Robust dynamic optimization: theory and applications

Saumya Sinha

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2018

Reading Committee:

Archis Ghate, Chair

Aleksandr Aravkin

Michael R. Wagner

Program Authorized to Offer Degree:
Applied Mathematics

University of Washington

Abstract

Robust dynamic optimization: theory and applications

Saumya Sinha

Chair of the Supervisory Committee:
Professor Archis Ghate
Industrial & Systems Engineering, and Applied Mathematics

Many applications in decision-making use a dynamic optimization framework to model a system evolving uncertainly in discrete time, and an agent who chooses actions/controls from a set of available choices in order to minimize a suitable cost function. An important aspect of model formulation is the choice of input parameters. These are traditionally estimated from historical data and prior domain knowledge, and treated as known quantities in the decision-making process. This approach ignores any estimation errors or misspecification in the problem data, leading to potentially suboptimal solutions. Robust optimization addresses this issue by treating the parameters themselves as unknown quantities, known only to lie within some set of plausible values called the 'uncertainty set'. The decision-maker then follows a conservative approach and minimizes a 'worst-case' cost over all possible values of the parameter. Problems of this nature are the subject of this dissertation.

The *first* chapter provides a background on infinite-horizon Markov decision processes (MDPs) and the Newsvendor model. MDPs are sequential decision-making problems with infinitely many decision epochs. At the end of every epoch, the next state of the system is prescribed via a transition probability depending on the current state and the action chosen. The robust formulation allows for these transition probabilities to be unknown, and the decision-maker minimizes the maximum expected total discounted cost. A detailed analytical treatment of robust MDPs with bounded immediate costs, along with robust versions of the

the standard solution methods of value iteration and policy iteration, is available in the literature. However, these methods cannot be implemented when the state-space is countable. Further, no theoretical framework is available for the case when costs are unbounded. These issues are addressed in Chapters 2 to 4. The Newsvendor model is a classical framework for inventory management over a finite horizon under demand ambiguity, and a robust formulation described in Chapter 5 circumvents the issue of assuming distributional information on this demand.

Robust nonstationary MDPs: In the *second* chapter, I consider an infinite-horizon robust MDP for which immediate costs are time-dependent but uniformly bounded, and the uncertainty sets vary with time. The state- and action-spaces are assumed to be finite. The optimal value function can be obtained from the robust Bellman equations [28], but the nonstationarity of the data results in an infinite system of equations to be solved. I provide a policy iteration algorithm which uses finite-dimensional approximations to policy evaluation and policy improvement, so that each step of the algorithm requires a finite amount of memory and computation, and as such, can be used in practice. These approximations are chosen adaptively to guarantee that the algorithm achieves sufficient improvement in each iteration, so that the values of the policies generated by the algorithm monotonically converge pointwise to the optimal. The policies converge subsequentially to an optimal policy.

Robust countable-state MDPs with bounded costs: In the *third* chapter, I generalize the above setup to solve robust stationary MDPs with countable state-spaces. Immediate costs as well as the uncertainty sets are time-invariant in this case. The costs are non-negative and bounded, and the action-spaces are finite. In this case as well, an as-is execution of the existing policy iteration method is not possible, owing to three main reasons. The first issue arises due to the countable nature of the state-space that necessitates the solution of an infinite system of equations, and is addressed via state-space truncation. The other two complications arise from the nonlinearity of the robust evaluation operator and the need for solving the so-called inner problems to arbitrary accuracy. These are addressed by successive approximation and a careful selection of uncertainty sets. Thus, I present an approximate policy iteration algorithm that can be used in practice. Value functions of the policies generated by the algorithm converge to the optimal, while the policies themselves converge subsequentially to an optimal policy. Robust MDPs with interval uncertainty sets, robust MDPs with bounded state-transitions, and a robust equipment replacement model are presented as examples where the algorithm can be implemented.

Robust countable-state MDPs with unbounded costs: The *third* chapter further widens the scope by allowing the immediate cost functions to be unbounded. A theoretical treatment of these MDPs is not available in the literature, and I develop such a framework here. Standard assumptions for unbounded-cost MDPs are generalized to the robust case. The robust Bellman operator is shown to be a J-step contraction mapping, which guarantees the existence of a unique solution to the robust Bellman equations. Optimality of the robust Bellman equations is also established.

A robust multi-period newsvendor model with inventory balance constraints: In the *fourth* chapter, I study a different approach to dynamic optimization by means of an application in inventory control. A seller managing the inventory of a single product over multiple periods must determine the optimal order quantity per period in the face of uncertain demand. This problem is solved via a newsvendor model, and the optimal solution is a function of the purchase, shortage and holding costs as well as the revenue earned per unit. Here, I formulate a robust multi-period newsvendor model to address the ambiguity in demand, and the seller maximizes his 'worst-case' total profit. Closed-form expressions for robust optimal order quantities are provided, and their relationship with various cost parameters is analyzed. Explicit optimal solutions to the inner-problems are obtained for a large class of uncertainty sets. Additionally, a numerical comparison of the robust model with a stochastic one is presented for benchmarking.

TABLE OF CONTENTS

Page

List of Figures iii

Chapter 1: Background 1

1.1 Markov decision processes 1

1.2 Policy iteration 2

1.3 Robust Markov decision processes 3

1.4 Newsvendor model 5

Chapter 2: Policy iteration for robust nonstationary Markov decision processes . . 7

2.1 Introduction 7

2.2 Problem setup and algorithm 8

2.3 Convergence results 13

Chapter 3: Approximate policy iteration for robust countable-state Markov decision processes with bounded costs 26

3.1 Introduction 26

3.2 Problem formulation 27

3.3 Challenges in designing policy iteration 29

3.4 Algorithm 33

3.5 Convergence results 36

3.6 Examples 68

3.7 Conclusion 75

Chapter 4: Robust countable-state Markov decision processes with unbounded costs 77

4.1 Introduction 77

4.2 Problem Setup 77

4.3 Theoretical Results 80

Chapter 5: A robust multi-period newsvendor model with inventory-balance constraints 93

5.1 Introduction 93

5.2 Model 96

5.3 Closed-form Solutions for an Arbitrary Uncertainty Set 102

5.4 Benchmarking with Computational Experiments 124

5.5 Conclusion 131

LIST OF FIGURES

Figure Number Page

3.1 A schematic representation of the interdependence of results in Section 3.5. . 39

5.1 An illustrative plot of $Q_j^* + I_0$ for a robust newsvendor model with seven periods under various relations between the parameters. 115

5.2 Numerical results for the case where the stochastic model assumes an incorrect demand distribution with correct moments. 129

5.3 Numerical results for the case where the stochastic model assumes the correct demand distribution with incorrect moments. 131

Chapter 1

BACKGROUND

1.1 Markov decision processes

Markov decision processes (MDPs) model a large class of sequential decision-making problems under uncertainty [35]. A system evolves in discrete time and the beginning of each time-period is called a decision epoch. The total number of epochs can be finite or infinite, and the corresponding MDPs are called finite-horizon and infinite-horizon, respectively. The latter model systems that do not have a predefined time of extinction. In this case, the epochs are indexed by $t \in \{0, 1, 2, \ldots\}$. At each epoch, the system occupies a state $s \in \mathcal{S}$, where $\mathcal{S}$ is the set of all possible states. A decision-maker observes the current state and chooses an action a from a set $\mathcal{A}$ of available choices. Once an action has been chosen, the system advances to a new state $s' \in \mathcal{S}$ according to a transition probability distribution $p(\cdot|s, a)$ that depends on the current state s and the action chosen a. This transition incurs a cost $c(s, a, s')$. Different cost criteria can be employed to measure performance, and the expected total discounted cost criterion is one of the most common. Here, the cost incurred in period t is discounted by a factor λ^t for some constant $\lambda \in (0, 1)$. A (Markovian deterministic stationary) policy is a rule that assigns an action to each possible state. The decision-maker's objective is to find a policy that minimizes the expected total discounted cost over the infinite horizon.

For example, in a queueing model, the state can be be the number of jobs waiting in a queue, and the action would be the number of jobs accepted for service. In an inventory management model, the state can be defined as the current inventory level and the action would be the order quantity. Other common applications include scheduling, medical treatment planning, and transportation [16].

Cost is minimized by solving Bellman's equations of dynamic programming

$$v^*(s) = \inf_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} p(s'|s, \pi^k(s))[c(s, \pi^k(s), s') + \lambda v^*(s')], \quad s \in \mathcal{S}. \tag{1.1}$$

Here, $v^* : \mathcal{S} \to \mathbb{R}$ is the optimal value function. An optimal policy, if it exists, is constructed by choosing an action from the argmin set in (1.1) for each state. Therefore, solving an MDP amounts to solving the system of equations in (1.1), and this is done primarily by three methods – policy iteration, value iteration, and linear programming. In this dissertation, we restrict our attention to policy iteration.

1.2 Policy iteration

Policy iteration is standard method for solving MDPs [26]. The algorithm starts out with an arbitrary initial policy π^1. Then, in very iteration $k = 1, 2, \ldots$, the following steps are executed.

1. **Policy evaluation:** This step computes $v^{\pi^k}(s)$, the expected total discounted cost incurred under policy π^k when the system is initially in state s. The function $v^{\pi^k} : \mathcal{S} \to \mathbb{R}$ is called the value function of π^k. We have

$$v^{\pi^k}(s) = \sum_{s' \in \mathcal{S}} p(s'|s, \pi^k(s))[c(s, \pi^k(s), s') + \lambda v^{\pi^k}(s')], \quad s \in \mathcal{S}. \tag{1.2}$$

2. **Policy improvement:** Once the value of the current policy has been computed, a new policy is constructed as follows.

   (a) For each state-action pair (s, a), compute

$$\gamma^{\pi^k}(s, a) = \sum_{s' \in \mathcal{S}} p(s'|s, a(s))[c(s, a(s), s') + \lambda v^{\pi^k}(s')] - v^{\pi^k}(s). \tag{1.3}$$

   The term $\gamma^{\pi^k}(s, a)$ determines the improvement in total discounted cost if action

a is chosen in state s (instead of that prescribed by π^k). Therefore, $\gamma(s, \pi^k(s)) = 0$ for all s. A negative value of $\gamma(s, a)$ indicates that action a is a better choice in state s since it leads to a reduction in overall cost.

(b) Maximum improvement – The algorithm then identifies a state-action pair (s^k, a^k) which gives the largest reduction in overall cost, by choosing

$$(s^k, a^k) = \underset{s\in\mathcal{S},\ a\in\mathcal{A}}{\operatorname{argmin}}\ \gamma^{\pi^k}(s, a). \tag{1.4}$$

(c) Policy update – If $\gamma^{\pi^k}(s^k, a^k) \geq 0$, it implies that the total cost cannot be improved any further, and the current policy must be optimal.

Otherwise, a new policy π^{k+1} is constructed by choosing action a^k in state s^k and keeping all other actions the same. That is, $\pi^{k+1}(s^k) = a^k$ and $\pi^{k+1}(s) = \pi^k(s)$ for all $s \neq s^k$.

The above steps are repeated in every iteration until an optimal solution has been found. When the state- and action-spaces are finite, this algorithm discovers an optimal policy in a finite number of iterations (see Theorem 6.4.2 in [35]). The general policy iteration algorithm allows for actions across multiple states to be updated in each iteration. The method described above is the so-called ‘simple’ version of policy iteration where an action in only one state is updated in each iteration. Simple policy iteration is easier to implement and has been proven to exhibit strongly polynomial complexity [46].

1.3 Robust Markov decision processes

In the traditional study of MDPs as described above, the state-transition probabilities are treated as model parameters known *a priori* to the decision-maker. In practice, however, statistical estimates of these probabilities are obtained from historical data and used as a proxy for the true distributions. The resulting estimation errors are not accounted for, which may lead to potentially suboptimal solutions. Robust MDPs try to address this limitation by assuming that the transition probabilities are ambiguous and known only to lie in an "uncertainty set" of plausible distributions. In particular, for each state-action pair (s, a), there is a set $\mathcal{P}_s^a$ of probability distributions over $\mathcal{S}$ such that $p(\cdot|s, a) \in \mathcal{P}_s^a$. The objective then is to minimize the worst-case expected total discounted cost over all transition probabilities from these uncertainty sets. When the immediate costs $c(s, a, s')$ are assumed to be uniformly bounded over all states s, s' and actions a, it has been shown in [28] and nilim that the optimal value function v^* is the solution of a robust counterpart of Bellman's equations. That is,

$$v^*(s) = \sup_{p(\cdot|s,a)\in\mathcal{P}_s^a} \sum_{s'\in\mathcal{S}} p(s'|s, \pi^k(s))[c(s, \pi^k(s), s') + \lambda v^*(s')], \quad s \in \mathcal{S}.$$

Variants of policy iteration and value iteration for solving robust MDPs have been described in [28], where it is also noted that the linear programming formulation does not have a natural robust extension. Once again, we will focus on policy iteration.

Robust policy iteration: The main idea behind policy iteration remains the same in the robust variant. The algorithm starts with an arbitrary initial guess for an optimal policy. In the k-th iteration, $k = 1, 2, \ldots$, the worst-case value function for the current policy is computed via the equation

$$v^{\pi^k}(s) = \sup_{p\in\mathcal{P}_s^{\pi^k(s)}} \sum_{s'\in\mathcal{S}} p(s'|s, \pi^k(s))[c(s, \pi^k(s), s') + \lambda v^{\pi^k}(s')], \quad s \in \mathcal{S}.$$

Similarly, the most improving state-action pair (s^k, a^k) is determined by a robust counterpart of Equations (1.3) and (1.4), as defined below.

$$(s^k, a^k) = \operatorname*{argmin}_{s\in\mathcal{S},a\in\mathcal{A}} \sup_{p\in\mathcal{P}_s^{\pi^k(s)}} \sum_{s'\in\mathcal{S}} p(s'|s, a(s))[c(s, a(s), s') + \lambda v^{\pi^k}(s')] - v^{\pi^k}(s).$$

Then, if the current policy is found to be suboptimal, it is updated in state s^k as before, and the algorithm proceeds to the next iteration. Once again, [28] establishes that the algorithm finds an optimal policy in a finite number of steps if $\mathcal{S}$ and $\mathcal{A}$ are finite. Convergence in the countable-state case, however, has not been established. Moreover, the theoretical framework does not apply when the immediate costs are unbounded. These issues are addressed in Chapters 2-4 of this dissertation.

1.4 Newsvendor model

Chapter 5 explores a different approach to dynamic optimization by way of an application in inventory management. Consider a seller managing the inventory of a single product. He must decide how much of a product to order in every period $j = 1, 2, \ldots, n$ over some (finite) horizon. Let $d_j \geq 0$ be the demand for this product in the j-th period, and let $q_j \geq 0$ be the amount of new product that the seller purchases in the same period. If the demand in any period exceeds the current inventory level, it is backlogged and the seller seeks to satisfy it in a future period, but this incurs a shortage cost s per unit. On the other hand, any surplus inventory can be utilized later as well, but the seller pays a holding cost of h per unit. As such, the decision in any period must account for its future consequences. If $c \geq 0$ and $r \geq 0$ are the purchase cost and sale revenue per unit respectively, the profit function is given by

$$\Pi(q, d) = \underbrace{r \min\left\{\sum_{j=1}^{n} q_j, \sum_{j=1}^{n} d_j\right\}}_{\text{Total revenue}} - \underbrace{\left(\sum_{j=1}^{n} c q_j + \max\{hI_j, -sI_j\}\right)}_{\text{Total cost}}.$$

Here, $I_j = \sum_{i=1}^{j} q_i - d_i$ is the inventory at the end of period j, assuming that there is no initial inventory. The seller seeks to find optimal order quantities in order to maximize his total profit over the n periods. This is achieved by solving the following linear program.

$$\begin{aligned} \min_{q\geq 0} \quad & \sum_{j=1}^{n} y_j \\ \text{s.t.} \quad & y_j \geq (h+\delta_{jn}c)\sum_{i=1}^{j}(q_i-d_i),\ j=1,\dots,n, \\ & y_j \geq (s+\delta_{jn}(r-c))\sum_{i=1}^{j}(d_i-q_i)\ j=1,\dots,n, \end{aligned}$$

where δ_{jn} is the kronecker delta which takes the value 1 when $j=n$, and 0 otherwise. This is easy to solve if the demand is known, which is unrealistic in practice. The true demand is almost never known *a priori*. Trasditionally, this is resolved by assuming that the demand is a random variables with known distribution. This gives rise to the classical stochastic newsvendor model, and optimal order quantities are recovered through stochastic optimization techniques. As before, this approach is prone to suboptimality due to estimation errors; robust optimization addresses this limitation. A robust newsvendor model for worst-case profit maximization appears in Chapter 5.

Chapter 2

POLICY ITERATION FOR ROBUST NONSTATIONARY MARKOV DECISION PROCESSES

2.1 Introduction

Nonstationary MDPs[1] are a generalization of stationary MDPs, where the problem data are no longer assumed to be time-invariant [17, 19, 25]. An asymptotically convergent simple policy iteration algorithm for "nominal" (i.e., non-robust) MDPs was developed recently in [22]. That paper also analyzed in detail a close connection between this simple policy iteration and an infinite-dimensional simplex method. In this chapter, we develop a solution method for robust nonstationary MDPs.

The classic policy iteration algorithm was extended to the robust case in [28]. For finite-state, finite-action, stationary MDPs, it discovers a robust optimal policy in a finite number of iterations. This result was proven in [28] by invoking Theorem 6.4.2 from [35]. In fact, the policy iteration algorithm in [28] was presented for robust *countable-state* stationary MDPs. Hence, it is, in principle, applicable to *nonstationary* MDPs because, as shown in [22], nonstationary MDPs can be viewed as a special case of countable-state stationary MDPs by appending the states with a time-index. An "as is" execution of this algorithm, however, is not possible for countable-state or for nonstationary MDPs because it would call for infinite computations in both the policy evaluation and policy improvement steps of every iteration. Specifically, an implementable and provably convergent version of policy iteration is currently not available for robust nonstationary MDPs. We develop such an algorithm in this paper.

[1] Most MDPs discussed in this chapter are finite-state, finite-action and infinite-horizon; we therefore omit such qualifiers for brevity throughout, unless they are essential for clarity.

The key idea in our approach is that it proposes finitely implementable approximations of policy evaluation and simple policy improvement with steepest descent. These approximations are designed adaptively such that the resulting sequence of policies has monotonically decreasing costs. Moreover, the cost-improvement in consecutive iterations is large enough to guarantee convergence to optimality (see [22] for a counterexample of a nonstationary MDP where simply guaranteeing a cost-improvement in each iteration is not enough for convergence to optimality). These statements are made precise in the next two sections. We focus on the simple version of policy iteration to keep notation at a minimum, but our algorithm and proof of convergence can be generalized to a full version without technical difficulty. The only change needed in this full version is that instead of choosing a single period-state pair for updating an action, we select each pair that provides a sufficient improvement.

2.2 Problem setup and algorithm

Consider a nonstationary MDP with decision epochs $n = 1, 2, \ldots$. At the beginning of each period n, the system occupies a state $s \in \mathcal{S}$, where $\mathcal{S} = \{1, 2, \ldots, S\}$ is a finite set. A decision-maker observes this state and chooses an action $a \in \mathcal{A}$, where $\mathcal{A} = \{1, 2, \ldots, A\}$ is also a finite set. Given that action a was chosen in state s in period n, the system makes a transition to state s' at the beginning of period $n+1$ with probability $p_n(s'|s, a)$, incurring a nonnegative and bounded cost $0 \leq c_n(s, a, s') \leq c$ for some bound c. This process continues ad infinitum, starting the first period in some initial state $s_1 \in \mathcal{S}$. A (deterministic Markovian) policy π is a mapping that prescribes actions $\pi_n(s)$ in states $s \in \mathcal{S}$ in periods $n \in \mathbb{N}$. The decision-maker's objective is to find a policy that simultaneously (for all $s \in \mathcal{S}$ and all $n \in \mathbb{N}$) minimizes the infinite-horizon discounted expected cost incurred on starting period n in state s. The single-period discount factor is denoted by $0 \leq \lambda < 1$. We note, as an aside, that it is not possible in general to finitely describe the input data needed to completely specify a nonstationary MDP. It is therefore standard in the literature to assume the existence of a "forecast oracle" that, when queried by supplying a positive integer m, returns the cost and probability data for the first m periods. We work in this paper with nonstationary MDPs defined in this manner and refer the reader to [17, 20, 22] for detailed discussions of this issue. Following the language of robust optimization, we will call the problem described in this paragraph a *nominal* nonstationary MDP.

In the above nominal MDP, the transition probabilities $p_n(s'|s,a)$ are assumed to be known. *Robust* nonstationary MDPs account for estimation errors in these transition probabilities by instead assuming that for each state-action pair (s,a) in period n, the (conditional) probability mass function (pmf) $p_n(\cdot|s,a)$ of the next state is only known to lie in some nonempty compact set $\mathcal{P}^a_{n,s}$. This set is called the uncertainty set and it is a subset of the probability simplex $\mathcal{M}(\mathcal{S}) = \{q \in \mathbb{R}^S_+ \mid q_1 + \ldots + q_S = 1\}$. Specifically, robust nonstationary MPDs pursue an adversarial modeling approach where the adversary, also often called "nature", observes the state s in period n as well as the action a chosen there by the decision-maker and then selects a pmf $p_n(\cdot|s,a)$ from the uncertainty set $\mathcal{P}^a_{n,s}$. As per the standard "rectangularity assumption", nature's pmf selection in n, s, a is assumed to be independent of the history of previously visited states and actions and also of the actions chosen in other states (see [28, 33]). The decision-maker's objective is to find a policy that simultaneously (for all $s \in \mathcal{S}$ and all $n \in \mathbb{N}$) minimizes the "worst-case" (with respect to all possible adversarial choices) infinite-horizon discounted expected cost incurred on starting period n in state s.

This finite-state, finite-action robust nonstationary MDP can be equivalently viewed as a robust *stationary* MDP with the *countable* state-space $\mathcal{S} \times \mathbb{N}$ by appending states s with the time-index n. Let $v^*_n(s)$ denote the decision-maker's minimum worst-case cost, against all adversarial policies, on starting period $n \in \mathbb{N}$ in state $s \in \mathcal{S}$. The functions $v^*_n : \mathcal{S} \to \mathbb{R}_+$ are called robust optimal cost-to-go functions, and according to the theory of robust countable-state stationary MDPs from [28], they are unique solutions of the Bellman's equations

$$v^*_n(s) = \min_{a \in \mathcal{A}} \Bigg\{ \underbrace{\max_{p_n(\cdot|s,a) \in \mathcal{P}^a_{n,s}} \Bigg(\sum_{s' \in \mathcal{S}} p_n(s'|s,a) \left[c_n(s,a,s') + \lambda v^*_{n+1}(s') \right] \Bigg)}_{\text{inner problem}} \Bigg\}, \tag{2.1}$$

for $s \in \mathcal{S}$ and $n \in \mathbb{N}$. Actions that achieve the outer minima in the above equations define a robust optimal policy. Similarly, the infinite-horizon expected discounted cost incurred by implementing a policy π starting in state s in period n is denoted by $v_n^{\pi}(s)$. These costs-to-go are characterized by the infinite system of equations

$$v_n^{\pi}(s) = \max_{p_n(\cdot|s,\pi_n(s)) \in \mathcal{P}_{n,s}^{\pi_n(s)}} \left(\sum_{s' \in \mathcal{S}} p_n(s'|s,\pi_n(s)) \left[c_n(s,\pi_n(s),s') + \lambda v_{n+1}^{\pi}(s') \right] \right), \; s \in \mathcal{S}, \; n \in \mathbb{N}. \tag{2.2}$$

For the robust nonstationary MDP described above, an “as is” execution of robust policy iteration from [28] would roughly amount to the following algorithm. Start with an initial policy π^1. In iteration $k \geq 1$, solve the infinite system of equations in (2.2) to obtain the cost-to-go function v^{π^k} of policy π^k. This is the policy evaluation step. Then, update policy π^k to a new policy π^{k+1} that prescribes an action from the set

$$\operatorname*{argmin}_{a \in \mathcal{A}} \left\{ \max_{p_n(\cdot|s,a) \in \mathcal{P}_{n,s}^{a}} \left(\sum_{s' \in \mathcal{S}} p_n(s'|s,a) \left[c_n(s,a,s') + \lambda v_{n+1}^{\pi^k}(s') \right] \right) \right\} \tag{2.3}$$

in each state $s \in \mathcal{S}$ in each period $n \in \mathbb{N}$. This is the policy improvement step. Unfortunately, both these steps require infinite computations, rendering this algorithm unimplementable.

We remedy the above situation by proposing approximate implementations of policy evaluation and simple policy improvement. Specifically, in the policy evaluation step of the kth iteration, the cost-to-go function of policy π^k is approximated by the cost-to-go function of an $m(k)$-horizon truncation of that policy. In the simple policy improvement step of the kth iteration, an action is updated in state $s(k)$ in period $n(k)$ somewhere in the first $m(k)$-periods via the steepest descent rule applied to this cost-to-go function approximation. In order to guarantee that all actual infinite-horizon costs $v_n^{\pi^{k+1}}(s)$ of the resulting new policy π^{k+1} improve upon the actual infinite-horizon costs $v_n^{\pi^k}(s)$ of the old policy π^k, the truncation-length $m(k)$ is chosen adaptively via an iterative procedure such that the corresponding steepest improvement in the $m(k)$-horizon cost-approximations is large enough. In fact, the discussion in [22] and a counterexample in [23] show that even in the context of nominal nonstationary MDPs, it is not enough (for value convergence to optimality) to simply ensure that π^{k+1} improves upon π^k; it is essential to guarantee that the improvement is sufficiently large. As we shall see in Section 2.3, our choice of $m(k)$ also carefully handles this delicate issue. The details of this procedure are listed in Algorithm 1 below.

Note that although policy π^k in the kth iteration of this algorithm is "infinite-dimensional", it is described finitely because (i) π^1 is chosen such that it has a finite representation, and (ii) only a single component is changed in each iteration. Consequently, π^k can be stored on a computer. In addition, we emphasize that each iteration of this algorithm performs only a finite amount of computations. We also make the minor observation that the value of m is initiated at $n(k-1)$ in Step 2(a) of our algorithm, whereas m was initiated at 1 in the simple policy iteration algorithm for nominal nonstationary MDPs in [22]. This initial value of $m = 1$ was inefficient (in the sense that it called for unnecessary additional computations) because $m(k)$ is bounded below by $n(k-1)$ in their nominal case as well as in our robust case. This holds because the steepest descent action in the kth iteration cannot be found for a horizon m shorter than $n(k-1)$ as policies π^{k-1} and π^k prescribe identical actions in the first $n(k-1)-1$ periods.

We prove in the next section that the sequence of costs $v_n^{\pi^k}(s)$ corresponding to the policies π^k produced by this algorithm monotonically converges pointwise to the optimal costs $v_n^*(s)$ as $k \to \infty$. We also establish subsequential convergence of the corresponding policies π^k to an optimal policy. The main ideas in our algorithm and proofs are similar to the aforementioned recent work on simple policy iteration for nominal nonstationary MDPs [22]; the details are modified to accommodate our robust counterpart. For instance, the proofs in [22] for the nominal case thoroughly exploited the close connection between simple policy iteration and an infinite-dimensional simplex algorithm with the steepest descent pivoting rule. We cannot pursue that approach here because robust MDPs do not have an equivalent LP formulation (see [28]).

Algorithm 1 Simple policy iteration for robust nonstationary MDPs.

```
1: Initialize: Set iteration counter $k = 1$. Arbitrarily fix the initial policy $\pi^1$ to one that prescribes the first action in $\mathcal{A}$ in every state in every period. Let $n(0) = 1$.
2: for iterations $k = 1, 2, 3, \dots,$ do
     (a) Set $m = n(k-1)$. Let $m(k) = \infty$ and $\gamma^{k,\infty} = 0$.
     Approximate policy evaluation:
     (b) Compute the $m$-horizon approximation $v^{k,m}$ of the cost-to-go function $v^{\pi^k}$ as
         $v^{k,m}_{m+1}(s) = 0, \ \forall s \in \mathcal{S}, \qquad (2.4)$
         $v^{k,m}_{n}(s) = \max_{p_n(\cdot|s,\pi^k_n(s)) \in \mathcal{P}^{\pi^k_n(s)}_{n,s}} \left( \sum_{s' \in \mathcal{S}} p_n(s'|s,\pi^k_n(s)) \left[ c_n(s,\pi^k_n(s),s') + \lambda v^{k,m}_{n+1}(s') \right] \right), \ \forall s \in \mathcal{S}, \ n \le m. \qquad (2.5)$
     Approximate simple policy improvement:
     (c) Compute the approximate $Q$-function
         $Q^{k,m}_n(s,a) = \max_{p_n(\cdot|s,a) \in \mathcal{P}^a_{n,s}} \left( \sum_{s' \in \mathcal{S}} p_n(s'|s,a) \left[ c_n(s,a,s') + \lambda v^{k,m}_{n+1}(s') \right] \right), \ s \in \mathcal{S}, \ a \in \mathcal{A}, \ n \le m. \qquad (2.6)$
     (d) Compute $\gamma^{k,m}_n(s,a) = \lambda^{n-1}\big(Q^{k,m}_n(s,a) - v^{k,m}_n(s)\big)$, for $s \in \mathcal{S}$, $a \in \mathcal{A}$, and $n \le m$. Then calculate the amount of steepest descent
         $\gamma^{k,m} = \min_{\substack{s \in \mathcal{S} \\ a \in \mathcal{A}, a \neq \pi^k_n(s) \\ 1 \le n \le m}} \gamma^{k,m}_n(s,a). \qquad (2.7)$
     (e) If $\gamma^{k,m} < -\lambda^m \frac{c}{1-\lambda}$, set $m(k) = m$, let $(n(k), s(k), a(k))$ be an argmin in (2.7), and update $\pi^k$ to $\pi^{k+1}$ by replacing $\pi^k_{n(k)}(s(k))$ with $a(k)$; else set $m = m + 1$ and go to Step 2(b) above.
3: end for
```

2.3 Convergence results

Our two main convergence results in this paper appear toward the end of this section in Theorems 2.3.7 and 2.3.8. The proofs of these two theorems utilize several lemmas that we prove next.

The lemma below establishes a simple, fundamental property of Bellman's equations.

Lemma 2.3.1. *Suppose policy π is not optimal. Then there exist a state $s \in \mathcal{S}$, an action $a \in \mathcal{A}$, and a period $n \in \mathbb{N}$ such that*

$$Q_n^\pi(s,a) = \max_{p_n(\cdot|s,a)\in\mathcal{P}_{n,s}^a} \left(\sum_{s'\in\mathcal{S}} p_n(s'|s,a) \left[c_n(s,a,s') + \lambda v_{n+1}^\pi(s') \right] \right) < v_n^\pi(s). \tag{2.8}$$

Proof. Suppose not. Then, for each $n \in \mathbb{N}$ and each $s \in \mathcal{S}$, we have,

$$\max_{p_n(\cdot|s,a)\in\mathcal{P}_{n,s}^a} \left(\sum_{s'\in\mathcal{S}} p_n(s'|s,a) \left[c_n(s,a,s') + \lambda v_{n+1}^\pi(s') \right] \right) \geq v_n^\pi(s), \ \forall a \in \mathcal{A}.$$

Consequently, for each $n \in \mathbb{N}$ and each $s \in \mathcal{S}$, we obtain,

$$\begin{aligned} v_n^\pi(s) &= \max_{p_n(\cdot|s,\pi_n(s))\in\mathcal{P}_{n,s}^{\pi_n(s)}} \left(\sum_{s'\in\mathcal{S}} p_n(s'|s,\pi_n(s)) \left[c_n(s,\pi_n(s),s') + \lambda v_{n+1}^\pi(s') \right] \right) \\ &\geq \min_{a\in\mathcal{A}} \left\{ \max_{p_n(\cdot|s,a)\in\mathcal{P}_{n,s}^a} \left(\sum_{s'\in\mathcal{S}} p_n(s'|s,a) \left[c_n(s,a,s') + \lambda v_{n+1}^\pi(s') \right] \right) \right\} \geq v_n^\pi(s). \end{aligned}$$

This shows that, for each $n \in \mathbb{N}$ and each $s \in \mathcal{S}$,

$$v_n^\pi(s) = \min_{a\in\mathcal{A}} \left\{ \max_{p_n(\cdot|s,a)\in\mathcal{P}_{n,s}^a} \left(\sum_{s'\in\mathcal{S}} p_n(s'|s,a) \left[c_n(s,a,s') + \lambda v_{n+1}^\pi(s') \right] \right) \right\}.$$

This shows that the cost-to-go functions v_n^π satisfy Bellman's equations. Then π must be optimal. This is a contradiction. □

Within each iteration, our algorithm computes an m-horizon approximation $v^{k,m}$ to the true infinite-horizon cost-to-go function v^{π^k} of the policy π^k. The lemma below provides bounds for the quality of this approximation.

Lemma 2.3.2. *The approximation $v^{k,m}$ of v^{π^k} in Step 2(b) of Algorithm 1 satisfies*

$$v_n^{k,m}(s) \leq v_n^{\pi^k}(s) \leq v_n^{k,m}(s) + \lambda^{m+1-n}\frac{c}{1-\lambda}, \ \forall s \in \mathcal{S}, \ n = 1, 2, \ldots, m+1. \tag{2.9}$$

Proof. We prove the claim by backward induction on $n = m+1, m, \ldots, 1$.

Since the costs are nonnegative and bounded above by c, we know that v^{π^k} satisfies $0 \leq v_{m+1}^{\pi^k}(s) \leq c/(1-\lambda)$. Also, $v_{m+1}^{k,m}(s) = 0$ for all $s \in \mathcal{S}$ by (2.4). So, for $n = m+1$, we trivially have,

$$v_{m+1}^{k,m}(s) \leq v_{m+1}^{\pi^k}(s) \leq v_{m+1}^{k,m}(s) + \lambda^{m+1-n}\frac{c}{1-\lambda}, \ \forall s \in \mathcal{S}.$$

Now, assume, as the inductive hypothesis, that the claim is true for $n+1$. That is,

$$v_{n+1}^{k,m}(s') \leq v_{n+1}^{\pi^k}(s') \leq v_{n+1}^{k,m}(s') + \lambda^{m-n}\frac{c}{1-\lambda}, \ \forall s' \in \mathcal{S}.$$

After multiplying each term by λ and then adding $c_n(s, a, s')$ to all terms, this implies that

$$c_n(s,a,s') + \lambda v_{n+1}^{k,m}(s') \leq c_n(s,a,s') + \lambda v_{n+1}^{\pi^k}(s') \leq c_n(s,a,s') + \lambda v_{n+1}^{k,m}(s') + \lambda^{m+1-n}\frac{c}{1-\lambda},$$

for all $s, s' \in \mathcal{S}$ and $a \in \mathcal{A}$. Consequently, for the specific actions $\pi_n^k(s)$ prescribed by the policy π^k, we have,

$$c_n(s,\pi_n^k(s),s') + \lambda v_{n+1}^{k,m}(s') \leq c_n(s,\pi_n^k(s),s') + \lambda v_{n+1}^{\pi^k}(s') \leq c_n(s,\pi_n^k(s),s') + \lambda v_{n+1}^{k,m}(s') + \lambda^{m+1-n}\frac{c}{1-\lambda},$$

for all $s, s' \in \mathcal{S}$. Now, for a fixed $s \in \mathcal{S}$, consider any pmf $p_n(\cdot|s, \pi_n^k(s)) \in \mathcal{P}_{n,s}^{\pi_n^k(s)}$. By multiplying the above inequalities with this pmf and then adding over all $s' \in \mathcal{S}$, we obtain,

$$\sum_{s' \in \mathcal{S}} p_n(s'|s, \pi_n^k(s)) \left[c_n(s, \pi_n^k(s), s') + \lambda v_{n+1}^{k,m}(s')\right] \leq \sum_{s' \in \mathcal{S}} p_n(s'|s, \pi_n^k(s)) \left[c_n(s, \pi_n^k(s), s') + \lambda v_{n+1}^{\pi^k}(s')\right]$$
$$\leq \sum_{s' \in \mathcal{S}} p_n(s'|s, \pi_n^k(s)) \left[c_n(s, \pi_n^k(s), s') + \lambda v_{n+1}^{k,m}(s')\right] + \lambda^{m+1-n} \frac{c}{1-\lambda}.$$

Then, by taking the maximum of each side of these inequalities over all such pmfs in $\mathcal{P}_{n,s}^{\pi_n^k(s)}$, we obtain,

$$\max_{p_n(\cdot|s,\pi_n^k(s)) \in \mathcal{P}_{n,s}^{\pi_n^k(s)}} \left(\sum_{s' \in \mathcal{S}} p_n(s'|s, \pi_n^k(s)) \left[c_n(s, \pi_n^k(s), s') + \lambda v_{n+1}^{k,m}(s')\right] \right)$$
$$\leq \max_{p_n(\cdot|s,\pi_n^k(s)) \in \mathcal{P}_{n,s}^{\pi_n^k(s)}} \left(\sum_{s' \in \mathcal{S}} p_n(s'|s, \pi_n^k(s)) \left[c_n(s, \pi_n^k(s), s') + \lambda v_{n+1}^{\pi^k}(s')\right] \right)$$
$$\leq \max_{p_n(\cdot|s,\pi_n^k(s)) \in \mathcal{P}_{n,s}^{\pi_n^k(s)}} \left(\sum_{s' \in \mathcal{S}} p_n(s'|s, \pi_n^k(s)) \left[c_n(s, \pi_n^k(s), s') + \lambda v_{n+1}^{k,m}(s')\right] \right) + \lambda^{m+1-n} \frac{c}{1-\lambda}.$$

These maxima preserve the order of the earlier inequalities because of the following property: if one function is everywhere smaller than another function, then the maximum of the first function is smaller than the maximum of the second function provided that the maxima are taken over identical sets. Then, by (2.2) and (2.5), we have,

$$v_n^{k,m}(s) \leq v_n^{\pi^k}(s) \leq v_n^{k,m}(s) + \lambda^{m+1-n} \frac{c}{1-\lambda}, \ \forall s \in \mathcal{S}.$$

This restores the inductive hypothesis and completes the proof by induction. □

Step 2 of the algorithm iteratively increases the value of the approximating horizon's length m until a horizon that guarantees a large enough improvement is discovered. The algorithm thus runs the risk of being caught in an infinite loop, wherein it never discovers such a horizon. The next lemma tackles this issue.

Lemma 2.3.3. *Step 2 of Algorithm 1 terminates at a finite value of m if and only if policy π^k is not optimal.*

Proof. Suppose π^k is not optimal. Then, by Lemma 2.3.1, there exist a period $n \in \mathbb{N}$, $s \in \mathcal{S}$, and an action $a \in \mathcal{A}$ such that $Q_n^{\pi^k}(s,a) < v_n^{\pi^k}(s)$. Thus, let $\epsilon = v_n^{\pi^k}(s) - Q_n^{\pi^k}(s,a) > 0$. Then, by Lemma 2.3.2, we have,

$$\begin{aligned}
\epsilon = v_n^{\pi^k}(s) - Q_n^{\pi^k}(s,a) &= v_n^{\pi^k}(s) - \max_{p_n(\cdot|s,a)\in\mathcal{P}^a_{n,s}} \left(\sum_{s'\in\mathcal{S}} p_n(s'|s,a) \left[c_n(s,a,s') + \lambda v_{n+1}^{\pi^k}(s') \right] \right) \\
&\leq v_n^{k,m}(s) + \lambda^{m+1-n}\frac{c}{1-\lambda} - \max_{p_n(\cdot|s,a)\in\mathcal{P}^a_{n,s}} \left(\sum_{s'\in\mathcal{S}} p_n(s'|s,a) \left[c_n(s,a,s') + \lambda v_{n+1}^{k,m}(s') \right] \right) \\
&= v_n^{k,m}(s) + \lambda^{m+1-n}\frac{c}{1-\lambda} - Q_n^{k,m}(s,a) = \lambda^{1-n}\left(\lambda^m \frac{c}{1-\lambda} - \lambda^{n-1}\big(Q_n^{k,m}(s,a) - v_n^{k,m}(s)\big) \right) \\
&\leq \lambda^{1-n}\left(\lambda^m \frac{c}{1-\lambda} - \gamma^{k,m} \right).
\end{aligned}$$

where the last inequality holds by the definition of $\gamma^{k,m}$ in Step 2(d) of the algorithm. This inequality yields $\lambda^{1-n}\gamma^{k,m} \leq \lambda^{m+1-n}\frac{c}{1-\lambda} - \epsilon$. For a sufficiently large m, we have that $\lambda^{m+1-n}\frac{c}{1-\lambda} < \epsilon/2$ because $0 \leq \lambda < 1$. Therefore, for any such large m, we have, $\lambda^{m+1-n}\frac{c}{1-\lambda} - \epsilon < -\epsilon/2 < -\lambda^{m+1-n}\frac{c}{1-\lambda}$. Thus, we obtain, $\lambda^{1-n}\gamma^{k,m} < -\lambda^{m+1-n}\frac{c}{1-\lambda}$, that is, $\gamma^{k,m} < -\lambda^m\frac{c}{1-\lambda}$. Thus, if the policy π^k is not optimal, there exists a large enough m for which the stopping condition in Step 2(e) of the algorithm is satisfied, and Step 2 terminates finitely.

Conversely, suppose policy π^k is optimal, and Step 2 of the algorithm terminates for some $m(k)$. Then, $\gamma^{k,m(k)} + \lambda^{m(k)}\frac{c}{1-\lambda} < 0$. That is, $\gamma_{n(k)}^{k,m(k)}(s(k),a(k)) + \lambda^{m(k)}\frac{c}{1-\lambda} < 0$, where $(n(k),s(k),a(k))$ is the argmin in (2.7) with $a(k) \neq \pi^k_{n(k)}(s(k))$. That is, $\lambda^{1-n(k)}\gamma_{n(k)}^{k,m(k)}(s(k),a(k)) + \lambda^{m(k)+1-n(k)}\frac{c}{1-\lambda} < 0$. Then, by the definition of $\gamma_{n(k)}^{k,m(k)}(s(k),a(k))$, this implies that $Q_{n(k)}^{k,m(k)}(s(k),a(k)) - v_{n(k)}^{k,m(k)}(s(k)) + \lambda^{m(k)+1-n(k)}\frac{c}{1-\lambda} < 0$. Then, by using the definition of $Q_{n(k)}^{k,m(k)}(s(k),a(k))$ as in (2.6), we get,

$$\begin{aligned}
\max_{p_{n(k)}(\cdot|s(k),a(k))\in\mathcal{P}^{a(k)}_{n(k),s(k)}} \left(\sum_{s'\in\mathcal{S}} p_{n(k)}(s'|s(k),a(k)) \left[c_n(s(k),a(k),s') + \lambda v_{n(k)+1}^{k,m(k)}(s') \right] \right) - \\
v_{n(k)}^{k,m(k)}(s(k)) + \lambda^{m(k)+1-n(k)}\frac{c}{1-\lambda} < 0.
\end{aligned}$$

By applying Lemma 2.3.2, the above strict inequality implies that $Q_{n(k)}^{\pi^k}(s(k),a(k)) < v_{n(k)}^{\pi^k}(s(k))$.

