Robust Optimization of a Class of Queuing and Inventory Control Problems

by

Zeynep Turgay

A Thesis Submitted to the
Graduate School of Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Philosophy of Doctorate

in

Industrial Engineering

Koç University

December, 2011

Koç University

Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a thesis by

Zeynep Turgay

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Committee Members:

Prof. Dr. Fikri Karaesmen (Advisor)

Assoc. Prof. Lerzan Örmeci (Co-Advisor)

Prof. Dr. Ş Ilker Birbil

Asst. Prof. Dr. Onur Kaya

Assoc. Prof. Levent Kockesen

Asst. Prof. Dr. Pelin Canbolat

Prof. Dr. Serpil Sayin

Date:

To my grandparents...

ABSTRACT

In this thesis, we study the robust counterparts of some classical stochastic dynamic programming problems. In classical stochastic dynamic programming, the transition probabilities of the underlying Markov Chain or other problem parameters are assumed to be known with certainty. We focus on the case where the transition probabilities or other input parameters have to be estimated from data and therefore are defined as an uncertainty set. Robust dynamic programming addresses this problem by defining a max-min game between Nature and the controller such that Nature's solution is incorporated to the problem as the minimizing argument whose feasible set is the uncertainty set. We consider robust counterparts of classical problem using this approach. For a wide set of examples from inventory and queueing control, we examine the structure of such robust counterparts and the structure of their optimal policies. Constructing a systematic approach for exploiting the usefulness of the event-based method is the primary tool in order to identify these properties. This systematic approach enables us to show that the structure that governs the optimal policy of the classical problem is retained for its robust counterpart for a wide set of cases at the highest level of generality. In addition to this, we elaborate on the relationship between the perfect duality property of a robust counterpart and its optimal policy and an associated computationally efficient solution. Based on this latter approach, we propose less conservative robust approaches that are both computationally tractable and responsive to changes in the problem parameters.

ACKNOWLEDGMENTS

I would like to take this opportunity to express my deepest gratitude to my advisor Fikri Karaesmen.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ÖZETÇE

Bu tezde bir dizi klasik stokastik dinamik problemin dayanıklı versiyonları ele alınmıştır. Klasik stokastik dinamik programlamada, Markov Zinciri olasılık parametreleri veya diğer problem parametrelerinin kesin olarak bilindiği varsayılır. Çalışmamızda, tarihsel veriden veya diğer kaynaklardan tahmin edilmek durumunda olan bu parametreler bir belirsizlik kümesi olarak ele alınmaktadır. Dayanıklı dinamik programlamada, Doğa ve kontrolör arasında tanımlanan bir max-min oyunu çözülmesi ile söz konusu belirsizlik kümesinin problem içerisinde modellenmesi sağlanır ve Doğa bu belirsizlik kümesi içerisinden problemin hedef fonksiyonunu minimize edecek argümanı seçer. Bu sayede elde edilen problem klasik problemin dayanıklı eşleniğidir. Tezde, envanter ve kuyruk teorisi problemlerinden geniş bir küme çalışılarak, elde edilen bu dayanıklı eşlenik problemlerin ve bu problemlerin optimal çözümlerinin yapısal özellikleri incelenmiştir. Söz konusu özelliklerin belirlenmesi öncelikle olay tabanlı yaklaşımın sistematik olarak kullanılması ile olmuştur. Kurulan bu sistematik yaklaşım sayesinde, olabilecek en genel seviyede klasik problemde ve ilişkili optimal çözümünde varolan matematiksel özelliklerin dayanıklı eşlenikte de varolduğu gösterilmektedir. Bunun yanında dayanıklı eşleniğin mükemmel ikilik özellik özelliğine sahip olması ile optimal politikasının ve bu politikanın hesaplanabilirliğine yönelik ilişkiler incelenmektedir. Bu inceleme sayesinde, max-min yaklaşıma göre daha esnek olup bunun yanında gerek hesaplanabilirlik açısından gerekse de problem parametrelerindeki değişikliklere duyarlılık açısından etkili olan dayanıklı çözümler önerilmektedir.

Chapter 1

INTRODUCTION

1.1 Motivation

In this dissertation, we study a set of problems from queueing and inventory control theory. These problems are consist of different replenishment (service) and demand (arrival) components. Each component has a certain probability of event occurrence which could correspond to the arrival of a customer or the completion of a service/production. The optimal solution maximizes the total expected revenue and is obtained by stochastic dynamic programming for discrete time intervals by evaluating the state of the system in the perception of immediate gain and future expected gain. These problems have been widely studied by several authors for more than 30 years and the event-based dynamic programming approach has been found to be useful in order to identify the structure of the optimal policies determined by the thresholds such as protection levels, base stock levels or switching curves. In order to present an example, consider the classical single item inventory management problem without replenishment (also known as single leg airline revenue management problem) where a certain number of inventory items have to be sold over a fixed time horizon to different classes of customers that are distinguished by the prices they are willing to pay. The demand is uncertain and the probability of selling to any customer class is estimated from the historical data. The objective of the firm is to maximize expected revenue over the sales horizon. The optimal actions are obtained for discrete time intervals by evaluating each possible inventory status for the classical problem and the optimal action for each stage and state does not depend on the decisions of other stage and states. Therefore, the objective function (value function) is obtained at each stage and state separately. Mathematical properties of the value function such as concavity, supermodularity (submodularity) and increasingness (decreasingness) are used to define the optimal policy, i.e. the set of optimal actions over the horizon. The optimal rationing policy defines the optimal sales decision for each stage and state independently and has a monotone structure and defined as a threshold type, i.e. to sell to the lower classes if the inventory level is higher than the protection level at the given time. As one can already see, if the time left is short enough, then it is optimal to satisfy demand from all customer classes, i.e., the protection level is zero. Similarly, when the time left is long enough then it may be optimal to sell only to the highest paying class. A replenishment component can be added to the model without violating the structure of the optimal rationing policy, and the optimal replenishment policy (set of all optimal actions defined for each stage and state) is characterized by base stock level. Therefore, the optimal controller does not make a decision to produce unless the number of inventory items falls below a certain level (base stock). Such kind of a structure that governs the optimal policy enables managing the operations easily and improves the solution time of the problem considerably.

Recently, as robust optimization became a popular topic, robust counterparts of these problems where problem parameters (transition probabilities, cost and reward values) are represented by uncertainty models rather than certain values have been studied by several authors. There are several approaches for incorporating robustness into the model and one of the most well known methods is based on defining a max-min game between the controller and Nature. The controller decides according to the fact that Nature will select its decision variable (uncertain problem parameters) in order to minimize the objective function (value function). Although the max-min is a basic game theoretic model, it is a great challenge for dynamic programming (DP) algorithms. The problem is twofold. The first question is whether it is possible to solve these DPs in a reasonable time or not. It has been shown that the robust counterparts of the DP problems can be solved in the same polynomial time depending on the structure of the uncertainty set ([25]). The second question is the existence of a structure that governs the optimal policy. Several authors from DP literature studied specific problems for special uncertainty sets in order to find solutions to real life problems. Our aim is to analyze the structure of these problems at the highest level of generality. Therefore, we examine a wide set of problems and we use a general definition of uncertainty set whenever possible. This approach also lets us propose well performing semi-robust optimal solutions that are computationally tractable.

1.2 Contributions

The main emphasis of the thesis is to analyze a wide set of problems in order to analyze structure of the optimal policy and to explore how the robust optimal policy and the objective function change with respect to perturbations in the uncertainty sets. Instead of restricting ourselves to a specific uncertainty model, we put our efforts into obtaining these relationships by using general uncertainty model. The perfect duality is a commonly-used term throughout the thesis, it is the case where the sequence of the game does not have any effect on the objective function. Perfect duality is an efficient tool for identifying the structural properties associated to the optimal policy. Because of this reason, we quest the relationship between the perfect duality and the structure of the optimal policy. This relationship has an importance especially for computational tractability and solution time of the problem and is thoroughly studied in Chapter 3. We show in Chapter 3 that all of the problems considered in the scope of this dissertation are perfectly dual if the transition probabilities are uncertain. Hence, the structure of the optimal policy of the classical problem is retained for the robust counterpart for all of the problems we consider here. If we consider the single-item inventory management with replenishment problem mentioned above, the optimal rationing policy and the optimal replenishment policy are of threshold type regardless of the structure of the uncertainty set representing transition probabilities. It is important to note that the reverse statement is not always true. The optimal solution can have the same structure with the classical problem even if the problem is not perfectly dual and an illustration is included in Chapter 5. We also explore the case where other problem parameters such as cost and rewards are uncertain. In this case, we show that the problem is neither perfectly dual nor the optimal policy is structured. However, it is interesting that the robust value function has the same mathematical properties with the classical value function.

Apart from the difficulties of obtaining a solution, the robust approaches are subject to debate as they are deemed to be too conservative and their solution may deviate considerably from optimality. Our analysis on the structural properties of robust problems enables us to construct better performing semi-robust solutions while preserving computational efficiency. We performed several numerical experiments and demonstrated how robust approaches can be useful for improving performance when problem parameters are uncertain. It is notable that semi-robust solutions are also efficient in improving the variability of the solution when problem parameters are known with certainty.

1.3 Outline

The outline of the thesis is as follows:

Chapter 2: In Chapter 2, we provide definitions of the terms and methods used throughout the thesis such as dynamic programming, robust dynamic programming, perfect duality and event-based approach. We also describe the components that constitute the problem models and define the operators which represent the mathematical definitions of these components. Structural properties of the optimal policies are derived from mathematical properties of the value function function such as concavity, supermodularity and K-concavity. We explain these mathematical properties in that perspective and describe how the optimal policies relate to these mathematical properties. Last, we provide a brief summary for some well-known publications from the robust dynamic programming literature in that section.

Chapter 3: In this chapter, we provide the general results for a wide range of problems that can be constituted of 12 different operators representing components. Here, we consider a general definition of the uncertainty set and do not impose any particular mathematical structure on the uncertainty model. First, we consider that the transition probabilities are uncertain and at that level of generality we show that all of the mathematical properties of the value function and the structural properties of the associated optimal policies propagate to the robust counterpart. We show a stronger result than perfect duality in that case; the independence of the optimal action from Nature's posteriori decision. This kind of an independence enables us to exploit the event-based representation and improves the solution time of the problem considerably. Then, we extend our results for the uncertain parameters such as cost and rewards. In the latter case, neither the problem is perfectly dual nor the optimal policy associated to the operators (components) whose parameters are uncertain is structured. However, the structural properties of the robust value function are the same as the classical problem. Last, we discuss some multi-dimensional problems.

Chapter 4: In this chapter, we consider that the transition probabilities are uncertain and elaborate on our results given in Section 3 by restricting the uncertainty set to be represented by the interval uncertainty model. We provide two particular models here, single-item inventory management problem with and without replenishment. As already established in Section 3, the optimal policy is of threshold type, i.e. optimal rationing policy is defined as protection levels and optimal production policy is defined as base stock level. In this section, we show that these thresholds monotonically change with respect to the size of the uncertainty set. We utilize this behavior in order to obtain semi-robust solutions and compare their performances.

Chapter 5: In this chapter, we investigate a classical inventory control problem of Scarf [14] and consider the case where the replenishment has a fixed setup cost. This is an interesting example where the solution of the problem is not perfectly dual, however the optimal policy has the same structure with the classical problem. The second consideration of this chapter is the variability of the expected profit. We demonstrate how a robust solution improves the variability of a solution considerably without deviating from the optimal policy even if all the problem parameters are certainly known.

Chapter 6: In contrast with the rest of the thesis, we study different economical methods than the maximin game in order to propose further research. We demonstrate an interesting finding of the research, a myopic policy based on robustness that almost achieves the same results with the optimal policy for the admission control problem in the presence of financial commitments. The problem is very similar to the classical single-item inventory management problem without replenishment with a major difference. There are installments that have to be paid at given due dates. There is a penalty that is equal to the interest rate for the unpaid amount or a one-time financial loss in the case of a missed opportunity. The challenge of the problem is due to the solution time since the cash position is also a state variable that can take continuous values. Besides, the optimal policy has no easy structure and theoretically, the cash position can change between (-∞,∞). Hence, the state space is uncountable. We compare two algorithms, the first algorithm is based on the semi-robust decision. The second algorithm dynamically adjust robustness by evaluating the cash position. Although the semi-robust algorithm performs well, the second algorithm performs almost up to 100% of the optimal policy.

Chapter 2

LITERATURE REVIEW AND BACKGROUND

2.1 Background

2.1.1 Robust Optimization Overview

In many practical optimization problems, the input data to the problem is not known with certainty and is approximated or estimated. This, if ignored, may cause significant suboptimality or infeasibility for the optimization problem considered. Robust optimization is a specific methodology that addresses this problem and has received a lot of attention lately [4].

There are several ways of incorporating robustness in a problem. Among them we use the maximin approach -also known as the absolute robust decision- suggested by Ben-Tal, El Ghaoui and Nemirovski [4] to be used in dynamic programming problems. In the following, we present a summary of the maximin approach.

Consider the following linear optimization problem of real numbers:

$$\max_x \{c^T x : Ax \leq b\},$$

where T denotes matrix transpose operation, $x \in \Re^n$ represents the decision variables, $c \in \Re^n$ is the cost vector, A is the $m \times n$ constraint matrix, and $b \in \Re^m$ is the right hand side vector. In most of the situations, at least some of the values associated to c, A, b are not certain, rather they belong to an uncertainty set $U \subset \Re^{(m+1)\times(n+1)}$, where U is any discrete or compact subset of $\Re^{(m+1)\times(n+1)}$. The robust counterpart of the problem is represented in the following formulation in order to model the uncertainty as a parameter in the problem:

$$\max_x \{ \min_{(c,A,b) \in U} c^T x : \; Ax \leq b\}. \tag{2.1}$$

As can be clearly seen, the robust solution approach models all possible values instead of certain values that are introduced in the original problem as constant values.

This model also defines a structured game between two players. The **controller** represents the decision maker in the original problem while **Nature** represents the opponent that acts on observing the controller's choice. While the controller tries to maximize its revenue, Nature always takes the worst action with respect to the controller in order minimize the revenue. The optimal action of the controller is the best performing outcome among all actions that are incorporated as a parameter of nature's inner problem (which may be a linear or nonlinear optimization problem). In most simple words, the controller knows in advance that Nature will present conditions to ruin its decision and the best decision is the best performing one among all ruined ones.

We use the term "*perfect duality*" throughout the thesis. In order to define perfect duality, let's first define the dual counterpart for the revenue maximization problem. In the dual setting, the controller observes Nature's solution then decides its optimal policy. Nature's optimal policy in that case is the solution that achieves the minimum result among them. Hence, the dual counterpart of the robust problem is given in the following equation:

$$\min_{(c,A,b)\,\in\, U} \{\max_{x} c^T x : \; Ax \leq b\}. \tag{2.2}$$

The dual counterpart of the solution is the case where the resultant values of the Equation (2.1) and Equation (2.2) are equal to each other, i.e:

$$\max_{x}\{\min_{(c,A,b)\,\in\, U} c^T x : \; Ax \leq b\} = \min_{(c,A,b)\,\in\, U} \{\max_{x} c^T x : \; Ax \leq b\}.$$

2.1.2 Other Approaches

Apart from the maximin approach that is used as the main approach in the context of this dissertation, there are several other ways of formulating robustness. One particular method is the minimax regret suggested by Kouvelis and Yu [20] that minimizes the total deviation from the optimal solution of each scenario that may be realized throughout the uncertainty set. In other words, minimax regret seeks a decision that has the minimum total deviation from optimality. The minimax regret is less conservative than the maximin approach but it is usually computationally intractable. Instead, we adopt the minimax regret approach in order to evaluate the performance of the robust solution. Throughout the thesis there are numerical examples demonstrating the performance of robust solutions with a trade off from the optimality.

Apart from the maximin approach and minimax regret approach different techniques can be adopted in order to attain robust solutions. We adopt the *"prospect theory"* from behavioral economics. Prospect theory is explained in Gilboa [12] with the following example: Consider the following two cases that have the same expectation for each alternative:

CASE 1: Suppose you have $1000 for sure and the following two options:

1. To get additional $500 with a probability of 100%
2. To get $1000 with a probability of 50% and get nothing with a probability of 50%.

CASE 2: Suppose you have $2000 for sure and the following two options:

1. To lose additional $500 with a probability of 100%
2. To lose $1000 with a probability of 50% and lose nothing with a probability of 50%.

Most people prefer Alternative 1 in the first case whereas they prefer Alternative 2 in the second case. Preferences of people change with respect to the total amount they already have.

Inspiring from Prospect theory we introduce the cash position to the dynamic programming equation as a state variable in Chapter 6.1. However this modification makes the algorithm computationally intractable. On the other hand, it is a practical approach to evaluate the cash position on realization. We show that such a simple algorithm performs better than the classical and robust solutions for a specific problem. For more information, please refer to the Chapter 6.1.

2.1.3 Certainty, Stochasticity and Uncertainty

In order to present robust optimization, definitions of certainty, stochasticity and uncertainty shall be revisited. Under certainty, problem parameters have certain values, or they may have forecasted values and may be expressed as a best guess corrected by a safety factor. Hence, the problem is solved with a single set of values for the parameters. The difference between a stochastic problem and a robust problem is less obvious. According to the decision analysis literature, uncertainty refers to random quantities with known probability measures whereas ambiguity refers to unknown probability measures. Therefore, uncertain data of an optimization problem can be converted into a stochastic optimization problem. However, in order to stochastically model the problem, certain conditions must be met. First, the uncertain data have to have a computationally tractable probability distribution. Second, the probability distribution has to be determined from the uncertainty set. Even if these conditions are perfectly met, stochastic optimization usually applies only in risk neutral decision making. In order to overcome this issue, the problem is constrained with the probability of deviation from the expected result. In this case, the problem is represented as chance-constrained stochastic optimization as given in the following :

$$\max_{x,t}\{t : Prob_{(c,A,b)\sim p}\{c^T x \geq t \quad Ax \leq b\} \geq (1-\varepsilon)\},$$

where ε is the desired tolerance and P is the distribution data, i.e. the probability distribution of the uncertain quantities. Although chance constrained stochastic optimization is a more flexible approach than robust optimization, there are still issues associated to stochastic optimization that have to be addressed. There may be more than one distribution that fits a given uncertainty set. This last issue is the most important one since it may result in degradation of performance for disadvantageous realizations in the uncertainty set. Hence, there needs be an efficient subjective determination or very careful estimation of probabilities in order to convert an uncertain problem into a stochastic problem.

Throughout this dissertation, the nominal problem will refer to the case where the transition probabilities and/or the problem parameters (cost, reward) are represented with certain values and the robust problem will refer to the case where the transition probabilities and/or the problem parameters (cost, reward) are represented by their respective uncertainty sets.

2.1.4 Basic Stochastic Dynamic Programming Problem

In this section, we present the basic stochastic dynamic programming problem that is also known as a Markov Decision Process (MDP) and its optimality equation. The information we give in this section is based on Chapter 1 of Bertsekas [5].

In our problem, $t = 0, 1, ..., T$ denotes the horizon where T denotes the last stage in the horizon. The system state $x \in X_t$ is the state of the system that is selected from set X_t. A_{X_t} denotes set of admissible actions defined for every stage t and state x independently. Note that the controller is allowed to choose the action $a \in A_{X_t}$ independently for all (x, t) pairs. The random disturbance ω_t in the period t is characterized by a probability distribution $p_{t,a}(x, y)$, therefore, w_t is independent of prior disturbances. At each stage, depending on the controller action $R_{t,a}(x)$ is gained as an immediate reward, but every decision has both immediate and long-term consequences. At each stage t, $p_{t,a}(x, y)$ denotes the transition probability from state x to state y when decision a is selected, $\sum_y p_{t,a}(x, y) = 1$. The sequence of actions $\pi = \{a_0, a_1, ...a_{N-1}\}$ is referred to as a policy. The revenue $j_0^{\pi}(x_0)$ represents the collected revenue when policy π is applied and initial state is x_0. The optimal policy is the one that achieves the optimal revenue $j_0^{\pi^*}(x_0)$ over all polices.The basic problem is defined as:

$$\max_{\pi \epsilon \Pi} j_T^{\pi}(x_0) \; \forall \; x_0 \; \epsilon \; X_0$$

$$j_T^{\pi}(x_0) = E\{R_{T,a_T}(x_T) + \sum_{t=0}^{t=T-1} R_{t,a}(x_t, \omega_t)\}.$$

The problem can be solved recursively starting from the last stage. Then at every stage a tail subproblem is solved by backward induction in order to optimize the total reward and future expected revenue. The basic dynamic programming algorithm -also known as Bellman recursion- is as follows:

$$v_t(x) = \max_{a \epsilon A_{X_t}} \{R_{t,a}(x) + \sum p_{t,a}(x, y) v_{t+1}(y)\},$$

where $v_T(x) = R_T(x)$ and $v_t(x)$ is the expected optimal value for a tail subproblem from t to T and expected value of any policy $\pi = \{a_0, a_1, ...a_{T-1}\}$ is given by:

$$v_t^{\pi}(x) = R_{t,a}(x) + \sum p_{t,a}(x, y) v_{t+1}(y).$$

It is clear that $v_0(x_0) = j_T^{\pi^*}(x_0)$, and the dynamic programming algorithm selects the action that maximizes the immediate reward and the future expected revenue for each state and stage.

2.1.5 Robust Dynamic Programming

In this section, we present the background on discrete time robust stochastic dynamic programming. Nilim and El Ghaoui [25] and Iyengar [16] simultaneously studied robust stochastic dynamic programs and showed some important properties under certain assumptions.

In order to present details of robust dynamic programming, first the definition of uncertainty shall be given. Rather than a fixed value, transition probabilities are represented as a set -uncertainty set- in the robust problem. The uncertainty defining transition probabilities or rewards (cost) may vary with action, state and stage. At any stage t, and state x, if action a is chosen, then probability of next state is $p_{t,a}(x, y) \epsilon \mathcal{P}_{t,a}(x, y)$. Hence, state transition is uncertain and defines a set rather than specific values. In addition to this, state transitions do not restrict each other, for example probability of transition from state x to state y are independent, i.e:

$$\boldsymbol{\mathcal{P}}_{t,a}(y) = P_{t,a}(1, y) \times P_{t,a}(2, y) \times P_{t,a}(3, y)......P_{t,a}(n, y),$$

for all $y \in X_{t+1}$ where $X_t = \{1, 2, ...n\}$. This property is called rectangularity ([16], [25]). Rectangularity provides nature the capability of independently selecting its action for every stage, state and controller action. Hence, it designates the independence of uncertainty set variables for all states, stages and action combination (a, t, x).
The representation of rectangularity is given by Nilim and El Ghaoui [25] with the following definition for set of all admissible policies $\mathcal{P}$:

$$\mathcal{P} = [\bigotimes_{a \in A_{X_t}} \mathcal{P}_{(t,a)}(x, y)]^T,$$

where $\bigotimes$ represents the direct product. Therefore, the set of all admissible policies of Nature given π is:

$$\tau^\pi = \tau_0 \ \times \tau_1 \ \times \tau_2 \ \times ... \times \tau_T.$$

Based on this independence property, the following properties were shown by Iyengar [16] and Nilim and El Ghaoui [25] on slightly different formulations. The problems they studied are presented in the following together with the main results.

The first problem shown in this section is defined by Iyengar is given in below equation:

$$j_N(x) = \max_{\pi \epsilon \Pi}\{j_N^\pi(x)\} = \max_{\pi \epsilon \Pi}\{\min_{P \epsilon \tau^\pi}\{R_{t,a}(x, y) + \sum_{t=0}^{N-1} p_{t,a}(x, y) v_{t+1}(y)\}\},$$

where controller policy $\pi = \{a_0, a_1, a_2, ...\}$ and Nature policy against a given controller policy $\tau^\pi = \tau_0 \times \tau_1 \times \tau_2 \ \times ... \times \tau_N$.
Note that, different from our nominal problem R_t depends not only on the current stage but also on the next stage. Hence, $p_{t,a}(x, y)$ also affects R_t. The main results are:

1. The robust optimal policy π^* satisfies the Belman equation for the finite horizon problem. Hence:

$$v_N(x) = r_N(x_N)$$
$$v_t(x_t) = \max_{a \epsilon A_{x_t}} \left\{ \min_{P \epsilon \tau^{\pi}} \{R_{t,a}(x,y) + \sum p_{t,a}(x,y) v_{t+1}(y)\} \right\}$$
$$t = 0, 1, ...N - 1.$$

The infinite horizon problem was first studied by Satia and Lave [28], they proposed a robust policy iteration algorithm that solves the robust DP problem in infinite horizon setting. However, they fail to show that Nature's policy in the robust setting is stationary. Results of Iyengar [16] also include the following statement:

2. Nature's optimal policy is stationary for the infinite horizon problem, i.e. its optimal policy is fixed for all x, a pairs. Thus, the controller can be restricted to deterministic policies without any loss in performance in the infinite horizon. Hence, the robust optimal policy can be obtained by policy iteration.

The problem studied by Nilim and El Ghaoui [25] was given in the following recursion:

$$v_t(x) = \max_{a \epsilon A} \{R_{t,a}(x) + \min_{p \epsilon P_{t,a}(x)} \sum p_{t,a}(x,y) v_{t+1}(y)\}$$
$$t = 0, 1, ..., T - 1.$$

The main difference between these two equations is that in the above formulation the instant reward or the instant cost of each stage is independent of next stage. In addition to Iyengar's results [16] , Nilim and El Ghaoui [25] also show that:

1. The problem is perfectly dual, i.e, the optimal actions of the controller and Nature are independent of the sequence of the game. In the dual form of the game, controller acts upon observing the Nature's optimal action.

$$\max_{\pi \epsilon \Pi} \min_{P \epsilon \tau} v_0(x) = \min_{P \epsilon \tau} \max_{\pi \epsilon \Pi} v_0(x).$$

According to the results of Nilim and El Ghaoui [25] perfect duality only holds under the assumption that Nature can select an independent probability distribution for each action and state.

2. Perfect duality still applies for the infinite horizon. Moreover if we restrict the controller and Nature to stationary optimal policies then where Π_s, τ_s denotes the allowable set of actions of the controller and Nature respectively in that case:

$$\max_{\pi \epsilon \Pi} \min_{P \epsilon \tau} v_\lambda(x) = \max_{\pi \epsilon \Pi_s} \min_{P \epsilon \tau_s} v_\lambda(x) = \max_{\pi \epsilon \Pi_s} \min_{P \epsilon \tau} v_\lambda(x) = \max_{\pi \epsilon \Pi} \min_{P \epsilon \tau_s} v_\lambda(x),$$

   for all $x \epsilon X$ where λ is discount vector.

2.1.6 Uncertainty Sets

In this section, we provide the uncertainty sets that are commonly-used in the literature in order to model the uncertainty representing the transition probabilities. We provide some basic properties of these sets. Further information can be found in Ben-Tal, El Ghaoui and Nemirovskiĭ [4] Section 2 and Section 13. Although our results also apply to discrete sets, we give the definitions of continuous uncertainty sets used from the literature. First we present some information for the uncertainty sets used in order to represent transition probabilities, later we summarize uncertainty sets used in order to represent other problem parameters, i.e. cost and reward, in our setting.

Uncertainty sets defining transition probabilities:

Uncertainty sets defining transition probabilities are defined in below:

1. The Scenario Model defines a finite collection of distributions:
   $\mathcal{P} = Conv\{p^1, ..., p^k\}$ where $p^s \in \triangle_n$, $\triangle_n$ is a standard simplex in $\Re_{n+1}$ and defined as $\triangle_n = \{p = [p(1)...p(n)] \in \Re^n_+\}, : \sum_j p(j) = 1,\ s = 1, ..., k$ are given, and $\mathcal{P}$ is the

convex hull of $\{p^1, ..., p^k\}$. This is also known as the polytope model. The scenario model represents the convex hull of a finite number of realizations.

2. The interval model is a special case of the polytope model, where event probabilities do not have a dependence on each other, i.e, $\mathcal{P} = \{p : \underline{p} \leq p \leq \overline{p}, \sum_j p(j) = 1\}$.

