed up the algorithm (see [40, 77]).

However, the convergence rate of SDDP can be extremely slow for high-dimensional problems. This is because SDDP is based on the cutting-plane procedure of [83], whose worst-case complexity grows exponentially with the dimension of the problem (see page 160 in [114]). It is well-known that bundle methods have better convergence rates than cutting-plane procedures for convex optimization problems (see [81, 94]). Specifically, the bundle methods use a quadratic regularization scheme that stabilizes the generated solutions around a set of stability centers. By definition, a stability center is the best candidate solution known at the start of an iteration. Variants of this quadratic regularization method has been successfully applied to two-stage stochastic programs (see [68, 128, 130]). However, similar extensions for multistage stochastic programs are significantly harder due to the complexity of choosing stability centers at each node of a scenario tree (see [129]). Consequently, the literature on regularization methods for multistage models is relatively sparse.

Sen and Zhou [134] developed one of the first quadratic regularization schemes for multistage stochastic programming. Their algorithm begins with a forward pass where a collection of convex quadratic programs are solved to obtain the trial solutions. The backward pass is similar to that of SDDP. At the end of each iteration, the stability center at each node is updated. Clearly, such an updating strategy is impractical for large scenario trees. A more scalable quadratic regularization scheme was developed by [7], which facilitates the sharing of stability centers among different nodes. The authors showed that any random sequence of trial solutions generated by their algorithm converges almost surely to the optimal decisions. Interestingly, their convergence results hold for any choice of stability centers. A similar regularization strategy was implemented in [62] to solve a select class of portfolio-optimization problems. Recently, the authors of [151] devised a general convex regularization scheme based on the level-bundle methods (see [95]) that subsumes the method of [7]. The proposed method has attractive convergence properties and provides considerable flexibility in selecting regularization functions and stability centers of choice.

It is noted that the aforementioned regularization schemes were employed in the forward pass of SDDP. Essentially, regularization is used to steer current trial solutions towards good stability centers by penalizing the deviation between them, which is usually measured using a quadratic penalty function. Good stability centers refer to trial solutions that generated high-quality cutting planes in previous iterations. However, initial stability centers are usually poor due to the weak approximations at the start of the algorithm. Current regularization schemes ([7, 62]) aggressively penalize deviations in the initial iterations, which shrinks the trial solutions towards poor stability centers (exploitation). However, the weight of the penalty term is progressively reduced in subsequent iterations, which allows trial solutions to be far from good stability centers induced by stronger approximations (exploration). That is, regularization is enforced when it is least needed and dismissed when it would be most beneficial.

In this chapter, a new SDDP regularization scheme is introduced, which addresses the adverse effect of exploration versus exploitation in existing regularization approaches. Specifically, the proposed regularization scheme is based on a class of nonconvex penalty functions called folded-concave penalty, which have been widely used for high-dimensional statistical learning problems (see [49, 101, 164]). The key idea behind nonconvex regularization is the use of a bounded, concave penalty function that does not over-penalize large deviations, but penalizes small deviations more aggressively than the quadratic function. This promotes initial exploration of trial solutions away from poor stability centers and subsequent exploitation around good stability centers. We prove that our regularized SDDP algorithm has almost sure convergence properties and establish theoretical guarantees of the algorithm's asymptotic performance. We empirically demonstrate the potential benefits of our regularization scheme for two large-scale multistage stochastic optimization problems in energy and finance. We found that our regularized algorithm significantly outperformed the standard SDDP and the quadratic regularization-based SDDP procedures in terms of solution quality and convergence rate of the lower bounds. Interestingly, the benefits of nonconvex regularization were more pronounced for higher dimensional problem instances. Next, we summarize the main contributions of this chapter.

1. We develop a novel, nonconvex regularization scheme that is employed in the forward pass of SDDP. We consider two well-known folded concave penalty (FCP) functions – the smoothly clipped absolute deviation of [49] and the minimax concave penalty of [164] – to generate high-quality trial solutions. We establish the connection between nonconvex regularization and quadratic programming by deriving a nonconvex quadratic programming formulation for the FCP regularization problem. This facilitates the analysis of the FCP regularization scheme via mathematical programming techniques.
2. For each of the FCP functions, we derive an equivalent mixed-integer linear programming (MILP) model for the regularization problem. Specifically, we reformulate the nonconvex

quadratic program as a linear program with complementarity constraints and show that is equivalent to a mixed-integer linear program with big-M constraints. Furthermore, we show that the optimal solutions of the obtained MILP models are bounded and finite, which indicates that our regularization scheme is numerically stable. The proposed MILP strategy has two key advantages. First, it admits the use of branch-and-bound algorithms to determine global optimal solutions of the nonconvex regularization problem in a finite number of steps. Second, it facilitates the use of state-of-the-art MILP solvers within the SDDP framework. With the recent advances in MILP algorithms and computer architectures, such a scheme provides obvious computational benefits in solving large-scale problem instances. Our numerical experiments suggest that if the number of forward-pass scenarios are kept low, then the overall MILP computation times remain within reasonable time limits.

3. We show that our regularized SDDP algorithm has almost sure convergence properties under some mild regularity conditions for the sample-average functions and the feasibility sets. Specifically, we prove that the sequences of outer approximations and trial solutions converge almost surely to sample-average function and an optimal solution, respectively. Furthermore, we show that the sequence of lower bounds converges almost surely to the optimal SAA value. It is noted that our convergence results hold for any choice of stability centers and do not require the stage-wise independence assumption or the cut-sharing feature of SDDP, which make our results more general. However, the results rely on the assumption that the scenarios are sampled independently in the forward pass.

The remainder of the chapter is organized as follows. The next section provides a brief introduction to multistage stochastic linear programming models and the main steps of the standard SDDP and the SDDP with quadratic regularization algorithms. Section 4.3 introduces the proposed nonconvex regularization scheme for the SDDP algorithm. The equivalent MILP formulations for the nonconvex regularization problem are presented in Section 4.4. Section 4.5 presents the convergence analysis of the SDDP algorithm with nonconvex regularization. Finally, Section 4.6 provides computational results for two large-scale problems that illustrate the benefits of nonconvex regularization.

4.2 MULTISTAGE STOCHASTIC LINEAR PROGRAMMING

Here, we provide a brief description of multistage stochastic linear programming (MSLP) models where the underlying uncertainty has an uncountable number of realizations. First, we discuss the sample average approximation (SAA) scheme of solving such computationally intractable MSLP models.

4.2.1 Sample Average Approximation (SAA) Model

Consider a finite planning horizon $\mathcal{T} = \{1, 2, \ldots, T\}$, where $T \in \mathbb{N}$ is the number of decision stages and $t \in \mathcal{T}$ is the index of stage t. For future use, we also define $\hat{\mathcal{T}} \equiv \mathcal{T} \setminus \{1\}$ and $\check{\mathcal{T}} = \mathcal{T} \setminus \{T\}$. It is assumed that there is no uncertainty in the first stage, i.e., uncertainty is realized only from stage 2 onwards. Let $(\Omega, \mathscr{A}, \{\mathscr{A}_t\}_{t\in\mathcal{T}}, \mathbb{P})$ be a complete, filtered probability space with natural filtration $\{\mathscr{A}_t : t \in \mathcal{T}\}$, i.e., $\mathscr{A}_t$ contains the information available up to stage t. The uncertainty in stage $t \in \hat{\mathcal{T}}$ is described by a measurable mapping $\xi_t : (\Omega, \mathscr{A}) \to (\Xi_t, \mathcal{B}(\Xi_t))$, where $\Xi_t \subseteq \mathbb{R}^m$ is the m-dimensional support set of ξ_t, and $\mathcal{B}(\Xi_t)$ is the Borel σ-algebra on Ξ_t. For $\omega \in \Omega$, a realization of ξ_t is denoted by $\tilde{\xi}_t \equiv \xi_t(\omega)$. The uncertainty evolves as a discrete-time, continuous-state stochastic process $\xi = \{\xi_t : t \in \hat{\mathcal{T}}\}$ with support $\Xi \equiv \Xi_2 \times \cdots \times \Xi_T$. A scenario (or sample path) of ξ is a vector of the form $\xi(\omega) = (\xi_t(\omega) : t \in \hat{\mathcal{T}})$, which will be denoted by $\tilde{\xi}$ for notational brevity.

Let x_1 be the vector of first-stage decisions whose feasible region is the n_1-dimensional, convex polytope

$$\mathcal{X}_1 = \{x_1 \in \mathbb{R}^{n_1} : A_1 x_1 = b_1\},$$

where $A_1 \in \mathbb{R}^{k_1 \times n_1}$ and $b \in \mathbb{R}^{k_1}$ are deterministic, and let $\xi_t^h = \{\xi_i : i \in \hat{\mathcal{T}}, i \leq t\}$ be the history process up to stage t with support $\Xi_t^h \equiv \Xi_2 \times \cdots \times \Xi_t$. A feasible decision rule at stage t, denoted by $x_t \equiv x_t(\xi_t^h)$, is an $\mathscr{A}_t$-measurable function whose range is the n_t-dimensional, convex polytope

$$\mathcal{X}_t(x_{t-1}, \xi_t) = \{x_t \in \mathbb{R}^{n_t} : A_t x_t = b_t - B_t x_{t-1}\},$$

where $A_t \in \mathbb{R}^{k_t \times n_t}$, $B_t \in \mathbb{R}^{k_t \times n_{t-1}}$ and $b_t \in \mathbb{R}^{k_t}$ might all depend on ξ_t. The vector of cost coefficients at stage t is denoted by $c_t \equiv c_t(\xi_t) \in \mathbb{R}^{n_t}$, and we assume $|c_t' x_t| < \infty$ for each

$t \in \mathcal{T}$. A policy, $\pi = \{x_t : t \in \mathcal{T}\}$, is a set of feasible decision rules, and an optimal policy π^* is one that solves the following nested formulation of a general MSLP:

$$\begin{aligned} z^* = \min_{x_1 \in \mathcal{X}_1} \Bigg\{ c_1' x_1 + \mathbb{E}_{\xi_2} \Bigg(\min_{x_2 \in \mathcal{X}_2(x_1, \xi_2)} c_2' x_2 + \mathbb{E}_{\xi_3 | \xi_2^h} \Bigg(\min_{x_3 \in \mathcal{X}_3(x_2, \xi_3)} c_3' x_3 + \dots \\ \mathbb{E}_{\xi_T | \xi_{T-1}^h} \Bigg(\min_{x_T \in \mathcal{X}_T(x_{T-1}, \xi_T)} c_T' x_T \Bigg) \cdots \Bigg) \Bigg) \Bigg\}, \end{aligned} \tag{4.1}$$

where $\mathbb{E}_{\xi_t | \xi_{t-1}^h}$ is the conditional expectation taken with respect to the probability distribution of ξ_t, given ξ_{t-1}^h. Henceforth, we assume that ξ is stage-wise independent, i.e., $\xi_t \perp\!\!\!\perp \xi_{t-1}^h$ for all $t \in \hat{\mathcal{T}}$. We invoke this assumption only to simplify notation; it has no bearing on the main results. Bellman's optimality equations associated with problem (4.1) are

$$z^* = \min \left\{ c_1' x_1 + \mathcal{V}_2(x_1) : A_1 x_1 = b_1,\, x_1 \in \mathbb{R}^{n_1} \right\}, \qquad t = 1, \tag{4.2a}$$

$$V_t(x_{t-1}, \tilde{\xi}_t) = \min \left\{ c_t' x_t + \mathcal{V}_{t+1}(x_t) : A_t x_t = b_t - B_t x_{t-1},\, x_t \in \mathbb{R}^{n_t} \right\}, \quad t \in \hat{\mathcal{T}},\, \tilde{\xi}_t \in \Xi_t, \tag{4.2b}$$

where $\mathcal{V}_t(x_{t-1}) \equiv \mathbb{E}(V_t(x_{t-1}, \xi_t))$ is the value function at stage t, and $\mathcal{V}_{T+1}(x_T) = 0$ by definition. Suppose $\mathcal{J}_t \subset \Xi_t$ be a random (i.i.d) sample of realizations of ξ_t, where $N \equiv |\mathcal{J}_t| < \infty$, and let $\mathcal{J} \equiv \mathcal{J}_2 \times \cdots \times \mathcal{J}_T$ be a random sample comprising of N^{T-1} scenarios. Then the SAA of (4.2) consists of the following problems:

$$\hat{z}_\ell = \min \left\{ c_1' x_1 + \widehat{\mathcal{V}}_2(x_1) : A_1 x_1 = b_1,\, x_1 \in \mathbb{R}^{n_1} \right\}, \qquad t = 1, \tag{4.3a}$$

$$\widehat{V}_t(x_{t-1}, \tilde{\xi}_t) = \min \left\{ c_t' x_t + \widehat{\mathcal{V}}_{t+1}(x_t) : A_t x_t = b_t - B_t x_{t-1},\, x_t \in \mathbb{R}^{n_t} \right\}, \quad t \in \hat{\mathcal{T}}, \tilde{\xi}_t \in \mathcal{J}_t, \tag{4.3b}$$

where $\widehat{\mathcal{V}}_t(x_{t-1})$ is the sample-average estimate of $\mathcal{V}_t(x_{t-1})$ given by

$$\widehat{\mathcal{V}}_t(x_{t-1}) = \frac{1}{N} \sum_{\tilde{\xi}_t \in \mathcal{J}_t} \widehat{V}_t(x_{t-1}, \tilde{\xi}_t), \quad t \in \hat{\mathcal{T}}. \tag{4.4}$$

It is well known that $\hat{z}_\ell$ is a downward-biased estimator of z^*, i.e., $\mathbb{E}(\hat{z}_\ell) \leq z^*$, and is, therefore, a valid statistical lower bound of z^*. Henceforth, we refer to $\hat{z}_\ell$ as the SAA lower bound of z^*. Usually, the scenarios in $\mathcal{J}$ are stored as a scenario tree and (4.3) is solved using a scenario-tree formulation of (4.1). In the parlance of scenario trees, (4.3a) and (4.3b) are the *nodal problems* at stage 1 and stage t, respectively.

It has been shown (see [135]) that, under some mild regularity conditions, $\hat{z}_\ell$ is a consistent estimator of z^*, i.e., $\mathbb{E}(\hat{z}_\ell) \to z^*$ as $N \to \infty$. However, the size of a scenario tree grows exponentially in the number of stages T, rendering the SAA model intractable for even for small sample sizes. Next, we discuss how the stochastic dual dynamic programming (SDDP) algorithm can be used to solve large-scale SAA models.

4.2.2 SDDP Algorithm for the SAA Model

The SDDP algorithm uses a sampling-based decomposition strategy to generate a sequence of lower and upper bounds of the SAA optimal value $\hat{z}_\ell$. The algorithm begins with an initial approximation of $\widehat{\mathcal{V}}_t(x_{t-1})$ for each $t \in \hat{\mathcal{T}}$. Let $k \in \mathbb{N}$ denote the iteration index and $\widehat{\mathcal{V}}_t^{k-1}(x_{t-1})$ be the current approximation of $\widehat{\mathcal{V}}_t(x_{t-1})$ at the start of iteration k. Each iteration of the algorithm consists of a forward pass and a backward pass, which are described next.

The forward pass generates a statistical upper bound of $\hat{z}_\ell$ in the following manner. Let $\mathcal{J}^k \subseteq \mathcal{J}$ be a set of M scenario tree samples with $M \ll |\mathcal{J}|$. Starting at stage 1 and moving forward to stage T, the following nodal problem is solved for each $t \in \mathcal{T}$ and $\tilde{\xi} \in \mathcal{J}^k$:

$$\hat{x}_t^k(\tilde{\xi}_t^h) \in \operatorname{argmin} \left\{ c_t' x_t + \widehat{\mathcal{V}}_{t+1}^{k-1}(x_t) : A_t x_t = b_t - B_t \hat{x}_{t-1}^k(\tilde{\xi}_{t-1}^h),\, x_t \in \mathbb{R}^{n_t} \right\}, \tag{4.5}$$

where $\hat{x}_t^k(\tilde{\xi}_t^h)$ is a trial solution. Let $\hat{x}_t^k = (\hat{x}_t^k(\tilde{\xi}_t^h) : \tilde{\xi} \in \mathcal{J}^k)$ be the vector of all trial solutions at stage t, and note that $\hat{\pi}^k \equiv \{\hat{x}_t^k : t \in \mathcal{T}\}$ represents a partially-characterized policy induced by the current approximations. Henceforth, we refer to $\hat{\pi}^k$ as the induced policy at iteration k. The total cost incurred under $\hat{\pi}^k$ for scenario $\tilde{\xi}$ is $\hat{z}^k(\tilde{\xi}) \equiv \sum_{t \in \mathcal{T}} c_t' \hat{x}_t^k(\tilde{\xi}_t^h)$. The forward pass terminates on computing the sample-average estimate

$$\hat{z}_u^k = \frac{1}{M} \sum_{\tilde{\xi} \in \mathcal{J}^k} \hat{z}^k(\tilde{\xi}). \tag{4.6}$$

Note that $\hat{z}_u^k$ is a statistical upper bound of $\hat{z}_\ell$ as $\hat{z}_\ell \leq \mathbb{E}(\hat{z}_u^k)$. For $M > 1$, a more conservative upper bound of $\hat{z}_\ell$ is the term $(\hat{z}_u^k + \hat{t}_{\delta/2,\,M}\, \hat{\sigma}_k/\sqrt{M})$, where $\delta \in (0, 1]$ is a given significance level, $\hat{t}_{\delta/2,\,M}$ is the $(1 - \delta/2)$% quantile of a t-distribution with M degrees of freedom, and $\hat{\sigma}_k^2$ is the unbiased sample variance of $\hat{z}_u^k$, i.e.,

$$\hat{\sigma}_k^2 = \frac{1}{M-1} \sum_{\tilde{\xi} \in \mathcal{J}^k} \left(\hat{z}^k(\tilde{\xi}) - \hat{z}_u^k \right)^2. \tag{4.7}$$

Next, we describe the backward pass of the SDDP algorithm. The current approximations are improved by generating new Benders' cuts in the backward pass starting at stage T and moving backward in time to stage 1. A Benders' cut at stage t is constructed using a trial solution from stage $t-1$ and the optimal dual solutions of the stage t nodal problems. For notational convenience, denote $\hat{x}_t^k(\tilde{\xi}_t^h)$ simply as $\hat{x}_t^k$. At stage $t = T$, the following nodal problem is solved for each $\tilde{\xi}_T \in \mathcal{J}_T$:

$$\bar{V}_T^k(\hat{x}_{T-1}^k, \tilde{\xi}_T) = \min\left\{c_T' x_T : A_T x_T = b_T - B_T \hat{x}_{T-1}^k,\, x_T \in \mathbb{R}^{n_T}\right\}, \quad \tilde{\xi}_T \in \mathcal{J}_T. \tag{4.8}$$

The optimal dual vector of (4.8) is used to construct a Benders' cut of the form

$$\ell_T^k(x_{T-1}) = \hat{\alpha}_{T,k} + \hat{\beta}_{T,k}'(x_{T-1} - \hat{x}_{T-1}^k) \tag{4.9}$$

satisfying $\widehat{\mathcal{V}}_T(x_{T-1}) \geq \ell_T^k(x_{T-1})$, where $\hat{\alpha}_{T,k} = \mathbb{E}(\bar{V}_T^k(\hat{x}_{T-1}^k, \xi_T))$ and $\hat{\beta}_{T,k} \in \partial\widehat{\mathcal{V}}_T(\hat{x}_{T-1}^k)$. Note that $\widehat{\mathcal{V}}_T(\hat{x}_{T-1}^k) = \mathbb{E}(\bar{V}_T^k(\hat{x}_{T-1}^k, \xi_T))$ as $\widehat{\mathcal{V}}_{T+1}(x_T) = 0$. Next, the current approximation is updated via

$$\widehat{\mathcal{V}}_T^k(x_{T-1}) = \max\left\{\widehat{\mathcal{V}}_T^{k-1}(x_{T-1}), \ell_T^k(x_{T-1})\right\}. \tag{4.10}$$

Similarly, at an intermediate stage $t \in \{T-1, T-2, \ldots, 2\}$, the following nodal problem is solved for each $\tilde{\xi}_t \in \mathcal{J}_t$:

$$\bar{V}_t^k(\hat{x}_{t-1}^k, \tilde{\xi}_t) = \min\left\{c_t' x_t + \widehat{\mathcal{V}}_{t+1}^k(x_t) : A_t x_t = b_t - B_t \hat{x}_{t-1}^k,\, x_t \in \mathbb{R}^{n_t}\right\}. \tag{4.11}$$

A Benders' cut at stage t is of the form

$$\ell_t^k(x_{t-1}) = \hat{\alpha}_{t,k} + \hat{\beta}_{t,k}'(x_{t-1} - \hat{x}_{t-1}^k), \tag{4.12}$$

where $\hat{\alpha}_{t,k} = \mathbb{E}(\bar{V}_t^k(\hat{x}_{t-1}^k, \xi_t))$ and $\hat{\beta}_{t,k} \in \partial\mathbb{E}(\bar{V}_t(\hat{x}_{t-1}^k, \xi_t))$. the stage t approximation is updated via

$$\widehat{\mathcal{V}}_t^k(x_{t-1}) = \max\left\{\widehat{\mathcal{V}}_t^{k-1}(x_{t-1}), \ell_t^k(x_{t-1})\right\}. \tag{4.13}$$

Finally, the backward pass terminates on solving the stage 1 nodal problem,

$$\hat{z}_\ell^k = \min\left\{c_1' x_1 + \widehat{\mathcal{V}}_2^k(x_1) : A_1 x_1 = b_1,\, x_1 \in \mathbb{R}^{n_1}\right\}. \tag{4.14}$$

Because problem (4.14) is a relaxation of (4.3a), it follows immediately that $\hat{z}_\ell^k \leq \hat{z}_\ell$ almost surely (a.s.). That is, $\hat{z}_\ell^k$ is a valid lower bound of $\hat{z}_\ell$. The algorithm is terminated when the following convergence criterion is satisfied:

$$\left|\hat{z}_u^k + \hat{t}_{\delta/2,\,M}\,\hat{\sigma}_k/\sqrt{M} - \hat{z}_\ell^k\right| \leq \epsilon, \tag{4.15}$$

where $\epsilon > 0$ is a specified level of accuracy.

4.2.3 Quadratic Regularization for the SDDP Algorithm

The lower bounds generated by the SDDP algorithm converge slowly, in part, because many of the initial cuts are of poor quality. The convergence rate can be improved if good trial solutions can be explored initially to generate stronger cuts. Subsequent iterations can then be used to exploit these higher-quality trial solutions to stabilize the lower bounds. The quadratic regularization scheme of [7] is motivated by this tradeoff between exploration and exploitation of trial solutions. For each $t \in \check{\mathcal{T}} = \mathcal{T} \setminus \{T\}$, they proposed solving the forward-pass problem

$$\hat{x}_t^k \in \operatorname{argmin}\, \left\{ c_t' x_t + \widehat{\mathcal{V}}_{t+1}^{k-1}(x_t) + \lambda_k (x_t - \check{x}_t)' Q_t (x_t - \check{x}_t) : A_t x_t = b_t - B_t \hat{x}_{t-1}^k,\, x_t \in \mathbb{R}^{n_t} \right\}, \tag{4.16}$$

where Q_t is a positive semi-definite scaling matrix, λ_k is a nonnegative penalty parameter satisfying $\lim_{k\to\infty} \lambda_k = 0$, and $\check{x}_t$ is a known stability center. The quadratic term in (4.16) represents the scaled Euclidean (L_2) distance between x_t and $\check{x}_t$. [7] demonstrated significant improvement in the convergence rate when $\check{x}_t = \hat{x}_t^{k-1}$ and $\lambda_k = \lambda_0 \rho^k$, where $\lambda_0 > 0$ and $\rho \in (0, 1)$.

The quadratic regularization scheme of (4.16) has some disadvantages. Specifically, when the stability centers are initially poor (due to weak approximations), a high value of λ_k shrinks $\hat{x}_t^k$ towards $\check{x}_t$, inhibiting exploration of good trial solutions. Furthermore, as $\lambda_k \downarrow 0$, the contribution of the quadratic term in (4.16) is reduced, inhibiting exploitation of good trial solutions. That is, regularization is enforced when it is least needed and dismissed when it would be most beneficial. In the next section, we propose and analyze a new regularization scheme that overcomes this limitation of quadratic regularization.

4.3 NONCONVEX REGULARIZATION FOR THE SDDP ALGORITHM

Here, we introduce a novel regularization scheme for SDDP that is based on folded concave penalty (FCP) functions. Specifically, we consider two common FCP functions – the minimax concave penalty (MCP) and smoothly clipped absolute deviation (SCAD) – and show that, in either case, the regularized forward-pass problem is equivalent to a quadratic program. In what follows, λ_k is denoted simply as λ for notational brevity, ∇f is the first-order derivative of a real-valued function $f : \mathbb{R} \to \mathbb{R}$, and $\mathbb{I}_A(x)$ is the indicator function on an interval $A \subseteq \mathbb{R}$.