In other words, the optimal cost-to-go function v^{π^k} violates Bellman's equation (2.1) in state $s(k)$ in period $n(k)$. But this contradicts the optimality of π^k. This completes the proof of the lemma. $\square$

The lemma implies that if π^k is optimal, then Step 2 of the algorithm never terminates. One subtlety here is that the algorithm therefore cannot tell that it has discovered an optimal policy. As explained in more detail in [22], this, however, is not a limitation of our algorithm. Rather, it is rooted in a fundamental property of nonstationary sequential decision problems that optimality cannot be verified, in general, with finite computation.

The next lemma shows that, despite the approximations they employ, our policy evaluation and simple policy improvement steps produce a sequence of policies with nonincreasing cost-to-go functions.

Lemma 2.3.4. *Suppose policy π^k is not optimal. Then $v_n^{\pi^{k+1}}(s) \le v_n^{\pi^k}(s)$ for all periods $n \in \mathbb{N}$ and all states $s \in \mathcal{S}$, with this inequality being strict when $n = n(k)$ and $s = s(k)$. Furthermore, $v_{n(k)}^{\pi^{k+1}}(s(k)) - v_{n(k)}^{\pi^k}(s(k)) \le \lambda^{1-n(k)}\left(\lambda^{m(k)}\frac{c}{1-\lambda} + \gamma^{k,m(k)}\right)$.*

Proof. Since policy π^k is not optimal, Step 2 of the algorithm terminates at some $m(k)$ by Lemma 2.3.3. Also, policies π^{k+1} and π^k differ only in the actions they prescribe in period $n(k) \le m(k)$ in state $s(k)$. Consequently, $v_n^{\pi^{k+1}}(s) = v_n^{\pi^k}(s)$ for all $s \in \mathcal{S}$ and all $n > n(k)$. Similarly, $v_{n(k)}^{\pi^{k+1}}(s) = v_{n(k)}^{\pi^k}(s)$ for all $s(k) \neq s \in \mathcal{S}$. Moreover, (2.2) implies that

$$\begin{aligned}
sv^{\pi^{k+1}}_{n(k)}(s(k)) &= \max_{p_n(\cdot|s(k),a(k))\in\mathcal{P}^{a(k)}_{n(k),s(k)}} \Bigg(\sum_{s'\in\mathcal{S}} p_{n(k)}(s'|s(k),a(k)) \Big[c_{n(k)}(s(k),a(k),s') + \lambda v^{\pi^{k+1}}_{n(k)+1}(s')\Big]\Bigg) \\
&= \max_{p_n(\cdot|s(k),a(k))\in\mathcal{P}^{a(k)}_{n(k),s(k)}} \Bigg(\sum_{s'\in\mathcal{S}} p_{n(k)}(s'|s(k),a(k)) \Big[c_{n(k)}(s(k),a(k),s') + \lambda v^{\pi^{k}}_{n(k)+1}(s')\Big]\Bigg) \\
&\le \max_{p_n(\cdot|s(k),a(k))\in\mathcal{P}^{a(k)}_{n(k),s(k)}} \Bigg(\sum_{s'\in\mathcal{S}} p_{n(k)}(s'|s(k),a(k)) \Bigg[c_{n(k)}(s(k),a(k),s') + \\
&\lambda\Bigg(v^{k,m(k)}_{n(k)+1}(s') + \lambda^{m(k)-n(k)}\frac{c}{1-\lambda}\Bigg)\Bigg]\Bigg) \\
&= \max_{p_n(\cdot|s(k),a(k))\in\mathcal{P}^{a(k)}_{n(k),s(k)}} \Bigg(\sum_{s'\in\mathcal{S}} p_{n(k)}(s'|s(k),a(k)) \Bigg[c_{n(k)}(s(k),a(k),s') + \lambda v^{k,m(k)}_{n(k)+1}(s')\Bigg]\Bigg) + \\
&\lambda^{m(k)+1-n(k)}\frac{c}{1-\lambda}, \\
&= Q^{k,m(k)}_{n(k)}(s(k),a(k)) + \lambda^{m(k)+1-n(k)}\frac{c}{1-\lambda} \\
&= \lambda^{1-n(k)}\gamma^{k,m(k)} + v^{k,m(k)}_{n(k)}(s(k)) + \lambda^{m(k)+1-n(k)}\frac{c}{1-\lambda} \\
&\le \lambda^{1-n(k)}\gamma^{k,m(k)} + v^{\pi^k}_{n(k)}(s(k)) + \lambda^{m(k)+1-n(k)}\frac{c}{1-\lambda} < v^{\pi^k}_{n(k)}(s(k)).
\end{aligned}$$

Here, the first inequality follows by Lemma 2.3.2, the penultimate equality holds by the definition in (2.6) of $Q^{k,m(k)}_{n(k)}(s(k),a(k))$, the last equality holds by the definition of $\gamma^{k,m(k)}$, the penultimate inequality holds by Lemma 2.3.2, and finally, the strict inequality follows by the stopping condition in Step 2(e). In summary, we have shown that $v^{\pi^{k+1}}_{n(k)}(s(k)) < v^{\pi^k}_{n(k)}(s(k))$ and that $v^{\pi^{k+1}}_{n(k)}(s(k)) - v^{\pi^k}_{n(k)}(s(k)) < \lambda^{1-n(k)}\Bigg(\lambda^{m(k)}\frac{c}{1-\lambda} + \gamma^{k,m(k)}\Bigg)$. Now we complete the rest of the proof by induction on $n = n(k), n(k)-1, \ldots, 1$. To start off this induction process, we note that the argument thus far has established that $v^{\pi^{k+1}}_n(s) \le v^{\pi^k}_n(s)$, for all $s \in \mathcal{S}$ and $n = n(k)$. Now, as the inductive hypothesis, suppose that $v^{\pi^{k+1}}_n(s) \le v^{\pi^k}_n(s)$, for all $s \in \mathcal{S}$

and some $n \leq n(k)$. Then, from (2.2), we have, for each $s \in \mathcal{S}$, that

$$\begin{aligned}
v^{\pi^{k+1}}_{n-1}(s) &= \max_{p_{n-1}(\cdot|s,\pi^{k+1}_{n-1}(s))\in\mathcal{P}^{\pi^{k+1}_{n-1}(s)}_{n-1,s}} \left(\sum_{s'\in\mathcal{S}} p_{n-1}(s'|s,\pi^{k+1}_{n-1}(s)) \left[c_{n-1}(s,\pi^{k+1}_{n-1}(s),s') + \lambda v^{\pi^{k+1}}_{n}(s') \right] \right) \\
&= \max_{p_{n-1}(\cdot|s,\pi^{k}_{n-1}(s))\in\mathcal{P}^{\pi^{k}_{n-1}(s)}_{n-1,s}} \left(\sum_{s'\in\mathcal{S}} p_{n-1}(s'|s,\pi^{k}_{n-1}(s)) \left[c_{n-1}(s,\pi^{k}_{n-1}(s),s') + \lambda v^{\pi^{k+1}}_{n}(s') \right] \right) \\
&\leq \max_{p_{n-1}(\cdot|s,\pi^{k}_{n-1}(s))\in\mathcal{P}^{\pi^{k}_{n-1}(s)}_{n-1,s}} \left(\sum_{s'\in\mathcal{S}} p_{n-1}(s'|s,\pi^{k}_{n-1}(s)) \left[c_{n-1}(s,\pi^{k}_{n-1}(s),s') + \lambda v^{\pi^{k}}_{n}(s') \right] \right) \\
&= v^{\pi^k}_{n-1}(s).
\end{aligned}$$

Here, the second equality holds because π^{k+1} and π^k prescribe identical actions in all states in period $n-1$, the inequality holds by the inductive hypothesis, and the last equality follows by the definition of $v^{\pi^k}_{n-1}(s)$. This restores the inductive hypothesis and completes our proof. □

Lemma 2.3.5. *We have that* $\left[\lambda^{m(k)}\frac{c}{1-\lambda} + \gamma^{k,m(k)}\right] \to 0$ *as* $k \to \infty$.

Proof. Observe that if π^k is optimal for any k, then, by Lemma 2.3.3, Step 2 of the algorithm does not terminate finitely; hence $\left[\lambda^{m(k)}\frac{c}{1-\lambda} + \gamma^{k,m(k)}\right] = 0$ because the algorithm is initiated with $m(k) = \infty$ and $\gamma^{k,\infty} = 0$ and hence the claim holds. Now suppose that π^k is not optimal for any k. The algorithm thus produces a sequence of solutions v^{π^k}. Now define, for each k, $f^k = \sum\limits_{n\in\mathbb{N}} \sum\limits_{s\in\mathcal{S}} \lambda^{n-1} v^{\pi^k}_n(s)$ and let $\delta^k = f^{k+1} - f^k$. Then, since the sum f^k is finite for all k, we have,

$$\begin{aligned}
\delta^k &= \sum_{n\in\mathbb{N}} \sum_{s\in\mathcal{S}} \lambda^{n-1} \left[v^{\pi^{k+1}}_n(s) - v^{\pi^k}_n(s) \right] \leq \lambda^{n(k)-1} \left[v^{\pi^{k+1}}_{n(k)}(s(k)) - v^{\pi^k}_{n(k)}(s(k)) \right] \\
&\leq \gamma^{k,m(k)} + \lambda^{m(k)} \frac{c}{1-\lambda} < 0,
\end{aligned}$$

where the first inequality follows since every term in the sum is non-positive, the second inequality holds by the second claim in Lemma 2.3.4, and the last inequality follows from the stopping condition in Step 2(e) of the algorithm. That is, f^k is a nonnegative decreasing sequence of real numbers, hence it converges. This implies that $\delta^k \to 0$ as $k \to \infty$. Since

$\delta^k \leq \gamma^{k,m(k)} + \lambda^{m(k)} \frac{c}{1-\lambda} < 0$ for all k, the second claim holds. □

The sequence of approximating horizons $m(k)$ in not monotonically increasing in k. The next lemma shows that it nevertheless diverges to infinity as $k \to \infty$. This also implies that the amount of steepest descent improvement converges to zero.

Lemma 2.3.6. *The sequence $m(k) \to \infty$ as $k \to \infty$. Also, $\gamma^{k,m(k)} \to 0$ as $k \to \infty$.*

Proof. Identical to the proof of Lemma 5.7 in [22] hence omitted. □

Theorem 2.3.7 (Value Convergence)**.** *The sequence of cost-to-go functions produced by Algorithm 1 converges pointwise to the optimal cost-to-go function. That is,*

$$\lim_{k\to\infty} v_n^{\pi^k}(s) = v_n^*(s) \quad \textit{for all} \quad n \in \mathbb{N}, s \in \mathcal{S}. \tag{2.10}$$

Proof. Policies for the nonstationary MDP lie in the strategy space $\Phi = \prod_{n=1}^{\infty} \mathcal{A}^S \subset \prod_{n=1}^{\infty} \mathbb{R}^S$, which is compact in the metrizable product topology by Tychonoff's product theorem (see Theorem 2.61 on page 52 of [1]). In fact, $\rho(\cdot, \cdot)$ defined by

$$\rho(\pi, \tilde{\pi}) = \sum_{n=1}^{\infty} \frac{1}{2^n} \left(\frac{d(\pi_n, \tilde{\pi}_n)}{1 + d(\pi_n, \tilde{\pi}_n)} \right)$$

where $d(\cdot, \cdot)$ is the Euclidean metric on $\mathbb{R}^S$, is an example of a metric which induces the product topology on $\prod_{n=1}^{\infty} \mathbb{R}^S$ (see Theorem 3.36 on page 89 of [1]). Further, the cost-to-go functions lie in the set $V = \left\{ v \in \prod_{n=1}^{\infty} \mathbb{R}^S : 0 \leq v_n(s) \leq \frac{c}{1-\lambda}, n \in \mathbb{N}, s \in \mathcal{S} \right\}$. Again, by Tychonoff's theorem, $V \subset \prod_{n=1}^{\infty} \mathbb{R}^S$ is compact in the metrizable product topology. Since Φ is compact, the sequence of policies π^k has a convergent subsequence π^{k_i}. Let $\bar{\pi}$ be the limit of this sequence. By the same reasoning, the corresponding sequence of cost-to-go functions $v^{\pi^{k_i}}$ has a convergent subsequence, $v^{\pi^{k_{i_j}}}$, whose limit is, say, $\bar{v}$.

We first show that $\bar{v} = v^{\bar{\pi}}$, that is, $\bar{v}$ is the cost-to-go function corresponding to the policy $\bar{\pi}$.

Consider any $n \in \mathbb{N}$ and $s \in \mathcal{S}$. Then, for any j,

$$v_n^{\pi^{k_{i_j}}}(s) - \max_{p(\cdot|s,\pi_n^{k_{i_j}}(s)) \in \mathcal{P}_{n,s}^{\pi_n^{k_{i_j}}(s)}} \sum_{s' \in \mathcal{S}} p(s'|s, \pi_n^{k_{i_j}}(s)) \left[c_n(s, \pi_n^{k_{i_j}}(s), s') + \lambda v_{n+1}^{k_{i_j}}(s') \right] = 0.$$

Now, since $\pi^{k_{i_j}} \to \bar{\pi}$ in the product topology, we have that $\pi_n^{k_{i_j}}(s) \to \bar{\pi}_n(s)$ as $j \to \infty$. Since $\mathcal{A}$ is a finite set, this implies that there exists a number $J(n,s)$ such that for all $j \geq J(n,s)$, $\pi_n^{k_{i_j}}(s) = \bar{\pi}_n(s)$. Hence, for all $j \geq J(n,s)$, the sets $\mathcal{P}_{n,s}^{\pi_n^{k_{i_j}}(s)}$ and $\mathcal{P}_{n,s}^{\bar{\pi}_n(s)}$ are identical, and we have,

$$v_n^{\pi^{k_{i_j}}}(s) - \max_{p(\cdot|s,\bar{\pi}_n(s)) \in \mathcal{P}_{n,s}^{\bar{\pi}_n(s)}} \sum_{s' \in \mathcal{S}} p(s'|s, \bar{\pi}_n(s)) \left[c_n(s, \bar{\pi}_n(s), s') + \lambda v_{n+1}^{k_{i_j}}(s') \right] = 0. \tag{2.11}$$

For each fixed $p(\cdot|s, \bar{\pi}_n(s)) \in \mathcal{P}_{n,s}^{\bar{\pi}_n(s)}$, we have,

$$v_n^{\pi^{k_{i_j}}}(s) - \sum_{s' \in \mathcal{S}} p(s'|s, \bar{\pi}_n(s)) \left[c_n(s, \bar{\pi}_n(s), s') + \lambda v_{n+1}^{k_{i_j}}(s') \right] \geq 0.$$

Taking limits as $j \to \infty$, this yields,

$$\bar{v}_n(s) - \sum_{s' \in \mathcal{S}} p(s'|s, \bar{\pi}_n(s)) \left[c_n(s, \bar{\pi}_n(s), s') + \lambda \bar{v}_{n+1}(s') \right] \geq 0,$$

for all $p(\cdot|s, \bar{\pi}_n(s)) \in \mathcal{P}_{n,s}^{\bar{\pi}_n(s)}$. This implies

$$\bar{v}_n(s) - \max_{p(\cdot|s,\bar{\pi}_n(s)) \in \mathcal{P}_{n,s}^{\bar{\pi}_n(s)}} \sum_{s' \in \mathcal{S}} p(s'|s, \bar{\pi}_n(s)) \left[c_n(s, \bar{\pi}_n(s), s') + \lambda \bar{v}_{n+1}(s') \right] \geq 0. \tag{2.12}$$

Now, we show that inequality (2.12) cannot be strict. For each $j \geq J(n,s)$, let $p^{k_{i_j}}(\cdot|s, \bar{\pi}_n(s))$ be an argmax in (2.11). Then, we rewrite (2.11) as

$$v_n^{\pi^{k_{i_j}}}(s) - \sum_{s' \in \mathcal{S}} p^{k_{i_j}}(s'|s, \bar{\pi}_n(s)) \left[c_n(s, \bar{\pi}_n(s), s') + \lambda v_{n+1}^{k_{i_j}}(s') \right] = 0.$$

Note that as $\mathcal{P}_{n,s}^{\bar{\pi}_n(s)}$ is a compact subset of $\mathbb{R}^S$, the sequence $\{p^{k_{i_j}}(\cdot|s, \bar{\pi}_n(s)),\ j \geq J(n,s)\}$ has a convergent subsequence $p^{k_{i_{j_l}}}$. Let $\bar{p}(\cdot|s, \bar{\pi}_n(s))$ be the limit of this subsequence. For each l, we have,

$$v_n^{k_{i_{j_l}}}(s) - \sum_{s' \in \mathcal{S}} p^{k_{i_{j_l}}}(s'|s, \bar{\pi}_n(s)) \left[c_n(s, \bar{\pi}_n(s), s') + \lambda v_{n+1}^{k_{i_{j_l}}}(s')\right] = 0.$$

Taking limits as $l \to \infty$, this gives us

$$\bar{v}_n(s) - \sum_{s' \in \mathcal{S}} \bar{p}(s'|s, \bar{\pi}_n(s)) \left[c_n(s, \bar{\pi}_n(s), s') + \lambda \bar{v}_{n+1}(s')\right] = 0.$$

Hence, the inequality in (2.12) must be an equality, and we have

$$\bar{v}_n(s) - \max_{p(\cdot|s,\bar{\pi}_n(s)) \in \mathcal{P}_{n,s}^{\bar{\pi}_n(s)}} \sum_{s' \in \mathcal{S}} p(s'|s, \bar{\pi}_n(s)) \left[c_n(s, \bar{\pi}_n(s), s') + \lambda \bar{v}_{n+1}(s')\right] = 0. \tag{2.13}$$

Since the above is true for all (n, s), we have proved that the limiting cost-to-go function $\bar{v}$ is the evaluation of the limiting policy $\bar{\pi}$, and we denote it by $v^{\bar{\pi}}$.

We now show, by contradiction, that the limiting policy $\bar{\pi}$ must be optimal. Suppose $\bar{\pi}$ is not optimal. Then, by Lemma 2.3.1, there exists a period n, a state s and an action a such that

$$\begin{aligned} 0 < \epsilon &= v_n^{\bar{\pi}}(s) - Q_n^{\bar{\pi}}(s,a) \\ &= v_n^{\bar{\pi}}(s) - \max_{p_n(\cdot|s,a) \in \mathcal{P}_{n,s}^a} \left(\sum_{s' \in \mathcal{S}} p_n(s'|s,a) \left[c_n(s,a,s') + \lambda v_{n+1}^{\bar{\pi}}(s')\right] \right). \end{aligned} \tag{2.14}$$

For any j, let $p_n^{k_{i_j}}(\cdot|s,a)$ be the argmax in

$$Q_n^{\pi^{k_{i_j}}}(s,a) = \max_{p_n(\cdot|s,a) \in \mathcal{P}_{n,s}^a} \left(\sum_{s' \in \mathcal{S}} p_n(s'|s,a) \left[c_n(s,a,s') + \lambda v_{n+1}^{\pi^{k_{i_j}}}(s')\right] \right).$$

As before, the sequence $p_n^{k_{i_j}}(\cdot|s,a)$ has a convergent subsequence $p_n^{k_{i_{j_l}}}(\cdot|s,a)$. Let $\bar{p}_n(\cdot|s,a)$

be the limit of this subsequence. Then, we have,

$$\begin{aligned}\lim_{l\to\infty}\left(v_n^{\pi^{k_{i_{j_l}}}}(s) - Q_n^{\pi^{k_{i_{j_l}}}}(s,a)\right) &= \lim_{l\to\infty}\left(v_n^{k_{i_{j_l}}}(s) - \sum_{s'\in\mathcal{S}} p_n^{k_{i_{j_l}}}(s'|s,a)\left[c_n(s,a,s') + \lambda v_{n+1}^{\pi^{k_{i_{j_l}}}}(s')\right]\right)\\ &= v_n^{\bar{\pi}}(s) - \sum_{s'\in\mathcal{S}} \bar{p}_n(s'|s,a)\left[c_n(s,a,s') + \lambda v_{n+1}^{\bar{\pi}}(s')\right]\\ &\geq v_n^{\bar{\pi}}(s) - \max_{p_n(\cdot|s,a)\in\mathcal{P}_{n,s}^a}\left(\sum_{s'\in\mathcal{S}} p_n(s'|s,a)\left[c_n(s,a,s') + \lambda v_{n+1}^{\bar{\pi}}(s')\right]\right)\\ &= \epsilon.\end{aligned}$$

Then, there exists an integer L such that for $l \geq L$,

$$\epsilon/2 < v_n^{\pi^{k_{i_{j_l}}}}(s) - Q_n^{\pi^{k_{i_{j_l}}}}(s,a) = v_n^{\pi^{k_{i_{j_l}}}}(s) - \max_{p_n(\cdot|s,a)\in\mathcal{P}_{n,s}^a}\left(\sum_{s'\in\mathcal{S}} p_n^{k_{i_{j_l}}}(s'|s,a)\left[c_n(s,a,s') + \lambda v_{n+1}^{\pi^{k_{i_{j_l}}}}(s')\right]\right).$$

Since $m(k) \to \infty$, we have that for large enough l, $m(k_{i_{j_l}}) \geq n$. Then, applying Lemma 2.3.2 gives us that

$$\begin{aligned}\epsilon/2 &< v_n^{k_{i_{j_l}}, m(k_{i_{j_l}})}(s) + \lambda^{m(k_{i_{j_l}})+1-n}\frac{c}{1-\lambda}\\ &\quad - \max_{p_n(\cdot|s,a)\in\mathcal{P}_{n,s}^a}\left(\sum_{s'\in\mathcal{S}} p_n^{k_{i_{j_l}}}(s'|s,a)\left[c_n(s,a,s') + \lambda v_{n+1}^{k_{i_{j_l}}, m(k_{i_{j_l}})}(s')\right]\right)\\ &\leq \lambda^{m(k_{i_{j_l}})+1-n}\frac{c}{1-\lambda} - \lambda^{1-n}\gamma^{k_{i_{j_l}}, m(k_{i_{j_l}})} = \lambda^{1-n}\left(\lambda^{m(k_{i_{j_l}})}\frac{c}{1-\lambda} - \gamma^{k_{i_{j_l}}, m(k_{i_{j_l}})}\right).\end{aligned}$$

But this contradicts the fact from Lemma 2.3.6 that both $\lambda^{m(k_{i_{j_l}})}\frac{c}{1-\lambda}$ and $\gamma^{k_{i_{j_l}}, m(k_{i_{j_l}})}$ converge to zero as $l \to \infty$. Hence, our assumption is false and the limiting policy $\bar{\pi}$ must be optimal.

We remark that so far we have only proven that v^{π^k} converges subsequentially to the optimal value function v^*. But from Lemma 2.3.4, we know that each component $v_n^{\pi^k}(s)$ is a nonincreasing sequence of nonnegative real numbers, and therefore must converge. This combined with the subsequential convergence proves that $\lim_{k\to\infty} v_n^{\pi^k}(s) = v_n^*(s)$ for all $s \in \mathcal{S}$ and $n \in \mathbb{N}$. □

Theorem 2.3.8 (Policy Convergence)**.** *For any $\epsilon > 0$, there exists an iteration counter k_ϵ such that $\rho(\pi^k, \pi^{*k}) < \epsilon$ for some optimal policy π^{*k}, for all $k \geq k_\epsilon$. In fact, if the MDP has a unique optimal policy π^*, then $\lim\limits_{k\to\infty} \pi^k = \pi^*$. Further, for every period n, there exists an iteration counter K_n such that for all $k \geq K_n$, actions $\pi^k_m(s)$ are optimal for the robust non-stationary MDP in all states $s \in \mathcal{S}$ and all periods $m \leq n$.*

Proof. We prove the first claim by contradiction. Suppose this is not true. Then, there exists an $\epsilon > 0$ and a subsequence π^{k_i} of π^k such that $\rho(\pi^{k_i}, \pi) > \epsilon$ for all optimal policies π, for all $i \in \mathbb{N}$. Since the space of all policies is compact, the sequence π^{k_i} has a convergent subsequence $\pi^{k_{i_j}}$, whose limit is, say, $\bar{\pi}$. Then, there exists an integer J such that $\rho(\pi^{k_{i_j}}, \bar{\pi}) < \epsilon$ for all $j \geq J$. Further, as in the proof of Theorem 2.3.7, $\bar{\pi}$ must be an optimal policy. This leads to a contradiction. Hence, the first claim is true.

Further, suppose that π^* is the unique optimal policy. Then, as shown above, for every $\epsilon > 0$, there exists an integer k_ϵ, such that $\rho(\pi^k, \pi^*) < \epsilon$ for all $k \geq k_\epsilon$. This implies that $\lim\limits_{k\to\infty} \pi^k = \pi^*$.

Now, for the third claim, we note that the result is trivially true if π^k is optimal for some k. When this is not the case, we first claim that given $\epsilon > 0$ and any period n, there exists an iteration counter K_n such that for all $k \geq K_n$, $|\pi^k_m(s) - \pi^{k*}_m(s)| < \epsilon$, for all $m \leq n$ and for all $s \in \mathcal{S}$, for some optimal policy π^{k*}. Suppose not. Then, there exists a subsequence k_i, and for each i, a period $m_i \leq n$ and state $s_i \in \mathcal{S}$ such that $|\pi^{k_i}_{m_i}(s_i) - \pi^*_{m_i}(s_i)| \geq \epsilon$ for all i, for all optimal policies π^*. But k_i has a further subsequence k_{i_j} such that $\pi^{k_{i_j}}$ converges to an optimal policy $\bar{\pi}$ as in the proof of Theorem 2.3.8. This leads to a contradiction. Now, fix $0 < \epsilon < 1$ and a period n, and consider any iteration $k \geq K_n$. For any $m \leq n$ and $s \in \mathcal{S}$, we have, $|\pi^k_m(s) - \pi^{k*}_m(s)| < \epsilon$ for some optimal action $\pi^{k^*}_m(s)$. Then, since $\epsilon < 1$ and $\pi^k_m(s), \pi^{k*}_m(s) \in \mathcal{A} = \{1, 2, \ldots, A\}$, we have $\pi^k_m(s) = \pi^{k*}_m(s)$. This proves that all actions up to period n are optimal for policies π^k with $k \geq K_n$. □

We comment that this type of subsequential convergence is the most one can obtain, in general without exploiting any problem-specific features, in infinite-horizon nonstationary sequential decision problems [37, 38].

In this discussion, we did not consider the question of how to solve the inner maximization problems in (2.5) and (2.6) within Algorithm 1. These problems can be solved by following standard procedures from robust MDPs, and in particular, this can be done efficiently when the uncertainty sets $\mathcal{P}^a_{n,s}$ are convex. We refer the readers to [9, 28, 33] for detailed discussions of this issue.

As we stated in Section 2.1, nonstationary MDPs are a special case of countable-state stationary MDPs. The simple policy iteration algorithm for nonstationary MDPs in [22] was extended to countable-state stationary MDPs in [30]. Along similar lines, the work in this chapter is extended to robust countable-state stationary MDPs in the next chapter.

Chapter 3

APPROXIMATE POLICY ITERATION FOR ROBUST COUNTABLE-STATE MARKOV DECISION PROCESSES WITH BOUNDED COSTS

3.1 Introduction

Policy iteration for "nominal" (i.e., non-robust) MDPs starts with an initial guess policy. In every iteration, the value of the current policy is computed and a new policy is obtained by minimizing the Q-function of dynamic programming for each state. For finite-state, finite-action MDPs, policy iteration finds an optimal policy in a finite number of iterations (Theorem 6.4.2 in [35]). For countable-state MDPs, however, this method cannot be implemented directly since it entails solving an infinite system of equations and searching for minima over infinite sets. This issue was recently addressed for the nominal case via approximate versions of policy iteration [22, 30].

In the case of robust MDPs as well, policy evaluation and policy improvement are rendered unimplementable when the state-space is not finite. In fact, a practical method for solving robust countable-state MDPs is not available in the literature, and we provide such an algorithm in this chapter. An as-is implementation of robust policy iteration runs into issues arising from three different sources. The first of these is due to the countable nature of the state-space, analogous to challenges in the nominal case, and is resolved via state-truncation. The algorithm includes only finitely many states in each iteration. This yields approximate versions of both steps of policy iteration, which now comprise finite systems of equations. The main idea behind this approach is that the expected cost for far-away states is small. Such a property holds trivially in the nominal case. In the robust counterpart, this is ensured by a natural assumption on the uncertainty sets, which states that the probability of transitioning to states outside the truncated state-space shrinks uniformly as the size of the state-space is increased.

The other issues are particular to the robust variant. The robust policy evaluation step entails the solution of a non-linear implicit equation, which cannot be performed exactly in general. This is addressed via successive approximation to compute an approximate value of the current policy. Additionally, an "inner" maximization problem must be solved in each successive approximation step to compute the worst-case value. While the exact value may occasionally be found, the maximization needs to be performed numerically to some finite accuracy in most cases. Numerical errors arising from this step are also incorporated into our algorithm.

Thus, we present in this chapter an approximate policy iteration algorithm that can be used in practice. We prove that the algorithm generates a sequence of policies whose value functions converge monotonically to the optimal value function. The policies themselves converge subsequentially to an optimal policy. We also provide examples of robust MDPs which fall within our framework. Any robust MDP with interval uncertainty sets can be solved via the proposed method. In fact, the inner problem can be solved analytically and calculations within the algorithm are greatly simplified. We also consider robust MDPs with bounded transitions, wherein the change in state in a single period is bounded above. Finally, we discuss how our algorithm can be implemented on a robust equipment replacement application that does not fit within these two classes of problems. The MDPs in this chapter are infinite-horizon with stationary data and discounted costs. Immediate costs are bounded, state-spaces are countable and action-spaces are finite. We henceforth omit these qualifiers for brevity.

3.2 Problem formulation

Consider an infinite-horizon stationary MDP with a countable state-space $\mathcal{S} = \{1, 2, 3, \ldots\}$. In any period, the decision-maker observes the current state $s \in \mathcal{S}$ and chooses an action a from a finite set of possible actions $\mathcal{A} = \{1, 2, \ldots, A\}$. We assume, for notational convenience, that the set of actions is independent of the state s, but our analysis can easily be extended to state-dependent finite action sets $\mathcal{A}(s)$. Once an action has been chosen, the system transitions to a state s' with probability $p(s'|s,a)$, incurring a nonnegative cost $c(s,a,s')$. The immediate costs are assumed to be uniformly bounded above by some constant $c > 0$. Thus, $0 \leq c(s,a,s') \leq c$ for all $s, s' \in \mathcal{S}$ and $a \in \mathcal{A}$. Further, the cost incurred in period t is discounted by a factor λ^t, where $\lambda \in (0,1)$ is a constant.

A (deterministic stationary) policy σ is a rule which assigns a unique action to every state. For a policy σ, let $v^\sigma(s)$ be the expected total discounted cost incurred over an infinite horizon when the system is initially in state s. The decision-maker's objective is to find a policy which minimizes this cost for all possible initial states, and this is achieved by solving the Bellman equations

$$v^*(s) = \min_{a \in \mathcal{A}} \ \mathbf{E}_p[c(s,a,s') + \lambda v^*(s')], \ \ s \in \mathcal{S}. \tag{3.1}$$

Here $\mathbf{E}_p[\cdot]$ denotes the expectation with respect to the probability distribution $p(\cdot|s,a)$. That is, $\mathbf{E}_p[u(s')] = \sum\limits_{s' \in \mathcal{S}} p(s'|s,a)u(s')$ for any function u defined on $\mathcal{S}$. The state-action dependence of p is omitted since it is implied by context. An optimal policy is constructed by choosing an action from the argmin set in (3.1) for each state.

In the above setup, the transition probabilities are treated as known model parameters, and we call this model the "nominal" MDP. In practice, these probabilities may be estimated from historical data. The resulting estimation errors are ignored. Since the choice of optimal policy may be sensitive to these errors, robust MDPs try to mitigate their effect by assuming that the transition probabilities are ambiguous and only known to lie in certain prescribed uncertainty sets. More precisely, for each state-action pair (s,a), the probability mass function $p(\cdot|s,a)$ is assumed to lie in a set $\mathcal{P}_s^a$. Here, $\mathcal{P}_s^a$ is a (known) subset of $\mathcal{M}(\mathcal{S})$, the space of all probability mass functions defined on $\mathcal{S}$. For instance, $\mathcal{P}_s^a$ may consist of probability distributions that are "close" to some statistically-estimated nominal distribution. In the robust formulation, the decision-maker follows a conservative approach and seeks to minimize the worst-case expected total discounted cost. Under standard rectangularity assumptions which stipulate that the transition probabilities be independent across periods, the optimal value function for the robust MDP is obtained by solving the robust Bellman equations

$$v^*(s) = \min_{a \in \mathcal{A}} \Big(\sup_{p \in \mathcal{P}_s^a} \mathbf{E}_p[c(s, a, s') + \lambda v^*(s')] \Big), \ s \in \mathcal{S}. \tag{3.2}$$

A robust optimal (stationary) policy is defined for each state by choosing an action from the argmin set above. The maximization problem inside the square brackets is called the inner probLem Detailed analytical treatments of robust MDPs are available in [9, 28, 33].

Observe that if all uncertainty sets are chosen to be singleton, containing only the nominal distribution, equations (3.1) and (3.2) are identical, and the robust MDP reduces to the nominal MDP. We further point out that solving the system of equations (3.2) amounts to finding a value function that is simultaneously optimal for all states s. This is equivalent to optimizing a weighted l^1-norm (called the β-norm) of the value function, defined as $\|u\|_\beta = \sum\limits_{s \in \mathcal{S}} \beta(s)|u(s)|$ for all $u \in V$. Here, β is a strictly positive function on $\mathcal{S}$ such that $\sum\limits_{s \in \mathcal{S}} \beta(s) < \infty$, and V is the space of all bounded functions on $\mathcal{S}$. In particular, β can be viewed as an initial state distribution and the β-norm as the expected value of the worst-case expected total discounted cost defined in Equation (3.2). Policy iteration is a standard method for solving this problem but its implementation is not possible when the state-space is countable. We describe the issues that arise, and our approach to resolving them, in the next section.

3.3 Challenges in designing policy iteration

A detailed description of the policy iteration algorithm for solving robust MDPs is given in [28]. We outline below the two key steps of the simple version of policy iteration. The method is initialized by choosing an arbitrary policy σ. Then, the first step is policy evaluation, wherein the value v^σ of policy σ is computed via the system of equations

$$v^\sigma(s) = \sup_{p \in \mathcal{P}_s^\sigma} \mathbf{E}_p[c(s, \sigma(s), s') + \lambda v^\sigma(s')], \quad s \in \mathcal{S}. \tag{3.3}$$

Here $\mathcal{P}_s^\sigma$ is short-hand for $\mathcal{P}_s^{\sigma(s)}$. In particular, v^σ is a fixed point of the robust evaluation operator $\mathcal{L}^\sigma$ defined on V as

$$\mathcal{L}^\sigma(u)(s) = \sup_{p \in \mathcal{P}_s^\sigma} \mathbf{E}_p[c(s, \sigma(s), s') + \lambda u(s')], \quad s \in \mathcal{S}, \text{ for all } u \in V.$$

The second step is simple policy improvement, wherein a state-action pair $(\bar{s}, \bar{a})$ is chosen as follows.

$$(\bar{s}, \bar{a}) \in \operatorname*{argmin}_{s \in \mathcal{S}, a \in \mathcal{A}} \Big\{ \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_p[c(s, a, s') + \lambda v^\sigma(s')] - v^\sigma(s) \Big\}. \tag{3.4}$$

For any state-action pair (s, a), the term inside the brackets in (3.4) gives the change in cost when action a is chosen in state s instead of that prescribed by the policy σ. This term is zero if $a = \sigma(s)$, and takes a negative value if choosing action a in state s gives a lower total cost. Then, by definition, $(\bar{s}, \bar{a})$ is a state-action pair which gives the largest reduction in total cost, and policy σ is updated by prescribing action $\bar{a}$ in state $\bar{s}$. This policy iteration algorithm, while well-defined in principle, is not implementable for three main reasons that we describe below.

Countable state-space: The first difficulty arises because $\mathcal{S}$ is countable. As such, (3.3) calls for solving infinitely many equations. Similarly, solving (3.4) consists of searching for a minimum over an infinite set. This issue arises in the nominal case as well, and was addressed in Ghate and Smith [22] for non-stationary finite-state MDPs, which can equivalently be viewed as a special case of stationary countable-state MDPs. They developed an implementable and convergent policy iteration algorithm using finite-dimensional approximations. A similar algorithm was also designed in the previous chapter for robust non-stationary finite-state MDPs. The approach in Ghate and Smith was also generalized by Lee et al. in [30] to solve nominal stationary countable-state MDPs with unbounded immediate costs. The key idea was to use finite truncations of the state-space to render the policy evaluation and improvement steps implementable. We use a similar idea of state-truncation. In each iteration, the algorithm includes only the first N states from $\mathcal{S}$, where N itself is chosen adaptively.

Nonlinear evaluation operator: The second issue is particular to the robust MDP. In the nominal case, $\mathcal{L}^{\sigma}$ is a linear operator, and the policy evaluation step in equation (3.3) reduces to solving a finite system of linear equations once the state-space has been truncated. The robust policy evaluation step, however, consists of solving a non-linear implicit equation; this is not possible in general, even when the state-space is finite. We were able to overcome this hurdle in the previous chapter because the policy evaluation equations in the non-stationary case were not implicit owing to the time-staged structure of the probLem In the absence of such structure here, we instead approximate the value function by performing a finite number of iterations of successive approximation. This idea is similar to that used in modified policy iteration [29, 35].

With these two modifications, we now need to solve the system of equations

$$v^{\sigma,N}(s;0) = 0, \qquad s \in \mathcal{S}_N, \tag{3.5}$$

$$v^{\sigma,N}(s;t) = \sup_{p \in \mathcal{P}_s^{\sigma}} \mathbf{E}_{p_N}[c(s,\sigma(s),s') + \lambda v^{\sigma,N}(s';t-1)], \qquad s \in \mathcal{S}_N;\ t = 1,2,\ldots,T. \tag{3.6}$$

Here, $\mathcal{S}_N = \{1,2,\ldots,N\}$ consists of the first N states from $\mathcal{S}$, and T is some integer in the set of natural numbers $\mathbb{N}$. $\mathbf{E}_{p_N}$ denotes the "expectation" over the first N states, that is, $\mathbf{E}_{p_N}[u(s')] = \sum\limits_{s' \le N} p(s'|s,a)u(s')$ for any function $u(\cdot)$ on $\mathcal{S}$. The complementary sum $\sum\limits_{s' > N} p(s'|s,a)u(s')$ is denoted by $\mathbf{E}_{\overline{p_N}}[u(s')]$. Then, $v^{\sigma,N}(s;T)$ is an approximation to the true value function v^{σ}. We can also view $v^{\sigma,N}(s;T)$ as the value of a stationary policy σ for an N-state, T-horizon robust MDP with stationary immediate costs and zero terminal costs. We will call this an (N,T)-approximation to the original MDP.

Solution of the inner problem: Finally, there is one more challenge associated with robust MDPs. The inner problem in (3.6) is itself a maximization problem, and while a closed-form solution may occasionally be found, this problem must be solved numerically in most cases. Moreover, the numerical error from this step must also be incorporated into the algorithm. This issue arises even in the case of finite state-spaces, and Kaufman and Schaefer addressed it in [29] by introducing an "inexact" modified policy iteration algorithm. They solve the inner problem to some pre-defined accuracy, and include the error term in their stopping condition. We use a similar idea here, but the issue is complicated by the countable nature of the state-space. As more states are included in the truncated state-space, the numerical error from this step must asymptotically vanish for the algorithm to converge. More precisely, if the inner problem with N states is solved to some accuracy ϵ_N, we must have $\epsilon_N \to 0$ as $N \to \infty$. This, however, may not always be possible. Note that the inner problem in our model is an infinite-dimensional optimization problem and numerical techniques may fail to solve it to any given tolerance. This necessitates a careful selection of uncertainty sets over which a linear function may be optimized to arbitrary accuracy. This is discussed in more detail in Section 3.5, and natural examples of such sets are provided in Section 3.6. For now, this final level of approximation leads to the following system of equations, which our approximate policy iteration algorithm seeks to solve.

$$\hat{v}^{\sigma,N}(s;0) = 0, \qquad s \in \mathcal{S}_N, \tag{3.7}$$

$$\hat{v}^{\sigma,N}(s;t) \overset{\epsilon_N}{\approx} \sup_{p \in \mathcal{P}_s^\sigma} \mathbf{E}_{p_N}[c(s,\sigma(s),s') + \lambda \hat{v}^{\sigma,N}(s';t-1)], \qquad s \in \mathcal{S}_N;\ t = 1,2,\ldots,T. \tag{3.8}$$

The notation $\overset{\epsilon}{\approx}$ is used to denote an ϵ-approximation, that is, $u \overset{\epsilon}{\approx} \hat{u} \iff |u - \hat{u}| < \epsilon$. In the event that the inner problem can be solved exactly, we choose ϵ_N to be zero for all N, which amounts to solving Equations (3.5)-(3.6).

These three levels of approximation address the issues that would arise in an "as-is" implementation of standard robust policy iteration, and lead to an approximate policy iteration algorithm, every step of which requires a finite amount of memory and computation. Moreover, the state-truncation level N and the number of successive approximation steps T are chosen adaptively via an iterative procedure, so that the action update in the approximate policy improvement step guarantees strict improvement in value in each iteration.

The adaptive procedure is designed such that this improvement is sufficient to ensure value convergence to optimality. These properties of the algorithm and convergence results are discussed in detail in Section 3.5, but we first present the algorithm itself in the next section.

3.4 Algorithm

The proposed approximate policy iteration method is described in Algorithm 2. It starts out with an initial policy. In each iteration $k = 1, 2, \dots$, Step 2(a) initializes the number of states N in the truncated state-space and the number of successive approximation steps T. Steps 2(b)-2(d) find a state-action pair which gives the largest approximate reduction in cost. If this improvement is sufficient and condition (3.13) is satisfied, the variables $N(k)$ and $T(k)$ are assigned values N and T respectively, the current policy is updated and the algorithm proceeds to iteration $k+1$. If not, both N and T are incremented by one and Steps 2(b)-2(d) are repeated. In this manner, the algorithm generates a sequence of policies which strictly improve in value and their value functions converge to the optimal value function v^*.