3. The likelihood model has the form:
$\mathcal{P} = \mathcal{P}(\rho) := \{p\epsilon\triangle_n : L(p) := \sum_{i=1}^{n} q(i)ln[q(i)/p(i)] \leq \rho\}$, $\triangle_n = \{p = [p(1)...p(n)]\epsilon\Re_+^n : \sum_j p(j) = 1\}$ where $q\epsilon\triangle_n$ is a fixed reference distribution (initial estimate) and $\rho \geq 0$ is the uncertainty level, with $\rho = 0$ the uncertainty set is fixed to initial the estimate q and $L(p)$ is the likelihood function. Maximum likelihood models define an uncertainty around that prior distribution whose deviation is measured by ρ. Examples of likelihood models can be found in [9], [25].

4. The entropy model is defined as:
$\mathcal{P} = \mathcal{P}(\rho) := \{p \in \triangle_n : D(p \setminus \setminus q) := \sum_{i=1}^{n} p(j)ln[q(i)/p(i)] \leq \rho\}$, where $\triangle_n$ $= \{p = [p(1)...p(n)] \in \Re_+^n : \sum_j p(j) = 1\}$, $D(p\setminus\setminus q)$ is the Kullback-Leibler divergence between distribution p and the reference distribution $q \in \triangle_n$, and $\rho \geq 0$ is the uncertainty level. Examples of entropy based models can be found in [17], [23].

5. Ellipsoidal model is defined as $\mathcal{P}(\rho) = \{p \in \triangle_n : (p-q)^T H(p-q) \leq \rho^2\}$ where $q \in \triangle_n$, is a fixed reference distribution (initial estimate) and $\rho \geq 0$ is the uncertainty level (deviation from q), $\triangle_n = \{p = [p(1)...p(n)] \in \Re_+^n : \sum_j p(j) = 1\}$, $H \geq 0$ is given. Examples of ellipsoidal models can be found in [7].

Uncertainty Sets Defining Cost And Reward:

There are several approaches for defining uncertainties on parameters. The commonly employed uncertainty model is to define the parameters between lower bound and upper bound values. Chance constrained criterion is another common approach for linear and nonlinear optimization problems. Delage and Mannor [9] used a chance constrained MDP problem that is computationally tractable. The following equation defines the chance constrained form of a linear equality given in Section 1.1. In the chance constrained uncertainty model, ζ is the perturbation vector that has a known probability distribution P. The decision variable x is robust feasible with respect to perturbation set that belongs to the convex hull of ζ with the desired tolerance ε. We define the chance constrained form of x as: $p(x) \equiv Prob_{\zeta \sim P}\{\zeta : [a^0]^T x + \sum_{l=1}^{L} \zeta_l [a^0]^T x \geq b^0 + \sum_{l=1}^{L} \zeta_l b\} \leq \varepsilon$.

Uncertainty Set Models Used in the Thesis

Most of the results in the thesis are independent of the structure of the uncertainty set. However, we show further characteristics considering the interval uncertainty set and use polyhedral sets for numerical purposes. In Chapter 4 we represent the interval uncertainty models as in the following equation:

$$\mathcal{P}_t = \left\{ \boldsymbol{y} = (y_1, \ldots, y_{n+1}) : \quad 0 \leq \underline{y}_{i,t} \leq y_i \leq \bar{y}_{i,t},\ 0 \leq q \leq \sum_{i=1}^{n} y_i \leq 1 \right\}.$$

According to the above representation the uncertainty set is allowed to vary over the time horizon. Especially for finite horizon inventory problems this flexibility is important in order to represent the fluctuations and patterns over the demand. However, due to the characteristic of the problems we solve, we consider that the uncertainty set representing our problem parameters is independent of the controller's action. In order to provide the rationale, consider the "*single-leg airline revenue management problem*" where the customer arrival probabilities are not affected by the admission/rejection of the controller.

2.2 *Models and Their Structural Properties*

2.2.1 *event-based Dynamic Programming*

In this dissertation, we use the event-based dynamic programming approach suggested by Koole [18] in order to show certain structural properties of the robust dynamic programming problem. Event-based dynamic programming expresses the value function of a given control problem in terms of the composition of different operators corresponding to individual events. Establishing the structure of an optimal policy can then be performed by verifying that the different operators constituting the problem satisfy certain properties. Instead of formulating the problem as a whole, it is formulated as components that apply to the value function $v_{t+1}(x)$, then $v_t(x)$ is obtained by using these operators. The strength of this approach is that it significantly facilitates demonstrating structural properties.

We present the example given by [18] in order to clarify: Consider a queueing problem, where customers arrive according to a Poisson arrival process with a rate of λ and are considered for admission to the system. The probability of service completion is μ, and $\lambda + \mu \leq 1$. If a customer is not admitted to the system then a positive penalty of c is incurred. After uniformization and discretetization, the value function $v_t(x)$ is given as follows:

$$v_t(x) = \lambda \min\{v_{t+1}(x), v_{t+1}(x) + c\} + \mu v_{t+1}(x-1)^+ + (1 - \lambda - \mu)v_{t+1}(x),$$

where $(x-1)^+$ represents $\max(0, x-1)$. Now we construct the dynamic programming algorithm step by step. In the context of event-based dynamic programming, problem is composed of two events, completion of a service and arrival of a new customer. The representation of these operators are as follows:

$$T_D v_t(x) = v(x-1)^+$$
$$T_A v_t(x) = min\{v(x), v(x) + c\}.$$

In the second step, we combine these operators with the uniformization operator as given in the following equation:

$$T_U(T_A v(x), T_D v_t(x)) = p_A T_1 v(x) + (1 - p_A) T_2 v(x),$$

where p_A is the event probability of the first operator.

Clearly, the uniformization operator is just a convex combination operation. Suppose that $v_{t+1}(x)$ is convex and ND (nondecreasing) in x at $t+1$, then it is straightforward to argue that the departure operator and the arrival operator preserve this property. Then, the uniformization operator defined on the departure and the arrival event, $T_U(T_D v_{t+1}(x), T_A v_{t+1}(x))$ preserves these properties also. We can also add a fictitious event operator to the equation via uniformization operator. Since $v_{t+1}(x)$ is convex and ND, it is clear that the resultant equation is convex and ND. Then the value function defined as:

$$v_t(x) = T_U(T_U(T_D v_{t+1}(x), T_A v_{t+1}(x)), T_F v_{t+1}(x)).$$

New operators can be added in this fashion. Several models can be constructed in the same manner such as the below example, that considers admission from a queue upon each event that is either a fictitious event or completion of a service:

$$v_t(x) = T_A(T_U(T_D v_{t+1}(x), T_F v_{t+1}(x))),$$

is convex and ND.

The event-based DP facilitates obtaining structural properties by focusing on the individual terms (events) rather than the whole value function and has been proved to be useful in certain dynamic programming models. We explore the robust counterparts of the classical dynamic programming problems under this framework. Hence, we define a set of operators and their associated characteristics. We give some illustrations from the literature below and then present our operators.

2.2.2 Some Illustrations from Literature

In the thesis, we study robust counterparts of a set of problems from inventory and queueing theory. In this section, we give a brief summary of some of these which are well-known and extensively studied. In the next section, we define the operators used to model these problems.

1. single-item Revenue Management Problem without Replenishment

   Consider the single resource capacity control problem of revenue management where there is neither replenishment nor holding cost. The customers are distinguished depending on their rewards and the first class customer is the one who offers the highest reward. In the discrete formulation of this problem, the total horizon is divided into equal intervals and each stage is treated by its own arrival probability vector, i.e. each demand class has a certain arrival probability and sum of these probabilities is less than or equal to 1. Moreover, the arrival probabilities of different classes are allowed to vary over time in order to model the customer class behavior. In order to maximize the total expected revenue, at each stage, any customer is either accepted or rejected in order to reserve seats for future potential customers that may probably be from a higher class. This problem was previously studied by several authors (Lautenbacher [22], Stidham [31] and Çil, Karaesmen and Örmeci [8]).

2. Make-to-Stock Queue Problem with Multiple Customer Classes

   The make-to-stock queue is a continuous time problem where customers with different rewards arrive according to independent Poisson processes with exponential rates. Similar to the dynamic seat selling problem, the customers are ordered according to their rewards and the first class customer is the one who offers the highest reward. A single server whose processing time is exponentially distributed produces items one-by-one. If a customer is admitted and if there is at least one unit of inventory on hand, its demand is immediately satisfied and a class-dependent instant reward is obtained. If inventory is empty, arriving demands are assumed to be lost. This problem was previously modeled and explored by Ha [14]. The objective function minimizes the total expected cost over an infinite horizon. Similar to the single-item revenue management problem without replenishment, a customer might be rejected in the anticipation of future expected revenue. In addition to this, the controller makes a decision for production in order to establish a balance between the holding cost of inventory items and lost sales.

3. Dynamic Pricing Problem

   The dynamic pricing problem is similar to the single-item revenue management problem with some particular differences. In this case, the controller dynamically adjusts the prices of identical goods to correspond to a customers' willingness to pay. The arriving customer purchases the product at the offered price R with a probability of $1 - F_R$ and does not purchase the product with a probability of F_R. This problem was previously studied by Talluri and Van Ryzin [32], and Gallego et. al [11] and various other authors.

4. Server Assignment Models

In the previous examples, considered state information x of the value function is one dimensional. However, there can be various customers served with different holding costs by the same resource. The objective function is to minimize the long-run average cost of the system. Intuitively, anyone can argue that it is better to serve to the customer with the highest holding cost if their rewards are equal to each other. Further, suppose that the server rate is different for different types of customers. It is again intuitive to conclude that it is better to serve customers that are faster to serve and have higher holding costs than customers that are slower to serve and have lower holding costs. The problem can be modeled and solved recursively by dynamic programming and the optimal policy can be described by μc rule which is also called 'smallest index policy' SIP, i.e. the classes are ordered according to the server rate multiplied by holding cost, i.e. μc and served according to this order. Further details might be found in Section 3.6.1. Extensive knowledge on structural behavior of these models can be found in [19].

2.2.3 Operators

As discussed earlier, we use an event-based approach in order to represent the models we study. Throughout the thesis, the models are constructed with operators. In order to do this, we use a set of operators and illustrate our theoretical results on them. These operators and their definitions are:

ONE DIMENSIONAL OPERATORS

One dimensional operators are used to represent the inventory problems with single-item and the queueing problems with a single class of customers. The operators used in the thesis are given below:

1. *INVENTORY CONTROL OPERATORS*

   Batch Rationing Operator

   The batch rationing operator represents the choice of the number of the customers to be admitted from an arriving batch of class-i customers with batch size B in the inventory systems. Some of the customers in a batch can be admitted while the remaining ones are rejected, which is defined as partial acceptance and κ_i is the number of class-i customers admitted from this batch, and R_i is the reward obtained by admitting one class-i customer.

Definition of the Operator

$T_{BR_i}v(x) = \max_{\kappa_i \leq \min(x,B)}\{\kappa_i R_i + v(x - \kappa_i)\}$,

$T_{BR_i}v(x) = \max_{\kappa_i \epsilon \min(x,B)}\{v(x - \kappa_i) - v(x) + \kappa_i R_i\} + v(x)$.

2. **Rationing Operator**

   The rationing operator is a special case of the batch rationing operator where the batch size B is exactly 1. This operator is defined in below:

   Definition of the Operator

   $T_{R_i}v(x) = \max\{R_i + v(x - 1), v(x)\}$,

   $T_{R_i}v(x) = \{v(x - 1) - v(x) + R_i\}^+ + v(x)$.

3. **Production Rate Operator**

   The production rate operator represents the choice of best service rate in production-inventory systems for production unit i. If the system uses Π_i portion of the service rate, then a nonnegative cost of C_{Π_i} is incurred. The production rate operator is defined as:

   Definition of the Operator

   $T_{PR_i}v(x) = \max_{\Pi \epsilon [0,1]}\{-C_{\Pi_i} + \Pi_i v(x + 1) + (1 - \Pi_i)v(x)\}$,

   $T_{PR_i}v(x) = \max_{\Pi_i \epsilon [0,1]}\{\Pi_i\{v(x + 1) - v(x)\} - C_{\Pi_i}\} + v(x)$.

4. **Production Operator**

   The production operator is a special case of production rate operator where $\Pi_i = \{0, 1\}$ and $C_{i_0} = 0$. The production rate operator is defined according to the following equation:

   Definition of the Operator

   $T_{P_i}v(x) = \max\{v(x + 1) - C_i, v(x)\}$,

   $T_{P_i}v(x) = \{v(x + 1) - v(x) - C_i\}^+ + v(x)$.

5. **Inventory Pricing Operator**

   The inventory pricing operator represents the optimal price to be charged for the arriving customers in inventory systems. $F_Z(.)$ is the cumulative distribution function of the reservation price of an arriving customer, where R is the maximum price a customer is willing to pay. The inventory pricing operator is given by:

   Definition of the Operator

   $T_{IP}v(x) = \max_R\{\overline{F}_Z(R)[v(x-1)+R] + F_Z(R)v(x)\}$,

   $T_{IP}v(x) = max_R\overline{F}_Z(R)\{v(x-1) - v(x) + R\} + v(x)$.

QUEUING OPERATORS

The queueing operators we consider throughout this thesis is given in the following. It is important to note that the waiting room is taken as infinite. The definitions of the queueing operators are symmetric counterparts of the inventory operators that are given in the above part.

1. **Batch Admission**

   The batch admission operator represents the choice of the number of the customers to be admitted from an arriving batch of class-i customers with batch size B in queueing systems. Some of the customers in a batch can be admitted while the remaining ones are rejected, which is defined as partial acceptance κ_i is the number of class-i customers admitted from this batch, and R_i is the reward obtained by admitting one class-i customer. The definition of the batch admission operator is given in below:

   Definition of the Operator

   $T_{BA_i}v(x) = \max_{\kappa_i \leq \min(x,B)}\{\kappa_i R_i + v(x+\kappa_i\}$.

   $T_{BA_i}v(x) = \max_{\kappa_i \epsilon \min(x,B)}\{v(x+\kappa_i) - v(x) + \kappa_i R_i\} + v(x)$.

2. **Admission**

   The admission operator is a special case of the batch admission operator where the batch size B is exactly 1. Its definition is as follows:

Definition of the Operator

$T_{A_i}v(x) = \max\{R_i + v(x+1), v(x)\}$,

$T_{A_i}v(x) = \{v(x+1) - v(x) + R_i\}^+ + v(x)$.

3. **Departure Rate Operator**

The departure rate operator represent the choice of the best service rate in queueing systems. If the system uses Π portion of the service rate, then a nonnegative cost of C_Π is incurred. The definition of the departure rate operator is as follows:

Definition of the Operator

$T_{DR_i}v(x) = \max_{\Pi\epsilon[0,1]}\{-C_\Pi + \Pi v(x-1) + (1-\Pi)v(x)\}$,

$T_{DR_i}v(x) = \max_{\Pi\epsilon[0,1]}\{\Pi\{v(x-1) - v(x)\} - C_\Pi\} + v(x)$.

4. **Controlled Departure Operator**

The controlled departure operator is a special case of the departure rate operator where $\Pi = \{0,1\}$ and $C_0 = 0$. The definition of the controlled departure operator is given:

Definition of the Operator

$T_{CD_i}v(x) = \max\{v(x-1) - C_i, v(x)\}$,

$T_{CD_i}v(x) = \{v(x-1) - v(x) - C_i\}^+ + v(x)$.

5. **Queue Pricing Operator**

The queue pricing operator represent the optimal price to be charged for the arriving customers in queueing systems. $F_Z(.)$ is the cumulative distribution function of the reservation price of an arriving customer, where R is the maximum price a customer is willing to pay. The definition of the queue pricing operator is given in the following equation:

Definition of the Operator

$T_{QP}v(x) = \max_R\{\overline{F}_Z(R)[v(x+1) + R] + F_Z(R)v(x)\}$,

$T_{QP}v(x) = \max_R \overline{F}_Z(R)\{v(x+1) - v(x) + R\} + v(x)$.

6. **Uncontrolled Arrival to a Queue**

   The uncontrolled arrival operator represent the arrival process to a queueing system. The function $a(x)$ is, the probability that an arriving customer joins the system when there are x customers, which we refer to as the joining probability. We assume that $a(x)$ is NI in x. When a is constant, arrival operator models a system where customers enter the system with a fixed probability a, independent of the state, or choose not to enter the system with probability $1-a$. We will call this type of arrivals as regular arrivals, since they do not depend on the state of the system. The definition of the uncontrolled arrival operator is given in the following equation:

   Definition of the Operator

   $T_{UA}v(x) = a(x)v(x+1) + (1-a(x))v(x)$,

   $T_{UA}v(x) = a(x)\{v(x+1) - v(x)\} + v(x)$.

   The special case of this operator is the case where $a = 1$:

   $T_{UA}v(x) = v(x+1)$.

7. **Uncontrolled Departure from a Queue**

   The uncontrolled departure operator represents the departure of an existing customer from the system, where the service rate may depend on the state of the system. The function $b(x)$ corresponds to the probability of a service completion when the system has x customers. We assume that $b(x)$ is an ND function of x. The definition of the uncontrolled arrival operator is given in the following equation:

   Definition of the Operator

   $T_{UD}v(x) = b(x)v(x-1) + (1-b(x))v(x)$,

   $T_{UD}v(x) = b(x)\{v(x-1) - v(x)\} + v(x)$.

   The special case of this operator is the case where $b = 1$:

   $T_{UA}v(x) = v(x-1)$.

   Our theoretical results can be extended to multidimensional queuing operators as well.

Below we list some operators from queueing theory. The multidimensional queues are described in Section 3.6.2 in more detail.

MULTIDIMENSIONAL OPERATORS

The multidimensional operators are mainly used in the queueing systems when there is more than one server with different properties. In the multidimensional queues x represents a row vector where x_i designates the number of items in the ith queue. When costs associated to different classes are different, such as holding cost or service cost, the system state is defined as a row vector in order to handle this situation. In the following, the multidimensional counterparts of the one dimensional queueing operators are given:

1. **Uncontrolled Arrival to a Queue**

   The uncontrolled arrival operator of a class-i customer is defined according to the following equation, where e_i denotes a unit row vector whose ith component is 1 and the rest is 0:

   $T_{UA_i}v(x) = v(x + e_i)$.

2. **Admission Control Operator**

   The admission control operator of a class-i customer is defined according to the following equation:

   $T_{A_i}v(x) = \max\{v(x + e_i) + R_i, v(x)\}$.

3. **Uncontrolled Departure from a Queue**

   The uncontrolled departure operator of a class-i customer is defined according to the following equation:

   $T_{UD_i}v(x) = v(x - e_i)^+$. where $(x - e_i)^+ = \max(0, x - e_i)$.

The additional queueing operators that represents the multidimensional servers are represented in below also:

4. **Movable Server Operator**

   The movable server operator models the queue where decision maker is allowed to decide on the customer class that will be served in the queue where $\mu(j) \leq 1.0$ is the rate of production of class-j.

$$T_{MS}v(x) = \max_{j \in I: x_j > 0} \mu(j)v(x - e_j) + (1 - \mu(j))v(x) \text{ if } \sum_{j \in I} x_j > 0, \quad (2.3)$$

$$T_{MS}v(x) = v(x) \text{otherwise.} \quad (2.4)$$

5. **Movable Tandem Server Operator**

   The movable tandem server operator models the tandem queue where the decision maker is allowed to decide on which job on the sequence that will be served in the tandem queue where 0 designates leaving the system and e_1 designates the job at the first server in the tandem sequence.

$$T_{MTS}v(x) = \max_{j \in I: x_j > 0} \mu(j)v(x - e_i + e_k) + (1 - \mu(j))v(x) \text{ if } \sum_{j \in I} x_j > 0, \quad (2.5)$$

$$T_{MTS}v(x) = v(x) \text{ otherwise.}$$

   where $\sum_{k=0}^{m} \mu(i,k) = 1$, $\mu(i,j) = 0$ for all i and $0 < j < i - 1$ and $e_0 = 0$, i.e. $k = 0$ means leaving the system. Hence, the movable tandem server models a serial queue where customers either leave the system or forwarded to a higher indexed queue in the system.

2.2.4 Mathematical Properties

The emphasis of this thesis is to provide the structural properties of optimal policies for robust counterparts of a set of problems from queueing and inventory theory. We use a set of mathematical properties throughout the thesis in order to show these properties. In this part, we explain some of these properties and describe how relevant these properties are to the thesis in the context of revenue optimization. Other properties that we do not present here will be explained in the relevant sections.

1. *Increasingness and Decreasingness*

   In the context of revenue management systems, increasingness/decreasingness of a function used to describe the value of an additional item in the inventory or one unit of time in the horizon to the value function x. If $f(t,x)$ is increasing (strictly increasing) both in t and x, this means that an additional one unit of inventory/one unit of time has always have a nonnegative (positive) value to the optimal expected revenue. This may seem quite intuitive, however in the presence of holding costs the optimal expected revenue is not increasing in the inventory status x.

2. *Concavity*

   A function $f(x)$ is said to be concave in x if it satisfies the following inequality.

$$f(\alpha x_1 + (1-\alpha)x_2) \geq \alpha f(x_1) + (1-\alpha)f(x_2) \text{ for all } x_1, x_2 \; and \; 0 \leq \alpha \leq 1.$$

   By putting $x-1$ instead of x_1, $x+1$ instead of x_2 and setting $alpha = 0.5$ we obtain:

$$f(x) - f(x-1) \geq f(x+1) - f(x) \text{ for all } x$$

   Now suppose that there is the following option, we have an inventory of x and there is a customer who is willing to pay a price of R for this product. If we sell the product, our inventory decreases by 1 unit and we will obtain a revenue of $f(x-1)+R$, and if we do not sell the product the inventory remains the same and we will obtain a revenue of $f(x)$. If it is optimal not to sell the product, this means that $f(x) \geq f(x-1)+R$ and therefore $f(x)-f(x-1) \geq R$. If the function $f(x)$ is concave in x, then $f(x-1)-f(x-2) \geq R$ and therefore it is not optimal to sell the product at $x-1$ also. This is known as an optimal threshold policy and concavity implies optimality of threshold policies in many inventory control problems. The optimal policy is structured such that the controller sells the product to the customer if the inventory status is higher than a certain level, i.e. threshold or protection level. If the inventory is below this threshold then the controller reserves the item for future in the expectation of customers who may pay a higher price than R for the item.

3. *k-concavity*

   $k - concavity$ is a mathematical property invented by Scarf [13] for the inventory problems where fixed cost of purchasing is not zero and the purchasing cost C is described as below where u is the order quantity, k is the fixed cost of purchasing and c is the unit cost of item:

$$C = k + uc \text{ if } u > 0$$
$$C = 0 \text{ otherwise.}$$

   Function $f(x)$ is said to be $k - concave$ in x if it satisfies the following inequality:

$$f(x) \geq -k + f(y) - (y - x)\frac{f(x) - f(x-b)}{b}. \tag{2.6}$$

   for all $y \geq x \geq x - b \geq 0$.

   In order to have a simplified representation we put 1 instead of b and $x + 1$ instead of y and the following inequality is obtained:

$$2f(x) \geq -k + f(x+1) + f(x-1).$$

   It is easy to show that a function that is concave, is also $k - concave$ for all values of $k > 0$. The $k - concavity$ property implies that the (s, S) policy is optimal, and if it is optimal to order at x also it is optimal to order at $x - 1$. Moreover if it is optimal to order $S - x$ items at x, then it is optimal to order $S - x + 1$ items at $x - 1$, i.e. the optimal policy is described by an order up to level S and a reorder point s. More formally the function for the optimal purchasing order quantity u is given by:

$$u = S - x \ \text{ if } x \leq s$$

$$u = 0 \ \text{ otherwise.}$$

The complete proof of the optimality of an (s, S) policy is given in Scarf [13], [29], however we provide a simple intuition here and we show that "**if it is optimal to purchase at x then it is optimal to purchase at $x - 1$**". Suppose the value function satisfies the $k - \textit{concavity}$ property given in the Equation 5.2. By taking y as S, and 1 as b, we obtain the following:

$$f(x) + (S - x)(f(x) - f(x - 1)) \geq -k + f(S).$$

This means that:

$$S - x \geq \frac{f(S) - k - f(x)}{f(x) - f(x - 1)}.$$

It is given that it is optimal to purchase at x and therefore $f(x) \leq f(S) - c(S - x) - k$ so that the controller prefers to raise the inventory to level S. Hence:

$$\frac{f(S) - k - f(x)}{c} \geq S - x$$

Please note that this is a revenue maximization problem and $f(S) - c(S - x) - k > 0$. We therefore conclude that $f(x - 1) \leq f(x) - c$, this means that if inventory status is $x - 1$ it would be better to raise the inventory to x by paying the unit price c if there were no holding cost. More specifically: $f(x - 1) \leq f(S) - c(S - x + 1) - k$ since $f(x - 1) \leq f(x) - c \leq f(S) - k - c(S - x) - c = f(S) - k - c(S - x + 1)$. Hence, if the optimal decision is to purchase at x, then it is also optimal to purchase at $x - 1$. This proves the optimality of reorder point s, and the optimality of order up to point S is similar and can be found in Scarf [13].

4. *Supermodularity/Submodularity*

We use the supermodularity and the submodularity properties throughout the thesis. A function $f(x)$ is called supermodular in (x_1, x_2), if it satisfies the following equation.

$$f(x_1 + 1, x_2) - f(x_1, x_2) \leq f(x_1 + 1, x_2 + 1) - f(x_1, x_2 + 1) \text{ for all } x_1, x_2.$$

Similar to our previous example for the concavity property, suppose that we have two types of products in the inventory and (x_1, x_2) denotes the inventory status of the product 1, product 2 respectively. Now suppose that our initial inventory is $(x_1 + 1, x_2)$ and we have an option to sell one unit of product 1 and obtain a revenue of $f(x_1, x_2) + R$ or do not sell the product and remain with the same inventory with a revenue of $f(x_1 + 1, x_2)$. It is clear that it is optimal not to sell one unit of product 1 if and only if $f(x_1 + 1, x_2) - f(x_1, x_2) \geq R$. This means that it is optimal not to sell this product also when inventory status is $f(x_1 + 1, x_2 + 1)$ since $f(x_1 + 1, x_2 + 1) - f(x_1, x_2 + 1) \geq R$ is easily implied by supermodularity. Hence, there is a threshold in terms of x_2 such that controller always prefers to keep one unit of inventory 1 at that inventory status. This is known as switching curve and also a symmetric argument, i.e. we sell Product 1 alone only when the inventory for Product 2 is lower than a certain level and sell Product 2 alone only when the inventory for Product 1 is lower than a certain level. Otherwise, the controller reserves the product for customers who are likely to pay more. We can also say that an additional one unit of inventory for product 1 has more value when there is an additional one unit of inventory for product 2, i.e. these products are more valuable when introduced as a package to customer. This is also known as complementarity in economical theory.

Submodularity and its interpretation is quite similar. Hence we only provide the submodularity equation here. A function $f(x)$ is called submodular in (x_1, x_2) if it satisfies the following equation.

$$f(x_1 + 1, x_2) - f(x_1, x_2) \geq f(x_1 + 1, x_2 + 1) - f(x_1, x_2 + 1) \text{ for all } x_1, x_2.$$

2.3 Previous Work

In this part, we present examples from literature considering robust versions of the MDPs. In the next two subsequent parts of this section, we summarize two well-known papers that use the maximin approach in order to introduce robustness to their revenue models. In the last part of this section we summarize the work of Çil, Karaesmen and Örmeci [8]. They studied how the changes in the transition parameters affect the value function and the structure of the optimal policies for a set of revenue and queuing theory problems. Although the main focus of this thesis is structural properties of robust dynamic programming problems, most of their work are useful for our work.