4.3.1 Regularization via Folded Concave Penalty

Let $f_\lambda : \mathbb{R}_+ \to \mathbb{R}_+$ be a general penalty function with $\lambda > 0$. The MCP and the SCAD functions, denoted by f_λ^M and f_λ^S, respectively, are defined as follows:

$$\text{(MCP)}: \quad f_\lambda^M(u) = \lambda \int_0^u \max\left\{1 - \frac{v}{a\lambda}, 0\right\} dv, \qquad u \in \mathbb{R}_+,\ a > 1,\ \lambda > 0, \tag{4.17}$$

$$\text{(SCAD)}: \quad f_\lambda^S(u) = \lambda \int_0^u \left(\mathbb{I}_{[0,\lambda]}(v) + \frac{(a\lambda - v)_+}{\lambda(a-1)}\,\mathbb{I}_{(\lambda,\infty)}(v)\right) dv, \quad u \in \mathbb{R}_+,\ a > 2,\ \lambda > 0, \tag{4.18}$$

where a is the shape parameter. Let $x_{t,i}$ and $\check{x}_{t,i}$ denote the ith components of x_t and $\check{x}_t$, respectively. Without loss of generality, assume that the scaling matrix $Q_t = I$ (the identity matrix). For each $t \in \check{\mathcal{T}}$, we propose solving the regularized forward-pass problem,

$$\hat{x}_t^k \in \operatorname{argmin} \left\{c_t' x_t + \widehat{\mathcal{V}}_{t+1}^{k-1}(x_t) + F_\lambda(x_t, \check{x}_t) : A_t x_t = b_t - B_t \hat{x}_{t-1}^k,\ x_t \in \mathbb{R}^{n_t}\right\}, \tag{4.19}$$

where

$$F_\lambda(x_t, \check{x}_t) = \sum_{i=1}^{n_t} f_\lambda(|x_{t,i} - \check{x}_{t,i}|), \tag{4.20}$$

with $f_\lambda = f_\lambda^M$ or $f_\lambda = f_\lambda^S$. Henceforth, we refer to (4.19) as the FCP regularization problem and F_λ as the regularization function. Note that F_λ penalizes the absolute deviations between the components of x_t and $\check{x}_t$. This regularization scheme benefits from these properties of the FCP functions:

(P1) f_λ is increasing, Lipschitz continuous, and concave on $\mathbb{R}_+$ with

$$f_\lambda(0) = 0;\;\; f_\lambda(a\lambda) < \infty;\;\; f_\lambda(u) \leq f_\lambda(a\lambda),\; \forall\, u \in [0, a\lambda);\;\; f_\lambda(u) = f_\lambda(a\lambda),\; \forall\, u \in [a\lambda, \infty);$$

(P2) ∇f_λ is decreasing and locally Lipschitz continuous on $\mathbb{R}_+$ with

$$0 < \nabla f_\lambda(u) \leq \lambda,\; \forall\, u \in (0, a\lambda);\;\; \nabla f_\lambda(u) = 0,\; \forall\, u \in (a\lambda, \infty);$$

(P3) For any $u \in \mathbb{R}_+$,

$$\begin{aligned} f_{\lambda_1}(u) &\leq f_{\lambda_2}(u), && \text{for } \lambda_1 < \lambda_2 \text{ and } a \text{ fixed;} \\ f_\lambda(u; a_1) &\leq f_\lambda(u; a_2), && \text{for } a_1 < a_2 \text{ and } \lambda \text{ fixed.} \end{aligned}$$

It is clear from (P1) that f_λ^M and f_λ^S are bounded functions. It follows immediately from (P1) that for all $|x_{t,i} - \check{x}_{t,i}| \geq a\lambda$, $f_\lambda(|x_{t,i} - \check{x}_{t,i}|) = f_\lambda(a\lambda)$. That is, large deviations are not excessively penalized by the FCP functions; hence, they promote exploration of trial solutions, particularly at the start of the algorithm. Furthermore, $f_\lambda(|x_{t,i} - \check{x}_{t,i}|) \downarrow 0$ as $\lambda \downarrow 0$ by (P1) and (P4), which promotes exploitation as the weight of the penalty term is reduced. The left-hand graph in Figure 19 depicts the contrasting behaviors of the FCP and quadratic functions. It is instructive to examine the properties of the penalization rate ∇f_λ, due to its importance in the KKT conditions for (4.19). It is clear from (P3) that $\nabla f_\lambda = 0$ for all $|x_{t,i} - \check{x}_{t,i}| \geq a\lambda$, and $\nabla f_\lambda \uparrow \lambda$ as $|x_{t,i} - \check{x}_{t,i}| \downarrow 0$. In fact, FCP functions penalize small deviations (close to zero) at a higher rate than the quadratic function, which aids in exploitation in later iterations. Interestingly, for a fixed λ, the penalization rates of the FCP functions are bounded by those of the L_1 penalty functions. The right-hand graph in Figure 19 depicts the penalization rates of different penalty functions.

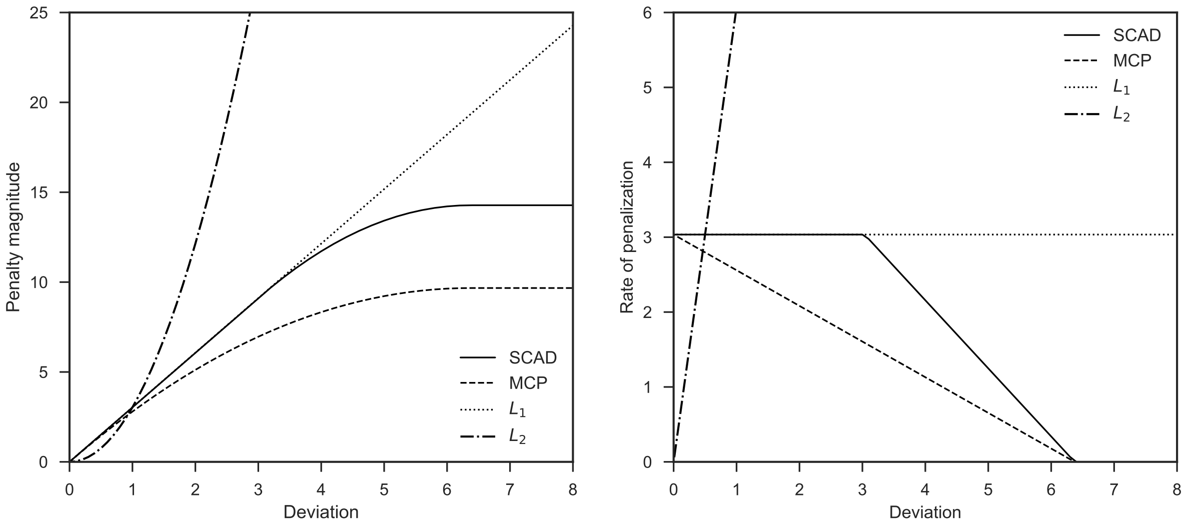


Figure 19: Magnitudes and penalization rates of the FCP, L_1, and L_2 functions.

4.3.2 Nonconvex Quadratic Programming Formulations

In this section, we derive equivalent quadratic programming (QP) formulations of (4.19) for each of the FCP functions. First, we introduce some notation. Let $\mathbf{1}$ be a column vector of ones whose dimension is inferred from the context. For each $t \in \{2, \dots, T\}$, collect the elements $\hat{\alpha}_{t,k}$ and $\hat{\beta}_{t,k}$ in the vector $\hat{\alpha}_t$ and the matrix $\hat{\beta}_t$, respectively. Then, the collection of Benders' cuts at stage $t+1$ can be represented in the vector form $\theta_{t+1}\mathbf{1} \geq \hat{\alpha}_{t+1} + \hat{\beta}'_{t+1} x_t$.

Unless otherwise noted, proofs of the results that follow are provided in the electronic companion. The following lemma provides closed-form expressions of the FCP functions.

Lemma 4.1. *For any $u \in \mathbb{R}_+$,*

$$(i) \quad f_\lambda^M(u) = \min\left\{\frac{v^2}{2a} - \frac{uv}{a} \,:\, v \in [0, a\lambda]\right\} + \lambda u = \left(u\lambda - \frac{u^2}{2a}\right)\mathbb{I}_{[0,a\lambda]}(u) + \left(\frac{a\lambda^2}{2}\right)\mathbb{I}_{(a\lambda\infty)}(u).$$

$$(ii) \quad f_\lambda^S(u) = \min\left\{(u - a\lambda)v + \left(\frac{a-1}{2}\right)v^2 \,:\, v \in [0, \lambda]\right\} + \lambda^2\left(\frac{a+1}{2}\right),$$

$$= u\lambda\mathbb{I}_{[0,\lambda]}(u) + \left(\frac{2au\lambda - \lambda^2 - u^2}{2(a-1)}\right)\mathbb{I}_{(\lambda,a\lambda]}(u) + \lambda^2\left(\frac{a+1}{2}\right)\mathbb{I}_{(a\lambda,\infty)}(u).$$

Proof. Proof. The proof mirrors that of Proposition 2.3 in [101] and is omitted here. □

Next, we show that F_λ can be expressed using the optimal value of a general quadratic program whose decision variables are independent of x_t and $\check{x}_t$.

Proposition 4.1. *The regularization function F_λ can be expressed as follows:*

(i) If $f_\lambda = f_\lambda^M$, then for any $x_t, \check{x}_t \in \mathbb{R}^{n_t}$,

$$F_\lambda(x_t, \check{x}_t) = \min\left\{\frac{y_t' y_t}{2a} - \frac{y_t'|x_t - \check{x}_t|}{a} \;:\; y_t \in [0, a\lambda\mathbf{1}]\right\} + \lambda\mathbf{1}'|x_t - \check{x}_t|; \tag{4.21}$$

(ii) If $f_\lambda = f_\lambda^S$, then for any $x_t, \check{x}_t \in \mathbb{R}^{n_t}$,

$$F_\lambda(x_t, \check{x}_t) = \min\left\{\left(\frac{a-1}{2}\right) y_t' y_t + y_t'|x_t - \check{x}_t| - a\lambda\mathbf{1}' y_t \;:\; y_t \in [0, \lambda\mathbf{1}]\right\} + \lambda^2\left(\frac{a+1}{2}\right) n_t. \tag{4.22}$$

Proof. To prove part (i), we define a vector $y_t \equiv (y_{t,i} : i = 1, \ldots, n_t) \in \mathbb{R}^{n_t}$ whose each component satisfies the inequality $0 \le y_{t,i} \le a\lambda$. Let $[0, a\lambda\mathbf{1}] \equiv \{y_t \in \mathbb{R}^{n_t} : 0 \le y_{t,i} \le a\lambda,\ i = 1, \ldots, n_t\}$. Using part (i) of Lemma 4.1, we express $F_\lambda(x_t, \check{x}_t)$ as

$$F_\lambda(x_t, \check{x}_t) = \sum_{i=1}^{n_t}\left(\min\left\{\frac{y_{t,i}^2}{2a} - \frac{y_{t,i}|x_{t,i} - \check{x}_{t,i}|}{a} \;:\; y_{t,i} \in [0, a\lambda]\right\} + \lambda|x_{t,i} - \check{x}_{t,i}|\right). \tag{4.23}$$

It is noted that there are no constraints in (4.23) that link the components $y_{t,i}$. Moreover, y_t is not constrained by x_t and $\check{x}_t$. Therefore, the min and the sum operators in (4.23) can be interchanged as follows:

$$\begin{aligned} F_\lambda(x_t, \check{x}_t) &= \min\left\{\sum_{i=1}^{n_t}\left(\frac{y_{t,i}^2}{2a} - \frac{y_{t,i}|x_{t,i} - \check{x}_{t,i}|}{a}\right) \;:\; y_{t,i} \in [0, a\lambda], i = 1, \ldots, n_t\right\} + \lambda\sum_{i=1}^{n_t}|x_{t,i} - \check{x}_{t,i}|, \\ &= \min\left\{\frac{y_t' y_t}{2a} - \frac{y_t'|x_t - \check{x}_t|}{a} \;:\; y_t \in [0, a\lambda\mathbf{1}]\right\} + \lambda\mathbf{1}'|x_t - \check{x}_t|, \end{aligned}$$

where the expression following the second equality is the vector notation of the first. This completes the proof of part (i). To prove part (ii), redefine y_t such that $0 \le y_{t,i} \le \lambda$ for

$i = 1, \ldots, n_t$, and let $[0, \lambda\mathbf{1}] \equiv \{y_t \in \mathbb{R}^{n_t} : 0 \leq y_{t,i} \leq \lambda, i = 1, \ldots, n_t\}$. Using part (ii) of Lemma 4.1, the regularization function can be expressed as

$$\begin{aligned} F_\lambda(x_t, \check{x}_t) &= \sum_{i=1}^{n_t} \left(\min \left\{ (|x_{t,i} - \check{x}_{t,i}| - a\lambda) y_{t,i} + \left(\frac{a-1}{2} \right) y_{t,i}^2 \, : \, y_{t,i} \in [0, \lambda] \right\} + \lambda^2 \left(\frac{a+1}{2} \right) \right), \\ &= \min \left\{ \sum_{i=1}^{n_t} \left((|x_{t,i} - \check{x}_{t,i}| - a\lambda) y_{t,i} + \left(\frac{a-1}{2} \right) y_{t,i}^2 \right) \, : \, y_t \in [0, \lambda\mathbf{1}] \right\} + \lambda^2 \left(\frac{a+1}{2} \right) n_t, \\ &= \min \left\{ \left(\frac{a-1}{2} \right) y_t' y_t + y_t' |x_t - \check{x}_t| - a\lambda \mathbf{1}' y_t \, : \, y_t \in [0, \lambda\mathbf{1}] \right\} + \lambda^2 \left(\frac{a+1}{2} \right) n_t, \end{aligned}$$

where the sum and min operators were interchanged using the exact arguments used in part (i). This completes the proof. □

Proposition 4.1 allows us to formulate the FCP regularization problem as a quadratic program. Distinct formulations that use the MCP and SCAD functions are given in Propositions 4.2 and 4.3, respectively.

Proposition 4.2. *Suppose $f_\lambda = f_\lambda^M$. For each $t \in \check{\mathcal{T}}$, the FCP regularization problem (4.19) is equivalent to the following quadratic program:*

$$\min \; c_t' x_t + \theta_{t+1} + \frac{y_t' y_t}{2a} - \frac{y_t' h_t}{a} + \lambda \mathbf{1}' h_t \tag{4.24a}$$

$$\text{s.t.} \; A_t x_t = b_t - B_t \hat{x}_{t-1}, \tag{4.24b}$$

$$\theta_{t+1} \mathbf{1} \geq \hat{\alpha}_{t+1} + \hat{\beta}_{t+1}' x_t, \tag{4.24c}$$

$$h_t \geq x_t - \check{x}_t, \tag{4.24d}$$

$$h_t \geq \check{x}_t - x_t, \tag{4.24e}$$

$$y_t \leq a\lambda \mathbf{1}, \tag{4.24f}$$

$$y_t \geq 0, \tag{4.24g}$$

$$x_t, y_t, h_t \in \mathbb{R}^{n_t}, \theta_{t+1} \in \mathbb{R}. \tag{4.24h}$$

Proof. Define $h_t \equiv |x_t - \check{x}_t|$. Using the expression in part (i) of Proposition 4.1 for $F_\lambda(x_t, \check{x}_t)$ in (4.19), we obtain

$$\begin{aligned} & c_t' x_t + \theta_{t+1} + \min \left\{ \frac{y_t' y_t}{2a} - \frac{y_t' h_t}{a} \, : \, y_t \in [0, a\lambda\mathbf{1}] \right\} + \lambda \mathbf{1}' h_t, \\ =& \min \left\{ c_t' x_t + \theta_{t+1} + \frac{y_t' y_t}{2a} - \frac{y_t' h_t}{a} + \lambda \mathbf{1}' h_t : y_t \in [0, a\lambda\mathbf{1}] \right\}, \end{aligned}$$

where the equality holds because y_t is not constrained by x_t, h_t and θ_{t+1}. Let $\mathcal{X}_t$ denote the feasibility set of (4.19). Then, (4.19) can be expressed as the following minimization problem:

$$\begin{aligned}&\min\left\{\min\left\{c_t'x_t+\theta_{t+1}+\frac{y_t'y_t}{2a}-\frac{y_t'h_t}{a}+\lambda\mathbf{1}'h_t\ :\ y_t\in[0,a\lambda\mathbf{1}]\right\}\ :\ (x_t,\theta_{t+1})\in\mathcal{X}_t, h_t=|x_t-\check{x}_t|\right\},\\ =&\min\left\{c_t'x_t+\theta_{t+1}+\frac{y_t'y_t}{2a}-\frac{y_t'h_t}{a}+\lambda\mathbf{1}'h_t\ :\ y_t\in[0,a\lambda\mathbf{1}],(x_t,\theta_{t+1})\in\mathcal{X}_t, h_t=|x_t-\check{x}_t|\right\}.\end{aligned}\tag{4.25}$$

It is noted that the nonconvex constraint $h_t = |x_t - \check{x}_t|$ in (4.25) can be replaced by an equivalent system of linear inequalities $\{h_t \geq x_t - \check{x}_t, h_t \geq \check{x}_t - x_t\}$, and the proof is complete. □

Proposition 4.3. *Suppose $f_\lambda = f_\lambda^S$. For each $t \in \check{\mathcal{T}}$, the FCP regularization problem (4.19) is equivalent to the following quadratic program:*

$$\min\ c_t'x_t+\theta_{t+1}+\left(\frac{a-1}{2}\right)y_t'y_t+y_t'h_t-a\lambda\mathbf{1}'y_t \tag{4.26a}$$

$$\text{s.t.}\ A_tx_t=b_t-B_t\hat{x}_{t-1}, \tag{4.26b}$$

$$\theta_{t+1}\mathbf{1}\geq\hat{\alpha}_{t+1}+\hat{\beta}_{t+1}'x_t, \tag{4.26c}$$

$$h_t\geq x_t-\check{x}_t, \tag{4.26d}$$

$$h_t\geq\check{x}_t-x_t, \tag{4.26e}$$

$$y_t\leq\lambda\mathbf{1}, \tag{4.26f}$$

$$y_t\geq 0, \tag{4.26g}$$

$$x_t,y_t,h_t\in\mathbb{R}^{n_t},\theta_{t+1}\in\mathbb{R}. \tag{4.26h}$$

Proof. We use the expression in part (ii) of Proposition 4.1 for $F_\lambda(x_t, \check{x}_t)$ in (4.19). The remainder of the proof is similar to that of Proposition 4.2 and is omitted here. □

The following result asserts that formulations (4.24) and (4.26) are nonsmooth, nonconvex quadratic programs.

Proposition 4.4. *The quadratic programs (4.24) and (4.26) are nonconvex. Furthermore, for each $\check{x}_t$, the regularization function $F_\lambda(x_t, \check{x}_t)$ is nonsmooth in x_t.*

Proof. First, we show that model (4.24) is nonconvex. The result is proved using a contradiction argument. In what follows, denote an identity matrix of dimension $n \times n$ by $I_{n\times n}$ and a zero matrix (of any dimension) by 0. Let $\Pi_t(\theta_{t+1}, x_t, y_t, h_t)$ denote the objective function of (4.24). Suppose that (4.24) is a convex problem. From the second-order necessary condition for convexity, it immediately follows that the Hessian matrix of Π_t, denoted by H, is positive semi-definite (PSD), i.e.,

$$H = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & H_1 & H_2 \\ 0 & 0 & H_2 & 0 \end{bmatrix} \succeq 0,$$

where H_1 and H_2 are invertible matrices with the following definitions:

$$H_1 \equiv \frac{\partial^2 \Pi_t}{\partial y_t^2} = \left(\frac{1}{a}\right) I_{n_t \times n_t}; \quad H_2 \equiv \frac{\partial^2 \Pi_t}{\partial y_t \partial h_t} = -\left(\frac{1}{a}\right) I_{n_t \times n_t}.$$

As any symmetric sub-matrix of a PSD matrix is also PSD, we have

$$B \equiv \begin{bmatrix} H_1 & H_2 \\ H_2 & 0 \end{bmatrix} \succeq 0.$$

Further, by Schur's complementarity condition, it follows that $B \succeq 0$ only if

$$-H_2' H_1^{-1} H_2 \succeq 0. \tag{4.27}$$

Substituting the values of H_1 and H_2 in (4.27), we obtain

$$-\left(-\frac{1}{a}\right) I'_{n_t \times n_t} \left(\left(\frac{1}{a}\right) I_{n_t \times n_t}\right)^{-1} \left(-\frac{1}{a}\right) I_{n_t \times n_t} \succeq 0 \implies -\left(\frac{1}{a}\right) I_{n_t \times n_t} \succeq 0, \tag{4.28}$$

which implies that $a < 0$. This contradicts the fact that $a > 1$ in the definition of f_λ^M in (4.17). Therefore, (4.24) is not convex. Similarly, for model (4.26), it can be shown that the corresponding Schur's complementarity condition

$$-\left(\frac{1}{a-1}\right) I_{n_t \times n_t} \succeq 0$$

cannot hold as $a > 2$ in the definition of f_λ^S in (4.18), and the proof is complete.

Next, we show that $F_\lambda(x_t, \check{x}_t)$ is nonsmooth in x_t for a given $\check{x}_t$. We show that F_λ is not differentiable at the point $x_t = \check{x}_t$. First, we consider the case $f_\lambda = f_\lambda^M$. Fix a component $i \in \{1, \ldots, n_t\}$ of x_t, and let $g_\lambda(x_{t,i}) \equiv f_\lambda(|x_{t,i} - \check{x}_{t,i}|)$. Using part (i) of Lemma 4.1, we express $g_\lambda(x_{t,i})$ as

$$g_\lambda(x_{t,i}) = \begin{cases} \left(\lambda(x_{t,i} - \check{x}_{t,i}) - \frac{(x_{t,i}-\check{x}_{t,i})^2}{2a}\right) \mathbb{I}_{[0,a\lambda]}(x_{t,i} - \check{x}_{t,i}) + \left(\frac{a\lambda^2}{2}\right) \mathbb{I}_{(a\lambda\infty)}(x_{t,i} - \check{x}_{t,i}), & x_{t,i} \geq \check{x}_{t,i} \\ \left(\lambda(\check{x}_{t,i} - x_{t,i}) - \frac{(x_{t,i}-\check{x}_{t,i})^2}{2a}\right) \mathbb{I}_{[0,a\lambda]}(\check{x}_{t,i} - x_{t,i}) + \left(\frac{a\lambda^2}{2}\right) \mathbb{I}_{(a\lambda\infty)}(\check{x}_{t,i} - x_{t,i}), & x_{t,i} \leq \check{x}_{t,i} \end{cases}.$$

Next, we show that $g_\lambda(x_{t,i})$ is not differentiable at $x_{t,i} = \check{x}_{t,i}$. The left-hand derivative of g_λ at $x_{t,i} = \check{x}_{t,i}$ is

$$\nabla g_\lambda^-(\check{x}_{t,i}) \equiv \lim_{x_{t,i} \to \check{x}_{t,i}^-} \left\{ \frac{g_\lambda(x_{t,i}) - g_\lambda(\check{x}_{t,i})}{x_{t,i} - \check{x}_{t,i}} \right\} = \lim_{x_{t,i} \to \check{x}_{t,i}^-} \left\{ -\lambda - \frac{x_{t,i} - \check{x}_{t,i}}{a} \right\} = -\lambda,$$

while the corresponding right-hand derivative is

$$\nabla g_\lambda^+(\check{x}_{t,i}) \equiv \lim_{x_{t,i} \to \check{x}_{t,i}^+} \left\{ \frac{g_\lambda(x_{t,i}) - g_\lambda(\check{x}_{t,i})}{x_{t,i} - \check{x}_{t,i}} \right\} = \lim_{x_{t,i} \to \check{x}_{t,i}^+} \left\{ \lambda - \frac{x_{t,i} - \check{x}_{t,i}}{a} \right\} = \lambda.$$

As $\nabla g_\lambda^-(\check{x}_{t,i}) \neq \nabla g_\lambda^+(\check{x}_{t,i})$, it follows that $g_\lambda(x_{t,i})$ is not differentiable at $x_{t,i} = \check{x}_{t,i}$. It can be shown that identical values of $\nabla g_\lambda^-(\check{x}_{t,i})$ and $\nabla g_\lambda^+(\check{x}_{t,i})$ are obtained for the case $f_\lambda = f_\lambda^S$. Thus, $g_\lambda(x_{t,i})$ is a non-smooth function. As the sum of non-smooth functions is non-smooth, we conclude that $F_\lambda(x_t, \check{x}_t)$ is a non-smooth function and the proof is complete. □

Nonconvexity stems from the fact that $F_\lambda(x_t, \check{x}_t)$ is concave in x_t for a given $\check{x}_t$. [18] recently showed that such nonsmooth, nonconvex quadratic programming (NSNC-QP) problems are NP-hard. Although sequential QP methods can be used to solve NSNC-QP problems (see [19] and [58]), they do not guarantee global optimal solutions and can be computationally slow ([111, 35]). In Section 4.4, we show that globally optimal solutions to problems (4.24) and (4.26) are attainable via mixed-integer linear programming (MILP).