Step 2(e) checks if the approximate cost-reduction is sufficiently negative so as to provide an improvement in the true value function. This requires the calculation of a parameter $\bar{\delta}(s, a, N, T)$. It is a bound defined in Lemma 3.5.8 via a recursive expression, and it converges to zero as N and T grow to infinity. Algorithm 3 gives a subroutine for computing this expression. The bound $\bar{\delta}$ (and hence the subroutine) does not depend on the current policy, and its computation does not need to be repeated in every iteration. Also, it may sometimes be easier to compute an upper bound on $\bar{\delta}$ which also has the same convergence behavior. In that case, it suffices to replace $\bar{\delta}$ with the said bound. An example of this appears in Section 3.6.2.

Note that policies σ^{k-1} and σ^k only differ in state s^{k-1} and are identical in all other states. In particular, if we set $N < s^{k-1}$ in iteration k, the two policies coincide over $\mathcal{S}_N$. Such an N did not get sufficient cost-reduction in iteration $k-1$, and it will also fail to do so in iteration k. Hence, it is sufficient to initialize $N = s^{k-1}$ instead of $N = 1$. Moreover, our algorithm follows a "diagonal" approach in which N and T are always equal to each

Algorithm 2 Approximate policy iteration for robust countable-state MDPs.

```
1: Initialize: Set iteration counter $k = 1$. Arbitrarily fix the initial policy $\sigma^1$ to one that prescribes the first action in $\mathcal{A}$ in every state. Set $s^0 = 1$.
2: for iterations $k = 1, 2, 3, \ldots,$ do
   (a) Set $N = s^{k-1} = T$, $N(k) = \infty$, and $T(k) = \infty$. Let $\mathcal{S}_N = \{1, 2, \ldots, N\}$.

   Approximate policy evaluation:
   (b) Let $\mathcal{P}_s^k \equiv \mathcal{P}_s^{\sigma^k(s)}$. Compute the approximate value function $v^{k,N}(s;T)$ for all $s \in \mathcal{S}_N$ by performing $T$ steps of successive approximation. That is,

       $v^{k,N}(s;0) = 0, \quad s \in \mathcal{S}_N, \qquad (3.9)$
       $v^{k,N}(s;t) \overset{\epsilon_N}{\approx} \sup_{p \in \mathcal{P}_s^k} \mathbf{E}_{p_N}[c(s, \sigma^k(s), s') + \lambda v^{k,N}(s'; t+1)], \quad s \in \mathcal{S}_N;\ t = 1, 2, \ldots, T. \qquad (3.10)$

   Approximate simple policy improvement:
   (c) For $s \in \mathcal{S}_N$ and $a \in \mathcal{A}$, compute the approximate improvement $\gamma^{k,N}(s,a;T)$ by solving the following system of equations.

       $\gamma^{k,N}(s,a;T) \overset{\epsilon_N}{\approx} \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_{p_N}[c(s,a,s') + \lambda v^{k;N}(s';T)] - v^{k;N}(s;T). \qquad (3.11)$

   (d) Find a state-action pair which minimizes the $\beta$-weighted approximate improvement across all states in $\mathcal{S}_N$ and actions in $\mathcal{A}$. That is, let

       $(s^{k,N}(T), a^{k,N}(T)) \in \operatorname*{argmin}_{s \in \mathcal{S}_N,\ a \in \mathcal{A}} \beta(s)\gamma^{k,N}(s,a;T). \qquad (3.12)$

   (e) Compute $\bar{\delta}(s^{k,N}(T), a^{k,N}(T), N, T)$ via Algorithm 3.
       If

       $\gamma^{k,N}(s^{k,N}(T), a^{k,N}(T); T) < -\bar{\delta}(s^{k,N}(T), a^{k,N}(T), N, T), \qquad (3.13)$

   set $N(k) = N$, $T(k) = T$, $(s^k, a^k) = (s^{k,N}(T), a^{k,N}(T))$, and update policy $\sigma^k$ by choosing $\sigma^{k+1}(s^k) = a^k$, $\sigma^{k+1}(s) = \sigma^k(s)$ for all $s \neq s^k$;
       else set $N = N + 1, T = T + 1$, and go to Step 2(b).
3: end for
```

other. This is a convenient choice which further implies that $N(k) = T(k)$ in every iteration, a property used in Lemma 3.5.16 to establish that both $N(k)$ and $T(k)$ diverge to infinity as k increases. While the convergence of the algorithm relies on this divergence, it does not

Algorithm 3 Subroutine for computing $\bar{\delta}(\bar{s}, \bar{a}, N, T)$

```
**Input** $\bar{s}$, $\bar{a}$, $N$, $T$.
For all $s \in \mathcal{S}_N$ and $a \in \mathcal{A}$, compute
    $\tilde{M}_N(s,a) \overset{\epsilon_N}{\approx} \sup_{p \in \mathcal{P}_s^a} \sum_{s' > N} p(s'|s,a), \quad M_N(s) = \max_{a \in \mathcal{A}} \tilde{M}_N(s,a).$
Initialize $\overline{B}_N(s,a;0) = 0$ for all $s \in \mathcal{S}_N$ and $a \in \mathcal{A}$. For $t = 1, \dots, T$, compute
    $B_N(s;t) = \lambda \max_{a \in \mathcal{A}} \overline{B}_N(s,a;t-1) + \frac{c(1-\lambda^t)}{1-\lambda} M_N(s), \quad s \in \mathcal{S}_N,$
    $\overline{B}_N(s,a;t) \overset{\epsilon_N}{\approx} \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_{p_N}[B_N(s';t)], \quad s \in \mathcal{S}_N, a \in \mathcal{A}.$
Compute
    $\bar{\delta}(\bar{s}, \bar{a}, N, T) = \frac{c\lambda^T}{1-\lambda} + \frac{c}{1-\lambda}\tilde{M}_N(\bar{s},\bar{a}) + B_N(\bar{s};T) + \lambda \overline{B}_N(\bar{s},\bar{a};T) + \frac{2\lambda\epsilon_N}{1-\lambda} + \frac{4c\epsilon_N}{(1-\lambda)^2}.$
```

explicitly use the equality of $N(k)$ and $T(k)$. In other words, any choice of these parameters which ensures that Condition (3.13) is satisfied and $N(k), T(k) \to \infty$ as $k \to \infty$, would yield an implementable convergent algorithm.

Next, we justify our claim from Section 3.3 that this algorithm can be used in practice. The initial policy arbitrarily assigns the first action ($1 \in \mathcal{A}$) to each state. Thus, it has a finite representation even though it is an infinite-dimensional policy. Similarly, since a single action is updated in every iteration, the stationary policy σ^k has at most k actions different from 1 and also has a finite representation. As such, every policy can be stored on a computer with finite memory. The approximate policy evaluation Step 2(b) now consists of solving only finitely many equations, each of which contains finite sums, as opposed to the infinite system in (3.3). Similarly, Step 2(d) in the approximate policy improvement step now entails the search for a minimum over a finite set. Step 2(e) contains an "if" condition which may seemingly never be satisfied, allowing the algorithm to get stuck in an infinite loop. However, we show in Lemma 3.5.10 that if the current policy σ^k is not optimal, then the condition in Step 2(e) is satisfied for some large enough N and T and the loop terminates finitely. The uncertainty sets are chosen so that the inner maximization in Steps 2(c) and 2(d) can be solved to accuracy ϵ_N in finite time. Therefore, each iteration of our algorithm requires finite memory and a finite amount of computation, resolving the difficulty we would have encountered in implementing the standard policy iteration algorithm described in Section 3.2. As such, this approximate version of simple policy iteration is implementable.

Finally, we remark that the algorithm is *not* simply solving to optimality a sequence of finite-state MDPs with increasing state-spaces. We do employ a sequence of finite-state approximations but our approach is more subtle. Values of the truncated MDPs are only computed approximately, which ensures that the policy evaluation step can be performed finitely. Moreover, the size of the approximation and the policy update-scheme are chosen suitably so that monotonic value improvement in each iteration as well as asymptotic convergence to optimality are guaranteed. The convergence results are discussed in the next section.

3.5 Convergence results

Algorithm 2 generates a sequence of policies σ^k. Let v^k denote the corresponding value functions. While the algorithm does not compute v^k explicitly, we prove, via a sequence of lemmas, that these values improve in every iteration and must converge to the optimal value function v^* in the β-norm as $k \to \infty$. Before presenting the proofs, though, we discuss a few subtle issues about the uncertainty sets $\mathcal{P}_s^a$.

First, note that the main premise for using state-truncation as an approximation for a countable-state MDP is that far-away states are less significant. Mathematically, this means that for any state-action pair (s, a), the expected tail cost incurred from states that are sufficiently far away from s, is small and shrinks to zero as more and more states are included in the approximation. This is always true for nominal MDPs, since the expected tail cost can be bounded above by c times the tail probability $\sum\limits_{s'>N} p(s'|s, a)$, which vanishes as N grows. For robust MDPs, however, this issue is non-trivial and and depends on the choice of uncertainty sets. In this case, we need the *worst-case* expected tail cost to shrink to zero when more states are included in the approximate MDP, and this property does not always hold. For example, suppose $\mathcal{P}_s^a$ contains all pmfs which place an atomic mass at one of the states. For simplicity, assume also that $c(s, a, s') \geq 1$ for all $s, s' \in \mathcal{S}$ and $a \in \mathcal{A}$. Then, for any N, the worst-case expected tail cost $\sup\limits_{p \in \mathcal{P}_s^a} \sum\limits_{s' > N} p(s'|s, a)c(s, a, s')$ is at least 1. This implies that a finite-state approximation will never give an arbitrarily good estimate of the total cost for the original MDP.

Such cases can be avoided by a careful choice of uncertainty sets. Intuitively, we need the tail transition-probability to be small regardless of which distribution from $\mathcal{P}_s^a$ is chosen, and this is guaranteed by Assumption 3.5.1 below.

Assumption 3.5.1. *For each state-action pair (s, a) and $N \in \mathcal{S}$, define the maximum tail probability as*

$$M_N(s, a) = \sup_{p \in \mathcal{P}_s^a} \sum_{s' > N} p(s'|s, a). \tag{3.14}$$

Then, $M_N(s, a) \to 0$ as $N \to \infty$.

In the language of probability theory, this is equivalent to stating that the set of pmfs $\mathcal{P}_s^a$ is "tight". By Prokhorov's theorem [15], this further implies that $\mathcal{P}_s^a$ is weakly precompact. That is, for any sequence $p^n \in \mathcal{P}_s^a$, there exists a subsequence $p^{r_n} \in \mathcal{P}_s^a$ and a pmf $p \in \mathcal{M}(\mathcal{S})$ such that $\mathbf{E}_{p^{r_n}}[u(s')] \to \mathbf{E}_p[u(s')]$ as $n \to \infty$, for all functions u on $\mathcal{S}$. Note that the limit p does not necessarily lie in the set $\mathcal{P}_s^a$. This property is used later in the proof of value convergence of the algorithm.

The second issue arises from the inner maximization probLem Recall from Equation (3.6) that for an (N, T)-approximation, the inner problem for a fixed state-action pair (s, a) is of the form

$$\tilde{u}(s) = \sup_{p \in \mathcal{P}_s^a} \sum_{s' \leq N} p(s'|s, a)[c(s, a, s') + \lambda u(s')]. \tag{3.15}$$

The optimization variable is the probability mass function (pmf), the objective function is a linear function of this variable, and the feasible region is the uncertainty set $\mathcal{P}_s^a$. For finite-state robust MDPs, this problem can easily be solved numerically under suitable choice of uncertainty sets – closed convex sets, for example. In the countable-state case, however, this is not straightforward since (3.15) becomes an infinite-dimensional optimization probLem While an analytical solution may occasionally be available, numerical methods must be used in general. As such, it may not even be possible to practically solve this problem to arbitrary accuracy, which is essential for the implementability and convergence of the proposed policy iteration algorithm. We observe that the objective function depends only on the first N components of p. Thus, optimizing over $\mathcal{P}_s^a$ is equivalent to optimizing over its algebraic projection onto the finite-dimensional space $\mathbb{R}^N$, effectively making it a finite-dimensional optimization probLem Thus, in theory, solving the inner problem here is as easy as the finite-state case. But this idea is not very useful in practice since it requires an algebraic representation of the projection $\mathcal{P}_s^a$ onto its first N components. This may not be possible for many common uncertainty sets, thereby calling for a more delicate handling of this issue. We omit further details here, and just state that the uncertainty sets must be chosen so that the inner problem can be solved to arbitrary accuracy. In Section 3.6, we provide examples of problems where this can be achieved.

Now, we state and prove several lemmas, which help us establish the main convergence results in Theorems 3.5.17 and 3.5.18. For ease of exposition, interdependence of the various results and their contributions are summarized in Figure 3.1 and Table 3.1.

The first lemma gives a necessary and sufficient condition for a policy to be optimal, and is simply a restatement of the fact that the robust Bellman equations must be satisfied at optimality. For a given policy σ, v^σ denotes its value function. For each state-action pair (s, a), let $\gamma^\sigma(s, a)$ be the improvement obtained by changing the policy in a single state s by

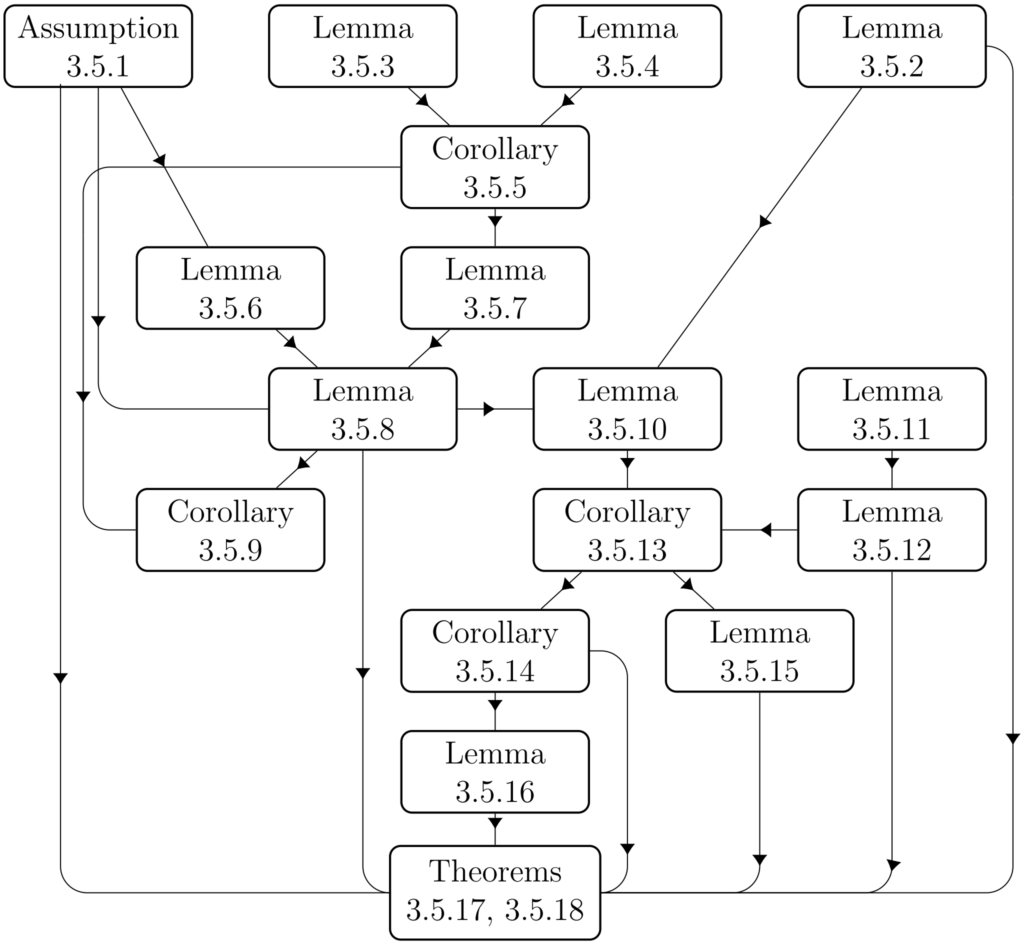


Figure 3.1: A schematic representation of the interdependence of results in Section 3.5.

replacing action $\sigma(s)$ with a. Then,

$$\gamma^{\sigma}(s,a) = \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_p[c(s,a,s') + \lambda v^{\sigma}(s')] - v^{\sigma}(s). \tag{3.16}$$

Lemma 3.5.2. *A policy σ is optimal if and only if $\gamma^{\sigma}(s,a) \geq 0$ for all $s \in \mathcal{S}$ and $a \in \mathcal{A}$.*

Proof. For any state s,

$$v^{\sigma}(s) = \sup_{p\in\mathcal{P}_s^{\sigma}} \mathbf{E}_p[c(s,\sigma(s),s') + \lambda v^{\sigma}(s')] \;\geq\; \min_{a\in\mathcal{A}} \Big\{ \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_p[c(s,a,s') + \lambda v^{\sigma}(s')] \Big\}. \tag{3.17}$$

	Description	Results used
Assum 3.5.1	Tail probabilities vanish uniformly.	—
Lem 3.5.2	Optimal policies satisfy Bellman equations.	—
Lem 3.5.3	Error due to approximate solution of inner prob-Lem	—
Lem 3.5.4	Error due to (N, T)-approximation of the MDP.	—
Cor 3.5.5	Approximation bounds for true versus approximate value functions.	Lem 3.5.3, 3.5.4
Lem 3.5.6	Uniform convergence of expectations.	Assum 3.5.1
Lem 3.5.7	Policy-dependent bound δ on difference between true and approximate improvement.	Cor 3.5.5
Lem 3.5.8	Existence and convergence of a policy-independent bound $\bar{\delta} \geq \delta$.	Assum 3.5.1, Lem 3.5.6, 3.5.7
Cor 3.5.9	Convergence of approximate value of a policy to its true value.	Cor 3.5.5, Lem 3.5.8
Lem 3.5.10	Finite termination of the 'if' loop in the algorithm.	Lem 3.5.2, 3.5.8
Lem 3.5.11	Contraction property of the robust evaluation operator.	—
Lem 3.5.12	Policy update by an improving state-action pair gives a better policy.	Lem 3.5.12
Cor 3.5.13	Policies generated by the algorithm are nonincreasing in value.	Lem 3.5.11, 3.5.12
Cor 3.5.14	The algorithm never repeats a non-optimal policy.	Cor 3.5.13
Lem 3.5.15	Weighted improvement for (s^k, a^k) asymptotically vanishes.	Cor 3.5.14
Lem 3.5.16	The number of states included $N(k)$ and the number of successive approximation steps $T(k)$ diverge to infinity.	Cor 3.5.14
Theorems 3.5.17 & 3.5.18	Policies generated by the algorithm converge in value to optimal; policies converge subsequentially to an optimal policy.	Assum 3.5.1, Lem 3.5.2, 3.5.8, 3.5.12, 3.5.15, 3.5.16, Cor 3.5.14

Table 3.1: A summary of contribution and interdependence of the results in Section 3.5.

First suppose that $\gamma^\sigma(s,a) \geq 0$ for all $s \in \mathcal{S}, a \in \mathcal{A}$. Then, by definition of $\gamma^\sigma(\cdot,\cdot)$,

$$v^\sigma(s) \leq \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_p[c(s,a,s') + \lambda v^\sigma(s')] \text{ for all } s \in \mathcal{S}, a \in \mathcal{A}$$

$$\implies v^\sigma(s) \leq \min_{a \in \mathcal{A}} \left\{ \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_p[c(s,a,s') + \lambda v^\sigma(s')] \right\} \text{ for all } s \in \mathcal{S}.$$

This combined with (3.17) implies that

$$v^\sigma(s) = \min_{a\in\mathcal{A}} \left\{ \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_p[c(s,a,s') + \lambda v^\sigma(s')] \right\} \text{ for all } s \in \mathcal{S}.$$

Thus, v^σ satisfies the robust Bellman equations and must be the optimal value function. Hence, the policy σ is optimal.

Conversely, suppose that policy σ is optimal. Then,

$$\begin{aligned} v^\sigma(s) &= \min_{a\in\mathcal{A}} \left\{ \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_p[c(s,a,s') + \lambda v^\sigma(s')] \right\} \text{ for all } s \in \mathcal{S} \\ &\le \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_p[c(s,a,s') + \lambda v^\sigma(s')] \text{ for all } s \in \mathcal{S}, a \in \mathcal{A} \\ \implies & 0 \le \gamma^\sigma(s,a) \text{ for all } s \in \mathcal{S}, a \in \mathcal{A}. \end{aligned}$$

This completes the proof. □

We noted in Section 3.3 that it may not be possible to solve the inner maximization problem in Equation (3.6) exactly. As such, for any N and T, the algorithm solves the inner problem to a pre-defined accuracy ϵ_N in each step of successive approximation. The next lemma computes the total error accumulated as a result.

Lemma 3.5.3. *For fixed N and T, and $t = 1, \ldots, T$, let $v^{\sigma,N}(\cdot,t)$ be obtained from Equations (3.5)-(3.6); and let $\hat{v}^{\sigma,N}(\cdot,t)$ be obtained from Equations (3.7)-(3.8). Then,*

$$|v^{\sigma,N}(s,t) - \hat{v}^{\sigma,N}(s,t)| < \epsilon_N \frac{1-\lambda^t}{1-\lambda}, \quad \textit{for all } s \in \mathcal{S}_N,\ t = 1,2,\ldots,T. \tag{3.18}$$

Proof. Recall that $v^{\sigma,N}(\cdot,t)$ is the value function for the (N,T)-approximation when the inner problem in Equation (3.6) is solved exactly. Similarly, $\hat{v}^{\sigma,N}(\cdot,t)$ is the value function obtained when the inner problems are solved to accuracy ϵ_N as in Equation (3.8). We will prove the result by induction on $t = 0, 1, \ldots, T$.

We introduce an intermediate function $\tilde{v}^{\sigma,N}(s;t)$ obtained by solving (3.8) exactly. That is,

$$\tilde{v}^{\sigma,N}(s;t) = \sup_{p\in\mathcal{P}_s^\sigma} \mathbf{E}_{p_N}[c(s,\sigma(s),s') + \lambda\hat{v}^{\sigma,N}(s';t-1)], \; s \in \mathcal{S}_N, t = 1, 2, \ldots, T. \tag{3.19}$$

Thus, $\hat{v}^{\sigma,N}(s;t) \overset{\epsilon_N}{\approx} \tilde{v}^{\sigma,N}(s;t)$ for all states $s \in \mathcal{S}_N$ and for all $t = 1, \ldots, T$.

Since $v^{\sigma,N}(s;0) = 0 = \hat{v}^{\sigma,N}(s;0)$ for all $s \in \mathcal{S}_N$, it follows from Equation (3.19) that

$$\begin{aligned}\tilde{v}^{\sigma,N}(s;1) &= \sup_{p\in\mathcal{P}_s^\sigma} \mathbf{E}_{p_N}[c(s,\sigma(s),s') + \lambda\hat{v}^{\sigma,N}(s';0)] \\ &= \sup_{p\in\mathcal{P}_s^\sigma} \mathbf{E}_{p_N}[c(s,\sigma(s),s') + \lambda v^{\sigma,N}(s';0)] = \hat{v}^{\sigma,N}(s;1).\end{aligned}$$

Therefore,

$$|\hat{v}^{\sigma,N}(s;1) - v^{\sigma,N}(s;1)| \leq |\hat{v}^{\sigma,N}(s;1) - \tilde{v}^{\sigma,N}(s;1)| + |\tilde{v}^{\sigma,N}(s;1) - v^{\sigma,N}(s;1)| < 0 + \epsilon_N = \epsilon_N \; \forall s \in \mathcal{S}_N.$$

So the result holds for $t = 1$. Now, suppose the result is true for some $t < T$ and we will prove it for $t + 1$. Once again, for any $s \in \mathcal{S}_N$,

$$\begin{aligned}\tilde{v}^{\sigma,N}(s;t+1) &= \sup_{p\in\mathcal{P}_s^\sigma} \mathbf{E}_{p_N}[c(s,\sigma(s),s') + \lambda\hat{v}^{\sigma,N}(s;t)] \\ &\leq \sup_{p\in\mathcal{P}_s^\sigma} \mathbf{E}_{p_N}\left[c(s,\sigma(s),s') + \lambda\Big(v^{\sigma,N}(s';t) + (1-\lambda^t)\frac{\epsilon_N}{1-\lambda}\Big)\right] \\ &\leq \sup_{p\in\mathcal{P}_s^\sigma} \mathbf{E}_{p_N}[c(s,\sigma(s),s') + \lambda v^{\sigma,N}(s';t)] + (\lambda - \lambda^{t+1})\frac{\epsilon_N}{1-\lambda} \\ &= v^{\sigma,N}(s;t+1) + (\lambda - \lambda^{t+1})\frac{\epsilon_N}{1-\lambda} \\ \implies \hat{v}^{\sigma,N}(s;t+1) &\leq \tilde{v}^{\sigma,N}(s;t+1) + \epsilon_N \\ &\leq v^{\sigma,N}(s;t+1) + (\lambda - \lambda^{t+1})\frac{\epsilon_N}{1-\lambda} + \epsilon_N = v^{\sigma,N}(s;t+1) + (1 - \lambda^{t+1})\frac{\epsilon_N}{1-\lambda}.\end{aligned}$$

Similarly,

$$\begin{aligned}
v^{\sigma,N}(s;t+1) &= \sup_{p\in\mathcal{P}_s^\sigma} \mathbf{E}_{p_N}[c(s,\sigma(s),s') + \lambda v^{\sigma,N}(s';t)] \\
&\le \sup_{p\in\mathcal{P}_s^\sigma} \mathbf{E}_{p_N}\Big[c(s,\sigma(s),s') + \lambda\Big(\hat{v}^{\sigma,N}(s';t) + (1-\lambda^t)\frac{\epsilon_N}{1-\lambda}\Big)\Big] \\
&\le \sup_{p\in\mathcal{P}_s^\sigma} \mathbf{E}_{p_N}[c(s,\sigma(s),s') + \lambda \hat{v}^{\sigma,N}(s';t)] + (\lambda-\lambda^{t+1})\frac{\epsilon_N}{1-\lambda} \\
&= \tilde{v}^{\sigma,N}(s;t+1) + (\lambda-\lambda^{t+1})\frac{\epsilon_N}{1-\lambda} \\
&\le \hat{v}^{\sigma,N}(s;t+1) + \epsilon_N + (\lambda-\lambda^{t+1})\frac{\epsilon_N}{1-\lambda} = \hat{v}^{\sigma,N}(s;t+1) + (1-\lambda^{t+1})\frac{\epsilon_N}{1-\lambda}.
\end{aligned}$$

Thus, $|v^{\sigma,N}(s;t) - \hat{v}^{\sigma,N}(s;t)| < \epsilon_N(1-\lambda^t)/(1-\lambda)$ for all $t = 1,\dots,T$. This completes the proof. □

Even when the inner problem can be solved exactly, the algorithm never computes the true value v^σ of policy σ. Instead, it computes an (N,T)-approximation to v^σ via equations (3.5)-(3.6). Our next lemma provides bounds on the quality of this approximation.

Lemma 3.5.4. *Let σ be a fixed policy with value v^σ. For fixed N and T, let $v^{\sigma,N}(\cdot;T)$ be its approximate value function obtained via Equations (3.5)-(3.6). For all states $s\in\mathcal{S}_N$, we have,*

$$v^{\sigma,N}(s;T) \le v^\sigma(s) \le v^{\sigma,N}(s;T) + \frac{c\lambda^t}{1-\lambda} + \mathcal{E}_N(s;\sigma,T), \tag{3.20}$$

where $\mathcal{E}_N(s;\sigma,0)=0$ and for $t=1,2,\dots,T$,

$$\mathcal{E}_N(s;\sigma,t) = \lambda \sup_{p\in\mathcal{P}_s^\sigma} \mathbf{E}_{p_N}[\mathcal{E}_N(s';\sigma,t-1)] + \frac{c(1-\lambda^t)}{1-\lambda} M_N(s,\sigma(s)). \tag{3.21}$$

Proof. We will prove the result in two steps by introducing an intermediate approximation to the value function. Let $v^\sigma(\cdot;T)$ be the approximate value of a policy σ obtained through T steps of successive approximation, starting with an initial guess 0. Note that $v^\sigma(\cdot;T)$ is defined for all states in $\mathcal{S}$. So,

$$v^\sigma(s;0) = 0, \quad s \in \mathcal{S} \tag{3.22}$$

$$v^\sigma(s;t) = \sup_{p \in \mathcal{P}_s^\sigma} \mathbf{E}_p[c(s,\sigma(s),s') + \lambda v^\sigma(s';t-1)], \quad s \in \mathcal{S};\ t = 1,\ldots,T. \tag{3.23}$$

We first obtain a relationship between $v^\sigma(\cdot;T)$ and $v^\sigma(\cdot)$, using induction on $t = 0,1,\ldots,T$. For any state $s \in \mathcal{S}$, the true value function is non-negative and bounded above by $c/(1-\lambda)$, that is, $0 \le v^\sigma(s) \le c/(1-\lambda)$. This implies that $v^\sigma(s;0) \le v^\sigma(s) \le v^\sigma(s;0) + c/(1-\lambda)$ for all $s \in \mathcal{S}$. Now suppose for some $t < T$, we have,

$$v^\sigma(s';t) \le v^\sigma(s') \le v^\sigma(s';t) + \frac{c\lambda^t}{1-\lambda},\ \forall\ s' \in \mathcal{S}.$$

We will show that the result holds for $t+1$ as well. Fix a state $s \in \mathcal{S}$. Multiplying the above inequality by λ and adding $c(s,\sigma(s),s')$ gives

$$c(s,\sigma(s),s') + \lambda v^\sigma(s';t) \le c(s,\sigma(s),s') + \lambda v^\sigma(s') \le c(s,\sigma(s),s') + \lambda v^\sigma(s';t) + \frac{c\lambda^{t+1}}{1-\lambda},\ \forall\ s' \in \mathcal{S}.$$

Further multiplying the above with any $p(\cdot|s,\sigma(s)) \in \mathcal{P}_s^\sigma$ and summing over all $s' \in \mathcal{S}$, we have

$$\begin{aligned}
\mathbf{E}_p[c(s,\sigma(s),s') + \lambda v^\sigma(s';t)] &\le \mathbf{E}_p[c(s,\sigma(s),s') + \lambda v^\sigma(s')] \\
&\le \mathbf{E}_p\Big[c(s,\sigma(s),s') + \lambda v^\sigma(s';t) + \frac{c\lambda^{t+1}}{1-\lambda}\Big] \\
&= \mathbf{E}_p[c(s,\sigma(s),s') + \lambda v^\sigma(s';t)] + \frac{c\lambda^{t+1}}{1-\lambda}.
\end{aligned}$$

Finally, taking suprema over $\mathcal{P}_s^\sigma$ gives

$$v^\sigma(s;t+1) \le v^\sigma(s) \le v^\sigma(s;t+1) + \frac{c\lambda^{t+1}}{1-\lambda}\ \forall\ s \in \mathcal{S}.$$

Thus, we have for all $t = 1, \ldots, T$ and for all $s \in \mathcal{S}$,

$$v^{\sigma}(s;t) \leq v^{\sigma}(s) \leq v^{\sigma}(s;t) + \frac{c\lambda^t}{1-\lambda}. \tag{3.24}$$

Next, we compute the error introduced by state-truncation. For this, we obtain a relationship between $v^{\sigma,N}(\cdot;T)$ defined in Equations (3.5)-(3.6), and $v^{\sigma}(\cdot;T)$ defined in Equations (3.22)-(3.23). We claim that for $t = 0, 1, \ldots, T$, we have

$$v^{\sigma,N}(s;t) \leq v^{\sigma}(s;t) \leq v^{\sigma,N}(s;t) + \mathcal{E}_N(s;\sigma,t), \tag{3.25}$$

where $\mathcal{E}_N(s;\sigma,t)$ is defined in Equation (3.21). Since $\mathcal{E}_N(s;\sigma,0) = 0$ and $v^{\sigma,N}(s;0) = 0 = v^{\sigma}(\cdot;0)$ for all $s \in \mathcal{S}_N$, the result is true for $t = 0$. We complete the proof by induction on $t = 0, 1, \ldots, T$. Observe that

$$v^{\sigma}(s;t) \leq c(1 + \lambda + \ldots + \lambda^{t-1}) = \frac{c}{1-\lambda}(1-\lambda^t) \quad \forall\ s \in \mathcal{S},\ t = 1, 2, \ldots, T. \tag{3.26}$$

Suppose our claim is true for some $t < T$. Then, for $t+1$, for a fixed state $s \in \mathcal{S}_N$, we have

$$\begin{aligned} v^{\sigma,N}(s;t+1) &= \sup_{p \in \mathcal{P}_s^{\sigma}} \mathbf{E}_{p_N}[c(s,\sigma(s),s') + \lambda v^{\sigma,N}(s';t)] \\ &\leq \sup_{p \in \mathcal{P}_s^{\sigma}} \mathbf{E}_{p_N}[c(s,\sigma(s),s') + \lambda v^{\sigma}(s';t)] \ \text{ (by induction hypothesis)} \\ &\leq \sup_{p \in \mathcal{P}_s^{\sigma}} \mathbf{E}_{p}[c(s,\sigma(s),s') + \lambda v^{\sigma}(s';t)] = v^{\sigma}(s;t+1). \end{aligned}$$

Conversely,

$$\begin{aligned}
v^{\sigma}(s;t+1) &\leq \sup_{p\in\mathcal{P}_s^{\sigma}} \mathbf{E}_{p_N}[c(s,\sigma(s),s') + \lambda v^{\sigma}(s';t)] + \sup_{p\in\mathcal{P}_s^{\sigma}} \mathbf{E}_{\overline{p_N}}[c(s,\sigma(s),s') + \lambda v^{\sigma}(s';t)] \\
&\leq \sup_{p\in\mathcal{P}_s^{\sigma}} \mathbf{E}_{p_N}[c(s,\sigma(s),s') + \lambda(v^{\sigma,N}(s';t) + \mathcal{E}_N(s';\sigma,t))] \\
&\quad + \sup_{p\in\mathcal{P}_s^{\sigma}} \mathbf{E}_{\overline{p_N}}\Big[c + \frac{c\lambda}{1-\lambda}(1-\lambda^t)\Big] \\
&\leq \sup_{p\in\mathcal{P}_s^{\sigma}} \mathbf{E}_{p_N}[c(s,\sigma(s),s') + \lambda v^{\sigma,N}(s';t)] + \lambda \sup_{p\in\mathcal{P}_s^{\sigma}} \mathbf{E}_{p_N}[\mathcal{E}_N(s';\sigma,t)] \\
&\quad + c\Big(\frac{1-\lambda^{t+1}}{1-\lambda}\Big) M_N(s,\sigma(s)) \\
&= v^{\sigma,N}(s;t+1) + \lambda \sup_{\substack{p(\cdot|s,\sigma(s))\\ \in\mathcal{P}_s^{\sigma}}} \mathbf{E}_{p_N}[\mathcal{E}_N(s';\sigma,t)] + c\left(\frac{1-\lambda^{t+1}}{1-\lambda}\right) M_N(s,\sigma(s)) \\
&= v^{\sigma,N}(s;t+1) + \mathcal{E}_N(s;\sigma,t+1),
\end{aligned}$$

where $\mathcal{E}_N(s;\sigma,t+1)$ is defined in Equation (3.21). Thus, the claim is true for all $t = 0,1,\ldots,T$.

Finally, combining Equations (3.24) and (3.25) and plugging in $t = T$ completes the proof. □

The algorithm explicitly computes the approximate value function $\hat{v}^{\sigma,N}(\cdot;T)$, and the next corollary combines Lemmas 3.5.3 and 3.5.4 to bound the deviation of this approximation from the true value function $v^{\sigma}(\cdot)$.

Corollary 3.5.5.

$$\hat{v}^{\sigma,N}(s;T) - \frac{1-\lambda^T}{1-\lambda}\epsilon_N \leq v^{\sigma}(s) \leq \hat{v}^{\sigma,N}(s;T) + \frac{c\lambda^T}{1-\lambda} + \mathcal{E}_N(s;\sigma,T) + \epsilon_N\frac{1-\lambda^T}{1-\lambda},\ \forall\ s\in\mathcal{S}_N.$$

Proof. The result follows from Equations (3.18) and (3.20). □

The algorithm uses state truncation, successive approximation and approximate solution of inner problems to explicitly compute an approximate value function. A desirable property in any good approximation is that it asymptotically recover the true value. Towards this goal, the following results will show that the error terms obtained in Corollary 3.5.5 vanish as N and T are made arbitrarily large. But we first present a result that establishes uniform convergence of certain expectations. This will be utilized in subsequent proofs.

Suppose $u_N(\cdot)$ is a bounded sequence of functions that converges pointwise to zero on $\mathcal{S}$ as $n \to \infty$. Then, for any pmf $p \in \mathcal{M}(\mathcal{S})$, the expected value $\mathbf{E}_p[u_N(s')] \to 0$ by the Dominated Convergence Theorem. However, the approximation bounds obtained in the previous lemmas contain terms of the form $\sup\limits_{p \in \mathcal{P}_s^a} \mathbf{E}_p[u_N(s')]$. It is not obvious whether these worst-case expected values vanish as well. The next lemma establishes that they do, provided the uncertainty sets satisfy Assumption 3.5.1.

Lemma 3.5.6. *For all $s \in \mathcal{S}$, let $u_N(s)$ be a non-negative sequence that converges to 0 as $N \to \infty$, and is uniformly bounded over $\mathcal{S}$ by some constant U. For all $N \in \mathbb{N}$, define*

$$a_N = \sup_{p \in \mathcal{P}} \mathbf{E}_p[u_N(s')],$$

where $\mathcal{P}$ satisfies Assumption 3.5.1. Then, a_N converges to 0 as $N \to \infty$ and is uniformly bounded above by U.

Proof. It is easy to see that $0 \leq a_N \leq U$ for all N.

Further, for any $n \in \mathbb{N}$, we can write

$$\begin{aligned} a_N &= \sup_{p \in \mathcal{P}} \left(\mathbf{E}_{p_n}[u_N(s']) + \mathbf{E}_{\overline{p_n}}[u_N(s')] \right) \\ &\leq \sup_{p \in \mathcal{P}} \mathbf{E}_{p_n}[u_N(s']) + \sup_{p \in \mathcal{P}} \mathbf{E}_{\overline{p_n}}[u_N(s')] \\ &\leq \sup_{p \in \mathcal{P}} \mathbf{E}_{p_n}[u_N(s']) + U \sup_{p \in \mathcal{P}} \mathbf{E}_{\overline{p_n}}[1]. \end{aligned}$$

Given $\epsilon > 0$, choose n so that $\sup\limits_{p \in \mathcal{P}} \mathbf{E}_{\overline{p_n}}[1] = \sup\limits_{p \in \mathcal{P}} \sum\limits_{s' > n} p(s') < \epsilon/(2U)$. Such an n exists since $\mathcal{P}$ satisfies Assumption 3.5.1.

Further, note that $u_N(s)$ vanishes uniformly over $s \in \{1, 2, \ldots, n\}$, that is, we can find

$N_0 \in \mathbb{N}$ such that for $N \geq N_0$, $u_N(s) < \epsilon/2$ for all $s \in \{1, 2, \dots, n\}$. Then, for $N \geq N_0$, we have

$$a_N \leq \frac{\epsilon}{2} \sup_{p \in \mathcal{P}} \mathbf{E}_{p_n}[1] + U \frac{\epsilon}{2U} \leq \epsilon.$$

Since $\epsilon > 0$ was arbitrary, this completes the proof. □

The policy improvement step uses the value of the current policy to choose a state-action pair which gives the maximum improvement. Since the true value v^σ is not available, we cannot compute the true improvement either. Instead, the algorithm computes an approximation $\gamma^{\sigma,N}(\cdot, \cdot; T)$ defined as

$$\gamma^{\sigma,N}(s, a; T) \overset{\epsilon_N}{\approx} \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_{p_N}[c(s, a, s') + \lambda \hat{v}^{\sigma,N}(s'; T)] - \hat{v}^{\sigma,N}(s; T), \ \forall \ s \in \mathcal{S}_N, \ a \in \mathcal{A}. \tag{3.27}$$

The next lemma establishes a relation between the approximate improvement and the true improvement defined in Equation (3.16).