2.3.1 Robust Optimization of single-item Inventory Management Problem without Replenishment

Birbil, Frenk, Gromicho and Zhang [7] studied the robust version of a typical revenue management problem -single-leg airline revenue management problem- in their paper. Remember that the customers are distinguished by their rewards, and each class-i customer has a known reward R_i and an arrival probability p_i. The value function of the nominal problem is described below where the total number of customers is n:

$$v_t(x) = \sum_{i=1}^{n} p_i \max\{R_i + v_{t+1}(x-1), v_{t+1}(x)\} + (1 - \sum_{i=1}^{n})v_{t+1}(x).$$

The uncertainty model they use is a typical and useful one. The main elements of the uncertainty set are the prior distribution $\widehat{\boldsymbol{p}}$ and the deviation from that distribution $\boldsymbol{\Delta}$ where the total deviation is constrained by the value δ. Hence, the arrival probability of each class is $p_i = \widehat{p}_i + \Delta_i$ and $\sum_{i=1}^{n} p_i = 1$ by just considering the fictitious event (no arrival) as a special customer with no reward. The total deviation from the prior distribution $\widehat{\boldsymbol{p}}$ is constrained, and the representation of the uncertainty set is as follows:

$$\mathcal{P}_t = \left\{ \boldsymbol{y} = (y_1, \ldots, y_{n+1}) : \quad \sum_{i=1}^{n} \left(\frac{p_i - \widehat{p}_i}{\widehat{p}_i} \right)^2 \leq \delta^2 \right\}.$$

Apparently, the total deviation from the prior distribution $\sum_{i=1}^{n} \Delta_i = 0$ in order to preserve the constraint $\sum_{i=1}^{n} p_i = 1$. In typical situations, the higher classes show up when there is shorter time left in the horizon, whereas the economy class customers are more likely to arrive in the early phases of the horizon. The parameters of the uncertainty set are allowed to change between the stages in order to represent the behavior of each class in the time horizon. Therefore:

$$\sum_{i=1}^{n} \left(\frac{p_{i,t} - \widehat{p_{i,t}}}{\widetilde{p}_{i,t}} \right)^2 \leq \delta_t^2.$$

Birbil, Frenk, Gramicho and Zhang [7] used the minimax approach in order to introduce the uncertainty to their model. Hence, the robust value function $w_t(x)$ is given as:

$$w_t(x) = \min_{p_{i,t}} \left\{ \sum_{i=1}^{n} \Delta_i \{R_i + w_{t+1}(x-1) + w_{t+1}(x)\}^+ \right\} + w_{t+1}(x).$$

Birbil, Frenk, Gramicho and Zhang [7] obtain an analytical solution of the problem. They also explore the performance of their robust solution on real data.

2.3.2 Robust Dynamic Pricing and Queueing Problem

Lim and Shantikumar [23] studied a robust dynamic pricing problem. In this paper, they utilize the relative entropy approach in order to model the uncertainty among the transition probabilities. The dynamic pricing problem considers an arrival rate $\lambda(R)$ function that is modeled as a function of price R, i.e. $\lambda(R) = Ae^{BR}$ where A, B are parameters that are derived from historical data. Additional research that uses the relative entropy concept was carried out by Jain, Lim and Shantikumar [17] where they model a queueing problem in which the controller decides on arrival and departure rates. In both of these studies, they succeed to show that the structure of the optimal policies are the same with the nominal problem. In these papers, they do not utilize the event-based dynamic programming approach. In this thesis, we cover the same problems and do not restrict our models to a specific uncertainty set.

2.3.3 Effects of Parameters on Structure of Optimal Policies

Çil, Örmeci and Karaesmen [8] study the effects of parameters on the structure of optimal policies. They explore a wide set of problems and most of them are represented in Section 3.1. In the following, we summarize how optimal polices are affected by parameters according to their work. Please note that an increase in willingness to admit customers means a nonincrease in thresholds (protection levels) in a queueing system, similarly willingness to produce means a nondecrease in thresholds (basestocks) for all states and stages in an inventory system.

In a queueing system represented as a combination of operators introduced in section 3.1:

1. The willingness of controller to admit customers decreases by an increase in an arrival rate and increases by an increase in service capacity as well as waiting room capacity,
2. The willingness of controller to serve customers increases by an increase in an arrival rate and decreases by an increase in service capacity as well as waiting room capacity.

In an inventory system represented as a combination of operators introduced in section 3.1:

1. The willingness to satisfy demand decreases by an increase in demand rate and increases by an increase in finite storage capacity, and by an increase in production rate,
2. The willingness to produce increases by an increase in demand rate and an increase in finite storage capacity, whereas it decreases by an increase in production rate.

In this thesis, we also investigate the behavior of optimal policies with respect to changes in uncertainty sets. Unlike our earlier results, in order to provide such a comparison, additional constraints have to be imposed on the uncertainty set. In Chapter 4 we study the interval uncertainty model and investigate the behavior of optimal policy against perturbations in the uncertainty set.

Chapter 3

ROBUST CONTROL OF STOCHASTIC INVENTORY AND QUEUING PROBLEMS

3.1 Introduction

As already discussed in Chapter 2, the input data to the problem is not known with certainty and is approximated or estimated. This, if ignored, may cause significant suboptimality or infeasibility for the solution considered. Robust optimization is a specific methodology that addresses this problem and has received a lot of attention lately [4] and we employ the maximin approach illustrated in Chapter 2.

Our focus in this chapter is on a set of problems from inventory and queueing theory. The models considered include discrete-time versions of some well established cases such as service rate control and admission control problems of Lippman [24], the stock rationing problem of Ha [14] and the dynamic revenue management problem of Lautenbacher [22].

To address parameter uncertainty we formulate a robust stochastic dynamic program within a maximin approach also known as the absolute robust decision. The maximin approach defines a game between the controller (system manager) and Nature. For instance, in the context of demand admission control, the controller's aim is to maximize the expected profit by choosing the allowable actions (admitting a given class of demand or not), whereas Nature tries to minimize the expected profit by choosing the worst-possible parameters (arrival rates) and acts upon observing the controller choice. This formulation is known as the robust counterpart of the classical problem. The robust optimal policy designates the policy which yields the highest expected profit result after minimization by Nature.

The robust formulation of a Markov Decision Process with an uncertain transition probability distribution goes back to Satia and Lave [28] who proposed a solution by a policy iteration approach. More recently, Nilim and El Ghaoui [25] and Iyengar [16] simultaneously studied robust stochastic dynamic programs and established the existence of a robust Bellman recursion whose solution yields the robust value function and the corresponding optimal policy. In addition, both papers emphasized that under appropriate choice of uncertainty sets, the additional complexity brought by the robust formulation is reasonable if the standard formulation has a tractable solution.

The dynamic queueing and inventory control literature has a strong tradition in characterizing the structure of optimal policies. This is in part due to the computational efficiency of structured policies. However, the main motivation for looking for structured policies is that they are usually expressed in a few parameters and tend to be easy to understand and communicate. There are known effective techniques to investigate the structure of the solution of a stochastic dynamic program. Event-based dynamic programming proposed by Koole [18] and further extended in [19] streamlines this procedure for a class of queueing control problems. Recently, Çil, Örmeci and Karaesmen [8] employed event-based programming to explore structural properties in a class of production/inventory problems and proposed an extension to study the effects of perturbations of input parameters. In this chapter, we employ event-based dynamic programming to formulate a robust counterpart of the stock rationing problem and exploit the known properties of the standard problem.

The organization of this part is as follows; in section 3.2 we present the classical dynamic programming problem and in section 3.3.1 we present the robust counterpart of the dynamic programming problem. We provide our results in a general setting for a generic problem that can be constituted of the operators listed in Section 3.2. In section 3.4 we provide information on how the optimal policies respond to uncertainty sets. In the last section we conclude our results.

3.2 Structural Properties of Some Queueing and Inventory Control Problems

In this section, we introduce the class of queueing and inventory control problems addressed in this chapter and present a common a framework investigating their structural properties.

The models we consider are the discrete-time revenue maximization problems that involve demand or customer admissions and processing or replenishment decisions for queueing or inventory systems. The demanding customers and offered jobs are categorized into classes depending on the reward that they offer. Batch arrivals may also occur. The controller decides whether to accept the incoming customer or job based on the instant reward and the future expected revenue. Dynamic pricing is an alternative to admission control where the controller sets the optimum price at each stage based on the reservation price distribution of customers. There may also be processing decisions in the queuing and inventory systems that we consider. The production control decision may be given depending on the future sales and instant production costs. There may be more than one supplier/production unit with different purchasing costs/production costs. Our main focus in the latter sections of this chapter is on structural properties of the robust versions of the above problems. However, in order to show how structural results on the value function propagate to the robust counterpart we first present the standard versions of these systems within a common perspective.

The problems we consider in the scope of this chapter are represented in discrete time. In our setting, the stages (time) are denoted by $t = 0, 1, ..., T$, where T is the last stage in the horizon. We use the event-based representation introduced by Koole [18]. Let $i = 0, 1, 2, ..., n$ denote the event indices, where event 0 corresponds to a fictitious event. Note that a fictitious event occurs if the system observes no state change. The system state x can be any integer value at any stage t, then we have $x \in X$ where X is a subset of integers. We let a_i denote the controller action regarding event i, so that an action can be defined by $a_i \in \{0, 1\}$ as in the case admission/rejection or by $a_i \in \Re$ as in the case of pricing, where $\Re$ is the set of real numbers. Note that the controller is allowed to choose her/his actions independently for all (x, t) pairs, and there is no restriction among the actions regarding different events. The action vector $\boldsymbol{a} = (a_1, a_2, ..., a_n)$ denotes the actions for all possible events. At each stage, depending on the controller action, $R_i(a, x)$ $(C_i(a, x))$ is gained (incurred) as an immediate reward (cost). The randomness is characterized by a transition probability distribution at each stage which is assumed to be independent of prior uncertainties. The probability that event i occurs at stage t is given by $p_{i,t}$, so $\sum_{i=0}^{n} p_{i,t} = 1$

for all t. When event i occurs in state x at stage t, the conditional probability that the next state is y if controller selects action a is denoted by $q_t(a,x,y|i)$. Hence, $\sum_y q_t(a,x,y|i) = 1$ for all t, a and x. We express the optimal value function $v_t(x)$ according to the following equation when the time left until the end of the horizon is $T-t$:

$$v_t(x) = \max_{\boldsymbol{a}} \left\{ \sum_{i=0}^{n} p_{i,t} \sum_{y \in X} q_t(a,x,y|i)(v_{t+1}(y) + R_i(a,x)) \right\}. \tag{3.1}$$

Note that the action vector $\boldsymbol{a}$ is a function of x and t, but we suppress this for notational simplicity. The event-based approach allows us to define the value function $v_t(x)$ as a convex combination of operators $T^i v_{t+1}(x) = \max_{a\in A} \left\{ \sum_{y\in X} q_t(a,x,y|i)(v_{t+1}(y) + R_i(a,x)) \right\}$, $v_t(x)$ can be written as:

$$v_t(x) = \sum_{i=0}^{n} p_{i,t} \left\{ \max_{a} \sum_{y \in X} q_t(a,x,y|i)(v_{t+1}(y) + R_i(a,x)) \right\} = \sum_{i=0}^{n} p_{i,t}\, T_i v_{t+1}(x). \tag{3.2}$$

Next, we investigate three important properties of the value function. Non-increasingness (non-decreasingness) refers to $v_t(x) \leq (\geq) v_t(x-1)$ for all x, t. Supermodularity (submodularity) refers to $v_t(x) - v_t(x-1) \geq (\leq)\ v_{t+1}(x) - v_{t+1}(x-1)$ for all x, t. Finally, concavity (convexity) refers to : $v_t(x+1) - v_t(x) \leq (\leq) v_t(x) - v_t(x-1)$ for all x, t. Monotonicity and concavity properties have been studied in detail previously ([19], [8]). Here, we restate the known results in a different perspective: we assume that all the operators defined in Table 3.1 preserve concavity, which is shown to be true by [19] and [8]. Then, we deduce all other properties preserved by the operators from concavity. This approach will facilitate our analysis of robust MDPs in the next section. Note that concavity and monotonicity properties of the value functions determine the structure of the optimal policies as well, e.g., optimal threshold policies for admission control and optimal base stock policies for inventory control. Moreover, we elaborate on supermodularity/submodularity of the value functions in x and t, which has not been studied in this perspective to our knowledge. These properties ensure that the parameters of the structured optimal policies are also monotone in time, e.g., the optimal base stock levels and optimal thresholds are monotone in t.

In order to adress supermodularity, we first define $B_i v_{t+1}(x)$ as the marginal time benefit (MTB) function, so that we have:

$$T_i v_{t+1}(x) = B_i v_{t+1}(x) + v_{t+1}(x). \tag{3.3}$$

Note that this is conceptually similar to the *marginal benefit of the operator* discussed in Çil et al. [8]. The MTB function is not only a tool to represent the value function $v_t(x)$, but it has a direct relation with supermodularity/submodularity properties in the context of these systems. The NI property of an MTB function, $B_i v_{t+1}(x)$, implies that submodularity is preserved by the corresponding operator, $T_i v_{t+1}(x)$, whereas the ND property of an MTB function, $B_i v_{t+1}(x)$, implies that supermodularity is preserved by that operator. The properties of MTB's for different operators are presented in the next Proposition 4.2.1. We first define the operators in below:

Type	Operator Name	Notation	Definition
I	Rationing	$T_{R_i}v(x)$	$\max\{R_i + v(x-1), v(x)\}$
I	Batch Rationing	$T_{BR_i}v(x)$	$\max_{\kappa_i \in \min(x,B)}\{\kappa_i R_i + v(x-\kappa_i\}$
I	Production Rate	$T_{PR_i}v(x)$	$\max_{\Pi\in[0,1]}\{-C_{\Pi_i} + \Pi_i v(x+1) + (1-\Pi_i)v(x)\}$
I	Production	$T_{P_i}v(x)$	$\max\{v(x+1) - C_i, v(x)\}$
I	Inventory Pricing	$T_{IP}v(x)$	$\max_R\{\overline{F}_Z(R)[v(x-1)+R] + F_Z(R)v(x)\}$
Q	Admission	$T_{A_i}v(x)$	$\max\{R_i + v(x+1), v(x)\}$
Q	Batch Admission	$T_{BA_i}v(x)$	$\max_{\kappa_i \in \min(x,B)}\{\kappa_i R_i + v(x+\kappa_i\}$
Q	Controlled Departure	$T_{CD_i}v(x)$	$\max\{v(x-1) - C_i, v(x)\}$
Q	Departure Rate	$T_{DR_i}v(x)$	$\max_{\Pi\in[0,1]}\{-C_{\Pi} + \Pi v(x-1) + (1-\Pi)v(x)\}$
Q	Queue Pricing	$T_{QP}v(x)$	$\max_R\{\overline{F}_Z(R)[v(x+1)+R] + F_Z(R)v(x)\}$
Q	Uncontrolled Arrival	$T_{UA}v(x)$	$a(x)v(x+1) + (1-a(x))v(x)$
Q	Uncontrolled Departure	$T_{UD}v(x)$	$b(x)v(x-1) + (1-b(x))v(x)$

Table 3.1: Operators and Their Definitions (Q denotes a queueing operator and I denotes an inventory operator)

Proposition 3.2.1 *Let a value function $v_t(x)$ be concave in x for all x and t, then:*

1. *The MTB functions of the below operators are nonincreasing (NI) in x for all x and t.*

   (a) *The queueing operators, admission control $B_{A_i}v_t(x)$, batch admission $B_{BA_i}v_t(x)$, queue pricing $B_{QP_i}v_t(x)$ and uncontrolled arrival $B_{UA_i}v_t(x)$,*

   (b) *The inventory operators, production control $B_{P_i}v_t(x)$ and production rate control $B_{PR_i}v_t(x)$.*

2. *The MTB functions of the below operators are nondecreasing (ND) in x for all x at t.*

   (a) *The queueing operators, controlled departure $B_{CD_i}v_t(x)$, departure rate $B_{DR_i}v_t(x)$, uncontrolled departure $B_{UD_i}v_t(x)$,*

   (b) *The inventory operators, rationing $B_{R_i}v_t(x)$, batch rationing $B_{BR_i}v_t(x)$ and inventory pricing $B_{IP_i}v_t(x)$.*

Proof: The proofs are given in the Section 3.2.1.

Finally, using the event-based dynamic programming framework, we summarize the results when operators are combined to constitute a particular model.

Theorem 3.2.1 *Let a value function $v_t(x)$ consist of convex combinations of the operators preserving concavity in x for all x and t. Then:*

1. *The value function $v_t(x)$ is NI (ND) and concave in x for all x, t for the queueing (inventory) systems.*

2. *Furthermore, assume that the value function $v_t(x)$ consists of convex combinations of the operators such that their MTB functions, $B_i v_{t+1}(x)$, are all NI (all ND). Then, $v_t(x)$ is supermodular (submodular) in x, t for all x, t.*

Proof The proof of Part 1 is omitted, since it borrows the main ideas from Koole [18] and is straightforward. Here, we only provide the proof of part (2) when $B_i v_{t+1}(x)$ is NI, since the other case is similar. The value function, $v_t(x)$, can be written as follows by (3.3):

$$v_t(x) = \sum_{i=0}^{n} p_{i,t} B_i v_{t+1}(x) + v_{t+1}(x),$$

since $\sum_{i=0}^{n} p_{i,t} = 1$ for all t. Assuming that MTB functions, $B_i v_{t+1}(x)$, are NI in x for all x and t, we have:

$$v_t(x) - v_{t+1}(x) = \sum_{i=0}^{n} p_{i,t} B_i v_{t+1}(x) \geq \sum_{i=0}^{n} p_{i,t} B_i v_{t+1}(x-1) = v_t(x-1) - v_{t+1}(x-1). \quad (3.4)$$

Equation 3.4 completes the proof. □

Theorem 3.2.1 ensures the submodularity (supermodularity) property for a general class of models under the limitation that all the operators of a given model have NI (ND) MTB functions. It applies, for example, for single-resource capacity control problems in revenue management but does not hold in a queueing control model which has both admission control and uncontrolled departure operators.

In order to complete our arguments given in this part, we show the individual results of the operators in below section.

3.2.1 Operators and Properties

We introduce a number of commonly-used operators for queueing and inventory problems as in Koole [18], Koole [19] and Çil et al. [8]. Table 3.1 presents the type ("I" for inventory and "Q" for queueing), names, notations and mathematical formulations of these operators. Remember that their descriptions are already given in Chapter 2. The structural properties of them are summarized in Table 3.2. Please remember that we used the previous result of Koole [18], Koole [19] and Çil et al. [8] on concavity of all of these operators in x for all t. We set $v_N(x) = 0$ for all values of x, however, this assumption can be replaced with any other condition as far as it is convenient to the statement.

Inventory Control Operators

1. Batch Rationing Operator

   Definition of the Operator

   $T_{BR_i} v(x) = \max_{\kappa_i \in \min(x,B)} \{\kappa_i R_i + v(x - \kappa_i)\}$,

   $T_{BR_i} v(x) = \max_{\kappa_i \in \min(x,B)} \{v(x - \kappa_i) - v(x) + \kappa_i R_i\} + v(x)$

Marginal Time Benefit

$B_{BR_i}v(x) = \max_{\kappa_i \in \min(x,B)}\{v(x-\kappa_i) - v(x) + \kappa_i R_i\}$ is ND in x $\forall$ t.

Proof:

Suppose a_x, a_{x-1} is the optimal number of customers admitted in states x and $x-1$ respectively. As it is stated in Çil, Örmeci and Karaesmen [8] concavity of $v(x)$ implies that customers admitted to x can be either same as with $x-1$ or it can be only differ by 1. Hence, a_x is either a_{x-1} or $a_{x-1}+1$. For $a_{x-1} = 0$, the claim is obvious. Now suppose $a_{x-1} \geq 1$. Concavity of $v(x)$ implies $v(x-1-a) - v(x-1) \leq v(x-a) - v(x)$

Therefore:

$$v(x-1-a_{x-1}) - v(x-1) \leq v(x-a_{x-1}) - v(x)$$
$$v(x-1-a_{x-1}) - v(x-1) + a_{x-1}R_i \leq v(x-a_{x-1}) - v(x) + a_{x-1}R_i$$
$$v(x-a_{x-1}) - v(x) + a_{x-1}R_i \leq v(x-a_x) - v(x) + a_x R_i.$$

The second line proves the first case while the third line proves the second case by optimality. □

Monotonicity of Value Function

$T_{BR_i}v(x)$ is ND in x for all t

Proof:

$v_N(x) = 0$. By optimality,

$$v(x-a_x) + a_x R_i \geq v(x-a_{x-1}) + a_{x-1}R_i.$$

Suppose $v(x) \geq v(x-1)$:

$$v(x-a_{x-1}) + a_{x-1}R_i \geq v(x-1-a_{x-1}) + a_{x-1}R_i.$$

This completes proof. □

2. Rationing Operator

Definition of the Operator

$T_{R_i}v(x) = \max\{R_i + v(x-1), v(x)\}$,

$T_{R_i}v(x) = \{v(x-1) - v(x) + R_i\}^+ + v(x)$

Marginal Time Benefit

$B_{R_i}v(x) = \{v(x-1) - v(x) + R_i\}^+$ is ND in x for all t.

Proof:

$$v(x-1) - v(x) \leq v(x) - v(x+1)$$
$$v(x-1) - v(x) + R_i \leq v(x) - v(x+1) + R_i$$
$$\{v(x-1) - v(x) + R_i\}^+ \leq \{v(x) - v(x+1) + R_i\}^+.$$

First line holds by concavity, second line is obvious. For the third line, the inequality is obvious if both sides of the inequalities are either greater than zero, or less than zero. If $v(x-1) - v(x) + R_i \leq 0$ and $v(x) - v(x+1) + R_i \geq 0$ the inequality also holds. This completes the proof. □

Monotonicity of Value Function

$T_{R_i}v(x)$ is ND in x for all t.

Proof:

$v_N(x) = 0$. Suppose $v(x) \geq v(x-1)$ for all $x \geq 1$. By concavity property, there are only three alternatives: 1. Do not admit at x, do not admit at $x-1$. 2. Admit at x, do not admit at $x-1$. 3. Admit at x, admit at $x-1$.

$T_{R_i}v(x) = \{v(x-1) - v(x) + R_i\}^+ + v(x)$. Therefore, since $v(x) \geq v(x-1)$, in case 1, the monotonicity holds trivially. $v(x-1) + R_i \geq v(x-1)$, so in case 2, the monotonicity again holds. $v(x-1) + R_i \geq v(x-2) + R_i$, so in case 3, the monotonicity holds.

This completes our proof. □

3. Production Rate Operator

Definition of the Operator

$T_{PR_i}v(x) = \max_{\Pi \in [0,1]}\{-C_{\Pi_i} + \Pi_i v(x+1) + (1-\Pi_i)v(x)\}$,

$T_{PR_i}v(x) = \max_{\Pi_i \in [0,1]}\{\Pi_i\{v(x+1) - v(x)\} - C_{\Pi_i}\} + v(x)$

Marginal Time Benefit

$B_{PR_i}v(x) = \max_{\Pi_i \in [0,1]}\{\Pi_i\{v(x+1) - v(x)\} - C_{\Pi_i}\}$ is NI in x for all t.

Proof:

Suppose Π_{i_x} and $\Pi_{i_{x-1}}$ are the optimal production levels for x and $x-1$ for t.

$$v(x+1) - v(x) \leq v(x) - v(x-1)$$

$$\Pi_{i_x}[v(x+1) - v(x)] \leq \Pi_{i_x}[v(x) - v(x-1)]$$

$$\Pi_{i_x}[v(x+1) - v(x)] - C_{\Pi_{i_x}} \leq \Pi_{i_x}[v(x) - v(x-1)] - C_{\Pi_{i_x}}.$$

Since $\Pi_{i_{x-1}}$ is optimal for $x-1$ at t:

$\Pi_{i_x}[v(x+1) - v(x)] - C_{\Pi_{i_x}} \leq \Pi_{i_{x-1}}[v(x) - v(x-1)] - C_{\Pi_{i_{x-1}}}$, this proves our result.□

Monotonicity of Value Function

$T_{PR_i}v(x)$ is ND in x for all t.

Proof:

Remember that $v_N(x) = 0$. By optimality:

$$\Pi_{i_x}\{v(x+1) - v(x)\} - C_{\Pi_{i_x}} + v(x) \geq \Pi_{i_{x-1}}\{v(x+1) - v(x)\} - C_{\Pi_{i_{x-1}}} + v(x).$$

Suppose $v(x) \geq v(x-1)$, then following must be true:

$$\Pi_{i_{x-1}}\{v(x+1) - v(x)\} - C_{\Pi_{i_{x-1}}} + v(x) \geq$$
$$\Pi_{i_{x-1}}\{v(x) - v(x-1)\} - C_{\Pi_{i_{x-1}}} + v(x-1)$$

by rearranging the terms:

$$(1 - \Pi_{i_{x-1}})v(x) + \Pi_{i_{x-1}}v(x+1) \geq (1 - \Pi_{i_{x-1}})v(x-1) + \Pi_{i_{x-1}}v(x),$$

this completes proof. □

4. Production Operator

Definition of the Operator

$T_{P_i}v(x) = \max\{v(x+1) - C_i, v(x)\}$,

$T_{P_i}v(x) = \{v(x+1) - v(x) - C_i\}^+ + v(x)$

for all t.

Marginal Time Benefit

$B_{P_i}v(x) = \{v(x+1) - v(x) - C_i\}^+$ is NI in x for all t.

Proof:

$$v(x+1) - v(x) \leq v(x) - v(x-1)$$
$$v(x+1) - v(x) - C_i \leq v(x) - v(x-1) - C_i$$
$$\{v(x+1) - v(x) - C_i\}^+ \leq \{v(x) - v(x-1) - C_i\}^+.$$

The first line holds by concavity of value function in x. The second line is obvious. If the both side of the inequality have the same sign, then the monotonicity property holds. If $v(x+1) - v(x) - C_i \leq 0$ and $v(x) - v(x-1) - C_i > ge0$ the monotonicity holds as well. □

Monotonicity of Value Function

$T_{P_i}v(x)$ is ND in x for all t.

Proof:

$v_N(x) = 0$. Suppose $v(x) \geq v(x-1)$ By concavity property, there are only three alternatives:

1. Do not produce at x, do not produce at $x-1$.
2. Do not produce at x,produce at $x-1$.
3. Produce at x, produce at $x-1$.

$T_{P_i}v(x) = \{v(x+1) - v(x) - C_i\}^+ + v(x)$.

Since $v(x) \geq v(x-1)$, in case 1, the monotonicity holds trivially. $v(x) \geq v(x) - C_i$, so in case 2, the monotonicity also holds. $v(x+1) - C_i \geq v(x) - C_i$, so in case 3, the monotonicity also holds.This completes our proof. □

5. Inventory Pricing Operator

Definition of the Operator

$T_{IP}v(x) = \max_R\{\overline{F}_Z(R)[v(x-1)+R] + F_Z(R)v(x)\}$,

$T_{IP}v(x) = max_R\overline{F}_Z(R)\{v(x-1)-v(x)+R\}+v(x)$

Marginal Time Benefit

$B_{IP}v(x) = \max_R \overline{F}_Z(R)\{v(x-1)-v(x)+R\}$ is ND in x for all t.

Proof: Suppose R_x and R_{x-1} are the optimal prices for x and $x-1$ for t.

$$v(x-1)-v(x) \leq v(x)-v(x+1)$$
$$\overline{F}_Z(R_x)\{v(x-1)-v(x)+R_x\} \leq \overline{F}_Z(R_x)\{v(x)-v(x+1)+R_x\}.$$

Since R_{x+1} is optimal for $x+1$ at t:

$\overline{F}_Z(R_x)\{v(x)-v(x+1)+R_x\} \leq \overline{F}_Z(R_{x+1})\{v(x)-v(x+1)+R_{x+1}\}$ this proves our result. □

Monotonicity of Value Function

$T_{IP}v(x)$ is ND in x for all t.

Proof:

Please remember that $v_N(x)=0$, $v(x) \geq v(x-1)$ and R_x and R_{x-1} are the optimal prices for x and $x-1$ for t. By optimality:

$$\overline{F}_Z(R_x)\{v(x-1)-v(x)+R_x\}+v(x) \geq$$
$$\overline{F}_Z(R_{x-1})\{v(x-1)-v(x)+R_{x-1}\}+v(x)$$

The following holds by concavity and ND property at $t+1$:

$$\overline{F}_Z(R_{x-1})\{v(x-1)-v(x)+R_{x-1}\}+v(x) \geq$$
$$\overline{F}_Z(R_{x-1})\{v(x-2)-v(x-1)+R_{x-1}\}+v(x-1)$$

This completes the proof. □

Queueing Operators

Queueing operators we consider throughout this dissertation are given in the following. It is important to note that the waiting room is infinite.

1. Batch Admission

   Definition of the Operator

   $T_{BA_i}v(x) = \max_{\kappa_i \in \min(x,B)}\{\kappa_i R_i + v(x+\kappa_i)\}$.

   $T_{BA_i}v(x) = \max_{\kappa_i \in \min(x,B)}\{v(x+\kappa_i) - v(x) + \kappa_i R_i\} + v(x)$

   Marginal Time Benefit

   $B_{BA_i}v(x) = \max_{\kappa_i \in \min(x,B)}\{v(x+\kappa_i) - v(x) + \kappa_i R_i\}$ is NI in x for all t.