4.4 MIXED INTEGER LINEAR PROGRAMMING FORMULATIONS

Our aim here is to establish that problem (4.19) admits equivalent MILP formulations that can be solved to global optimality within finitely many iterations. These formulations are inspired by the work of [55], who showed that a general quadratic program can be expressed as an equivalent linear program with complementarity constraints.

Let $d_t \equiv b_t - B_t\hat{x}_{t-1}$, and denote the current number of Benders' cuts at stage t by C. Next, we define $\pi_t \in \mathbb{R}^{k_t}$, $\mu_t \in \mathbb{R}^C$, $\nu_t^1, \nu_t^2 \in \mathbb{R}^{n_t}$, and $\delta_t^1, \delta_t^2 \in \mathbb{R}^{n_t}$ as the dual vectors associated with constraints (4.24b), (4.24c), (4.24d)–(4.24e), and (4.24f)–(4.24g), respectively; similar notation is adopted for the dual vectors linked to (4.26b)–(4.26g). A complementarity constraint involving two vectors $u, v \in \mathbb{R}^n$ is denoted by $u \perp v$ and is defined as $u \perp v \equiv \{u \geq 0,\, v \geq 0,\, u_i v_i = 0,\ \forall i = 1, \ldots, n\}$. Our first result reveals that model (4.24) is equivalent to a linear program with complementarity constraints (LPCC). Likewise, an LPCC model can be derived for model (4.26).

Proposition 4.5. *Suppose $f_\lambda = f_\lambda^M$. For each $t \in \check{\mathcal{T}}$, the QP (4.24) is equivalent to the LPCC,*

$$\min\ c_t' x_t + \theta_{t+1} - a\lambda \mathbf{1}'\delta_t^1 - \pi_t' d_t + \mu_t'\hat{\alpha}_{t+1} + (\nu_t^2 - \nu_t^1)'\check{x}_t + \lambda \mathbf{1}' h_t \tag{4.29a}$$

$$\text{s.t.}\ \ \text{constraints}\,(4.24\text{b}) - (4.24\text{g}), \tag{4.29b}$$

$$A_t'\pi_t + \hat{\beta}_{t+1}\mu_t + \nu_t^1 - \nu_t^2 + c_t = 0, \tag{4.29c}$$

$$y_t - h_t + a\delta_t^1 - a\delta_t^2 = 0, \tag{4.29d}$$

$$y_t + a\nu_t^1 + a\nu_t^2 = a\lambda\mathbf{1}, \tag{4.29e}$$

$$\mathbf{1}'\mu_t = 1, \tag{4.29f}$$

$$\delta_t^1 \perp (a\lambda\mathbf{1} - y_t);\ \delta_t^2 \perp y_t, \tag{4.29g}$$

$$\nu_t^1 \perp (\check{x}_t - x_t + h_t);\ \nu_t^2 \perp (x_t - \check{x}_t + h_t), \tag{4.29h}$$

$$\mu_t \perp (\theta_{t+1}\mathbf{1} - \hat{\alpha}_{t+1} - \hat{\beta}_{t+1}' x_t), \tag{4.29i}$$

$$x_t, y_t, h_t \in \mathbb{R}^{n_t},\ \theta_{t+1} \in \mathbb{R}, \tag{4.29j}$$

$$\pi_t \in \mathbb{R}^{k_t}, \delta_t^1, \delta_t^2, \nu_t^1, \nu_t^2 \in \mathbb{R}_+^{n_t}, \mu_t \in \mathbb{R}_+^C. \tag{4.29k}$$

Proof. The result is proved using the KKT optimality conditions. Let $w_t \equiv (x_t, y_t, h_t, \theta_{t+1})$ and $s_t \equiv (\pi_t, \mu_t, \nu_t^1, \nu_t^2, \delta_t^1, \delta_t^2)$. The Lagrangian of (4.24) is

$$L(w_t, s_t) = c_t' x_t + \theta_{t+1} + \frac{y_t' y_t}{2a} - \frac{y_t' h_t}{a} + \lambda \mathbf{1}' h_t + (A_t x_t - d_t)' \pi_t + (\hat{\alpha}_{t+1} + \hat{\beta}_{t+1}' x_t - \theta_{t+1} \mathbf{1})' \mu_t$$
$$+ (x_t - \check{x}_t - h_t)' \nu_t^1 + (\check{x}_t - x_t - h_t)' \nu_t^2 + (y_t - a\lambda \mathbf{1})' \delta_t^1 - y_t' \delta_t^2.$$

The linear independence constraint qualification conditions are satisfied by (4.24) because the constraints (4.24b)–(4.24g) are affine functions of w_t and A_t is a full rank matrix under the relatively complete recourse assumption for (4.1). A local optimal solution of (4.24) satisfies the following first-order KKT conditions:

(i) *KKT first-order stationarity conditions.*

$$\partial L / \partial x_t = 0 \qquad \longrightarrow \qquad c_t + A_t' \pi_t + \hat{\beta}_{t+1} \mu_t + \nu_t^1 - \nu_t^2 = 0, \tag{4.30a}$$

$$\partial L / \partial y_t = 0 \qquad \longrightarrow \qquad \frac{y_t}{a} - \frac{h_t}{a} + \delta_t^1 - \delta_t^2 = 0, \tag{4.30b}$$

$$\partial L / \partial h_t = 0 \qquad \longrightarrow \qquad -\frac{y_t}{a} + \lambda \mathbf{1} - \nu_t^1 - \nu_t^2 = 0, \tag{4.30c}$$

$$\partial L / \partial \theta_{t+1} = 0 \qquad \longrightarrow \qquad \mathbf{1}' \mu_t = 1. \tag{4.30d}$$

(ii) *KKT complementary slackness conditions (for inequality constraints).*

$$\mu_t \perp \left(\theta_{t+1} \mathbf{1} - \hat{\alpha}_{t+1} - \hat{\beta}_{t+1}' x_t \right), \tag{4.31a}$$

$$\delta_t^1 \perp (a\lambda \mathbf{1} - y_t), \tag{4.31b}$$

$$\delta_t^2 \perp y_t, \tag{4.31c}$$

$$\nu_t^1 \perp (\check{x}_t - x_t + h_t), \tag{4.31d}$$

$$\nu_t^2 \perp (x_t - \check{x}_t + h_t). \tag{4.31e}$$

(iii) *KKT primal and dual feasibility conditions.*

$$\text{constraints } (4.24\text{b}) - (4.24\text{g}), \tag{4.32a}$$

$$\mu_t \in \mathbb{R}_+^C, \delta_t^1, \delta_t^2, \nu_t^1, \nu_t^2 \in \mathbb{R}_+^{n_t}, \ \pi_t \in \mathbb{R}^{k_t}. \tag{4.32b}$$

It is noted that adding constraints (4.30a)–(4.32b) to the feasibility set of (4.24) results in an equivalent QP model because the KKT conditions are necessary for local optimality. Next, we show that the objective function of this equivalent QP model can be linearized. Let Π_t denote the objective function (4.24a). Then,

$$\begin{aligned}
\Pi_t &= c_t'x_t + \theta_{t+1} + \frac{y_t'y_t}{2a} - \frac{y_t'h_t}{a} + \lambda\mathbf{1}'h_t, \\
&= c_t'x_t + \theta_{t+1} + \frac{y_t'}{2}\left(\frac{y_t}{a} - \frac{h_t}{a}\right) + \left(\lambda\mathbf{1}' - \frac{y_t'}{a}\right)\frac{h_t}{2} + \frac{\lambda\mathbf{1}'h_t}{2}, \\
&= c_t'x_t + \theta_{t+1} + \frac{y_t'(\delta_t^2 - \delta_t^1)}{2} + \frac{(\nu_t^1 + \nu_t^2)'h_t}{2} + \frac{\lambda\mathbf{1}'h_t}{2},
\end{aligned} \tag{4.33}$$

where the third and fourth terms in (4.33) are obtained via (4.30b) and (4.30c), respectively. Next, we simplify the quadratic terms $y_t'(\delta_t^2 - \delta_t^1)$ and $(\nu_t^1 + \nu_t^2)'h_t$ in (4.33). Adding (4.31b) to (4.31c) and using the fact that $u \perp v \implies u'v = 0$, we obtain

$$y_t'(\delta_t^2 - \delta_t^1) = -a\lambda\mathbf{1}'\delta_t^1.$$

Similarly, on adding (4.31d) to (4.31e), we have

$$(\nu_t^1 + \nu_t^2)'h_t = (\nu_t^1 - \nu_t^2)'x_t + (\nu_t^2 - \nu_t^1)'\breve{x}_t.$$

Substituting the values of $y_t'(\delta_t^2 - \delta_t^1)$ and $(\nu_t^1 + \nu_t^2)'h_t$ in (4.33), we obtain

$$\Pi_t = c_t'x_t + \theta_{t+1} - \frac{a\lambda\mathbf{1}'\delta_t^1}{2} + \frac{(\nu_t^1 - \nu_t^2)'x_t}{2} + \frac{(\nu_t^2 - \nu_t^1)'\breve{x}_t}{2} + \frac{\lambda\mathbf{1}'h_t}{2}. \tag{4.34}$$

Next, we linearize the quadratic term $(\nu_t^1 - \nu_t^2)'x_t$ in (4.34) as follows:

$$\begin{aligned}
(\nu_t^1 - \nu_t^2)'x_t &= \left(-c_t - A_t'\pi_t - \hat{\beta}_{t+1}\mu_t\right)'x_t, && \text{(using (4.30a))} \\
&= -c_t'x_t - \pi_t'd_t - \mu_t'\hat{\beta}_{t+1}'x_t, && \text{(using } A_tx_t = d_t) \\
&= -c_t'x_t - \pi_t'd_t + \mu_t'\hat{\alpha}_{t+1} - \mu_t'\theta_{t+1}\mathbf{1}, && \text{(using (4.31a))} \\
&= -c_t'x_t - \pi_t'd_t + \mu_t'\hat{\alpha}_{t+1} - \theta_{t+1}. && \text{(using (4.30d))}
\end{aligned}$$

Substituting the value of $(\nu_t^1 - \nu_t^2)'x_t$ in (4.34) and after some algebra, we obtain

$$\Pi_t = 0.5\left(c_t'x_t + \theta_{t+1} - a\lambda\mathbf{1}'\delta_t^1 - \pi_t'd_t + \mu_t'\hat{\alpha}_{t+1} + (\nu_t^2 - \nu_t^1)'\breve{x}_t + \lambda\mathbf{1}'h_t\right). \tag{4.35}$$

We drop the constant 0.5 in (4.35) as it bears no effect on the optimal solutions of (4.24), and the proof is complete. □

Proposition 4.6. *Suppose $f_\lambda = f_\lambda^S$. For each $t \in \check{\mathcal{T}}$, the QP (4.26) is equivalent to the LPCC,*

$$\min \; c_t' x_t + \theta_{t+1} - \lambda \mathbf{1}'(\delta_t^1 + a y_t) - \pi_t' d_t + \mu_t' \hat{\alpha}_{t+1} + (\nu_t^2 - \nu_t^1)' \check{x}_t \tag{4.36a}$$

$$\text{s.t.} \;\; \text{constraints}\,(4.26\text{b}) - (4.26\text{g}), \tag{4.36b}$$

$$A_t' \pi_t + \hat{\beta}_{t+1} \mu_t + \nu_t^1 - \nu_t^2 + c_t = 0, \tag{4.36c}$$

$$(a-1) y_t + h_t - a\lambda \mathbf{1} + \delta_t^1 - \delta_t^2 = 0, \tag{4.36d}$$

$$y_t - \nu_t^1 - \nu_t^2 = 0, \tag{4.36e}$$

$$\mathbf{1}' \mu_t = 1, \tag{4.36f}$$

$$\delta_t^1 \perp (\lambda \mathbf{1} - y_t); \;\; \delta_t^2 \perp y_t, \tag{4.36g}$$

$$\nu_t^1 \perp (\check{x}_t - x_t + h_t); \;\; \nu_t^2 \perp (x_t - \check{x}_t + h_t), \tag{4.36h}$$

$$\mu_t \perp (\theta_{t+1} \mathbf{1} - \hat{\alpha}_{t+1} - \hat{\beta}_{t+1}' x_t), \tag{4.36i}$$

$$x_t, y_t, h_t \in \mathbb{R}^{n_t}, \theta_{t+1} \in \mathbb{R}, \tag{4.36j}$$

$$\pi_t \in \mathbb{R}^{k_t}, \delta_t^1, \delta_t^2, \nu_t^1, \nu_t^2 \in \mathbb{R}_+^{n_t}, \mu_t \in \mathbb{R}_+^C. \tag{4.36k}$$

Next, we show that the complementarity constraints in (4.29) and (4.36) can be equivalently represented using a system of linear inequality constraints (using the so-called big-Λ method).

Proposition 4.7. *For each $t \in \check{\mathcal{T}}$ and some $\Lambda_t \in (0, \infty)$,*

(i) the complementarity constraints (4.29g)–(4.29i) are equivalent to

$$a\lambda \mathbf{1} - y_t \le \Lambda_t q_t^1; \quad \delta_t^1 \le \Lambda_t (\mathbf{1} - q_t^1), \tag{4.37a}$$

$$y_t \le \Lambda_t q_t^2; \quad \delta_t^2 \le \Lambda_t (\mathbf{1} - q_t^2), \tag{4.37b}$$

$$\check{x}_t - x_t + h_t \le \Lambda_t v_t^1; \quad \nu_t^1 \le \Lambda_t (\mathbf{1} - v_t^1), \tag{4.37c}$$

$$x_t - \check{x}_t + h_t \le \Lambda_t v_t^2; \quad \nu_t^2 \le \Lambda_t (\mathbf{1} - v_t^2), \tag{4.37d}$$

$$\theta_{t+1} \mathbf{1} - \hat{\alpha}_{t+1} - \hat{\beta}_{t+1}' x_t \le \Lambda_t u_t; \quad \mu_t \le \Lambda_t (\mathbf{1} - u_t), \tag{4.37e}$$

$$q_t^1, q_t^2, v_t^1, v_t^2 \in \{0,1\}^{n_t}, \; u_t \in \{0,1\}^C. \tag{4.37f}$$

(ii) the complementarity constraints (4.36g)–(4.36i) *are equivalent to*

$$\lambda \mathbf{1} - y_t \leq \Lambda_t q_t^1; \quad \delta_t^1 \leq \Lambda_t(\mathbf{1} - q_t^1), \tag{4.38a}$$

$$(4.37b) - (4.37f). \tag{4.38b}$$

It immediately follows from Proposition 4.7 that the LPCC models (4.29) and (4.36) can be formulated as MILP problems that can be solved using branch-and-bound (BAB) methods that ensure global optimality (see [73]). These MILP formulations are amenable to solution by commercial solvers (e.g., Gurobi or CPLEX) that implement highly scalable variants of the standard BAB (see [123]), thereby obviating the need for specialized algorithms to solve (4.19). Next, we formally present the equivalent MILP formulations in Theorems 4.1 and 4.2.

Theorem 4.1. *Suppose $f_\lambda = f_\lambda^M$. For each $t \in \check{\mathcal{T}}$, a global optimal solution of problem* (4.19) *can be obtained with finitely many iterations by solving the equivalent MILP model,*

$$\begin{aligned} \min\ & c_t' x_t + \theta_{t+1} - a\lambda \mathbf{1}'\delta_t^1 - \pi_t' d_t + \mu_t'\hat{\alpha}_{t+1} + (\nu_t^2 - \nu_t^1)'\check{x}_t + \lambda \mathbf{1}' h_t && (4.39a)\\ \text{s.t. } & (4.29b) - (4.29f), && (4.39b)\\ & (4.37a) - (4.37f). && (4.39c) \end{aligned}$$

Formulation (4.39) follows directly from Proposition 4.5 and part (i) of Proposition 4.7. The global optimal solution of the MILP model is attainable via branch-and-bound ([109]).

Theorem 4.2. *Suppose $f_\lambda = f_\lambda^S$. For each $t \in \check{\mathcal{T}}$, a global optimal solution of problem* (4.19) *can be obtained with finitely many iterations by solving the equivalent MILP model,*

$$\begin{aligned} \min\ & c_t' x_t + \theta_{t+1} - \lambda \mathbf{1}'(\delta_t^1 + a y_t) - \pi_t' d_t + \mu_t'\hat{\alpha}_{t+1} + (\nu_t^2 - \nu_t^1)'\check{x}_t && (4.40a)\\ \text{s.t. } & (4.36b) - (4.36f), && (4.40b)\\ & (4.38a) - (4.38b). && (4.40c) \end{aligned}$$

Formulation (4.40) follows directly from Proposition 4.6 and part (ii) of Proposition 4.7. Likewise, a global optimal solution of the MILP model can be obtained using the branch-and-bound method.

Note that, if the KKT dual multipliers of (4.29) and (4.36) are unbounded, then the corresponding MILP models (4.39) and (4.40) can be numerically unstable ([32]). To address this issue, we first examine the NCNS-QP model (4.24) and the LPCC model (4.29) associated with the MCP function. Let $w_t \equiv (x_t, y_t, h_t, \theta_{t+1})$ and $s_t \equiv (\delta_t^1, \delta_t^2, \nu_t^1, \nu_t^2, \mu_t)$ be the vector of primal and dual variables in (4.29), respectively, and let w_t^* and s_t^* be the global optimal vectors. Further, let $\mathcal{W}_t$ and $\Pi_t(w_t)$ be the feasibility set and the objective function of (4.24), respectively. Note that w_t^* is bounded because $\mathcal{W}_t$ is compact; however, as strong duality does not hold for (4.24), s_t^* may not necessarily be bounded. Nonetheless, for compact $\mathcal{W}_t$, it can be shown that the first-order partial derivatives of Π_t are bounded on $\mathcal{W}_t$, i.e., there exists some $\Psi_t \in \mathbb{R}_+$ such that for all $w_t \in \mathcal{W}_t$,

$$\max\left\{\|\nabla_{x_t}\Pi_t(w_t)\|_1, \|\nabla_{y_t}\Pi_t(w_t)\|_1, \|\nabla_{h_t}\Pi_t(w_t)\|_1, \|\nabla_{\theta_{t+1}}\Pi_t(w_t)\|_1\right\} \leq \Psi_t, \tag{4.41}$$

where the L_1 norm $\|\cdot\|_1$ is chosen without loss of generality. In what follows, we choose a value of Ψ_t that is no less than the maximum number of unique Benders' cuts that can be generated at stage t (which is finite by Lemma 1 in [118]). Our next result reveals that the optimal dual vector cannot be arbitrarily large under the condition given in (4.41).

Proposition 4.8. *For each $t \in \check{\mathcal{T}}$, the components in s_t^* satisfy the inequalities*

$$\max\{\|\nu_t^{1*}\|_1, \|\nu_t^{2*}\|_1, \|\delta_t^{1*}\|_1, \|\delta_t^{2*}\|_1, \|\mu_t^*\|_1\} \leq \Psi_t; \qquad \|A_t'\pi_t^*\|_1 \leq (3+\Gamma_t)\Psi_t, \tag{4.42}$$

where $\Gamma_t \in [0,\infty)$ depends on $\hat{\beta}_{t+1}$.

Proof. The vectors w_t^* and s_t^* satisfy the KKT conditions for (4.24). We rewrite the KKT stationarity conditions in terms of the partial derivatives in (4.41) as follows:

$$\nabla_{x_t}\Pi_t(w_t) + A_t'\pi_t + \hat{\beta}_{t+1}\mu_t + \nu_t^1 - \nu_t^2 = 0, \tag{4.43a}$$

$$\nabla_{y_t}\Pi_t(w_t) + \delta_t^1 - \delta_t^2 = 0, \tag{4.43b}$$

$$\nabla_{h_t}\Pi_t(w_t) - \nu_t^1 - \nu_t^2 = 0, \tag{4.43c}$$

$$\nabla_{\theta_{t+1}}\Pi_t(w_t) - 1 = 0 \implies \mathbf{1}'\mu_t = 1. \tag{4.43d}$$

From (4.43c), we have $\|\nu_t^1 + \nu_t^2\|_1 = \|\nabla_{h_t}\Pi_t(w_t)\|_1 \leq \Psi_t$, where the inequality holds by (4.41). But $\nu_t^1, \nu_t^2 \geq 0$ in (4.29), and therefore, $\max\{\|\nu_t^1\|_1, \|\nu_t^2\|_1\} \leq \|\nu_t^1 + \nu_t^2\|_1$ by definition. Combining the two results, we have $\max\{\|\nu_t^1\|_1, \|\nu_t^2\|_1\} \leq \Psi_t$.

Next, we prove that $\max\{\|\delta_t^1\|_1, \|\delta_t^2\|_1\} \leq \Psi_t$. To show this, we first prove that $\langle\delta_t^1, \delta_t^2\rangle = 0$, where $\langle\cdot,\cdot\rangle$ denotes the inner product operator. Consider the following cases for y_t:

Case 1. If $y_t = a\lambda\mathbf{1}$, then $\delta_t^2 = 0$ by (4.31c), and therefore, $\langle\delta_t^1, \delta_t^2\rangle = 0$.

Case 2. If $y_t = 0$, then $\delta_t^1 = 0$ by (4.31b), and therefore, $\langle\delta_t^1, \delta_t^2\rangle = 0$.

Case 3. If $y_t \in (0, a\lambda\mathbf{1})$, then $\delta_t^1, \delta_t^2 = 0$ by (4.31b) and (4.31c), and therefore, $\langle\delta_t^1, \delta_t^2\rangle = 0$.

Using $\langle\delta_t^1, \delta_t^2\rangle = 0$ and the fact that $\delta_t^1, \delta_t^2 \geq 0$ in (4.24), it immediately follows that $\delta_{t,i}^1\delta_{t,i}^2 = 0$ for $i = 1, \ldots, n_t$. Next, consider the following cases for $\delta_{t,i}^1$ and $\delta_{t,i}^2$:

Case 1. If $\delta_{t,i}^1 > \delta_{t,i}^2 = 0$, then $\delta_{t,i}^1 - \delta_{t,i}^2 = \max\{\delta_{t,i}^1, \delta_{t,i}^2\}$.

Case 2. If $\delta_{t,i}^2 > \delta_{t,i}^1 = 0$, then $\delta_{t,i}^1 - \delta_{t,i}^2 = -\max\{\delta_{t,i}^1, \delta_{t,i}^2\}$.

Case 3. If $\delta_{t,i}^2 = \delta_{t,i}^1 = 0$, then $\delta_{t,i}^1 - \delta_{t,i}^2 = \max\{\delta_{t,i}^1, \delta_{t,i}^2\} = 0$.

Therefore, $\max\{\delta_{t,i}^1, \delta_{t,i}^2\} = |\delta_{t,i}^1 - \delta_{t,i}^2|$ for $i = 1, \ldots, n_t$, and it immediately follows that

$$\|\delta_t^1 - \delta_t^2\|_1 = \sum_{i=1}^{n_t} |\delta_{t,i}^1 - \delta_{t,i}^2| = \sum_{i=1}^{n_t} \max\{\delta_{t,i}^1, \delta_{t,i}^2\}. \tag{4.44}$$

It is noted that the inequalities $\max\{\delta_{t,i}^1, \delta_{t,i}^2\} \geq \delta_{t,i}^1$ and $\max\{\delta_{t,i}^1, \delta_{t,i}^2\} \geq \delta_{t,i}^2$ hold by definition of $\max\{\cdot\}$. It immediately follows from (4.44) that

$$\|\delta_t^1 - \delta_t^2\|_1 \geq \sum_{i=1}^{n_t} \delta_{t,i}^1 = \|\delta_t^1\|_1, \tag{4.45a}$$

$$\|\delta_t^1 - \delta_t^2\|_1 \geq \sum_{i=1}^{n_t} \delta_{t,i}^2 = \|\delta_t^2\|_1, \tag{4.45b}$$

where the equalities hold because $\delta_{t,i}^1, \delta_{t,i}^2 \geq 0$. Together the constraints in (4.45) imply that $\|\delta_t^1 - \delta_t^2\|_1 \geq \max\{\|\delta_t^1\|_1, \|\delta_t^2\|_1\}$. On the other hand,

$$\|\delta_t^1 - \delta_t^2\|_1 = \|\nabla_{y_t}\Pi_t(w_t)\|_1 \leq \Psi_t,$$

where the equality holds by (4.43b) and the inequality follows from (4.41). Therefore, $\max\{\|\delta_t^1\|_1, \|\delta_t^2\|_1\} \leq \Psi_t$. Next, using (4.43d), it is easy to show that $0 \leq \mu_t^j \leq 1$ for $j = 1, \ldots, C$, where $C \equiv |\mathcal{C}_{t+1}|$. Thus, we have

$$\|\mu_t\|_1 = \sum_{j=1}^{C} \mu_t^j \leq C \leq \Psi_t,$$

where the second inequality follows from our definition of Ψ_t. Finally, we show that $\|A_t'\pi_t\|_1$ is bounded. Consider equation (4.43a). Then,

$$\begin{aligned} \|A_t'\pi_t\|_1 &= \|\nu_t^2 - \nu_t^1 - \hat{\beta}_{t+1}\mu_t - \nabla_{x_t}\Pi_t(w_t)\|_1, \\ &\leq \|\nu_t^2\|_1 + \|\nu_t^1\|_1 + \|\hat{\beta}_{t+1}'\mu_t\|_1 + \|\nabla_{x_t}\Pi_t(w_t)\|_1, \\ &= 3\Psi_t + \|\hat{\beta}_{t+1}'\mu_t\|_1, \end{aligned} \tag{4.46}$$

where the inequality follows from the triangle inequality, and the equality holds because ν_t^1, ν_t^2 and $\nabla_{x_t}\Pi_t(w_t)$ are bounded by Ψ_t. Now, consider the term $\|\hat{\beta}_{t+1}'\mu_t\|_1$. Using the sub-multiplicative property of matrix norms and the fact that $\|\mu_t\|_1 \leq \Psi_t$, we have

$$\|\hat{\beta}_{t+1}'\mu_t\|_1 \leq \|\hat{\beta}_{t+1}'\|_1 \|\mu_t\|_1 \leq \|\hat{\beta}_{t+1}'\|_1 \Psi_t.$$

Let $\hat{\beta}_{t+1,j}'$ denote the jth n_t-dimensional column vector of the matrix $\hat{\beta}_{t+1}'$, where $j = 1, \ldots, C$. By part (b) of Proposition 6.3.1 in [90], we have

$$\|\hat{\beta}_{t+1}'\|_1 = \max_{j \in \{1, \ldots, C\}} \|\hat{\beta}_{t+1,j}'\|_1,$$

which is a finite value that depends on the elements in $\hat{\beta}_{t+1}'$. Define $\Gamma_t \equiv \|\hat{\beta}_{t+1}'\|_1$ and substitute it in (4.46) to obtain $\|A_t'\pi_t\|_1 \leq \Psi_t(3 + \Gamma_t)$, and the proof is complete. □

An identical result can be shown for NCNS-QP model (4.26) and the LPCC model (4.36) associated with the SCAD function. Finally, we provide the steps of our regularized SDDP algorithm, called regSDDP, as follows:

1) **Initialization**.