Lemma 3.5.7. *For a fixed policy σ and for any state $s \in \mathcal{S}_N$, action $a \in \mathcal{A}$,*

$$\left|\gamma^{\sigma,N}(s, a; T) - \gamma^\sigma(s, a)\right| \leq \delta(s, a, \sigma, N, T), \tag{3.28}$$

where

$$\delta(s, a, \sigma, N, T) = \frac{c}{1 - \lambda} \left(\lambda^T + 2\epsilon_N + M_N(s, a)\right) + \mathcal{E}_N(s; \sigma, T) + \lambda \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_{p_N}[\mathcal{E}_N(s'; \sigma, T)]. \tag{3.29}$$

Proof. For any state $s \in \mathcal{S}_N$ and action $a \in \mathcal{A}$,

$$
\begin{aligned}
\gamma^\sigma(s,a) &= \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_p[c(s,a,s') + \lambda v^\sigma(s')] - v^\sigma(s) \\
&\leq \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_{p_N}[c(s,a,s') + \lambda v^\sigma(s')] + \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_{\overline{p_N}}[c(s,a,s') + \lambda v^\sigma(s')] - v^\sigma(s) \\
&\leq \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_{p_N}\Big[c(s,a,s') + \lambda\Big(\hat{v}^{\sigma,N}(s';T) + \frac{c\lambda^T}{1-\lambda} + \epsilon_N \frac{1-\lambda^T}{1-\lambda} + \mathcal{E}_N(s';\sigma,T)\Big)\Big] \\
&\quad + \Big(c + \frac{\lambda c}{1-\lambda}\Big) M_N(s,a) - \hat{v}^{\sigma,N}(s;T) + \epsilon_N \frac{1-\lambda^T}{1-\lambda} \qquad \text{(by Corollary 3.5.5)} \\
&\leq \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_{p_N}[c(s,a,s') + \lambda \hat{v}^{\sigma,N}(s;T)] - \hat{v}^{\sigma,N}(s;T) \\
&\quad + \frac{c\lambda^{T+1}}{1-\lambda} + \lambda\epsilon_N \frac{1-\lambda^T}{1-\lambda} + \lambda \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_{p_N}[\mathcal{E}_N(s';\sigma,T)] + \frac{c}{1-\lambda} M_N(s,a) + \epsilon_N \frac{1-\lambda^T}{1-\lambda} \\
&\leq \gamma^{\sigma,N}(s,a;T) + \epsilon_N \\
&\quad + \frac{c\lambda^{T+1}}{1-\lambda} + (1+\lambda)\epsilon_N \frac{1-\lambda^T}{1-\lambda} + \lambda \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_{p_N}[\mathcal{E}_N(s';\sigma,T)] + \frac{c}{1-\lambda} M_N(s,a) \\
&\leq \gamma^{\sigma,N}(s,a;T) + \frac{c\lambda^{T+1}}{1-\lambda} + 2\epsilon_N \frac{1-\lambda^{T+1}}{1-\lambda} + \lambda \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_{p_N}[\mathcal{E}_N(s';\sigma,T)] + \frac{c}{1-\lambda} M_N(s,a).
\end{aligned}
$$

Conversely, for any state $s \in \mathcal{S}_N$,

$$\begin{aligned}
\gamma^{\sigma,N}(s,a;T) &\overset{\epsilon_N}{\approx} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}[c(s,a,s') + \lambda\hat{v}^{\sigma,N}(s';T)] - \hat{v}^{\sigma,N}(s;T) \\
&\leq \epsilon_N + \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}[c(s,a,s') + \lambda\hat{v}^{\sigma,N}(s';T)] - \hat{v}^{\sigma,N}(s;T) \\
&\leq \epsilon_N + \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}\Big[c(s,a,s') + \lambda v^{\sigma}(s') + \epsilon_N \frac{1-\lambda^T}{1-\lambda}\Big] \\
&- v^{\sigma}(s) + \frac{c\lambda^T}{1-\lambda} + \mathcal{E}_N(s;\sigma,T) + \epsilon_N\frac{1-\lambda^T}{1-\lambda} \qquad \text{(by Corollary 3.5.5)} \\
&\leq \epsilon_N + \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p}\Big[c(s,a,s') + \lambda v^{\sigma}(s') + \epsilon_N \frac{1-\lambda^T}{1-\lambda}\Big] \\
&- v^{\sigma}(s) + \frac{c\lambda^T}{1-\lambda} + \mathcal{E}_N(s;\sigma,T) + \epsilon_N\frac{1-\lambda^T}{1-\lambda} \\
&\leq \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p}[c(s,a,s') + \lambda v^{\sigma}(s')] - v^{\sigma}(s) \\
&+ \epsilon_N\Big(1 + \lambda\frac{1-\lambda^T}{1-\lambda} + \frac{1-\lambda^T}{1-\lambda}\Big) + \frac{c\lambda^T}{1-\lambda} + \mathcal{E}_N(s;\sigma,T) \\
&= \gamma^{\sigma}(s,a) + 2\epsilon_N\frac{1-\lambda^{T+1}}{1-\lambda} + \frac{c\lambda^T}{1-\lambda} + \mathcal{E}_N(s;\sigma,T).
\end{aligned}$$

Thus, for all $s \in \mathcal{S}_N, a \in \mathcal{A}$, we have,

$$\begin{aligned}
\left|\gamma^{k,N}(s,a;T) - \gamma^{k}(s,a)\right| &\leq \frac{c}{1-\lambda}\left(\lambda^T + 2\epsilon_N + M_N(s,a)\right) + \mathcal{E}_N(s;\sigma,T) + \lambda \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}[\mathcal{E}_N(s';\sigma,T)] \\
&= \delta(s,a,\sigma,N,T).
\end{aligned}$$

This completes the proof. □

The bound δ in the previous lemma allows us to choose a suitable state-action pair in the policy update step so that the true value function improves. For the algorithm to be convergent, however, we need an upper bound on δ which is policy-independent and vanishes asymptotically as N and T grow. The existence of such a bound $\bar{\delta}$ is established in the next lemma. It is obtained via a recursive expression, and a subroutine for computing it is provided in Algorithm 3. In some cases, it may be easier to compute an upper bound on $\bar{\delta}$ which demonstrates the same convergence behavior. We remark that replacing $\bar{\delta}$ with such a bound works just as well in the algorithm. An example of this appears in Section 3.6.2.

Lemma 3.5.8. *There exists a policy-independent bound $\bar{\delta}(s, a, N, T)$ such that $\delta(s, a, \sigma, N, T) \leq \bar{\delta}(s, a, N, T)$ for all $s \leq N$, $a \in \mathcal{A}$ and policies σ, and $\bar{\delta}(s, a, N, T) \to 0$ as $N, T \to \infty$.*

Proof. For every $N \in \mathcal{S}$ and state $s \in \mathcal{S}_N$, let $\tilde{M}_N(s, a) \overset{\epsilon_N}{\approx} M_N(s, a)$ and let $M_N(s) = \max\limits_{a \in \mathcal{A}} \tilde{M}_N(s, a)$. Since there are finitely many actions in $\mathcal{A}$, it follows from Assumption 3.5.1 and by choice of ϵ_N that $M_N(s)$ also vanishes as $N \to \infty$.

Further, let $B_N(s; 0) = 0 = \overline{B}_N(s, a; 0) = 0$. For $t = 1, \ldots, T$, define

$$\begin{aligned} B_N(s; t) &= \lambda \max_{a \in \mathcal{A}} \overline{B}_N(s, a; t-1) + \frac{c(1-\lambda^t)}{1-\lambda} M_N(s), & \forall\ s \in \mathcal{S}_N, \\ \overline{B}_N(s, a; t) &\overset{\epsilon_N}{\approx} \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_{p_N}[B_N(s', t)], & \forall\ s \in \mathcal{S}_N,\ a \in \mathcal{A}. \end{aligned}$$

Recall the definition of error $\mathcal{E}_N(\cdot, \cdot; t)$ from Equation (3.21). For a fixed policy σ, state $s \in \mathcal{S}_N$ and $t = 1, \ldots, T$, we show that $\mathcal{E}_N(s; \sigma, t) \leq B_N(s; t) + (\lambda + c(1 + \ldots + \lambda^{t-1}))\epsilon_N/(1-\lambda)$. For $t = 1$, we have

$$\mathcal{E}_N(s; \sigma, 1) = cM_N(s, \sigma(s)) \leq cM_N(s) + c\epsilon_N = B_N(s, 1) + c\epsilon_N \leq B_N(s, 1) + \frac{\lambda + c}{1-\lambda}\epsilon_N.$$

Suppose the relation holds for some $t < T$. Then, for $t+1$, we have

$$\begin{aligned}
\mathcal{E}_N(s;\sigma,t+1) &= \lambda \sup_{p\in\mathcal{P}_s^\sigma} \mathbf{E}_{p_N}[\mathcal{E}_N(s';\sigma,t)] + \frac{c(1-\lambda^{t+1})}{1-\lambda} M_N(s,\sigma(s)) \quad \text{(by definition of } \mathcal{E}_N\text{)} \\
&\leq \lambda \sup_{p\in\mathcal{P}_s^\sigma} \mathbf{E}_{p_N}\Big[B_N(s';t) + \frac{\lambda + c(1+\ldots+\lambda^{t-1})}{1-\lambda}\epsilon_N\Big] + \frac{c(1-\lambda^{t+1})}{1-\lambda}(M_N(s)+\epsilon_N) \\
&\leq \lambda \sup_{p\in\mathcal{P}_s^\sigma} \mathbf{E}_{p_N}[B_N(s';t)] + \frac{\lambda^2 + c(\lambda+\ldots+\lambda^{t})}{1-\lambda}\epsilon_N + \frac{c(1-\lambda^{t+1})}{1-\lambda}(M_N(s)+\epsilon_N) \\
&\leq \lambda \overline{B}_N(s,\sigma(s);t) + \lambda\epsilon_N + \frac{c(1-\lambda^{t+1})}{1-\lambda}M_N(s) + \frac{\lambda^2 + c(\lambda+\ldots+\lambda^{t})}{1-\lambda}\epsilon_N + \frac{c}{1-\lambda}\epsilon_N \\
&\leq \lambda \max_{a\in\mathcal{A}} \overline{B}_N(s,a;t) + \frac{c(1-\lambda^{t+1})}{1-\lambda}M_N(s) + \epsilon_N\Big(\lambda + \frac{\lambda^2 + c(\lambda+\ldots+\lambda^{t})}{1-\lambda} + \frac{c}{1-\lambda}\Big) \\
&= B_N(s;2) + \epsilon_N\Big(\frac{\lambda + c(1+\lambda+\ldots+\lambda^{t})}{1-\lambda}\Big).
\end{aligned}$$

Thus, for any t,

$$\mathcal{E}_N(s;\sigma,t) \leq B_N(s;t) + \Big(\frac{\lambda + c(1+\lambda+\ldots+\lambda^{t})}{1-\lambda}\Big)\epsilon_N.$$

Also,

$$\begin{aligned}
\lambda \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}[\mathcal{E}_N(s';\sigma,T)] &\leq \lambda \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}[B_N(s';T)] + \Big(\frac{\lambda^2 + c(\lambda+\ldots+\lambda^{T+1})}{1-\lambda}\Big)\epsilon_N \\
&\leq \lambda \overline{B}_N(s,a;T) + \lambda\epsilon_N + \Big(\frac{\lambda^2 + c(\lambda+\ldots+\lambda^{T+1})}{1-\lambda}\Big)\epsilon_N \\
&= \lambda \overline{B}_N(s,a;T) + \Big(\frac{\lambda + c(\lambda+\ldots+\lambda^{T+1})}{1-\lambda}\Big)\epsilon_N.
\end{aligned}$$

Therefore,

$$\begin{aligned}
\delta(s,a,\sigma,N,T) &= \frac{c}{1-\lambda}\Big(\lambda^T + 2\epsilon_N + M_N(s,a)\Big) + \mathcal{E}_N(s;\sigma,T) + \lambda \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}[\mathcal{E}_N(s';\sigma,T)] \\
&\leq \frac{c}{1-\lambda}\Big(\lambda^T + 3\epsilon_N + \tilde{M}_N(s,a)\Big) + B_N(s;T) + \Big(\frac{\lambda + c(1+\lambda+\ldots+\lambda^T)}{1-\lambda}\Big)\epsilon_N \\
&+ \lambda\overline{B}_N(s,a;T) + \Big(\frac{\lambda + c(\lambda+\ldots+\lambda^{T+1})}{1-\lambda}\Big)\epsilon_N \\
&= \frac{c\lambda^T}{1-\lambda} + \frac{c}{1-\lambda}\tilde{M}_N(s,a) + B_N(s;T) + \lambda\overline{B}_N(s,a;T) + \frac{2\lambda\epsilon_N}{1-\lambda} \\
&+ \frac{c\epsilon_N}{1-\lambda}\Big(3 + 1 + \ldots + \lambda^T + \lambda + \ldots + \lambda^{T+1}\Big) \\
&\leq \frac{c\lambda^T}{1-\lambda} + \frac{c}{1-\lambda}\tilde{M}_N(s,a) + B_N(s;T) + \lambda\overline{B}_N(s,a;T) + \frac{2\lambda\epsilon_N}{1-\lambda} + \frac{4c\epsilon_N}{(1-\lambda)^2} \\
&\triangleq \bar{\delta}(s,a;N,T).
\end{aligned}$$

Now, we show that $\bar{\delta}(s,a;N,T) \to 0$ as $N,T\to\infty$. In order to prove this, it suffices to show that $B_N(s;T)$ vanishes asymptotically. Note that this would imply that $\overline{B}_N(s,a;T)$ also converges to zero.

Fix $s\in\mathcal{S}$. For any $N\geq s$ and T, we have

$$\begin{aligned}
B_N(s;T) &\leq \frac{c}{1-\lambda}M_N(s) + \lambda\epsilon_N + \lambda \max_{a\in\mathcal{A}} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}[B_N(s_1,T-1)] \\
&\leq \frac{c}{1-\lambda}M_N(s) + \lambda\epsilon_N \\
&+ \lambda \max_{a\in\mathcal{A}} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}\Big[\frac{c}{1-\lambda}M_N(s_1) + \lambda\epsilon_N + \lambda \max_{a_1\in\mathcal{A}} \sup_{p^1\in\mathcal{P}_{s_1}^{a_1}} \mathbf{E}_{p_N^1}[B_N(s_2,T-2)]\Big] \\
&\leq \frac{c}{1-\lambda}M_N(s) + \frac{c\lambda}{1-\lambda} \max_{a\in\mathcal{A}} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}[M_N(s_1)] + (\lambda+\lambda^2)\epsilon_N \\
&+ \lambda^2 \max_{a\in\mathcal{A}} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}\Big[\max_{a_1\in\mathcal{A}} \sup_{p^1\in\mathcal{P}_{s_1}^{a_1}} \mathbf{E}_{p_N^1}[B_N(s_2,T-2)]\Big].
\end{aligned}$$

Here, we have used the fact that $\sup_x(f(x)+g(x)) \leq \sup_x f(x) + \sup_x g(x)$ for any functions f and g.

Recursively repeating this argument, we get

$$\begin{aligned}
B_N(s;T) \leq (\lambda + \ldots + \lambda^{T-1})\epsilon_N \\
&+ \frac{c}{1-\lambda}\Bigg\{ M_N(s) + \lambda \max_{a\in\mathcal{A}} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}[M_N(s_1)] + \ldots \\
&+ \lambda^{T-1} \max_{a\in\mathcal{A}} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}\Big[\max_{a_1\in\mathcal{A}} \sup_{p^1\in\mathcal{P}_{s_1}^{a_1}} \Big[\ldots \big[\max_{a_1\in\mathcal{A}} \sup_{p^2\in\mathcal{P}_{s_{T-2}}^{a_{T-2}}} \mathbf{E}_{p_N^{T-2}}[B_N(s_{T-1},1)]\big]\Big]\Big] \Bigg\} \\
&= \frac{\lambda-\lambda^T}{1-\lambda}\epsilon_N + \frac{c}{1-\lambda}M_N(s) + \frac{c\lambda}{1-\lambda} \max_{a\in\mathcal{A}} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}[M_N(s_1)] + \ldots \\
&+ \frac{c\lambda^{T-1}}{1-\lambda} \max_{a\in\mathcal{A}} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}\Big[\max_{a_1\in\mathcal{A}} \sup_{p^1\in\mathcal{P}_{s_1}^{a_1}} \Big[\ldots \big[\max_{a_1\in\mathcal{A}} \sup_{p^2\in\mathcal{P}_{s_{T-2}}^{a_{T-2}}} \mathbf{E}_{p_N^{T-2}}[B_N(s_{T-1},1)]\big]\Big]\Big].
\end{aligned}$$

By choice of ϵ_N and by Assumption 3.5.1 respectively, the first two terms in the above expression converge to 0 as $N \to \infty$. Further, $M_N(s)$ is a nonnegative sequence which vanishes asymptotically and is bounded uniformly over $\mathcal{S}$ by 1. Thus, by Lemma 3.5.6, the third term in the above expression also converges to zero as $N \to \infty$, and is bounded by 1. Repeatedly applying this argument gives us that for a *fixed* T, each term vanishes as N grows. However, we still need to establish the convergence of the double sequence as both N and T grow simultaneously.

Observe that the t+1-st term is bounded above by $c\lambda^{t-1}/(1-\lambda)$. For any $T_0 \in \mathbb{N}$ and

$T > T_0$, we have

$$\begin{aligned}
B_N(s;T) &\leq \frac{\lambda - \lambda^T}{1-\lambda}\epsilon_N \\
&+ \frac{c}{1-\lambda}\Bigg\{ M_N(s) + \lambda \max_{a\in\mathcal{A}} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}[M_N(s_1)] + \ldots \\
&+ \lambda^{T_0-1} \max_{a\in\mathcal{A}} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}\Big[\max_{a_1\in\mathcal{A}} \sup_{p^1\in\mathcal{P}_{s_1}^{a_1}} \Big[\ldots \big[\max_{a_1\in\mathcal{A}} \sup_{p^{T_0-2}\in\mathcal{P}_{s_{T_0-2}}^{a_{T_0-2}}} \mathbf{E}_{p_N^{T_0-2}}[B_N(s_{T_0-1},1)]\big]\Big]\Big] \Bigg\} \\
&+ \frac{c}{1-\lambda}\{\lambda^{T_0} + \ldots + \lambda^{T-1}\} \\
&\leq \frac{\lambda - \lambda^T}{1-\lambda}\epsilon_N \\
&+ \frac{c}{1-\lambda}\Bigg\{ M_N(s) + \lambda \max_{a\in\mathcal{A}} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}[M_N(s_1)] + \ldots \\
&+ \lambda^{T_0-1} \max_{a\in\mathcal{A}} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}\Big[\max_{a_1\in\mathcal{A}} \sup_{p^1\in\mathcal{P}_{s_1}^{a_1}} \Big[\ldots \big[\max_{a_1\in\mathcal{A}} \sup_{p^{T_0-2}\in\mathcal{P}_{s_{T_0-2}}^{a_{T_0-2}}} \mathbf{E}_{p_N^{T_0-2}}[B_N(s_{T_0-1},1)]\big]\Big]\Big] \Bigg\} \\
&+ \frac{c\lambda^{T_0}}{(1-\lambda)^2}.
\end{aligned}$$

We will show that this upper bound on the error term converges to 0 as $N, T \to \infty$.

Given $\eta > 0$, choose T_0 such that $\lambda^{T_0} < \eta$. Also, let $N_0 \in \mathbb{N}$ be such that for $N > N_0$, $\epsilon_N < \eta$, and each of the summations inside the brackets is also less than η. (T_0 is a fixed number given η.) Then, for $N > N_0$ and $T > T_0$,

$$B_N(s;T) \leq \frac{\eta}{1-\lambda} + \frac{c}{1-\lambda}\Big\{\eta + \lambda\eta + \ldots + \lambda^{T_0-1}\eta\Big\} + \frac{c\;\eta}{(1-\lambda)^2} \leq \frac{2c+1}{(1-\lambda)^2}\;\eta.$$

Since $\eta > 0$ was arbitrary, we conclude that the error term can be made arbitrarily small for sufficiently large N and T. Thus, $B_N(s;T) \to 0$ as $N, T \to \infty$.

This completes the proof. □

In Corollary 3.5.5, we found the difference between the true value of a policy and its approximation that the algorithm computes. Our next corollary states that as N and T are increased, that is, more states are included and more iterations of successive approximation are performed in the policy evaluation step, the algorithm asymptotically recovers the true value of the policy. While this result is not used later, it is of independent interest as it establishes that $\hat{v}^{\sigma,N}(\cdot,T)$ actually estimates the true value function.

Corollary 3.5.9. *For any policy σ and state $s \in \mathcal{S}$, $\hat{v}^{\sigma,N}(s;T) \to v^{\sigma}(s)$ as $N,T \to \infty$.*

Proof. As shown in the proof of Lemma 3.5.8, we have

$$\mathcal{E}_N(s;\sigma,T) \leq B_N(s;T) + \frac{\lambda + c(1+\ldots+\lambda^T)}{1-\lambda}\epsilon_N.$$

Since the right hand of the above inequality converges to zero, it follows that so does $\mathcal{E}_N(s;\sigma,T)$. This, combined with Corollary 3.5.5, completes the proof. □

In Section 3.4, we noted that each step of the algorithm requires finite computation and memory. At first glance, however, it appears that Step 2 of the algorithm may get caught in an infinite loop if condition (3.13) is not satisfied for any N and T. Our next lemma proves that this loop must terminate if the current policy is not optimal. For notational convenience, we will denote $\gamma^{\sigma^k}(\cdot,\cdot)$ and $\gamma^{\sigma^k;N}(\cdot,\cdot)$ by $\gamma^k(\cdot,\cdot)$ and $\gamma^{k;N}(\cdot,\cdot)$ respectively.

Lemma 3.5.10. *In iteration k, Step 2 of the algorithm terminates finitely if and only if the policy σ^k is not optimal.*

Proof. Suppose a policy σ^k is not optimal. Then, by Lemma 3.5.2, there is a state-action pair $(\bar{s},\bar{a})$ for which $-\epsilon = \gamma^k(\bar{s},\bar{a}) < 0$. Since $\bar{\delta}(\bar{s},\bar{a},N,T) \to 0$ as $N,T \to \infty$, there exist integers $N_1 \geq \bar{s}$ and T_1 such that for all $T \geq T_1$ and $N \geq N_1$, we have $\bar{\delta}(\bar{s},\bar{a},N,T) < \epsilon/2$. This implies that

$$\gamma^{k,N}(\bar{s},\bar{a};T) \leq \gamma^k(\bar{s},\bar{a}) + \bar{\delta}(\bar{s},\bar{a},N,T) = -\epsilon + \bar{\delta}(\bar{s},\bar{a},N,T) < -\epsilon + \frac{\epsilon}{2} = -\frac{\epsilon}{2}. \tag{3.30}$$

Now, for any N and T, and any state-action pair (s, a) with $s \leq N$, we have that

$$\begin{aligned} \left|\gamma^{k,N}(s,a;T)\right| &\leq \left|\Big(\sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}[c(s,a,s') + \lambda v^{k;N}(s';T)]\Big) - v^{k;N}(s;T)\right| + \epsilon_N \\ &\leq \left|\sup_{p\in\mathcal{P}_s^a} \mathbf{E}_{p_N}[c(s,a,s') + \lambda v^{k;N}(s';T)]\right| + \left|v^{k;N}(s;T)\right| + \epsilon_N \\ &\leq c + \frac{c\lambda}{1-\lambda} + \frac{c}{1-\lambda} + \epsilon_N = \frac{2c}{1-\lambda} + \epsilon_N \\ \implies -\gamma^{k,N}(s,a;T) &\leq \frac{2c}{1-\lambda} + \epsilon_N. \end{aligned} \tag{3.31}$$

Since $\sum\limits_{s\in\mathcal{S}} \beta(s) < \infty$, it follows that $\beta(s) \to 0$ as $s \to \infty$. Also, ϵ_N approaches zero as $N \to \infty$. Thus, there exists an $s_1 \geq \bar{s}$ and an integer $N_2 \geq \bar{s}$ such that for all $s \geq s_1$ and $N \geq N_2$,

$$\beta(s)\left(\frac{2c}{1-\lambda} + \epsilon_N\right) < \frac{\epsilon}{2}\beta(\bar{s}). \tag{3.32}$$

Then, it follows from Equations (3.30), (3.31) and (3.32) that for all $N \geq \max\{s_1, N_1, N_2\}$, $T \geq T_1$ and $s_1 \leq s \leq N$,

$$\begin{aligned} -\beta(s)\gamma^{k,N}(s,a;T) \;&\leq\; \beta(s)\left(\frac{2c}{1-\lambda} + \epsilon_N\right) \;<\; \frac{\epsilon}{2}\beta(\bar{s}) \\ \implies \beta(s)\gamma^{k,N}(s,a;T) \;&>\; -\frac{\epsilon}{2}\beta(\bar{s}) \;\geq\; \beta(\bar{s})\gamma^{k,N}(\bar{s},\bar{a};T). \end{aligned}$$

Thus, the maximum improvement in Step 2(d) of the algorithm must occur in a state $s < s_1$, and we have,

$$\begin{aligned} \beta(s^{k,N}(T))\gamma^{k,N}(T) &= \min_{s\leq N,\ a\in\mathcal{A}} \beta(s)\gamma^{k,N}(s,a;T) = \min_{s\leq s_1,\ a\in\mathcal{A}} \beta(s)\gamma^{k,N}(s,a;T) \\ &\leq \beta(\bar{s})\gamma^{k,N}(\bar{s},\bar{a};T) \\ \implies \gamma^{k,N}(T) &\leq \frac{\beta(\bar{s})}{B}\gamma^{k,N}(\bar{s},\bar{a};T), \end{aligned}$$

where $B = \min\{\beta(s) : s < s_1\}$.

Now, let N_3, T_2 be such that for all $N \geq N_3$ and $T \geq T_2$, $\bar{\delta}(\bar{s},\bar{a},N,T) < \epsilon\,\beta(\bar{s})/(B+\beta(\bar{s}))$.

Then, it follows that for $N \geq \max\{s_1, N_1, N_2, N_3\}$ and $T \geq \max\{T_1, T_2\}$,

$$\begin{aligned}
\gamma^{k,N}(T) &\leq \frac{\beta(\bar{s})}{B}\gamma^{k,N}(\bar{s}, \bar{a}; T) \leq \frac{\beta(\bar{s})}{B}\Bigg\{\gamma^k(\bar{s}, \bar{a}) + \bar{\delta}(\bar{s}, \bar{a}, N, T)\Bigg\} \\
&= \frac{\beta(\bar{s})}{B}\Bigg\{ - \epsilon + \bar{\delta}(\bar{s}, \bar{a}, N, T)\Bigg\} \\
&< \frac{\beta(\bar{s})}{B}\Bigg\{ - \epsilon + \frac{\beta(\bar{s})}{B + \beta(\bar{s})}\epsilon\Bigg\} = \frac{\epsilon\ \beta(\bar{s})}{B}\Bigg\{ - 1 + \frac{\beta(\bar{s})}{B + \beta(\bar{s})}\Bigg\} \\
&= -\frac{\epsilon\ \beta(\bar{s})}{B + \beta(\bar{s})} < -\bar{\delta}(\bar{s}, \bar{a}, N, T).
\end{aligned}$$

Thus, for all $N \geq \max\{s_1, N_1, N_2, N_3\}$ and $T \geq \max\{T_1, T_2\}$, condition (3.13) is satisfied, and Step 2 of the algorithm terminates finitely.

Conversely, suppose Step 2 of the algorithm terminates for some $N = N(k)$, $T = T(k)$. Then,

$$\begin{aligned}
&\gamma^{k,N(k)}(T(k)) = \gamma^{k,N(k)}(s^k, a^k; T(k)) < -\bar{\delta}(s^k, a^k, N(k), T(k)) \\
\implies &\gamma^k(s^k, a^k) \leq \gamma^{k,N(k)}(s^k, a^k; T(k)) + \bar{\delta}(s^k, a^k, N(k), T(k)) < 0.
\end{aligned}$$

This implies, by Lemma 3.5.2, that the policy σ^k is not optimal. □

We point out here that if the algorithm does find an optimal policy σ^k in some iteration, then the inner loop in Step 2 of the algorithm does not terminate. As such, our algorithm cannot tell if it has indeed discovered an optimal policy. This, however, is an inherent feature of countable-state MDPs and not just a limitation of our algorithm. In particular, given a policy σ, it is not possible to check with finite computations if the policy is optimal, and this subtle issue persists as in the previous chapter and in [22, 30]. Note also that if σ^k is optimal, the algorithm does not proceed further and thus only generates a finite number of policies. In that case, for notational convenience, we interpret that the sequence of policies σ^t is still infinite, with $\sigma^t \equiv \sigma^k$ for all $t \geq k$.

We now proceed to study the values of the sequence of policies generated by the algorithm. The following sequence of results proves that the policies generated by our algorithm are strictly non-increasing in value. Unlike similar results in the previous chapter and [22, 30], we are no longer able to prove this result by examining the difference of the subsequent value functions. Instead, we use the properties of the robust evaluation operator. Recall that V was defined as the space of all bounded functions on $\mathcal{S}$, and it is a Banach space in the supremum norm. The evaluation operator $\mathcal{L}^\sigma$ for a policy σ was defined as

$$\mathcal{L}^\sigma(u)(s) = \sup_{p \in \mathcal{P}_s^\sigma} \mathbf{E}_p[c(s, \sigma(s), s') + \lambda u(s')].$$

$\mathcal{L}^\sigma$ is clearly a monotone operator, i.e., $u \leq v \implies \mathcal{L}^\sigma(u) \leq \mathcal{L}^\sigma(v)$. The following lemma is a simple consequence of Theorem 3 in [28], and we include it here for completeness. The proof is very similar to that of the original theorem and is omitted.

Lemma 3.5.11. *For any fixed stationary policy σ, its evaluation operator $\mathcal{L}^\sigma$ is a contraction mapping on V.*

Since the value function v^k is the fixed point of the operator $\mathcal{L}^{\sigma^k}$, it follows that for any $u \in V$, $(\mathcal{L}^{\sigma^k})^n u \to v^k$ uniformly as $n \to \infty$. Now, suppose a policy σ is not optimal. Let μ be a new policy obtained by updating σ in a single state with an action which gives strict reduction in cost. The following lemma shows that μ must be strictly better than σ in value. We point out that the result seems intuitively true, but the proof is not straightforward due to the implicit, nonlinear nature of the robust Bellman equations.

Lemma 3.5.12. *Let σ and μ be two stationary policies which satisfy the following: $\gamma^\sigma(\bar{s}, \bar{a}) < 0$, and $\mu(\bar{s}) = \bar{a} \neq \sigma(\bar{s})$; $\mu(s) = \sigma(s)$ for all $s \neq \bar{s}$. Then, $v^\mu(s) \leq v^\sigma(s)$ for all $s \in \mathcal{S}$, and $v^\mu(\bar{s}) \leq v^\sigma(\bar{s}) + \gamma^\sigma(\bar{s}, \bar{a}) < v^\sigma(\bar{s})$.*

Proof. Recall from Equation (3.16) that the improvement is defined as

$$\gamma^\sigma(\bar{s}, \bar{a}) = \sup_{p \in \mathcal{P}_{\bar{s}}^{\bar{a}}} \Big(\mathbf{E}_p[c(\bar{s}, \bar{a}, s') + \lambda v^\sigma(s')] \Big) - v^\sigma(\bar{s}) < 0.$$

Let $v^0 = v^\sigma$ and define functions $v^n = \mathcal{L}^\mu(v^{n-1}) = (\mathcal{L}^\mu)^n(v^\sigma)$ for $n = 1, 2, \ldots$. Then, since $\mu(\bar{s}) = \bar{a}$, we have

$$v^1(\bar{s}) = \mathcal{L}^\mu(v^\sigma)(\bar{s}) = \sup_{p \in \mathcal{P}_{\bar{s}}^{\bar{a}}} \Big(\mathbf{E}_p[c(\bar{s}, \bar{a}, s') + \lambda v^\sigma(s')] \Big) = v^\sigma(\bar{s}) + \gamma^\sigma(\bar{s}, \bar{a}) < v^\sigma(\bar{s}).$$

Further, for $s \neq \bar{s}$, $\mu(s) = \sigma(s)$. So we have

$$v^1(s) = \sup_{p \in \mathcal{P}_s^\mu} \Big(\mathbf{E}_p[c(s, \mu(s), s') + \lambda v^\sigma(s')] \Big) = \sup_{p \in \mathcal{P}_s^\sigma} \Big(\mathbf{E}_p[c(s, \sigma(s), s') + \lambda v^\sigma(s')] \Big) = v^\sigma(s).$$

Thus, $v^1 \leq v^\sigma$. Since $\mathcal{L}^\mu$ is a monotone operator, we have that $v^2 = \mathcal{L}^\mu(v^1) \leq \mathcal{L}^\mu(v^\sigma) = v^1 \leq v^\sigma$. Repeating this process gives that $v^n = \mathcal{L}^\mu(v^{n-1}) = (\mathcal{L}^\mu)^n(v^\sigma) \leq v^\sigma$ for all n. Then, taking limits as $n \to \infty$, we have that $v^\mu \leq v^\sigma$.

By the same argument, we also have $v^\mu(s) \leq v^1(s)$ for all $s \in \mathcal{S}$. In particular, $v^\mu(\bar{s}) \leq v^1(\bar{s}) = v^\sigma(\bar{s}) + \gamma^\sigma(\bar{s}, \bar{a}) < v^\sigma(\bar{s})$. This completes the proof. □

Our algorithm updates the policies in a similar manner as described in the previous lemma, except that it uses the approximate improvement to determine an improving state-action pair in each iteration. Even so, the adaptive choice of N and T ensures that the true values of the policies generated by the algorithm improve in each iteration. The following corollary establishes this.

Corollary 3.5.13. *If a policy σ^k is not optimal, then $v^{k+1}(s) \leq v^k(s)$ for all states $s \in \mathcal{S}$, with $v^{k+1}(s^k) \leq v^k(s^k) + \gamma^k(s^k, a^k) < v^k(s^k)$.*

Proof. Suppose a policy σ^k is not optimal. Then, by Lemma 3.5.10, Step 2 of the algorithm terminates finitely for some $N = N(k)$, $T = T(k)$ and

$$\gamma^k(s^k, a^k) \leq \gamma^{k,N(k)}(s^k, a^k; T(k)) + \bar{\delta}(s^k, a^k, N(k), T(k)) < 0.$$

Also, $\sigma^{k+1}(s^k) = a^k$ and $\sigma^{k+1}(s) = \sigma^k(s)$ for all $s \neq s^k$. Thus, by Lemma 3.5.12, $v^{k+1}(s) \leq v^k(s)$ for all $s \in \mathcal{S}$, and $v^{k+1}(s^k) \leq v^k(s^k) + \gamma^k(s^k, a^k) < v^k(s^k)$. □

The policy update scheme ensures that every non-optimal policy generated by the algorithm is strictly better than the previous one. The following result states that once a policy has been found to be non-optimal, the algorithm never generates it again.

Corollary 3.5.14. *The algorithm does not repeat any non-optimal policy σ^k.*

Proof. Let σ^k be any non-optimal policy. Suppose there exists $j > k$ such that $\sigma^j \equiv \sigma^k$. Then, $v^k \equiv v^j$. But from Corollary 3.5.13, we know that

$$v^j \leq v^{j-1} \leq \ldots \leq v^k.$$

Then all the inequalities above must be equalities, which is a contradiction to Corollary 3.5.13, since v^k and v^{k+1} differ strictly in at least one state as σ^k is not optimal. Hence, no two non-optimal policies generated by the algorithm can be identical. □

Recall that we are minimizing $\sum_{s \in \mathcal{S}} \beta(s)v(s)$. In exact policy iteration, iteration k would update the policy in some state $\bar{s}^k$ with an action $\bar{a}^k$ which gives the largest improvement in the weighted value function $\beta(\cdot)v^k(\cdot)$ across all state-action pairs. As the algorithm goes on, the value functions would approach optimality leaving less and less room for improvement, and the amount of cost-reduction $\beta(\bar{s}^k)\gamma^k(\bar{s}^k, \bar{a}^k)$ would shrink to zero. Our algorithm also looks for the largest improvement, but only among states $s \leq N(k)$ and uses the approximate improvement function $\hat{\gamma}^{k,N(k)}(\cdot, \cdot; T(k))$ to do so. As such, our update may not be the best possible across all states in $\mathcal{S}$. Nonetheless, the true weighted improvement for (s^k, a^k) still vanishes asymptotically. This is established in the next lemma.

Lemma 3.5.15. *The weighted improvement $\beta(s^k)\gamma^k(s^k, a^k) \to 0$ as $k \to \infty$.*

Proof. Define $f^k = \sum_{s \in \mathcal{S}} \beta(s)v^k(s)$. Then, $0 \leq f^k < \infty$ for all k. By Corollary 3.5.13, we have that $f^{k+1} \leq f^k + \beta(s^k)\gamma^k(s^k, a^k) \leq f^k$ for all k. This implies that the sequence f^k converges. Further, $f^{k+1} - f^k \leq \beta(s^k)\gamma^k(s^k, a^k) < 0$ for all k. Taking limits as $k \to \infty$ gives that $\beta(s^k)\gamma^k(s^k, a^k) \to 0$. □

In iteration k, our approximate algorithm includes the first $N(k)$ states from $\mathcal{S}$ and performs $T(k)$ steps of successive approximation. Our next lemma establishes two things. First, $N(k) \to \infty$ as $k \to \infty$. That is, the algorithm asymptotically includes all of $\mathcal{S}$. Second, $T(k) \to \infty$ as $k \to \infty$, which implies that the approximate policy evaluation step becomes exact at infinity. We also point out that this is the only place where we use the fact that the algorithm always sets N and T equal to each other.

Lemma 3.5.16. *$N(k) \to \infty$ and $T(k) \to \infty$ as $k \to \infty$.*

Proof. We will first prove that $N(k) \to \infty$, and the idea of the proof is similar to Lemma 3.5.7 in [22].

The lemma holds trivially if σ^k is optimal for any k; hence we assume that this is not the case. So the algorithm produces an infinite sequence of distinct policies σ^k.

Now, first suppose that $N(k) \not\to \infty$ as $k \to \infty$. Then, there exists an integer M such that $N(k) = M$ for infinitely many k. In particular, let r_k be a subsequence such that $N(r_k) = M$ for all k. Then, the policies σ^{r_k} differ only in states $s \leq M$. Since $\mathcal{A}$ is finite, there are only finitely many distinct policies of this kind. In particular, this implies that there are policies $\sigma^{r_k} \equiv \sigma^{r_j}$ for $r_k \neq r_j$. This is a contradiction, as we have from Corollary 3.5.14 that the algorithm does not repeat any non-optimal policy. Hence, we must have $N(k) \to \infty$.

Finally, since our algorithm varies N and T so that they are always equal, we conclude that as $k \to \infty$, $T(k) \to \infty$ as well. This completes the proof. □

Finally, we proceed to prove our main theorem, which establishes that the value functions of the sequence of policies generated by our algorithm converge to the optimal value in the β-norm.

Theorem 3.5.17. *Let $v^*(\cdot)$ denote the optimal value function of the robust countable-state MDP. Then,*

$$\|v^k\|_\beta \to \|v^*\|_\beta \quad \text{as} \quad k \to \infty.$$

Proof. The theorem is trivially true if σ^k is optimal for some k. So let us assume that this is not the case. Then, from Corollary 3.5.14, we know that the algorithm generates an infinite sequence of distinct policies.

We first claim that there exists a subsequence r_j such that $s^{r_j} \to \infty$ as $j \to \infty$. We will prove this by contradiction. Suppose there exists an N such that $s^k \leq N$ for all k. Then, the algorithm updates the policy only in states $s \leq N$ and keeps all actions in states $s > N$ fixed. Since $\mathcal{A}$ is finite, there are only finitely many distinct policies that the algorithm can find, which is a contradiction.

Now, let $\mathcal{F}$ denote the set of all stationary policies, i.e., $\mathcal{F} = \prod_{s \in \mathcal{S}} A$. The product topology on $\mathcal{F}$ is metrizable, and let ρ denote a corresponding metric on $\mathcal{F}$. Then, $\mathcal{F}$ is sequentially compact with respect to ρ by Tychonoff's theorem. Therefore, the sequence σ^{r_j} has a convergent subsequence σ^{t_j}. Denote the limit of this sequence by $\tilde{\sigma}$. Further, the value functions v^{t_j} lie in the set $V = \{v \in V : 0 \leq v(s) \leq c/(1-\lambda) \text{ for all } s \in \mathcal{S}\}$, which is also compact in the product topology by Tychonoff's theorem. Thus, there is a convergent subsequence v^{u_j} of v^{t_j}. Let $\tilde{v}$ be the limit of this sequence. Note that σ^{u_j} also converges to $\tilde{\sigma}$.

We now show that $\tilde{v}$ is the value of $\tilde{\sigma}$. Fix a state $s \in \mathcal{S}$. For any j,

$$v^{u_j}(s) - \sup_{p \in \mathcal{P}_s^{\sigma^{u_j}}} \mathbf{E}_p[c(s, \sigma^{u_j}(s), s') + \lambda v^{u_j}(s')] = 0. \tag{3.33}$$

Convergence in the product topology gives us that $\sigma^{u_j}(s) \to \tilde{\sigma}(s)$ in $\mathcal{A}$ as $j \to \infty$. Since $\mathcal{A}$ is finite, there exists a number $J_1(s)$ and an action $a(s)$ such that for all $j \geq J_1(s)$, $\sigma^{u_j}(s) = a(s)$. Then, for $j \geq J_1(s)$, we rewrite (3.33) as

$$v^{u_j}(s) - \sup_{p \in \mathcal{P}_s^{a(s)}} \mathbf{E}_p[c(s, a(s), s') + \lambda v^{u_j}(s')] = 0. \tag{3.34}$$

Let A be defined as

$$A = \tilde{v}(s) - \sup_{p \in \mathcal{P}_s^{a(s)}} \mathbf{E}_p[c(s, a(s), s') + \lambda \tilde{v}(s')]. \tag{3.35}$$

We must show that $A = 0$. For any fixed $p \in \mathcal{P}_s^{a(s)}$ and $j \geq J_1(s)$,

$$\begin{aligned} v^{u_j}(s) &= \sup_{p \in \mathcal{P}_s^{a(s)}} \mathbf{E}_p[c(s, a(s), s') + \lambda v^{u_j}(s')] \\ &\geq \mathbf{E}_p[c(s, a(s), s') + \lambda v^{u_j}(s')] \\ \implies \lim_{j \to \infty} v^{u_j}(s) &\geq \lim_{j \to \infty} \mathbf{E}_p[c(s, a(s), s') + \lambda v^{u_j}(s')], \\ \implies \tilde{v}(s) &\geq \mathbf{E}_p[c(s, a(s), s') + \lambda \tilde{v}(s')], \end{aligned}$$

where we have used the Dominated Convergence theorem on the right hand side of the last inequality. Since the above is true for all $p \in \mathcal{P}_s^{a(s)}$, it follows that

$$\tilde{v}(s) \geq \sup_{p \in \mathcal{P}_s^{a(s)}} \mathbf{E}_p[c(s, a(s), s') + \lambda \tilde{v}(s')] \implies A \geq 0.$$

We need to further prove that this inequality cannot be strict. For this, consider an arbitrary $\epsilon > 0$. For $j \geq J_1(s)$,

$$A = \tilde{v}(s) - \sup_{p \in \mathcal{P}_s^{a(s)}} \mathbf{E}_p[c(s, a(s), s') + \lambda \tilde{v}(s')] - v^{u_j}(s) + \sup_{p \in \mathcal{P}_s^{a(s)}} \mathbf{E}_p[c(s, a(s), s') + \lambda v^{u_j}(s')].$$

For each j, let $p^j \in \mathcal{P}_s^{a(s)}$ be such that

$$\sup_{p \in \mathcal{P}_s^{a(s)}} \mathbf{E}_p[c(s, a(s), s') + \lambda v^{u_j}(s')] < \mathbf{E}_{p^j}[c(s, a(s), s') + \lambda v^{u_j}(s')] + \epsilon.$$

Then,

$$\begin{aligned}A &\leq \tilde{v}(s) - \mathbf{E}_{p^j}[c(s, a(s), s') + \lambda\tilde{v}(s')] - v^{u_j}(s) + \mathbf{E}_{p^j}[c(s, a(s), s') + \lambda v^{u_j}(s')] + \epsilon \\ &= \tilde{v}(s) - v^{u_j}(s) + \lambda\mathbf{E}_{p^j}[v^{u_j}(s') - \tilde{v}(s')] + \epsilon.\end{aligned}$$

For any $n \in \mathbb{N}$, we can write

$$\begin{aligned}A &\leq \tilde{v}(s) - v^{u_j}(s) + \epsilon + \lambda\mathbf{E}_{p^j_n}[v^{u_j}(s') - \tilde{v}(s')] + \lambda\mathbf{E}_{\overline{p^j}_n}[v^{u_j}(s') - \tilde{v}(s')] \\ &\leq \tilde{v}(s) - v^{u_j}(s) + \epsilon + \lambda\mathbf{E}_{p^j_n}[v^{u_j}(s') - \tilde{v}(s')] + \frac{2c\lambda}{1-\lambda}M_n(s, a(s)).\end{aligned}$$

Since $\mathcal{P}_s^{a(s)}$ satisfies Assumption 3.5.1, there exists an integer $n_0 \geq s$ such that $M_{n_0}(s, a(s)) < \epsilon$. Further, once n_0 is fixed, we know that $v^{u_j}(\cdot)$ converges to $\tilde{v}(\cdot)$ uniformly on $\{1, 2, \ldots, n_0\} \subset \mathcal{S}$. Then, choose an integer $J_2(s) \geq J_1(s)$ such that $|v^{u_j}(s') - \tilde{v}(s')| < \epsilon$ for all $j \geq J_2(s)$ for all $s' \in \{1, 2, \ldots, n_0\}$. Thus, for $j \geq J_2(s)$,

$$A \leq \epsilon + \epsilon + \lambda\mathbf{E}_{p^j_{n_0}}[\epsilon] + \frac{2c\lambda}{1-\lambda}\epsilon \leq \epsilon + \epsilon + \lambda\epsilon + \frac{2c\lambda\epsilon}{1-\lambda} = \epsilon\left(2 + \lambda + \frac{2c\lambda}{1-\lambda}\right).$$

Since $\epsilon > 0$ was arbitrary and A was already shown to be nonnegative, we conclude that $A = 0$. Thus, the limiting value function $\tilde{v}(\cdot)$ is in fact the value of the limiting policy $\tilde{\sigma}$.