   Proof: Suppose a_x, a_{x-1} is the number of customers admitted to states x and $x-1$ respectively. As it is stated in Çil, Örmeci and Karaesmen [8] concavity of $v(x)$ implies that customers admitted to x can be either same as with $x-1$ or it can be only differ by 1. Hence, a_{x-1} is either a_x or $a_x + 1$. For $a_x = 0$, claim is obvious. Now suppose $a_x \geq 1$. The concavity of $v(x)$ implies $v_t(x-1+a) - v_t(x-1) \geq v_t(x+a) - v_t(x)$:

   $$v(x+a_x) - v(x) \leq v(x-1+a_x) - v(x-1)$$
   $$v(x+a_x) - v(x) + a_x R_i \leq v(x-1+a_x) - v(x-1) + a_x R_i$$
   $$v(x-1+a_x) - v(x-1) + a_x R_i \leq v(x-1+a_{x-1}) - v(x-1) + a_{x-1} R_i.$$

   The second line proves the first case while the third line proves the second case by optimality. □

   Monotonicity of Value Function $T_{BA_i}v(x)$ is NI in x for all t

   Proof:

   $v_N(x) = 0$ so the inequality holds for the last stage. By optimality:

   $$\{v(x-1+a_x) - v(x-1) + a_x R_i\} + v(x-1) \leq$$
   $$\{v(x-1+a_{x-1}) - v(x-1) + a_{x-1} R_i\} + v(x-1)$$

   Suppose $v(x) \leq v(x-1)$:

$$\{v(x + a_x) + a_x R_i\} \leq \{v(x - 1 + a_x) + a_x R_i\} +$$

This completes proof. □

2. Admission

Definition of the Operator

$T_{A_i} v(x) = \max\{R_i + v(x+1), v(x)\}$,

$T_{A_i} v(x) = \{v(x+1) - v(x) + R_i\}^+ + v(x)$.

Marginal Time Benefit

$B_{A_i} v(x) = \{v(x+1) - v(x) + R_i\}^+$ is NI in x for all t.

Proof:

$$v(x+1) - v(x) \leq v(x) - v(x-1)$$
$$v(x+1) - v(x) + R_i \leq v(x) - v(x-1) + R_i$$
$$\{v(x+1) - v(x) + R_i\}^+ \leq \{v(x) - v(x-1) + R_i\}^+.$$

The first line holds by concavity of $v(x)$ in x and the second line is obvious. For the third line, if both sides of the inequality has the same sign, then the monotonicity result holds. If $v(x+1) - v(x) + R_i \leq 0$ whereas $v(x) - v(x-1) + R_i \geq 0$, the monotonicity result holds as well. This completes the proof. □ ***Monotonicity of Value Function***

$T_{A_i} v(x)$ is NI in x for all t.

Proof:

$v_N(x) = 0$. Suppose $v(x) \leq v(x-1)$ By concavity property, there are only three alternatives: 1. Do not admit at x, do not admit at $x-1$. 2. Do not admit at x, admit at $x-1$. 3.Admit at x, admit at $x-1$.

$T_{A_i} v(x) = \{v(x+1) - v(x) + R_i\}^+ + v(x)$. Therefore, Since $v(x) \leq v(x-1)$, in case 1, the monotonicity holds trivially. $v(x) \leq v(x) + R_i$, so in case 2 the monotonicity holds. $v(x+1) + R_i \leq v(x) + R_i$, so in case 3 holds, the monotonicity also holds. This completes our proof. □

3. Controlled Departure Operator

Definition of the Operator

$T_{CD_i}v(x) = \max\{v(x-1) - C_i, v(x)\}$,

$T_{CD_i}v(x) = \{v(x-1) - v(x) - C_i\}^+ + v(x)$

Marginal Time Benefit

$B_{CD}v(x) = \{v(x-1) - v(x) - C_i\}^+$ is ND in x for all t.

Proof:

$$v(x-1) - v(x) \leq v(x) - v(x+1)$$
$$v(x-1) - v(x) - C_i \leq v(x) - v(x+1) - C_i$$
$$\{v(x-1) - v(x) - C_i\}^+ \leq \{v(x) - v(x+1) - C_i\}^+.$$

The first line holds by concavity of $v(x)$ in x and the second line is obvious. For the third line, if both sides of the inequality has the same sign, then the monotonicity result holds. If $v(x-1) - v(x) - C_i \leq 0$ whereas $v(x) - v(x+1) - C_i \geq 0$, the monotonicity result holds as well. This completes the proof. □

Monotonicity of Value Function

$T_{CD_i}v(x)$ is NI in x for all t.

Proof:

$v_N(x) = 0$. Suppose $v(x) \leq v(x-1)$ By concavity property there are only three alternatives. 1. Do not depart at x, do not depart at $x-1$. 2. Depart at x, do not depart at $x-1$. 3.Depart at x, depart at $x-1$.

$T_{CD_i}v(x) = \{v(x-1) - v(x) - C_i\}^+ + v(x)$. Therefore, Since $v(x) \leq v(x-1)$, in case 1, the monotonicity holds trivially. $v(x-1) - C_i \leq v(x-1)$, so in case 2, the monotonicity holds. $v(x-1) - C_i \leq v(x-2) - C_i$, so in case 3, the monotonicity also holds.

This completes our proof. □

4. Departure Rate Operator

Definition of the Operator

$T_{DR_i}v(x) = \max_{\Pi\in[0,1]}\{-C_\Pi + \Pi v(x-1) + (1-\Pi)v(x)\}$,
$T_{DR_i}v(x) = \max_{\Pi\in[0,1]}\{\Pi\{v(x-1) - v(x)\} - C_\Pi\} + v(x)$

Marginal Time Benefit

$B_{DR_i}v(x) = \max_{\Pi\in[0,1]}\{\Pi\{v(x-1) - v(x)\} - C_\Pi\}$ is ND in x for all t.

Proof:

Suppose Π_{i_x} and $\Pi_{i_{x+1}}$ are optimal results for x and $x+1$ for t.

$$v(x-1) - v(x) \leq v(x) - v(x+1)$$

$$\Pi_{i_x}[v(x-1) - v(x)] \leq \Pi_{i_x}[v(x) - v(x+1)]$$

$$\Pi_{i_x}[v(x-1) - v(x)] - C_{\Pi_{i_x}} \leq \Pi_{i_x}[v(x) - v(x+1)] - C_{\Pi_{i_x}}.$$

Since $\Pi_{i_{x+1}}$ is optimal for $x+1$ at t:

$\Pi_{i_x}[v(x) - v(x+1)] - C_{\Pi_{i_x}} \leq \Pi_{i_{x+1}}[v(x) - v(x+1)] - C_{\Pi_{i_{x+1}}}$ this proves our result.□

Monotonicity of Value Function

$T_{DR_i}v(x)$ is NI in x for all t.

Proof:

Remember that $v_N(x) = 0$. By optimality:

$$\Pi_{i_x}[v(x-2) - v(x-1)] - C_{\Pi_{i_x}} + v(x-1) \leq$$

$$\Pi_{i_{x-1}}[v(x-2) - v(x-1)] - C_{\Pi_{i_{x-1}}} + v(x-1)$$

Suppose $v(x) \leq v(x-1)$ The following must be true:

$$\Pi_{i_x}[v(x-1) - v(x)] - C_{\Pi_{i_x}} + v(x) \leq$$

$$\Pi_{i_x}[v(x-2) - v(x-1)] - C_{\Pi_{i_x}} + v(x-1)$$

By rearranging the terms:

$$\Pi_{i_x}v(x-1) + (1-\Pi_{i_x})v(x) \leq \Pi_{i_x}v(x-2) + (1-\Pi_{i_x})v(x-1),$$

this completes the proof. □

5. Queue Pricing Operator

Definition of the Operator

$T_{QP}v(x) = \max_R\{\overline{F}_Z(R)[v(x+1)+R] + F_Z(R)v(x)\}$,

$T_{QP}v(x) = \max_R \overline{F}_Z(R)\{v(x+1) - v(x) + R\} + v(x)$

Marginal Time Benefit

$B_{QP}v(x) = \max_R \overline{F}_Z(R)\{v(x+1) - v(x) + R\}$ is NI in x for all t.

Proof: Suppose R_x and R_{x-1} are optimal results for x and $x-1$ for t.

$$v(x+1) - v(x) \leq v(x) - v(x-1)$$

$$\overline{F}_Z(R_x)\{v(x+1) - v(x) + R_x\} \leq \overline{F}_Z(R_x)\{v(x) - v(x-1) + R_x\}.$$

Since R_{x-1} is optimal for $x-1$ at t:

$\overline{F}_Z(R_x)\{v(x) - v(x-1) + R_x\} \leq \overline{F}_Z(R_{x-1})\{v(x) - v(x-1) + R_{x-1}\}$ this proves our result. □

Monotonicity of Value Function

$T_{QP}v(x)$ is NI in x for all t.

Proof:

Remember $v_N(x) = 0$ and $v(x) \leq v(x-1)$. By optimality:

$$\overline{F}_Z(R_x)\{v(x) - v(x-1) + R_x\} + v(x-1) \leq$$
$$\overline{F}_Z(R_{x-1})\{v(x) - v(x-1) + R_{x-1}\} + v(x-1).$$

The following is true by concavity and NI at $t+1$:

$$\overline{F}_Z(R_x)\{v(x+1) - v(x) + R_x\} + v(x) \leq$$
$$\overline{F}_Z(R_x)\{v(x) - v(x-1) + R_x\} + v(x-1).$$

This completes the proof. □

6. Uncontrolled Arrival to a Queue

Definition of the Operator

$T_{UA}v(x) = a(x)v(x+1) + (1 - a(x))v(x),$

$T_{UA}v(x) = a(x)\{v(x+1) - v(x)\} + v(x)$

The Uncontrolled arrival operator preserves concavity in x for all t.

Marginal Time Benefit

$B_{UA}v(x) = a(x)\{v(x+1) - v(x)\}$ is NI in x $\forall x$, where $a(x)$ is NI in x for all t. *Proof:*

$$a(x)\{v(x+1) - v(x)\} \leq a(x)\{v(x) - v(x-1)\}$$

$$a(x)\{v(x) - v(x-1)\} \leq a(x-1)\{v(x) - v(x-1)\}.$$

First line holds by concavity and the second line holds since $a(x) \leq a(x-1)$. This completes the proof. □

Monotonicity of Value Function

$T_{UA}v(x)$ is NI in x for all t.

Proof:

Remember $v_N(x) = 0$. Suppose $v(x) \leq v(x-1)$ By concavity property proof follows automatically. □

7. Uncontrolled Departure to a Queue

Definition of the Operator

$T_{UD}v(x) = b(x)v(x-1) + (1 - b(x))v(x),$

$T_{UD}v(x) = b(x)\{v(x-1) - v(x)\} + v(x)$

Marginal Time Benefit

$B_{UD}v(x) = b(x)\{v(x-1) - v(x)\}$ is ND in x $\forall x$, where $b(x)$ is ND in x for all t. *Proof:*

$$b(x)\{v(x-1) - v(x)\} \leq b(x)\{v(x) - v(x+1)\}$$

$$b(x)\{v(x-1) - v(x)\} \leq b(x+1)\{v(x) - v(x+1)\}.$$

First line holds by concavity and the second line holds since $b(x) \leq b(x+1)$. This completes the proof. □

Monotonicity of Value Function

$T_{UD}v(x)$ is NI in x for all t.

Proof:

Remember that $v_N(x) = 0$. Suppose $v(x) \leq v(x-1)$. Supermodularity implies $b(x-1) \leq b(x)$. Hence, proof follows automatically. An alternative proof is given in Çil, Örmeci and Karaesmen [8].

The properties of the operators we present here are summarized in Table 3.2:

Table 3.2: Monotonicity Results

Operator Name	Supermodularity in (x,t)	$Bv_t(x)$	$v_t(x)$
Rationing	supermodular	ND. in x	ND. in x
Batch Rationing	supermodular	ND. in x	ND. in x
Production	submodular	NI. in x	ND. in x
Production Rate	submodular	NI. in x	ND. in x
Inventory Pricing	supermodular	ND. in x	ND. in x
Admission	submodular	NI. in x	NI. in x
Batch Admission	submodular	NI. in x	NI. in x
Controlled Departure	supermodular	ND. in x	NI. in x
Departure Rate	supermodular	ND. in x	NI. in x
Queue Pricing	submodular	NI. in x	NI. in x
Uncontrolled Arrival to a Queue	submodular	NI. in x	NI. in x
Uncontrolled Departure from a Queue	supermodular	ND. in x	NI. in x

3.3 Structural Properties of Some Robust Queueing and Inventory Problems

In this section, we consider the robust version of the dynamic optimization problem (which will be referred to as the nominal problem) described in Section 2. In this version of the problem, a subset of the problem parameters is assumed to be uncertain. Typically, the system controller decides on his actions before observing the uncertain parameters. Once his decisions are taken, Nature selects these parameters from an uncertainty set in order to minimize the benefit of the system. Hence, the controller has to consider the worst case scenario when choosing his actions. Our robust formulations are based on the maximin approach suggested by Nilim and El Ghaoui [25] and Iyengar [16]. We present the formulation and the underlying assumptions in Section 3.3.1, and investigate the structural results of the optimal policy under these assumptions in Section 3.3.2.

3.3.1 The Robust Dynamic Programming Formulation

In the robust formulation, the transition probabilities or the rewards (costs) belong to an uncertainty set, rather than being fixed values as in the nominal problem. In general, an uncertainty set may vary with action, state and stage. In this chapter, we mainly focus on having uncertain transition probabilities, while considering uncertain rewards (costs) in extensions (Section 6).

In our formulation, the transition probabilities consist of two components as observed in Equation (3.1): $q_t(a, x, y|i)$ and $p_{i,t}$. We assume that $q_t(a, x, y|i)$ is known with certainty and $p_{i,t}$ is uncertain and can depend on the chosen action and state of the system. This may be a limitation for certain models but is appropriate for queueing control represented by event-based dynamic programming. In such models, $q_t(a, x, y|i)$ is either 0 or 1 and the probabilistic structure is only reflected through the event probability $p_{i,t}$.

We let $p_{i,t}(x, \boldsymbol{a})$ be the probability of observing event i when the system is in state x and action vector $\boldsymbol{a}$ is chosen at stage t. We assume that $p_{i,t}(x, \boldsymbol{a})$ belongs to an uncertainty set $\mathcal{P}_t$, where $\mathcal{P}_t$ represents the available information on the event probability distribution. Our uncertainty model is based on the model proposed by Nilim and El Ghaoui [25], where the so-called rectangularity property is the main condition for obtaining a recursive solution. When the rectangularity property holds, nature can independently select its action for every stage, state and the controller action. We consider a special case of Nilim and El Ghaoui [25] by assuming that the uncertainty set at each stage t, $\mathcal{P}_t$, is independent of the controller's action vector $\boldsymbol{a}$ as well as the state x. Then, this assumption together with the rectangularity property implies that Nature's choice of a particular distribution at time t does not limit the choices of nature in the future. Note that this additional assumption is not very restrictive when handling real life problems: The state-dependent event probability distribution is difficult to estimate from limited statistical data. In addition, in typical examples, these probabilities represent demand rates which do not depend on the state of the system or the actions taken.

Letting $\pi = (a_1, a_2, \ldots, a_T)$ denote a policy, we define the set of all admissible policies of Nature for a given policy π
as:

$$\tau^{\pi} = \mathcal{P}_1 \times \mathcal{P}_2 \times \mathcal{P}_3 \times ... \times \mathcal{P}_T,$$

regardless of the particular states and actions.

Now we let $w_t(x)$ be the robust counterpart of the value function given in (3.2), and consider the following robust DP equation:

$$w_t(x) = \max_{\boldsymbol{a}} \left\{ \min_{\boldsymbol{p}_t(x,\boldsymbol{a})} \sum_{i=0}^{n} p_{i,t}(x, \boldsymbol{a}) \sum_{y \in X} q_t(a, x, y|i)(w_{t+1}(y) + R_i(a, x)) \right\}, \tag{3.5}$$

where $\boldsymbol{p}_t(x, \boldsymbol{a}) = (p_{1,t}(x, \boldsymbol{a}), p_{2,t}(x, \boldsymbol{a}), ..., p_{n,t}(x, \boldsymbol{a}))$. Note that, as in the previous section, we suppress the dependence of the action vector $\boldsymbol{a}$ on x and t for notational simplicity. This problem can be solved by recursively due to the above assumptions, as shown by Nilim and El Ghaoui [25] and Iyengar [16].

3.3.2 Properties of the Value Function and the Optimal Policy

To establish certain properties of the robust value function, we start by exploring the effects of the decision sequence in the game between the controller and Nature. In particular, we prove that the optimal action is independent of the uncertainty choice at any stage, which implies the perfect duality of the game. These results are then used to show that the robust formulation inherits the structural properties of the nominal problem.

In the robust formulation given by (3.5), Nature chooses the event probabilities to minimize the value function $w_t(x)$ after observing the action of the controller. The value function $\widetilde{w}_t(x)$ is defined as the dual counterpart of $w_t(x)$, and refers to the game when the controller chooses his action after he observes the event probabilities chosen by Nature. We let $\boldsymbol{p}_t(x) = (p_{1,t}(x), p_{2,t}(x), ..., p_{n,t}(x))$ be the event probabilities chosen by Nature and $\boldsymbol{a}_t(x, \boldsymbol{p}_t(x))$ be the corresponding action vector chosen by the controller in state x at stage t. Then, the value function, $\widetilde{w}_t(x)$, is given as follows:

$$\widetilde{w}_t(x) = \min_{\boldsymbol{p}_t(x)} \left\{ \max_{\boldsymbol{a}} \sum_{i=0}^{n} p_{i,t}(x) \sum_{y \in X} q_t(a, x, y|i)(\widetilde{w}_{t+1}(y) + R_i(a, x)) \right\}, \tag{3.6}$$

where $\boldsymbol{a} = \boldsymbol{a}_t(x, \boldsymbol{p}_t(x))$. The value function, $\widetilde{w}_t(x)$, can be expressed in terms of the operators, T_i, as shown by the next lemma:

Lemma 3.3.1 *The optimal action vector, $\boldsymbol{a}_t(x, \boldsymbol{p}_t)$, does not depend on the event probabilities, $\boldsymbol{p}_t(x) \in \mathcal{P}_t$, for the value function, $\widetilde{w}_t(x)$, given in (3.6). Therefore, the value function, $\widetilde{w}_t(x)$, can be expressed as follows:*

$$\widetilde{w}_t(x) = \min_{\boldsymbol{p}_t(x)} \left\{ \sum_{i=0}^{n} p_{i,t}(x) T_i \widetilde{w}_{t+1}(x)) \right\}.$$

Proof In equation (3.6), the inside maximization corresponds to the actions of the controller given the event probabilities, $\boldsymbol{p}_t(x) = (p_{1,t}(x), p_{2,t}(x), ..., p_{n,t}(x))$. Then, it can be written as a convex combination of the operators, $T_i w_{t+1}(x) = \max_a \sum_{y \in X} q_t(a, x, y|i)(\widetilde{w}_{t+1}(y) + R_i(a, x))$. Therefore, similarly to the derivation of equation (3.2), we have:

$$\begin{aligned} \widetilde{w}_t(x) &= \min_{\boldsymbol{p}_t(x)} \left\{ \sum_{i=0}^{n} p_{i,t}(x) \max_a \sum_{y \in X} q_t(a, x, y|i)(\widetilde{w}_{t+1}(y) + R_i(a, x)) \right\} \\ &= \min_{\boldsymbol{p}_t(x)} \left\{ \sum_{i=0}^{n} p_{i,t}(x) T_i \widetilde{w}_{t+1}(x)) \right\}. \end{aligned}$$

This completes the proof. □ Now we present the well-known property of the maximin theorem, which relates the robust value function, $w_t(x)$, with its dual counterpart, $\widetilde{w}_t(x)$:

Property 3.3.1 *The value function, $w_t(x)$, is less than or equal to its dual counterpart, $\widetilde{w}_t(x)$, i.e., $w_t(x) \leq \widetilde{w}_t(x)$ for all states x and stages t.*

When a game is perfectly dual, then the two value functions are equal to each other, i.e., $w_t(x) = \widetilde{w}_t(x)$ for all states x and stages t. Perfect duality has been shown for multistage stochastic games by various authors (Nilim and El Ghaoui [25], Nowak [26], and Altman [1]). However, we restate the perfect duality in Theorem 3.3.1 in our terminology.

Theorem 3.3.1 *The game between the controller and Nature is perfectly dual so that* $w_t(x) = \widetilde{w}_t(x)$ *for all states* x *and stages* t*. Hence, the robust value function* $w_t(x)$ *can be expressed as:*

$$w_t(x) = \min_{\boldsymbol{p}_t(x)} \left\{ \sum_{i=0}^{n} p_{i,t}(x) T_i w_t(x) \right\}, \tag{3.7}$$

or

$$w_t(x) = \min_{\boldsymbol{p}_t(x)} \left\{ \sum_{i=0}^{n} p_{i,t}(x) B_i w_t(x) \right\} + w_{t+1}(x), \tag{3.8}$$

where the probability vector $\boldsymbol{p}_t(x) = (p_{1,t}(x), p_{2,t}(x), ..., p_{n,t}(x))$ *is the Nature's corresponding optimal action.*

Proof We first consider the dual counterpart of the game so that the controller decides on his actions after observing the choice of Nature, $\boldsymbol{p}_t(x)$. Let $\boldsymbol{a}_t^*(x) = (a_{1,t}^*(x), \ldots, a_{n,t}^*(x))$ be the optimal action vector in state x at stage t for the value function $\widetilde{w}_t(x)$. Let's denote the Nature's respective solution $\boldsymbol{p}_t^*(x)$ according to the following equation:

$$\boldsymbol{p}_t^*(x) = \arg\min_{\boldsymbol{p}_t(x)} (\sum_{i=0}^{n} p_{i,t}(x) T_i \widetilde{w}_{t+1}(x)). \tag{3.9}$$

Now we feed the optimal action vector $\boldsymbol{a}_t^*(x)$ to the game in which the controller decides on his actions before observing the choice of Nature. Apparently, Nature's respective solution is $\boldsymbol{p}_t^*(x)$ according to Equation 3.5 implying that ($\boldsymbol{a}_t^*(x)$,$\boldsymbol{p}_t^*(x)$) is a feasible pair for $w_t(x)$. Hence, by Property 3.3.1, perfect duality holds, i.e. $w_t(x) = \widetilde{w}_t(x)$ for all states x and stages t. Then, equation (3.7) holds due to Lemma 3.3.1, and equation (3.8) holds by equations (3.7), (3.3) and $\sum_{i=0}^{n} p_{i,t} = 1$ for all t. □

By Theorem 3.3.1, we have established that the robust counterpart can be represented similarly to the nominal problem. This enables us to define the problem as in (3.7) or (3.8). In fact, Lemma 3.3.1 and Theorem 3.3.1 provide a much stronger result than perfect duality. By reformulating Equation (3.5) in this way, we establish a sort of independence of the controller's action from the Nature's posteriori action. Instead of solving the inner problem of nature for every controller action, we solve it only for one action (i.e., the optimal action of the dual counterpart) of the controller at each stage and state, which improves the solution time dramatically as the controller's action set gets larger. Hence, our contribution is important especially for the problems with uncertainty sets that are not proven to be solvable in polynomial time (see Nilim and El Ghaoui [25]). Now we will show that the structural properties of the value function $v_t(x)$ propagate to the robust counterpart for these situations. We start by considering the concavity:

Lemma 3.3.2 *Let the robust value function $w_t(x)$ be concave in x for all x and t, then $T_i w_t(x)$ is concave in x for all x and t. Table 3.1.*

Proof This result follows from [19] and [8], since concavity is preserved by the considered operators without any additional condition. □

As in the nominal problem, concavity guarantees the monotonicity of the value functions:

Lemma 3.3.3 *Let the robust value function $w_t(x)$ be concave in x for all x at t, then the queueing operators preserve NI property in x and the inventory operators preserve ND property in x for all x and t in the robust formulation.*

Proof The proofs are given in section 3.2.1. □

Our next result shows that concavity induces the same monotonicity properties for the MTB functions as in the nominal problem:

Lemma 3.3.4 *Let the robust value function $w_t(x)$ be concave in x for all x and t, then:*

1. *The MTB functions of the below operators are nonincreasing (NI) in x for all x and t.*

   (a) *The queueing operators, admission control $B_{A_i} w_t(x)$, batch admission $B_{BA_i} w_t(x)$, queue pricing $B_{QP_i} w_t(x)$ and uncontrolled arrival $B_{UA_i} w_t(x)$,*

   (b) *The inventory operators, production control $B_{P_i}w_t(x)$ and production rate control $B_{PR_i}w_t(x)$.*

2. *The MTB functions of the stated operators are nondecreasing (ND) in x for all x and t.*

   (a) *The queueing operators, controlled departure $B_{CD_i}w_t(x)$, departure rate $B_{DR_i}w_t(x)$, uncontrolled departure $B_{UD_i}w_t(x)$,*

   (b) *The inventory operators, rationing $B_{R_i}w_t(x)$, batch rationing $B_{BR_i}w_t(x)$ and inventory pricing $B_{IP_i}w_t(x)$.*

Proof The proofs are given in section 3.2.1. □

Finally, we present our main result, which extends Theorem 3.2.1 to the robust counterpart of these problems. More explicitly, for any problem that can be expressed in the form of Equation (2), all the structural results given in Theorem 3.2.1 extend to the value function of the robust counterpart, $w_t(x)$:

Theorem 3.3.2 *Let a value function of the robust counterpart, $w_t(x)$, consist of convex combinations of the operators preserving concavity in x for all x and t. Then:*

1. *The value function $w_t(x)$ is NI (ND) and concave in x for all x, t for queueing (inventory) systems.*

2. *If a value function $w_t(x)$ consists of convex combinations of operators such that their MTB functions, $B_iw_{t+1}(x)$, are all NI (all ND), then $w_t(x)$ is supermodular (submodular) in x, t for all x, t.*

Proof We let $p_t(x-1)$, $p_t(x)$, $p_t(x+1)$ denote the optimal choices of Nature in states $x-1$, x and $x+1$, respectively, at stage t, in the rest of the proof.
We first show concavity: Since $w_{t+1}(x)$ is concave in x for all x at stage $t+1$, then $T_iw_{t+1}(x)$ is concave in x for every operator i by Lemma 3.3.2, so that the convex combination of these operators also preserve this property. Hence, we can write the following inequality:

$$\sum_i p_{i,t}(x)T_iw_{t+1}(x-1) + \sum_i p_{i,t}(x)T_iw_{t+1}(x+1) \leq 2\left[\sum_i p_{i,t}(x)T_iw_{t+1}(x)\right]. \quad (3.10)$$

We know that:

$$\sum_i p_{i,t}(x-1)T_i w_{t+1}(x-1) \leq \sum_i p_{i,t}(x)T_i w_{t+1}(x-1),$$

by the optimality of $p_{i,t}(x-1)$ in state $x-1$, and:

$$\sum_i p_{i,t}(x+1)T_i w_{t+1}(x+1) \leq \sum_i p_{i,t}(x)T_i w_{t+1}(x+1),$$

by the optimality of $p_{i,t}(x+1)$ in state $x+1$. Then, the sum of the left-hand-sides (LHSs) of these two inequalities is less than the LHS of inequality (3.10), which proves that $w_t(x)$ is concave in x at stage t.