   Set $k = 1$ and $\widehat{\mathcal{V}}_t^0 = -\infty$ for all $t \in \hat{\mathcal{T}}$. Initialize the values of M, ϵ, δ.

2) **Forward Pass**.

2a) Sample M distinct SAA scenarios from $\mathscr{J}$ to form $\mathscr{J}^k$;

2b) For each $\tilde{\xi} \in \mathscr{J}^k$ and $t = 1, \dots, T-1$, solve the regularization problem (4.19) to obtain $\hat{x}_t^k(\tilde{\xi})$;

2c) If $t = T$, then solve the traditional forward-pass problem (4.5) to obtain $\hat{x}_T^k(\tilde{\xi})$;

2d) Compute the sample-average estimate of the total costs, $\hat{z}_u^k$, via (4.6), and the sample standard deviation of $\hat{z}_u^k$ via (4.7);

3) **Backward Pass**.

3a) For each $\tilde{\xi} \in \mathscr{J}^k$ and $t = T, \dots, 2$, solve problem (4.11) using $\hat{x}_{t-1}^k(\tilde{\xi})$, generate a cut $\ell_t^k(x_{t-1})$ via (4.12), and update the current approximation as $\widehat{\mathcal{V}}_t^k(x_{t-1}) \leftarrow \max\{\widehat{\mathcal{V}}_t^{k-1}(x_{t-1}), \ell_t^k(x_{t-1})\}$;

3b) For $t = 1$, solve the root-node problem (4.14) and obtain the lower bound $\hat{z}_\ell^k$; set $k \leftarrow k+1$ and go to Step 2.

4) **Convergence Test**.

Terminate the algorithm if $\left|\hat{z}_u^k + \hat{t}_{\delta/2,\,M}\,\hat{\sigma}_k/\sqrt{M} - \hat{z}_\ell^k\right| \leq \epsilon$; else, go to Step 2.

4.5 CONVERGENCE ANALYSIS

Here, we show that the lower bounds generated by the regSDDP algorithm converge almost surely to the SAA lower bound. We first introduce some notation pertaining to the scenario-tree formulation of (4.3). Let $\mathscr{T}$ be a scenario tree with T stages. The set of all nodes in $\mathscr{T}$ is denoted by $\mathcal{N}$, while the set of all nodes at stage $t \in \mathcal{T}$ is denoted by $\mathcal{N}_t$. The root node at stage 1 is denoted by r. For $t \in \hat{\mathcal{T}}$, each node $n \in \mathcal{N}_t$ represents a realization of ξ_t. Let ζ_n and Θ_n denote the parent and the set of children, respectively, of node $n \in \mathcal{N}$. The leaf nodes at stage T have no children, and the root node has no parent. The transition probability from node n to node $m \in \Theta_n$ is denoted by p_n^m, and for each $n \in \mathcal{N} \setminus \mathcal{N}_T$, the collection $\{p_n^m : m \in \Theta_n\}$ satisfies

$$\sum_{m \in \Theta_n} p_n^m = 1.$$

Next, let $\xi^k(\omega) \in \mathcal{J}^k$ denote a scenario sampled in the forward pass at iteration k. Define the following event for each $\tilde{\xi} \in \mathcal{J}$ and $k \in \mathbb{N}$:

$$\Delta(k, \tilde{\xi}) = \left\{ \omega \in \Omega : \xi^k(\omega) = \tilde{\xi} \right\}. \tag{4.47}$$

Sampling a scenario corresponds to selecting a unique sequence of T nodes, one from each stage t. For each $n \in \mathcal{N}_t$ and $k \in \mathbb{N}$, define the (random) indicator variable

$$b_n^k = \begin{cases} 1, & \text{if node } n \text{ is selected in iteration } k, \\ 0, & \text{otherwise,} \end{cases} \tag{4.48}$$

and let $b_k \equiv (b_n^k : n \in \mathcal{N})$. The process $\mathcal{S} = \{b_k : k \in \mathbb{N}\}$ describing the evolution of the nodes selected in each iteration is adapted to the natural filtration, $\mathscr{F} = \{\mathcal{F}_k : k \in \mathbb{N}\}$, where $\mathcal{F}_k = \sigma(b_1, \dots, b_k)$ is the σ-field generated in the first k iterations. For $n \in \mathcal{N}_t$, let $\widehat{\mathcal{V}}_{t+1}(x_n)$ and $\widehat{\mathcal{V}}_{t+1}^k(x_n)$ be the scenario-tree analogs of $\widehat{\mathcal{V}}_{t+1}(x_t)$ and $\widehat{\mathcal{V}}_t^k(x_t)$, respectively. Finally, let $\hat{x}_n^k$ be a trial solution at node n in iteration k, i.e.,

$$\hat{x}_n^k \in \operatorname{argmin}\{c_t(x_n, \tilde{\xi}_n) + \widehat{\mathcal{V}}_{t+1}^{k-1}(x_n) + F_{\lambda_k}(x_n, \check{x}_n^k) : x_n \in \mathcal{X}_t(\hat{x}_{\zeta_n}^k, \tilde{\xi}_n)\},$$

where $\mathcal{X}_t(\hat{x}_{\zeta_n}^k, \tilde{\xi}_n) = \{x_n \in \mathbb{R}^{n_t} : A_n x_n + B_n \hat{x}_{\zeta_n}^k = b_n\}$. Before establishing our convergence results, we first state a few important assumptions regarding the sampling process.

Assumption 4.1. *The forward-pass scenarios at each step are sampled independently.*

Assumption 4.2. *For each $\tilde{\xi} \in \mathcal{J}$ and $k \in \mathbb{N}$, $\mathbb{P}(\Delta(k, \tilde{\xi})) > 0$.*

Assumption 4.3. *For each $n \in \mathcal{N}$ and $k \in \mathbb{N}$, b_n^k is independent of $\mathcal{F}_{k-1}$.*

Under Assumptions 4.1 and 4.2, it immediately follows from the Borel zero-one law that

$$\mathbb{P}\left(\limsup_{k \to \infty} \Delta(k, \tilde{\xi}) \right) = 1, \quad \forall \tilde{\xi} \in \mathcal{J}. \tag{4.49}$$

Condition (4.49) is called the forward-pass sampling (FPS) property, which ensures that each node in the tree is selected infinitely often almost surely. Assumption 4.3 implies that prior information has no bearing on current node selections. If Assumption 4.3 is not imposed, then it is possible that the outer approximations are updated only at a select set of nodes.

Note that under the Assumptions 4.1–4.3, if the algorithm converges for the case $M = 1$, then it also converges for $M > 1$ because the number of possible cuts in each stage is finite (see Lemma 1 in [118]); therefore, any sequence of scenarios will produce the same collection of cuts in the limit. Hence, in what follows, we assume that $M = 1$. Next, we discuss the assumption on the sequence of penalty parameters $\{\lambda_k : k \in \mathbb{N}\}$.

Assumption 4.4. *As $k \to \infty$, $\lambda_k \to 0$.*

Assumption 4.4 ensures that the effect of regularization diminishes after a sufficiently large number of iterations and does not require that $\{\lambda_k : k \in \mathbb{N}\}$ be a monotone decreasing sequence. A sufficient condition for this assumption to hold is

$$\lim_{k\to\infty} \frac{\lambda_{k+1}}{\lambda_k} < 1$$

(see Theorems 3.23 and 3.34 in [127]), which is satisfied by any sequence with a decreasing tail. Hence, one can choose a locally increasing (but tail decreasing) sequence that promotes exploration in the initial iterations.

Assumption 4.5. *The feasibility sets in (4.3) are nonempty, convex, and compact.*

Assumption 4.6. *For each $n \in \mathcal{N}_{t-1}$, $\widehat{\mathcal{V}}_t(x_n)$ is convex, finite and Lipschitz continuous in x_n.*

Recall that the feasibility sets were assumed to be convex polytopes in Section 4.2, and the relatively complete recourse property of (4.1) ensures that these sets are nonempty. Assumption 4.6 immediately follows from Lemma 3.1 in [56], as the one-step costs are linear and finite. Our final assumption states that if a node is not selected, then the outer approximation at that node remains unchanged (often referred to as the null updates).

Assumption 4.7. *For each $n \in \mathcal{N}_{t-1}$ and $k \in \mathbb{N}$, if $b_n^k = 0$, then $\widehat{\mathcal{V}}_t^k(x_n) = \widehat{\mathcal{V}}_t^{k-1}(x_n)$.*

Assumption 4.7 does not imply that $\hat{x}_n^k = \hat{x}_n^{k-1}$ when $b_n^k = 0$. Note that a trial solution at a node, whether it is selected or not, can be determined only when the parent of that node is selected. Therefore, the sequence of trial solutions, $\{\hat{x}_n^k\}$, is defined only for $k \in \mathcal{K}_n \equiv \left\{k \in \mathbb{N} : b_{\zeta_n}^k = 1\right\}$. The set of iterations where node n is also selected is denoted by $\Upsilon_n = \left\{k \in \mathbb{N} : b_n^k = 1\right\}$. It is obvious that $\Upsilon_n \subset \mathcal{K}_n = \Upsilon_{\zeta_n}$. As the root node is always selected in the forward pass, we set $\mathcal{K}_r = \Upsilon_r = \mathbb{N}$. From the FPS property, it immediately follows that $\mathcal{K}_n$ and Υ_n are countably infinite sets.

Theorem 4.3 asserts that the sequence of outer approximations at each node converges almost surely (a.s.) to the SAA value function in the neighborhood of the trial points generated by the regSDDP algorithm.

Theorem 4.3. *For each $t \in \hat{\mathcal{T}}$, $n \in \mathcal{N}_{t-1}$, and $k \in \mathcal{K}_n$,*

$$\lim_{k\to\infty} \widehat{\mathcal{V}}_t(\hat{x}_n^k) - \widehat{\mathcal{V}}_t^k(\hat{x}_n^k) = 0 \quad (a.s.). \tag{4.50}$$

Proof. For each $t \in \hat{\mathcal{T}}$, let $\mathscr{C}(t)$ denote the condition in (4.50). We use backward induction on t to show that $\mathscr{C}(t)$ holds for all $t \in \hat{\mathcal{T}}$. First, consider the case $t = T$. Choose any node $n \in \mathcal{N}_{T-1}$. As $\widehat{\mathcal{V}}_T^k$ is a lower bound of $\widehat{\mathcal{V}}_T$, we have $\widehat{\mathcal{V}}_T(\hat{x}_n^k) \geq \widehat{\mathcal{V}}_T^k(\hat{x}_n^k)$ for all $k \in \mathcal{K}_n$. However, $\widehat{\mathcal{V}}_T^k(\hat{x}_n^k) \geq \widehat{\mathcal{V}}_T(\hat{x}_n^k)$ as

$$\widehat{\mathcal{V}}_T^k(\hat{x}_n^k) \geq \sum_{m\in\Theta_n} p_m \bar{V}_T^k(\hat{x}_n^k, \tilde{\xi}_m)) = \sum_{m\in\Theta_n} p_m \widehat{V}_T(\hat{x}_n^k, \tilde{\xi}_m)) = \widehat{\mathcal{V}}_T(\hat{x}_n^k),$$

where the first inequality follows from (4.9) and (4.10), the first equality holds because $\widehat{\mathcal{V}}_{T+1}(x_m) = 0$ for all $m \in \Theta_n$, and the second equality follows from definition. Therefore, $\widehat{\mathcal{V}}_T(\hat{x}_n^k) = \widehat{\mathcal{V}}_T^k(\hat{x}_n^k)$. As the choice of n was arbitrary, we conclude that $\mathscr{C}(T)$ is true.

For the inductive step, suppose $\mathscr{C}(t+1)$ is true, where $(t+1) \in \check{\mathcal{T}}$ (recall that $\check{\mathcal{T}} \equiv \mathcal{T}\backslash\{T\}$). Consider a node $n \in \mathcal{N}_{t-1}$. Define $\Upsilon_n \equiv \{k \in \mathbb{N} : b_n^k = 1\}$, and note that $|\Upsilon_n| = \infty$ by the FPS property (4.49). Next, we consider two mutually exclusive cases for $k \in \mathcal{K}_n$ and show that $\mathscr{C}(t)$ holds for each case separately.

Case 1. Suppose $k \in \mathcal{K}_n \cap \Upsilon_n$, i.e., both n and ζ_n are selected at step k. Note that $\Upsilon_n \subseteq \mathcal{K}_n$ as n can be selected only if ζ_n is selected (the opposite statement is not true). Therefore, $k \in \mathcal{K}_n \cap \Upsilon_n \implies k \in \Upsilon_n$. Let $k' \in \Upsilon_n$, where $k' < k$, be the last iteration before k when n was selected. For ease of exposition, denote the objective functions of the nodal problems (4.3b), (4.19), and (4.11) at $m \in \Theta_n$ by

$$Q_t(x_m, \tilde{\xi}_m) = c_t(x_m, \tilde{\xi}_m) + \widehat{\mathcal{V}}_{t+1}(x_m), \tag{4.51}$$

$$J_t^k(x_m, \tilde{\xi}_m) = c_t(x_m, \tilde{\xi}_m) + \widehat{\mathcal{V}}_{t+1}^{k'}(x_m) + F_{\lambda_k}(x_m, \check{x}_m^k), \tag{4.52}$$

$$Q_t^k(x_m, \tilde{\xi}_m) = c_t(x_m, \tilde{\xi}_m) + \widehat{\mathcal{V}}_{t+1}^k(x_m), \tag{4.53}$$

respectively, where $x_m \in \mathcal{X}_t(\hat{x}_n^k, \tilde{\xi}_m)$, and $\check{x}_m^k$ is a given stability center. It is noted that, under Assumption 4.7, $\widehat{\mathcal{V}}_{t+1}^{k'}(x_m)$ is the current approximation of $\widehat{\mathcal{V}}_{t+1}(x_m)$ at the start of iteration k. For notational brevity, denote the transition probability p_n^m simply as p_m. Then, we use (4.12) and (4.13) to obtain

$$\widehat{\mathcal{V}}_t^k(\hat{x}_n^k) \geq \sum_{m \in \Theta_n} p_m \bar{V}_t^k(\hat{x}_n^k, \tilde{\xi}_m). \tag{4.54}$$

Consider a set of feasible points $\{\bar{x}_m\}_{m \in \Theta_n}$ such that

$$Q_t^{k'}(\bar{x}_m, \tilde{\xi}_m) = \bar{V}_t^{k'}(\hat{x}_n^k, \tilde{\xi}_m) \equiv \min \left\{ Q_t^{k'}(x_m, \tilde{\xi}_m) : x_m \in \mathcal{X}_t(\hat{x}_n^k, \tilde{\xi}_m) \right\}.$$

Note that $\bar{x}_m$ exists by the relatively complete recourse property of (4.1). It is noted that

$$\hat{x}_m^k \in \operatorname{argmin} \left\{ J_t^k(x_m, \tilde{\xi}_m) : x_m \in \mathcal{X}_t(\hat{x}_n^k, \tilde{\xi}_m) \right\}. \tag{4.55}$$

Next, consider the following sequence of constraints:

$$\begin{aligned}
\widehat{\mathcal{V}}_t(\hat{x}_n^k) - \widehat{\mathcal{V}}_t^k(\hat{x}_n^k) &\leq \sum_{m \in \Theta_n} p_m \left(\widehat{V}_t(\hat{x}_n^k, \tilde{\xi}_m) - \bar{V}_t^k(\hat{x}_n^k, \tilde{\xi}_m) \right), \\
&\leq \sum_{m \in \Theta_n} p_m \left(\widehat{V}_t(\hat{x}_n^k, \tilde{\xi}_m) - \bar{V}_t^{k'}(\hat{x}_n^k, \tilde{\xi}_m) \right), \\
&= \sum_{m \in \Theta_n} p_m \left(\widehat{V}_t(\hat{x}_n^k, \tilde{\xi}_m) - Q_t^{k'}(\hat{x}_m^k, \tilde{\xi}_m) \right) \\
&\quad + \sum_{m \in \Theta_n} p_m \left(Q_t^{k'}(\hat{x}_m^k, \tilde{\xi}_m) - Q_t^{k'}(\bar{x}_m, \tilde{\xi}_m) \right).
\end{aligned} \tag{4.56}$$

The first inequality is true by (4.54). The second inequality follows from the fact that $\bar{V}_t^{k'}(\hat{x}_n^k, \tilde{\xi}_m) \leq \bar{V}_t^k(\hat{x}_n^k, \tilde{\xi}_m)$, as problem (4.11) at iteration k' is a relaxation of the corresponding problem at iteration k. The first equality holds by definition of $Q_t^{k'}(\bar{x}_m, \tilde{\xi}_m)$, and the second equality is obtained by adding and subtracting the term $Q_t^{k'}(\hat{x}_m^k, \tilde{\xi}_m)$. Next, consider the first summation term in (4.56). Note that

$$\widehat{V}_t(\hat{x}_n^k, \tilde{\xi}_m) - Q_t^{k'}(\hat{x}_m^k, \tilde{\xi}_m) = \widehat{V}_t(\hat{x}_n^k, \tilde{\xi}_m) - Q_t(\hat{x}_m^k, \tilde{\xi}_m) + \widehat{\mathcal{V}}_{t+1}(\hat{x}_m^k) - \widehat{\mathcal{V}}_{t+1}^{k'}(\hat{x}_m^k), \tag{4.57}$$

where the equality follows from (4.51) and (4.53). However,

$$\widehat{V}_t(\hat{x}_n^k, \tilde{\xi}_m) \equiv \min \left\{ Q_t(x_m, \tilde{\xi}_m) : x_m \in \mathcal{X}_t(\hat{x}_n^k, \tilde{\xi}_m) \right\} \leq Q_t(\hat{x}_m^k, \tilde{\xi}_m), \tag{4.58}$$

where the inequality holds because $\hat{x}^k_m$ is a feasible, but not necessarily an optimal, solution of the problem in (4.58). Combining (4.57) and (4.58), we obtain

$$\widehat{V}_t(\hat{x}^k_n, \tilde{\xi}_m) - Q^{k'}_t(\hat{x}^k_m, \tilde{\xi}_m) \le \widehat{\mathcal{V}}_{t+1}(\hat{x}^k_m) - \widehat{\mathcal{V}}^{k'}_{t+1}(\hat{x}^k_m), \quad \forall m \in \Theta_n. \tag{4.59}$$

Next, consider the second summation term in (4.56). Add and subtract the terms $J^k_t(\hat{x}^k_m, \tilde{\xi}_m)$ and $J^k_t(\bar{x}_m, \tilde{\xi}_m)$ to each term in the summation to obtain

$$\begin{aligned}
Q^{k'}_t(\hat{x}^k_m, \tilde{\xi}_m) - Q^{k'}_t(\bar{x}_m, \tilde{\xi}_m) &= Q^{k'}_t(\hat{x}^k_m, \tilde{\xi}_m) - J^k_t(\hat{x}^k_m, \tilde{\xi}_m) + J^k_t(\hat{x}^k_m, \tilde{\xi}_m) - J^k_t(\bar{x}_m, \tilde{\xi}_m) \\
&\quad + J^k_t(\bar{x}_m, \tilde{\xi}_m) - Q^{k'}_t(\bar{x}_m, \tilde{\xi}_m), \\
&\le Q^{k'}_t(\hat{x}^k_m, \tilde{\xi}_m) - J^k_t(\hat{x}^k_m, \tilde{\xi}_m) + J^k_t(\bar{x}_m, \tilde{\xi}_m) - Q^{k'}_t(\bar{x}_m, \tilde{\xi}_m), \quad (4.60)
\end{aligned}$$

where the inequality holds because $J^k_t(\hat{x}^k_m, \tilde{\xi}_m) \le J^k_t(\bar{x}_m, \tilde{\xi}_m)$ as $\bar{x}_m$ is a feasible solution of the problem in (4.55). Next, observe that for any $x_m \in \mathcal{X}_t(\hat{x}^k_n, \tilde{\xi}_m)$,

$$J^k_t(x_m, \tilde{\xi}_m) - Q^{k'}_t(x_m, \tilde{\xi}_m) = F_{\lambda_k}(x_m, \check{x}^k_m) = \sum_{i=1}^{n_t} f_{\lambda_k}(|\, x_{m,i} - \check{x}^k_{m,i}|), \tag{4.61}$$

where the first equality follows from (4.52) and (4.53), and the second equality is true from definition of F_{λ_k} in (4.20). Combining (4.60) and (4.61), we obtain

$$\begin{aligned}
Q^{k'}_t(\hat{x}^k_m, \tilde{\xi}_m) - Q^{k'}_t(\bar{x}_m, \tilde{\xi}_m) &\le \sum_{i=1}^{n_t} f_{\lambda_k}(|\, \bar{x}_{m,i} - \check{x}^k_{m,i}|) - \sum_{i=1}^{n_t} f_{\lambda_k}(|\, \hat{x}_{m,i} - \check{x}^k_{m,i}|), \\
&\le \sum_{i=1}^{n_t} f_{\lambda_k}(|\, \bar{x}_{m,i} - \check{x}^k_{m,i}|), \\
&\le n_t f_{\lambda_k}(a\lambda_k), \quad (4.62)
\end{aligned}$$

where the second inequality is true as $f_{\lambda_k}(|\, \hat{x}_{m,i} - \check{x}^k_{m,i}|) \ge 0$, and the third inequality holds by Property (P3) in Section 4.3.1. Combining (4.59) and (4.62), we express (4.56) as

$$\begin{aligned}
\widehat{\mathcal{V}}_t(\hat{x}^k_n) - \widehat{\mathcal{V}}^k_t(\hat{x}^k_n) &\le \sum_{m \in \Theta_n} p_m \left(\widehat{\mathcal{V}}_{t+1}(\hat{x}^k_m) - \widehat{\mathcal{V}}^{k'}_{t+1}(\hat{x}^k_m) \right) + \sum_{m \in \Theta_n} p_m n_t f_{\lambda_k}(a\lambda_k), \\
&= \sum_{m \in \Theta_n} p_m \left(\widehat{\mathcal{V}}_{t+1}(\hat{x}^k_m) - \widehat{\mathcal{V}}^{k'}_{t+1}(\hat{x}^k_m) \right) + n_t f_{\lambda_k}(a\lambda_k), \quad (4.63)
\end{aligned}$$

where the equality holds because $\sum_{m\in\Theta_n} p_m = 1$. However, $\widehat{\mathcal{V}}_t(\hat{x}_n^k) \geq \widehat{\mathcal{V}}_t^k(\hat{x}_n^k)$ as $\widehat{\mathcal{V}}_t^k$ is a lower bound of $\widehat{\mathcal{V}}_t$ for all $k \in \Upsilon_n$. Therefore,

$$0 \leq \widehat{\mathcal{V}}_t(\hat{x}_n^k) - \widehat{\mathcal{V}}_t^k(\hat{x}_n^k) \leq \sum_{m\in\Theta_n} p_m \left(\widehat{\mathcal{V}}_{t+1}(\hat{x}_m^k) - \widehat{\mathcal{V}}_{t+1}^{k'}(\hat{x}_m^k)\right) + n_t f_{\lambda_k}(a\lambda_k). \tag{4.64}$$

As $\mathscr{C}(t+1)$ was assumed to be true, it immediately follows that for $k \in \Upsilon_n$,

$$\lim_{k\to\infty} \left\{\widehat{\mathcal{V}}_{t+1}(\hat{x}_m^k) - \widehat{\mathcal{V}}_{t+1}^k(\hat{x}_m^k)\right\} \overset{a.s.}{=} 0, \quad \forall m \in \Theta_n. \tag{4.65}$$

However, under Assumptions 4.5 and 4.6, we know from Lemma A.1. in [56] that (4.65) holds if and only if the following condition holds:

$$\lim_{k\to\infty} \left\{\widehat{\mathcal{V}}_{t+1}(\hat{x}_m^k) - \widehat{\mathcal{V}}_{t+1}^{k'}(\hat{x}_m^k)\right\} \overset{a.s.}{=} 0, \quad \forall m \in \Theta_n. \tag{4.66}$$

Note that

$$\lim_{k\to\infty} f_{\lambda_k}(a\lambda_k) = 0, \tag{4.67}$$

by Property (P1) in Section 4.3.1 and Assumption 4.4. Taking the limit $k \to \infty$, where $k \in \Upsilon_n$, on both sides of (4.64) and using the results in (4.66) and (4.67), it immediately follows from the Sandwich Theorem that

$$\lim_{k\to\infty} \left\{\widehat{\mathcal{V}}_t(\hat{x}_n^k) - \widehat{\mathcal{V}}_t^k(\hat{x}_n^k)\right\} \overset{a.s.}{=} 0. \tag{4.68}$$

As the choice of n was arbitrary, we conclude that $\mathscr{C}(t)$ holds for all $n \in \mathcal{N}_{t-1}$.