We now show by contradiction that the limiting policy must be optimal. Suppose $\tilde{\sigma}$ is not optimal. Then, by Lemma 3.5.2, there exists a state-action pair (s, a) such that

$$0 < \epsilon \ = \ \tilde{v}(s) - \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_p[c(s, a, s') + \lambda\tilde{v}(s')].$$

Again, for any j, let $p^j(\cdot|s, a) \in \mathcal{P}_s^a$ be such that

$$\sup_{p \in \mathcal{P}_s^a} \mathbf{E}_p[c(s, a, s') + \lambda v^{u_j}(s')] < \mathbf{E}_{p^j}[c(s, a, s') + \lambda v^{u_j}(s')] + \epsilon/5.$$

Then,

$$\begin{aligned}
&\epsilon - v^{u_j}(s) + \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_p[c(s,a,s') + \lambda v^{u_j}(s')] \\
&= \tilde{v}(s) - \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_p[c(s,a,s') + \lambda \tilde{v}(s')] - v^{u_j}(s) + \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_p[c(s,a,s') + \lambda v^{u_j}(s')] \\
&\leq \tilde{v}(s) - \mathbf{E}_{p^j}[c(s,a,s') + \lambda \tilde{v}(s')] - v^{u_j}(s) + \mathbf{E}_{p^j}[c(s,a,s') + \lambda v^{u_j}(s')] + \epsilon/5 \\
&= (\tilde{v}(s) - v^{u_j}(s)) + \lambda \mathbf{E}_{p^j}[v^{u_j}(s') - \tilde{v}(s')] + \epsilon/5 \\
&= (\tilde{v}(s) - v^{u_j}(s)) + \lambda \mathbf{E}_{p_n^j}[v^{u_j}(s') - \tilde{v}(s')] + \lambda \mathbf{E}_{\overline{p^j}_n}[v^{u_j}(s') - \tilde{v}(s')] + \epsilon/5 \\
&\leq (\tilde{v}(s) - v^{u_j}(s)) + \lambda \mathbf{E}_{p_n^j}[v^{u_j}(s') - \tilde{v}(s')] + \frac{2c\lambda}{1-\lambda} \mathbf{E}_{\overline{p^j}_n}[1] + \epsilon/5 \\
&\leq (\tilde{v}(s) - v^{u_j}(s)) + \lambda \mathbf{E}_{p_n^j}[v^{u_j}(s') - \tilde{v}(s')] + \frac{2c\lambda}{1-\lambda} M_n(s,a) + \epsilon/5 \qquad (3.36)
\end{aligned}$$

for any integer n. Choose $n \geq s$ so that $2c\lambda M_n(s,a)/(1-\lambda) < \epsilon/5$. Choose $J_3(s)$ so that $|v^{u_j}(s') - v(s')| < \epsilon/5$ for all $j \geq J_3(s)$ and for all $s' \leq n$. For such j, the expression (3.36) can be bounded above by $4\epsilon/5$. Thus,

$$\epsilon - v^{u_j}(s) + \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_p[c(s,a,s') + \lambda v^{u_j}(s')] \leq 4\epsilon/5 \implies \epsilon/5 \leq v^{u_j}(s) - \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_p[c(s,a,s') + \lambda v^{u_j}(s')].$$

By Lemma 3.5.16, $N(u_j)$ can be made arbitrarily large as $j \to \infty$. In particular, there exists $J_4(s) \geq J_3(s)$ such that for all $j \geq J_4(s)$, we must have $N(u_j) \geq s$. Then,

$$\begin{aligned}
\epsilon/5 &\leq v^{u_j}(s) - \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_p[c(s,a,s') + \lambda v^{u_j}(s')] \\
&= -\gamma^{u_j}(s,a) \\
&\leq -\gamma^{u_j, N(u_j)}(s,a,T(u_j)) + \bar{\delta}(s,a,N(u_j),T(u_j)) \quad \forall\, j \geq J_4(s). \qquad (3.37)
\end{aligned}$$

Since $N(u_j), T(u_j) \to \infty$ as $j \to \infty$, the second term vanishes asymptotically by Lemma 3.5.8. We show that the limit-superior of the first term must also be non-positive.

Recall that in iteration k, the policy is updated in state s^k by choosing action a^k to give the largest weighted improvement. Thus, $\beta(s^{u_j})\gamma^{u_j,N(u_j)}(s^{u_j}, a^{u_j}, T(u_j)) \leq \beta(s)\gamma^{u_j,N(u_j)}(s, a, T(u_j))$. Therefore,

$$\begin{aligned}-\beta(s)\gamma^{u_j,N(u_j)}(s, a, T(u_j)) &\leq -\beta(s^{u_j})\gamma^{u_j,N(u_j)}(s^{u_j}, a^{u_j}, T(u_j)) \\ &\leq -\beta(s^{u_j})\gamma^{u_j}(s^{u_j}, a^{u_j}) + \beta(s^{u_j})\bar{\delta}(s^{u_j}, a^{u_j}, N(u_j), T(u_j)).\end{aligned}$$

The two terms asymptotically vanish by Lemmas 3.5.15 and 3.5.8 respectively. Thus, the sequence $-\beta(s)\gamma^{u_j,N(u_j)}(s^{u_j}, a^{u_j}, T(u_j))$ is dominated by a sequence which converges to zero. Since $\beta(s) > 0$, we conclude that $\limsup_{j\to\infty}(-\gamma^{w_j,N(w_j)}(s^{w_j}, a^{w_j}, T(w_j))) \leq 0$.

This yields a contradiction to (3.37), and we conclude that the policy $\tilde{\sigma}$ must be optimal, and its value $\tilde{v}$ is the optimal value function, that is, $\tilde{v} = v^*$.

Note that so far we have only established point-wise subsequential convergence of the value functions. However, for each state $s \in \mathcal{S}$, we have from Lemma 3.5.12 that the sequence $v^k(s)$ is a monotonically decreasing non-negative sequence of real numbers; hence it must be convergent. This proves that $v^k(s) \to v^*(s)$ as $k \to \infty$ for every $s \in \mathcal{S}$. Finally, we invoke the Dominated Convergence Theorem once again to conclude that

$$\sum_{s\in\mathcal{S}} \beta(s)v^k(s) \to \sum_{s\in\mathcal{S}} \beta(s)v^*(s) \text{ i.e. } \|v^k\|_\beta \to \|v^*\|_\beta \text{ as } k \to \infty.$$

This completes the proof. □

Our next result proves that the sequence of policies generated by the algorithm reaches arbitrarily close to an optimal policy as $k \to \infty$. The proof is identical to [22], but we still include it here for completeness.

Theorem 3.5.18 (Policy Convergence). *For any $\epsilon > 0$, there exists an iteration counter k_ϵ such that $\rho(\sigma^k, \sigma^{k*}) < \epsilon$ for some optimal policy σ^{k*}, for all $k \geq k_\epsilon$. In fact, if the MDP has a unique optimal policy σ^*, then $\lim_{k\to\infty} \sigma^k = \sigma^*$. Further, for every period n, there exists an iteration counter K_n such that for all $k \geq K_n$, actions $\sigma^k(s)$ are optimal for all states $s \leq n$.*

Proof. We prove the first claim by contradiction. Suppose this is not true. Then, there exists an $\epsilon > 0$ and a subsequence σ^{u_k} of σ^k such that $\rho(\sigma^{u_k}, \sigma) > \epsilon$ for all optimal policies σ, for all $k \in \mathbb{N}$. Since the space of all policies $\mathcal{F}$ is compact, the sequence σ^{u_k} has a convergent subsequence σ^{t_k}, whose limit is, say, $\tilde{\sigma}$. Then, there exists an integer K such that $\rho(\sigma^{t_k}, \tilde{\sigma}) < \epsilon$ for all $k \geq K$. Further, as in the proof of Theorem 3.5.17, $\tilde{\sigma}$ must be an optimal policy. This leads to a contradiction. Hence, the first claim is true.

Further, suppose that σ^* is the unique optimal policy. Then as shown already, for every $\epsilon > 0$, there exists an integer k_ϵ, such that $\rho(\sigma^k, \sigma^*) < \epsilon$ for all $k \geq k_\epsilon$. This implies that $\lim\limits_{k\to\infty} \sigma^k = \sigma^*$.

Now, for the third claim, we note that the result is trivially true if σ^k is optimal for some k. When this is not the case, we first claim that given $\epsilon > 0$ and any state n, there exists an iteration counter K_n such that for all $k \geq K_n$, $|\sigma^k(s) - \sigma^{k*}(s)| < \epsilon$, for all $s \leq n$, for some optimal policy σ^{k*}. Suppose this is not true. Then, there exists a subsequence u_k, and for each k, a state $s_k \leq n$ such that $|\sigma^{u_k}(s_k) - \sigma^*(s_k)| \geq \epsilon$ for all k, for all optimal policies σ^*. But u_k has a further subsequence t_k such that σ^{t_k} converges to an optimal policy $\tilde{\sigma}$ as in the proof of Theorem 3.5.17. This leads to a contradiction.

Now, fix $0 < \epsilon < 1$ and a state n, and consider any iteration $k \geq K_n$. Fix a state $s \leq n$. Then, $|\sigma^k(s) - \sigma^{k*}(s)| < \epsilon$ for some optimal action $\sigma^{k^*}(s)$. Since $\epsilon < 1$ and $\sigma^k(s), \sigma^{k*}(s) \in \mathcal{A} = \{1, 2, \ldots, A\}$, it follows that $\sigma^k(s) = \sigma^{k*}(s)$. This proves that all actions up to state n are optimal for policies σ^k with $k \geq K_n$. □

3.6 Examples

In Section 3.5, we briefly discussed the choice of uncertainty sets and the properties they must satisfy in order that the proposed algorithm be implementable and convergent. In this section, we provide some examples that fall within our robust MDP framework. We explain the intuition behind how appropriate uncertainty sets may be chosen, and demonstrate that these sets naturally have the desirable properties.

3.6.1 Interval uncertainty

In the first example, we show that our proposed method can be used to solve any robust MDP with interval uncertainty sets for the transition probabilities. These sets are commonly used in the robust optimization literature; see Chapter 14 of [9] for details.

Consider an instance where a decision-making agent obtains statistical estimates of each component of the transition probabilities. Confidence bounds on these estimates lead to the formulation of interval uncertainty sets, where each component $p(s'|s,a)$ of the pmf is known to lie between some bounds $l_s^a(s')$ and $u_s^a(s')$. More precisely,

$$\mathcal{P}_s^a = \{p(\cdot) \in \mathcal{M}(\mathcal{S}) : l_s^a(s') \leq p(s'|s,a) \leq u_s^a(s') \ \forall \ s' \in \mathcal{S}\}. \tag{3.38}$$

Without loss of generality, let $0 \leq l_s^a(s') \leq u_s^a(s') \leq 1$ for all $s' \in \mathcal{S}$. For $\mathcal{P}_s^a$ to be non-empty, we must also have $\sum\limits_{s' \in \mathcal{S}} l_s^a(s) \leq 1 \leq \sum\limits_{s' \in \mathcal{S}} u_s^a(s)$. In fact, we can assume that these inequalities are strict so that $\mathcal{P}_s^a$ is not a singleton. Suppose, in addition, that $\sum\limits_{s' \in \mathcal{S}} u_s^a(s') < \infty$. This ensures that the uncertainty sets satisfy Assumption 3.5.1, since

$$M_N(s,a) = \sup_{p \in \mathcal{P}_s^a} \sum_{s' > N} p(s'|s,a) \leq \sum_{s' > N} u_s^a(s') \to 0 \quad \text{as} \quad N \to \infty.$$

A special feature of these uncertainty sets is that the inner problem can be solved in closed form. In fact, any linear objective function $\sum\limits_{s' \in \mathcal{S}} a(s')p(s')$ in which the coefficients are of fixed sign and only finitely many of them are nonzero, can be maximized in closed form over these sets. The proof for the case with all nonnegative coefficients is given below. A similar logic can be used when the coefficients are non-positive. We also derived a similar result in the context of a (finite-dimensional) healthcare application in [40].

Consider the LP

$$\begin{aligned} \max \; & \sum_{s' \leq m} \alpha(s')p(s') \\ \text{s.t.} \; & l(s') \leq p(s') \leq u(s'), \quad s' \in \mathcal{S}, \\ & \sum_{s' \in \mathcal{S}} p(s') = 1. \end{aligned}$$

Suppose $\alpha(s') \geq 0$ for all s'. Sort and re-index the non-zero coefficients (if necessary) so that $\alpha(1) \geq \alpha(2) \geq \ldots \geq \alpha(m) \geq 0$. Then, define the switching-index j as the smallest positive integer such that $\sum_{s' \leq j} u(s') + \sum_{s' > j} l(s') \geq 1$. Such an index exists since $\sum_{s' \in \mathcal{S}} u_s^a(s) > 1$. The optimal solution is given by

$$p^*(s') = \begin{cases} u(s'), & s' < j, \\ 1 - \sum_{s' < j} u(s') - \sum_{s' > j} l(s'), & s' = j, \\ l(s'), & s' > j. \end{cases}$$

Clearly, the components of p^* sum up to 1 and $l(s') \leq p^*(s') \leq u(s')$ for all $s' \neq j$. Moreover, by choice of j,

$$\begin{aligned} \sum_{s' < j} u(s') + \sum_{s' \geq j} l(s') < 1 \leq \sum_{s' \leq j} u(s') + \sum_{s' > j} l(s') \\ \implies l(j) \leq 1 - \sum_{s' < j} u(s') - \sum_{s' > j} l(s') \leq u(j). \end{aligned}$$

Thus, the proposed solution is feasible. Let $F(\cdot)$ be the objective function. To prove optimality, we show that $F(p^*) \geq F(p)$ for any feasible solution p. If $j \geq m$, we have

$$F(p^*) - F(p) = \sum_{s' \leq m} \alpha(s')(u(s') - p(s')) \geq 0.$$

Otherwise,

$$\begin{aligned}
F(p^*) - F(p) &= \sum_{s'<j} \alpha(s')(u(s') - p(s')) + \alpha(j)(p^*(j) - p(j)) + \sum_{j<s'\leq m} \alpha(s')(l(s') - p(s')) \\
&= \sum_{s'\leq j} \alpha(s')(u(s') - p(s')) + \sum_{j<s'\leq m} \alpha(s')(l(s') - p(s')) \\
&+ \alpha(j)\Big(1 - \sum_{s'<j} u(s') - \sum_{s'>j} l(s') - 1 + \sum_{s'\neq j} p(s')\Big) \\
&= \sum_{s'\leq j} [\underbrace{\alpha(s') - \alpha(j)}_{\geq 0}](\underbrace{u(s') - p(s')}_{\geq 0}) + \sum_{j<s'\leq m} [\underbrace{\alpha(s') - \alpha(j)}_{\leq 0}](\underbrace{l(s') - p(s')}_{\leq 0}) \\
&+ \alpha(j)\Big(\sum_{s'>m} (\underbrace{p(s') - l(s')}_{\geq 0})\Big) \\
&\geq 0.
\end{aligned}$$

Hence, the proposed solution is optimal. This implies that the N-state inner problem can be solved in closed form, by defining $\alpha(s') = c(s, a, s') + \lambda v(s') \geq 0$ for $s' \leq N$, and reordering the coefficients in descending order. Moreover, $M_N(s, a) = \sup_{p\in\mathcal{P}_s^a} \sum_{s'>N} p(s') = 1 + \sup_{p\in\mathcal{P}_s^a} \big(-\sum_{s'\leq N} p(s')\big)$ can also be computed in closed form. The same is also true for the LPs which arise in the computation of the bound $\bar{\delta}(s, a, N, T)$. In all these problems, we can choose ϵ_N to be identically zero.

Thus, robust MDPs with interval uncertainty sets for the transition probabilities can be solved using the proposed method.

3.6.2 Bounded reachability

This example considers a class of problems where the *change* in state of a system in a single period is uniformly bounded above by some constant M. That is, if the system transitions from state s to s' under some action a, we must have $s' \leq s + M$. Note that the set of all possible states in any period is still the entire set $\mathcal{S}$.

A particular example of this model is the infinite-horizon inventory management problem described in Example 1 of [30], wherein a seller controls the inventory of a single product with unlimited inventory capacity. The objective is to minimize the total discounted cost. The state s of the system is defined as the current inventory level and can take any value in $\mathcal{S} = \{0, 1, 2, \ldots\}$. The seller chooses an order quantity $a \in \mathcal{A} = \{0, 1, \ldots, M\}$, where M is some upper limit on the order quantity in a period. If the demand in a period is t units and a units of the product are ordered, the inventory level changes from s to $s + a - t$, which is at most $s + a \leq s + M$. Thus, this problem fits within the framework described above, and our method applies to a robust bounded-cost variant of the same.

In the nominal case, the bound on state transitions implies that the transition probability $p(\cdot|s, a)$ for any state-action pair (s, a) is supported on $\{0, 1, \ldots, s + M\}$. Keeping the same interpretation in mind, the uncertainty sets $\mathcal{P}_s^a$ are also chosen so that $p(s'|s, a) = 0$ for all $s' > s + M$. We first note that such an uncertainty set always satisfies Assumption 3.5.1, as

$$M_N(s, a) = \sup_{p \in \mathcal{P}_s^a} \sum_{s' > N} p(s') = 0 \text{ for all } N > s + M.$$

In fact, the inner problem also reduces to a finite-dimensional optimization problem, and can be solved to arbitrary accuracy. Therefore, our uncertainty sets have all the requisite properties and the proposed policy iteration algorithm can be used. Moreover, we note that the computation of a uniform bound $\overline{\delta}(s, a, N, T)$ can also be simplified in this case.

We have,

$$M_N(s,a) = \sup_{p\in\mathcal{P}_s^a}\sum_{s'>N} p(s'|s,a) = \sup_{p\in\mathcal{P}_s^a}\sum_{N<s'\le s+a} p(s'|s,a) \le \sup_{p\in\mathcal{P}_s^a}\sum_{N<s'\le s+M} p(s'|s,a)\ \forall\ a$$

$$\implies M_N(s) \le \begin{cases} 0, & N \ge s+M, \\ 1, & N < s+M \end{cases} = \mathbf{1}\{N < s+M\}.$$

$$\begin{aligned}\sup_{p\in\mathcal{P}_s^a}\mathbf{E}_{p_N}[M_N(s')] &= \sup_{p\in\mathcal{P}_s^a}\sum_{s'\le N} p(s'|s,a)\mathbf{1}\{N < s'+M\} \\ &= \sup_{p\in\mathcal{P}_s^a}\sum_{\substack{s'\le N,s+a \\ N-M<s'}} p(s'|s,a) \le \sup_{p\in\mathcal{P}_s^a}\sum_{\substack{s'\le N,s+M \\ N-M<s'}} p(s'|s,a) \\ &\le \begin{cases} 0, & \min\{N,s+M\} \le N-M, \\ 1, & \min\{N,s+M\} > N-M \end{cases} = \mathbf{1}\{N < s+2M\}.\end{aligned}$$

Proceeding in this manner gives that

$$B_N(s;T) \le \frac{c}{1-\lambda}\sum_{t=1}^{T}\lambda^{t-1}\mathbf{1}\{N < s+tM\}. \tag{3.39}$$

This expression is the same as that obtained in the nominal case for the inventory control model in [30]. Recall that $B_N(s;T)$ was used in the recursive computation of the uniform bound $\bar{\delta}$ in Lemma 3.5.8. In this model, however, it is easier to compute an upper bound on $\bar{\delta}$ using the right-hand-side of (3.39) instead of $B_N(s;T)$. This bound also vanishes as N and T grow, and can be used in place of $\bar{\delta}$ in Step 2(e) of the algorithm.

3.6.3 Stochastic equipment replacement

Finally, we provide a specific application which does not fall under the previous two classes of models. A nominal version of this model is discussed in [21], and an unbounded-cost variant appears in Section 6.10 of [35].

Consider a stochastic equipment replacement model where the state $s \in \{0,1,2,...\}$

denotes the condition of the equipment at the beginning of a time-period. State 0 corresponds to a new equipment; larger states represent poorer equipment conditions. At the beginning of each time-period, the decision-maker can either choose to replace the equipment with a new one (action 0) or keep the existing equipment (action 1). Between two decision epochs, the condition of the equipment worsens by $i \geq 0$ states with probability $q(i)$. This leads to the transition probabilities

$$p(s'|s,0) = q(s'), s' \geq 0,$$

$$p(s'|s,1) = \begin{cases} 0, & s' < s, \\ q(s'-s), & s' \geq s. \end{cases}$$

The costs in this model are given by

$$c(s,0) = \alpha + h(0), \qquad c(s,1) = h(s),$$

where $\alpha > 0$ is the cost of buying a new piece of equipment and $h(s)$ is the cost of operating an equipment in condition s for one period. It is natural to expect that $h(s)$ is non-decreasing in s as it should be cheaper to operate an equipment that is in a better condition. Assume that $h(\cdot)$ is bounded and non-negative as well. For example, $h(s) = 1 - \exp(-s)$. Thus, the immediate costs are bounded between zero and $\alpha + 1$.

Now, suppose we have some empirical estimate of the distribution $q(\cdot)$. It is reasonable to assume that $q(i)$ is non-increasing in i, since the probability to worsen by $i+1$ states should not be larger than the probability of worsening by i states. An example of this would be if q were a geometric distribution, where $q(i) = \beta^i(1-\beta)$, $i = 0, 1, \ldots$. In practice, one would often estimate the parameter β instead of the distribution directly. Suppose the $0 < l \leq u < 1$ represent some lower and upper confidence bounds on the value of β.

This gives us the following uncertainty sets.

$$\mathcal{P}_s^0 = \Big\{ p \in \mathcal{M}(\mathcal{S}) : p(s') = (1-\beta)\beta^{s'}, s' \in \mathcal{S},\ l \le \beta \le u \Big\},$$
$$\mathcal{P}_s^1 = \Big\{ p \in \mathcal{M}(\mathcal{S}) : p(s') = 0, s' < s; p(s') = (1-\beta)\beta^{s'-s}, s' \ge s,\ l \le \beta \le u \Big\}.$$

Note that each of these sets satisfies Assumption 3.5.1, since $\sum_{s'>N} p(s') \le (1-l)u^{N+1}/(1-u)$ for all $p \in \mathcal{P}_s^0$, and similarly for $\mathcal{P}_s^1$.

For any $N \in \mathcal{S}$ and states $s \le N$,

$$\begin{aligned} M_N(s,0) &= \sup_{p(\cdot|s,0)\in\mathcal{P}_s^0} \sum_{s'>N} p(s'|s,0) = \sup_{\beta\in[l,u]} \sum_{s'>N} (1-\beta)\beta^{s'} = \sup_{\beta\in[l,u]} \beta^{N+1} = u^{N+1}. \\ M_N(s,1) &= \sup_{p(\cdot|s,1)\in\mathcal{P}_s^1} \sum_{s'>N} p(s'|s,1) = \sup_{\beta\in[l,u]} \sum_{s'>N} (1-\beta)\beta^{s'-s} = \sup_{\beta\in[l,u]} \beta^{N+1-s} = u^{N+1-s}. \\ M_N(s) &= \max\{M_N(s,0), M_N(s,1)\} = u^{N+1-s}. \end{aligned}$$

Computation of the uniform bound $\bar{\delta}$ using Algorithm 3, as well as the solution of the inner problem, consists of numerically solving problems similar in structure to the following one-variable nonlinear maximization probLem

$$\begin{array}{ll} \max & \sum_{s'\le N} \alpha(s')p(s'|s,0) \\ \text{s.t.} & p \in \mathcal{P}_s^0 \end{array} \quad \equiv \quad \begin{array}{ll} \max & (1-\beta)\sum_{s'\le N} \alpha(s')\beta^{s'} \\ \text{s.t.} & l \le \beta \le u. \end{array}$$

These problems can easily be solved to arbitrary accuracy, and the proposed policy iteration method can be used to solve the robust MDP.

3.7 Conclusion

An as-is execution of policy iteration on robust MDPs encounters severe hurdles when the state-space is countable. We used approximation techniques to resolve these challenges and delivered and algorithm that can be implemented in practice. The policy evaluation and improvement steps in the existing method require an infinite amount of computation, whereas we reduced them to finite systems of equations via a state-space truncation approach. Additional complications in policy evaluation ensued due to the nonlinearity of the robust evaluation operator. These were resolved by employing a finite number of successive approximation steps to compute an approximate value function. Further, exact solutions to the inner problems might not always be available, and our method accounted for the errors arising from their numerical solution to some nonzero accuracy. These ideas led to an approximate policy iteration algorithm where each step requires a finite amount of memory and computation. This algorithm generates a sequence of policies and computes their approximate values. Although the true values of these policies are unknown, the method guarantees that they improve in each iteration. We proved that the proposed algorithm converges in value to the optimal value function, and the policies generated converge subsequentially to an optimal policy. We also provided three examples which fall within our framework — robust MDPs with interval uncertainty sets, robust MDPs where the change in state in a single period is bounded, and a robust equipment replacement probLem We showed that these models possess the desired properties that render approximate policy iteration a viable algorithm.

A natural direction for future research would be to extend our policy iteration algorithm to the case where immediate cost functions are allowed to be unbounded. This is not straightforward, primarily because the theory of countable-state robust MDPs is currently available only for the bounded-cost case [28]. As such, any algorithmic work would first require an extension of the theory in Section 6.10 of Puterman [35] to the robust setting, including the optimality of the robust Bellman equations and the existence of optimal solutions. These results are developed in the next chapter.

Chapter 4

ROBUST COUNTABLE-STATE MARKOV DECISION PROCESSES WITH UNBOUNDED COSTS

4.1 Introduction

In the previous chapter, we assumed that the immediate cost function $c(s, a, s')$ was uniformly bounded. This greatly simplified the calculations within the proofs, in addition to ensuring convergence of various infinite sums and expectations that arise therein. More importantly, it allowed us to invoke the existing theory for robust MDPs and restrict our attention to algorithm development. In many applications, however, a naturally arising cost function violates this assumption. Thus, we widen the scope in this chapter to allow for more general cost functions similar to those considered in [30] and Section 6.10 of [35].

The ultimate objective is to develop a practical convergent method for solving robust unbounded-cost MDPs, but the first hurdle in this case arises from the fact that a theoretical treatment of this class of MDPs is not available in the literature. Hence, in this chapter, we develop a theoretical framework for robust countable-state MDPs with unbounded cost functions. We establish the optimality of the robust Bellman equations. We show that the robust Bellman operator is a J-step contraction mapping on an appropriately defined Banach space, thus guaranteeing the existence and uniqueness of an optimal value function.

4.2 Problem Setup

Consider an infinite-horizon MDP with decision-epochs $t = 0, 1, 2, \ldots$. The state-space $\mathcal{S} = \{1, 2, \ldots, \}$ is assumed to be countable, while the action-set $\mathcal{A}(s) = \{1, 2, \ldots, A\}$ is discrete (finite or countably infinite). At the start of period t, the system occupies a state $s \in \mathcal{S}$. A decision-making agent observes this state and chooses an action $a \in \mathcal{A}(s)$. Then, at the end of the period, the system transitions to a state $s' \in \mathcal{S}$ with probability $p(s'|s,a)$. This transition incurs a cost $c(s,a,s')$ discounted by a factor λ^t, where $\lambda \in (0,1)$ is a fixed parameter. A decision rule is a function which assigns an action to every state in $\mathcal{S}$, while a policy $\sigma = (d_1, d_2, \ldots)$ is a function prescribing a decision rule d_t for every period t. Let Π be the collection of all admissible policies. The agent aims to find a policy in Π that minimizes the expected total discounted cost over the entire horizon.

As in the previous chapters, the nominal setup described above assumes that transition probability $p(\cdot|s,a)$ for any state-action pair (s,a) is a model parameter known to the decision-maker. In practice, these probabilities are estimated statistically, allowing for estimation errors to affect the choice of optimal policy. The robust approach seeks to immunize the decision-maker against these errors by assuming that the transition probabilities are ambiguous. Let $\mathcal{M}(\mathcal{S})$ be the set of all probability mass functions (pmfs) on $\mathcal{S}$. We assume that for each state-action pair (s,a), the pmf $p(\cdot|s,a)$ is only known to lie in an uncertainty set $\mathcal{P}_s^a \subset \mathcal{M}(\mathcal{S})$ comprising plausible choices for the true transition probabilities. Given these sets, the set of transition probabilities consistent with a fixed decision-rule d is

$$\mathcal{T}^d = \Big\{ \mathbf{p} : \mathcal{S} \to \mathcal{M}(\mathcal{S}) : \forall s \in \mathcal{S}, \mathbf{p}(s) \triangleq p(\cdot|s, d(s)) \in \mathcal{P}_s^{d(s)} \Big\}.$$

Following the rectangularity assumption in [28], we also assume that for a policy $\sigma = (d_1, d_2, \ldots)$, the set of probability distributions consistent with σ is given by $\mathcal{T}^\sigma = \{\tau = (\mathbf{p}_1, \mathbf{p}_2, \ldots) : \mathbf{p}_t \in \mathcal{T}^{d_t}\}$. In the robust variant, the decision-maker follows a conservative approach and tries to minimize the worst-case expected total discounted cost. This amounts to solving the following optimization problem.

$$v^*(s) = \inf_{\sigma \in \Pi} \sup_{\tau \in \mathcal{T}^\sigma} \mathbf{E}_\tau \big[\sum_{t=0}^{\infty} \lambda^t c(s_t, d_t(s_t), s_{t+1}) \big], \quad s \in \mathcal{S}. \tag{4.1}$$

In the previous chapter, we assumed that the immediate costs $c(s,a,s')$ were uniformly bounded for all $s, s' \in \mathcal{S}$ and $a \in \mathcal{A}(s)$. Here, we drop this assumption to allow for more general cost functions that occur naturally in many applications. In particular, the costs are allowed to be unbounded, provided their growth with s' is sufficiently slow. This is made precise with the following assumptions on the behavior of the cost function. For a fixed state-action pair (s, a) and pmf $p(\cdot|s, a)$, let $\mathbf{E}_p[u(s')]$ be the expected value (or weighted average) of any function u defined on $\mathcal{S}$. That is, $\mathbf{E}_p[u(s')] = \sum_{s' \in \mathcal{S}} p(s'|s, a)u(s')$. We suppress the (s, a)-dependence in this notation since it is implied by context. Let w be a (known) function on $\mathcal{S}$ such that $\inf_{s \in \mathcal{S}} w(s) > 0$, and the following properties are satisfied.

Assumption 4.2.1. *There exists a constant $\mu < \infty$ such that*

$$\sup_{a \in \mathcal{A}(s)} \sup_{p \in \mathcal{P}_s^a} \left| \mathbf{E}_p[c(s, a, s')] \right| \leq \mu w(s), \quad \textit{for all } s \in \mathcal{S}. \tag{4.2}$$

Assumption 4.2.2. *There exists a constant κ, $0 \leq \kappa < \infty$, for which*

$$\sup_{p \in \mathcal{P}_s^a} \sum_{s' \in \mathcal{S}} p(s'|s, a)w(s') \leq \kappa w(s) \quad \textit{for all } s \in \mathcal{S}, a \in \mathcal{A}(s). \tag{4.3}$$

Assumption 4.2.3. *There exists a constant α, $0 \leq \alpha < 1$ and an integer J such that*

$$\lambda^J \sum_{s' \in \mathcal{S}} \mathbf{P}_\sigma^J(s'|s)w(s') \leq \alpha w(s) \tag{4.4}$$

for all $\sigma = (d_1, \ldots, d_J)$ and $\mathbf{P}_\sigma^J = \prod_{j=1}^{J} \mathbf{p}_{d_j}$, where d_j is an admissible decision-rule and $\mathbf{p}_{d_j} \in \mathcal{T}^{d_j}$, for $1 \leq j \leq J$.

Assumption 4.2.1 states that starting in state s and for any choice of action a, the worst-case expected cost occurred in a single-period transition is at most $\mu w(s)$. On the other hand, Assumptions 4.2.2 and 4.2.3 ascertain that the function w itself has some desirable behavior. Non-robust versions of these assumptions are standard in the unbounded-cost MDP literature. See [30] and [35] for a detailed discussion and for examples of cost functions c such that the assumptions are satisfied by a suitably chosen w. A theoretical framework for unbounded cost robust MDPs is not available in the literature, and we proceed to develop the same in the next section.

4.3 Theoretical Results

The robust Bellman equations are given by

$$v(s) = \inf_{a \in \mathcal{A}(s)} \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_p[c(s,a,s') + \lambda v(s')], \quad \text{for all } s \in \mathcal{S}. \tag{4.5}$$

In this section, we establish that a unique solution to (4.5) always exists, and that this solution must be the optimal value function v^* for the robust MDP defined in Section 4.2. Additionally, we prove that solving a sequence of finite-state approximate MDPs to optimality, asymptotically recovers the optimal value function.

For any function v on $\mathcal{S}$, define the w-norm of v as $\|v\|_w \triangleq \sup_{s \in \mathcal{S}} |v(s)|/w(s)$. This is a well-defined norm since w is a positive function bounded away from zero. Let V_w be the set of all functions on $\mathcal{S}$ whose w-norm is finite. It is easy to see that V_w is a Banach space under the w-norm. The following lemma shows that the value function for every policy in Π lies in V_w.

Lemma 4.3.1 (Norm bounds)**.** *For each $\sigma \in \Pi$ and $s \in \mathcal{S}$, we have*

$$|v^\sigma(s)| \leq \frac{\mu}{1-\alpha}\big[1 + \lambda\kappa + \ldots + (\lambda\kappa)^{J-1}\big] w(s). \tag{4.6}$$

Further, $\|v^\sigma\|_w \leq \frac{\mu}{1-\alpha}\big[1 + \lambda\kappa + \ldots + (\lambda\kappa)^{J-1}\big]$.

Proof. Fix a policy σ and state s_0. Then,

$$\begin{aligned} v^\sigma(s_0) &= \sup_{\tau \in \mathcal{T}^\sigma} \Big(\mathbf{E}_\tau \big[\sum_{t=0}^{\infty} \lambda^t c(s_t, d_t(s_t), s_{t+1}) \big] \Big) \\ \implies |v^\sigma(s_0)| &= \Big| \sup_{\tau \in \mathcal{T}^\sigma} \Big(\mathbf{E}_\tau \big[\sum_{t=0}^{\infty} \lambda^t c(s_t, d_t(s_t), s_{t+1}) \big] \Big) \Big| \\ &\le \sup_{\tau \in \mathcal{T}^\sigma} \Big(\mathbf{E}_\tau \big[\sum_{t=0}^{\infty} \lambda^t |c(s_t, d_t(s_t), s_{t+1})| \big] \Big). \end{aligned}$$

For any n and given a $\tau \in \mathcal{T}^\sigma$, let τ^n denote transition probabilities starting from period n, that is, $\tau^n = (\mathbf{p}_n, \mathbf{p}_{n+1}, \ldots)$ and $\mathcal{T}^\sigma_n = \{\tau^n = (\mathbf{p}_n, \mathbf{p}_{n+1}, \ldots) : \mathbf{p}_t \in \mathcal{T}^{d_t}\}$.

Then,

$$\begin{aligned} |v^\sigma(s_0)| &\le \sup_{\tau \in \mathcal{T}^\sigma} \Big(\mathbf{E}_\tau \big[\sum_{t=0}^{\infty} \lambda^t |c(s_t, d_t(s_t), s_{t+1})| \big] \Big) \\ &= \sup_{\substack{(\mathbf{p}_1, \tau^2) \\ \in \mathcal{T}^{d_1} \times \mathcal{T}^\sigma_2}} \Big(\mathbf{E}_{\mathbf{p}_1, \tau^1} \big[\sum_{t=0}^{\infty} \lambda^t |c(s_t, d_t(s_t), s_{t+1})| \big] \Big) \\ &= \sup_{\substack{(\mathbf{p}_1, \tau^2) \\ \in \mathcal{T}^{d_1} \times \mathcal{T}^\sigma_2}} \Big(\mathbf{E}_{\mathbf{p}_1, \tau^1} \big[|c(s_0, d_0(s_0), s_1)| + \sum_{t=1}^{\infty} \lambda^t |c(s_t, d_t(s_t), s_{t+1})| \big] \Big) \\ &= \sup_{\substack{(\mathbf{p}_1, \tau^2) \\ \in \mathcal{T}^{d_1} \times \mathcal{T}^\sigma_2}} \Big(\mathbf{E}_{\mathbf{p}_1} \big[|c(s_0, d_0(s_0), s_1)| + \mathbf{E}_{\tau^2} \big[\sum_{t=1}^{\infty} \lambda^t |c(s_t, d_t(s_t), s_{t+1})| \big] \big] \Big), \end{aligned}$$

where the last equality holds since the term $|c(s_0, d_0(s_0), s_1)|$ does not depend on τ^1. It follows that

$$\begin{aligned}
|v^\sigma(s_0)| &\le \sup_{\mathbf{p}_1 \in \mathcal{T}^{d_1}} \Big(\mathbf{E}_{\mathbf{p}_1}\big[|c(s_0, d_0(s_0), s_1)| + \sup_{\tau^2 \in \mathcal{T}_2^\sigma} \Big(\mathbf{E}_{\tau^2}\big[\sum_{t=1}^{\infty} \lambda^t |c(s_t, d_t(s_t), s_{t+1})|\big]\Big)\big]\Big) \\
&\le \sup_{\mathbf{p}_1 \in \mathcal{T}^{d_1}} \Big(\mathbf{E}_{\mathbf{p}_1}\big[|c(s_0, d_0(s_0), s_1)|\big]\Big) + \sup_{\mathbf{p}_1 \in \mathcal{T}^{d_1}} \Big(\mathbf{E}_{\mathbf{p}_1}\big[\sup_{\tau^2 \in \mathcal{T}_2^\sigma} \Big(\mathbf{E}_{\tau^2}\big[\sum_{t=1}^{\infty} \lambda^t |c(s_t, d_t(s_t), s_{t+1})|\big]\Big)\big]\Big) \\
&= \sup_{\mathbf{p}_1 \in \mathcal{T}^{d_1}} \Big(\mathbf{E}_{\mathbf{p}_1}\big[|c(s_0, d_0(s_0), s_1)|\big]\Big) + \sup_{\tau \in \mathcal{T}^\sigma} \Big(\mathbf{E}_{\tau}\big[\sum_{t=1}^{\infty} \lambda^t |c(s_t, d_t(s_t), s_{t+1})|\big]\Big) \\
&\le \mu w(s_0) + \sup_{\tau \in \mathcal{T}^\sigma} \Big(\mathbf{E}_{\tau}\big[\sum_{t=1}^{\infty} \lambda^t |c(s_t, d_t(s_t), s_{t+1})|\big]\Big).
\end{aligned}$$

In the second inequality, we used the fact that $\sup(f+g) \le \sup(f) + \sup(g)$ for any functions f and g. The last inequality follows from Assumption 4.2.1. Using a similar argument as above, we have that

$$\begin{aligned}
|v^\sigma(s_0)| &\le \mu w(s_0) + \sup_{\substack{(\mathbf{p}_1, \mathbf{p}_2, \tau_3) \\ \in \mathcal{T}^{d_1} \times \mathcal{T}^{d_2} \times \mathcal{T}_3^\sigma}} \Big(\mathbf{E}_{(\mathbf{p}_1, \mathbf{p}_2, \tau_3)}\big[\lambda |c(s_1, d_1(s_1), s_2)| + \sum_{t=2}^{\infty} \lambda^t |c(s_t, d_t(s_t), s_{t+1})|\big]\Big) \\
&\le \mu w(s_0) + \lambda \sup_{\substack{(\mathbf{p}_1, \mathbf{p}_2) \\ \in \mathcal{T}^{d_1} \times \mathcal{T}^{d_2}}} \Big(\mathbf{E}_{(\mathbf{p}_1, \mathbf{p}_2)}\big[|c(s_1, d_1(s_1), s_2)|\big]\Big) + \sup_{\tau \in \mathcal{T}^\sigma} \Big(\mathbf{E}_{\tau}\big[\sum_{t=2}^{\infty} \lambda^t |c(s_t, d_t(s_t), s_{t+1})|\big]\Big).
\end{aligned}$$

Then, for any $\mathbf{p}_1 \in \mathcal{T}^{d_1}$ and $\mathbf{p}_2 \in \mathcal{T}^{d_2}$,

$$\begin{aligned}
\mathbf{E}_{(\mathbf{p}_1, \mathbf{p}_2)}\big[|c(s_1, d_1(s_1), s_2)|\big] &= \mathbf{E}_{\mathbf{p}_1}\big[\mathbf{E}_{\mathbf{p}_2}\big[|c(s_1, d_1(s_1), s_2)|\big]\big] \\
&\le \mathbf{E}_{\mathbf{p}_1}\big[\mu w(s_1)\big] \\
&\le \mu \kappa w(s_0).
\end{aligned}$$

Therefore,

$$|v^\sigma(s_0)| \le \mu[1 + \lambda\kappa] w(s_0) + \sup_{\tau \in \mathcal{T}^\sigma} \Big(\mathbf{E}_{\tau}\big[\sum_{t=2}^{\infty} \lambda^t |c(s_t, d_t(s_t), s_{t+1})|\big]\Big).$$

Repeating this process J times gives

$$\begin{aligned}
|v^\sigma(s_0)| &\le \mu[1+(\lambda\kappa)+(\lambda\kappa)^{J-1}]w(s_0)+\sup_{\tau\in\mathcal{T}^\sigma}\Big(\mathbf{E}_\tau\big[\sum_{t=J}^{\infty}\lambda^t|c(s_t,d_t(s_t),s_{t+1})|\big]\Big)\\
&= \mu[1+(\lambda\kappa)+(\lambda\kappa)^{J-1}]w(s_0)+\\
&\quad \sup_{\tau\in\mathcal{T}^\sigma}\Big(\mathbf{E}_\tau\big[\lambda^J|c(s_J,d_J(s_J),s_{J+1})|+\sum_{t=J+1}^{\infty}\lambda^t|c(s_t,d_t(s_t),s_{t+1})|\big]\Big)\\
&\le \mu[1+(\lambda\kappa)+(\lambda\kappa)^{J-1}]w(s_0)+\lambda^J\sup_{\substack{(\mathbf{p}_1,\ldots,\mathbf{p}_{J+1})\\ \in\mathcal{T}^{d_1}\times\ldots\times\mathcal{T}^{d_{J+1}}}}\Big(\mathbf{E}_{(\mathbf{p}_1,\ldots,\mathbf{p}_{J+1})}\big[|c(s_J,d_J(s_J),s_{J+1})|\big]\Big)\\
&\quad +\sup_{\tau\in\mathcal{T}^\sigma}\Big(\mathbf{E}_\tau\big[\sum_{t=J+1}^{\infty}\lambda^t|c(s_t,d_t(s_t),s_{t+1})|\big]\Big).
\end{aligned}$$

Again, for any $(\mathbf{p}_1,\ldots,\mathbf{p}_{J+1})\in\mathcal{T}^{d_1}\times\ldots\times\mathcal{T}^{d_{J+1}}$,

$$\begin{aligned}
\lambda^J\mathbf{E}_{(\mathbf{p}_1,\ldots,\mathbf{p}_{J+1})}\big[|c(s_J,d_J(s_J),s_{J+1})|\big] &= \lambda^J\mathbf{E}_{(\mathbf{p}_1,\ldots,\mathbf{p}_J)}\big[\mathbf{E}_{p_{J+1}}\big[|c(s_J,d_J(s_J),s_{J+1})|\big]\big]\\
&\le \lambda^J\mathbf{E}_{(\mathbf{p}_1,\ldots,\mathbf{p}_J)}\big[\mu w(s_J)\big]\le\alpha\mu w(s_0).
\end{aligned}$$

Therefore,

$$|v^\sigma(s_0)|\le\mu[1+(\lambda\kappa)+(\lambda\kappa)^{J-1}]w(s_0)+\alpha\mu w(s_0)+\sup_{\tau\in\mathcal{T}^\sigma}\Big(\mathbf{E}_\tau\big[\sum_{t=J+1}^{\infty}\lambda^t|c(s_t,d_t(s_t),s_{t+1})|\big]\Big).$$

Repeating the above arguments for every group of J terms gives us that

$$\begin{aligned}
|v^\sigma(s_0)| &\le [1+\lambda\kappa+\ldots+(\lambda\kappa)^{J-1}]\mu w(s_0)+\alpha[1+\lambda\kappa+\ldots+(\lambda\kappa)^{J-1}]\mu w(s_0)\\
&\quad +\alpha^2[1+\lambda\kappa+\ldots+(\lambda\kappa)^{J-1}]\mu w(s_0)+\ldots\\
&= \frac{\mu}{1-\alpha}[1+\lambda\kappa+\ldots+(\lambda\kappa)^{J-1}]w(s_0)\\
\implies \|v^\sigma\|_w &\le \frac{\mu}{1-\alpha}[1+\lambda\kappa+\ldots+(\lambda\kappa)^{J-1}].
\end{aligned}$$

This completes the proof. □

Define the robust Bellman operator $\mathcal{L}$ on V_w as

$$\mathcal{L}(u)(s) = \inf_{a\in\mathcal{A}(s)} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_p[c(s,a,s') + \lambda u(s')] \quad \text{for all } u \in V_w. \tag{4.7}$$

First, we verify that the operator is well-defined on V_w. For any $s \in \mathcal{S}$,

$$\begin{aligned}
|\mathcal{L}(u)(s)| &= \Big| \inf_{a\in\mathcal{A}(s)} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_p\big[c(s,a,s') + \lambda u(s')\big]\Big| \\
&\le \sup_{a\in\mathcal{A}(s)} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_p\big[\big|c(s,a,s') + \lambda u(s')\big|\big] \\
&\le \sup_{a\in\mathcal{A}(s)} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_p\big[\big|c(s,a,s')\big|\big] + \lambda \mathbf{E}_p\big[\big|u(s')\big|\big] \\
&\le \sup_{a\in\mathcal{A}(s)} \sup_{p\in\mathcal{P}_s^a} \mathbf{E}_p\big[\big|c(s,a,s')\big|\big] + \lambda \|u\|_w \mathbf{E}_p\big[w(s')\big] \\
&\le \sup_{a\in\mathcal{A}(s)} \big(\mu w(s) + \lambda \|u\|_w \kappa w(s)\big) \\
&= (\mu + \lambda\kappa\|u\|_w) w(s).
\end{aligned}$$

Since this is true for all $s \in \mathcal{S}$, it follows that

$$\|\mathcal{L}(u)\|_w \le \mu + \lambda\kappa\|u\|_w < \infty.$$

Thus, $\mathcal{L} : V_w \to V_w$ is a well-defined operator. A function v satisfies the Bellman equations if and only if it is a fixed point of $\mathcal{L}$. The existence (and uniqueness) of such a fixed point is established in the following theorem. For bounded-cost MDPs, the contraction property of the robust Bellman operator guarantees that an optimal solution to the Bellman equations always exists. However, when costs are unbounded, $\mathcal{L}$ is no longer a contraction mapping on V_w. The next theorem shows that $\mathcal{L}^J$ is a contraction mapping on V_w, where J was defined in Assumption 4.2.3. The idea of the proof is similar to [28].