Now we prove the value function is NI for queueing operators. The result for the inventory operators can be proven similarly. We suppose that $w_{t+1}(x)$ is NI in x. Then $T_i w_{t+1}(x)$ is NI in x by Lemma 3.3.3. Hence, we have:

$$\sum_i p_{i,t}(x)T_i w_{t+1}(x) \leq \sum_i p_{i,t}(x-1)T_i w_{t+1}(x) \leq \sum_i p_{i,t}(x-1)T_i w_{t+1}(x-1),$$

where the first inequality is due to the optimality of $p_{i,t}(x)$ in state x, and the second inequality follows since $T_i w_{t+1}(x)$ is NI in x. This completes the proof of Part 1.
We prove part (2) only for the case when MTB functions of all the operators are NI, since the proof of the other case is similar. By equation (3.8) in Theorem 3.3.1, we have:

$$w_t(x) - w_{t+1}(x) = \sum_i p_{i,t}(x)B_i w_{t+1}(x).$$

On the other hand:

$$\sum_i p_{i,t}(x)B_i w_{t+1}(x) \geq \sum_i p_{i,t}(x)B_i w_{t+1}(x-1) \geq \sum_i p_{i,t}(x-1)B_i w_{t+1}(x-1),$$

where the first inequality is due to the optimality of $p_{i,t}(x)$ in state x, and the second inequality follows since $B_i w_{t+1}(x)$ is NI in x. Therefore:

$$w_t(x) - w_{t+1}(x) = \sum_i p_{i,t}(x) B_i w_{t+1}(x) ge \sum_i p_{i,t}(x-1) B_i w_{t+1}(x-1) = w_t(x-1) - w_{t+1}(x-1).$$

This completes the proof. □

Theorem 3.3.2 together with Lemmas 3.3.3 and 3.3.4 establishes a general result: The value function $w_t(x)$ has the same monotonicity properties with the nominal value function $v_t(x)$, independently of how the uncertainty set is constructed. Note that the nominal policy is a special case of robust policy where uncertainty set is defined as a single point. As a result, we show that optimal policies have the same structure for any system which is built by the operators listed in Table 3.1 of Section 3.2. In the next section, we discuss the effect of the degree of uncertainty on the structure of the value function and the optimal policies. Please note that it is not, a queueing system is generally constituted of operators whose MTB functions are both NI and ND.

3.4 Effect of Uncertainty on the Structure of the Optimal Policy

In this section we present some results on the effect of transition probability uncertainty on the robust value function $w_t(x)$ and the threshold $\Delta w_t(x)$. In order to analyze the effect of uncertainty, we compare two uncertainty sets $\boldsymbol{\mathcal{P}}$ and $\boldsymbol{\mathcal{P}}^{\varepsilon}$ where the former is the subset of the latter, i.e. $\boldsymbol{\mathcal{P}} \subseteq \boldsymbol{\mathcal{P}}^{\varepsilon}$.

Lemma 3.4.1 *Let $\boldsymbol{\mathcal{P}}$ and $\boldsymbol{\mathcal{P}}^{\varepsilon}$ be two uncertainty sets and $w(x)$ and $w^{\varepsilon}(x)$ be the corresponding robust value functions constituted of a subset of operators given in the Table 3.1. If $\mathcal{P} \subseteq \mathcal{P}^{\varepsilon}$ then $w(x) \geq w^{\varepsilon}(x)$.*

Proof Assume that induction hypothesis holds. Suppose we have two systems where all system parameters are the same for all stages except from the stage t. The uncertainty set defining event probabilities at stage t is $\mathcal{P}_t \subseteq \mathcal{P}_t^{\varepsilon}$. Now suppose that $\boldsymbol{p}_t$ and $\boldsymbol{p}_t^{\varepsilon}$ are the respective solutions of Nature associated with the uncertainty sets $\boldsymbol{\mathcal{P}}$ and $\boldsymbol{\mathcal{P}^{\varepsilon}}$. Then we have:

$$\sum_i p_{i,t}^{\varepsilon}(x) T_i w_{t+1}^{\varepsilon}(x) \leq \sum_i p_{i,t}(x) T_i w_{t+1}^{\varepsilon}(x) \leq \sum_i p_{i,t}(x) T_i w_{t+1}(x). \tag{3.11}$$

The first inequality follows since $\boldsymbol{p}_t$ is a feasible solution for both systems and the second is due to the induction hypothesis. Hence, we have $w_t(x) \geq w_t^{\varepsilon}(x)$ for all x at stage t. Now consider the production rate operator $T_{PR_i} w_t(x)$ and suppose that $\Pi_{i,t}$ and $\Pi_{i,t}{}^{\varepsilon}$ are the respective optimal solutions at stage t:

$$\begin{aligned} -C_{\Pi_i} + \Pi_i w_t(x+1) + (1-\Pi_i) w_t(x)\} &\geq \\ -C_{\Pi_i^{\varepsilon}} + \Pi_i^{\varepsilon} w_t(x+1) + (1-\Pi_i^{\varepsilon}) w_t(x)\} &\geq \\ -C_{\Pi_i^{\varepsilon}} + \Pi_i^{\varepsilon} w_t^{\varepsilon}(x+1) + (1-\Pi_i^{\varepsilon}) w_t^{\varepsilon}(x). \end{aligned}$$

We do not provide proofs for all of the operators since they are similar. Equation 3.11 implies $w_{t-1}(x) \geq w_{t-1}^{\varepsilon}(x)$. This completes the proof.

□

Before concluding the theoretical results, it is important to mention that the thresholds are not necessarily monotone with respect to uncertainty sets. Hence, neither $\Delta w_t^{\boldsymbol{\mathcal{P}}}(x) \geq \Delta w_t^{\boldsymbol{\mathcal{P}}^{\varepsilon}}(x)$ nor $\Delta w_t^{\boldsymbol{\mathcal{P}}}(x) \leq \Delta w_t^{\boldsymbol{\mathcal{P}}^{\varepsilon}}(x)$ holds for all x and t. To present a counter example, suppose that, without loss of generality, $\Delta w_t^{\boldsymbol{\mathcal{P}}}(x) \geq \Delta w_t^{\boldsymbol{\mathcal{P}}^{\varepsilon}}(x)$ holds, then $w_t^{\boldsymbol{\mathcal{P}}}(x) - w_t^{\boldsymbol{\mathcal{P}}^{\varepsilon}}(x) \geq w_t^{\boldsymbol{\mathcal{P}}}(x-1) - w_t^{\boldsymbol{\mathcal{P}}^{\varepsilon}}(x-1)$. This means that, as we enlarge the uncertainty set, the loss in $w_t(x)$ is greater than the corresponding loss in $w_t(x-1)$. Further suppose that $\boldsymbol{\mathcal{P}}$ contains all improving feasible directions for $w_t(x-1)$ but improving feasible directions for $w_t(x)$. Hence,

as we enlarge the uncertainty set by this way, the loss in value function $w_t^{\mathcal{P}}(x) - w_t^{\mathcal{P}^\varepsilon}(x) = 0$ whereas $w_t^{\mathcal{P}}(x-1) - w_t^{\mathcal{P}^\varepsilon}(x-1) \geq 0$. In the following we present a numerical counter example where we compare two systems. The reward of each class is as follows: $R_1 = 40$, $R_2 = 35$, $R_3 = 30$, $R_4 = 20$ and they have the same values for both systems.
The uncertainty set defining probabilities are denoted as $\mathcal{P}$ and defined as follows:

$$4p_1 + 2p_2 + 3p_3 + p_4 \geq 1.65$$
$$0.1 \leq p_i \leq 0.2 \; i = 1, ..., 4. \tag{3.12}$$

The calculations for states 1 and 2 for stages $T-1, T-2, T-3, T-4, T-5$ are presented in the following for the first system:
When stage $t = T-1$ and the state $x = 1$ the computed values are as follows:

i	1	2	3	4
$B_i w_{T-1}(1)$	40	35	30	20
$B_i w_{T-1}(1)/b_i$	10	17.5	10	20
$p_{i,T-1}(1)$	0.1875	0.1	0.2	0.1
$b_i p_{i,T-1}(1)$	0.75	0.2	0.6	0.1

$w_{T-1}(1) = 19$ and $\boldsymbol{p}_{T-1}(1) = (0.1875, 0.1, 0.2, 0.1)$. When stage $t = T-1$ and the state $x = 2$ the computed values are as follows:

i	1	2	3	4
$B_i w_{T-1}(2)$	40	35	30	20
$B_i w_{T-1}(2)/b_i$	10	17.5	10	20
$p_{i,T-1}(2)$	0.1875	0.1	0.2	0.1
$b_i p_{i,T-1}(2)$	0.75	0.2	0.6	0.1

The $w_{T-1}(x) = 19$ for all $x > 0$ and $\boldsymbol{p}_{T-1}(x) = (0.1875, 0.1, 0.2, 0.1)$. When stage $t = T-2$ and the state $x = 1$ the computed values are as follows:

i	1	2	3	4
$B_i w_{T-2}(1)$	21	16	11	1
$B_i w_{T-2}(1)/b_i$	5.25	8	3.666667	1
$p_{i,T-2}(1)$	0.1625	0.1	0.2	0.2
$b_i p_{i,T-2}(1)$	0.65	0.2	0.6	0.2

$w_{T-2}(1) = 26.415$ and $\boldsymbol{p}_{T-2}(1) = (0.1625, 0.1, 0.2, 0.2)$. When stage $t = T-2$ and the state $x = 2$ the computed values are as follows:

i	1	2	3	4
$B_i w_{T-2}(2)$	40	35	30	20
$B_i w_{T-2}(2)/b_i$	10	17.5	10	20
$p_{i,T-2}(2)$	0.1875	0.1	0.2	0.1
$b_i p_{i,T-2}(2)$	0.75	0.2	0.6	0.1

$w_{T-2}(2) = 38$ and $\boldsymbol{p}_{T-2}(2) = (0.1875, 0.1, 0.2, 0.1)$. When stage $t = T-3$ and the state $x = 1$ the computed values are as follows:

i	1	2	3	4
$B_i w_{T-3}(1)$	13.5875	8.5875	3.5875	0
$B_i w_{T-3}(1)/b_i$	3.396875	4.29375	1.195833	0
$p_{i,T-3}(1)$	0.1625	0.1	0.2	0.2
$b_i p_{i,T-3}(1)$	0.65	0.2	0.6	0.2

$w_{T-3}(1) = 30.19672$ and $\boldsymbol{p}_{T-3}(1) = (0.1625, 0.1, 0.2, 0.2)$. When stage $t = T-3$ and the state $x = 2$ the computed values are as follows:

i	1	2	3	4
$B_i w_{T-3}(2)$	28.4125	23.4125	18.4125	8.4125
$B_i w_{T-3}(2)/b_i$	7.103125	11.70625	6.1375	8.4125
$p_{i,T-3}(2)$	0.1875	0.1	0.2	0.1
$b_i p_{i,T-3}(2)$	0.75	0.2	0.6	0.1

$w_{T-3}(2) = 50.19234$ and $\boldsymbol{p}_{T-3}(2) = (0.1875, 0.1, 0.2, 0.1)$. When stage $t = T-4$ and the state $x = 1$ the computed values are as follows:

i	1	2	3	4
$B_i w_{T-4}(1)$	9.803281	4.803281	0	0
$B_i w_{T-4}(1)/b_i$	2.45082	2.401641	0	0
$p_{i,T-4}(1)$	0.1125	0.2	0.2	0.2
$b_i p_{i,T-4}(1)$	0.45	0.4	0.6	0.2

$w_{T-4}(1) = 32.26024$ and $\boldsymbol{p}_{T-4}(1) = (0.1125, 0.2, 0.2, 0.2)$. When stage $t = T-4$ and the state $x = 2$ the computed values are as follows:

i	1	2	3	4
$B_i w_{T-4}(2)$	20.00438	15.00438	10.00438	0.004375
$B_i w_{T-4}(2)/b_i$	5.001094	7.502188	3.334792	0.004375
$p_{i,T-4}(2)$	0.1625	0.1	0.2	0.2
$b_i p_{i,T-4}(2)$	0.65	0.2	0.6	0.2

$w_{T-4}(2) = 56.94524$ and $\boldsymbol{p}_{T-4}(2) = (0.1625, 0.1, 0.2, 0.2)$. When stage $t = T-5$ and the state $x = 1$ the computed values are as follows::

i	1	2	3	4
$B_i w_{T-5}(1)$	7.739756	2.739756	0	0
$B_i w_{T-5}(1)/b_i$	1.93	1.37	0.00	0.00
$p_{i,T-5}(1)$	0.1125	0.2	0.2	0.2
$b_i p_{i,T-5}(1)$	0.45	0.4	0.6	0.2

$w_{T-5}(1) = 33.67892$ and $\boldsymbol{p}_{T-5}(1) = (0.1125, 0.2, 0.2, 0.2)$. When stage $t = T-5$ and the state $x = 2$ the computed values are as follows:

i	1	2	3	4
$B_i w_{T-5}(2)$	15.315	10.315	5.315002	0
$B_i w_{T-5}(2)/b_i$	3.83	5.16	1.77	0.00
$p_{i,T-5}(2)$	0.1625	0.1	0.2	0.2
$b_i p_{i,T-5}(2)$	0.65	0.2	0.6	0.2

$w_{T-5}(2) = 61.52843$ and $\boldsymbol{p}_{T-5}(2) = (0.1625, 0.1, 0.2, 0.2)$. Now consider the second system. The uncertainty set $\mathcal{P}'$ of that system is larger than the uncertainty set of the first system $\mathcal{P}' \supseteq \mathcal{P}$. The definition of $\mathcal{P}'$ is as follows:

$$p_1 + 2p_2 + 3p_3 + p_4 \geq 1.65 \text{ if } t \neq T-5$$
$$0.1 \leq p_{i,t} \leq 0.1 \; i = 1, ..., 4 \text{ if } t = T-5$$
$$0.05 \leq p_{i,t} \leq 0.325 \; i = 1, ..., 4$$

. If we denote the value function and probabilities of the second system with $w'_t(x)$ and $p'_{i,t}(x)$, it is apparent that $w'_t(x) = w_t(x)$ and $p'_t(x) = p_t(x)$ for $t = T-1, ..., T-4$. Hence, we just show the values for $T-5$ When stage $t = T-5$ and the state $x = 1$ the computed values are as follows:

i	1	2	3	4
$B'_i w_{T-5}(1)$	7.739756	2.739756	0	0
$B'_i w_{T-5}(1)/b_i$	1.93	1.37	0.00	0.00
$p'_{i,T-5}(1)$	0.05	0.325	0.2	0.2
$b_i p'_{i,T-5}(1)$	0.2	0.65	0.6	0.2

$w'_{T-5}(1) = 33.53765$ and $\boldsymbol{p}'_{T-5}(1) = (0.05, 0.325, 0.2, 0.2)$. When stage $t = T-5$ and the state $x = 2$ the computed values are as follows:

i	1	2	3	4
$B'_i w_{T-5}(2)$	15.315	10.315	5.315002	0
$B'_i w_{T-5}(2)/b_i$	3.83	5.16	1.77	0.00
$p'_{i,T-5}(2)$	0.1875	0.05	0.2	0.2
$b_i p'_{i,T-5}(2)$	0.75	0.1	0.6	0.2

$w'_{T-5}(2) = 61.39556$ and $\boldsymbol{p}'_{T-5}(2) = (0.1875, 0.05, 0.2, 0.2)$.

The calculated values are as follows where the greater values are shown with boldface letters:

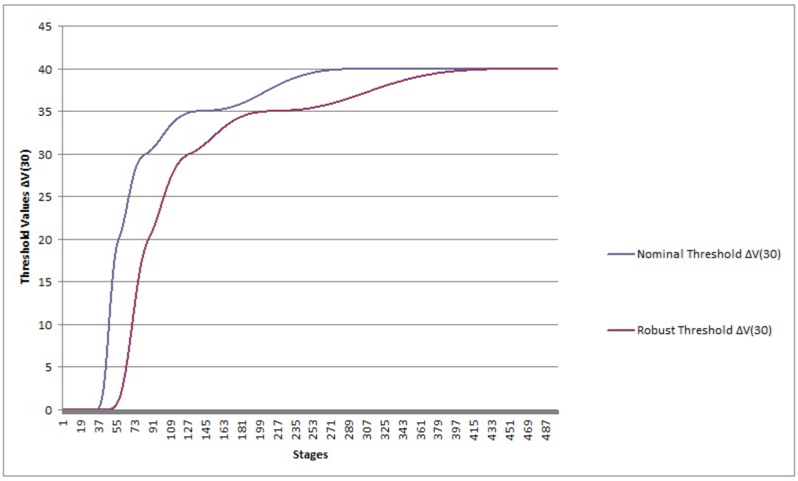

Figure 3.1: Behavior of Thresholds wrt Uncertainty Size at Various Stages

	$w_{T-5}(x)$	$w'_{T-5}(x)$	$\Delta w_{T-5}(x)$	$\Delta w'_{T-5}(x)$
X=1	**33.67892**	33.53765	**33.67892**	33.53765
X=2	**61.52843**	61.39556	27.84951	**27.8579**

This is a counterintuitive argument stating that larger the uncertainty set does not necessitate more pessimistic decisions taken by the controller unless certain conditions are imposed on the structure of the uncertainty set. Further suppose that, we add a new class with a reward $R = 27.85$ to both of these systems whose arrival probability is zero everywhere except from the $T-6$th stage. Although, the uncertainty set of second system is larger, the controller of this system behaves more optimistically than the controller of the first system at this stage when inventory level is 2 and rejects this customer in the anticipation of a higher paying customer. However, the controller of the first system accepts this class at this stage and state.

Finally, we note that $\Delta w_t^{\mathcal{P}}(x) - \Delta w_t^{\mathcal{P}^{\varepsilon}}(x)$ is not necessarily monotone in t. In Figure 3.1, optimal $\Delta w_t(x)$ for state $x = 30$ and various stages associated to nominal and robust solution of a rationing problem. However, Figure 3.1 represents the tendency of the system against uncertainty set.

3.4.1 Different Robust Approaches

In addition to the maximin approach it is possible to show the properties of value function of the robust counterpart for different approaches. We present here the approach based on bicriteria optimization suggested by Xu and Mannor [34] that is less conservative and more responsive to the uncertainty set. In their work, they propose a nested structure where $\mathcal{P}^\lambda \subseteq \mathcal{P}$. Here any distribution within the uncertainty set belong to $\mathcal{P}^\lambda$ with probability of λ whereas it belongs to $\mathcal{P}$ with probability of 1. They call a robust policy S-robust if it satisfies the following condition when only event probabilities are considered:

$$w_t(x) = \lambda \max_{\boldsymbol{a}} \left\{ \min_{\boldsymbol{p}_t(x,\boldsymbol{a}) \in \mathcal{P}^\lambda} \sum_{i=0}^{n} p_{i,t}(x,\boldsymbol{a}) \sum_{y \in X} q_t(a,x,y|i)(w_{t+1}(y) + R_i(a,x)) \right\}$$
$$+(1-\lambda) \max_{\boldsymbol{a}} \left\{ \min_{\boldsymbol{p}_t(x,\boldsymbol{a}) \in \mathcal{P}} \sum_{i=0}^{n} p_{i,t}(x,\boldsymbol{a}) \sum_{y \in X} q_t(a,x,y|i)(w_{t+1}(y) + R_i(a,x)) \right\},$$

Since the problem is in recursive form, Theorem 3.3.1 also applies for the value function of the S-robust counterpart. Hence, the S-robust counterparts of the classical problem have the same structural properties with them and the S-robust value function $w_t(x)$ can be represented as in the following equation:

$$w_t(x) = \lambda \min_{\boldsymbol{p}_t^\lambda(x) \in \mathcal{P}_t^\lambda} \left\{ \sum_i p_{i,t}^\lambda(x) T_i w_{t+1}(x) \right\} + (1-\lambda) \min_{\boldsymbol{p}_t(x) \in \mathcal{P}_t} \left\{ \sum_i p_{i,t}(x) T_i w_{t+1}(x) \right\}.$$

or equivalently:

$$w_t(x) = \lambda \min_{\boldsymbol{p}_t^\lambda(x) \in \mathcal{P}_t^\lambda} \left\{ \sum_i p_{i,t}^\lambda(x) B_i w_{t+1}(x) \right\} + (1-\lambda) \min_{\boldsymbol{p}_t(x) \in \mathcal{P}_t} \left\{ \sum_i p_{i,t}(x) B_i w_{t+1}(x) \right\} + w_{t+1}(x).$$

It is clear that Theorem 3.3.2 also applies for this equation and the value function obtained in this way is a convex combination of two particular value functions having the same properties.

3.5 Illustrations from the Literature

In this section, we present some conclusions on the structure of optimal policies for robust versions of some well-known examples from the literature.

The first example is the single- resource capacity control from revenue management system ([22]) that is modeled by the inventory rationing operator:

$$v_t(x) = \sum_{i=1}^{n} p_{i,t} T_{BR_i} v_{t+1}(x) + p_{0,t} v_{t+1}(x) \quad \text{for } x > 0$$

where x denotes the number of available inventory (seats) and $p_{i,t}$ is the probability of an arrival of customer class-i with corresponding reward R_i.

By Theorem 3, in the robust version of this problem the value function is concave and optimal admission policies are of threshold type. In addition, by part 2 of the theorem the thresholds are monotone over time.

Let us now consider an extended version of this capacity control problem that comprises dynamic pricing. In this case, rationing operators and dynamic pricing operators are used together in order to model a special customer segment (class $n+1$) that is offered a spot price). A typical value function is then given by:

$$v_t(x) = \sum_{i=1}^{n} p_{i,t} T_{BR_i} v_{t+1}(x) + p_{n+1,t} T_{IP} v_{t+1}(x) + p_{0,t} v_{t+1}(x) \quad \text{for } x > 0$$

Once again, using Theorem 3 we conclude that for the corresponding robust problem threshold policies are optimal for admission control and optimal prices are non-increasing in x and in t.

Apart from the discrete time models, our results apply for continuous-time models under certain assumptions. In particular, a class of continuous-time problems can be converted to equivalent discrete time problems using uniformization (Lippman [24]). If the uncertainty in the model can be represented after the conversion in discrete time (i.e. if the transition rates of the continuous-time model are uncertain), the results continue to apply. As an example, let us consider the uniformized version of a typical admission control problem to a single-server Markovian queue (Koole [18]):

$$v_t(x) = -h(x) + p_{1,t}T_A v_{t+1}(x) + p_{2,t}T_{UD} v_{t+1}(x) + p_{0,t} v_{t+1}(x) \text{ for } x > 0,$$

where x denotes the number of customers in the system, $p_{1,t}$ is the probability of an arrival, $p_{2,t}$ is the probability of a service completion and $h(x)$ is a non-decreasing and convex holding cost function.

From Theorem 3, the optimal admission control policy of the robust version of the above problem is a threshold policy but optimal thresholds are not necessarily monotone over time.

Finally, let us consider a dynamic pricing and production control problem for a make-to-stock queue with lost sales ([14]). After uniformization, the value function is expressed as:

$$v_t(x) = -h(x) + p_{1,t}T_{IP} v_{t+1}(x) + p_{2,t}T_P v_{t+1}(x) + p_{0,t} v_{t+1}(x) \text{ for } x > 0,$$

where x denotes the available inventory, $p_{1,t}$ is the probability of a demand arrival, $p_{2,t}$ is the probability of production completion and $h(x)$ is an non-decreasing and convex holding cost function.

Using Theorem 3, we observe that the optimal production policy is determined by a threshold and that the optimal prices are non-decreasing in x even for the robust version of this problem.

Finally, let us briefly discuss the infinite horizon extension. Iyengar [16] and Nilim and El Ghaoui [25] establish that the respective controller and nature policies are stationary for the infinite horizon problem. Moreover, Nilim and El Ghaoui [25] show that the optimal value function of the infinite horizon problem with a discounted cost function can be obtained as the unique limit of the finite horizon problem. This suggests that the optimal policy structure can be extended to the infinite horizon case.

3.6 Discussion

3.6.1 Uncertainty Affecting Parameters

So far uncertainty affecting transition probability distributions was considered. Alternatively, other problem parameters such as costs, rewards or batch sizes may also be subject to uncertainty. In general, uncertainty of other parameters than the transition probability distributions leads to a more challenging situation and the previous results do not seem to generalize easily.

In order to present some of the contrasts and the challenges, we present an illustrative example where we consider the inventory rationing problem under the assumption that the rewards of the arriving customer classes are uncertain. Like transition probabilities, we assume that the uncertainty set defining reward (cost) values possesses the rectangularity property and that Nature independently selects its optimal action from the uncertainty set $\mathcal{R}_t$ in period t. Hence, the set of all admissible policies of Nature given π is:

$$\tau^{\pi} = \mathcal{R}_1 \times \mathcal{R}_2 \times \mathcal{R}_3 \times ... \times \mathcal{R}_T,$$

In order to model the robust counterpart of the problem we represent the combined inventory rationing operator $\mathbf{T}_{R_I}$ as follows:

$$\mathbf{T}_{R_I} w_{t+1}(x) = \max_{\boldsymbol{a}} \left\{ \min_{\boldsymbol{R}_{i,t}(a,x)} \sum_{i=0}^{n} p_{i,t}(a_i R_{i,t}(a,x) + w_{t+1}(x - a_i)) \right\} \tag{3.13}$$

Since $a_i \in \{0, 1\}$ and noting that $\sum_{i=0}^{n} p_{i,t} \leq 1$, the preceding equation can also be expressed as:

$$\mathbf{T}_{R_I} w_{t+1}(x) = \max_{\boldsymbol{a}} \left\{ \min_{\boldsymbol{R}_{i,t}(a,x)} \sum_{i=0}^{n} p_{i,t} a_i (R_{i,t}(a,x) - \Delta w_{t+1}(x)) + \sum_{i=0}^{n} p_{i,t} w_{t+1}(x) \right\}. \tag{3.14}$$

Please note that the above representation represents the inventory rationing operators as a combined operator $\mathbf{T}_{R_I} w_{t+1}(x))$. Now consider the robust counterpart of the inventory rationing operator when rewards are uncertain:
The dual counterpart is given as follows:

$$\mathbf{T}_{R_I} \widetilde{w}_{t+1}(x) = \min_{\boldsymbol{R}_t(x)} \{ \max_{\boldsymbol{a}(\boldsymbol{R}_t(x),x)} \sum_{i=0}^{n} p_{i,t}(a_i R_{i,t}(x) + \widetilde{w}_{t+1}(x - a_i)) \}.$$

It turns out that the above problem does not always have the perfect duality property. To provide an example, suppose each class has an arrival probability of 0.10 and the uncertainty set consists of two distinct points over the horizon, i.e. $\mathcal{R} = \boldsymbol{R}^1, \boldsymbol{R}^2$ and $\mathcal{R}^1 = (60, 40, 45, 10, 10)$ and $\mathcal{R}^2 = (60, 40, 42.5, 20, 10)$. Consider the first stage and the dual counterpart $\widetilde{w}_1(x)$ given in equation (3.13). In that case, it is apparent that it is optimal to accept all classes for both cases and the optimal solution $\boldsymbol{a}_1(\boldsymbol{R}^1, x)$ and $\boldsymbol{a}_1(\boldsymbol{R}^2, x)$ are $(1, 1, 1, 1, 1)$ and the corresponding Nature solution is $\boldsymbol{R}^2$ respectively. If we consider $w_1(x)$ we obtain the optimal solution by enumerating each action $\boldsymbol{a}_1(x)$. It is found that the optimal solutions do not change, i.e. perfect duality holds at stage 1 for all x and $\widetilde{w}_1(x) = w_1(x) = 15.5$ for all $x > 0$.
Now consider the second stage and suppose the inventory level is 1. First, consider the dual counterpart $\widetilde{w}_t(x)$, if Nature selects $\boldsymbol{R}^1$, the optimal action $\boldsymbol{a}_2(\boldsymbol{R}^1, 1)$ is $(1, 1, 1, 0, 0)$ with a value function 17.6. If Nature selects $\boldsymbol{R}^2$, the optimal action $\boldsymbol{a}_2(\boldsymbol{R}^2, 1)$ is $(1, 1, 1, 1, 0)$ with a value function 17.8. Hence, the solution pair is $(1, 1, 1, 0, 0)$ and 17.6 respectively in that setting. We obtain $w_t(x)$ similarly by enumerating all possible actions $\boldsymbol{a}$:
The optimal actions of the controller and Nature are $(1, 1, 1, 0, 0)$ and $\boldsymbol{R}^2$ respectively with a corresponding value function 17.35. Please note that the fourth class with a reward $R_4 = 20$ is rejected although $20 > 15.5$.

Even though perfect duality does not hold, it is clear that Nature's optimal solution depends only on the action chosen by the controller, i.e. is independent of the state x. Then, it is straightforward to show that the structural properties we mention in Theorem 3.3.2 propagate to the robust counterpart $w_t(x)$. Now we state our next Lemma.