Case 2. Suppose $k \in \mathcal{K}_n \cap \Upsilon_n^c$, i.e., ζ_n is selected, but n is not selected, at step k. It is noted that $\mathcal{K}_n \cap \Upsilon_n^c = \mathcal{K}_n \setminus \Upsilon_n$ as $\Upsilon_n \subseteq \mathcal{K}_n$. Suppose $\mathscr{C}(t)$ is false. Then, there exists $\varepsilon > 0$ such that there is an infinite number of iterations $k \in \mathcal{K}_n$ where $\widehat{\mathcal{V}}_t(\hat{x}_n^k) - \widehat{\mathcal{V}}_t^k(\hat{x}_n^k) \geq \varepsilon$. Let $k' \in \mathcal{K}_n$, where k' denotes the last iteration before k when ζ_n was selected. Note that $\widehat{\mathcal{V}}_t^k(\hat{x}_n^k) \geq \widehat{\mathcal{V}}_t^{k'}(\hat{x}_n^k)$ by the monotonicity of the outer approximations. Then, the set

$$\mathcal{K}_n^\varepsilon = \left\{k \in \mathcal{K}_n : \widehat{\mathcal{V}}_t(\hat{x}_n^k) - \widehat{\mathcal{V}}_t^{k'}(\hat{x}_n^k) \geq \varepsilon\right\}$$

has an infinite number of elements. Define $\mathcal{S}_n \equiv \{b_n^k : k \in \mathcal{K}_n\}$ (recall that $b_n^k = 1$ if $k \in \Upsilon_n$, and $b_n^k = 0$ otherwise). From Assumption 4.3, we know that $b_n^k \perp\!\!\!\perp \mathcal{F}_{k'}$ for any $k, k' \in \mathcal{K}_n$. It is noted that $\hat{x}_n^k$ and $\widehat{\mathcal{V}}_t^{k'}(\hat{x}_n^k)$ are measurable w.r.t. $\mathcal{F}_{k'}$ under Assumption

4.7. Next, consider the process $\mathcal{W}_n = \{w_n^k : k \in \mathcal{K}_n\}$, where $w_n^k = 1$ if $k \in \mathcal{K}_n^\varepsilon$, and $w_n^k = 0$ otherwise. It immediately follows from the definition of $\mathcal{K}_n^\varepsilon$ that w_n^k is measurable w.r.t $\mathcal{F}_{k'}$, and is therefore, independent of b_n^k, i.e., $w_n^k \perp\!\!\!\perp b_n^k$. Consider the derived process $\mathcal{S}_n^\varepsilon = \{b_n^k : w_n^k = 1, k \in \mathcal{K}_n\} = \{b_n^k : k \in \mathcal{K}_n^\varepsilon\}$. From Lemma A.3. in [56], we know that the elements in $\mathcal{S}_n^\varepsilon$ are i.i.d. and have the same distribution as b_n^1. Observe that $\mathbb{E}(b_n^1) = \mathbb{P}(b_n^1 = 1) > 0$ by Assumption 4.2. Therefore, from the Strong Law of Large Numbers, it immediately follows that

$$\lim_{L\to\infty} \frac{1}{L}\sum_{i=1}^{L} b_n^{k_i} = \mathbb{E}(b_n^1) > 0, \tag{4.69}$$

where $b_n^{k_i}$ is the ith element in $\mathcal{S}_n^\varepsilon$. However, note that $\mathcal{K}_n^\varepsilon \cap \Upsilon_n$ is finite as

$$\lim_{k\to\infty} \widehat{\mathcal{V}}_t(\hat{x}_n^k) - \widehat{\mathcal{V}}_t^{k'}(\hat{x}_n^k) = 0$$

holds (for $k \in \Upsilon_n$) by Lemma A.1. in [56] and (4.68). Therefore, $\mathcal{S}_n^\varepsilon$ must have a finite number of elements that are equal to 1, the rest all being equal to 0. Thus,

$$\lim_{L\to\infty} \frac{1}{L}\sum_{i=1}^{L} b_n^{k_i} = 0, \tag{4.70}$$

which contradicts the result in (4.69). Therefore, $\mathscr{C}(t)$ holds, and the proof is complete. □

Theorem 4.3 does not establish pointwise convergence of the outer approximations for all feasible decisions at a node, but rather implies convergence only for the trial points generated by the algorithm. Our next result follows directly from Theorem 4.3 and shows that the sequence of lower bounds converges to the SAA optimal value $\hat{z}_\ell$ almost surely.

Theorem 4.4. *The limit of the sequence* $\{\hat{z}_\ell^k\}_{k\in\mathbb{N}}$ *is* $\hat{z}_\ell$*, i.e.,* $\lim_{k\to\infty} \hat{z}_\ell^k = \hat{z}_\ell$ *(a.s.).*

Proof. It is noted that the root node r is always selected in the forward pass, i.e, $\Upsilon_r = \mathbb{N}$. Consequently, the lower bound $\hat{z}_\ell^k$ is updated at each $k \in \mathbb{N}$. For ease of exposition, let

$$Q_1(x_r) = c_1(x_r) + \widehat{\mathcal{V}}_2(x_r), \tag{4.71}$$

$$J_1^k(x_r) = c_1(x_r) + \widehat{\mathcal{V}}_2^{k-1}(x_r) + F_{\lambda_k}(x_r, \check{x}_r^k), \tag{4.72}$$

$$Q_1^k(x_r) = c_1(x_r) + \widehat{\mathcal{V}}_2^k(x_r), \tag{4.73}$$

where $x_r \in \mathcal{X}_1$ (recall that $\mathcal{X}_1$ is the feasibility set at stage 1). Using the relationships in (4.72) and (4.73) for iteration k, we obtain the following equality:

$$J_1^k(x_r) + Q_1^{k-1}(x_r) = F_{\lambda_k}(x_r, \check{x}_r^k), \quad x_r \in \mathcal{X}_1. \tag{4.74}$$

Define $\hat{z}_\ell^{k-1} \equiv Q_1^{k-1}(\bar{x}_r)$, where $\bar{x}_r \in \operatorname{argmin}\left\{Q_1^{k-1}(x_r) : x_r \in \mathcal{X}_1\right\}$. As z_ℓ^k is a valid lower bound of $\hat{z}_\ell$, we have $\hat{z}_\ell \geq \hat{z}_\ell^k$. Moreover, $\hat{z}_\ell^k \geq \hat{z}_\ell^{k-1}$ by the monotone property of outer approximations. Then, the following sequence of constraints hold:

$$0 \leq \hat{z}_\ell - \hat{z}_\ell^k \leq \hat{z}_\ell - \hat{z}_\ell^{k-1} = \hat{z}_\ell - Q_1^{k-1}(\bar{x}_r) = \left(\hat{z}_\ell - Q_1^{k-1}(\hat{x}_r^k)\right) + \left(Q_1^{k-1}(\hat{x}_r^k) - Q_1^{k-1}(\bar{x}_r)\right), \tag{4.75}$$

where the second equality follows from the addition and subtraction of the term $Q_1^{k-1}(\hat{x}_r^k)$. However, the term within the first bracket in (4.75) can be constrained as follows:

$$\begin{aligned} \hat{z}_\ell - Q_1^{k-1}(\hat{x}_r^k) &= \hat{z}_\ell - c_1(\hat{x}_r^k) + \widehat{\mathcal{V}}_2^{k-1}(\hat{x}_r^k), \\ &= \hat{z}_\ell - Q_1(\hat{x}_r^k) + \widehat{\mathcal{V}}_2(\hat{x}_r^k) - \widehat{\mathcal{V}}_2^{k-1}(\hat{x}_r^k), \\ &\leq \widehat{\mathcal{V}}_2(\hat{x}_r^k) - \widehat{\mathcal{V}}_2^{k-1}(\hat{x}_r^k), \end{aligned} \tag{4.76}$$

where the first and second equalities follow from (4.73) and (4.71), respectively, and the inequality holds because $\hat{z}_\ell \leq Q_1(\hat{x}_r^k)$ as $\hat{x}_r^k$ is a feasible, but not necessarily an optimal, solution of the SAA root-node problem (4.3a). Next, consider the following sequence of relationships concerning the term within the second bracket in (4.75):

$$\begin{aligned} Q_1^{k-1}(\hat{x}_r^k) - Q_1^{k-1}(\bar{x}_r) &= Q_1^{k-1}(\hat{x}_r^k) - J_1^k(\hat{x}_r^k) + J_1^k(\hat{x}_r^k) - J_1^k(\bar{x}_r) + J_1^k(\bar{x}_r) - Q_1^{k-1}(\bar{x}_r), \\ &\leq Q_1^{k-1}(\hat{x}_r^k) - J_1^k(\hat{x}_r^k) + J_1^k(\bar{x}_r) - Q_1^{k-1}(\bar{x}_r), \\ &= F_{\lambda_k}(\bar{x}_r, \check{x}_r^k) - F_{\lambda_k}(\hat{x}_r, \check{x}_r^k), \\ &\leq F_{\lambda_k}(\bar{x}_r, \check{x}_r^k), \\ &\leq n_1 f_{\lambda_k}(a\lambda_k). \end{aligned} \tag{4.77}$$

The first equality follows from the addition and subtraction of the terms $J_1^k(\hat{x}_r^k)$ and $J_1^k(\bar{x}_r)$. The first inequality is true because $J_1^k(\hat{x}_r^k) \leq J_1^k(\bar{x}_r)$ as $\bar{x}_r$ is a feasible solution of the problem $\min\{J_1^k(x_r) : x_r \in \mathcal{X}_1\}$ whose optimal solution is $\hat{x}_r^k$ by definition. The second equality is true by (4.74). The second inequality holds because $F_{\lambda_k}(\hat{x}_r, \check{x}_r^k) \geq 0$. The final inequality follows from Property (P3) in Section 4.3.1. Combining (4.76) and (4.77), we can express (4.75) as follows:

$$0 \leq \hat{z}_\ell - \hat{z}_\ell^k \leq \widehat{\mathcal{V}}_2(\hat{x}_r^k) - \widehat{\mathcal{V}}_2^{k-1}(\hat{x}_r^k) + n_1 f_{\lambda_k}(a\lambda_k). \tag{4.78}$$

From Proposition 4.3 and Lemma A.1. in [56], we know that

$$\lim_{k\to\infty} \left\{ \widehat{\mathcal{V}}_2(\hat{x}_r^k) - \widehat{\mathcal{V}}_2^{k-1}(\hat{x}_r^k) \right\} \overset{a.s.}{=} 0. \tag{4.79}$$

Moreover, under Assumption 4.4, it follows from Property (P1) in Section 4.3.1 that

$$\lim_{k\to\infty} f_{\lambda_k}(a\lambda_k) = 0. \tag{4.80}$$

Taking the limit $k \to \infty$ on both sides of (4.78) and using (4.79) and (4.80), we obtain the desired result by the Sandwich Theorem, and the proof is complete. □

Taken together, Theorems 4.3 and 4.4 guarantee almost sure convergence of the optimal values of the nodal problems at each stage. However, the quality of the solutions generated at each node cannot be inferred from these results. However, Theorem 4.5 establishes the asymptotic optimality of the trial solutions produced by regSDDP. Specifically, any sequence of trial solutions at a node converges to an optimal decision at that node almost surely. In what follows, let x_n^* be the optimal decision rule at node $n \in \mathcal{N}$.

Lemma 4.2. *For $t \in \hat{\mathcal{T}}$, $n \in \mathcal{N}_{t-1}$, and $k', k \in \mathcal{K}_n$ such that $k' < k$,*

$$\lim_{k\to\infty} \widehat{V}_t(\hat{x}_{\zeta_n}^k, \tilde{\xi}_n) - Q_t^{k'}(\hat{x}_n^k, \tilde{\xi}_n) = 0, \quad (a.s.). \tag{4.81}$$

Proof. Define $u \equiv \zeta_n$. Using the same arguments that were used to derive (4.59), we obtain the following inequality:

$$\widehat{V}_t(\hat{x}_u^k, \tilde{\xi}_n) - Q_t^{k'}(\hat{x}_n^k, \tilde{\xi}_n) \leq \widehat{\mathcal{V}}_{t+1}(\hat{x}_n^k) - \widehat{\mathcal{V}}_{t+1}^{k'}(\hat{x}_n^k). \tag{4.82}$$

As $\widehat{\mathcal{V}}_{t+1}^{k}(x_n) \leq \mathcal{V}_{t+1}(x_n)$ for all $k \in \mathcal{K}_n$, it is straightforward to show that $\widehat{V}_t(\hat{x}_u^k, \tilde{\xi}_n) \geq \bar{V}_t^k(\hat{x}_u^k, \tilde{\xi}_n)$. Consider a feasible solution $\bar{x}_n \in \mathcal{X}_t(\hat{x}_u^k, \tilde{\xi}_n)$ such that $\bar{V}_t^{k'}(\hat{x}_u^k, \tilde{\xi}_n) = Q_t^{k'}(\bar{x}_n, \tilde{\xi}_n)$. Then, we can obtain the following sequence of constraints:

$$\begin{aligned}
0 &\leq \widehat{V}_t(\hat{x}_u^k, \tilde{\xi}_n) - \bar{V}_t^k(\hat{x}_u^k, \tilde{\xi}_n), \\
&\leq \widehat{V}_t(\hat{x}_u^k, \tilde{\xi}_n) - \bar{V}_t^{k'}(\hat{x}_u^k, \tilde{\xi}_n), \\
&= \widehat{V}_t(\hat{x}_u^k, \tilde{\xi}_n) - Q_t^{k'}(\bar{x}_n, \tilde{\xi}_n), \\
&= \widehat{V}_t(\hat{x}_u^k, \tilde{\xi}_n) - Q_t^{k'}(\hat{x}_n^k, \tilde{\xi}_n) + Q_t^{k'}(\hat{x}_n^k, \tilde{\xi}_n) - Q_t^{k'}(\bar{x}_n, \tilde{\xi}_n), \\
&\leq \widehat{V}_t(\hat{x}_u^k, \tilde{\xi}_n) - Q_t^{k'}(\hat{x}_n^k, \tilde{\xi}_n) + n_t f_{\lambda_k}(a\lambda_k), \qquad (4.83)
\end{aligned}$$

where the second inequality holds due to the monotonicity of $\bar{V}_t^k$, and the last inequality follows from similar arguments that were used to show (4.62). Combining (4.82) with (4.83), we obtain

$$-n_t f_{\lambda_k}(a\lambda_k) \leq \widehat{V}_t(\hat{x}_u^k, \tilde{\xi}_n) - Q_t^{k'}(\hat{x}_n^k, \tilde{\xi}_n) \leq \widehat{\mathcal{V}}_{t+1}(\hat{x}_n^k) - \widehat{\mathcal{V}}_{t+1}^{k'}(\hat{x}_n^k). \qquad (4.84)$$

Take the limit $k \to \infty$ in (4.84). Then, using Theorem 4.3 and the fact that $\lim_{k\to\infty} f_{\lambda_k}(a\lambda_k) = 0$, it immediately follows from the Sandwich theorem that

$$\lim_{k\to\infty} \widehat{V}_t(\hat{x}_{\zeta_n}^k, \tilde{\xi}_n) - Q_t^{k'}(\hat{x}_n^k, \tilde{\xi}_n) \overset{a.s.}{=} 0,$$

and the proof is complete. □

Theorem 4.5. *For each $n \in \mathcal{N}$, let $\hat{x}_n$ be a limit point of $\{\hat{x}_n^k\}_{k\in\mathcal{K}_n}$. Then, $\hat{x}_n = x_n^*$ (a.s.).*

Proof. Consider a node $n \in \mathcal{N}_t$, where $t \in \hat{\mathcal{T}}$, and let $u \equiv \zeta_n$ denote the parent of n. Observe that $\mathcal{K}_n = \Upsilon_u$ by definition (and therefore, $\mathcal{K}_n = \Upsilon_u \subset \mathcal{K}_u$). Moreover, it is noted that the sequence $\{(\hat{x}_u^k, \hat{x}_n^k)\}_{k\in\mathcal{K}_n}$ belongs to the closed, convex set $\mathcal{X}_t(u, n) = \{(x_u, x_n) \in \mathbb{R}^{n_{t-1}} \times \mathbb{R}^{n_t} : A_n x_n + B_n x_u = b_n\}$. Therefore, $\{(\hat{x}_u^k, \hat{x}_n^k)\}_{k\in\mathcal{K}_n}$ must converge to a limit point in $\mathcal{X}_t(u, n)$ (see Theorem 3.2 (d) in [127]), i.e., $\lim\limits_{k\to\infty} \hat{x}_u^k = \hat{x}_u$ and $\lim\limits_{k\to\infty} \hat{x}_n^k = \hat{x}_n$, for

$k \in \mathcal{K}_n$ and $(\hat{x}_u, \hat{x}_n) \in \mathcal{X}_t(u,n)$. Next, we consider the following sequence of equalities that hold for $k \in \mathcal{K}_n$:

$$\begin{aligned}
\widehat{V}_t(\hat{x}_u, \tilde{\xi}_n) &= \lim_{k\to\infty} \widehat{V}_t(\hat{x}_u^k, \tilde{\xi}_n), \\
&= \lim_{k\to\infty} Q_t^{k'}(\hat{x}_n^k, \tilde{\xi}_n), \\
&= \lim_{k\to\infty} \{c_t(\hat{x}_n^k, \tilde{\xi}_n) + \widehat{\mathcal{V}}_{t+1}^{k'}(\hat{x}_n^k)\}, \\
&= c_t(\hat{x}_n, \tilde{\xi}_n) + \lim_{k\to\infty} \widehat{\mathcal{V}}_{t+1}^{k'}(\hat{x}_n^k), \\
&= c_t(\hat{x}_n, \tilde{\xi}_n) + \lim_{k\to\infty} \widehat{\mathcal{V}}_{t+1}(\hat{x}_n^k), \\
&= c_t(\hat{x}_n, \tilde{\xi}_n) + \widehat{\mathcal{V}}_{t+1}(\hat{x}_n).
\end{aligned} \tag{4.85}$$

The first equality follows from the continuity of $\widehat{V}_t(x_u, \tilde{\xi}_m)$ with respect to x_u. The second equality is obtained via part (i) of Lemma 4.2. The third equality follows from the definition of $Q_t^{k'}(\hat{x}_n^k, \tilde{\xi}_n)$. The fourth equality holds by the absolute continuity of $c_t(x_n, \tilde{\xi}_n)$. The fifth equality follows from Theorem 4.3. Finally, the sixth equality holds because $\widehat{\mathcal{V}}_{t+1}(x_n)$ is a continuous function. From the definition of $\widehat{V}_t(\hat{x}_u, \xi_m)$, it immediately follows from (4.85) that $\hat{x}_n$ minimizes the objective function (4.3b), and the proof is complete. □

It is important to note that the stage-wise independence assumption is not required for our convergence results. However, this assumption facilitates the sharing of cuts among the nodes at each stage $t \in \hat{\mathcal{T}}$, resulting in an improved outer approximation $\widehat{\mathcal{Q}}_t^k$ satisfying the inequality

$$\widehat{\mathcal{V}}_t^k(\hat{x}_n^k) \leq \widehat{\mathcal{Q}}_t^k(\hat{x}_n^k) \leq \widehat{\mathcal{V}}_t(\hat{x}_n^k), \quad \forall n \in \mathcal{N}_{t-1}, k \in \mathcal{K}_n. \tag{4.86}$$

However, it follows immediately from Theorem 4.3 that

$$\lim_{k\to\infty} \widehat{\mathcal{V}}_t(\hat{x}_n^k) - \widehat{\mathcal{Q}}_t^k(\hat{x}_n^k) = 0 \quad (a.s.),$$

i.e., the addition of extra Benders' cuts does not impact the convergence properties of the algorithm.

4.6 COMPUTATIONAL EXPERIMENTS

Here, we present computational results illustrating the benefits of regSDDP as compared to the standard SDDP and SDDP with quadratic regularization (quadSDDP) procedures. Specifically, we present two sets of empirical results for large-scale SAA instances of the multistage capacity expansion (MCE) and the multistage portfolio optimization (MPO) problems.

First, we introduce the MCE problem and provide detailed descriptions of the source data, experimental setup, and computational study. For the MCE problem, we compare the performances of regSDDP and standard SDDP only.

4.6.1 The Multistage Capacity Expansion Problem

We consider a variant of the MCE problem described in [24], in which the objective is to determine an optimal operational strategy for conventional power generators (e.g., coal, gas, nuclear etc.) in a distribution network with high renewable penetration. Let I be the number of different types of conventional generators in the network. Define $\mathcal{I} \equiv \{1, \ldots, I\}$. The uncertainty in stage $t \in \hat{\mathcal{T}}$ is $\xi_t = (\xi_t^d, \xi_t^w)$, where ξ_t^d and ξ_t^w denote the growth rates of demand and renewable generation at stage t, respectively. The net demand (demand minus renewable generation) at stage t is $d_t = \max\left\{d_1\xi_t^d - \eta w_1 \xi_t^w, 0\right\}$, where d_1 and w_1 are the (deterministic) initial demand and renewable generation levels in stage 1, and η is the renewable generation efficiency. It is assumed that no decisions are made in stage 1, while operational decisions are made from stage 2 onwards, which are described next. Let $x_{t,i}^n$ be the new capacity installed for generator type i in stage t, which incurs a per-unit installation cost $c_{t,i}^n$. Let $x_{t,i}^r$ be the capacity curtailed for generator type i in stage t at a per-unit curtailment cost $c_{t,i}^r$. The total installed capacity of generator type i in stage t is $x_{t,i}^h$, which incurs a per-unit holding cost $c_{t,i}^h$. The operating level of generator type i in stage t is $x_{t,i}^g$, and let $c_{t,i}^g$ be the per-unit operating cost. The amount of unsatisfied demand in stage t is denoted by x_t^s, and let c_t^s be the per-unit shortage cost. Furthermore, let $U_{t,i}^n, U_{t,i}^r$ and $U_{t,i}^h$ denote the maximum values of $x_{t,i}^n, x_{t,i}^r$ and $x_{t,i}^h$, respectively. Then, the MCE problem can be formulated as the following MSLP:

$$\min \ \mathbb{E}\left(\sum_{t\in\hat{\mathcal{T}}} c_t^s x_t^s + \sum_{t\in\hat{\mathcal{T}}}\sum_{i\in\mathcal{I}} \left(c_{t,i}^n x_{t,i}^n + c_{t,i}^r x_{t,i}^r + c_{t,i}^h x_{t,i}^h + c_{t,i}^g x_{t,i}^g\right)\right) \tag{4.87a}$$

$$\text{s.t.} \ x_{t,i}^h - x_{t-1,i}^h - x_{t,i}^n + x_{t,i}^r = 0, \qquad \forall t\in\hat{\mathcal{T}}, i\in\mathcal{I}, \tag{4.87b}$$

$$x_t^s + \sum_{i\in\mathcal{I}} x_{t,i}^g \geq d_t, \qquad \forall t\in\hat{\mathcal{T}}, \tag{4.87c}$$

$$0 \leq x_{t,i}^g \leq x_{t,i}^h, \qquad \forall t\in\hat{\mathcal{T}}, i\in\mathcal{I}, \tag{4.87d}$$

$$0 \leq x_{t,i}^n \leq U_{t,i}^n, \qquad \forall t\in\hat{\mathcal{T}}, i\in\mathcal{I}, \tag{4.87e}$$

$$0 \leq x_{t,i}^r \leq U_{t,i}^r, \qquad \forall t\in\hat{\mathcal{T}}, i\in\mathcal{I}, \tag{4.87f}$$

$$0 \leq x_{t,i}^h \leq U_{t,i}^h, \qquad \forall t\in\hat{\mathcal{T}}, i\in\mathcal{I}, \tag{4.87g}$$

$$0 \leq x_t^s \leq d_t, \qquad \forall t\in\hat{\mathcal{T}}, \tag{4.87h}$$

where, without loss of generality, we assume that $x_{0,i}^h = 0$ for all $i \in \mathcal{I}$. The constraints (4.87b) describe the evolution of the total capacity of each generator type, while (4.87c) represent the supply-demand balance constraints that account for demand shortages, if any. The constraints (4.87d) – (4.87h) bound the operational decisions in each stage. We assume $U_{t,i}^r \geq U_{t-1,i}^h$ for all $t \in \hat{\mathcal{T}}$ so that (4.87) has the relatively complete recourse property (see the discussion in [24]).