Theorem 4.3.2. *Let $\mathcal{L}$ be the robust Bellman operator defined in Equation (4.7).*

(a) $\mathcal{L}^J$ is a contraction mapping on V_w, that is,

$$\|\mathcal{L}^J(u) - \mathcal{L}^J(v)\|_w \leq \alpha \|u - v\|_w, \quad \textit{for all } u, v \in V_w.$$

(b) $\mathcal{L}$ has a unique fixed point $\bar{v}$ in V_w, and

$$\bar{v}(s) = \inf_{\sigma \in \Pi} \sup_{\tau \in \mathcal{T}^\sigma} \mathbf{E}_\tau \big[\sum_{t=0}^{\infty} \lambda^t c(s_t, d_t(s_t), s_{t+1}) \big] \quad \textit{for all } s \in \mathcal{S}.$$

Proof. We first prove that $\mathcal{L}^J$ is a contraction mapping. Let u and v be any two functions in V_w, and ϵ be an arbitrary positive number. Fix a state $s \in \mathcal{S}$. Suppose $\mathcal{L}^J(u)(s) \leq \mathcal{L}^J(v)(s)$. Then,

$$0 \leq \mathcal{L}^J(v)(s) - \mathcal{L}^J(u)(s) = \mathcal{L}(\mathcal{L}^{J-1}(v)(s)) - \mathcal{L}(\mathcal{L}^{J-1}(u)(s)).$$

By definition of the infimum, there exists an action $a(s) \in \mathcal{A}(s)$ such that

$$\begin{aligned} \mathcal{L}(\mathcal{L}^{J-1}(u)(s)) &= \inf_{a \in \mathcal{A}(s)} \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_p \big[c(s, a, s_1) + \lambda \mathcal{L}^{J-1}(u)(s_1) \big] \\ &> \sup_{p \in \mathcal{P}_s^{a(s)}} \mathbf{E}_p \big[c(s, a(s), s_1) + \lambda \mathcal{L}^{J-1}(u)(s_1) \big] - \epsilon. \end{aligned}$$

Then,

$$\begin{aligned} 0 &\leq \mathcal{L}^J(v)(s) - \mathcal{L}^J(u)(s) \\ &< \sup_{p \in \mathcal{P}_s^{a(s)}} \mathbf{E}_p \big[c(s, a(s), s_1) + \lambda \mathcal{L}^{J-1}(v)(s_1) \big] \\ &\quad - \sup_{p \in \mathcal{P}_s^{a(s)}} \mathbf{E}_p \big[c(s, a(s), s_1) + \lambda \mathcal{L}^{J-1}(u)(s_1) \big] + \epsilon. \end{aligned}$$

Further, there exists a pmf $p^s \in \mathcal{P}_s^{a(s)}$ such that

$$\sup_{p \in \mathcal{P}_s^{d_J(s)}} \mathbf{E}_p\big[c(s,a(s),s_1) + \lambda \mathcal{L}^{J-1}(v)(s_1)\big] - \epsilon < \mathbf{E}_{p^s}\big[c(s,a(s),s_1) + \lambda \mathcal{L}^{J-1}(v)(s_1)\big].$$

Therefore,

$$\begin{aligned} 0 &\leq \mathcal{L}^J(v)(s) - \mathcal{L}^J(u)(s) \\ &< \mathbf{E}_{p^s}\big[c(s,a(s),s_1) + \lambda \mathcal{L}^{J-1}(v)(s_1)\big] + \epsilon - \mathbf{E}_{p^s}\big[c(s,a(s),s_1) + \lambda \mathcal{L}^{J-1}(u)(s_1)\big] + \epsilon \\ &= \lambda \mathbf{E}_{p^s}\big[\mathcal{L}^{J-1}(v)(s_1) - \mathcal{L}^{J-1}(u)(s_1)\big] + 2\epsilon. \end{aligned}$$

Note that we find such an action $a(s)$ for each state s, which defines a decision-rule d_J. Moreover, we can also define $\mathbf{p}_J \in \mathcal{T}^{d_J}$ such that $\mathbf{p}_J(s) = p^s$ as defined above. In this new notation, we have

$$0 \leq \mathcal{L}^J(v)(s) - \mathcal{L}^J(u)(s) \leq \lambda \mathbf{E}_{\mathbf{p}_J(s)}\big[\mathcal{L}^{J-1}(v)(s_1) - \mathcal{L}^{J-1}(u)(s_1)\big] + 2\epsilon$$

Repeating the same argument gives us another decision rule d_{J-1} and a transition probability 'matrix' $\mathbf{p}_{J-1} \in \mathcal{T}^{d_{J-1}}$ such that

$$\mathcal{L}^{J-1}(v)(s_1) - \mathcal{L}^{J-1}(u)(s_1) \leq \lambda \mathbf{E}_{\mathbf{p}_{J-1}(s_1)}\big[\mathcal{L}^{J-2}(v)(s_2) - \mathcal{L}^{J-2}(u)(s_2)\big] + 2\epsilon$$

Thus,

$$
\begin{aligned}
0 &\leq \mathcal{L}^J(v)(s) - L^J(u)(s) \\
&\leq \lambda \mathbf{E}_{\mathbf{p}_J(s)} \Big[\lambda \mathbf{E}_{\mathbf{p}_{J-1}(s_1)} \big[\mathcal{L}^{J-2}(v)(s_2) - \mathcal{L}^{J-2}(u)(s_2) \big] + 2\epsilon \Big] + 2\epsilon \\
&= \lambda^2 \mathbf{E}_{(\mathbf{p}_J, \mathbf{p}_{J-1})} \big[\mathcal{L}^{J-2}(v)(s_2) - \mathcal{L}^{J-2}(u)(s_2) \big] + 2(\lambda + 1)\epsilon \\
&\ \ \vdots \\
&\leq \lambda^J \mathbf{E}_{(\mathbf{p}_J, \mathbf{p}_{J-1}, \ldots, \mathbf{p}_1)} \big[v(s_J) - u(s_J) \big] + 2(\lambda^{J-1} + \ldots + 1)\epsilon \\
&\leq \|v - u\|_w \Big(\lambda^J \mathbf{E}_{(\mathbf{p}_J, \mathbf{p}_{J-1}, \ldots, \mathbf{p}_1)} \big[w(s_J) \big] \Big) + 2(\lambda^{J-1} + \ldots + 1)\epsilon \\
&= \|v - u\|_w \Big(\lambda^J \sum_{s_J \in \mathcal{S}} \mathbf{P}_\sigma^J(s_J | s) w(s_J) \Big) + 2(\lambda^{J-1} + \ldots + 1)\epsilon,
\end{aligned}
$$

where $\sigma = (d_1, \ldots, d_J)$ and $\mathbf{P}_\sigma^J = \mathbf{p}_1 \ldots \mathbf{p}_J$. Then, by Assumption 4.2.3, it follows that

$$
|\mathcal{L}^J(v)(s) - \mathcal{L}^J(u)(s)| = \mathcal{L}^J(v)(s) - \mathcal{L}^J(u)(s) \leq \|v - u\|_w \alpha w(s) + 2(\lambda^{J-1} + \ldots + 1)\epsilon.
$$

A similar argument works by interchanging the roles of u and v above when $\mathcal{L}^J(u)(s) \geq \mathcal{L}^J(v)(s)$.

Since $\epsilon > 0$ was arbitrary, we conclude that

$$
\begin{aligned}
&|\mathcal{L}^J(v)(s) - \mathcal{L}^J(u)(s)| \leq \|v - u\|_w \ \alpha w(s) \ \text{ for all } s \in \mathcal{S}, \\
\Longrightarrow \quad &\|\mathcal{L}^J(v) - \mathcal{L}^J(u)\|_w \leq \alpha \|v - u\|_w.
\end{aligned}
$$

Thus, $\mathcal{L}^J$ is a contraction mapping on V_w.

For the second part of the theorem, we invoke a generalized version of the Banach contraction mapping theorem. The theorem states that an operator T on a Banach space X has a unique fixed point in X, if T^n is a contraction mapping for some $n \in \mathbb{N}$. Since $\mathcal{L}^J$ is a contraction, $\mathcal{L}$ has a unique fixed point in the Banach space V_w. Let us call that fixed point $\bar{v}$.

Next, we prove that $\bar{v}(s)$ must be the optimal value function defined in (4.1). We do so by comparing $\bar{v}$ with the value v^σ of an arbitrary policy $\sigma = (d_0, d_1, \ldots) \in \Pi$. Note that for any state $s \in \mathcal{S}$ and decision-rule d, we have

$$\mathcal{L}(v)(s) = \inf_{a \in \mathcal{A}(s)} \sup_{p \in \mathcal{P}_s^a} \mathbf{E}_p\big[c(s,a,s') + \lambda v(s')\big] \le \sup_{p \in \mathcal{P}_s^{d(s)}} \mathbf{E}_p\big[c(s,d(s),s') + \lambda v(s')\big].$$

We use this argument repeatedly in the following calculations. Fix a state $s_0 \in \mathcal{S}$. Also, let $\mathcal{P}_s^d$ denote $\mathcal{P}_s^{d(s)}$. By definition,

$$v^\sigma(s_0) = \sup_{\tau \in \mathcal{T}^\sigma} \mathbf{E}_\tau\big[\sum_{t=0}^{\infty} \lambda^t c(s_t, d_t(s_t), s_{t+1})\big].$$

Moreover,

$$\begin{aligned}
\bar{v}(s_0) &= \mathcal{L}(\bar{v})(s_0) \\
&= \inf_{a \in \mathcal{A}(s)} \sup_{p_0 \in \mathcal{P}_s^a} \mathbf{E}_{p_0}\big[c(s_0,a,s_1) + \lambda \bar{v}(s_1)\big] \\
&\le \sup_{p_0 \in \mathcal{P}_s^{d_0}} \mathbf{E}_{p_0}\big[c(s,d_0(s),s_1) + \lambda \bar{v}(s_1)\big] \\
&\le \sup_{p_0 \in \mathcal{P}_s^{d_0}} \mathbf{E}_{p_0}\Big[c(s,d_0(s),s_1) + \lambda\Big(\sup_{p_1 \in \mathcal{P}_{s_1}^{d_1}} \mathbf{E}_{p_1}\big[c(s_1,d_1(s_1),s_2) + \lambda \bar{v}(s_2)\big]\Big)\Big] \\
&\le \sup_{p_0 \in \mathcal{P}_s^{d_0}} \mathbf{E}_{p_0}\Big[c(s,d_0(s),s_1) + \lambda\Big(\sup_{\mathbf{p}_1 \in \mathcal{T}^{d_1}} \mathbf{E}_{\mathbf{p}_1}\big[c(s_1,d_1(s_1),s_2) + \lambda \bar{v}(s_2)\big]\Big)\Big].
\end{aligned}$$

Recall that $\mathbf{p}_1$ is a transition probability 'matrix' such that $\mathbf{p}_1(s) = p_1(\cdot|s, d_1(s))$. Then, using the fact that $\sup(f+g) \le \sup f + \sup g$ for any functions f and g, it follows that

$$\begin{aligned}
\bar{v}(s_0) &\le \sup_{p_0 \in \mathcal{P}_s^{d_0}} \sup_{\mathbf{p}_1 \in \mathcal{T}^{d_1}} \mathbf{E}_{p_0}\Big[\mathbf{E}_{\mathbf{p}_1}\big[c(s,d_0(s),s_1) + \lambda c(s_1,d_1(s_1),s_2) + \lambda^2 \bar{v}(s_2)\big]\Big] \\
&= \sup_{\substack{(\mathbf{p}_0,\mathbf{p}_1) \\ \in \mathcal{T}^{d_0} \times \mathcal{T}^{d_1}}} \mathbf{E}_{(\mathbf{p}_0,\mathbf{p}_1)}\big[c(s,d_0(s),s_1) + \lambda c(s_1,d_1(s_1),s_2) + \lambda^2 \bar{v}(s_2)\big].
\end{aligned}$$

Repeating this argument n times gives that

$$\begin{aligned}
\bar{v}(s_0) &\leq \sup_{\substack{(\mathbf{p}_0,\ldots,\mathbf{p}_n)\\ \in \mathcal{T}^{d_0}\times\ldots\times\mathcal{T}^{d_n}}} \mathbf{E}_{(\mathbf{p}_0,\ldots,\mathbf{p}_n)}\big[\sum_{t=0}^{n}\lambda^t c(s_t, d_t(s_t), s_{t+1}) + \lambda^{n+1}\bar{v}(s_{n+1})\big] \\
&\leq \sup_{\substack{(\mathbf{p}_0,\ldots,\mathbf{p}_n)\\ \in \mathcal{T}^{d_0}\times\ldots\times\mathcal{T}^{d_n}}} \mathbf{E}_{(\mathbf{p}_0,\ldots,\mathbf{p}_n)}\big[\sum_{t=0}^{n}\lambda^t c(s_t, d_t(s_t), s_{t+1})\big] + \sup_{\substack{(\mathbf{p}_0,\ldots,\mathbf{p}_n)\\ \in \mathcal{T}^{d_0}\times\ldots\times\mathcal{T}^{d_n}}} \mathbf{E}_{(\mathbf{p}_0,\ldots,\mathbf{p}_n)}\big[\lambda^{n+1}\bar{v}(s_{n+1})\big] \\
&= \sup_{\tau\in\mathcal{T}^\sigma} \mathbf{E}_\tau\big[\sum_{t=0}^{n}\lambda^t c(s_t, d_t(s_t), s_{t+1})\big] + \sup_{\substack{(\mathbf{p}_0,\ldots,\mathbf{p}_n)\\ \in \mathcal{T}^{d_0}\times\ldots\times\mathcal{T}^{d_n}}} \mathbf{E}_{(\mathbf{p}_0,\ldots,\mathbf{p}_n)}\big[\lambda^{n+1}\bar{v}(s_{n+1})\big] \\
&\leq \sup_{\tau\in\mathcal{T}^\sigma} \mathbf{E}_\tau\big[\sum_{t=0}^{n}\lambda^t c(s_t, d_t(s_t), s_{t+1})\big] + \lambda^{n+1}\|\bar{v}\|_w \sup_{\substack{(\mathbf{p}_0,\ldots,\mathbf{p}_n)\\ \in \mathcal{T}^{d_0}\times\ldots\times\mathcal{T}^{d_n}}} \mathbf{E}_{(\mathbf{p}_0,\ldots,\mathbf{p}_n)}\big[w(s_{n+1})\big].
\end{aligned}$$

In particular, for $n = J-1$,

$$\begin{aligned}
\bar{v}(s_0) &\leq \sup_{\tau\in\mathcal{T}^\sigma} \mathbf{E}_\tau\big[\sum_{t=0}^{J-1}\lambda^t c(s_t, d_t(s_t), s_{t+1})\big] + \lambda^J\|\bar{v}\|_w \sup_{\substack{(\mathbf{p}_0,\ldots,\mathbf{p}_{J-1})\\ \in \mathcal{T}^{d_0}\times\ldots\times\mathcal{T}^{d_{J-1}}}} \mathbf{E}_{(\mathbf{p}_0,\ldots,\mathbf{p}_{J-1})}\big[w(s_J)\big] \\
&\leq \sup_{\tau\in\mathcal{T}^\sigma} \mathbf{E}_\tau\big[\sum_{t=0}^{J-1}\lambda^t c(s_t, d_t(s_t), s_{t+1})\big] + \|\bar{v}\|_w \alpha w(s),
\end{aligned}$$

where the last inequality follows from Assumption 4.2.3. In fact, for any positive integer m and $n = mJ-1$, we have

$$\bar{v}(s_0) \leq \sup_{\tau\in\mathcal{T}^\sigma} \mathbf{E}_\tau\big[\sum_{t=0}^{mJ-1}\lambda^t c(s_t, d_t(s_t), s_{t+1})\big] + \|\bar{v}\|_w \alpha^m w(s_0).$$

Let $m \to \infty$ to obtain

$$\bar{v}(s) \leq \sup_{\tau\in\mathcal{T}^\sigma} \mathbf{E}_\tau\big[\sum_{t=0}^{\infty}\lambda^t c(s_t, d_t(s_t), s_{t+1})\big] \; = v^\sigma(s) \;\text{ for all } s \in \mathcal{S}.$$

Since the policy $\sigma \in \Pi$ was arbitrary, we have

$$\bar{v}(s_0) \leq \inf_{\sigma \in \Pi} v^{\sigma}(s_0) \;\text{ for all } s_0 \in \mathcal{S}. \tag{4.8}$$

We now justify that this inequality cannot be strict. Given $\epsilon > 0$, we show that there there exists a policy σ such that $\bar{v}(s) \geq v^{\sigma}(s) - \epsilon/(1-\lambda)$. For any state s, there exists an action $a(s) \in \mathcal{A}(s)$ such that

$$\bar{v}(s) = \mathcal{L}(\bar{v})(s) > \sup_{\mathbf{p} \in \mathcal{P}_s^{a(s)}} \mathbf{E}_{\mathbf{p}}\big[c(s, a(s), s') + \lambda \bar{v}(s')\big] - \epsilon.$$

We use these actions $a(s)$ to define a decision-rule $d : \mathcal{S} \to \mathcal{A}(s)$, with $d(s) = a(s)$. Further, define a stationary policy $\sigma = (d, d, \ldots) \in \Pi$. Fix a state $s_0 \in \mathcal{S}$. Then, by construction,

$$\begin{aligned}
\bar{v}(s_0) &> \sup_{\mathbf{p}_0 \in \mathcal{T}^d} \mathbf{E}_{\mathbf{p}_0}\Big[c(s, d(s), s_1) + \lambda \bar{v}(s_1)\Big] - \epsilon \\
&\geq \sup_{\mathbf{p}_0 \in \mathcal{T}^d} \mathbf{E}_{\mathbf{p}_0}\Big[c(s, d(s), s_1) + \lambda\Big(\sup_{\mathbf{p}_1 \in \mathcal{T}^d} \mathbf{E}_{\mathbf{p}_1}\big[c(s_1, d(s_1), s_2) + \lambda \bar{v}(s_2)\big] - \epsilon\Big)\Big] - \epsilon \\
&= \sup_{\mathbf{p}_0 \in \mathcal{T}^d} \mathbf{E}_{\mathbf{p}_0}\Big[c(s, d(s), s_1) + \lambda\Big(\sup_{\mathbf{p}_1 \in \mathcal{T}^d} \mathbf{E}_{\mathbf{p}_1}\big[c(s_1, d(s_1), s_2) + \lambda \bar{v}(s_2)\big]\Big)\Big] - (1+\lambda)\epsilon \\
&= \sup_{\substack{(\mathbf{p}_0, \mathbf{p}_1) \\ \in \mathcal{T}^d \times \mathcal{T}^d}} \mathbf{E}_{(\mathbf{p}_0, \mathbf{p}_1)}\Big[c(s, d(s), s_1) + \lambda c(s_1, d(s_1), s_2) + \lambda^2 \bar{v}(s_2)\Big]\Big)\Big] - (1+\lambda)\epsilon.
\end{aligned}$$

Repeating this argument n times gives

$$\begin{aligned}
\bar{v}(s_0) &\geq \sup_{\substack{(\mathbf{p}_0, \ldots, \mathbf{p}_n) \\ \in \mathcal{T}^d \times \ldots \times \mathcal{T}^d}} \mathbf{E}_{(\mathbf{p}_0, \ldots, \mathbf{p}_n)}\Big[\sum_{t=0}^{n} \lambda^t c(s_t, d(s_t), s_{t+1}) + \lambda^{n+1} \bar{v}(s_{n+1})\Big] - (1 + \ldots + \lambda^n)\epsilon \\
&\geq \sup_{\substack{(\mathbf{p}_0, \ldots, \mathbf{p}_n) \\ \in \mathcal{T}^d \times \ldots \times \mathcal{T}^d}} \mathbf{E}_{(\mathbf{p}_0, \ldots, \mathbf{p}_n)}\Big[\sum_{t=0}^{n} \lambda^t c(s_t, d(s_t), s_{t+1})\Big] \\
&\quad - \sup_{\substack{(\mathbf{p}_0, \ldots, \mathbf{p}_n) \\ \in \mathcal{T}^d \times \ldots \times \mathcal{T}^d}} \mathbf{E}_{(\mathbf{p}_0, \ldots, \mathbf{p}_n)}\Big[\lambda^{n+1}(-\bar{v}(s_{n+1})\Big] - (1 + \ldots + \lambda^n)\epsilon.
\end{aligned}$$

Here, we have used the property that $\sup(f-g) \geq \sup f - \sup g$ for any functions f and g. Since $-\bar{v}(s_{n+1}) \leq |\bar{v}(s_{n+1})| \leq \|\bar{v}\|_w w(s_{n+1})$, it follows that

$$\begin{aligned}\bar{v}(s_0) \geq & \sup_{\substack{(\mathbf{p}_0,\ldots,\mathbf{p}_n)\\ \in\mathcal{T}^d\times\ldots\times\mathcal{T}^d}} \mathbf{E}_{(\mathbf{p}_0,\ldots,\mathbf{p}_n)}\Big[\sum_{t=0}^{n}\lambda^t c(s_t, d(s_t), s_{t+1})\Big] \\ & - \lambda^{n+1}\|\bar{v}\|_w \sup_{\substack{(\mathbf{p}_0,\ldots,\mathbf{p}_n)\\ \in\mathcal{T}^d\times\ldots\times\mathcal{T}^d}} \mathbf{E}_{(\mathbf{p}_0,\ldots,\mathbf{p}_n)}\big[w(s_{n+1})\big] - (1+\ldots+\lambda^n)\epsilon.\end{aligned}$$

Once again, for $n = mJ-1$, we have

$$\bar{v}(s_0) \geq \sup_{\substack{(\mathbf{p}_0,\ldots,\mathbf{p}_n)\\ \in\mathcal{T}^d\times\ldots\times\mathcal{T}^d}} \mathbf{E}_{(\mathbf{p}_0,\ldots,\mathbf{p}_n)}\Big[\sum_{t=0}^{n}\lambda^t c(s_t, d(s_t), s_{t+1})\Big] - \lambda^{mJ}\|\bar{v}\|_w\alpha^m w(s_0)\big] - (1+\ldots+\lambda^n)\epsilon.$$

Letting $m \to \infty$, we have

$$\bar{v}(s_0) \geq \sup_{\tau\in\mathcal{T}^\sigma} \mathbf{E}_\tau\big[\sum_{t=0}^{\infty}\lambda^t c(s_t, d(s_t), s_{t+1})\big] - \frac{\epsilon}{1-\lambda}.$$

Hence,

$$\bar{v}(s) \geq v^\sigma(s) - \frac{\epsilon}{1-\lambda} \quad \text{for all } s \in \mathcal{S}. \tag{4.9}$$

Thus, there exists a policy σ whose value is at most $\epsilon/(1-\lambda)$ greater than $\bar{v}$. Since $\epsilon > 0$ was arbitrary, we conclude that $\bar{v}$ must be equal to the infimum in (4.8). This completes the proof. □

As in [28], this theorem leads to two useful corollaries. The first states that the value of a statioanry policy can be computed using a robust Bellman evaluation operator. The second result states that given $\epsilon > 0$, there exists a stationary policy such whose value is at most ϵ greater than the optimal. Consequently, it is sufficient to solve the MDPs over stationary policies alone.

Corollary 4.3.3. *(a) Given a decision-rule d, the value of the stationary policy $\sigma = (d, d, \ldots)$ is given by*

$$v^{\sigma}(s) = \sup_{p \in \mathcal{P}_s^d} \mathbf{E}_p[c(s, d(s), s') + \lambda v^{\sigma}(s')], \qquad \text{for all } s \in \mathcal{S}.$$

(b) For all $\epsilon > 0$, there exists a decision-rule d and a stationary policy $\sigma = (d, d, \ldots)$ such that

$$v^*(s) \geq v^{\sigma}(s) - \epsilon.$$

Proof. The first result follows from Theorem 4.3.2 by choosing $\mathcal{A}(s) = \{d(s)\}$ for all s. The second result was established in the proof of Theorem 4.3.2; see Equation (4.9) and note that $\bar{v} = v^*$. □

Thus, we have established that the optimal value function for a robust MDP with unbounded immediate costs can be recovered from the robust Bellman equations. Moreover, an optimal policy can then be constructed by choosing actions from the argmin set in the Bellman equations for each state. We expect that the natural methods of policy and value iteration can be employed here as well, but there convergence behavior remains to be verified. The standard methods would also encounter implementability issues as in the bounded-cost case. As such, the eventual goal is to devise a convergent, implementable approximate policy iteration algorithm akin to the previous chapters, for this class of problems as well. Some immediate hurdles in this task arise from the fact that all expectations in the technical results from Chapter 2 relied on the exponential decay of the λ^T terms. This role is now played by the parameter α, so the expectations have to be dealt with in increments of J. This further complicates the algorithmic computation of a bound like $\bar{\delta}$ that guarantees sufficient improvement in the policy update step. Nonetheless, developing a practical way of solving unbounded-cost robust MDPs will be a valuable generalization of this work.

Chapter 5

A ROBUST MULTI-PERIOD NEWSVENDOR MODEL WITH INVENTORY-BALANCE CONSTRAINTS

5.1 Introduction

In recent years, robust optimization has steadily gained in popularity as a successful approach to difficult problems of optimization under uncertainty; an overview of robust optimization can be found in Ben-Tal et al. [9]. An area that has benefited significantly is inventory management, and our paper contributes to this stream. We study a generic multiple period inventory management problem that maximizes profit in a Newsvendor framework with inventory balance constraints, where sale revenues as well as ordering, holding, and shortage costs are captured. Our uncertainty sets are generic, but they can be parameterized to form various uncertainty sets, motivated by probabilistic limit theorems, that have recently gained popularity. Notably, we are able to derive closed-form solutions for our general problem. Furthermore, our model can be applied in both a static setting as well as in a dynamic rolling horizon manner. Our paper is related to four streams of literature: 1) robust inventory cost minimization, 2) robust newsvendor models, 3) design of robust uncertainty sets, and 4) dynamic robust optimization. We position our research relative to the most relevant papers in each of these streams.

5.1.1 Robust Inventory Cost Minimization

One of the first robust inventory management models, focused on minimizing cumulative ordering, holding and shortage costs over a finite horizon, is Bertsimas and Thiele [11], which applies the fundamental constructs of Bertsimas and Sim [10], such as “budgets of uncertainty”. Bienstock and Özbay [14] extends Bertsimas and Thiele [11] in various directions.

Chen et al. [18] studies generic robust uncertainty sets allowing for asymmetry and See and Sim [39] analyzes a "factor-based" model of uncertainty; both these approaches result in a non-robust second-order cone counterpart. Wagner [44] studies a similar cost minimization problem, except that the only property known about demand is non-negativity. Ardestani-Jaafari and Delage [2], extending the ideas from Gorissen and Hertog [24], analyzes more general robust optimization problems involving sums of piecewise linear functions, which can be applied to inventory management problems. Mamani et al. [31] studies a similar problem, except that the uncertainty sets are motivated by the central limit theorem, which results in closed-form solutions. Solyalı et al. [41] proposes a new robust formulation of inventory control based on ideas from facility location. Wagner [45] provides a continuous-time formulation of a similar problem, where the uncertainty set is motivated by the strong law of large numbers. Our paper differs from this stream in that we introduce revenues into the models via a Newsvendor approach, so that we maximize profit rather than minimize cost.

5.1.2 Robust Newsvendor Models

A popular approach for studying a robust Newsvendor model is to apply distributionally robust optimization: one assumes that the mean and variance of demand are known, but the distribution is not, and a max-min approach over all probabilistic distributions that have the given mean and variance is applied. Scarf [36] derives the optimal order quantity for the Newsvendor Problem under this scenario. Perakis and Roels [34] extends this setup to allow more (or less) information to be known about the demand distribution under a regret formulation and Natarajan et al. [32] analyzes similar extensions under a max-min approach over multiple products. Further examples of this style of research can be found in the comprehensive literature review of Natarajan et al. [32]. Our paper differs from this stream in that we apply uncertainty-set robust optimization techniques, rather than distributionally robust techniques; the value of our approach is that we can introduce and tractably analyze inventory balance constraints for a multiple period Newsvendor model. Note that Vairaktarakis [42] also utilizes uncertainty-set robust optimization to study the

Newsvendor problem, under either interval demand uncertainty or under a discrete set of demand scenarios; however, this paper only considers a single period and inventory balance constraints are not considered.

5.1.3 Design of Robust Uncertainty Sets

In early robust optimization papers, the uncertainty sets were selected to be interval, polyhedral, ellipsoidal, or, more generally, simply convex. More recently, researchers have attempted to design uncertainty sets that mimic the structure of limit theorems of probability. Bertsimas et al. [13] analyzes queuing networks with a robust uncertainty set motivated by the probabilistic law of the iterated logarithm. Bandi and Bertsimas [3] focuses on studying uncertainty sets motivated by the central limit theorem, which is applied in detailed investigations of option pricing [5], auction design [4], queueing theory [6, 7], and inventory management [31]. Wagner [45] uses the strong law of large numbers to motivate an uncertainty set in an inventory cost minimization context. Our paper continues this stream in that we consider a generic uncertainty set, which can be parameterized to result in many of the above sets.

5.1.4 Dynamic Robust Optimization

Ben-Tal et al. [8] studies an adjustable robust optimization problem, where variable values can be changed once unknowns become realized; this problem is shown to be NP-hard. Consequently, researchers have focused on approximations, such as an affinely adjustable robust optimization model, where the focus is to find optimal policies that are affine in the uncertain parameters. Bertsimas et al. [12] proves the optimality of affine policies for a general class of multi-stage robust optimization models where unknown parameters are constrained to lie in intervals; this research is extended by Iancu et al. [27], which more fully characterizes the problem structures where affine policies are optimal. Another avenue of approximation is to apply a static model in a rolling horizon framework. Mamani et al. [31] takes this approach in an inventory cost minimization context, and exhibits better performance than Bertsimas and Thiele [11] and Bertsimas et al. [12] under correlated demands. Solyalı et al. [41] also takes this approach, which outperforms Bertsimas and Thiele [11], Ben-Tal et al. [8], See and Sim [39], and others. Wagner [45] studies three rolling horizon contexts, which depend on whether or not the observed demand stream is consistent with the original robust uncertainty set. In our paper we adopt this rolling horizon approach to design dynamic strategies, though we are the first to do so for a multiple period Newsvendor problem with inventory balance constraints.

The only paper, to our knowledge, that combines a non-stochastic Newsvendor framework with a multiple period setting and inventory balance constraints is Wagner [43]; however, this paper approaches the problem via online optimization where nothing is known about demand other than non-negativity, which leads to overly conservative solutions. In contrast, our model allows the introduction of partial knowledge of demand, such as the mean and standard deviation; furthermore, our model admits parameters than can control the conservatism of the solution.

5.2 Model

Consider a seller managing the inventory of a single product over n periods. For $j \in \{1, 2, \dots, n\}$, let $d_j \geq 0$ be the demand for this product in the j-th period, and let $q_j \geq 0$ be the amount of new product that the seller purchases in the same period. Then the inventory level I_j at the end of this period is $I_j = I_{j-1} + q_j - d_j$, where the initial inventory level I_0 is assumed to be known. I_j can take any sign; a negative value of I_j indicates shortage while a positive inventory level implies a surplus. Both unmet demands and surplus inventory are carried over to the next period. Thus, the total demand which must be satisfied in period j is d_j plus any demand which has been carried over from the previous period. Similarly, the total amount of the product available for sale in period j is the sum of q_j and any inventory being held from period $j-1$. Therefore, the amount of inventory sold in period j is $\min\{d_j + I_{j-1}^-, q_j + I_{j-1}^+\}$ (where $a^+ = \max\{a, 0\}$, $a^- = \max\{-a, 0\}$ and $a = a^+ - a^-$ for any $a \in \mathbb{R}$). Let $q = (q_1, \dots, q_n)$ be the vector of order quantities, and $d = (d_1, \dots, d_n)$ be the demand vector. Also, let $Q_j = \sum\limits_{i=1}^{j} q_i$ and $D_j = \sum\limits_{i=1}^{j} d_i$ be the cumulative order quantity and cumulative demand up to period j respectively. If $r \geq 0$ is the sale revenue per unit, then the total revenue accrued over n periods is

$$\begin{aligned} R(q,d) &= \sum_{j=1}^{n} r \cdot \min\{d_j + I_{j-1}^{-}, q_j + I_{j-1}^{+}\} = r\Big(\sum_{j=1}^{n} \min\{0, -I_j\} + q_j + I_{j-1}^{+}\Big) \\ &= r\Big(\sum_{j=1}^{n} -I_j^{+} + q_j + I_{j-1}^{+}\Big) = r\Big(I_0^{+} + Q_n - I_n^{+}\Big). \end{aligned} \tag{5.1}$$

Next, we compute the total cost incurred over the entire horizon. The seller tries to avoid both excess inventory as well as shortage, and this is modeled by defining costs associated with both these phenomena. Suppose a holding cost of $h \geq 0$ per unit per period is incurred whenever the order quantities exceed the demand and the inventory level is positive. Similarly, let $s \geq 0$ be the shortage cost per unit per period which applies only when the demand from period j is not satisfied and $I_j < 0$. Finally, let $c \geq 0$ be the ordering cost per unit. Then, the total cost for order quantities $q = (q_1, \ldots, q_n)$ and demand $d = (d_1, \ldots, d_n)$, is

$$C(q,d) = \sum_{j=1}^{n} cq_j + hI_j^{+} + sI_j^{-}. \tag{5.2}$$

The seller's objective is to maximize profit. The profit earned over the entire horizon is $\Pi(q,d) = R(q,d) - C(q,d)$. The following lemma computes a convenient expression for the total profit which helps us formulate the profit maximization problem as a linear program (LP).

Lemma 5.2.1 (Profit function)**.** *For a demand vector d and order quantities q, define*

$$y_j = \max\{(h + \delta_{jn}c)(Q_j - D_j + I_0), (s + \delta_{jn}(r-c))(D_j - Q_j - I_0)\}, \quad j = 1, \ldots, n, \tag{5.3}$$

where δ_{jn} is the kronecker delta which takes the value 1 when $j = n$, and 0 otherwise. Then, *the profit function is given by*

$$\Pi(q,d) = cI_0^+ + (r-c)I_0^- + (r-c)D_n - \sum_{j=1}^{n} y_j. \tag{5.4}$$

Proof. Proof:

$$\begin{aligned}
\Pi(q,d) &= R(q,d) - C(q,d) \\
&= r(I_0^+ + Q_n - I_n^+) - cQ_n - \sum_{j=1}^{n}(hI_j^+ + sI_j^-) \\
&= rI_0^+ + (r-c)Q_n - rI_n^+ - \sum_{j=1}^{n}(hI_j^+ + sI_j^-) \\
&= rI_0^+ + (r-c)(D_n - I_0) + (r-c)(Q_n - D_n + I_0) - rI_n^+ - \sum_{j=1}^{n}(hI_j^+ + sI_j^-) \\
&= rI_0^+ - (r-c)I_0 + (r-c)D_n + (r-c)I_n - rI_n^+ - \sum_{j=1}^{n}(hI_j^+ + sI_j^-) \\
&= rI_0^+ - (r-c)(I_0^+ - I_0^-) + (r-c)D_n + (r-c)(I_n^+ - I_n^-) - rI_n^+ - \sum_{j=1}^{n}(hI_j^+ + sI_j^-) \\
&= cI_0^+ + (r-c)I_0^- + (r-c)D_n - cI_n^+ - (r-c)I_n^- - \sum_{j=1}^{n}(hI_j^+ + sI_j^-) \\
&= cI_0^+ + (r-c)I_0^- + (r-c)D_n - \sum_{j=1}^{n}((h + \delta_{jn}c)I_j^+ + (s + \delta_{jn}(r-c))I_j^-) \\
&= cI_0^+ + (r-c)I_0^- + (r-c)D_n - \sum_{j=1}^{n}\max\{(h + \delta_{jn}c)I_j, -(s + \delta_{jn}(r-c))I_j\}.
\end{aligned}$$

Substituting $I_j = Q_j - D_j + I_0$ for all j, we rewrite

$$\begin{aligned}
\Pi(q,d) = {} & cI_0^+ + (r-c)I_0^- + (r-c)D_n \\
& - \sum_{j=1}^{n}\max\{(h + \delta_{jn}c)(Q_j - D_j + I_0), (s + \delta_{jn}(r-c))(D_j - Q_j - I_0)\}.
\end{aligned}$$

Finally, using the definition of y_j from (5.3) gives the desired result.