Table 3.3: Calculation of the robust value function $w_{T-1}(1)$

	a_1	a_2	a_3	a_4	a_5	$w^1(\boldsymbol{a})$	a_1	a_2	a_3	a_4	a_5	$w^2(\boldsymbol{a})$	$min(w^1(\boldsymbol{a}), w^2(\boldsymbol{a}))$
$\boldsymbol{a}^1$	0	0	0	0	0	0	0	0	0	0	0	0	0
$\boldsymbol{a}^2$	0	0	0	0	1	0	0	0	0	0	1	1	0
$\boldsymbol{a}^3$	0	0	0	1	0	1	0	0	0	1	0	2	1
$\boldsymbol{a}^4$	0	0	0	1	1	1	0	0	0	1	1	3	1
$\boldsymbol{a}^5$	0	0	1	0	0	4.5	0	0	1	0	0	4.25	4.25
$\boldsymbol{a}^6$	0	0	1	0	1	4.5	0	0	1	0	1	5.25	4.5
$\boldsymbol{a}^7$	0	0	1	1	0	5.5	0	0	1	1	0	6.25	5.5
$\boldsymbol{a}^8$	0	0	1	1	1	5.5	0	0	1	1	1	7.25	5.5
$\boldsymbol{a}^9$	0	1	0	0	0	4	0	1	0	0	0	4	4
$\boldsymbol{a}^{10}$	0	1	0	0	1	4	0	1	0	0	1	5	4
$\boldsymbol{a}^{11}$	0	1	0	1	0	5	0	1	0	1	0	6	5
$\boldsymbol{a}^{12}$	0	1	0	1	1	5	0	1	0	1	1	7	5
$\boldsymbol{a}^{13}$	0	1	1	0	0	8.5	0	1	1	0	0	8.25	8.25
$\boldsymbol{a}^{14}$	0	1	1	0	1	8.5	0	1	1	0	1	9.25	8.5
$\boldsymbol{a}^{15}$	0	1	1	1	0	9.5	0	1	1	1	0	10.25	9.5
$\boldsymbol{a}^{16}$	0	1	1	1	1	9.5	0	1	1	1	1	11.25	9.5
$\boldsymbol{a}^{17}$	1	0	0	0	0	6	1	0	0	0	0	6	6
$\boldsymbol{a}^{18}$	1	0	0	0	1	6	1	0	0	0	1	7	6
$\boldsymbol{a}^{19}$	1	0	0	1	0	7	1	0	0	1	0	8	7
$\boldsymbol{a}^{20}$	1	0	0	1	1	7	1	0	0	1	1	9	7
$\boldsymbol{a}^{21}$	1	0	1	0	0	10.5	1	0	1	0	0	10.25	10.25
$\boldsymbol{a}^{22}$	1	0	1	0	1	10.5	1	0	1	0	1	11.25	10.5
$\boldsymbol{a}^{23}$	1	0	1	1	0	11.5	1	0	1	1	0	12.25	11.5
$\boldsymbol{a}^{24}$	1	0	1	1	1	11.5	1	0	1	1	1	13.25	11.5
$\boldsymbol{a}^{25}$	1	1	0	0	0	10	1	1	0	0	0	10	10
$\boldsymbol{a}^{26}$	1	1	0	0	1	10	1	1	0	0	1	11	10
$\boldsymbol{a}^{27}$	1	1	0	1	0	11	1	1	0	1	0	12	11
$\boldsymbol{a}^{28}$	1	1	0	1	1	11	1	1	0	1	1	13	11
$\boldsymbol{a}^{29}$	1	1	1	0	0	14.5	1	1	1	0	0	14.25	14.25
$\boldsymbol{a}^{30}$	1	1	1	0	1	14.5	1	1	1	0	1	15.25	14.5
$\boldsymbol{a}^{31}$	1	1	1	1	0	15.5	1	1	1	1	0	16.25	15.5
$\boldsymbol{a}^{32}$	1	1	1	1	1	15.5	1	1	1	1	1	17.25	15.5

Table 3.4: Calculation of the robust value function $w_{T-2}(1)$

	a_1	a_2	a_3	a_4	a_5	$w^1(\boldsymbol{a})$	a_1	a_2	a_3	a_4	a_5	$w^2(\boldsymbol{a})$	$min(w^1(\boldsymbol{a}), w^2(\boldsymbol{a}))$
$\boldsymbol{a}^1$	0	0	0	0	0	7.75	0	0	0	0	0	7.75	7.75
$\boldsymbol{a}^2$	0	0	0	0	1	7.2	0	0	0	0	1	7.2	7.2
$\boldsymbol{a}^3$	0	0	0	1	0	7.2	0	0	0	1	0	8.2	7.2
$\boldsymbol{a}^4$	0	0	0	1	1	6.65	0	0	0	1	1	7.65	6.65
$\boldsymbol{a}^5$	0	0	1	0	0	10.7	0	0	1	0	0	10.45	10.45
$\boldsymbol{a}^6$	0	0	1	0	1	10.15	0	0	1	0	1	9.9	9.9
$\boldsymbol{a}^7$	0	0	1	1	0	10.15	0	0	1	1	0	10.9	10.15
$\boldsymbol{a}^8$	0	0	1	1	1	9.6	0	0	1	1	1	10.35	9.6
$\boldsymbol{a}^9$	0	1	0	0	0	10.2	0	1	0	0	0	10.2	10.2
$\boldsymbol{a}^{10}$	0	1	0	0	1	9.65	0	1	0	0	1	9.65	9.65
$\boldsymbol{a}^{11}$	0	1	0	1	0	9.65	0	1	0	1	0	10.65	9.65
$\boldsymbol{a}^{12}$	0	1	0	1	1	9.1	0	1	0	1	1	10.1	9.1
$\boldsymbol{a}^{13}$	0	1	1	0	0	13.15	0	1	1	0	0	12.9	12.9
$\boldsymbol{a}^{14}$	0	1	1	0	1	12.6	0	1	1	0	1	12.35	12.35
$\boldsymbol{a}^{15}$	0	1	1	1	0	12.6	0	1	1	1	0	13.35	12.6
$\boldsymbol{a}^{16}$	0	1	1	1	1	12.05	0	1	1	1	1	12.8	12.05
$\boldsymbol{a}^{17}$	1	0	0	0	0	12.2	1	0	0	0	0	12.2	12.2
$\boldsymbol{a}^{18}$	1	0	0	0	1	11.65	1	0	0	0	1	11.65	11.65
$\boldsymbol{a}^{19}$	1	0	0	1	0	11.65	1	0	0	1	0	12.65	11.65
$\boldsymbol{a}^{20}$	1	0	0	1	1	11.1	1	0	0	1	1	12.1	11.1
$\boldsymbol{a}^{21}$	1	0	1	0	0	15.15	1	0	1	0	0	14.9	14.9
$\boldsymbol{a}^{22}$	1	0	1	0	1	14.6	1	0	1	0	1	14.35	14.35
$\boldsymbol{a}^{23}$	1	0	1	1	0	14.6	1	0	1	1	0	15.35	14.6
$\boldsymbol{a}^{24}$	1	0	1	1	1	14.05	1	0	1	1	1	14.8	14.05
$\boldsymbol{a}^{25}$	1	1	0	0	0	14.65	1	1	0	0	0	14.65	14.65
$\boldsymbol{a}^{26}$	1	1	0	0	1	14.1	1	1	0	0	1	14.1	14.1
$\boldsymbol{a}^{27}$	1	1	0	1	0	14.1	1	1	0	1	0	15.1	14.1
$\boldsymbol{a}^{28}$	1	1	0	1	1	13.55	1	1	0	1	1	14.55	13.55
$\boldsymbol{a}^{29}$	1	1	1	0	0	17.6	1	1	1	0	0	17.35	**17.35**
$\boldsymbol{a}^{30}$	1	1	1	0	1	17.05	1	1	1	0	1	16.8	16.8
$\boldsymbol{a}^{31}$	1	1	1	1	0	17.05	1	1	1	1	0	17.8	17.05
$\boldsymbol{a}^{32}$	1	1	1	1	1	16.5	1	1	1	1	1	17.25	16.5

Table 3.5: Illustration of concavity for each case

$a_i(x-1)$	$a_i(x+1)$	Inequality
0	0	$w_{t+1}(x) + w_{t+1}(x) \geq w_{t+1}(x-1) + w_{t+1}(x+1)$
1	1	$w_{t+1}(x-1) + R_{i,t}(x-1) + w_{t+1}(x-1) + R_{i,t}(x+1) \geq$
		$w_{t+1}(x-2) + R_{i,t}(x-1)w_{t+1}(x) + R_{i,t}(x+1)$
0	1	$w_{t+1}(x-1) + w_{t+1}(x) + R_{i,t}(x+1) \geq w_{t+1}(x-1) + w_{t+1}(x) + +R_{i,t}(x+1)$
1	0	$w_{t+1}(x-1) + R_{i,t}(x-1) + w_{t+1}(x) \geq w_{t+1}(x-2) + R_{i,t}(x-1) + w_{t+1}(x+1)$
		$w_{t+1}(x-1) - w_{t+1}(x-2) \geq w_{t+1}(x) - w_{t+1}(x-1) \geq w_{t+1}(x+1) - w_{t+1}(x-1)$

Lemma 3.6.1 *Let the robust value function $w_{t+1}(x)$ be concave in x for all x at stage $t+1$, then the combined rationing operator $\mathbf{T}_{R_I} w_{t+1}(x)$ has the following properties:*

1. $\mathbf{T}_{R_I} w_{t+1}(x)$ *is concave in* x*,*

2. $\mathbf{T}_{R_I} w_{t+1}(x)$ *is ND in* x*, and*

3. $\mathbf{T}_{R_I} w_{t+1}(x)$ *preserves supermodularity in* x, t*,*

Proof: Now suppose that set $\boldsymbol{a}_t(x+1)$ denotes the optimal action at $x+1$ and t, whereas $\boldsymbol{a}_t(x-1)$ denotes the optimal action at $x-1$ and t. Furthermore, let's denote the Nature's corresponding optimal decisions as $\boldsymbol{R}_t(x+1) = (R_{1,t}(x+1),\ R_{2,t}(x+1), ..., R_{n,t}(x+1)$ and $\boldsymbol{R}_t(x-1) = (R_{1,t}(x-1),\ R_{2,t}(x-1), ..., R_{n,t}(x-1)$ respectively, in state $x+1$ and $x-1$. Further suppose that we use the optimal actions of states $x+1$ and $x-1$ for state x in the following equation. Then we obtain the following inequality:

$$\sum_{i=0}^{n} a_{i,t}(x-1)p_{i,t}(w_{t+1}(x-1) + R_{i,t}(x-1)) + p_{i,t}(1 - a_{i,t}(x-1))w_{t+1}(x) +$$
$$\sum_{i=0}^{n} a_{i,t}(x+1)p_{i,t}(w_{t+1}(x-1) + R_{i,t}(x+1)) + p_{i,t}(1 - a_{i,t}(x+1))w_{t+1}(x) \geq$$
$$\sum_{i=0}^{n} a_{i,t}(x-1)p_{i,t}(w_{t+1}(x-2) + R_{i,t}(x-1)) + p_{i,t}(1 - a_{i,t}(x-1))w_{t+1}(x-1) +$$
$$\sum_{i=0}^{n} a_{i,t}(x+1)p_{i,t}(w_{t+1}(x) + R_{i,t}(x+1)) + p_{i,t}(1 - a_{i,t}(x+1))w_{t+1}(x+1).$$

In order to simplify the former inequality, we analyze the inequality for each operator i independently. The results are given in Table 3.6.1:

The proof of concavity and preservation of supermodularity are similar. Suppose $w_{t+1}(x)$ is ND in x at t for all x. Further, suppose that we use the optimal action of state $x-1$ for state x, then the result follows easily by optimality at x for every i:

$$a_{i,t}(x-1)\left[w_{t+1}(x-1)+R_{i,t}(x-1)\right]+(1-a_{i,t}(x-1))w_{t+1}(x) \geq$$
$$a_{i,t}(x-1)\left[w_{t+1}(x-2)+R_{i,t}(x-1)\right]+(1-a_{i,t}(x-1))w_{t+1}(x-1).$$

In order to prove the preservation of supermodularity we again use the optimal action of state $x-1$ for state x, Then by concavity every operator i satisfies the following inequality:

$$\sum_{i=0}^{n} p_{i,t}a_i[R_{i,t}(x-1)-\Delta w_{t+1}(x)] \geq \sum_{i=0}^{n} p_{i,t}a_i[R_{i,t}(x-1)-\Delta w_{t+1}(x-1)]. \tag{3.15}$$

Equation (3.15) . □

Lemma 3.6.1 states that the combined rationing operator $\mathbf{T_{R_I}}w_{t+1}(x)$ can be incorporated in any robust value function $w_t(x)$ constituted of the operators given in Table 3.1 with uncertain (certain) event probabilities without violating the structural properties of $w_t(x)$. It is important to note that Lemma 3.6.1 does not mean that the optimal admission policy has a monotone structure. However, if we introduce any inventory production operator given in Table 3.1 to the problem, the optimal production policy will be base stock level.

In fact, consider the following example presented in order to show that it may be optimal to sell to a customer class at lower levels of inventory and not to sell the same class at higher levels of inventory. Here we provide 5 customer classes. The parameters of these classes are as follows from $t = T-1$ to $t = T-13$.

Class	Reward	Arrival probability
1	60	0.1
2	50	0.1
3	45	0.1
4	23	0.13
5	20	0.1

Now suppose that at $t = T - 14$ the rewards take the following discrete values without any change in the arrival probabilities:

Scenario	R_1	R_2	R_3	R_4	R_5
1	45	50	40	46	50
2	60	50	42	31	20
3	50	55	40	27	30
4	50	60	45	23	20

The calculations are given in the following table, please note that a_i stands for controller decision for the class-i, where 1 denotes admission and 0 denotes rejection, S denotes the scenario.

At $x = 4$ the calculated values are as follows:

$\Delta w_{T-13}(4)$	a_1	a_2	a_3	a_4	a_5	Scn1	Scn2	Scn3	Scn4	Nature Solution
34.68142	0	0	0	0	0	0	0	0	0	0
34.68142	0	0	0	1	0	1.471415	-0.47858	-0.99858	-1.51858	-1.51858
34.68142	0	0	0	0	1	1.531858	-1.46814	-0.46814	-1.46814	-1.46814
34.68142	0	0	0	1	1	3.003274	-1.94673	-1.46673	-2.98673	-2.98673
34.68142	0	0	1	0	0	0.531858	0.731858	0.531858	1.031858	0.531858
34.68142	0	0	1	1	0	2.003274	0.253274	-0.46673	-0.48673	-0.48673
34.68142	0	0	1	0	1	2.063716	-0.73628	0.063716	-0.43628	-0.73628
34.68142	0	0	1	1	1	3.535132	-1.21487	-0.93487	-1.95487	-1.95487
34.68142	0	1	0	0	0	1.531858	1.531858	2.031858	2.531858	1.531858
34.68142	0	1	0	1	0	3.003274	1.053274	1.033274	1.013274	1.013274
34.68142	0	1	0	0	1	3.063716	0.063716	1.563716	1.063716	0.063716
34.68142	0	1	0	1	1	4.535132	-0.41487	0.565132	-0.45487	-0.45487
34.68142	0	1	1	0	0	2.063716	2.263716	2.563716	3.563716	2.063716
34.68142	0	1	1	1	0	3.535132	1.785132	1.565132	2.045132	1.565132
34.68142	0	1	1	0	1	3.595574	0.795574	2.095574	2.095574	0.795574
34.68142	0	1	1	1	1	5.06699	0.31699	1.09699	0.57699	0.31699
34.68142	1	0	0	0	0	1.031858	2.531858	1.531858	1.531858	1.031858
34.68142	1	0	0	1	0	2.503274	2.053274	0.533274	0.013274	0.013274
34.68142	1	0	0	0	1	2.563716	1.063716	1.063716	0.063716	0.063716
34.68142	1	0	0	1	1	4.035132	0.585132	0.065132	-1.45487	-1.45487
34.68142	1	0	1	0	0	1.563716	3.263716	2.063716	2.563716	1.563716
34.68142	1	0	1	1	0	3.035132	2.785132	1.065132	1.045132	1.045132
34.68142	1	0	1	0	1	3.095574	1.795574	1.595574	1.095574	1.095574
34.68142	1	0	1	1	1	4.56699	1.31699	0.59699	-0.42301	-0.42301
34.68142	1	1	0	0	0	2.563716	4.063716	3.563716	4.063716	2.563716
34.68142	1	1	0	1	0	4.035132	3.585132	2.565132	2.545132	2.545132
34.68142	1	1	0	0	1	4.095574	2.595574	3.095574	2.595574	2.595574
34.68142	1	1	0	1	1	5.56699	2.11699	2.09699	1.07699	1.07699
34.68142	1	1	1	0	0	3.095574	4.795574	4.095574	5.095574	3.095574
34.68142	1	1	1	1	0	4.56699	4.31699	3.09699	3.57699	3.09699
34.68142	1	1	**1**	**0**	1	**4.627432**	**3.327432**	**3.627432**	**3.627432**	**3.327432**
34.68142	1	1	1	1	1	6.098848	2.848848	2.628848	2.108848	2.108848

At $x = 5$ the calculated values are as follows:

$\Delta w_{T-13}(4)$	a_1	a_2	a_3	a_4	a_5	Scn1	Scn2	Scn3	Scn4	Nature Solution
26.99553	0	0	0	0	0	0	0	0	0	0
26.99553	0	0	0	1	0	2.470581	0.520581	0.000581	-0.51942	-0.51942
26.99553	0	0	0	0	1	2.300447	-0.69955	0.300447	-0.69955	-0.69955
26.99553	0	0	0	1	1	4.771028	-0.17897	0.301028	-1.21897	-1.21897
26.99553	0	0	1	0	0	1.300447	1.500447	1.300447	1.800447	1.300447
26.99553	0	0	1	1	0	3.771028	2.021028	1.301028	1.281028	1.281028
26.99553	0	0	1	0	1	3.600894	0.800894	1.600894	1.100894	0.800894
26.99553	0	0	1	1	1	6.071475	1.321475	1.601475	0.581475	0.581475
26.99553	0	1	0	0	0	2.300447	2.300447	2.800447	3.300447	2.300447
26.99553	0	1	0	1	0	4.771028	2.821028	2.801028	2.781028	2.781028
26.99553	0	1	0	0	1	4.600894	1.600894	3.100894	2.600894	1.600894
26.99553	0	1	0	1	1	7.071475	2.121475	3.101475	2.081475	2.081475
26.99553	0	1	1	0	0	3.600894	3.800894	4.100894	5.100894	3.600894
26.99553	0	1	1	1	0	6.071475	4.321475	4.101475	4.581475	4.101475
26.99553	0	1	1	0	1	5.901341	3.101341	4.401341	4.401341	3.101341
26.99553	0	1	1	1	1	8.371922	3.621922	4.401922	3.881922	3.621922
26.99553	1	0	0	0	0	1.800447	3.300447	2.300447	2.300447	1.800447
26.99553	1	0	0	1	0	4.271028	3.821028	2.301028	1.781028	1.781028
26.99553	1	0	0	0	1	4.100894	2.600894	2.600894	1.600894	1.600894
26.99553	1	0	0	1	1	6.571475	3.121475	2.601475	1.081475	1.081475
26.99553	1	0	1	0	0	3.100894	4.800894	3.600894	4.100894	3.100894
26.99553	1	0	1	1	0	5.571475	5.321475	3.601475	3.581475	3.581475
26.99553	1	0	1	0	1	5.401341	4.101341	3.901341	3.401341	3.401341
26.99553	1	0	1	1	1	7.871922	4.621922	3.901922	2.881922	2.881922
26.99553	1	1	0	0	0	4.100894	5.600894	5.100894	5.600894	4.100894
26.99553	1	1	0	1	0	6.571475	6.121475	5.101475	5.081475	5.081475
26.99553	1	1	0	0	1	6.401341	4.901341	5.401341	4.901341	4.901341
26.99553	1	1	0	1	1	8.871922	5.421922	5.401922	4.381922	4.381922
26.99553	1	1	1	0	0	5.401341	7.101341	6.401341	7.401341	5.401341
26.99553	1	1	1	1	0	**7.871922**	**7.621922**	**6.401922**	**6.881922**	**6.401922**
26.99553	1	1	**1**	**0**	1	7.701788	6.401788	6.701788	6.701788	6.401788
26.99553	1	1	1	1	1	10.17237	6.922369	6.702369	6.182369	6.182369

Hence, the respective optimal controller and Nature policies at $x = 4$ and $x = 5$ are:

x	Admission Policy	Reward Values
4	(1, 1, 1, 0, 1)	(60 50 42 20 31)
5	(1, 1, 1, 1, 0)	(50 55 40 27 30)

This shows that optimal policy does not have a threshold structure.

However, structural results of the optimal policies of other operators are not violated in that case. In order to present an illustration, let us consider a particular example where there is one or more production units with unknown production completion probabilities but certain production costs and the customer classes have known arrival rates but uncertain rewards.

In this case the optimal production policies are threshold type.

3.6.2 Robust Control of Server Assignment Models

In this section, we extend our results to some sample problems from queueing theory where multi-dimensional properties are of concern. In order to be consistent with the literature on queueing systems, we change our notation in this section slightly and use c instead of h in order to denote the holding cost. Second, unlike the other parts of the thesis, the problems of this section are cost minimization problems. The first model we consider is the server assignment model [19]. Customers of different types arrive according to a Poisson process at a service station with a single server. Class-i customers require an exponentially distributed service time with parameter $\mu(i)$. Each time unit a class-i customer spends waiting costs $c(i)$ and the service might be interrupted. According to the μc rule if Class i customer enters to the system while the server is busy with serving to customer j with $\mu(i)c(i) \geq \mu(j)c(j)$, the service that has been provided to the customer j shall be interrupted and switched to the Class i customer. We show that the optimal policy of robust counterpart of the server assignment model is of μc type under the given conditions. Another model is the tandem server model where the arriving customers are served by sequential servers. It is shown by Koole [19] that the optimal policy is to first serve the jobs at the earlier queues in the sequence. We show that this optimality condition also holds for the robust case. Similar to the server assignment model, the μc rule applies for the optimal control of movable tandem server where μ refers to the server rate and c refers to the holding cost. In order to show these characteristics we first define the operators we use in this section.

Operators The queueing operators we use are given in the following where $1 \leq \imath \leq m$

$T_{A_i}v(x) = \min v(x), v(x+e_i) - R_i$

$T_{UA_i}v(x) = v(x+e_i)$

$T_{UD_i}v(x) = v((x-e_i)^+)$

$T_{MTS}v(x) = \min_{j \in I: x_j > 0} \sum_{k=0}^{m} \mu(i,k)v(x-e_i+e_k)$ if $\sum_{j \in I} x_j > 0$

$T_{MTS}v(x) = v(x)$ otherwise

where $\sum_{k=0}^{m} \mu(i,k) = 1$ for all i and $\mu(i,j) = 0$ for all i and $0 < j < i-1$ and $e_0 = 0$

$T_{MS}v(x) = \min_{j \in I: x_j > 0} \mu(j)v(x - e_j) + (1 - \mu(j)v(x))$ if $\sum_{j \in I}$
$T_{MS}v(x) = v(x)$ otherwise

As discussed in the earlier sections the admission operator T_{A_i} models the queue where the decision is whether to admit the $Class - i$ job/customer to the queue and earn an instantaneous reward R_i or to reject the customer and reserve a one unit capacity in the system for more valuable jobs/customers that may arrive in the future. The uncontrolled arrival operator T_{UA_i} admits any arriving customer to the system, whereas the uncontrolled departure operator T_{UD_i} processes the next i-class customer in the system by interrupting an ongoing service if any. The controlled departure T_{MS} operator decides which customer class will be served in the system next and interruption of an ongoing service is allowed.

Properties of the Nominal Problem and the Robust Counterpart

Based on the results given in Koole the following operators preserve the following properties under the stated conditions:

Property 3.6.1 *The queueing operators, admission T_{A_i} and uncontrolled arrival operator T_{UA_i}, the uncontrolled departure operator T_{UD_i} T_{CD_i} preserve the UI property:*

$$T_i v(x + e_j) \leq T_i v(x + e_i) \text{ where } i < j$$

We show that the same results also hold for the value function $w_t(x)$ of the robust counterpart in the following.

Theorem 3.6.1 *The robust value function $w_t(x)$ constituted of the queueing operators T_{A_i}, T_{UA_i}, T_{UD_i} preserves UI property.*

Proof According to Property 3.6.1 the $w_t(x)$ holds the following inequality if uncertainty set $\mathcal{P}_t$ has a single element p. Therefore following holds for every $p \in \mathcal{P}_t$:

$$\sum_{i=1}^{n} p_t^i T_i w(x + e_j) \leq \sum_{i=1}^{n} p_t^i T_i w(x + e_i) \text{ where } i < j \; \forall \, p \in \mathcal{P}_t.$$

Now suppose that $p(x+e_j)$ and $p(x+e_i)$ are the respective solutions of the Nature in the robust setting. Hence, the following holds:

$$\sum_{i=1}^{n} p_t^i(x+e_j)T_i w(x+e_j) \leq$$
$$\sum_{i=1}^{n} p_t^i(x+e_j)T_i w(x+e_i) \leq$$
$$\sum_{i=1}^{n} p_t^i(x+e_i)T_i w(x+e_i) \text{ where } i < j.$$

This completes the proof.

Property 3.6.2 *The queueing operators uncontrolled arrival operator T_{UA_i} and the tandem movable server T_{MTS} preserve the gUI property, i.e:*

$$\sum_{k=1}^{n} \mu(i,k)Tv(x+e_{i+1}+e_k) \leq \sum_{k=1}^{n} \mu(i+1,k)Tv(x+e_i+e_k).$$

Theorem 3.6.2 *The robust value function $w_t(x)$ constituted of the queueing operators T_{UA_i}, T_{MTS} preserve the gUI property.*

Proof According to Property 3.6.2, $w_t(x)$ satisfies the following inequality if uncertainty set $\mathcal{P}_t$ has a single element p. Therefore, the following holds for every $p \in \mathcal{P}_t$:

$$\sum_{k=1}^{k=n} \mu(i,k) \sum_{i=1}^{n} p_t^i T_i w(x+e_{i+1}+e_k) \leq \sum_{k=1}^{k=n} \mu(i+1,k) \sum_{i=1}^{n} p_t^i T_i w(x+e_i+e_k) \quad \forall\, p \in \mathcal{P}_t.$$

Now suppose that $p(x+e_{i+1})$ and $p(x+e_i)$ are the respective solutions of the Nature in the robust setting. Hence, the following holds:

$$\sum_{k=1}^{k=n} \mu(i,k) \sum_{i=1}^{n} p_t^i(x+e_{i+1})T_i w(x+e_{i+1}+e_k) \leq \sum_{k=1}^{k=n} \mu(i+1,k) \sum_{i=1}^{n} p_t^i(x+e_i)T_i w(x+e_i+e_k).$$

This completes the proof. □

Theorem 3.6.2 has an important implication. The optimal robust policy is to first serve the jobs in the earlier queues in the tandem sequence similar to its classical counterpart.
Based on the result given by Koole [19] the following property holds:

Property 3.6.3 *The queueing operators uncontrolled arrival operator T_{UA_i} and the movable server operator T_{MS} preserve the wUI property if $m = 2$ and $\mu_1 = \mu_2$ i.e:*

1. $\mu v(x+e_1)+(1-\mu)v(x+e_1+e_2) \leq \mu v(x+e_2)+(1-\mu)v(x+e_1+e_2)$

2. $v(x+e_1)+v(x+e_2) \leq v(x)+v(x+e_1+e_2)$

According to Property 3.6.3, if we consider a queueing system consisting of one movable server with equal server rates serving both customer classes, then the optimal policy is to serve to the class with the higher holding cost. Thus, whenever a customer that has the higher holding cost (customer class-1) arrives, the server serving the other customer (customer class-2) interrupts the service and does not serve her until there is no remaining class-1 customer in the system.