Data Description. We used the annual demand and wind-generation data of a German distribution network described in [38]. All energy quantities are reported in gigawatt-hours (GW-h). From the data, we set $\eta = 0.52$, $d_1 = 4.88 \times 10^5$ GW-h, and $w_1 = 3.71 \times 10^5$ GW-h. We considered three types of conventional generators, the coal-fired power plant, combined cycle-gas turbine and open cycle-gas turbine, whose per-unit installation and operating costs (in million euros per GW-h) are summarized in Table 10. We assumed that there were no holding or curtailment costs, i.e., $c_{t,i}^r = c_{t,i}^h = 0$ for all $i \in \mathcal{I}$. Using a discount factor of 10%, we set $c_{t,i}^n = 0.9^t \times \ell_i$, $c_{t,i}^g = 0.9^t \times g_i$ and $c_t^s = 0.9^t \times 2000$. The upper bounds $U_{t,i}^n, U_{t,i}^r, U_{t,i}^h$ were all set to 10^5 GW-h.

Description of Experiments. We considered SAA instances of (4.87) where each stage represents a year. We assume $\xi = \{\xi_t : t \in \hat{\mathcal{T}}\}$ is stage-wise independent, and ξ_t^d and ξ_t^w are mutually independent log-normal random variables, i.e., $\log(\xi_t^d) \sim N\left(\mu_d, \sigma_d^2(t)\right)$

Table 10: Annual per-unit installation and operating costs (million euros per GW-h).

Index (i)	Generation system	Installation cost (ℓ_i)	Operating cost (g_i)
1	Coal-fired power plant	1229.28	133.95
2	Combined cycle-gas turbine	569.50	188.29
3	Open cycle-gas turbine	289.35	290.27

and $\log(\xi_t^w) \sim N\left(\mu_w, \sigma_w^2(t)\right)$, where $\mu_d = 0.20$, $\mu_w = 0.15$, $\sigma_d(t) = 0.10 + 0.010t$, and $\sigma_w(t) = 0.25 + 0.025t$. For the computational experiments, we varied the number of stages T from 50 to 250 in increments of 10, i.e., $T \in \{50, 60, \ldots, 250\}$. For each value of T, we generated 30 T-stage scenario trees where each non-leaf node had $N \in \{15, 30\}$ children; thus, each scenario tree had N^{T-1} scenarios. To construct a scenario tree, we randomly sampled N realizations of ξ_t for each $t \in \hat{\mathcal{T}}$ using the Latin hypercube sampling (LHS) method of [41]. We prefer the LHS method over standard Monte-Carlo sampling because it generates more representative scenario trees, thereby reducing the bias and the variance of the SAA lower bound $\hat{z}_\ell$ (see [99, 52]).

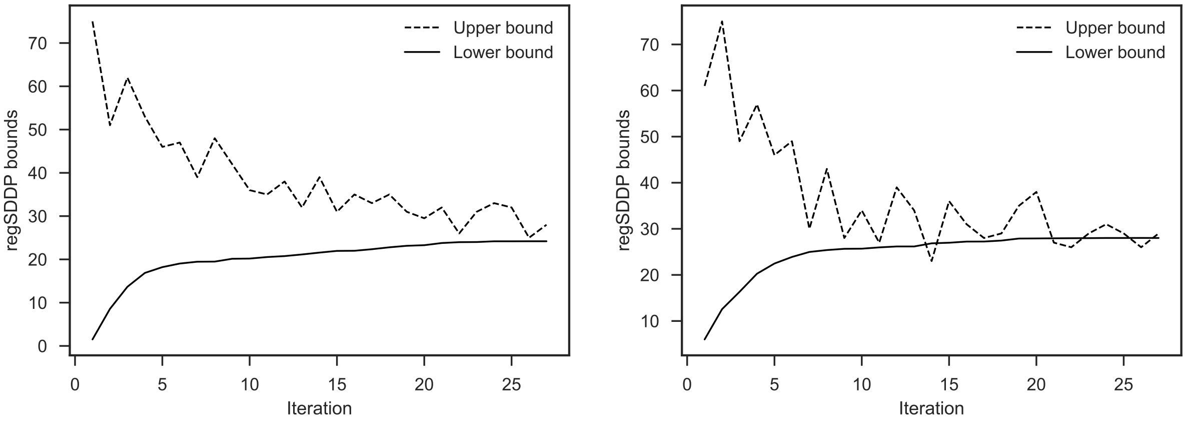
Each SAA instance was solved using the standard SDDP and regSDDP procedures. For both procedures, only one forward-pass scenario was sampled at each step, i.e., $M \equiv |\mathcal{J}^k| = 1$ for all $k \in \mathbb{N}$. This kept the overall computation times within reasonable limits for both procedures. We observed that the performance of both algorithms – in terms of lower-bound quality and convergence rate – was consistently better for $M = 1$ compared to $M > 1$, with the exception of a few instances with $T > 200$ in which the results were marginally better for $M \in \{2, 3\}$. Similar benefits for the case $M = 1$ were also observed by [138, 173] and [151]. The maximum number of iterations for both procedures was set to $k_{\max} = 500$. We used a more conservative termination criterion than (4.15) that is based on the rate of improvement of the lower bounds (see [136] for the disadvantages of using the criterion in (4.15)). Specifically, for $\epsilon = 10^{-4}$, both procedures were terminated at an iteration

$$k^* = \min\left\{k_{\max}, \operatorname{argmin}\left\{k \in \mathbb{N} : \frac{\hat{z}_\ell^k - \hat{z}_\ell^{k-15}}{\hat{z}_\ell^k} \times 100 \leq \epsilon\right\}\right\}, \tag{4.88}$$

where the inequality in (4.88) represents an upper bound on the percentage improvement in the lower bounds over a rolling window of 15 iterations (a similar criterion was used by [137]). We used the SCAD penalty function for regSDDP in our experiments. We chose a previous trial point as the current stability center at each stage, i.e., $\check{x}_t = \hat{x}_t^{k-1}(\tilde{\xi}_t^h)$ for each $t \in \hat{\mathcal{T}}$. We chose the penalty parameter at each step according to $\lambda_k = \lambda_0 \rho^k$, where $\lambda_0 = (2 \log(\max\{n_t : t \in \hat{\mathcal{T}}\}))^{0.5} = 1.47$ was set *a priori* using the universal threshold rule of [42]. We implemented a grid search over the sets $A = \{2.0, 2.1, 2.2, \ldots, 10.0\}$ and $P = \{0.7, 0.72, \ldots, 1.0\}$ to tune the parameters a and ρ, respectively. For each $(a, \rho) \in A \times P$, we ran the regSDDP algorithm for 100 iterations on 30 T-stage SAA instances, where $T \in \{50, 60, \ldots, 100\}$. The best lower bounds (on average) at the end of 100 iterations were obtained for $a = 4.2$ and $\rho = 0.86$; we used these parameter values for regSDDP in the rest of the experiments.

Both procedures were coded in Python 2.7. The nodal problems were solved using the Gurobi 7.0 solver. The MILP tolerance level for regSDDP was set to 10^{-3}. The algorithms were implemented on two 64-bit, 6th generation Intel® Core™ i7, 128 GB, 3.2 GHz Windows machines.

Results and Discussion. We first demonstrate the efficacy of the regSDDP algorithm in solving large SAA instances. Figure 20 illustrates the evolution of the upper and lower bounds generated by regSDDP for two distinct SAA instances with $T = 100$ and $N = 15$. For both problem instances, the lower bounds stabilized after 25 iterations. Note the rapid increase of the lower bounds in the initial iterations (exploration phase). Once a sufficient number of good-quality cuts is generated, the lower bounds improve slowly during the exploitation phase. Moreover, note that the first few upper bounds are loose due to the poor induced policies generated initially. Furthermore, these upper bounds exhibit greater variability as the algorithm searches for better trial solutions during exploration. However, as the outer approximations improve and exploitation becomes more dominant, the upper bounds exhibit less variability. Similar trends of the regSDDP bounds were observed for other SAA instances. For the second problem instance, note that the upper bound is less than the lower bound at certain iterations. However, this does not imply that the problem instance has been solved because $\hat{z}_u^k$ is a statistical, and not a valid, upper bound of $\hat{z}_\ell$, and the term $(\hat{z}_u^k - \hat{z}_\ell^k)$ is an upward-biased estimator of the true optimality gap $(\hat{z}_\ell - \hat{z}_\ell^k)$.



(a) Instance 1.

(b) Instance 2.

Figure 20: Evolution of regSDDP bounds for two SAA instances ($T = 100,\ N = 15$).

Figure 21 illustrates the evolution of the average values and the associated 95% confidence intervals (shaded regions) of the regSDDP bounds for the 30 SAA instances with 100 stages.

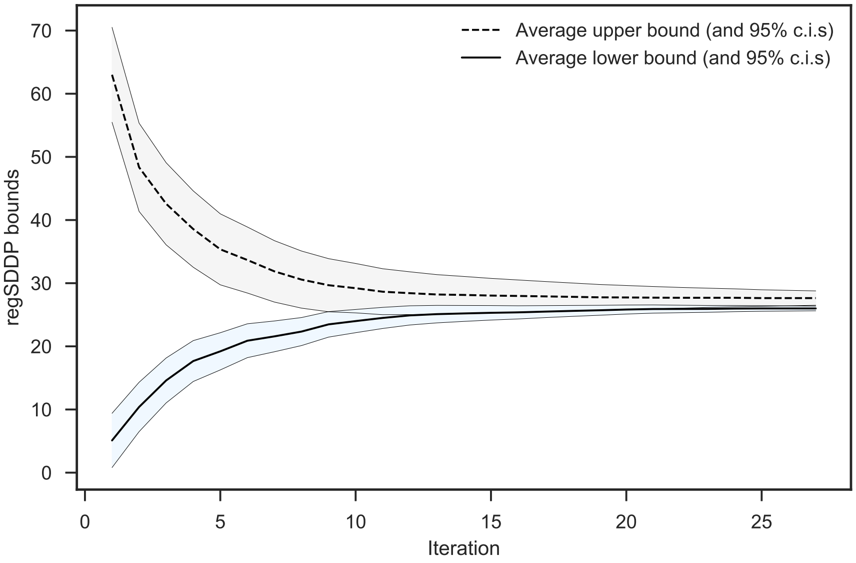


Figure 21: Average bounds and 95% confidence intervals for 30 instances ($T = 100,\ N = 15$).

On average, regSDDP took approximately 27 minutes and 25 iterations to solve the SAA instances. Note that the variance in the lower bounds diminishes rapidly after 25 iterations. Thus, the average value of the lower bound at the 25th iteration (or beyond) provides a high-quality estimate of the true optimal value z^* of model (4.87).

Next, we compare the performance of standard SDDP and regSDDP for SAA instances of different sizes. The computational results for the cases $N = 15$ and $N = 30$ are summarized in Tables 11 and 12, respectively. The results are expressed in the form $u \pm v$ (w), where u, v and w denote the average value, standard deviation and range, respectively, of the values observed for each row-column combination. Tables 11 and 12 reveal that, for all values of T, regSDDP produced superior lower bounds (on average), with less variability, compared to the lower bounds obtained via standard SDDP.

Table 11: Computational results of regSDDP and standard SDDP for $N = 15$.

Stages	**Lower bound**		**Number of iterations**		**Solution time (mins)**	
	SDDP	regSDDP	SDDP	regSDDP	SDDP	regSDDP
50	14.87 ± 0.42 (0.73)	14.88 ± 0.33 (0.57)	44 ± 2 (3)	16 ± 1 (2)	11.23 ± 3.59 (6.71)	9.50 ± 4.53 (8.48)
75	19.38 ± 0.84 (1.45)	19.62 ± 0.57 (0.98)	63 ± 4 (6)	19 ± 2 (3)	19.95 ± 5.71 (10.67)	15.01 ± 6.21 (11.62)
100	25.46 ± 1.44 (2.43)	25.92 ± 0.91 (1.58)	92 ± 5 (8)	27 ± 2 (3)	42.93 ± 8.01 (14.97)	27.45 ± 7.23 (13.53)
125	33.07 ± 2.04 (3.51)	33.87 ± 1.47 (2.53)	124 ± 7 (12)	38 ± 3 (6)	82.67 ± 11.72 (21.91)	55.12 ± 12.21 (22.84)
150	42.97 ± 2.73 (4.74)	43.24 ± 1.79 (3.06)	161 ± 9 (15)	51 ± 5 (9)	152.95 ± 15.62 (29.24)	96.07 ± 17.05 (31.89)
175	50.33 ± 3.52 (6.12)	52.24 ± 2.30 (3.99)	197 ± 11 (18)	65 ± 8 (13)	233.12 ± 19.11 (35.75)	153.84 ± 22.97 (42.97)
200	61.06 ± 4.46 (7.73)	63.78 ± 3.22 (5.57)	243 ± 13 (22)	83 ± 10 (17)	360.45 ± 23.65 (44.25)	242.08 ± 29.15 (54.55)
225	72.08 ± 5.15 (8.92)	74.82 ± 3.74 (6.48)	285 ± 15 (26)	107 ± 12 (21)	523.50 ± 28.72 (53.72)	383.47 ± 35.97 (67.19)
250	84.21 ± 5.89 (10.21)	87.69 ± 4.14 (7.17)	322 ± 17 (30)	136 ± 16 (27)	745.97 ± 34.51 (65.55)	589.33 ± 44.79 (83.79)

Figure 22 illustrates the higher quality of the regSDDP lower bounds for the case $N = 15$ (the dashed lines in Figure 22 indicate the first, second and third quartiles). Here, we make two important observations. First, note that the improvement in the lower bounds is more pronounced for larger SAA instances. For example, when $N = 15$, the average lower bound

Table 12: Computational results of regSDDP and standard SDDP for $N = 30$.

Stages	**Lower bound**		**Number of iterations**		**Solution time (mins)**	
	SDDP	regSDDP	SDDP	regSDDP	SDDP	regSDDP
50	15.05 ± 0.46 (0.79)	15.64 ± 0.33 (0.56)	66 ± 3 (4)	24 ± 1 (2)	28.05 ± 4.98 (8.96)	24.48 ± 5.37 (9.76)
75	19.88 ± 0.82 (1.42)	20.85 ± 0.50 (0.83)	94 ± 4 (7)	29 ± 2 (3)	50.87 ± 7.25 (13.07)	36.98 ± 7.24 (13.14)
100	27.03 ± 1.37 (2.38)	28.97 ± 0.89 (1.49)	138 ± 6 (9)	40 ± 3 (4)	109.48 ± 9.81 (17.70)	69.13 ± 9.58 (17.41)
125	35.68 ± 1.94 (3.35)	37.96 ± 1.59 (2.75)	186 ± 8 (14)	55 ± 4 (7)	210.80 ± 14.33 (25.84)	135.58 ± 14.78 (26.85)
150	44.43 ± 2.73 (4.72)	48.01 ± 1.94 (3.36)	242 ± 10 (17)	74 ± 6 (11)	390.83 ± 18.59 (33.52)	236.92 ± 20.73 (37.65)
175	54.69 ± 3.71 (6.42)	59.76 ± 2.43 (4.21)	296 ± 11 (20)	95 ± 9 (16)	595.45 ± 22.29 (40.17)	382.22 ± 27.58 (50.11)
200	66.71 ± 4.60 (7.96)	72.25 ± 3.40 (5.89)	365 ± 14 (25)	120 ± 11 (19)	920.41 ± 27.57 (49.71)	595.00 ± 33.04 (60.02)
225	79.82 ± 5.12 (8.87)	86.17 ± 3.83 (6.64)	427 ± 18 (29)	155 ± 14 (23)	1342.92 ± 32.85 (59.22)	944.21 ± 39.88 (72.45)
250	90.91 ± 5.733 (9.93)	95.87 ± 4.31 (7.47)	484 ± 21 (33)	198 ± 16 (28)	1906.15 ± 38.86 (70.05)	1458.6 ± 47.90 (87.04)

improved by 4.17% for the case $T = 250$, compared to 0.067% for $T = 50$. Second, note that the lower-bound distributions are slightly left-skewed. The skewness can be removed by solving a larger number (> 30) of SAA instances for each T; however, this will increase the computational overhead without necessarily improving the quality of the lower bounds.

Tables 11 and 12 also reveal that regSDDP required fewer iterations (on average) to solve an SAA instance, compared to the standard SDDP; i.e., the convergence rate of the SDDP algorithm improved when regularization was employed. This improvement is more pronounced for larger problem instances. For example, Figure 23 depicts the improved convergence rate for a particular SAA instance with 150 stages. For this SAA instance, regSDDP took 45 iterations to reach a lower-bound level $\hat{z}_{\ell}^{45} = 47.53$ before terminating. In comparison, standard SDDP required 173 iterations to reach the same value. That is, standard SDDP required over 384.44% more iterations to attain the same solution quality. To further illustrate the improved convergence rate of regSDDP, we analyzed the lower bounds obtained from both procedures after a sufficient number of iterations, without enforcing the

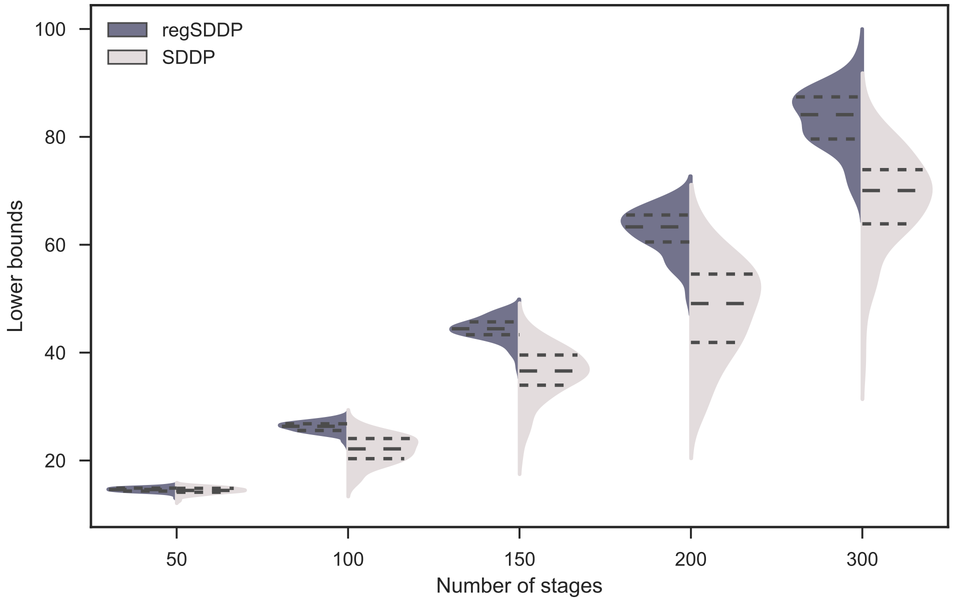


Figure 22: Distribution of the lower bounds of regSDDP and standard SDDP for $N = 15$.

termination criterion in (4.88). Figure 24 depicts a box plot of the lower bounds obtained at the end of the 100th iteration. Note that, for all values of T, the regSDDP lower bounds have higher average values and less variability compared to the lower bounds of standard SDDP. Again, the improvement is more pronounced at higher values of T due to the slow convergence rate of standard SDDP for larger problem instances.

In fact, we observed that a large fraction of the SAA instances were solved within 100 iterations of regSDDP, a trend that is illustrated in Figure 25. By contrast, far fewer instances were solved within 100 iterations of the standard SDDP. For example, when $T = 200$, regSDDP solved around 59% of the problem instances within 100 iterations, while SDDP could not solve a single such instance. Only for $T = 250$ was regSDDP unable to solve any SAA instance within 100 iterations.

Table 13 reports the average time per iteration (ATI) for both procedures. We expected the higher ATI values for regSDDP as MILP models, of increasing complexity, are solved in the forward pass. However, it is instructive to note that the overall solution times for

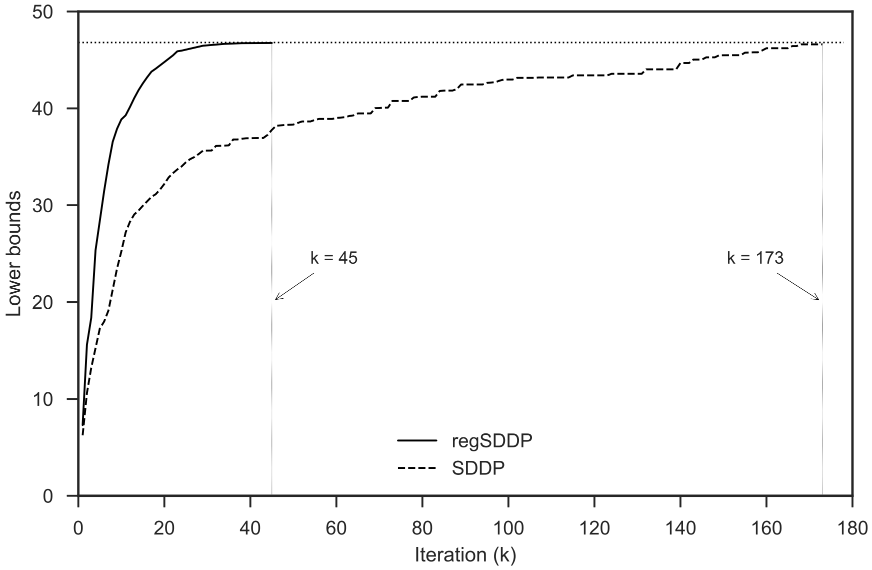


Figure 23: Reduction in the number of iterations to solve a SAA instance ($T = 150$, $N = 15$).

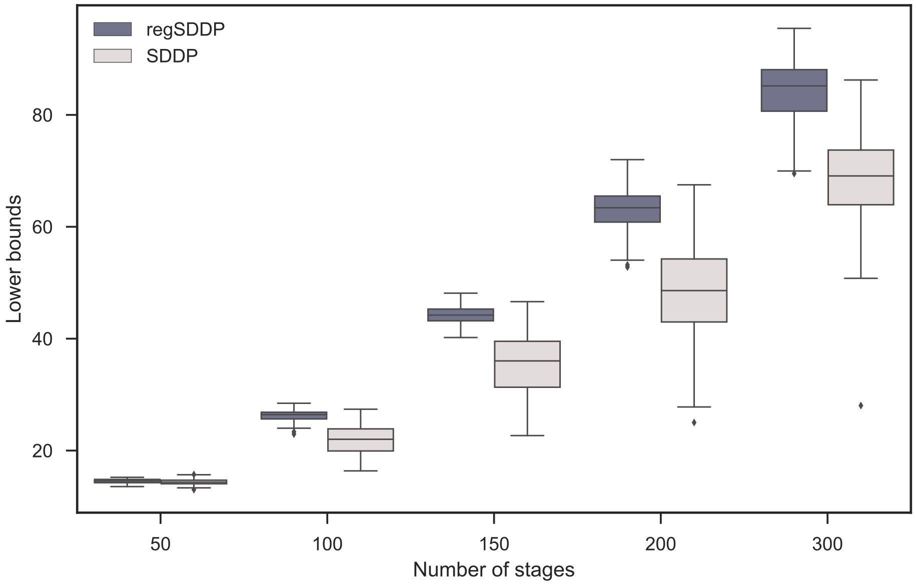


Figure 24: Lower bounds of regSDDP and standard SDDP at the end of the 100th iteration.