$\square$

The seller's goal is to find the optimal quantity to order in each period so as to maximize the total profit. The terms outside the summation in (5.4) are constant for any q, and the optimal order quantity is independent of these terms. Therefore, maximizing the profit is equivalent to solving $\left(\min \sum_{j=1}^{n} y_j\right)$ over all feasible vectors $q \geq 0$. Intuitively, the seller tries to simultaneously minimize both the terms inside the max in (5.3). The first term, being positive only when $Q_j + I_0 > D_j$, indicates the seller's aversion to ordering too much inventory which would incur a holding cost. Since the purchase cost for each period gets added up, the parameter c appears only in the last period. Similarly, the second term is positive only when $Q_j + I_0 < D_j$, that is, when the quantity ordered in the first j periods (plus the initial inventory) is insufficient to meet the demand in those periods. This is weighted by the shortage cost. Once again, the term $(r - c)$ appears only in period n to penalize any lost revenue when the total demand D_n exceeds the total quantity $Q_n + I_0$ of the product. Thus, the profit maximization problem can be formulated as the following LP.

$$\begin{aligned} \min \quad & \sum_{j=1}^{n} y_j \\ \text{s.t.} \quad & y_j \geq (h + \delta_{jn} c)(Q_j + I_0 - D_j), \; j = 1, \ldots, n, \\ & y_j \geq (s + \delta_{jn}(r - c))(D_j - Q_j - I_0) \; j = 1, \ldots, n, \\ & Q_n \geq Q_{n-1} \geq \ldots \geq Q_1 \geq 0. \end{aligned} \tag{5.5}$$

Note that we can drop the variables q_j since they can be recovered from Q_j. The nonnegativity of q_j is ensured by the constraints $Q_j \geq Q_{j-1}$ for all j.

In the simplest model, we can assume that the demands are known and solving (5.5) will yield the optimal order quantities ($Q_j = (D_j - I_0)^+$ for all j). This, however, is unrealistic in practice, as the true demand is almost never known *a priori*. Traditionally, this is resolved by assuming that the parameters D_j in (5.5) are random variables with known distributions.

This gives rise to the classical stochastic newsvendor model, and optimal order quantities are recovered by solving (5.5) through stochastic optimization techniques.

The demand distributions in the stochastic model are often constructed from historical data using heuristics and statistical methods, and may not be exact. These errors, in turn, may lead to suboptimal solutions. A robust optimization model accounts for this by instead assuming that the unknown demand vector d simply lies in some set of plausible values called the uncertainty set Ω. A robust optimization variant of the profit maximization problem in (5.5) can be formulated as follows.

$$\begin{aligned} \min \quad & \sum_{j=1}^{n} y_j \\ \text{s.t.} \quad & y_j \geq (h + \delta_{jn} c)(Q_j - D_j + I_0),\ j = 1, \dots, n,\ \forall\ d \in \Omega, \\ & y_j \geq (s + \delta_{jn}(r - c))(D_j - Q_j - I_0)\ j = 1, \dots, n,\ \forall\ d \in \Omega, \\ & Q_n \geq Q_{n-1} \geq \dots \geq Q_1 \geq 0. \end{aligned} \tag{5.6}$$

We remark that there are multiple ways of constructing a robust newsvendor model; hence we only talk about ‘a’ (and not ‘the’) robust counterpart of (5.5). We define uncertainty per constraint, which is the more common approach in the literature (e.g., Bertsimas and Sim [10], Bertsimas and Thiele [11], Bienstock and Özbay [14], Ben-Tal et al. [8], Bertsimas et al. [12], Mamani et al. [31], Solyalı et al. [41], Wagner [45], etc.). In this case, the terms y_j have a natural interpretation with respect to balancing the worst-case costs due to excess as well as insufficient order quantities. Moreover, this leads to closed-form solutions with intuitive properties and structure, as discussed in Section 5.3. However, other researchers have focused on determining a single worst-case demand instance per model, as in Gorissen and Hertog [24] and Ardestani-Jaafari and Delage [2].

Our robust model (5.6) has a linear objective function, and linear constraints, but the constraints are indexed by $d \in \Omega$ – there are infinitely many of them. However, we observe that for any j,

$$y_j \geq (h + \delta_{jn}c)(Q_j - D_j + I_0) \ \forall \ d \in \Omega \iff y_j \geq \max_{d \in \Omega} \ (h + \delta_{jn}c)(Q_j - D_j + I_0). \qquad (5.7)$$

The same is also true for the second set of constraints. Let $\underline{D}_j = \min_{d \in \Omega} \sum_{i=1}^{j} d_i$ and $\overline{D}_j = \max_{d \in \Omega} \sum_{i=1}^{j} d_i$ be the smallest and largest possible cumulative demands up to period j respectively. Then, using the observation in (5.7), the robust math program (5.6) can be reformulated as the non-robust linear program

$$\begin{aligned} \min \quad & \sum_{j=1}^{n} y_j \\ \text{s.t.} \quad & y_j \geq (h + \delta_{jn}c)(Q_j + I_0 - \underline{D}_j), \ j = 1, \dots, n, \\ & y_j \geq (s + \delta_{jn}(r - c))(\overline{D}_j - Q_j - I_0) \ j = 1, \dots, n, \\ & Q_n \geq Q_{n-1} \geq \dots \geq Q_1 \geq 0. \end{aligned}$$

This is equivalent to solving $\min_{Q \geq 0} \sum_{j=1}^{n} y_j$ where

$$y_j = \max\{(h + \delta_{jn}c)(Q_j + I_0 - \underline{D}_j), (s + \delta_{jn}(r - c))(\overline{D}_j - Q_j - I_0)\}, \qquad j = 1, \dots, n.$$

The terms inside the max are worst-case analogues of those in the nominal case in (5.3). The first term corresponds to a penalty for exceeding even the smallest possible demand, while the second term indicates a cost incurred when the quantity ordered is insufficient for the maximum possible demand. Finally, we define new variables $\tilde{Q}_j = Q_j + I_0$ for all

$j = 1, \dots, n$, and $\tilde{Q} = (\tilde{Q}_1, \dots, \tilde{Q}_n)$ to formulate the equivalent LP

$$\begin{aligned}
\min_{\tilde{Q} \geq 0} \quad & \sum_{j=1}^{n} y_j \\
\text{s.t.} \quad & y_j \geq (h + \delta_{jn} c)(\tilde{Q}_j - \underline{D}_j), \; j = 1, \dots, n, \\
& y_j \geq (s + \delta_{jn}(r - c))(\overline{D}_j - \tilde{Q}_j) \; j = 1, \dots, n, \\
& \tilde{Q}_n \geq \tilde{Q}_n \geq \dots \tilde{Q}_1 \geq I_0.
\end{aligned} \tag{5.8}$$

5.3 Closed-form Solutions for an Arbitrary Uncertainty Set

In this section, we provide closed-form expressions for the optimal solution to the robust newsvendor model (5.8), along with a proof of optimality. Here, we treat $\underline{D}_j$ and $\overline{D}_j$, $j = 1, \dots, n$, as known constants. Section 5.3.2 elaborates on how these may be obtained analytically for some uncertainty sets which commonly arise in practice.

For any j, y_j is the maximum of two linear functions of $\tilde{Q}$, and we first find the points of intersections of these straight lines by equating the right-hand-sides of the first two inequalities. For any j,

$$\begin{aligned}
(h + \delta_{jn} c)(\tilde{Q}_j - \underline{D}_j) &= (s + \delta_{jn}(r - c))(\overline{D}_j - \tilde{Q}_j) \\
\implies (s + h + \delta_{jn} r)\tilde{Q}_j &= (s + \delta_{jn}(r - c))\overline{D}_j + (h + \delta_{jn} c)\underline{D}_j \\
\implies \tilde{Q}_j &= \frac{(s + \delta_{jn}(r - c))\overline{D}_j + (h + \delta_{jn} c)\underline{D}_j}{s + h + \delta_{jn} r}.
\end{aligned}$$

Define

$$\overline{Q}_j = \begin{cases} \frac{h\underline{D}_j + s\overline{D}_j}{s+h}, & j < n \\ \frac{(h+c)\underline{D}_j + (s+r-c)\overline{D}_j}{s+h+r}, & j = n. \end{cases}$$

We emphasize that $\overline{Q}_j$ are constants which can be computed once $\underline{D}_j$ and $\overline{D}_j$ are known. They will serve as candidates for the optimal cumulative order quantity in period j when such a solution is feasible. They also have certain properties that are desirable in the optimal solution. In particular, we expect the optimal order quantities to be increasing in the per unit revenue r and the shortage cost s, but decreasing in the purchase cost c and holding cost h; these properties hold for $\overline{Q}_j$. Furthermore, note that $\overline{Q}_j \geq \overline{Q}_{j-1}$ for all $j = 2, \ldots, n-1$, since $\underline{D}_{j-1} \leq \underline{D}_j$ and $\overline{D}_{j-1} \leq \overline{D}_j$. However, we do not necessarily have $\overline{Q}_n \geq \overline{Q}_{n-1}$. We may also have $\overline{Q}_1 < I_0$. Hence, the solution $\tilde{Q}_j = \overline{Q}_j \ \forall\ j$ may not be feasible. Nonetheless, the quantities $\overline{Q}_j$ are used to define an optimal solution to (5.8) as shown below. Let $\tilde{Q}^* = (\tilde{Q}_1^*, \ldots, \tilde{Q}_n^*)$ denote this solution.

Theorem 5.3.1. *Let k be the largest integer in $\{1, 2, \ldots, n\}$ for which $\overline{Q}_{k-1} \leq \overline{Q}_n$ (where $\overline{Q}_0 = \min\{I_0, 0\}$).*

(A) If $(n-k)s \leq h + c$, then the optimal solution to (5.8) is

$$\tilde{Q}_j^* = \begin{cases} \max\{\overline{Q}_j, I_0\}, & j = 1, \ldots, k-1, \\ \max\{\overline{Q}_n, I_0\}, & j = k, \ldots, n. \end{cases}$$

(B) If $(n-k)s > h+c$, let m be the smallest integer in $\{k, \ldots, n-1\}$ for which $(n-m-1)s \leq h + c$. Then the optimal solution to (5.8) is

$$\tilde{Q}_j^* = \begin{cases} \max\{\overline{Q}_j, I_0\}, & j = 1, \ldots, m-1, \\ \max\{\overline{Q}_m, I_0\}, & j = m, \ldots, n. \end{cases}$$

Proof. Proof of Theorem 5.3.1: Note that k and m always exist. The theorem will be proved using duality. The dual of (5.8) is

$$\begin{aligned} \max \quad & \sum_{i=1}^{n} -(h + \delta_{in} c)\underline{D}_i u_i + (s + \delta_{in}(r-c))\overline{D}_i v_i + I_0 w_1 && (5.9) \\ \text{s.t.} \quad & -h u_i + s v_i + w_i - w_{i+1} \leq 0, \ i = 1, \ldots, n-1, && (5.10) \\ & -(h+c)u_n + (s + r - c)v_n + w_n \leq 0, && (5.11) \\ & u_i + v_i = 1, \ i = 1, \ldots, n, \\ & u_i, v_i, w_i \geq 0, \ i = 1, \ldots, n. \end{aligned}$$

Since $v_i = 1 - u_i$ for all i, constraints (5.10) and (5.11) can be rewritten as

$$-(s+h)u_i + s + w_i - w_{i+1} \leq 0,\ i = 1, \ldots, n-1, \tag{5.12}$$

$$-(s+h+r)u_n + s + r - c + w_n \leq 0. \tag{5.13}$$

The primal LP has primary variables $\tilde{Q}_j$ and auxiliary variables y_j which, given $\tilde{Q}$, can be solved for as below.

$$\begin{aligned}
y_j &= \max\Big\{(h+\delta_{jn}c)(\tilde{Q}_j - \underline{D}_j), (s+\delta_{jn}(r-c))(\overline{D}_j - \tilde{Q}_j)\Big\} \\
&= \max\Big\{(h+\delta_{jn}c)(\tilde{Q}_j - \underline{D}_j) - (s+\delta_{jn}(r-c))(\overline{D}_j - \tilde{Q}_j), 0\Big\} + (s+\delta_{jn}(r-c))(\overline{D}_j - \tilde{Q}_j) \\
&= (s+h+\delta_{jn}r)\max\Big\{\tilde{Q}_j - \frac{(h+\delta_{jn}c)\underline{D}_j) + (s+\delta_{jn}(r-c))(\overline{D}_j]}{(s+h+\delta_{jn}r)}, 0\Big\} \\
&\quad + (s+\delta_{jn}(r-c))(\overline{D}_j - \tilde{Q}_j) \qquad (5.14) \\
&= \begin{cases} (s+\delta_{jn}(r-c))(\overline{D}_j - \tilde{Q}_j), & \text{if } \tilde{Q}_j \leq \overline{Q}_j, \\ (h+\delta_{jn}c)(\tilde{Q}_j - \underline{D}_j), & \text{if } \tilde{Q}_j \geq \overline{Q}_j, \end{cases} \quad j = 1, \ldots, n. \qquad (5.15)
\end{aligned}$$

We will prove the theorem separately for cases (A) and (B).

In case (A), k is the largest index in $\{1, \ldots, n\}$ so that $\overline{Q}_{k-1} \leq \overline{Q}_n$, and $(n-k)s \leq h + c$. We will consider three sub-cases.

Case A-1: $I_0 < \overline{Q}_n$.

Then, let l be the largest index in $\{0, 1, \ldots, k-1\}$ for which $I_0 \geq \overline{Q}_l$. Therefore, $\overline{Q}_l \leq I_0 < \overline{Q}_{l+1}$. The proposed solution takes the form

$$\tilde{Q}_j^* = \begin{cases} I_0, & 1 \leq j \leq l, \\ \overline{Q}_j, & l < j < k, \\ \overline{Q}_n, & k \leq j. \end{cases}$$

Note that $\tilde{Q}_j^* \geq \overline{Q}_j$ for $1 \leq j \leq l$, and $\tilde{Q}_j^* \leq \overline{Q}_j$ for $j > l$. Therefore, we use Equation (5.15) to compute

$$y_j = \begin{cases} (h + \delta_{jn} c)(\tilde{Q}_j^* - \underline{D}_j), & 1 \leq j \leq l, \\ (s + \delta_{jn}(r - c))(\overline{D}_j - \tilde{Q}_j^*), & j > l, \end{cases} = \begin{cases} h(I_0 - \underline{D}_j), & 1 \leq j \leq l, \\ s(\overline{D}_j - \overline{Q}_j), & l < j < k, \\ (s + \delta_{jn}(r - c))(\overline{D}_j - \overline{Q}_n), & k \leq j. \end{cases}$$

Thus, the primal objective function value is

$$\begin{aligned} z_P = {} & lhI_0 - h\sum_{j=1}^{l} \underline{D}_j + \frac{sh}{s+h}\sum_{j=l+1}^{k-1}(\overline{D}_j - \underline{D}_j) + s\sum_{j=k}^{n-1}\overline{D}_j - (n-k)s\overline{Q}_n \\ & + \frac{(h+c)(s+r-c)}{s+h+r}(\overline{D}_n - \underline{D}_n). \end{aligned}$$

Consider the dual solution

$$u_i = \begin{cases} 1, & 1 \leq i \leq l, \\ \frac{s}{s+h}, & l < i < k, \\ 0, & k \leq i < n, \\ \frac{s+r-c+(n-k)s}{s+h+r}, & i = n, \end{cases} \quad v_i = \begin{cases} 0, & 1 \leq i \leq l, \\ \frac{h}{s+h}, & l < i < k, \\ 1, & k \leq i < n, \\ \frac{h+c-(n-k)s}{s+h+r}, & i = n, \end{cases} \quad w_i = \begin{cases} (l-i+1)h, & 1 \leq i \leq l, \\ 0, & l < i < k, \\ (i-k)s, & k \leq i \leq n. \end{cases}$$

Since $0 \leq h + c - (n-k)s \leq h + c \leq r + h + s$, we have $0 \leq v_n \leq 1$. Furthermore, $u_i = 1 - v_i$ for all i. Clearly, $u_i, v_i, w_i \geq 0$ for all i. We only need to verify constraints (5.12) and (5.13) to guarantee feasibility of the dual solution.

$$\begin{aligned} 1 \leq i \leq l: & -(s+h)u_i + s + w_i - w_{i+1} = -(s+h) + s + (l-i+1)h - (l-i)h = 0, \\ l < i < k: & -(s+h)u_i + s + w_i - w_{i+1} = -s + s + 0 - 0 = 0, \\ k \leq i < n: & -(s+h)u_i + s + w_i - w_{i+1} = 0 + s + (i-k)s - (i+1-k)s = 0, \\ i = n: & -(s+h+r)u_n + (s+r-c) + w_n = -(s+r-c) - (n-k)s + (s+r-c) + (n-k)s = 0. \end{aligned}$$

Hence, the proposed dual solution is feasible, and its objective function value is

$$\begin{aligned} z_D &= -h\sum_{i=1}^{l}\underline{D}_i + \frac{sh}{s+h}\sum_{i=l+1}^{k-1}(\overline{D}_i - \underline{D}_i) + s\sum_{i=k}^{n-1}\overline{D}_i + \frac{(s+r-c)(h+c)}{s+h+r}(\overline{D}_n - \underline{D}_n) \\ &- \frac{(n-k)s}{s+h+r}((h+c)\underline{D}_n + (s+r-c)\overline{D}_n) + lhI_0 \\ &= -h\sum_{i=1}^{l}\underline{D}_i + \frac{sh}{s+h}\sum_{i=l+1}^{k-1}(\overline{D}_i - \underline{D}_i) + s\sum_{i=k}^{n-1}\overline{D}_i + \frac{(s+r-c)(h+c)}{s+h+r}(\overline{D}_n - \underline{D}_n) \\ &- (n-k)s\overline{Q}_n + lhI_0 \\ &= z_P. \end{aligned}$$

Hence, the proposed solution is optimal.

Case A-2: $I_0 \geq \overline{Q}_n$ and $I_0 < \overline{Q}_l$ for some l.

In this case, the proposed solution is $\tilde{Q}_j^* = I_0$ for all j.

Let l be the smallest index in $\{k, k+1, \ldots, n-1\}$ for which $\overline{Q}_l > I_0$. Therefore, $\tilde{Q}_j^* < \overline{Q}_j$ for $l \leq j \leq n-1$ and $\tilde{Q}_j^* \geq \overline{Q}_j$ for all other j. Thus, from Equation (5.15),

$$y_j = \begin{cases} h(I_0 - \underline{D}_j), & j < l, \\ s(\overline{D}_j - I_0), & l \leq j \leq n-1, \\ (h+c)(I_0 - \underline{D}_n), & j = n. \end{cases}$$

The primal objective value for this solution is

$$z_P = (l-1)h - h\sum_{j=1}^{l-1}\underline{D}_j + s\sum_{j=l}^{n-1}\overline{D}_j - (n-l)sI_0 + (h+c)(I_0 - \underline{D}_n).$$

Consider the dual solution

$$u_i = \begin{cases} 1, & i < l, \\ 0, & l \le i < n, \\ 1, & i = n, \end{cases}, \; v_i = \begin{cases} 0, & i < l, \\ 1, & l \le i < n, \\ 0, & i = n, \end{cases}, \; w_i = \begin{cases} h + c - (n-l)s + (l-i)h, & i < l, \\ h + c - (n-i)s, & l \le i \le n. \end{cases}$$

Since $l \ge k$, we have $(n-l)s \le (n-k)s \le h + c$. So $u_i, v_i, w_i \ge 0$ and $u_i + v_i = 1$ for all i. Once again, verifying the constraints (5.12) and (5.13) gives us the following.

$$\begin{aligned} 1 \le i < l : &- (s+h)u_i + s + w_i - w_{i+1} = -(s+h) + s + (l-i)h - (l-i-1)h = 0, \\ l \le i < n : &- (s+h)u_i + s + w_i - w_{i+1} = 0 + s - (n-i)s + (n-i-1)s = 0, \\ i = n : &- (s+h+r)u_n + (s+r-c) + w_n = -(s+h+r) + (s+r-c) + (h+c) = 0. \end{aligned}$$

Once again, the proposed dual solution is feasible and the corresponding dual objective value is

$$z_D = -h \sum_{i=1}^{l-1} \underline{D}_i + s \sum_{i=l}^{n-1} \overline{D}_i - (h+c)\underline{D}_n + I_0(h + c - (n-l)s + (l-1)h) = z_P.$$

Hence the solution is optimal.

Case A-3: $I_0 > \overline{Q}_j$ for all $j \in \{1, \ldots, n\}$.

In this case, $\tilde{Q}_j^* = I_0 \ge \overline{Q}_j$ for all j. From Equation (5.15), we have

$$\begin{aligned} y_j &= (h + \delta_{jn}c)(\tilde{Q}_j^* - \underline{D}_j) = (h + \delta_{jn}c)(I_0 - \underline{D}_j) \;\forall\; j \\ \implies z_P &= \sum_{j=1}^{n} y_j = (nh + c)I_0 - \sum_{j=1}^{n}(h + \delta_{jn}c)\underline{D}_j. \end{aligned}$$

Consider the dual solution

$$u_i = 1 \ \forall i, \ v_i = 0 \ \forall i, \ w_i = c + (n - i + 1)h \ \forall i.$$

Then, for $i < n$, $-(s+h)u_i + s + w_i - w_{i+1} = -h + (n-i+1)h - (n-i)h = 0$, and $-(s+h+r)u_n + (s+r-c) + w_n = -(s+h+r) + (s+r-c) + h + c = 0$ as well. Thus, the proposed dual solution is feasible, and the dual objective value is

$$z_D = \sum_{i=1}^{n} -(h + \delta_{in} c)\underline{D}_i + I_0(c + nh) = z_P.$$

Hence, the proposed solution is optimal.

This concludes the proof for case (A), and we now proceed to case (B). Note that this arises only when $k < n$. So $\overline{Q}_{k-1} \leq \overline{Q}_n < \overline{Q}_k$. Also, $m \geq k$ is chosen so that $(n-m-1)s \leq h + c < (n-m)s$.

As before, we consider three sub-cases.

Case B-1: $I_0 < \overline{Q}_m$.

Let l be the largest index in $\{0, 1, \ldots, m-1\}$ for which $I_0 \geq \overline{Q}_l$. Therefore, $\overline{Q}_l \leq I_0 < \overline{Q}_{l+1}$. The proposed solution is

$$\tilde{Q}_j^* = \begin{cases} I_0, & 1 \leq j \leq l, \\ \overline{Q}_j, & l < j < m, \\ \overline{Q}_m, & m \leq j. \end{cases}$$

From Equation (5.15), we have

$$y_j = \begin{cases} h(\tilde{Q}_j^* - \underline{D}_j), & j \leq l, \\ s(\overline{D}_j - \tilde{Q}_j^*), & l < j < n, \\ (h+c)(\tilde{Q}_n^* - \underline{D}_n), & j = n \end{cases} = \begin{cases} h(I_0 - \underline{D}_j), & j \leq l, \\ s(\overline{D}_j - \overline{Q}_j), & l < j \leq m, \\ s(\overline{D}_j - \overline{Q}_m), & m < j < n, \\ (h+c)(\overline{Q}_m - \underline{D}_n), & j = n, \end{cases}$$

$$\implies z_P = \sum_{j=1}^{n} y_j = lhI_0 - h\sum_{j=1}^{l} \underline{D}_j + \frac{sh}{s+h}\sum_{j=l+1}^{m} (\overline{D}_j - \underline{D}_j) + s\sum_{j=m+1}^{n-1} \overline{D}_j$$
$$+ (h+c-(n-m-1)s)\overline{Q}_m - (h+c)\underline{D}_n.$$

Consider the dual solution

$$u_i = \begin{cases} 1, & i \leq l, \\ \frac{s}{s+h}, & l < i < m, \\ \frac{s-(h+c)+(n-m-1)s}{s+h}, & i = m, \\ 0, & m < i < n, \\ 1, & i = n, \end{cases} \quad v_i = \begin{cases} 0, & i \leq l, \\ \frac{h}{s+h}, & l < i < m, \\ \frac{h+(h+c)-(n-m-1)s}{s+h}, & i = m, \\ 1, & m < i < n, \\ 0, & i = n, \end{cases}$$

$$w_i = \begin{cases} (l-i+1)h, & i \leq l, \\ 0, & l < i \leq m, \\ h+c-(n-i)s, & m < i \leq n. \end{cases}$$

$u_m = (n-m)s - (h+c) \geq 0$ by choice of m. Also, $(h+c) - (n-m-1)s \geq 0$ implies that

$v_m \geq 0$. In fact, $u_i, v_i, w_i \geq 0$ and $u_i + v_1 = 1$ for all i. Furthermore,

$$\begin{aligned}
1 \leq i \leq l :& -(s+h)u_i + s + w_i - w_{i+1} = -h + (l-i+1)h - (l-i)h = 0, \\
l < i < m :& -(s+h)u_i + s + w_i - w_{i+1} = -s + s + 0 - 0 = 0, \\
i = m :& -(s+h)u_i + s + w_i - w_{i+1} = (h+c) - (n-m-1)s - (h+c) + (n-m-1)s = 0, \\
m < i < n :& -(s+h)u_i + s + w_i - w_{i+1} = s - (n-i)s + (n-i-1)s = 0, \\
i = n :& -(s+h+r)u_n + (s+r-c) + w_n = -(s+h+r) + (s+r-c) + (h+c) = 0.
\end{aligned}$$

Therefore, this dual solution is feasible, and the objective function evaluates to

$$\begin{aligned}
z_D = & -h\sum_{i=1}^{l} \underline{D}_i + \frac{sh}{s+h}\sum_{i=l+1}^{m}(\overline{D}_i - \underline{D}_i) + (h + c - (n-m-1)s)\overline{Q}_m \\
& + s\sum_{i=m+1}^{n-1}\overline{D}_i - (h+c)\underline{D}_n + I_0 lh = z_P.
\end{aligned}$$

Hence, the solution is optimal.

Case B-2: $\overline{Q}_m \leq I_0$ and $\overline{Q}_l > I_0$ for some l.

The proof follows exactly the same logic as Case A-2.

Then, $\tilde{Q}_j^* = I_0$ for all j. Let l be the smallest index in $\{m+1, \ldots, n-1\}$ for which $\overline{Q}_l > I_0$. Therefore, $\tilde{Q}_j^* < \overline{Q}_j$ for $l \leq j \leq n-1$ and $\tilde{Q}_j^* \geq \overline{Q}_j$ for all other j. Thus, from Equation (5.15),

$$y_j = \begin{cases} h(I_0 - \underline{D}_j), & j < l, \\ s(\overline{D}_j - I_0), & l \leq j \leq n-1, \\ (h+c)(I_0 - \underline{D}_n), & j = n \end{cases}$$

$$\implies z_P = \sum_{j=1}^{n} y_j = (l-1)h - h\sum_{j=1}^{l-1}\underline{D}_j + s\sum_{j=l}^{n-1}\overline{D}_j - (n-l)sI_0 + (h+c)(I_0 - \underline{D}_n).$$

Again, consider the dual solution

$$u_i = \begin{cases} 1, & i < l, \\ 0, & l \le i < n, \\ 1, & i = n, \end{cases} \quad v_i = \begin{cases} 0, & i < l, \\ 1, & l \le i < n, \\ 0, & i = n, \end{cases} \quad w_i = \begin{cases} h + c - (n-l)s + (l-i)h, & i < l, \\ h + c - (n-i)s, & l \le i \le n. \end{cases}$$

This solution is feasible as shown in proof of case A-2, and the corresponding objective function value is

$$z_D = -h \sum_{i=1}^{l-1} \underline{D}_i + s \sum_{i=l}^{n-1} \overline{D}_i - (h+c)\underline{D}_n + I_0(h + c - (n-l)s + (l-1)h) = z_P.$$

Hence the solution is optimal.

Case B-3: $I_0 > \overline{Q}_j$ for all $j \in \{1, \dots, n\}$.

This proof is identical to Case A-3, and therefore omitted.

□

Theorem 5.3.1 provides an optimal solution to (5.8), and the following corollary uses it to obtain the robust optimal order quantities q_j^* in each period j.

Corollary 5.3.2 (Optimal Order Quantities)**.** *Let k be the largest integer in $\{1, 2, \dots, n\}$ for which $\overline{Q}_{k-1} \le \overline{Q}_n$ (where $\overline{Q}_0 = \min\{I_0, 0\}$).*

(A) If $(n-k)s \le h + c$, then the optimal order quantities are given by

$$q_j^* = \begin{cases} (\overline{Q}_j - I_0)^+, & j = 1, \\ (Q_j - I_0)^+ - (Q_{j-1} - I_0)^+, & j = 2, \dots, k-1, \\ (Q_n - I_0)^+ - (Q_{k-1} - I_0)^+, & j = k, \\ 0, & j = k+1, \dots, n. \end{cases}$$

(B) If $(n-k)s > h+c$, let m be the smallest integer in $\{k, \ldots, n-1\}$ for which $(n-m-1)s \leq h+c$. Then the optimal order quantities are given by

$$q_j^* = \begin{cases} (\overline{Q}_1 - I_0)^+, & j = 1, \\ (\overline{Q}_j - I_0)^+ - (\overline{Q}_{j-1} - I_0)^+, & j = 1, \ldots, m, \\ 0, & j = m+1, \ldots, n. \end{cases}$$

Proof. Proof: From Theorem 5.3.1, $\tilde{Q}_1^* = \max\{\overline{Q}_1, I_0\}$. Therefore,

$$q_1^* = Q_1^* = \tilde{Q}_1^* - I_0 = \max\{\overline{Q}_1, I_0\} - I_0 = \max\{\overline{Q}_1 - I_0, 0\} + I_0 - I_0 = (\overline{Q}_1 - I_0)^+.$$

The rest of the proof is similar. For any i and j, we have

$$\begin{aligned} \max\{\overline{Q}_i, I_0\} - \max\{\overline{Q}_j, I_0\} &= (\max\{\overline{Q}_i - I_0, 0\} + I_0) - (\max\{\overline{Q}_j - I_0, 0\} + I_0) \\ &= (\overline{Q}_i - I_0)^+ - (\overline{Q}_j - I_0)^+. \end{aligned}$$

This, along with the fact that $q_j^* = Q_j^* - Q_{j-1}^* = \tilde{Q}_j^* - \tilde{Q}_{j-1}^*$ for $j = 2, \ldots, n$, completes the proof.

□

5.3.1 Discussion

Recall that the seller's objective is to minimize the sum of the variables y_j, where

$$y_j = \max\Big\{(h + \delta_{jn}c)(Q_j + I_0 - \underline{D}_j), (s + \delta_{jn}(r-c))(\overline{D}_j - Q_j - I_0)\Big\}, \ j = 1, \ldots, n.$$

For each j, y_j is the maximum of two costs, the first of which is incurred when the stock in period j exceeds the worst-case cumulative demand through the first j periods, penalizing any excess inventory. The second term corresponds to shortage and is incurred when the worst-case cumulative demand is more than the current stock and the seller is unable to fulfill it. Minimizing y_j amounts to simultaneously minimizing both these terms, a task best achieved when the two costs become equal. This happens when $Q_j = \overline{Q}_j + I_0$ for all j, which yields order quantities

$$\hat{q}_1 = \overline{Q}_1 - I_0, \text{ and } \hat{q}_j = Q_j - Q_{j-1} = \overline{Q}_j - \overline{Q}_{j-1}, \text{ for } j = 2, \ldots, n. \tag{5.16}$$

Of course, these order quantities may not be feasible. Infeasibility occurs if $\hat{q}_j$ is negative for any j. Note that, for $j = 2, \ldots, n-1$, $\overline{Q}_j \geq \overline{Q}_{j-1}$ by definition, so we always have $\hat{q}_2, \ldots, \hat{q}_{n-1} \geq 0$. In other words, only $\hat{q}_1$ and $\hat{q}_n$ can potentially take negative values.

For simplicity, let us first consider the case where $I_0 = 0$, that is, the seller has no initial inventory. This further implies that $\hat{q}_1 \geq 0$. When $\overline{Q}_n \geq \overline{Q}_{n-1}$, the solution defined in (5.16) is feasible and, in fact, optimal by case (A) of Corollary 5.3.2. On the other hand, when $\overline{Q}_n < \overline{Q}_{n-1}$, we have $\hat{q}_n < 0$ and the solution proposed in (5.16) is no longer feasible. Nonetheless, the quantities $\overline{Q}_j$ help us in constructing an optimal solution.

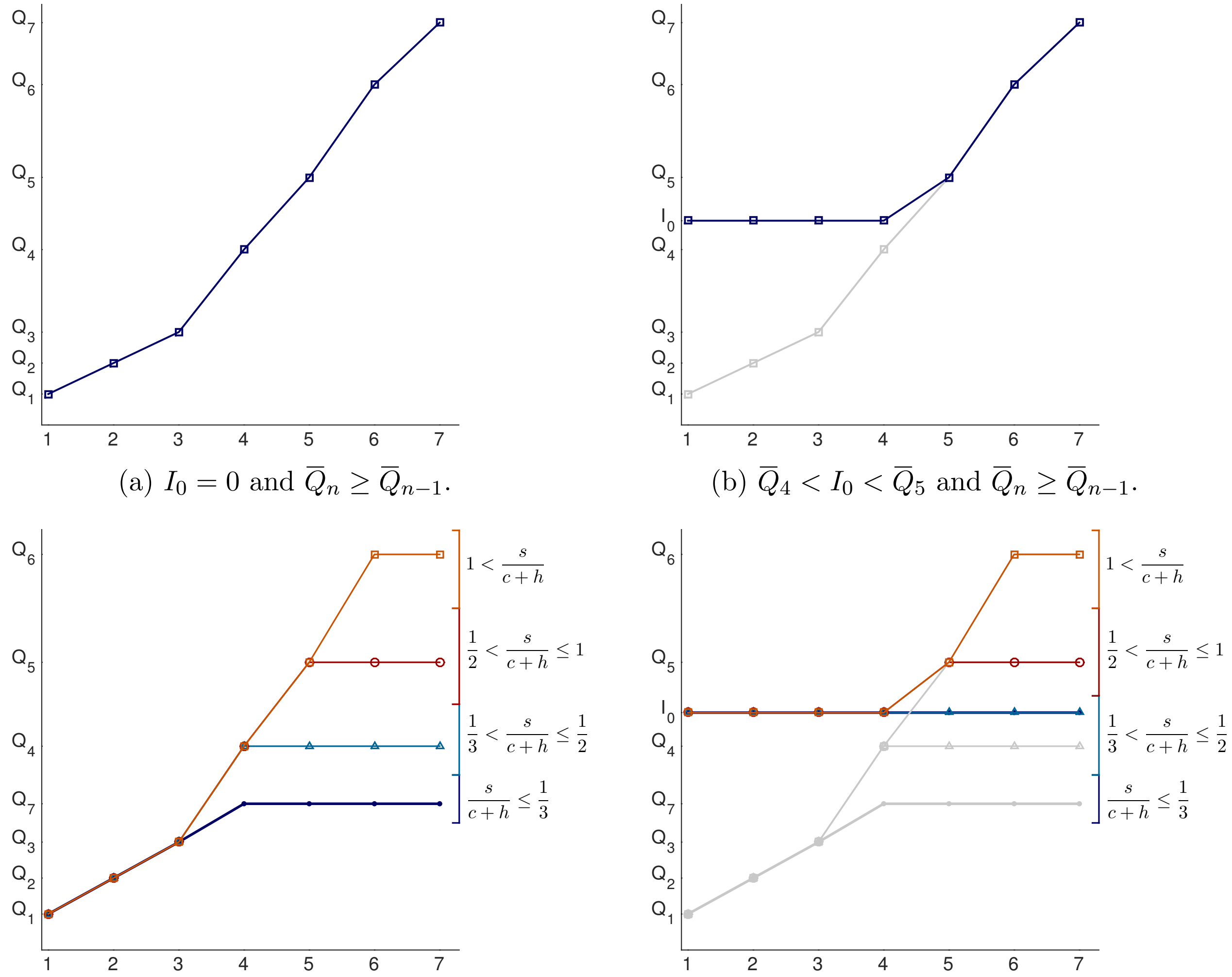
Let $k \in \{1, 2, \ldots, n-1\}$ be such that $\overline{Q}_{k-1} \leq \overline{Q}_n < \overline{Q}_k$ (where $\overline{Q}_0 = 0$). Based on the ordering of $\overline{Q}_j$, we now have several candidate solutions, corresponding to ordering up to some period $m \geq k$. A larger m corresponds both to ordering over a longer horizon, and also to acquiring a larger quantity overall. The choice of m depends on the relationship between the cost parameters. Intuitively, the seller would order smaller quantities if holding costs and purchase costs are higher relative to shortage costs, even if that means forgoing some of the demand. That is, if s is large relative to $h + c$, m is also large. The converse is true if this relationship is reversed.

Corollary (5.3.2) makes this idea more precise. The per-unit shortage cost s is a positive number and lies between some consecutive multiples of $h + c$. If $s \leq \frac{1}{n-k}(h + c)$, holding and purchase costs outweigh shortage costs and the seller only orders the product until period k and then stops; the total amount ordered over the entire horizon is $\overline{Q}_n$. However, if $\frac{1}{n-k}(h + c) < s \leq \frac{1}{n-(k+1)}(h + c)$, the shortage costs are more significant than in the previous case. The seller still orders up to period k but the final cumulative order quantity is $\overline{Q}_k > \overline{Q}_n$. Let us go one step further. If $\frac{1}{n-(k+1)}(h+c) < s \leq \frac{1}{n-(k+2)}(h+c)$, the shortage costs carry even more weight. Consequently, the seller orders up to period $k+1$ and the total quantity ordered is $\overline{Q}_{k+1} \geq \overline{Q}_k$. In general, suppose $\frac{1}{n-m}(h+c) < s \leq \frac{1}{n-(m+1)}(h+c)$ for some $m \in \{k, \ldots, n-1\}$ (where $1/0 = \infty$). A larger value of m indicates a greater weight on shortage costs, so the seller orders up to period m. The cumulative quantity ordered over the entire horizon is $\overline{Q}_m$ and it increases with m. Observe that no new product is ordered in the last period in any case.

Finally, if $I_0 \neq 0$, the optimal quantities in the above discussion can be viewed as 'target', or base-stock, inventory levels. Suppose $\overline{Q}_i$ was the optimal cumulative order quantity in period j with zero initial inventory (where i is not necessarily equal to j). If I_0 happens to be larger than $\overline{Q}_i$, the seller already has sufficient stock and no more product is ordered in period j. On the other hand, if $I_0 < \overline{Q}_i$, the optimal cumulative order quantity for period j is $\overline{Q}_i - I_0$. Thus, the seller orders enough product in period j to make up the difference and bring the current stock level up to $\overline{Q}_i$. This holds even when $I_0 < 0$ indicating unfulfilled demand at the beginning itself. This is particularly useful in the dynamic variant in Section 5.3.3.

Figure 5.3.1 illustrates the main result on a toy problem with 7 periods. The quantities $Q_j^* + I_0$ are plotted versus j. For $j > 1$, the jump from period $j-1$ to j gives the optimal quantity to order in period j. Hence, an incoming flat line indicates that no product is ordered in period j. The left column corresponds to the $I_0 = 0$ case while the right column illustrates the case where there is some positive initial inventory. In this example, $\overline{Q}_4 < I_0 < \overline{Q}_5$. The case with negative inventory is not illustrated but the general behavior is as in the second column. The top row is for the case where $\overline{Q}_n \geq \overline{Q}_{n-1}$. In the examples on the bottom row, $\overline{Q}_{k-1} \leq \overline{Q}_n < \overline{Q}_k$ for $k = 4$. The multiple curves in this case correspond to different values of m depending on the ratio of s to $c + h$.

In Figure 5.1a, we simply have $Q_j^* = \overline{Q}_j$ for all j. In Figure 5.1b, no product is ordered in the first four periods where the initial inventory level I_0 exceeds the 'target' levels $\overline{Q}_j$. In



(a) $I_0 = 0$ and $\overline{Q}_n \geq \overline{Q}_{n-1}$.

(b) $\overline{Q}_4 < I_0 < \overline{Q}_5$ and $\overline{Q}_n \geq \overline{Q}_{n-1}$.

(c) $I_0 = 0$ and $\overline{Q}_3 \leq \overline{Q}_7 < \overline{Q}_4$.

(d) $\overline{Q}_4 < I_0 < \overline{Q}_5$ and $\overline{Q}_3 \leq \overline{Q}_7 < \overline{Q}_4$.

Figure 5.1: An illustrative plot of $Q_j^* + I_0$ for a robust newsvendor model with seven periods under various relations between the parameters.

period 5, only the difference $\overline{Q}_5 - I_0$ is ordered to bring the stock up to the target level $\overline{Q}_5$. Thereafter, the two cases are identical.

In Figure 5.1c, there is no initial inventory, and the four curves correspond to different ranges on the ratio $s/(c+h)$. As this ratio increases, the total quantity ordered also increases and the product is ordered over a larger horizon as well. In Figure 5.1d, we again assume that there is some positive initial inventory. In the lower two cases, with $s/(c+h) \leq 1/3$ and $1/3 < s/(c+h) \leq 1/2$, the 'target' final order quantities are $\overline{Q}_7$ and $\overline{Q}_4$ respectively. Since I_0 exceeds these, no product is ordered at all. In the other two cases, product is ordered in period 5 to make up the difference and bring the stock up to target levels, and the seller then proceeds as in Case 5.1c.

In the end, we observe that the optimal order quantities defined in Corollary 5.3.2 are identical to those obtained in Mamani et al. [31] when $r = 0$ (which implies $c = 0$). We also remark that in some cases, the revenue and ordering cost parameters r and c do not appear *explicitly* in the optimal order quantities. This may give the impression that the optimal solution is independent of the same; but we note that there is an implicit dependence since the indices k and m are functions of these parameters. In fact, $\overline{Q}_n$ is increasing in r and decreasing in c. So it is the relation between these and other cost parameters that determines how $\overline{Q}_n$ is ordered among the other points $\overline{Q}_j$, $j \neq n$.

5.3.2 Closed-form solutions for inner problems

The optimal order quantities in Corollary 5.3.2 are defined in terms of the cost and revenue parameters, as well as the quantities $\underline{D}_j$ and $\overline{D}_j$. Recall that $\underline{D}_j$ is the least possible cumulative demand through period j over all possible demand vectors $d \in \Omega$. Similarly, $\overline{D}_j$ is the maximum cumulative demand up to period j. These can be computed by numerically solving the following so-called 'inner problems' for all j.

$$\begin{aligned} \underline{D}_j &= \min \sum_{i=1}^{j} d_i \qquad\qquad & \overline{D}_j &= \max \sum_{i=1}^{j} d_i \\ \text{s.t.} \quad & d \in \Omega, & \text{s.t.} \quad & d \in \Omega. \end{aligned}$$

The objective functions are linear in d, and these problems can easily be solved to arbitrary accuracy, especially when Ω is chosen to be a closed, convex set. Even so, the optimal solutions in Corollary 5.3.2 are easier to implement if the worst-case cumulative demands are also available in closed-form. In this section, we describe a class of uncertainty sets which frequently arises in practice, and explicitly compute $\underline{D}_j$ and $\overline{D}_j$ for these sets.