Theorem 3.6.3 *The robust value function $w_t(x)$ constituted of the queueing operators T_{A_i}, T_{UA_i} and T_{MS} has the following wUI property, i.e.* $\mu w(x+e_1)+(1-\mu)w(x+e_1+e_2) \leq \mu w(x+e_2)+(1-\mu)w(x+e_1+e_2)$

Proof According to Property 3.6.2 the $w_t(x)$ holds the following inequality if uncertainty set $\mathcal{P}_t$ has a single element p. Therefore, the following holds for every $p_t \in \mathcal{P}_t$:

$$\begin{aligned} &\mu\sum_{i=1}^{n} p_t^i T_i w(x+e_1)+(1-\mu)\sum_{i=1}^{n} p_t^i T_i w(x+e_1+e_2) \leq \\ &\quad \mu\sum_{i=1}^{n} p_t^i T_i w(x+e_2)+(1-\mu)\sum_{i=1}^{n} p_t^i T_i w(x+e_1+e_2). \end{aligned} \tag{3.16}$$

Now suppose that $p(x+e_1)$, $p(x+e_2)$ and $p(x+e_1+e_2)$ are the respective solutions of the Nature in the robust setting. Hence, the following holds:

$$\begin{aligned} \mu\sum_{i=1}^{n} p_t^i(x+e_1)T_i w(x+e_1) &\leq \mu\sum_{i=1}^{n} p_t^i(x+e_1)T_i w(x+e_2) \\ \mu\sum_{i=1}^{n} p_t^i(x+e_1)T_i w(x+e_1) &\leq \mu\sum_{i=1}^{n} p_t^i(x+e_2)T_i w(x+e_2). \end{aligned}$$

This completes the proof □.

Chapter 4

A SINGLE-PRODUCT REVENUE MANAGEMENT PROBLEM WITH INTERVAL UNCERTAINTY ON DEMAND/ PRODUCTION RATES

4.1 Introduction

In Chapter 3 we consider a general definition of uncertainty set but this chapter considers a particular type of uncertainty set. In order to represent the connection between the real problem data and the prior estimations, a number of uncertainty models have been proposed. We briefly discuss these uncertainty models in Chapter 2. One relatively simple uncertainty model is to define uncertainty intervals where parameters are allowed to lie between a lower bound and an upper bound [28]. Ben-Tal, Nemirovsky and El Ghaoui [4] and Nilim and El Ghaoui [25] propose a number of more sophisticated yet tractable uncertainty sets. Some recent applications of such uncertainty sets include Lim and Shanthikumar [23], Jain, Lim and Shanthikumar [17] which employ entropy based models of uncertainty in robust dynamic pricing problems. We employ the simpler model of interval uncertainty in this chapter because our focus is on exploring the structure of optimal policies rather than developing efficient uncertainty sets or fitting data to existing uncertainty sets. Some recent papers investigate the admission control problem under demand rate uncertainty. An absolute robust approach for both the static and dynamic versions of this problem is suggested by Birbil, Frenk, Gromicho and Zhang [7] where they employ an ellipsoidal model of uncertainty. It is shown that this uncertainty model leads to tractable solutions of the problem. By considering an interval uncertainty model, we complement the results of [7] by obtaining additional properties of optimal robust policies. Finally, Lan, Gao, Ball and Karaesmen [21] and Ball and Queyranne [3] propose and explore an alternative approach for addressing uncertainty based on a competitive analysis method.

In contrast to the above papers, we obtain results on the structure of the optimal policy for both cases with or without replenishment. We also explore how the optimal policies change with respect to uncertainty sets. The dynamic queueing and inventory control literature has a strong tradition in characterizing the structure of optimal policies. This is in part due to the computational efficiency of structured policies. However, the main reason for looking for structured policies is that they are usually expressed in a few parameters and tend to be easy to understand, communicate and implement. There are known general approaches to investigate the structure of the solution of a stochastic dynamic program (for instance Koole [18], [19], Çil, Örmeci and Karaesmen [8]). The current chapter can be seen as an exploration of such properties in the context of a robust stochastic dynamic program. In addition to the results in Chapter 3, we explore additional properties with respect to varying uncertainty sets and present a numerical study.

The organization of this chapter is as follows. In Section 4.2 we analyze the single-item inventory model without replenishment and its robust counterpart. In Section 4.3 we explore the properties obtained in Section 4.2 for a model where replenishment is allowed. In Section 4.4, we present some numerical results and explore some of the quantitative trade-offs. Finally, our conclusions are provided in Section 4.5.

4.2 A ROBUST DYNAMIC REVENUE MANAGEMENT PROBLEM

4.2.1 Nominal Problem: Dynamic Single-Product Revenue Management

We consider the discrete-time formulation of a single product (i.e. single resource or single-leg) revenue management problem. Consider a planning horizon consisting of T periods. There are n customer classes differentiated by their admission rewards R_i for class-i $(i = 1, 2, ..., n)$. Without loss of generality, we assume that the classes are ordered such that $R_i > R_j$, if $i < j$ $(i, j = 1, 2, ..., n)$. At most one arrival can take place in each period t $(t = 0, 1, 2, ..., T)$.

Let x (where $x \in \mathcal{Z}^+$) be the remaining inventory (number of seats available). A class-i customer arrives in a given period t with probability $\lambda_{i,t}$ and $\lambda_{n+1,t}$ denotes the probability of no-arrival which can also be considered as special customer class with 0 reward. Obviously, $\lambda_{i,t} \geq 0$ for all i, t and $\sum_{i=1}^{n+1} \lambda_{i,t} = 1$. We denote the arrival probability vector in period t by $\boldsymbol{\lambda}_t = (\lambda_{1,t}, \lambda_{2,t}, \ldots, \lambda_{n+1,t})$.

The admission control problem is to decide whether to admit or reject an arriving demand considering the remaining capacity x in each period in order to maximize the expected revenue over the planning horizon. Let $v_t(x)$ denote the expected optimal revenue starting from period t with a capacity of x. The optimality equation can be written as:

$$v_t(x) = \sum_{i=1}^{n} \lambda_{i,t} \max\{R_i + v_{t+1}(x-1), v_{t+1}(x)\} + \lambda_{n+1,t}(x) v_{t+1}(x), \tag{4.1}$$

with boundary conditions $v_t(0) = 0$ for all t and $v_T(x) = 0$ for all x. Let us denote the difference function by $\Delta v(x) = v(x) - v(x-1)$. Clearly, a class-$i$ customer is accepted at stage t if and only if $R_i - \Delta v_{t+1}(x) \geq 0$.
We can alternatively express (4.1) as follows:

$$v_t(x) = \sum_{i=1}^{n} \lambda_{i,t} \left(R_i - \Delta v_{t+1}(x)\right)^+ + v_{t+1}(x), \tag{4.2}$$

where $(R_i - \Delta v_{t+1}(x))^+$ denotes $\max(0, R_i - \Delta v_{t+1}(x))$.

A number of structural results for the optimal policy are well-known for the above problem. In particular, $v_t(x)$ is concave in x for all t (Lautenbacher and Stidham [22]). This implies the optimality of threshold type policies. In each period t and for each class-i, there is an admission threshold $\ell_{i,t}$. If the inventory available at time t, $x \geq \ell_{i,t}$ then the class-i demand will be admitted, otherwise it will be rejected. Moreover, the thresholds ordered according to the rewards of the classes: $\ell_{1,t} = 1$ and $\ell_{i,t} \leq \ell_{j,t}$ if $i < j$, for $t = 1, 2, ..., T$.

In addition to being concave, $v_t(x)$ is also supermodular in t and x, i.e. $\Delta v_t(x) \geq \Delta v_{t+1}(x)$ for all t, x (see Talluri and Van Ryzin [32] or Aydin, Akcay and Karaesmen [2]). This implies that the optimal thresholds are non-increasing over time: $\ell_{i,t} \geq \ell_{i,t+1}$ for all i and t.

We conclude this section by establishing an additional property of the value function $v_t(x)$ as a function of the arrival probability vector. To this end, let us employ the following partial order. An arrival probability vector is said to be *preferred* (denoted by the $\succeq$ operator) over another if it receives higher classes with higher probability:

$$\boldsymbol{\lambda}_t \succeq \boldsymbol{\lambda}'_t \; \Leftrightarrow \; \sum_{i=1}^{k} \lambda_{i,t} \geq \sum_{i=1}^{k} \lambda'_{i,t} \;\; \forall\, k = 1, 2, \ldots, n.$$

Theorem 4.2.1 *Consider two problems that are identical in their parameters except their arrival probabilities in period t. Let $\boldsymbol{\lambda}_t$ and $\boldsymbol{\lambda}'_t$ be two arrival probability vectors, and $v_t(x)$ and $v'_t(x)$ be the corresponding value functions respectively. If $\boldsymbol{\lambda}_t \succeq \boldsymbol{\lambda}'_t$ then:*

1. *$v_t(x) \geq v'_t(x)$ for all x, t,*

2. *$\Delta v_t(x) \geq \Delta v'_t(x)$ for all x, t.*

Proof:

Before proving Theorem 4.2.1 we introduce a simple algorithm that starts with $\boldsymbol{\lambda}$ and ends up with $\boldsymbol{\lambda}'$ (where $\boldsymbol{\lambda}_t \succeq \boldsymbol{\lambda}'_t$) by a sequence of reallocation of probabilities. By definition, $\sum_{i=1}^{n+1} \lambda_i = \sum_{i=1}^{n+1} \lambda'_i = 1$. Now consider the following sequence of vectors, $\boldsymbol{\lambda}^{(1)}, \boldsymbol{\lambda}^{(2)}, ..., \boldsymbol{\lambda}^{(n)}$. Let $\boldsymbol{\lambda}^{(1)} = \boldsymbol{\lambda}$.

Now let $\epsilon_1 = \lambda_1 - \lambda'_1$, we construct $\boldsymbol{\lambda}^{(2)} = \boldsymbol{\lambda}^{(1)} + (-\epsilon_1, \epsilon_1, 0, 0, ..., 0)$. Obviously $\epsilon_1 \geq 0$. In the next iteration, we let $\epsilon_2 = \lambda_2 + \lambda_1 - \lambda'_1 - \lambda'_2$, again $\epsilon_2 \geq 0$ and $\boldsymbol{\lambda}^{(3)} = \boldsymbol{\lambda}^{(2)} + (0, -\epsilon_2, \epsilon_2, 0, ..., 0)$. We continue to iterate similarly for $n-1$ steps. At step $n-1$ we have: $\boldsymbol{\lambda}^{(n)} = \boldsymbol{\lambda}^{(n-1)} + (0, 0, ..., -\epsilon_{n-1}, \epsilon_{n-1})$. By construction, we have $\boldsymbol{\lambda}^{(n)} = \boldsymbol{\lambda}'$. In addition, the sequence of vectors have the property $\boldsymbol{\lambda}^{(1)} \preceq \boldsymbol{\lambda}^{(2)} \preceq ... \preceq \boldsymbol{\lambda}^{(n)}$.

Proof:

i-Suppose at any stage t, we replace $\boldsymbol{\lambda}_t$ by $\boldsymbol{\lambda}'_t$ by the above algorithm. Hence, we consecutively decrease the arrival probability of a higher class-i by ϵ and increase the arrival probability of a lower class j by ϵ. By each iteration the value function $v_t(x)$ decreases by $\epsilon\,(R_i - \Delta v_{t+1}(x))^+$ and increases by $\epsilon\,(R_j - \Delta v_{t+1}(x))^+$ where $i < j$. Since $R_i > R_j$, $v_t(x)$ is non-increasing at each step. Applying the algorithm, we then obtain $v'_t(x) \leq v_t(x)$. For stage $t-1$, it is clear that $v'_{t-1}(x) \leq v_{t-1}(x)$ since $v'_{t-1}(x) =$

$\sum_{i\leq k}(v'_t(x-1)+R_i)+\sum_{i>k} v'_t(x)$, where k denotes the highest rejected class. Hence, $v_{t-1}(x) \geq \sum_{i\leq k}(v_t(x-1)+R_i)+\sum_{i>k} v_t(x) \geq v'_{t-1}(x)$ This completes the proof of part i.

ii- We prove the desired result in two phases corresponding to stages t and $t-1$. Consider two systems that are identical and substitute $\boldsymbol{\lambda}_t$ with $\boldsymbol{\lambda}'_t$. In the first phase, we prove that $\Delta v_t(x) \geq \Delta v'_t(x)$ holds at t, then in the second phase we prove that $\Delta v_{t-1}(x) \geq \Delta v'_{t-1}(x)$. Please note that at stage t, we have $v_{t+1}(x) = v'_{t+1}(x)$, for all x and $\boldsymbol{\lambda}_t \neq \boldsymbol{\lambda}'_t$ whereas in stage $t-1$, $v_t(x)$ and $v'_t(x)$ are not necessarily equal but $\boldsymbol{\lambda}_{t-1} = \boldsymbol{\lambda}'_{t-1}$.
Suppose $\Delta v_t(x) \geq \Delta v'_t(x)$ holds, then $v_t(x) - v'_t(x) \geq v_t(x-1) - v'_t(x-1)$. This implies that as we replace the $\boldsymbol{\lambda}_t$ with $\boldsymbol{\lambda}'_t$ at stage t the loss in $v_t(x)$ is greater than loss in $v_t(x-1)$. We again use the above algorithm in order to perform such a replacement. Hence, we consecutively decrease a class-i by ϵ and increase a class j by ϵ where $i < j$.

$$\begin{aligned} \epsilon\,(R_i - \Delta v_{t+1}(x))^+ - \epsilon\,(R_j - \Delta v_{t+1}(x))^+ \quad &\geq \quad \epsilon\,(R_i - \Delta v_{t+1}(x-1))^+ \\ &\quad -\epsilon\,(R_j - \Delta v_{t+1}(x-1))^+ . \qquad (4.3) \end{aligned}$$

We prove the inequality case by case, note that A stands for admission and R stands for rejection. The non-trivial cases are listed in below (we do not present the cases where all classes are accepted or all classes are rejected since these are obvious). Please note that accepting a lower class (j) means that a higher class (i) is always accepted. Also please note that, if a customer class is accepted at an inventory $x-1$ then it is also accepted at x, and if it is rejected at x then it is also rejected at $x-1$.

R_i	$-\ R_j$	$\geq$	R_i	$-\ R_j$	
$-\Delta v_{t+1}(x)$	$-\Delta v_{t+1}(x)$	$\geq$	$-\Delta v_{t+1}(x-1)$	$-\Delta v_{t+1}(x-1)$	Result
(A)	(A)		(A)	(R)	$R_i - R_j \ \geq$ $R_i - \Delta v_{t+1}(x-1)$
(A)	(R)		(A)	(R)	$R_i - \Delta v_{t+1}(x) \ \geq$ $R_i - \Delta v_{t+1}(x-1)$
(A)	(A)		(R)	(R)	$R_i - R_j \ \geq \ 0$
(A)	(R)		(R)	(R)	$R_i - \Delta v_{t+1}(x) \ \geq \ 0$

Except for the case in the first row, all inequalities follow easily by concavity of $v(x)$ (a summary of the result is provided in the last column). Consider the case in the first row: because class j is rejected at $x-1$ for this case, we must have $R_j - \Delta v_{t+1}(x-1) \ \leq \ 0$. This implies that $R_i - R_j \ \geq \ R_i - \Delta v_{t+1}(x-1)$.
Next, we consider the second phase where we need to establish that $\Delta v_{t-1}(x) \geq \Delta v'_{t-1}(x)$.

We use a similar approach here, but we consider only one operator T^i (admission decision for a single class) at a time. The cases related to accept all and reject all for both systems at states $x-1$ and x are obvious. Please note that since $\Delta v_t(x) \geq \Delta v'_t(x)$, therefore $R_i - \Delta v_t(x) \leq R_i - \Delta v'_t(x)$, which means that any class accepted to the initial system will always be accepted to the second system. Except from the obvious cases (accept all, reject all) there are only four alternatives:

$T^i v_{t-1}(x)$	$-T^i v'_{t-1}(x)$	$\geq$	$T^i v_{t-1}(x-1)$	$-T^i v'_{t-1}(x-1)$	Result
(A)	(A)		(R)	(R)	$R_i + v_t(x-1) - R_i - v'_t(x-1) \geq$ $v_t(x-1) - v'_t(x-1)$
(A)	(A)		(R)	(A)	$R_i + v_t(x-1) - R_i - v'_t(x-1) \geq$ $v'_t(x-1) - R_i - v'_t(x-2)$
(R)	(A)		(R)	(A)	$v_t(x) - R_i - v'_t(x-1) \geq$ $v_t(x-1) - R_i - v'_t(x-2)$
(R)	(A)		(R)	(R)	$v_t(x) - v'_t(x-1) - R_i \geq$ $v_t(x-1) - v'_t(x-1)$

The first case is clear. Consider the second case, since it is optimal to accept at $x-1$ for the second system, $R_i + v'_t(x-2) \geq v'_t(x-1)$. The third case is trivial, the LHS can be easily decreased by replacing the optimal action of the first system and the second case is attained. The last case is clear too, since the optimal action of the first system at state x is rejection, $v_t(x) \geq v_t(x-1) + R_i$. These results hold for each operator T_i and any convex combination of them satisfies the inequality. This completes the proof of part *ii* $\square$

4.2.2 The Robust Discrete-Time Revenue Management Problem with Arrival Uncertainty

In this section, we focus on a robust formulation that takes into account arrival uncertainty. Let us assume that the arrival probabilities -which may depend on x- $\lambda_{i,t}(x)$ are not known with certainty but are estimated to lie in some uncertainty set $\mathcal{P}_t$ in each period t. The uncertainty set may include additional constraints representing sample information in addition to the default constraints $\lambda_{i,t}(x) \geq 0$ for all i and t and $\sum_{i=1}^{n+1} \lambda_{i,t}(x) = 1$ for all t.

In order to model decision making under such an uncertainty, we employ the max-min formulation and formulate a robust dynamic program. The robust dynamic programming framework with transition uncertainty was established by Nilim and El Ghaoui [25] and Iyengar [16] and a revenue management example is studied in Birbil et al [7]. Under the max-min robust formulation, the controller plays a game against nature. It is assumed that nature selects the worst possible probability distribution from the uncertainty set in each state and time after observing the controller's action. To achieve this, we let nature choose an independent arrival vector for each state, time and action as in [25].

Let us now define the action space of the problem. Let a_i denote the action corresponding to class-i where $a_i = 1$ $(= 0)$ corresponds to admission (rejection). $\boldsymbol{a} = (a_1, a_2, ...)$ denotes the combined action vector and $\mathcal{A}$ denotes the set of admissible actions (combinations of accept/reject decisions for each class) of the controller. We also have to redefine the arrival probabilities as $\lambda_{i,t}(x, \mathbf{a})$ which denotes the probability of a class-i arrival at time t when the system is in state x and takes action $\mathbf{a}$. Let an arrival probability vector be $\boldsymbol{\lambda}_t(x, \boldsymbol{a}) = (\lambda_{1,t}(x, \boldsymbol{a}), ..., \lambda_{n+1,t}(x, \boldsymbol{a}))$. We assume that $\boldsymbol{\lambda}_t(x, \mathbf{a})$ belongs to an uncertainty set which does not depend on state x and action $\boldsymbol{a}$. This appears to be a reasonable assumption in the revenue management context.

We define two types of interval uncertainty models here, where one is a subset of the other. Let us define $\mathcal{P}_t \neq \emptyset$ an interval uncertainty set for the demand arrival vector:

$$\mathcal{P}_t = \left\{ \boldsymbol{y} = (y_1, \ldots, y_{n+1}) : \quad 0 \le \underline{y}_{i,t} \le y_{i,t} \le \bar{y}_{i,t},\ 0 \le q_t \le \sum_{i=1}^{n} y_{i,t} \le 1 \right\},$$

where $\underline{y}_{i,t}$ and $\bar{y}_{i,t}$ are respectively lower and upper bounds on the arrival probability and q is a lower bound on the minimum total probability of demand arrival.

$\mathcal{P}_t$ is a fairly standard interval uncertainty set for a probability vector. However, some of the results we present in this chapter can be extended to a modified uncertainty set $\mathcal{C}_t \subseteq \mathcal{P}_t$ where $\mathcal{C}_t \neq \emptyset$ is defined as follows:

$$\mathcal{C}_t = \left\{ \boldsymbol{y} = (y_1, \ldots, y_{n+1}) : \quad 0 \le \underline{y}_{i,t} \le y_{i,t} \le \bar{y}_{i,t},\ \sum_{i=1}^{n} b_i y_{i,t} \ge Q_t,\ 0 \le q_t \le \sum_{i=1}^{n} y_{i,t} \le 1 \right\},$$

where Q is a lower bound on a linear combination of the decision variables y_i. In particular, using this constraint and taking $b_i = R_i$ one can bound the expected reward per stage which is useful for revenue management applications.

Given the uncertainty set $\mathcal{P}_t$, the robust value function, for $x > 0$, is given by:

$$w_t(x) = \max_{\boldsymbol{a} \in \mathcal{A}} \min_{\boldsymbol{\lambda}_t(x,\boldsymbol{a}) \in \mathcal{P}_t} \{\sum_{i=1}^{n+1} \lambda_{i,t}(x, \boldsymbol{a}) (a_i R_i + w_{t+1}(x - a_i))\}, \tag{4.4}$$

where we set $a_{n+1} = 0$ and take the boundary conditions $w_T(x) = 0$ for all x and $w_t(0) = 0$ for all t.

Let us also define the dual version of the value function where the min and max are interchanged:

$$\widetilde{w}_t(x) = \min_{\boldsymbol{\lambda}_t(x,\boldsymbol{a}) \in \mathcal{P}_t} \max_{\boldsymbol{a} \in \mathcal{A}} \{\sum_{i=1}^{n+1} \lambda_{i,t}(x, \boldsymbol{a}) (a_i R_i + \widetilde{w}_{t+1}(x - a_i))\}. \tag{4.5}$$

Note that by the well-known property of the maximin theorem $w_t(x) \leq \widetilde{w}_t(x)$.

4.2.3 Structural Properties of the Robust Problem

In this subsection, we characterize the solution of Nature's problem at a particular stage t. In particular, we establish that Nature's optimal solution does not depend on the state x. To this end, we first investigate duality and its effects. Let us note that Nilim and El Ghaoui [25] show that the problem can be solved by a Bellman recursion as expressed in Equation (4.4)and investigate some duality properties of this solution. Our problem is a modified version of theirs. Hence, we establish a useful duality property based on their approach. In particular perfect duality implies that the objective function value does not change if the sequence of game is changed, i.e. if the controller is allowed to choose after observing Nature in our problem. This has been established for multistage stochastic games by various authors (Nilim and El Ghaoui [25], Nowak [26], and Altman [1]). Indeed, this seems to be a consequence of the Neumann minimax theorem as explained in Sion [30]. However, we present a stronger result than perfect duality in Proposition 4.2.1 where we also establish that the optimal action of the controller is independent of the Nature's posteriori action.

Proposition 4.2.1 *The optimal expected revenue at stage t in state x does not depend on the sequence of the game: $w_t(x) = \widetilde{w}_t(x)$ where $w_t(x)$ and $\widetilde{w}_t(x)$ are defined respectively in (4.4) and (4.5). Moreover, the optimal action $\boldsymbol{a}^*$ is independent of the Nature's posteriori* *decision. Hence, the robust value function can be represented as follows:*

$$\begin{aligned} w_t(x) &= \min_{\boldsymbol{\lambda}_t(x)\in\mathcal{P}_t}\{\sum_{i=1}^{n}\lambda_{i,t}(x)\max\{R_i+w_{t+1}(x-1),w_{t+1}(x)\}\} \\ &\quad +\lambda_{n+1,t}(x)w_{t+1}(x), \end{aligned} \tag{4.6}$$

or alternatively as:

$$w_t(x) = \min_{\boldsymbol{\lambda}_t(x)\in\mathcal{P}_t}\{\sum_{i=1}^{n}\lambda_{i,t}(x)\,(R_i-\Delta w_{t+1}(x))^+ + w_{t+1}(x)\}. \tag{4.7}$$

Proof:

Let us reconsider the nominal problem and define the $w_t^{\boldsymbol{\lambda}_t}(x)$ as in the following equation assuming that the problem is perfectly dual at stage $t+1$:

$$w_t^{\boldsymbol{\lambda_t}}(x) = \max_{\boldsymbol{a}\in\mathcal{A}}\{\sum_{i=1}^{n+1}\lambda_{i,t}(x,\boldsymbol{a})\,(a_iR_i+w_{t+1}(x-a_i))\}.$$

for all $\boldsymbol{\lambda}_t \in \mathcal{P}_t$. Apparently, the optimal action $\boldsymbol{a}$ is the same for every $\boldsymbol{\lambda}_t \in \mathcal{P}_t$, i.e. $a_i = 1$ if $R_i + w_{t+1}(x-a_i) \geq 0$ else $a_1 = 0$.

Now we define $\boldsymbol{\lambda}_t^*(x)$ follows:

$$\boldsymbol{\lambda}_t^*(x) = \arg\min_{\boldsymbol{\lambda}_t(x)\in\mathcal{P}_t}\{\sum_{i=1}^{n}\lambda_{i,t}(x)\max\{R_i+w_{t+1}(x-1),w_{t+1}(x)\}+\lambda_{n+1,t}(x)w_{t+1}(x)\}. \tag{4.8}$$

Therefore, the pair $(\boldsymbol{\lambda}_t^*(x),\boldsymbol{a}^*)$ is the equilibrium of the problem $\widetilde{w}_t(x)$ where $\boldsymbol{a}^*$ denotes the optimal action that admits customers according to Equation 4.1.

Now consider the original form of the game, i.e. robust value function and let the controller choose $\boldsymbol{a}^*$. As already given in Equation 4.8 Nature's best response to $\boldsymbol{a}^*$ is $\boldsymbol{\lambda}_t^*(x)$. Because $w_t(x) \leq \widetilde{w}_t(x)$, $(\boldsymbol{\lambda}_t^*(x),\boldsymbol{a}^*)$ is the optimal solution of 4.4. This completes the proof. □

In general, in a robust stochastic dynamic program, the worst case probability distribution may be state dependent. While this is not a major computational problem, it is an issue when structural optimal policies are desired. Next, we use representation (4.7) to establish that Nature's solution is not state dependent for our problem.

Theorem 4.2.2 *Consider the uncertainty set $\mathcal{P}_t$, then Nature's optimal choice of probability distribution can be obtained by a simple rule and is identical for all states at all times: $\boldsymbol{\lambda}_t(x) = \boldsymbol{\lambda}_t\ \forall\ x$.*

Proof:

Consider Nature's problem for a given x and t, which is a Linear Program with decision variables $\boldsymbol{\lambda}_t(x)$ and objective function coefficients $(R_1 - \Delta w_t(x))^+$, $(R_2 - \Delta w_t(x))^+$, ..., $(R_n - \Delta w_t(x))^+$. Since $R_1 \geq R_2 \geq ... \geq R_n$, $(R_1 - \Delta w_t(x))^+ \geq (R_2 - \Delta w_t(x))^+ \geq ... \geq (R_n - \Delta w_t(x))^+$ for all t, x. Please note that the problem can be also represented by the following equation through a transformation of the uncertainty set $\mathcal{P}_t$ to $\Delta\mathcal{P}_t$.

$$\Delta\mathcal{P}_t = \left\{ \boldsymbol{\Delta y} = (\Delta y_1, \ldots, \Delta y_{n+1}) : \quad 0 \ \leq \Delta y_{i,t} \ \leq \bar{y}_{i,t} - \underline{y}_{i,t}, \ 0 \ \leq \ q - \sum_{i=1}^{n} \underline{y}_{i,t} \ \leq \ \sum_{i=1}^{n} \Delta y_i \ \leq \ 1 \right\},$$

$$\begin{aligned} w_t(x) \quad &= \quad \sum_{i=1}^{n} \underline{\lambda}_{i,t}(x) \left(R_i - \Delta w_{t+1}(x)\right)^+ \\ &\quad + \min_{\boldsymbol{\Delta\lambda}_t(x) \in \Delta\mathcal{P}_t} \{ \sum_{i=1}^{n} \Delta\lambda_{i,t}(x) \left(R_i - \Delta w_{t+1}(x)\right)^+ \} + w_{t+1}(x). \end{aligned}$$

The minimization term corresponds to a continuous Knapsack Problem with upper bounds (with decision variables $\Delta\lambda_{i,t}(x)$) . In addition, the objective function coefficients $(R_i - \Delta w_{t+1}(x))^+$ are decreasing in i since $R_i > R_j$ if $i < j$ for any given state x. The optimal solution is then given by the following allocation where k denotes a class between $1, ..., n$:

$$\begin{aligned} \Delta\lambda_{i,t}(x) &= 0 \ \text{ if } \ 1 \leq i < k \\ \Delta\lambda_{k,t}(x) &= q_t - \sum_{i=k+1}^{n} \Delta\lambda_{i,t}(x) \\ \Delta\lambda_{i,t}(x) &= \bar{\lambda}_{i,t} - \underline{\lambda}_{i,t} \ \text{ if } \ k < i. \end{aligned}$$

The dual of the problem is as follows:

$$\begin{aligned} &\max_{z,u \geq 0} q'_t z - w \\ st. \ z - w_i &\leq \left(R_i - \Delta w_{t+1}(x)\right)^+. \end{aligned}$$

where $q' = q - \sum_{i=1}^{n} \Delta\bar{\lambda}_{i,t}$, let's take $z* = (R_k - \Delta w_{t+1}(x))^+$ and

$$\begin{aligned} w_i &= 0 \ \text{ if;} \quad 1 < i \leq k \\ w_i = (R_k - \Delta w_{t+1}(x))^+ &- (R_i - \Delta w_{t+1}(x))^+ \text{ if } \ k < i \leq n. \end{aligned}$$

The optimal solution is:

$$q_t' \left(R_k - \Delta w_{t+1}(x)\right)^+ - \left[(R_k - \Delta w_{t+1}(x))^+ - (R_{k+1} - \Delta w_{t+1}(x))^+\right](\bar{\lambda}_{k+1,t} - \underline{\lambda}_{k+1,t}) - ... \\ \left[(R_k - \Delta w_{t+1}(x))^+ - (R_n - \Delta w_{t+1}(x))^+\right](\bar{\lambda}_{n,t} - \underline{\lambda}_{n,t}).$$

Clearly the solution is dual feasible, has the same objective value with the primal problem and satisfies complementary slackness. The optimal solution clearly does not depend on x for all t. This completes the proof. □

We have established that Nature's solution is identical for all states x for any stage t. Moreover, if the uncertainty set is not time dependent, i.e. $\mathcal{P}_t = \mathcal{P}\ \forall\ t$, then nature's optimal choice of probability distribution is identical for all states at all times: $\boldsymbol{\lambda}_t(x) = \boldsymbol{\lambda}$ $\forall\ x,\ t$.