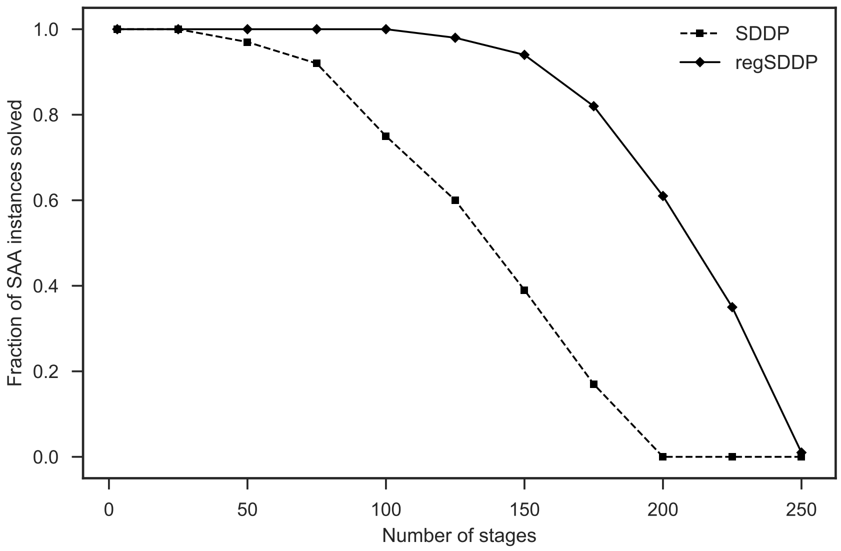


Figure 25: Fraction of SAA instances ($N = 15$) that were solved within 100 iterations.

regSDDP are smaller as compared to the standard SDDP, as revealed in Tables 11 and 12. For example, regSDDP required approximately 7.5 hours less (on average) than the standard SDDP to solve a SAA instance with 250 stages. This reduction in computation time stems from the fewer number of iterations required by regSDDP.

Table 13: Average computation time per iteration of regSDDP and standard SDDP (in seconds).

Stages	$N = 10$		$N = 30$	
	SDDP	regSDDP	SDDP	regSDDP
50	15.21	33.84	25.52	61.33
75	19.04	42.31	32.30	76.51
100	28.27	57.35	47.60	103.74
125	40.22	81.76	68.64	147.90
150	57.01	113.22	96.91	192.25
175	71.50	139.48	120.66	241.42
200	89.22	174.67	151.32	298.65
225	111.26	208.21	188.88	365.91
250	139.48	256.73	236.35	459.27

4.6.2 A Multistage Portfolio Optimization Problem

Next, we compare the performances of the regSDDP and the quadSDDP procedures for a variant of the multistage portfolio optimization (MPO) problem described in [88]. Considered is a risk-neutral investor who seeks to optimally allocate capital among a portfolio of risky assets with uncertain rates of returns. The objective is to maximize the expected total return on investments over a finite trading horizon. Suppose there are m assets in the investor's portfolio comprising the set $\mathcal{M} = \{1, \ldots, m\}$, where the first $m-1$ assets are risky (e.g., stocks) and the mth asset is risk-free (e.g., cash reserve); define $\mathcal{M}' \equiv \mathcal{M} \setminus \{m\}$ to represent the set of risky assets. The uncertainty in stage t is $\xi_t = (\xi_t^i : i \in \mathcal{M}')$, where ξ_t^i is the rate of return on the ith risky asset at stage t. It is assumed that the rate of return on the non-risky asset, denoted by ξ_t^m, is known with certainty. Let $x_{t,i}^h$ be the volume of asset $i \in \mathcal{M}$ that the investor holds at the end of stage t. The investor can either buy more or sell off a portion of a risky asset at each stage; let $x_{t,i}^b$ and $x_{t,i}^s$ denote the volumes of asset i bought and sold, respectively, at the start of stage t. Each trading decision incurs a transaction cost that is proportional to the trading volume; let c_i^b and c_i^s be the per-unit transaction costs of buying and selling asset i, respectively. Furthermore, only a limited volume of assets can be traded at each stage; let U_b^i and U_s^i be the upper bounds on $x_{t,i}^b$ and $x_{t,i}^s$, respectively. Without loss of generality, assume that $x_{1,i}^h = 0$ for all $i \in \mathcal{M}'$. The initial amount of the non-risky asset, $x_{1,m}^h$, is known with certainty. Then, the risk-neutral MPO problem can be formulated as the following MSLP:

$$\max \;\; \mathbb{E}\left(\sum_{i \in \mathcal{M}} \xi_T^i x_{T,i}^h\right) \tag{4.89a}$$

$$\text{s.t.} \;\; x_{t,i}^h = \xi_{t-1}^i x_{t-1,i}^h - x_{t,i}^s + x_{t,i}^b, \qquad \forall t \in \hat{\mathcal{T}}, i \in \mathcal{M}', \tag{4.89b}$$

$$x_{t,m}^h = \xi_{t-1}^m x_{t-1,m}^h + \sum_{i \in \mathcal{M}'} (1 - c_i^s) x_{t,i}^s - \sum_{i \in \mathcal{M}'} (1 + c_i^b) x_{t,i}^b, \qquad \forall t \in \hat{\mathcal{T}}, \tag{4.89c}$$

$$x_{t,i}^s \leq \xi_{t-1}^i x_{t-1,i}^h, \qquad \forall t \in \hat{\mathcal{T}}, i \in \mathcal{M}', \tag{4.89d}$$

$$0 \leq x_{t,i}^b \leq U_b^i, \qquad \forall t \in \hat{\mathcal{T}}, i \in \mathcal{M}', \tag{4.89e}$$

$$0 \leq x_{t,i}^s \leq U_s^i, \qquad \forall t \in \hat{\mathcal{T}}, i \in \mathcal{M}', \tag{4.89f}$$

$$x_{t,i}^h \geq 0, \qquad \forall t \in \hat{\mathcal{T}}, i \in \mathcal{M}. \tag{4.89g}$$

Constraints (4.89b) define the volumes of risky assets held at each stage after accounting for the intrastage trading decisions. Constraints (4.89c) describe the evolution of the non-risky asset volumes. Constraints (4.89d) prevents selling a volume of risky asset that exceeds the asset's current holding position. Constraints (4.89e)–(4.89f) are the bounding constraints that represent the limited liquidity of the risky assets. Constraints (4.89g) prevent any short sale of the risky assets.

Data Description. We considered SAA instances of (4.89) where each stage represents 5 working days (equivalent to a calender week). We considered all 102 stocks from the S&P 100 index to define our portfolio of risky assets. Historical pricing data of each stock were obtained from Yahoo Finance (https://finance.yahoo.com) for 4250 working days between January 1, 2000 and August 23, 2017. Let $p(i, j)$ be the closing price of stock i on day j, and

$$\mathcal{R}_i = \left\{ \frac{p(i, j+5)}{p(i, j)} : j = 1, \dots, 4245 \right\}$$

the set of historical weekly returns of stock i. For each $i \in \mathcal{M}'$, we set $\mathcal{R}_i$ as the support of ξ_t^i for all $t \in \hat{\mathcal{T}}$. The elements in $\mathcal{R}_i$ were assumed to be uniformly distributed. The investor was assumed to have 1000 units of cash at the start of the investment horizon, i.e., $x_{1,m}^h = 1000$.

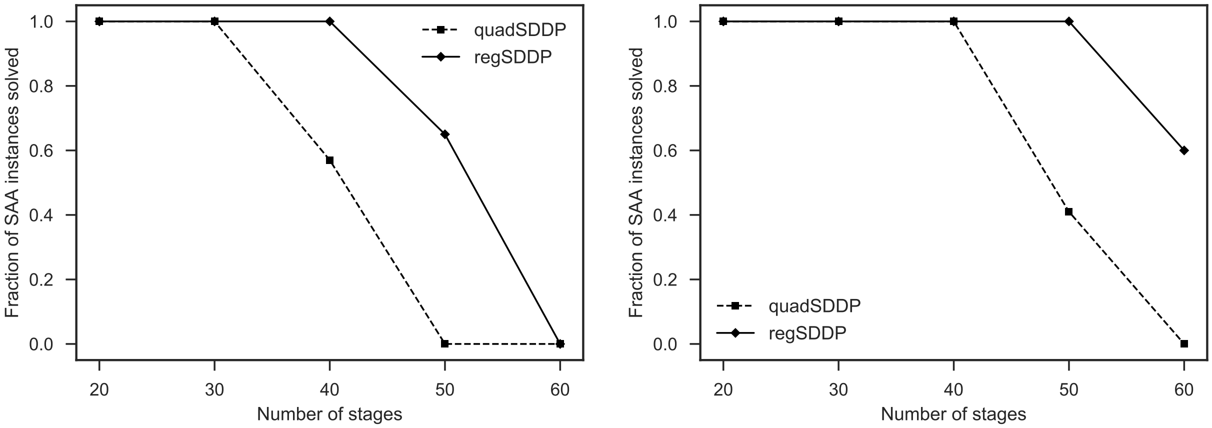
Description of Experiments. For the computational experiments, we varied the number of stages T from 20 to 60 in increments of 10. For each value of T, we again generated 30 T-stage scenario trees using the Latin hypercube sampling (LHS) method. Each non-leaf node in our scenario trees had $N = 20$ children. We used the SCAD penalty function for the regSDDP procedure. For both procedures, we chose a previous trial point as our current stability center and chose the penalty parameter at each step according to $\lambda_k = 2.28 \times 0.85^k$. To tune the parameter a, we ran 100 iterations of regSDDP for each $a \in \{2.0, 2.2, \dots, 10.0\}$ on the 30 SAA instances for $T = 20$; the best lower bounds were obtained for $a = 4.8$. We used the quadratic programming (QP) and MILP solvers in Gurobi 7.0 for our experiments. The tolerance level for the QP solver was set to 10^{-4}. For the MILP solver, we used a step function to progressively reduce the tolerance levels. Specifically, we set the MILP tolerance level to 10^{-1} for the first 20 iterations, 10^{-2} for iterations 21 to 40, and 10^{-4} for all subsequent iterations up to $k_{\max} = 500$. Finally, we set the parameter ϵ to 10^{-2} in (4.88).

Results and Discussion. The computational results for both procedures are summarized in Table 14. Note that both procedures produced lower bounds of similar quality for lower values of T.However, regSDDP produced higher lower bounds (on average) than quadSDDP for larger values of T. Moreover, note that regSDDP required fewer iterations (on average) than quadSDDP, indicating an improved convergence rate for regSDDP. Similar to the results for the MCE problem, the improvement in the convergence rate is more pronounced for larger problem instances. For example, when $T = 60$, regSDDP required about 38% fewer iterations (on average) than quadSDDP.

Table 14: Computational results of regSDDP and quadSDDP for $N = 20$.

Stages	**Lower bound**		**Number of iterations**		**Solution time (mins)**	
	quadSDDP	regSDDP	quadSDDP	regSDDP	quadSDDP	regSDDP
20	1046.36 ± 9.42 (16.25)	1046.53 ± 8.98 (17.49)	62 ± 4 (7)	36 ± 3 (5)	38.23 ± 3.12 (4.71)	42.71 ± 3.48 (5.05)
30	1068.94 ± 13.24 (26.37)	1069.12 ± 13.88 (25.18)	79 ± 7 (11)	48 ± 5 (9)	69.78 ± 4.36 (7.64)	78.26 ± 5.55 (8.72)
40	1086.38 ± 19.88 (37.17)	1087.05 ± 20.17 (35.92)	105 ± 10 (18)	69 ± 6 (13)	136.53 ± 5.91 (9.66)	164.39 ± 6.82 (13.55)
50	1102.79 ± 24.29 (47.49)	1104.18 ± 25.68 (56.31)	149 ± 13 (24)	106 ± 9 (17)	293.03 ± 8.35 (12.37)	328.44 ± 9.17 (14.49)
60	1114.17 ± 34.39 (59.73)	1116.31 ± 35.25 (62.34)	203 ± 17 (32)	147 ± 12 (20)	527.25 ± 10.39 (15.02)	594.47 ± 12.41 (17.84)

Figure 26 depicts the fraction of SAA instances solved by both procedures within 75 and 150 iterations, respectively. For example, when $T = 50$, regSDDP solved around 63% of the problem instances within 75 iterations as compared to none by quadSDDP; the corresponding values were 100% for regSDDP and only 41% for quadSDDP when the number of iterations was increased to 150. In fact, we observed that regSDDP produced better lower bounds (on average) than quadSDDP when both procedures were terminated after a fixed number of iterations ($< k^*$). This trend is illustrated in Figure 27, which depicts a box plot of the lower bounds obtained after 100th iterations of both procedures.



(a) Solved within 75 iterations.

(b) Solved within 150 iterations.

Figure 26: Fraction of SAA instances solved within (a) 75 and (b) 150 iterations, respectively.

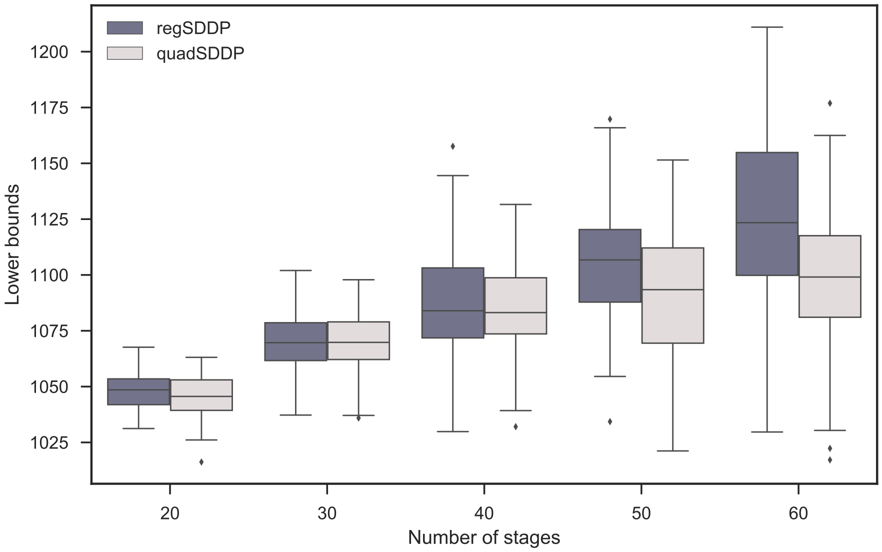


Figure 27: Lower bounds of regSDDP and quadSDDP procedures after 100 iterations.

5.0 CONCLUSIONS AND FUTURE RESEARCH

In Chapter 2, we examined optimal energy storage and flow strategies in a 2-bus distribution network with storage devices and line losses. The network operator's objective is to minimize the total expected discounted costs incurred over a finite planning horizon by optimally selecting the amount of energy to charge to, or discharge from, the storage devices, the amount of energy to buy from, or sell to, the grid and the amount of energy to transmit between the buses. By way of a finite-horizon, discounted cost MDP model, we established the monotonicity of the optimal policy with respect to the storage levels. Moreover, we proved the multimodularity of the value function in the storage levels, and that the optimal storage decisions at each stage exhibit bounded sensitivities. Significantly, we also established bounds that compare the cost of the 2-bus network to the costs of two comparable networks with pooled and decentralized storage configurations, respectively. The results of the 2-bus network were extended to more general multi-bus network topologies. The usefulness of the main results was illustrated by way of a numerical example using real pricing and wind generation data. Our results highlighted the benefits of using the network model over that of single-storage models that do not account for interactions in a distribution network.

While the model and main structural results are useful, they can be improved in a few important ways. First, it is important to note that the structural results were established without accounting for reactive power flow and voltage level constraints in distribution networks. It will be necessary to examine more rigorous power-flow models (cf. [17, 163]) and extend our results to more realistic distribution networks. Second, it will be instructive to examine models that consider multiple value-adding uses of storage (e.g., arbitrage, ancillary support, or backup energy). Third, it will be instructive to develop easily computable, tighter bounds for the optimal cost, similar to those established in Theorem 2.2.

In Chapter 3, we proposed a multistage stochastic programming model to obtain viable energy procurement and storage strategies for grid-connected microgrids. The model includes three sources of uncertainty: demand, renewable generation, and real-time electricity prices. This framework enables microgrid operators to determine the appropriate amount of electricity to procure from the main grid and the amount to charge to, or discharge from, local storage devices, to satisfy demand and power flow requirements during each stage of a finite planning horizon. Our extensive computational study on a realistic 4-bus microgrid revealed that the multistage stochastic programming model achieves significant cost reductions as compared to myopic and non-storage policies, as well as policies obtained using a two-stage SP formulation. Moreover, our customized SDDP algorithm is able to address the computational challenges associated with the multistage structure of the problem. Our customization, which uses dynamic cut selection and a novel lower bound improvement strategy, drastically outperforms the standard SDDP algorithm and also demonstrates its scalability to potentially much larger problem instances. It is also conjectured that our improved solution method can be extended to address the computational issues of multistage electric generation expansion and hydropower scheduling problems.

While the multistage stochastic programming model is very useful for prescribing solutions that reduce total electricity costs, it can be improved in several important ways. First, it will be instructive to model the case in which the microgrid operators have the flexibility to sell excess energy back to the main grid and exploit arbitrage opportunities in electricity markets. This feature is likely to alter the microgrid's procurement and storage strategies significantly. Second, the stage-wise independence assumption of the uncertain variables may be restrictive. For example, price and wind generation levels may exhibit autocorrelation over time; therefore, it will be instructive in future work to explore solution approaches that relax the stage-wise independence assumption. Third, more sophisticated sampling procedures, such as importance sampling, and other variance reduction techniques, can be used to identify a set of scenarios that balances the exploration-exploitation tradeoff in the SDDP algorithm (see [87] for additional details). Finally, extending the the SDDP algorithms to handle problems of higher dimensionality is an important area of future work.

In Chapter 4, we presented a new nonconvex regularization scheme to enhance the quality of outer approximations obtained via the SDDP algorithm. The proposed regularization scheme uses two well-known folded-concave penalty functions to regularize trial solutions generated in the forward pass of SDDP. We proved that the nonconvex regularization problem admits equivalent mixed-integer linear programming (MILP) formulations, which facilitates the use of state-of-the-art MILP solvers within the SDDP framework. Furthermore, we established provable convergence guarantees of this regularized SDDP algorithm, called regSDDP, under mild regularity conditions. The computational benefits of our regularization scheme were illustrated by way of a comprehensive numerical study for two large-scale multistage stochastic linear programming models. This study revealed significant improvement in solution quality and convergence rate for regSDDP over those of the standard SDDP and SDDP with quadratic regularization, especially for high-dimensional problem instances.

BIBLIOGRAPHY

[1] U. E. I. Adminstration. Annual energy outlook 2013. Technical report, U.S. Department of Energy, 2013.

[2] I. E. Agency. Distributed generation in liberalised electricity markets. Technical report, Organisation for Economic Co-operation and Development (OECD), 2002.

[3] K. Ahlert and C. van Dinther. Sensitivity analysis of the economic benefits from electricity storage at the end consumer level. In *Proceedings of the IEEE PowerTech Conference*, pages 1–8, 2009.

[4] S. Ahmed. Two-stage stochastic integer programming: A brief introduction. In J. Cochran, L. Cox, P. Keskinocak, J. Kharoufeh, and J. Smith, editors, *Wiley Encyclopedia of Operations Research and Management Science*. John Wiley & Sons, Inc., 2011.

[5] E. Altman, B. Gaujal, and A. Hordijk. Multimodularity, convexity, and optimization properties. *Mathematics of Operations Research*, 25(2):324–347, 2000.

[6] O. Ardakanian, S. Keshav, and C. Rosenberg. *Integration of Renewable Generation and Elastic Loads into Distribution Grids*. Springer, Cham, 2016.

[7] T. Asamov and W. Powell. Regularized decomposition of high-dimensional multistage stochastic programs with Markov uncertainty. 2015. Appeared in arXiv, URL: https://arxiv.org/abs/1505.02227v3.

[8] P. Asmus. Microgrids, virtual power plants and our distributed energy future. *The Electricity Journal*, 23(10):72–82, 2010.

[9] P. Asmus, A. Lauderbaugh, and M. Lawrence. Market data: Microgrids: Forecasts for commercial/industrial, community/utility, campus/institutional, military, and remote microgrids: 2013-2020. Technical report, Navigant Research, 2013.

[10] I. Atzeni, L. G. Ordóñez, G. Scutari, D. P. Palomar, and J. R. Fonollosa. Day-ahead bidding strategies for demand-side expected cost minimization. In *IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pages 91–96, 2012.

[11] M. Badawy, F. Cingoz, and Y. Sozer. Battery storage sizing for a grid tied PV system based on operating cost minimization. In *IEEE Energy Conversion Congress and Exposition (ECCE)*, pages 1–7, 2016.

[12] S. Bahramirad, W. Reder, and A. Khodaei. Reliability-constrained optimal sizing of energy storage system in a microgrid. *IEEE Transactions on Smart Grid*, 3(4):2056–2062, 2012.

[13] A. Bar-Noy, Y. Feng, M. Johnson, and O. Liu. When to reap and when to sow – lowering peak usage with realistic batteries. In *Lecture Notes in Computer Science*, volume 5038, pages 194–207. Springer, Berlin, 2008.

[14] A. Bar-Noy, M. Johnson, and O. Liu. Peak shaving through resource buffering. In *Approximation and Online Algorithms*, volume 5426, pages 147–159. Springer, Berlin, 2009.

[15] S. Batun, B. Denton, T. Huschka, and A. Schaefer. Operating room pooling and parallel surgery processing under uncertainty. *INFORMS Journal on Computing*, 23(2):220–237, 2011.

[16] A. Bhattacharya and J. Kharoufeh. Linear programming formulations for non-stationary, finite-horizon Markov decision process models. *Operations Research Letters*, 45(6):570–574, 2017.

[17] A. Bhattacharya, J. Kharoufeh, and B. Zeng. Managing energy storage in microgrids: A multistage stochastic programming approach. *To appear in IEEE Transactions on Smart Grid*, 2016. DOI: 10.1109/TSG.2016.2618621.

[18] W. Bian and X. Chen. Optimality and complexity for constrained optimization problems with nonconvex regularization. *To appear in Mathematics of Operations Research*, 2017. DOI: 10.1287/moor.2016.0837.

[19] W. Bian, X. Chen, and Y. Ye. Complexity analysis of interior point algorithms for non-Lipschitz and nonconvex minimization. *Mathematical Programming*, 149(1):301–327, 2015.

[20] J. Birge. Decomposition and partitioning methods for multistage stochastic linear programss. *Operations Research*, 33(5):989–1007, 1985.

[21] J. Birge and F. Louveaux. *Introduction to Stochastic Programming, Second Edition.* Springer, New York, 2011.

[22] E. Bitar, R. Rajagopal, P. Khargonekar, and K. Poolla. The role of co-located storage for wind power producers in conventional electricity markets. In *Proceedings of the American Control Conference*, pages 3886–3891, 2011.

[23] J. Blomvall and A. Shapiro. Solving multistage asset investment problems by the sample average approximation method. *Mathematical Programming*, 108(2):571–595, 2006.

[24] M. Bodur and J. Luedtke. Two-stage linear decision rules for multi-stage stochastic programming. Appeared in arXiv, URL: https://arxiv.org/abs/1701.04102v1, 2017.

[25] S. Boyd and L. Vandenberghe. *Convex Optimization.* Cambridge University Press, Cambridge, 2004.

[26] C. Brunetto and G. Tina. Optimal hydrogen storage sizing for wind power plants in day ahead electricity market. *IET Renewable Power Generation*, 1(4):220–226, 2007.

[27] R. Bühler. Integration of renewable energy sources using microgrids, virtual power plants and the energy hub approach. Technical report, Swiss Federal Institute of Technology (ETH) Zurich, Zurich, Switzerland, 2010.

[28] J. Casazza and F. Delea. *Understanding Electric Power Systems: An Overview of the Technology, the Marketplace, and Government Regulation, 2nd Edition.* John Wiley & Sons, Inc., Hoboken, NJ, 2010.

[29] A. Celik. Energy output estimation for small-scale wind power generators using weibull-representative wind data. *Journal of Wind Engineering and Industrial Aerodynamics*, 91(5):693–707, 2003.

[30] S. Cerisola, J. Latorre, and A. Ramos. Stochastic dual dynamic programming applied to nonconvex hydrothermal models. *European Journal of Operational Research*, 218(3):687–697, 2012.

[31] C. Chen, J. Wang, F. Qiu, and D. Zhao. Resilient distribution system by microgrids formation after natural disasters. *IEEE Transactions on Smart Grid*, 7(2):958–966, 2016.

[32] J. Chen and S. Burer. Globally solving nonconvex quadratic programming problems via completely positive programming. *Mathematical Programming Computation*, 4(1):33–52, 2012.

[33] Z. Chen and W. Powell. Convergent cutting-plane and partial-sampling algorithm for multistage stochastic linear programs with recourse. *Journal of Optimization Theory and Applications*, 102(3):497–524, 1999.

[34] J. Conti, P. Holtberg, J. Beamon, A. Schaal, G. Sweetnam, and A. Kydes. Annual Energy Outlook 2009: With Projections to 2030. Technical report, U.S. Energy Information Administration, 2009.

[35] F. Curtis, T. Mitchell, and M. Overton. A BFGS-SQP method for nonsmooth, nonconvex, constrained optimization and its evaluation using relative minimization profiles. *Optimization Methods and Software*, 32(1):148–181, 2017.

[36] A. Cutis and V. Gevorgian. Wind turbine generator system power quality test report for the Gaia wind 11-kW wind turbine. Technical report, National Renewable Energy Laboratory, 2011.