Consider an uncertainty set of the form

$$\Omega = \left\{ d = (d_1, \ldots, d_n) : A \leq \sum_{t=1}^{n} d_t \leq B,\ a_t \leq d_t \leq b_t,\ t = 1, \ldots, n \right\}, \tag{5.17}$$

where the parameters A, B, a_t and b_t, $t = 1, \ldots, n$ are chosen so that Ω is non-empty. In particular, $a_t \leq b_t$ for all t and $A \leq B$. Since the demand must be nonnegative, let $a_t \geq 0$ for all t without loss of generality. Similarly, we can also assume that $A \geq \sum\limits_{t=1}^{n} a_t$ and $B \leq \sum\limits_{t=1}^{n} b_t$. Many of the uncertainty sets studied in literature fall within this framework, and in the next two lemmas, we obtain the optimal solutions for computing $\underline{D}_j$ and $\overline{D}_j$ for all j. We find that the optimal *solution* for both of these quantities is independent of j.

Lemma 5.3.3. *Let ι be the largest integer in $\{1, \ldots, n\}$ for which*

$$A \leq \sum_{t=1}^{\iota-1} a_t + \sum_{t=\iota}^{n} b_t \quad \textit{and} \quad \sum_{t=1}^{\iota} a_t + \sum_{t=\iota+1}^{n} b_t \leq B. \tag{5.18}$$

Let $\underline{d}^* = (a_1, \ldots, a_{\iota-1}, l_\iota, b_{\iota+1}, \ldots, b_n)$, where $l_\iota = A - \sum\limits_{t=1}^{\iota-1} a_t - \sum\limits_{t=\iota+1}^{n} b_t$. Then, $\underline{D}_j = \sum\limits_{t=1}^{j} \underline{d}_t^*$ for all $j = 1, \ldots, n$.

Proof. Proof: We first prove by contradiction that such an ι must exist. Let $\mathcal{I}_A \subseteq \{1, \ldots, n\}$ be the collection of indices for which the first inequality in (5.18) is satisfied. Since $1 \in \mathcal{I}_A$, this set is non-empty. Similarly, the set $\mathcal{I}_B$ of indices which satisfy the second inequality in (5.18) is also non-empty, as $n \in \mathcal{I}_B$. By definition, $\iota = \max\{i : i \in \mathcal{I}_A \cap \mathcal{I}_B\}$.

Suppose ι does not exist. This can only happen if the intersection of $\mathcal{I}_A$ and $\mathcal{I}_B$ is empty. Let ι_A be the largest element of $\mathcal{I}_B$ and ι_B be the smallest element of $\mathcal{I}_B$. Then, $1 \leq \iota_A < \iota_B \leq n$ and $\iota_A + 1 \notin \mathcal{I}_A$. Therefore,

$$\sum_{t=1}^{\iota_A} a_t + \sum_{t=\iota_A+1}^{n} b_t < A \leq B \implies \iota_A \in \mathcal{I}_B,$$

which is a contradiction. Hence, $\mathcal{I}_A \cap \mathcal{I}_B \neq \varnothing$, and ι as defined in the statement of the lemma must exist.

Now, we verify the feasibility of the proposed solution. Clearly, $a_t \leq \underline{d}_t^* \leq b_t$ for all $t \neq \iota$. From the first inequality in Condition (5.18), we have that $\underline{d}_\iota^* = l_\iota \leq b_\iota$. Also, by choice of ι, $\iota + 1$ does not satisfy Condition (5.18). So it must violate one of the two inequalities. Since ι satisfies the second inequality, so must $\iota + 1$. Therefore, the first inequality in (5.18) must be violated and we have

$$\sum_{t=1}^{\iota} a_t + \sum_{t=\iota+1}^{n} b_t < A \implies a_\iota < A - \sum_{t=1}^{\iota-1} a_t + \sum_{t=\iota+1}^{n} b_t = l_\iota = \underline{d}_\iota^*.$$

So $a_t \leq \underline{d}_t^* \leq b_t$ for all t, and $\sum_{t=1}^{n} \underline{d}_t^* = A$ by construction. Hence, $\underline{d}^* \in \Omega$, that is, the proposed solution is feasible.

For the proof of optimality, we consider any other feasible solution $\hat{d}$. Let $S_j(d) = \sum_{t=1}^{j} d_t$ for any $d \in \Omega$. Then, for $j < \iota$, $S_j(\hat{d}) = \sum_{t=1}^{j} \hat{d}_t \geq \sum_{t=1}^{j} a_t = \sum_{t=1}^{j} \bar{d}_t^* = S_j(\underline{d}^*)$. Therefore, $S_j(\underline{d}^*) = \min_{d \in \Omega} S_j(d) = \underline{D}_j$, that is, $\underline{d}^*$ is an optimal solution for computing $\underline{D}_j$. For $j \geq \iota$,

$$S_j(\underline{d}^*) = \sum_{t=1}^{\iota-1} a_t + \Big(A - \sum_{t=1}^{\iota-1} a_t - \sum_{t=\iota+1}^{n} b_t\Big) + \sum_{t=\iota+1}^{j} b_t = A - \sum_{t=j+1}^{n} b_t.$$

Since $\hat{d} \in \Omega$, we have

$$\sum_{t=1}^{n} \hat{d}_t \geq A \implies S_j(\hat{d}) = \sum_{t=1}^{j} \hat{d}_t \geq A - \sum_{t=j+1}^{n} \hat{d}_t \geq A - \sum_{t=j+1}^{n} b_t = S_j(\underline{d}^*),$$

since $-\hat{d}_t \geq -b_t$ for all t. Thus, $S_j(\underline{d}^*) \leq S_j(\hat{d})$ for all $\hat{d} \in \Omega$ and all $j = 1, \ldots, n$. It follows that $\underline{D}_j = \sum_{t=1}^{j} \underline{d}_t^*$ for all j. □

Lemma 5.3.4. *Let ν be the largest integer in $\{1,\ldots,n\}$ for which*

$$A \le \sum_{t=1}^{\nu} b_t + \sum_{t=\nu+1}^{n} a_t \quad \textit{and} \quad \sum_{t=1}^{\nu-1} b_t + \sum_{t=\nu}^{n} a_t \le B. \tag{5.19}$$

Let $\bar{d}^* = (b_1,\ldots,b_{\nu-1},u_\nu,a_{\nu+1},\ldots,a_n)$, where $u_\nu = B - \sum\limits_{t=1}^{\nu-1} b_t - \sum\limits_{t=\nu+1}^{n} a_t$. Then, $\overline{D}_j = \sum\limits_{t=1}^{j} \bar{d}_t^*$ for all $j = 1,\ldots,n$.

Proof. Proof: The proof is similar to Lemma 5.3.3, and is omitted.

□

We now examine some special uncertainty sets motivated from the limit theorems in probability and studied in the robust optimization literature. Limit theorems are used to study the asymptotic behavior of sequences and series of random variables. They are particularly useful when the exact distribution of random variables is unknown, and only partial information (like certain moments) is available.

Central Limit Theorem.

The Central Limit Theorem (CLT) is one of the most powerful and widely used results in probability theory. If X_i, $i = 1,2,\ldots$, are independent and identically distributed (i.i.d.) random variables with mean μ and standard deviation σ, the CLT states that the distribution of the random variable $\big(\sum\limits_{i=1}^{n} X_i - n\mu\big)/(\sqrt{n}\sigma)$ approaches that of a standard normal distribution, as n increases. Viewing the demands d_j as i.i.d. random variables with known mean μ and standard deviation σ (but unknown distribution), Mamani et al. [31] uses the CLT to formulate the uncertainty set described below.

$$\Omega^{CLT} = \left\{ (d_1,\ldots,d_n) \; : \; -\Gamma \le \frac{\sum\limits_{t=1}^{n} d_t - n\mu}{\sqrt{n}\sigma} \le \Gamma, \; \mu - \Gamma\sigma \le d_t \le \mu + \Gamma\sigma, t = 1,\ldots,n \right\}. \tag{5.20}$$

This is a special case of (5.17) with $A = n\mu - \sqrt{n}\Gamma\sigma$, $B = n\mu + \sqrt{n}\Gamma\sigma$, $a_t = \mu - \Gamma\sigma$ and $b_t = \mu + \Gamma\sigma$ for all t, where we assume that $\mu - \Gamma\sigma \geq 0$. $\Gamma > 0$ is a tunable parameter which allows us to adjust the conservativeness of the robust approach. Note that $\Gamma = 0$ would make Ω^{CLT} a singleton set, and fix the demand in each period at the mean μ.

We now invoke Lemmas 5.3.3 and 5.3.4 to compute $\underline{D}_j$ and $\overline{D}_j$ for this uncertainty set. By (5.18), ι is the largest index in $\{1, \ldots, n\}$ for which

$$\begin{aligned}
& n\mu - \sqrt{n}\Gamma\sigma \leq (\iota - 1)(\mu - \Gamma\sigma) + (n - \iota + 1)(\mu + \Gamma\sigma) \\
\text{and} \quad & \iota(\mu - \Gamma\sigma) + (n - \iota)(\mu + \Gamma\sigma) \leq n\mu + \sqrt{n}\Gamma\sigma \\
\Longleftrightarrow \quad & -\sqrt{n}\Gamma\sigma \leq (n - 2\iota + 2)\Gamma\sigma \quad \text{and} \quad (n - 2\iota)\Gamma\sigma \leq \sqrt{n}\Gamma\sigma \\
\Longleftrightarrow \quad & n - \sqrt{n} \leq 2\iota \leq n + \sqrt{n} + 2.
\end{aligned}$$

A similar calculation using (5.19) shows that ν is the largest index in $\{1, \ldots, n\}$ for which

$$n - \sqrt{n} \leq 2\nu \leq n + \sqrt{n} + 2.$$

Thus, ι and ν coincide in this case, and we have $\iota = \nu = \lfloor \tau \rfloor + 1$, where $\tau = (n + \sqrt{n})/2$. Moreover, a simple calculation gives

$$l_\iota = \mu + \Gamma\sigma - 2(\tau - \lfloor \tau \rfloor)\Gamma\sigma, \quad u_\nu = \mu - \Gamma\sigma + 2(\tau - \lfloor \tau \rfloor \Gamma\sigma).$$

Therefore,

$$\begin{aligned}
\underline{D}_j^{CLT} &= \min_{d \in \Omega^{CLT}} \sum_{t=1}^{j} d_t = \begin{cases} j\mu - j\Gamma\sigma, & j \leq \lfloor \tau \rfloor, \\ j\mu - (2\tau - j)\Gamma\sigma, & j > \lfloor \tau \rfloor \end{cases} \\
\overline{D}_j^{CLT} &= \max_{d \in \Omega^{CLT}} \sum_{t=1}^{j} d_t = \begin{cases} j(\mu + \Gamma\sigma), & j \leq \lfloor \tau \rfloor, \\ j\mu + (2\tau - j)\Gamma\sigma, & j > \lfloor \tau \rfloor. \end{cases}
\end{aligned}$$

Our set Ω^{CLT} is a special case of the partial-sum uncertainty sets considered in Mamani et al. [31]. This allows us to get simple expressions for the inner-problem solutions, but we note that these can also be derived from the formulas provided in that paper. From here, we can directly use Corollary 5.3.2 to compute the optimal order quantities for each period.

Strong Law of Large Numbers.

The Strong Law of Large Numbers (SLLN) is another widely-used limit theorem about the asymptotic behavior of the average of i.i.d. random variables. For i.i.d. random variables $X_1, X_2, \dots,$ with mean μ, the SLLN states that their average $\sum\limits_{i=1}^{n} X_i/n$ concentrates at the mean μ almost surely as $n \to \infty$. This motivates the following SLLN-based uncertainty set.

$$\Omega^{SL} = \left\{ (d_1, \dots, d_n) \;:\; \mu - \epsilon \le \frac{\sum\limits_{t=1}^{n} d_t}{n} \le \mu + \epsilon,\ \mu - \delta \le d_t \le \mu + \delta, t = 1, \dots, n \right\}. \tag{5.21}$$

Here, δ and ϵ are tunable parameters such that $0 \le \epsilon, \delta \le \mu$. See Wagner [45] for a discussion on how these may be chosen. As in section 5.3.2, a simple calculation shows that $\iota = \nu$. Further, we get from Condition (5.18) that ι is the largest index for which

$$\frac{n(\delta - \epsilon)}{2\delta} \le \iota \le \frac{n(\epsilon + \delta)}{2\delta} + 1.$$

Thus, $\iota = \lfloor \xi \rfloor + 1$, where $\xi = n(\epsilon + \delta)/2\delta$. Moreover,

$$l_\iota = (\mu - \delta) - n\epsilon - (n - 2\iota)\delta, \quad u_\nu = (\mu + \delta) + n\epsilon + (n - 2\nu)\delta.$$

Therefore, using Lemmas 5.3.3 and 5.3.4, we have

$$\underline{D}_j^{SL} = \begin{cases} j(\mu - \delta), & j \le \lfloor \xi \rfloor, \\ j\mu - n\epsilon - (n - j)\delta, & j > \lfloor \xi \rfloor, \end{cases}, \qquad \overline{D}_j^{SL} = \begin{cases} j(\mu + \delta), & j \le \lfloor \xi \rfloor, \\ j\mu + n\epsilon + (n - j)\delta, & j > \lfloor \xi \rfloor. \end{cases}$$

These are discrete analogs of the expressions for worst-case cumulative demand derived in Wagner [45]. Once again, we can now use Corollary 5.3.2 to find the robust optimal order quantities.

Law of Iterated Logarithms

The previous limit theorems are obtained by scaling the centralized sum of n i.i.d. random variables by factors $\sqrt{n}\sigma$ and n respectively. A third kind of scaling yields the law of iterated logarithms (LIL). Define $\phi(n) = \sqrt{n \log \log n}$ for all n. For i.i.d. random variables X_i with mean μ and variance σ^2, the LIL describes the behavior of $(\sum\limits_{i=1}^{n} X_i - n\mu)/\sigma\sqrt{2}\phi(n)$. Bertsimas et al. [13] uses the LIL to construct uncertainty sets, and we use the same idea here to define Ω^{LIL} as below.

$$\Omega^{LIL} = \left\{ (d_1, \ldots, d_n) \;:\; -(1+\epsilon) \le \frac{\sum\limits_{t=1}^{n} d_t - n\mu}{\sigma\sqrt{2}\phi(n)} \le (1+\epsilon),\ \mu - \delta \le d_t \le \mu + \delta, t = 1, \ldots, n \right\}. \tag{5.22}$$

Here, too, $\epsilon \ge 0$ and $0 \le \delta \le \mu$ are adjustable parameters. This set is also of the form (5.17), with $a_t = \mu - \delta$, $b_t = \mu + \delta$, $A = n\mu - (1+\epsilon)\sigma\sqrt{2}\phi(n)$ and $B = n\mu + (1+\epsilon)\sigma\sqrt{2}\phi(n)$. ι and ν coincide, and take the value

$$\iota = \nu = \lfloor \frac{n\delta + (1+\epsilon)\sigma\sqrt{2}\phi(n)}{2} \rfloor + 1.$$

Moreover,

$$\underline{D}_j^{LIL} = \begin{cases} j\mu - j\delta, & j < \iota, \\ j\mu - (1+\epsilon)\sigma\sqrt{2}\phi(n) - (n-j)\delta, & j \ge \iota, \end{cases}$$

$$\overline{D}_j^{LIL} = \begin{cases} j\mu + j\delta, & j < \iota, \\ j\mu + (1+\epsilon)\sigma\sqrt{2}\phi(n) + (n-j)\delta, & j \ge \iota, \end{cases}$$

We remark that closed-form expressions for $\underline{D}_j$and $\overline{D}_j$ for LIL-based uncertainty sets are not available in the literature. Here, these formulas can now be used in conjunction with Corollary 5.3.2 to obtain the robust optimal order quantities.

5.3.3 Dynamic Variant

In the above setup, we solved a static robust optimization problem which finds the optimal order quantities for every period in one shot. In practice, a seller may use observations from the first $k-1$ periods to inform his decision in the k-th period. This motivates a dynamic variant of the robust model based on re-optimization.

At the end of period $k-1$, the seller knows the demand in the first $k-1$ periods $\hat{d}_1, \ldots, \hat{d}_{k-1}$, as well as the corresponding order quantities $\hat{q}_1, \ldots, \hat{q}_{k-1}$. Consequently, the inventory level I_{k-1} at the end of period $k-1$, given by $\hat{I}_{k-1} = \sum\limits_{j=1}^{k-1}(\hat{q}_j - \hat{d}_j) + I_0$, is also known. Note that $\hat{I}_{k-1}$ may be of any sign. This is used to define a new robust model for the remaining $n-k+1$ periods. The unknown demand for these periods now varies within a projected uncertainty set Ω_k, which comprises all those demand vectors in Ω whose first $k-1$ components match the observed demand. That is,

$$\Omega_k = \Omega \cap \{(d_1, \ldots, d_n) : d_i = \hat{d}_i,\ i = 1, \ldots, k-1\}.$$

This yields an $(n-k+1)$-period static robust optimization problem with known initial inventory $\hat{I}_{k-1}$, analogous to (5.6).

$$\begin{aligned}
\min\quad & \sum_{j=k}^{n} y_j \\
\text{s.t.}\quad & y_j \geq (h + \delta_{jn} c)\Big(\sum_{j=k}^{n}(q_j - d_j) + \hat{I}_{k-1}\Big),\ j = k, \ldots, n,\ \forall\ (d_k, \ldots, d_n) \in \Omega_k, \\
& y_j \geq (s + \delta_{jn}(r-c))\Big(\sum_{j=k}^{n}(d_j - q_j) - \hat{I}_{k-1}\Big)\ j = k, \ldots, n,\ \forall\ (d_k, \ldots, d_n) \in \Omega_k, \\
& q_k, q_{k+1}, \ldots, q_n \geq 0.
\end{aligned}$$

Optimal solutions to this problem are obtained in closed-form in the same manner as in the static case. Thus, the seller uses the original static model at the beginning of the first period. In every subsequent period, he solves a new robust optimization problem taking into account observations from the previous periods. This rolling-horizon framework is a standard approach for studying dynamic variants of static optimization models; see Mamani et al. [31] and Solyalı et al. [41] for more details.

5.4 Benchmarking with Computational Experiments

In the previous sections, we formulated a robust newsvendor model for profit maximization and obtained optimal order quantities for the same. To the best of our knowledge, this is the first model that accounts for revenue in addition to costs. In this section, we present the results of numerical experiments in order to benchmark our model. Since a comparable robust optimization model is not available in the literature, we employ a stochastic model for these experiments.

5.4.1 A Stochastic Newsvendor Model

Recall that for a given vector of demands d, the problem of finding optimal order quantities reduces to solving the following minimization.

$$\min_{Q \geq 0} \sum_{j=1}^{n} \max \Big\{ (h + \delta_{jn} c)(Q_j + I_0 - D_j),\ (s + \delta_{jn}(r - c))(D_j - Q_j - I_0) \Big\}. \tag{5.23}$$

In the stochastic model, demand is treated as a random variable, and the agent tries to find order quantities which maximize his expected profit over the entire horizon. This corresponds to minimizing the expected value of the sum in (5.23). Let F_j be the cdf of $D_j = \sum_{i=1}^{j} d_i$, and let $F = (F_1, \dots, F_n)$. Then, the stochastic demand variant of (5.23) is

$$\begin{aligned}
z_S^* &= \min_{Q \geq 0} \mathbf{E}_F \left[\sum_{j=1}^{n} \max \Big\{ (h + \delta_{jn} c)(Q_j + I_0 - D_j),\ (s + \delta_{jn}(r - c))(D_j - (Q_j + I_0)) \Big\} \right] \\
&= \min_{Q \geq 0} \mathbf{E}_F \left[\sum_{j=1}^{n} (h + \delta_{jn} c)(Q_j + I_0 - D_j)^+ + (s + \delta_{jn}(r - c))(D_j - (Q_j + I_0))^+ \right] \\
&= \min_{Q \geq 0} \ \sum_{j=1}^{n} \mathbf{E}_F \Big[(h + \delta_{jn} c)(Q_j + I_0 - D_j)^+ + (s + \delta_{jn}(r - c))(D_j - (Q_j + I_0))^+ \Big] \\
&= \min_{Q \geq 0} \ \sum_{j=1}^{n} \int_0^{\infty} (h + \delta_{jn} c)(Q_j + I_0 - D_j)^+ dF_j + \int_0^{\infty} (s + \delta_{jn}(r - c))(D_j - (Q_j + I_0))^+ dF_j \\
&= \min_{Q \geq 0} \ \sum_{j=1}^{n} (h + \delta_{jn} c) \int_0^{Q_j + I_0} (Q_j + I_0 - D_j) dF_j + (s + \delta_{jn}(r - c)) \int_{Q_j + I_0}^{\infty} (D_j - (Q_j + I_0)) dF_j .
\end{aligned} \tag{5.24}$$

Closed-form optimal solutions.

Let $z_S(Q)$ denote the objective function in (5.24). Then,

$$\begin{aligned}
\frac{\partial z_S}{\partial Q_j} &= \begin{cases} h \int\limits_0^{Q_j + I_0} dF_j - s \int\limits_{Q_j + I_0}^{\infty} dF_j, & j \neq n, \\ (h + c) \int\limits_0^{Q_j + I_0} dF_j - (s + r - c) \int\limits_{Q_j + I_0}^{\infty} dF_j, & j = n \end{cases} \\
&= \begin{cases} h F_j(Q_j + I_0) - s(1 - F_j(Q_j + I_0), & j \neq n, \\ (h + c) F_j(Q_j + I_0) - (s + r - c)(1 - F_j(Q_j + I_0), & j = n \end{cases} \\
&= \begin{cases} (s + h) F_j(Q_j + I_0) - s, & j \neq n, \\ (s + h + r) F_j(Q_j + I_0) - (s + r - c), & j = n \end{cases} .
\end{aligned}$$

Also,

$$\frac{\partial z_S^2}{\partial Q_i \partial Q_j} = \begin{cases} 0, & i \neq j, \\ (s + h + \delta_{jn} r) f_j(Q_j + I_0), & i = j. \end{cases}$$

Thus, the Hessian of $z_S(Q)$ is positive definite, which implies that $z_S(\cdot)$ is a convex function over $Q \geq 0$ and a local minimum must be the global minimum. Setting the partial derivatives of z_S equal to zero gives

$$\hat{Q}_j = \begin{cases} F_j^{-1}\left(\frac{s}{s+h}\right) - I_0, & j \neq n, \\ F_n^{-1}\left(\frac{s+r-c}{s+h+r}\right) - I_0, & j = n. \end{cases} \tag{5.25}$$

$\hat{Q}$ will serve as a candidate optimal solution, but we first check for feasibility. $\hat{Q}$ is feasible if $\hat{Q}_n \geq \hat{Q}_{n-1} \geq \ldots \geq \hat{Q}_1 \geq 0$. This ordering depends on the relation between the cdfs F_j, which is established in the following lemma.

Lemma 5.4.1. *The cumulative demands D_j are (stochastically) ordered as $D_1 \leq D_2 \leq \ldots \leq D_n$.*

Proof. Proof: Let f_{D_j} and f_{d_j} be the probability density functions for D_j and d_j respectively. Note that $D_{j+1} = D_j + d_{j+1}$ for all $j < n$. Therefore, for any $z > 0$,

$$f_{D_{j+1}}(z) = \int_0^z f_{D_j}(z-y) f_{d_{j+1}}(y) dy.$$

For $x > 0$, we have

$$\begin{aligned}
F_{j+1}(x) &= \int_0^x f_{D_{j+1}}(z)\ dz = \int_0^x \int_0^z f_{D_j}(z-y) f_{d_{j+1}}(y)\ dy\ dz \\
&= \int_0^x \int_y^x f_{D_j}(z-y) f_{d_{j+1}}(y)\ dz\ dy \qquad \text{(changing the order of integration)} \\
&= \int_0^x \left(\int_y^x f_{D_j}(z-y)\ dz \right) f_{d_{j+1}}(y)\ dy = \int_0^x \left(\int_0^{x-y} f_{D_j}(u)\ du \right) f_{d_{j+1}}(y)\ dy \\
&= \int_0^x F_j(x-y) f_{d_{j+1}}(y)\ dy \leq F_j(x) \int_0^x f_{d_{j+1}}(y)\ dy \leq F_j(x).
\end{aligned}$$

Since this is true for all $x > 0$, it follows that $D_j \leq D_{j+1}$.

□

Lemma 5.4.1 implies that $F_{j+1}^{-1}(y) \geq F_j^{-1}(y)$ for all real numbers y. It follows from Equation (5.25) that $\hat{Q}_1 \leq \hat{Q}_2 \leq \ldots \leq \hat{Q}_{n-1}$. Therefore, infeasibility occurs if $\hat{Q}_n < \hat{Q}_{n-1}$, or if $\hat{Q}_j < 0$ for some j. However, we notice that $z_S(Q)$ is increasing in each Q_j since $\frac{\partial z_S}{\partial Q_j} > 0$. So, whenever the value proposed in (5.25) is infeasible, Q_j simply takes the smallest feasible value. Therefore, the optimal solution Q^s to the stochastic model is given by

$$Q_j^s = \begin{cases} \hat{Q}_j^+, & j \neq n, \\ \max\{\hat{Q}_n^+, \hat{Q}_{n-1}^+\}, & j = n. \end{cases} \tag{5.26}$$

The stochastic optimal order quantities are $q_1^s = Q_1^s$ and $q_j^s = Q_j^s - Q_{j-1}^s$ for $j > 1$.

5.4.2 Computational Experiments

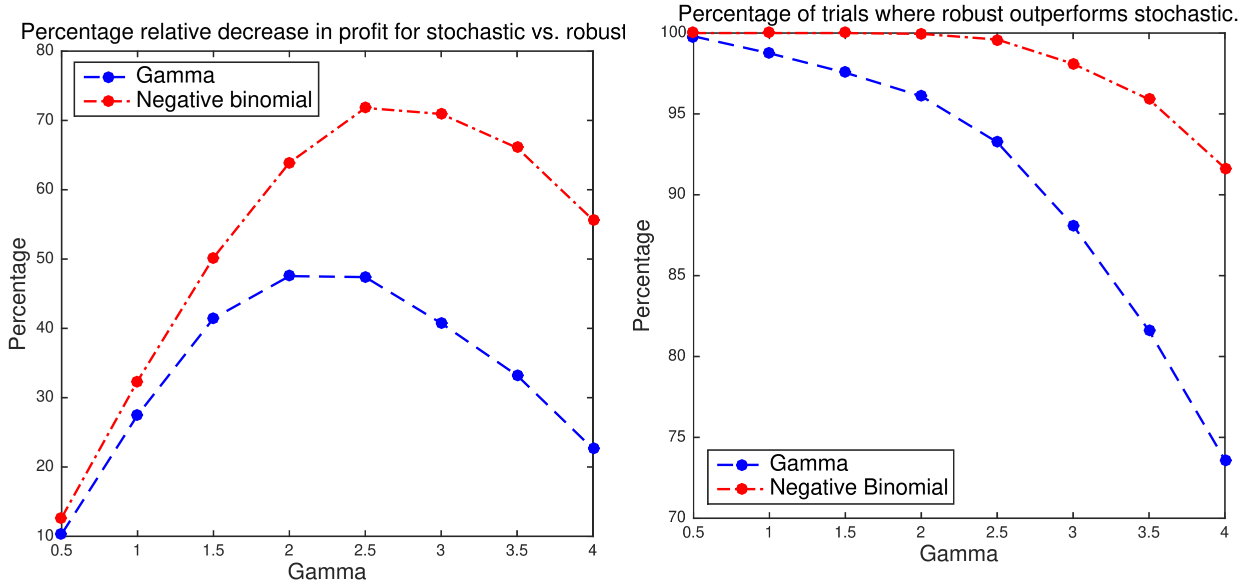
We perform simulation experiments to compare our robust model to the stochastic one described above. In order to compute the stochastic-optimal order quantities in (5.26), the seller must know the demand distribution exactly. This is often not the case in practice, and the robust model attempts to mitigate the effects of incorrect information. We mimic this phenomenon by assuming that the stochastic model uses a misspecified demand distribution. We consider two cases – one where the parametrized distribution is from the correct family with incorrect parameter values, and the other where the parameter values are accurate but the distribution is not.

Misspecified distribution:

In the first set of experiments, we examine the case where the assumed distribution $\hat{F}$ in the stochastic model is different from the *true* demand distribution F. We consider two examples corresponding to the assumed demand distribution being a gamma distribution and a negative binomial distribution respectively. These distributions were chosen because they have a non-negative support and are closed under addition. Then, the cumulative demand D_j for all j is also gamma and negative-binomially distributed respectively. This is helpful because the optimal solution for the stochastic model uses the cdf D_j, which is available in closed form for these two cases. An attractive feature of the robust model is that it does not require any such distributional information.

We arbitrarily set the various model parameters as $c = 1$, $h = 1$, $s = 1.5$, $n = 20$, $r = 1.5$, with the length of the horizon set at $n = 20$. The robust model employs the CLT-based uncertainty sets defined in (5.20), using the same values of mean μ and standard deviation σ as in the stochastic model. Recall that $\Gamma \geq 0$ was a parameter used to adjust the degree of conservativeness in the robust model. We repeat the experiment for values of Γ in the set $\{0.5, 1, \ldots, 4\}$. The stochastic optimal solutions do not depend on Γ, and the variation is Γ allows us to explore the relative performance of the two models as a function of the construction of the uncertainty sets.

We perform $n_{sims} = 2000$ simulations. In each simulation, the true demand is generated using a (truncated) normal distribution with the same mean (μ) and standard deviation (σ). The realized profit is computed using the optimal order quantities from both the robust and stochastic models. Two metrics are used to compare performance. The first metric, plotted in



(a) The average relative reduction in profit in the stochastic model vs. the robust model.

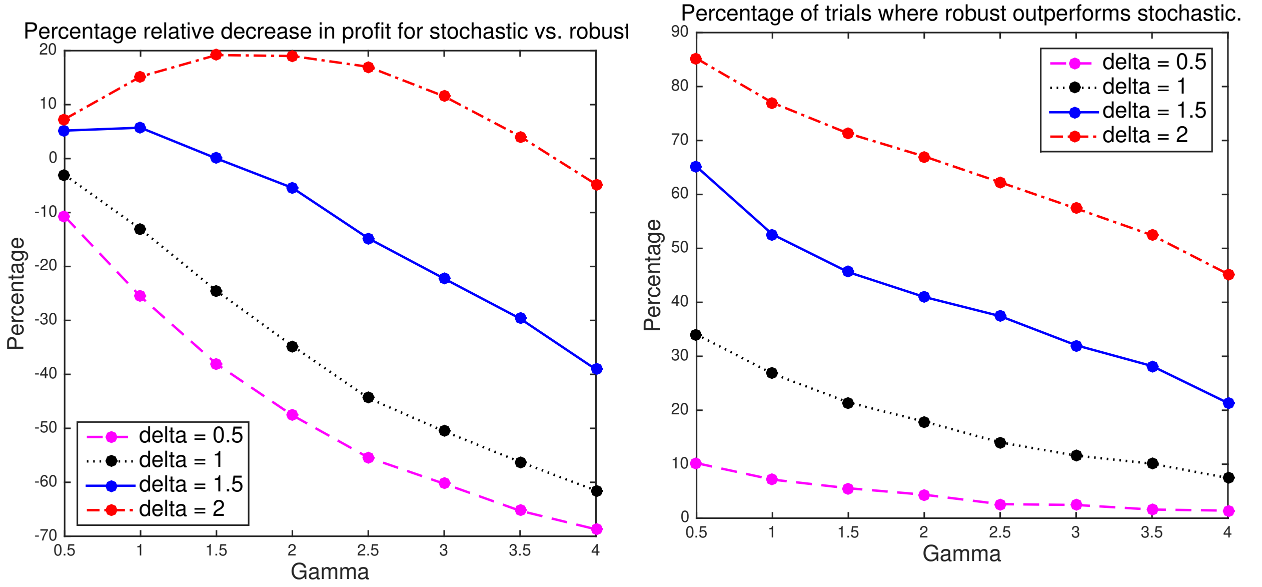
(b) The fraction of trials where the realized profit for the robust model exceeds that from stochastic model.

Figure 5.2: Numerical results for the case where the stochastic model assumes an incorrect demand distribution with correct moments.

Figure 5.2a, measures the relative *reduction* in profit (as a percentage) on using the stochastic model over the robust one, averaged over all the simulations. Thus, a positive value implies that the robust model performs better on average. Figure 5.2b displays the fraction of trials where the robust solution outperforms the other. The blue curves in these figures correspond to the case when the stochastic model assumes that the demand in each period follows a gamma distribution with shape parameter $k = 1/5$ and scale parameter $\theta = 2$. Therefore, the mean is $\mu = k\theta = 0.4$, and the standard deviation is $\sigma = \sqrt{k}\theta = 0.89$. The red curves show the results for the case where the stochastic model uses a negative binomial distribution for the demand in each period, with probability of success $p = 0.95$ and number of failures $k = 1$. In this case, $\mu = (1-p)k/p = 0.0526$, and $\sigma = \sqrt{(1-p)k}/p = 0.2354$. In both cases, the robust model performs better on average. Moreover, the performance of the robust model appears to peak around a mid-range value of Γ, which seems reasonable. A very small value of Γ corresponds to too small an uncertainty set that is unable to hedge against adverse affects of misspecifications in the data. On the other hand, a large value of Γ may be too conservative leading to lower worst-case profits.

Misspecified distribution:

In the second set of experiments, we assume that the stochastic model *correctly* assumes that the demand is gamma-distributed, but uses an incorrect estimate of the standard deviation. The CLT-based uncertainty sets also use the same incorrect value of σ. We examine multiple cases based on the relation between the true and assumed shape parameters for the gamma distribution. As before, we choose $k = 1/5$ and $\theta = 2$. Suppose the true shape parameter is $\hat{k} = \delta k$. We performed the experiments for $\delta \in \{1/2, 1, 3/2, 2\}$, and the results are plotted in Figure 5.3. Note that the robust model only uses the information on moments while the stochastic model additionally needs the distribution itself. In the case of misspecified moments but correct distribution, one can argue that the stochastic model at least has partially correct information. It is not unexpected, therefore, that the stochastic model appears to perform better in this case. However, the observed behavior is more interesting – the relative performance of the two methods changes as δ is varied. For the same scale θ, a gamma distribution with a smaller shape-parameter stochastically dominates one with a larger shape-parameter. Therefore, a larger value of k in the stochastic model leads to larger optimal order quantities. When $\delta < 1$, the stochastic model overestimates the shape-parameter and the seller orders more product. On the other hand, σ increases with k. So the CLT-based uncertainty set grows in size as k is increased, which may lead to overly conservative robust solutions. However, as δ is increased, the uncertainty set gets narrower while the stochastic model prescribes lower order quantities. This is one possible explanation for the varying behavior, but the main takeaway from this experiment is that the robust model is not universally better. In a real-world setting, the preferred model (stochastic or robust) may depend on a variety of factors like the underlying demand distribution or the



(a) The average relative reduction in profit in the stochastic model vs. the robust model.

(b) The fraction of trials where the realized profit for the robust model exceeds that from stochastic model.

Figure 5.3: Numerical results for the case where the stochastic model assumes the correct demand distribution with incorrect moments.

nature of inaccuracy in the estimated data.

5.5 Conclusion

In this chapter, we formulated a robust variant of the classical Newsvendor model which accounts for sale revenues as well as purchase, holding and shortage costs. The resulting profit maximization problem was reduced to solving an LP which can be solved analytically. We provided closed-form expressions for the optimal order quantities, and their natural dependence on the relations between the cost and revenue parameters was analyzed. We also solved in closed-form the inner problems for a class of uncertainty sets that subsumes many of the sets studied in the literature. Finally, we compared the performance of the robust model to a stochastic one through numerical experiments.

BIBLIOGRAPHY

[1] C D Aliprantis and K C Border. *Infinite-dimensional analysis: a hitchhiker's guide.* Springer-Verlag, Berlin, Germany, 1994.

[2] A. Ardestani-Jaafari and E. Delage. Robust optimization of sums of piecewise linear functions with application to inventory problems. *Operations Research*, 2016. forthcoming.

[3] C. Bandi and D. Bertsimas. Tractable stochastic analysis in high dimensions via robust optimization. *Mathematical Programming*, 134(1):23–70, 2012.

[4] C. Bandi and D. Bertsimas. Optimal design for multi-item auctions: A robust optimization approach. *Mathematics of Operations Research*, 39(4):1012–1038, 2014.

[5] C. Bandi and D. Bertsimas. Robust option pricing. *European Journal of Operational Research*, 239:842–853, 2014.

[6] C. Bandi, D. Bertsimas, and N. Youssef. Robust queueing theory. *Operations Research*, 63(3):676–700, 2015.

[7] C. Bandi, D. Bertsimas, and N. Youssef. Robust transient analysis of multi-server queueing systems and feed-forward networks. *Queueing Systems*, 2018. Forthcoming.

[8] A. Ben-Tal, A. Goryashko, E. Guslitzer, and A. Nemirovski. Adjustable robust solutions of uncertain linear programs. *Mathematical Programming*, 99(2):351–376, 2004.

[9] A Ben-Tal, L El Ghaoui, and A Nemirovski. *Robust optimization.* Princeton University Press, Princeton, NJ, USA, 2009.

[10] D. Bertsimas and M. Sim. Price of robustness. *Operations Research*, 52(1):35–53, 2004.

[11] D. Bertsimas and A. Thiele. A robust optimization approach to inventory theory. *Operations Research*, 54(1):150–168, 2006.

[12] D. Bertsimas, D. Iancu, and P. Parrilo. Optimality of affine policies in multi-stage robust optimization. *Mathematics of Operations Research*, 35(2):363–394, 2010.

[13] D. Bertsimas, D. Gamarnik, and A. Rikun. Performance analysis of queueing networks via robust optimization. *Operations Research*, 59(2):455–466, 2011.

[14] D. Bienstock and N. Özbay. Computing robust basestock levels. *Discrete Optimization*, 5(2):389–414, 2008.

[15] P Billingsley. *Convergence of probability measures.* John Wiley & Sons, Inc., New York, NY, USA, 2nd edition, 1999.

[16] R Boucherie and N M van Dijk. *Markov Decision Processes in Practice.* Springer, Cham, Switzerland, 2017.

[17] T Cheevaprawatdomrong, I E Schochetman, R L Smith, and A Garcia. Solution and forecast horizons for infinite-horizon non-homogeneous Markov decision processes. *Mathematics of Operations Research*, 32(1):51–72, 2007.

[18] X. Chen, M. Sim, and P. Sun. A robust optimization perspective on stochastic programming. *Operations Research*, 55(6), 2007.

[19] A Garcia and R L Smith. Solving nonstationary infinite horizon dynamic optimization problems. *Journal of Mathematical Analysis and Applications*, 244:304–317, 2000.

[20] A Ghate. Infinite Horizon Problems. Wiley Encyclopedia of Operations Research and Management Science, 2010.

[21] A Ghate. Circumventing the Slater conundrum in countably infinite linear programs. *European Journal of Operational Research*, 246(3):708–720, 2015.

[22] A Ghate and R L Smith. A linear programming approach to nonstationary infinite-horizon Markov decision processes. *Operations Research*, 61(2):413–425, 2013.

[23] A Ghate, D Sharma, and R L Smith. A shadow simplex method for infinite linear programs, forthcoming. *Operations Research*, 58(4):865–877, 2010.

[24] B. Gorissen and D. Hertog. Robust counterparts of inequalities containing sums of maxima of linear functions. *European Journal of Operational Research*, 227(1):30–43, 2013.

[25] W J Hopp, J C Bean, and R L Smith. A new optimality criterion for non-homogeneous Markov decision processes. *Operations Research*, 35:875–883, 1987.

[26] R A Howard. *Dynamic programming and Markov processes.* PhD thesis, MIT, Cambridge, MA, USA, 1960.

[27] D. Iancu, M. Sharma, and M. Sviridenko. Supermodularity and affine policies in dynamic robust optimization. *Operations Research*, 61(4):941–956, 2013.

[28] G N Iyengar. Robust dynamic programming. *Mathematics of Operations Research*, 30 (2):257–280, 2005.

[29] David L. Kaufman and Andrew J. Schaefer. Robust modified policy iteration. *INFORMS Journal on Computing*, 25(3):396–410, 2013. doi: 10.1287/ijoc.1120.0509. URL `http://dx.doi.org/10.1287/ijoc.1120.0509`.

[30] I Lee, M A Epelman, H E Romeijn, and R L Smith. Simplex algorithm for countable-state discounted Markov decision processes. `http://www.optimization-online.org/DB_HTML/2014/11/4645.html`, 2014.

[31] H. Mamani, S. Nassiri, and M. Wagner. Closed-form solutions for robust inventory management. *Management Science*, 63(5), 2017.

[32] K. Natarajan, M. Sim, and J. Uichanco. Asymmetry and ambiguity in newsvendor models. *Management Science*, 2018. Articles in Advance.

[33] A Nilim and L El Ghaoui. Robust control of markov decision processes with uncertain transition matrices. *Operations Research*, 53(5):780–798, 2005.

[34] G. Perakis and G. Roels. Regret in the newsvendor model with partial information. *Operations Research*, 56(1):188–203, 2008.

[35] M L Puterman. *Markov Decision Processes : Discrete Stochastic Dynamic Programming*. John Wiley & Sons, New York, NY, USA, 1994.

[36] H. Scarf. A min-max solution of an inventory problem. In *Studies in the Mathematical Theory of Inventory and Production*, pages 201–209. Stanford University Press, 1958.

[37] I E Schochetman and R L Smith. Infinite horizon optimization. *Mathematics of Operations Research*, 14(3):559–574, 1989.

[38] I E Schochetman and R L Smith. Finite dimensional approximation in infinite dimensional mathematical programming. *Mathematical Programming*, 54(3):307–333, 1992.

[39] C. See and M. Sim. Robust approximation to multi-period inventory management. *Operations Research*, 58(3):583–594, 2010.

[40] S Sinha, J Kotas, and A Ghate. Robust response-guided dosing. *Operations Research Letters*, 44(3):394–399, 2016.

[41] O. Solyalı, J. Cordeau, and G. Laporte. The impact of modeling on robust inventory management under demand uncertainty. *Management Science*, 62(4):1188–1201, 2016.

[42] G. Vairaktarakis. Robust multi-item newsboy models with a budget constraint. *International Journal of Production Economics*, 66:213–226, 2000.

[43] M. Wagner. Fully distribution-free profit maximization: The inventory management case. *Mathematics of Operations Research*, 35(4):728–741, 2010.

[44] M. Wagner. Online lot-sizing problems with ordering, holding and shortage costs. *Operations Research Letters*, 39(2):144–149, 2011.

[45] M. Wagner. Robust inventory management: An optimal control approach. *Operations Research*, 2018. Articles in Advance.

[46] Y Ye. The simplex and policy-iteration methods are strongly polynomial for the Markov decision problem with a fixed discount rate. *Mathematics of Operations Research*, 36 (4):593–603, 2011.