[37] G. Dantzig and G. Infanger. Multi-stage stochastic linear programs for portfolio optimization. *Annals of Operations Research*, 45(1):59–76, 1993.

[38] G. de Maere d'Aertrycke, A. Shapiro, and Y. Smeers. Risk exposure and Lagrange multipliers of nonanticipativity constraints in multistage stochastic problems. *Mathematical Methods of Operaations Research*, 77(3):393–405, 2013.

[39] V. de Matos, D. Morton, and E. Finardi. Assessing policy quality in a multistage stochastic program for long-term hydrothermal scheduling. *Annals of Operations Research*, 253(2):713–731, 2017.

[40] V. de Matos, A. Philpott, and E. Finardi. Improving the performance of stochastic dual dynamic programming. *Journal of Computational and Applied Mathematics*, 290:196–208, 2015.

[41] T. H. de Mello, V. de Matos, and E. Finardi. Sampling strategies and stopping criteria for stochastic dual dynamic programming: A case study in long-term hydrothermal scheduling. *Energy Systems*, 2(1):1–31, 2011.

[42] D. Donoho and I. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, 1994.

[43] S. E. Dreyfus. An analytic solution of the warehouse problem. *Management Science*, 4(1):99–104, 1957.

[44] B. Dunn, H. Kamath, and J. Tarascon. Electrical energy storage for the grid: A battery of choices. *Science*, 334(6058):928–935, 2011.

[45] J. Dupačová, N. Gröwe-Kuska, and W. Römisch. Scenario reduction in stochastic programming. *Mathematical Programming*, 95(3):493–511, 2003.

[46] J. Eto. Final report on the August 14, 2003 blackout in the United States and Canada: Causes and recommendations. Technical report, Washington, D.C., 2004.

[47] J. Eyer and G. Corey. Energy storage for the electricity grid: Benefits and market potential assessment guide. Technical report, Sandia National Laboratories, 2010.

[48] J. Eyer, J. Iannucci, and G. Corey. Energy storage benefits and market analysis handbook, a study for the DOE energy storage systems program. Technical report, Sandia National Labratories, 2004.

[49] J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001.

[50] W. E. Featheringill. Power transformer loading. *IEEE Transactions on Industry Applications*, IA-19(1):21–27, 1983.

[51] L. Frantzeskakis and W. Powell. A successive linear approximation procedure for stochastic, dynamic vehicle allocation problems. *Transportation Science*, 24(1):40–57, 1990.

[52] M. Freimer, J. Linderoth, and D. Thomas. The impact of sampling methods on bias and variance in stochastic linear programs. *Computational Optimization and Applications*, 51(1):51–75, 2012.

[53] J. García-González, R. de la Muela, L. Santos, and A. González. Stochastic joint optimization of wind generation and pumped-storage units in an electricity market. *IEEE Transactions on Power Systems*, 23(2):460–468, 2008.

[54] Y. Ghiassi-Farrokhfal, F. Kazhamiaka, C. Rosenberg, and S. Keshav. Optimal design of solar PV farms with storage. *IEEE Transactions on Sustainable Energy*, 6(4):1586–1593, 2015.

[55] F. Giannessi and E. Tomasin. Nonconvex quadratic programs, linear complementarity problems, and integer linear programs. In *5th Conference on Optimization Techniques Part I*, Lecture Notes in Computer Science, pages 437–449, Berlin, 1973. Springer.

[56] P. Girardeau, V. Leclere, and A. Philpott. On the convergence of decomposition methods for multistage stochastic convex programs. *Mathematics of Operations Research*, 40(1):130–145, 2015.

[57] J. Gonzalez, R. de la Muela, L. M. Santos, and A. González. Stochastic joint optimization of wind generation and pumped-storage units in an electricity market. *IEEE Transactions on Power Systems*, 23(2):460–468, 2008.

[58] N. Gould and D. Robinson. A second derivative SQP method: Global convergence. *SIAM Journal of Optimization*, 20(4):2023–2048, 2010.

[59] C. Gouveia, J. Moreira, C. L. Moreira, and J. A. P. Lopes. Coordinating storage and demand response for microgrid emergency operation. *IEEE Transactions on Smart Grid*, 4(4):1898–1908, 2013.

[60] S. Grijalva, M. Costley, and N. Ainsworth. Prosumer-based control architecture for the future electricity grid. In *IEEE International Conference on Control Applications (CCA)*, pages 43–48, 2011.

[61] S. Grijalva and M. Tariq. Prosumer-based smart grid architecture enables a flat, sustainable electricity industry. In *IEEE Innovative smart grid technologies (ISGT)*, pages 1–6, 2011.

[62] V. Guigues, M. Lejeune, and W. Tekaya. Regularized decomposition methods for deterministic and stochastic convex optimization and application to portfolio selection with direct transaction and market impact costs. Appeared in arXiv, URL: https://arxiv.org/abs/1701.03941v1, 2017.

[63] B. Hajek. Extremal splittings of point processes. *Mathematics of Operations Research*, 10(4):543–556, 1985.

[64] P. Harsha and M. Dahleh. Optimal management and sizing of energy storage under dynamic pricing for the efficient integration of renewable energy. *IEEE Transactions on Power Systems*, 30(3):1164–1181, 2015.

[65] N. Hatziargyriou, H. Asano, R. Iravani, and C. Marnay. Microgrids. *IEEE Power and Energy Magazine*, 5(4):78–94, 2007.

[66] H. Heitsch and W. Römisch. Scenario tree reduction for multistage stochastic programs. *Computational Management Science*, 6(2):117–133, 2009.

[67] Y. Herer, M. Tzur, and Yücesan. The multilocation transshipment problem. *IIE Transactions*, 38(3):185–200, 2006.

[68] J. Higle and S. Sen. Stochastic decomposition: Ain algorithm for two-stage linear programs with recourse. *Mathematics of Operations Research*, 16(3):650–669, 1991.

[69] C. Hill, M. Such, D. Chen, J. Gonzalez, and W. Grady. Battery energy storage for enabling integration of distributed solar power generation. *IEEE Transactions on Smart Grid*, 3(2):850–857, 2012.

[70] M. Hindsberger. ReSa: A method for solving multistage stochastic linear programs. *Journal of Applied Operational Research*, 6(1):2–15, 2014.

[71] A. Hoke, A. Brissette, S. Chandler, A. Pratt, and D. Maksimović. Look-ahead economic dispatch of microgrids with energy storage, using linear programming. In *1st IEEE Conference on Technologies for Sustainability (SusTech)*, pages 154–161, 2013.

[72] A. Hoke, A. Brissette, K. Smith, A. Pratt, and D. Maksimović. Accounting for lithium-ion battery degradation in electric vehicle charging optimization. *IEEE Journal of Emerging and Selected Topics in Power Electronics*, 2(3):691–700, 2014.

[73] R. Horst and H. Tuy. *Global optimization: Deterministic Approaches*. Springer, Berlin, 1996.

[74] W. Hu, Z. Chen, and B. Bak-Jensen. Optimal operation strategy of battery energy storage system to real-time electricity price in Denmark. In *Proceedings of the IEEE Power and Energy Society General Meeting*, pages 1–7, 2010.

[75] S. Huang, J. Xiao, J. Pekny, G. Reklaitis, and A. Liu. Quantifying system level benefits from distributed solar and energy storage. *Journal of Energy Engineering*, 138(2):33–42, 2011.

[76] Y. Huang, W. Hu, Y. Min, W. Zhang, W. Luo, Z. Wang, and W. Ge. Risk-constrained coordinative dispatching for battery energy storage systems of wind farms. In *IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)*, pages 1–6, 2013.

[77] G. Infanger and D. Morton. Cut sharing for multistage stochastic linear programs with interstage dependency. *Mathematical Programming*, 75(2):241–256, 1996.

[78] Y. Ji, J. Wang, S. Yan, W. Gao, and H. Li. Optimal microgrid energy management integrating intermittent renewable energy and stochastic load. In *IEEE Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pages 334–338, 2015.

[79] A. Joseph and M. Shahidehpour. Battery storage systems in electric power systems. In *IEEE Power Engineering Society General Meeting*, pages 8–16, 2006.

[80] J. Kaldellis and D. Zafirakis. Optimum energy storage techniques for the improvement of renewable energy sources-based electricity generation and economic efficiency. *Energy*, 32(12):2295–2305, 2007.

[81] A. Kaplan and R. Tichatschke. Proximal point methods and non-convex optimization. *Journal of Global Optimization*, 13(4):389–406, 1998.

[82] S. Kaplan. Power plants: Characteristics and costs. Technical Report RL34746, Congressional Research Service, 2008.

[83] J. Kelley. The cutting-plane method for solving convex programs. *Journal of the Society for Industrial and Applied Mathematics*, 8(4):703–712, 1960.

[84] J. Kim and W. Powell. Optimal energy commitments with storage and intermittent supply. *Operations Research*, 59(6):1347–1360, 2011.

[85] M. Korpass, A. Holen, and R. Hildrum. Operation and sizing of energy storage for wind power plants in a market system. *International Journal of Electrical Power and Energy Systems*, 25(8):599–606, 2003.

[86] I. Koutsopoulos, V. Hatzi, and L. Tassiulas. Optimal energy storage control policies for the smart power grid. In *IEEE International Conference on Smart Grid Communications*, pages 475–480, 2011.

[87] V. Kozmik. On variance reduction of mean-CVar Monte Carlo estimators. *Computational Management Science*, 12(2):221–242, 2014.

[88] P. Krokhmal, S. Uryasev, and J. Palmquist. Portfolio optimization with conditional value-at-risk objective and constraints. *Journal of Risk*, 4(2):43–68, 2002.

[89] L. Kuznia, B. Zeng, G. Centeno, and Z. Miao. Stochastic optimization for power system configuration with renewable energy in remote areas. *Annals of Operations Research*, 210(1):411–432, 2013.

[90] K. Lange. *Vector and Matrix Norms*, pages 77–91. Springer, New York, NY, 2010.

[91] R. Lasseter. Microgrids and distributed generation. *Journal of Energy Engineering*, 133(3):144–149, 2007.

[92] D. Lee, J. Kim, and R. Baldick. Stochastic optimal control of the storage system to limit ramp rates of wind power output. *IEEE Transactions on Smart Grid*, 4(4):2256–2265, 2013.

[93] Y. Lee and A. Sidford. Path finding methods for linear programming: Solving linear programs in $\tilde{\mathrm{O}}(\sqrt{\mathrm{rank}})$ iterations and faster algorithms for maximum flow. In *Proceedings of the 2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 424–433, 2014.

[94] C. Lemaréchal. An extension of davidon methods to non-differentiable problems. In M. Balinski and P. Wolfe, editors, *Nondifferentiable Optimization*, pages 95–109. Springer, Berlin, 1975.

[95] C. Lemaréchal, A. Nemirovskii, and Y. Nesterov. New variants of bundle methods. *Mathematical Programming*, 69(1):111–147, 1995.

[96] M. Lemmon, G. Venkataramanan, and P. Chapman. Using microgrids as a path towards smart grids. In *NSF Workshop on Cyber-Physical Energy Systems*, Baltimore, MD, 2009.

[97] Q. Li and P. Yu. Multimodularity and its applications in three stochastic dynamic inventory problems. *Manufacturing and Service Operations Management*, 16(3):455–463, 2014.

[98] M. Lijesen. The real-time price elasticity of electricity. *Energy Economics*, 29(2):249–258, 2007.

[99] J. Linderoth, A. Shapiro, and S. Wright. The empirical behavior of sampling methods for stochastic programming. *Annals of Operations Research*, 142(1):215–241, 2006.

[100] K. Linowsky and A. Philpott. On the convergence of sampling-based decomposition algorithms for multistage stochastic programs. *Journal of Optimization Theory and Applications*, 125(2):349–366, 2005.

[101] H. Liu, T. Yao, and R. Li. Global solutions to folded concave penalized nonconvex learning. *Annals of Statistics*, 44(2):629–659, 2016.

[102] L. Lu. Stand-alone wind power and hybrid solar-wind power. In *Handbook of Clean Energy Systems*, volume 1, chapter 30. John Wiley & Sons, Inc., Hoboken, NJ, 2015.

[103] V. Marano, S. Onori, Y. Guezennec, G. Rizzoni, and N. Madella. Lithium-ion batteries life estimation for plug-in hybrid electric vehicles. In *IEEE Vehicle Power and Propulsion Conference (VPCC)*, pages 536–543, 2009.

[104] G. McCormick. Computability of global solutions to factorable nonconvex programs: Part I – convex underestimating problems. *Mathematical Programming*, 10(1):147–175, 1976.

[105] S. Meyn, M. Negrete-Pincetic, G. Wang, A. Kowli, and E. Shafieepoorfard. The value of volatile resources in electricity markets. In *Proceedings of the 49th IEEE Conference on Decision and Control (CDC)*, pages 1029–1036. IEEE, 2010.

[106] P. Milgrom and C. Shannon. Monotone comparative statics. *Econometrica*, 62(1):157–180, 1994.

[107] P. Mokrian and M. Stephen. A stochastic programming framework for the valuation of electricity storage. In *Proceedings of the 26th USAEE/IAEE North American Conference*, pages 24–27. IAEE, 2006.

[108] J. Momoh. *Smart Grid: Fundamentals of Design and Analysis.* John Wiley & Sons, Inc., Hoboken, NJ, 2012.

[109] R. Moore. Global optimization to prescribed accuracy. *Computers and Mathematics with Applicationsn*, 21(6):25–39, 1991.

[110] B. Moradzadeh and K. Tomsovic. Two-stage residential energy management considering network operational constraints. *IEEE Transactions on Smart Grid*, 4(4):2339–2346, 2013.

[111] J. Morales, J. Nocedal, and Y. Wu. A sequential quadratic programming algorithm with an additional equality constrained phase. *IMA Journal of Numerical Analysis*, 32(2):553–579, 2012.

[112] K. Murota. Note on multimodularity and L-convexity. *Mathematics of Operations Research*, 30(3):658–661, 2005.

[113] R. Nelson. Power requirements for batteries in hybrid electric vehicles. *Journal of Power Sources*, 91(1):2–26, 2000.

[114] Y. Nesterov. Nonsmooth optimization. In *Introductory Lectures on Convex Optimization.* Springer, Boston, MA, 2004.

[115] H. Oh. Aggregation of buses for a network reduction. *IEEE Transactions on Power Systems*, 27(2):705–712, 2012.

[116] L. Pan and D. Politis. Bootstrap prediction intervals for linear, nonlinear and nonparametric autoregressions. *Journal of Statistical Planning and Inference*, 42(1):27–62, 2014.

[117] M. Pereira and L. Pinto. Multi-stage stochastic optimization applied to energy planning. *Mathematical Programming*, 52(1):359–375, 1991.

[118] A. Philpott and Z. Guan. On the convergence of stochastic dual dynamic programming and related methods. *Operations Research Letters*, 36(4):450–455, 2008.

[119] A. Philpott, F. Wahid, and F. Bonnans. MIDAS: A mixed integer dynamic approximation scheme. 2016. Appeared in Optimization Online, URL: http://www.optimization-online.org/DB_FILE/2016/05/5431.pdf.

[120] W. Powell. *Approximate Dynamic Programming: Solving the Curses of Dimensionality, 2nd Edition.* John Wiley & Sons, Inc., Hoboken, NJ, 2011.

[121] W. Powell and H. Topaloglu. Stochastic programming in transportation and logistics. In *Handbooks in Operations Research and Management Science*, volume 10, pages 555–635. Elsevier Science B.V., Amsterdam, 2003.

[122] M. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming.* John Wiley & Sons, Inc., Hoboken, NJ, 2009.

[123] T. Ralphs, Y. Shinano, T. Berthold, and T. Koch. Parallel solvers for mixed-integer linear optimization. In Y. Hamadi and L. Sais, editors, *Handbook of Parallel Constraint Reasoning*, chapter 14. Springer International, 2017.

[124] S. Rao and D. Tylavsky. Nonlinear network reduction for distribution networks using the holomorphic embedding method. In *North American Power Symposium (NAPS)*, pages 1–6, 2016.

[125] S. Resnick. *A Probability Path.* Birkhäuser, Boston, MA, 2005.

[126] R. Rockafellar and R. Wets. Scenarios and policy aggregation in optimization under uncertainty. *Mathematics of Operations Research*, 16(1):119–147, 1991.

[127] W. Rudin. *Principles of Mathematical Analysis.* International Series in Pure and Applied Mathematics. McGraw-Hill, New York, NY, 1964.

[128] A. Ruszczyński. A regularized decomposition method for minimizing a sum of polyhedral functions. *Mathematical Programming*, 35(3):309–333, 1986.

[129] A. Ruszczyński. Decomposition methods. In A. Ruszczyński and A. Shapiro, editors, *Handbooks in Operations Research and Management Science*, pages 144–211. Elsevier Science B.V., Amsterdam, 2003.

[130] A. Ruszczyński and A. Świętanowski. Accelerating the regularized decomposition method for two stage stochastic linear problems. *European Journal of Operational Research*, 101(2):328–342, 1997.

[131] N. Secomandi. Optimal commodity trading with a capacitated storage asset. *Management Science*, 56(3):449–467, 2010.

[132] N. Secomandi. Merchant commodity storage practice revisited. *Operations Research.*, 63(5):1131–1143, 2015.

[133] J. Seguro and T. Lambert. Modern estimation of the parameters of the weibull wind speed distribution for wind energy analysis. *Journal of Wind Engineering and Industrial Aerodynamics*, 85(1):75–84, 2000.

[134] S. Sen and Z. Zhou. Multistage stochastic decomposition: A bridge between stochastic programming and approximate dynamic programming. *SIAM Journal on Optimization*, 24(1):127–153, 2014.

[135] A. Shapiro. Inference of statistical bounds for multistage stochastic programming problems. *Mathematical Methods of Operations Research*, 58(1):57–68, 2003.

[136] A. Shapiro. Analysis of stochastic dual dynamic programming method. *European Journal of Operational Research*, 209(1):63–72, 2011.

[137] A. Shapiro, W. Tekaya, J. da Costa, and M. Soares. Report for technical cooperation between Georgia Institute of Technology and Operador Nacional do Sistema Elétrico. Technical report, 2011.

[138] A. Shapiro, W. Tekaya, J. da Costa, and M. Soares. Risk neutral and risk averse stochastic dual dynamic programming method. *European Journal of Operational Research*, 224(2):375–391, 2013.

[139] W. Shi, X. Xie, C. Chu, and R. Gadh. Distributed optimal energy management in microgrids. *IEEE Transactions on Smart Grid*, 6(3):1137–1146, 2015.

[140] D. Simchi-Levi, X. Chen, and J. Bramel. *Convexity and Supermodularity*, pages 15–44. Springer, New York, NY, 2014.

[141] R. Sioshansi, P. Denholm, T. Jenkin, and J. Weiss. Estimating the value of electricity storage in PJM: Arbitrage and some welfare effects. *Energy Economics*, 31(2):269–277, 2009.

[142] F. Sissine. Energy Independence and Security Act of 2007: A Summary of Major Provisions. Technical report, Defense Technical Information Center (DTIC), 2007.

[143] P. Sŭlc, S. Backhaus, and M. Chertkov. Optimal distributed control of reactive power via the alternating direction method of multipliers. *IEEE Transactions on Energy Conversion*, 29(4):968–977, 2014.

[144] S. Suryanarayanan, F. David, J. Mitra, and Y. Li. Achieving the smart grid through customer-driven microgrids supported by energy storage. In *Proceedings of the IEEE International Conference on Industrial Technology (ICIT)*, pages 884–890, 2010.

[145] S. Suryanarayanan and J. Mitra. Enabling technologies for the customer-driven microgrid. In *Proceedings of the IEEE Power and Energy Society General Meeting (PES)*, pages 1–3, 2009.

[146] S. Talari, M. Yazdaninejad, and M. Haghifam. Stochastic-based scheduling of the microgrid operation including wind turbines, photovoltaic cells, energy storages and responsive loads. *IET Generation, Transmission Distribution*, 9(12):1498–1509, 2015.

[147] S. Tan, J. Xu, and S. Panda. Optimization of distribution network incorporating distributed generators: An integrated approach. *IEEE Transactions on Power Systems*, 28(3):2421–2432, 2013.

[148] X. Tan, Q. Li, and H. Wang. Advances and trends of energy storage technology in microgrid. *Electrical Power and Energy Systems*, 44(1):179–191, 2013.

[149] L. Tang, W. Jiang, and G. Saharidis. An improved Benders decomposition algorithm for the logistics facility location problem with capacity expansions. *Annals of Operations Research*, 210(1):165–190, 2012.

[150] D. M. Topkis. *Supermodularity and Complemantarity.* Princeton University Press, Princeton, NJ, 1998.

[151] W. van Ackooij, W. de Oliviera, and Y. Song. On regularization with normal solutions in decomposition methods for multistage stochastic programming. Appeared in Optimization Online, URL: http://www.optimization-online.org/DB_FILE/2017/01/5806.pdf, 2017.

[152] P. Van de Ven, N. Hegde, L. Massoulie, and T. Salondis. Optimal control of residential energy storage under price fluctuations. In *Proceedings of the First International Conference on Smart Grids, Green Communications and IT Energy-Aware Technologies*, pages 159–162, 2011.

[153] P. Van de Ven, N. Hegde, L. Massoulie, and T. Salondis. Optimal control of end-user energy storage. *IEEE Transactions on Smart Grid*, 4(2):789–797, 2013.

[154] N. Wade, P. Taylor, P. Lang, and P. Jones. Evaluating the benefits of an electrical energy storage system in a future smart grid. *Energy Policy*, 38(11):7180–7188, 2010.

[155] R. Walling, R. Saint, R. Dugan, J. Burke, and L. Kojovic. Summary of distributed resources impact on power delivery systems. *IEEE Transactions on Power Delivery*, 23(3):1636–1644, 2008.

[156] Z. Wang, B. Chen, J. Wang, J. Kim, and M. Begovic. Robust optimization based optimal dg placement in microgrids. *IEEE Transactions on Smart Grid*, 5(5):2173–2182, 2014.

[157] Z. Wang and M. Lemmon. Stability analysis of weak rural electrification microgrids with droop-controlled rotational and electronic distributed generators. In *Proceedings of the IEEE Power Energy Society General Meeting*, pages 1–5, 2015.

[158] J. Ward. Equivalent circuits for power-flow studies. *Transactions of the American Institute of Electrical Engineers*, 68(1):373–382, 1949.

[159] R. Wiser and G. Barbose. Renewable portfolio standards in the United States: A status report with data through 2007. Technical report, Lawrence Berkeley National Laboratory, 2008.

[160] X. Xi, R. Sioshansi, and V. Marano. A stochastic dynamic programming model for co-optimization of distributed energy storage. *Energy Systems*, 5(3):475–505, 2014.

[161] P. Yang and A. Nehorai. Joint optimization of hybrid energy storage and generation capacity with renewable energy. *IEEE Transactions on Smart Grid*, 5(4):1566–1574, 2014.

[162] H. Yeh and S. Doan. Battery placement on performance of VAR controls. In *IEEE Green Energy and Systems Conference*, pages 1–6, 2013.

[163] W. Yuan, J. Wang, F. Qiu, C. Chen, C. Kang, and B. Zeng. Robust optimization-based resilient distribution network planning against natural disasters. *IEEE Transactions on Smart Grid*, PP(99):1–10, 2016.

[164] C. Zhang. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38(2):894–942, 2010.

[165] Y. Zhang, N. Gatsis, and G. B. Giannakis. Robust energy management for microgrids with high-penetration renewables. *IEEE Transactions on Sustainable Energy*, 4(4):944–953, 2013.

[166] Q. Zheng, J. Wang, and A. Liu. Stochastic optimization for unit commitment – A review. *IEEE Transactions on Power Systems*, 30(4):1913–1924, 2015.

[167] Y. Zhou, A. Scheller-Wolf, N. Secomandi, and S. Smith. Managing wind-based electricity generation in the presence of storage and transmission capacity. 2014. Appeared in SSRN, URL: https://ssrn.com/abstract=1962414.

[168] Z. Zhou, J. Zhang, P. Liu, Z. Li, M. C. Georgiadis, and E. N. Pistikopoulos. A two-stage stochastic programming model for the optimal design of distributed energy systems. *Applied Energy*, 103:135–144, 2013.

[169] Z. Zhu, S. Lu, B. Gao, T. Yi, and B. Chen. Life cycle cost analysis of three types of power lines in 10 kV distribution network. *Inventions*, 1(4):71–80, 2016.

[170] W. Zhuang and M. Li. Monotone optimal control for a class of Markov decision processes. *European Journal of Operations Research*, 217(2):342–350, 2012.

[171] P. H. Zipkin. *Foundations of Inventory Management*. McGraw-Hill, Boston, MA, 2000.

[172] P. H. Zipkin. On the structure of lost-sales inventory models. *Operations Research*, 56(4):937–944, 2008.

[173] J. Zou, S. Ahmed, and X. Sun. Stochastic dual dynamic integer programming. 2016. Appeared in Optimization Online, URL: http://www.optimization-online.org/DB_FILE/2016/05/5436.pdf.