Corollary 4.2.1 *Consider the uncertainty set $\mathcal{C}_t$, if $b_i \leq b_j$ for all $i \leq j$ then Nature's optimal choice of probability distribution can be obtained by a simple rule and is identical for all states at all times: $\boldsymbol{\lambda}_t(x) = \boldsymbol{\lambda}_t\ \forall\ x$.*

Proof:

Please remember the definition of $\Delta\mathcal{C}_t$:

$$\Delta\mathcal{C}_t = \left\{\boldsymbol{\Delta y} = (\Delta y_1, \ldots, \Delta y_{n+1}) :\ 0 \ \leq \Delta y_{i,t} \ \leq \bar{y}_{i,t} - \underline{y}_{i,t},\ \sum_{i=1}^{n} b_i \Delta y_i \geq Q - \sum_{i=1}^{n} b_i \Delta y_i,\ 0 \ \leq \ q - \sum_{i=1}^{n} \underline{y}_{i,t} \ \leq \ \sum_{i=1}^{n} \Delta y_i \ \leq \ 1\right\}.$$

Please note that $\sum_{i=0}^{n} b^i \lambda_{i,t}(x) = Q$, where $b_1 \ \leq \ b_2 \ \leq \ ... \ \leq \ b_n$, it is easy to conclude that:

$$\frac{(R_1 - \Delta w_{t+1}(x))^+}{b^1} \geq \frac{(R_2 - \Delta w_{t+1}(x))^+}{b^2} ... \geq \frac{(R_n - \Delta w_{t+1}(x))^+}{b^n}.$$

for all t, x. Since $\mathcal{C}_t \subseteq \mathcal{P}_t$ the resultant optimal solution is given as:

$$\Delta\lambda_{i,t}(x) = 0 \text{ if } 1 \leq i < k$$
$$\Delta\lambda_{k,t}(x) = \frac{\Delta Q_t - \sum_{i=k+1}^{n} b_i \Delta\lambda_{i,t}(x)}{b_k}$$
$$\Delta\lambda_{i,t}(x) = \bar{\lambda}_{i,t} - \underline{\lambda}_{i,t} \text{ if } k < i \leq n$$

.

Similar to Theorem 4.2.2 the problem is continuous Knapsack problem with upper bounds with the exception that the $q_t \leq \sum_{i=1}^{n} \lambda_{i,t}$ is the untight constraint. The dual of the problem together with the optimal values are as follows:

$$\max_{z,u \geq 0} q_t' z_1 + Q_t' z_2 - w$$
$$st. z_1 + b_i z_2 - w_i \leq R_i - \Delta w_{t+1}(x)^+.$$

where $q_t' = q_t - \sum_{i=1}^{n} \Delta\bar{\lambda}_{i,t}$ and $Q_t' = Q_t - \sum_{i=1}^{n} \Delta b_i \bar{\lambda}_{i,t}$. Let's take $z_1* = 0$, $z_2* = \frac{(R_k - \Delta w_{t+1}(x))^+}{b_k}$ and

$$w_i = 0 \text{ if } 1 < i \leq k$$
$$w_i = \frac{b_i}{b_k}(R_k - \Delta w_{t+1}(x))^+ - (R_i - \Delta w_{t+1}(x))^+ \text{ if } k < i \leq n.$$

The optimal solution is:

$$Q_t'(R_k - \Delta w_t(x))^+ - \left[\frac{b_{k+1}}{b_k}(R_k - \Delta w_{t+1}(x))^+ - (R_{k+1} - \Delta w_{t+1}(x))^+\right](\bar{\lambda}_{k+1,t} - \underline{\lambda}_{k+1,t}) - ...$$
$$\left[\frac{b_n}{b_k}(R_k - \Delta w_t(x))^+ - (R_n - \Delta w_t(x))^+\right](\bar{\lambda}_{n,t} - \underline{\lambda}_{n,t}).$$

Clearly the solution is dual feasible, has the same objective value with the primal problem and satisfies complementary slackness. The optimal solution clearly does not depend on x for all t. This completes the proof. □

As in Theorem 4.2.2, the optimal solution does not depend on x at all stages t. □

Corollary 4.2.2 *Consider the uncertainty set $\mathcal{C}_t$, if $b_i = R_i$ for all i then Nature's optimal choice of probability distribution is identical for all states at all stages:* $\boldsymbol{\lambda}_t(x) = \boldsymbol{\lambda}_t \ \forall\, x$.

Proof:

Please note that $\Delta w_t(x) \leq R_1$ and Class 1 customer is always accepted to the system. We have:

$$1 - \frac{\Delta w_t(x)}{R_1} \geq 1 - \frac{\Delta w_t(x)}{R_2} .... \geq 1 - \frac{\Delta w_t(x)}{R_n}.$$

Using a similar argument to the one in the Proof of Corollary 4.2.1 we find that Nature's optimal solution does not depend on x. This establishes the result. □

Under the conditions of Theorem 4.2.2, Nature's probability distribution cannot be state dependent. The controller is then playing a game against a state-independent arrival distribution which makes the problem a standard Markov Decision Processes as in Talluri and Van Ryzin [32] or Aydin et al. [2]. The next theorem establishes that all structural results of the nominal problem propagate to the robust counterpart.

Theorem 4.2.3 *Consider the uncertainty set $\mathcal{P}_t$, then the robust value has function has the following properties:*

1. *$w_t(x)$ is nondecreasing (ND) in x for all t,*

2. *$w_t(x)$ is concave in x for all t,*

3. *$w_t(x)$ is supermodular in x, t for all x, t.*

Proof:

Due to Theorem 4.2.2, nature's optimal policy does not depend on x. The controller's problem then becomes a Markov Decision Process with state indepedent demand arrival rates. The proof then follows from the results on the nominal problem given in Aydin, Akcay and Karaesmen [2]. □

Theorem 4.2.3 establishes that principal structural properties of the optimal polices also hold for the robust counterpart of the problem defined in Equation (4.4). This implies the optimality of threshold policies as in the nominal problem. Besides, supermodularity implies that the thresholds are also monotone over time.

Next we establish the structural results for the uncertainty set $\mathcal{C}$ under certain conditions.

Corollary 4.2.3 *Consider the uncertainty set $\mathcal{C}_t$, if $b_i \leq b_j$ for all $i \leq j$ or $b_i = R_i$ for all i, then the robust value function has the following properties:*

1. *$w_t(x)$ is nondecreasing (ND) in x for all t,*

2. *$w_t(x)$ is concave in x for all t,*

3. *$w_t(x)$ is supermodular in x, t for all x, t.*

Proof:

Due to Corollaries 4.2.1 and 4.2.2, nature's optimal policy does not depend on x. Once again, the proof then follows from the results on the nominal problem given in Aydin, Akcay and Karaesmen [2]. □

4.2.4 Behavior of the Optimal Policy for Nested Uncertainty Sets

In this section, we investigate the behavior of the optimal policy when the uncertainty set is relaxed. In particular, we consider nested uncertainty sets where a relaxation of ε leads to a larger uncertainty set that includes the first set $\mathcal{P}$ such that $\mathcal{P}_t \subseteq \mathcal{P}_t^\varepsilon$ for all t.

We first state the following property of Nature's optimal solutions:

Lemma 4.2.1 *Consider two problems that are identical in their parameters except their uncertainty sets that represent their arrival probabilities at stage t. Let $\mathcal{P}_t$ and $\mathcal{P}_t^\varepsilon$ be two uncertainty sets, $\boldsymbol{\lambda}_t$ and $\boldsymbol{\lambda}_t^\varepsilon$ be the corresponding Nature's optimal solutions respectively. If $\mathcal{P}_t \subseteq \mathcal{P}_t^\varepsilon$ then $\boldsymbol{\lambda}_t \succeq \boldsymbol{\lambda}_t^\varepsilon$.*

Proof:

According to definition of $\mathcal{P}_t$, $\mathcal{P}_t^\varepsilon \supseteq \mathcal{P}_t$ iff at least one of the conditions is true:

1. $q^\varepsilon \leq q$,

2. $\underline{y}_{i,t}^\varepsilon \leq \underline{y}_{i,t}$,

3. $\bar{y}^{\varepsilon}_{i,t} \geq \bar{y}_{i,t}$.

The corresponding solution of Nature is given in Theorem 4.2.2. For the first case the inequality is clear since $\sum_{i=1}^{n} \lambda^{\varepsilon}_{i,t} \geq \sum_{i=1}^{n} \lambda_{i,t}, \ \ \forall\, i$. For the second and third cases, solution assigns probabilities in increasing order of rewards, thereby implying higher probabilities to lower revenue classes for $\mathcal{P}^{\varepsilon}_{t}$. Hence,

$$\sum_{i=k}^{n} \lambda^{\varepsilon}_{i,t} \geq \sum_{i=k}^{n} \lambda_{i,t}, \ \ \forall\, k$$

which implies that $\boldsymbol{\lambda}_t \succeq \boldsymbol{\lambda}^{\varepsilon}_t$. □

Theorem 4.2.4 *Consider two problems that are identical in their parameters except their uncertainty sets that represent their arrival probabilities at stage t. Let $\mathcal{P}_t$ and $\mathcal{P}^{\varepsilon}_t$ be two uncertainty sets, $w_t(x)$ and $w^{\varepsilon}_t(x)$ be the corresponding value functions respectively. If $\mathcal{P}_t \subseteq \mathcal{P}^{\varepsilon}_t$ then:*

1. *$w_t(x) \geq w^{\varepsilon}_t(x)$ for all x, t,*

2. *$\Delta w_t(x) \geq \Delta w^{\varepsilon}_t(x)$ for all x, t.*

Proof:

1: This follows by Lemma 4.2.1 and the first part of Theorem 4.2.1. □

2: This follows from Lemma 4.2.1 and the second part of Theorem 4.2.1 □

Theorem 4.2.4 has important implications. The first part implies that the expected revenue of the controller is monotone in nested interval uncertainty sets. The second part is less obvious but implies that optimal admission policies are also monotone for nested uncertainty sets. If a class-i demand is admitted to the original system at a given state x under the uncertainty set $\mathcal{P}$, then it is always admitted at x under $\mathcal{P}^{\varepsilon}$. Similarly, a class-i demand that is rejected at x under $\mathcal{P}^{\varepsilon}$, is also rejected at x under $\mathcal{P}$. In short, the optimal admission thresholds for $\mathcal{P}$ and $\mathcal{P}^{\varepsilon}$ are ordered for all i and all t.

4.2.5 Nested Uncertainty Sets and a Weighted Optimization Approach

In this section, we investigate the structure of optimal policies under a more general robust dynamic programming formulation. The so-called S-Robust Policy framework was proposed by Xu and Mannor [34] who propose a weighted optimization approach between multiple uncertainty sets that have a nested structure. In particular, in this approach, it is assumed that the transition probability vector belongs to a concentration set $\mathcal{P}_t^{\mho} \subseteq \mathcal{P}_t$ with probability δ and it belongs to the larger set $\mathcal{P}_t$ with probability of 1, for all t. Here, the concentration set $\mathcal{P}_t^{\mho}$ can be viewed as a prior distribution and δ is a measure of reliance on that distribution. The concentration set weights could be represented as a vector $\boldsymbol{\delta} = (\delta_1, \delta_2, \delta_3, ..., \delta_T)$ if δ is allowed to vary between stages.

Xu and Mannor define an S-robust policy as the outcome of the following equation of optimality:

$$\begin{aligned} w_t(x) \quad = \quad & \max_{a \epsilon A} \left\{ \delta_t \min_{\boldsymbol{\lambda}_t^{\mho}(x,\boldsymbol{a}) \in \mathcal{P}_t^{\mho}} \left[\sum_i \lambda_{i,t}^{\mho}(x) \left(a_i R_i + w_{t+1}(x - a_i) \right) \right] \right. \\ & \left. + (1-\delta_t) \min_{\boldsymbol{\lambda}_t(x,\boldsymbol{a}) \in \mathcal{P}_t} \left[\sum_i \lambda_{i,t}(x) \left(a_i R_i + w_{t+1}(x - a_i) \right) \right] \right\}. \end{aligned}$$

By Proposition 4.2.1 we can rewrite the above as:

$$\begin{aligned} w_t(x) \quad = \quad & \delta_t \min_{\boldsymbol{\lambda}_t^{\mho}(x) \in \mathcal{P}_t^{\mho}} \left[\sum_i \lambda_{i,t}^{\mho}(x) (R_i - \Delta w_{t+1}(x))^+ \right] \\ & + (1-\delta_t) \min_{\boldsymbol{\lambda}_t(x) \in \mathcal{P}_t} \left[\sum_i \lambda_{i,t}(x) (R_i - \Delta w_{t+1}(x))^+ \right] + w_{t+1}(x). \qquad (4.9) \end{aligned}$$

The next corollary establishes that robust value functions are monotone with respect to the reliance weight vector $\boldsymbol{\delta}$.

Corollary 4.2.4 *Consider two problems that are identical in their parameters except their reliance weight factors at stage t. Let δ^1 and δ^2 be two reliance weight factors, $w_t^1(x)$ and $w_t^2(x)$ be the corresponding value functions respectively. If ${\delta^1}_t \geq {\delta^2}_t$ for all t:*

1. *$w_t^1(x) \;\geq\; w_t^2(x)$ for all t, x,*

2. $\Delta w_t^1(x) \geq \Delta w_t^2(x)$ *for all* $t,\ x$.

Proof:

First by Theorem 4.2.4 $w_t^{\mho}(x) \geq w_t(x)$ and $\Delta w_t^{\mho}(x) \geq \Delta w_t(x)$. It is then straightforward to show that for any two if ${\delta^1}_t \geq {\delta^2}_t$ both $w_t^1(x) \geq w_t^2(x)$ and $\Delta w_t^1(x) \geq \Delta w_t^2(x)$ for stages $t, ..., T$. Once again by Theorem 4.2.4 the inequalities also hold for stages $1, ..., t-1$. This completes the proof. □

Clearly, as $\boldsymbol{\delta}$ increases, the weight of the concentration set increases and the resulting policy becomes less robust and therefore less conservative. Corollary 4.2.4 implies that the optimal admission thresholds are non-decreasing in $\boldsymbol{\delta}$ for all customer classes. As the decision maker assigns more weight to the concentration set, she chooses to be more selective in customer admission at any given state x and time t.

4.3 Dynamic Single-Product Revenue Management with Replenishment

4.3.1 Problem:Dynamic Single-Product Revenue Management with Replenishment

In this section, we consider the revenue management of Section 4.2 but we allow replenishments. The model we explore is an example of a class of inventory problems in stock rationing. In order to maintain consistency with respect to Section 4.2, we consider the discrete version of the stock rationing problem for a multiple demand class $M/M/1$ make-to-stock queue with lost sales that was previously modeled and explore by Ha [14]. This is a model that has received significant attention and was later extended in several directions.

The discrete-time version of Ha's model [14] turns out to be very similar to the model in the previous section. The original continuous-time version has n classes of customers whose demands arrive according to independent Poisson processes with rate λ_i $(i = 1, 2, ..., n)$. A single server whose processing time is exponentially distributed with rate μ produces items one-by-one. If a demand of class-i is admitted when there is at least one unit of inventory on hand, it is immediately satisfied and a class-dependent instant reward of R_i is obtained. If inventory is empty, all arriving demands are assumed to be lost. As before, the classes are ordered such that if $i < j$ then $R_i > R_j$. At any time t, the inventory level is denoted by $X(t)$ (where $X(t) \in \mathcal{Z}^+$) and the inventory holding cost rate is $h(X(t))$ per unit of time. The holding cost function $h(x)$ is increasing and convex in x. Ha [14] considers an infinite-horizon discounted profit maximization objective with a discount rate of α.

The production can be stopped and started at any time t and the demands are admitted or rejected from the system upon arrival. Ha [14] shows that, after uniformization, the equivalent discrete time problem can be expressed as follows. Let $\gamma = \bar{\mu} + \sum_{i=1}^{n} \bar{\lambda_i} + \bar{\alpha}$ be the uniformization rate which can be set to 1 without loss of generality and by setting $\mu = \bar{\mu}/\gamma$, $\lambda_i = \bar{\lambda}_i/\gamma$ and $\alpha = \bar{\alpha}/\gamma$, we obtain:

$$\begin{aligned} v_t(x) \quad = \quad & \mu \max\{v_{t+1}(x+1), v_{t+1}(x)\} \\ & + \sum_{i=1}^{n} \lambda_i \max\{v_{t+1}(x-1) + R_i, v_{t+1}(x)\} - h(x) \;\; \text{if } x > 0 \end{aligned} \tag{4.10}$$

and

$$v_t(0) = \mu \max\{v_{t+1}(1), v_{t+1}(0)\} + \sum_{i=1}^{n} \lambda_i v_{t+1}(0) - h(0), \quad \text{if } x = 0 \tag{4.11}$$

.

Let $\alpha' = (1 - \alpha)$, Equation (4.11) can alternatively be represented as follows:

$$\begin{aligned} v_t(x) \quad = \quad & \mu(\Delta v_{t+1}(x+1))^+ \\ & + \sum_{i=1}^{n} \lambda_i \left(R_i - \Delta v_{t+1}(x)\right)^+ + \alpha' v_{t+1}(x) - h(x), \;\; \text{if } x > 0, \end{aligned} \tag{4.12}$$

with the boundary condition $v_T(x) = 0$ for all x. Ha [14] established that the value function $v_t(x)$ is concave for all finite t and for the infinite horizon value function as t tends to infinity. This implies that the optimal demand admission policy is of threshold type. In addition, the optimal production policy is of target level type. There is a target production level below which the system should produce and at or above which the system should stop production. These results can be extended to a number of more complicated cases including batch arrivals (Huang and Iravani [15] and Çil et al. [8]).

4.3.2 The Robust Discrete Time Revenue Management Problem with Arrival Uncertainty

In this section, we consider the discrete-time model in Section 4.3.1 but assume that the arrival and production rates are not known with certainty. We first consider a model of interval uncertainty for the finite horizon case where each rate parameter is estimated independently and is assumed to lie in an interval between upper and lower bounds rather than taking a specific value.

Let the arrival and production probability vector be: $(\boldsymbol{\lambda}_t(x,\boldsymbol{a}), \mu_t(x,\boldsymbol{a})) = (\lambda_{1,t}(x,\boldsymbol{a}), ..., \lambda_{n,t}(x,\boldsymbol{a}), \mu_t(x,$ We assume again that this vector belongs to an uncertainty set which does not depend on the state x and the action $\boldsymbol{a}$.

Let us define $\mathcal{P} \neq \emptyset$ an interval uncertainty set for the demand arrival - production probability vector:

$$\mathcal{P} = \left\{ \boldsymbol{z} = (z_1, \ldots, z_{n+1}) : \quad 0 \le \underline{z}_i \le z_i \le \bar{z}_i,\ 0 \le q \le \sum_{i=1}^{n} z_i \le \alpha' \right\}.$$

where $\underline{z}_i$ and $\bar{z}_i$ upper and lower bounds on individual event probabilities and q is a lower bound on the total probability of arrival and production.

Let $w_t(x)$ be the robust value function. Similarly to the previous section, the optimality equation for $w_t(x)$ can be expressed as:

$$\begin{aligned} w_t(x) &= \min_{(\boldsymbol{\lambda_t}(x),\mu_t(x))\in\mathcal{P}} \Big\{ \mu_t(x)\,(\Delta w_{t+1}(x+1))^+ \\ &\quad + \sum_{i=1}^{n} \lambda_{i,t}(x)\,(R_i - \Delta w_{t+1}(x))^+ + \alpha' w_{t+1}(x) - h(x) \Big\}, \quad \text{if } x > 0 \qquad (4.13) \\ w_t(0) &= \min_{(\boldsymbol{\lambda}_t(0),\mu_t(0))\in\mathcal{P}} \{\mu_t(0)\,(\Delta w_{t+1}(1))^+\}. \qquad (4.14) \end{aligned}$$

4.3.3 Structural Properties of the Robust Problem

In the below, we show that the results given for the single-resource revenue management problem of Section 4.2.3 can be extended to the case with replenishment. First, please note that Proposition 4.2.1 applies for the above problem with an identical argument. Hence, the optimal action of the controller at any stage t and state x is independent of the Nature's posteriori decision. Next, we establish the concavity of the robust value function.

Theorem 4.3.1 *Consider the uncertainty set $\mathcal{P}$, then the robust value function $w_t(x)$ is concave in x for all t.*

Proof:

According to Proposition 4.2.1, the optimal actions of the controller do not depend on the choice of nature. Suppose that $\boldsymbol{\lambda}_t(x)$ and $\mu_t(x)$ are the optimal solutions of the Nature for state x. We use an induction argument and assume that $w_{t+1}(x)$ is concave in x. Next, we have to show that under this assumption $w_t(x)$ preserves concavity. Using the concavity assumption, the following inequality holds if the arrival and rates are identical at states $x-1$, x and $x+1$ and are equal to $\boldsymbol{\lambda}_t(x)$ and $\mu_t(x)$ using the existing results (Ha [14], Çil et al. [8]):

$$\mu_t(x)\{\Delta w_{t+1}(x+1)\}^+ + \textstyle\sum_{i=1}^n \lambda_{i,t}(x)\{R_i - \Delta w_{t+1}(x)\}^+ + \alpha' w_{t+1}(x) - h(x) \geq$$
$$1/2\{\mu_t(x)\{\Delta w_{t+1}(x)\}^+ + \textstyle\sum_{i=1}^n \lambda_{i,t}(x)\{R_i - \Delta w_{t+1}(x-1)\}^+ + \alpha' w_{t+1}(x-1) - h(x-1)\}$$
$$+ 1/2\{\mu_t(x)\{\Delta w_{t+1}(x+2)\}^+ + \textstyle\sum_{i=1}^n \lambda_{i,t}(x)\{R_i - \Delta w_{t+1}(x+1)\}^+ + \alpha' w_{t+1}(x+1) - h(x+1)\}.$$

Now let us relax the assumption that the arrival and production rates are equal for all states. Because Nature's objective is to minimize the robust value function $w_t(x)$, the following holds:

$$\mu_t(x)\{\Delta w_{t+1}(x+1)\}^+ + \textstyle\sum_{i=1}^n \lambda_{i,t}(x)\{R_i - \Delta w_{t+1}(x)\}^+ + \alpha' w_{t+1}(x) - h(x) \geq$$
$$1/2\{\mu_t(x-1)\{\Delta w_{t+1}(x)\}^+ + \textstyle\sum_{i=1}^n \lambda_{i,t}(x-1)\{R_i - \Delta w_{t+1}(x-1)\}^+ + \alpha' w_{t+1}(x-1) - h(x-1)\}$$
$$+ 1/2\{\mu_t(x+1)\{\Delta w_{t+1}(x+2)\}^+ + \textstyle\sum_{i=1}^n \lambda_{i,t}(x+1)\{R_i - \Delta w_{t+1}(x+1)\}^+ + \alpha' w_{t+1}(x+1) - h(x+1)\}.$$

This establishes that $w_t(x)$ is concave in x. □

Theorem 4.3.1 implies that, as in the revenue management problem of Section 4.2 or the nominal problem of Ha [14], the optimal demand admission policy of the robust problem is of threshold type. Hence, in each period t and for each class-i, there is an admission threshold $l_{i,t}$. Similarly, there is a target level S_t for each period t, such that the controller stops production if the inventory on hand reaches this level. However, this time there are no corresponding results for supermodularity/submodularity of the value function $w_t(x)$ in

x, t or the monotonicity of thresholds over time.

It is important to note that if the uncertainty in event probabilities pertains to only one type of operator, i.e. either admission operators only or the production operator only, Nature's solution is independent of the state x for all stages t. In this case, although there may be alternative optimal solutions, the optimal policy of Nature can be reduced to a unique solution that has the identical profit.

4.3.4 Behavior of the Optimal Policy for Nested Uncertainty Sets

In this subsection, we explore the effects of increasing or decreasing uncertainty on the optimal robust value function and the optimal policy. The results we provided for the revenue management problem of Section 4.2 also hold for this problem under certain additional conditions.

Corollary 4.3.1 *Consider two problems that are identical in their parameters except their uncertainty sets that represent their arrival probabilities at stage t. Let $\mathcal{P}_t$ and $\mathcal{P}_t^{\varepsilon}$ be two uncertainty sets, $w_t(x)$ and $w_t^{\varepsilon}(x)$ be the corresponding value functions respectively. If $\mathcal{P}_t \subseteq \mathcal{P}_t^{\varepsilon}$ then:*

1. *$w_t(x) \geq w_t^{\varepsilon}(x)$ for all t, x,*
2. *(a) If only the arrival probabilities are uncertain, then $\Delta w_t(x) \geq \Delta w_t^{\varepsilon}(x)$ for all t, x,*
   (b) If only the production rate is uncertain, then $\Delta w_t(x) \leq \Delta w_t^{\varepsilon}(x)$ for all t, x.

Proof:

a- The beginning of this part is the same as above. For the second phase we need to show that production operator also preserves the inequality. Since $\Delta v_{t-1}(x) \geq \Delta v'_{t-1}(x)$ if it is not optimal to produce in the original system then it is also not optimal to produce in the perturbed system. By concavity we know that the base stock policy is optimal therefore the cases except from the produce at all of the conditions and not produce at all of the conditions the cases are as follows:

$T^i w_{t-1}(x)$	$-T^i w'_{t-1}(x)$	$\geq$	$T^i w_{t-1}(x-1)$	$-T^i w'_{t-1}(x-1)$	Result
(NP)	(NP)		(P)	(P)	$w_t(x) - w'_t(x) \geq$ $w_t(x) - w'_t(x)$
(P)	(NP)		(P)	(P)	$w_t(x+1) - w'_t(x) \geq$ $w_t(x) - w'_t(x)$
(NP)	(NP)		(P)	(NP)	$w_t(x) - w'_t(x) \geq$ $w_t(x) - w'_t(x-1)$
(P)	(NP)		(P)	(NP)	$w_t(x+1) - w'_t(x) \geq$ $w_t(x) - w'_t(x-1)$

The first case is clear. In the second case it is optimal to make a production in the original system therefore $w_t(x+1) \geq w_t(x)$. The third case is similar too, since it is not optimal to produce at the second system we have $w'_t(x) \leq w'_t(x-1)$. For the last case $w_t(x+1) - w'_t(x) \geq w_t(x) - w'_t(x)$ and $w_t(x) - w'_t(x) \geq w_t(x) - w'_t(x-1)$. This completes the proof of 2.a. □.
(2) For that case it is obvious that $\mu_t(x) \geq \mu_t^{\varepsilon}(x)$. Since $w_t(x) \leq w_t(x-1)$ loss at state x is less than loss at $x-1$ for all states. Rest of the proof is the same with the above. □.

The two properties of Corollary 4.3.1 have the following implications. As we enlarge the uncertainty set, the optimal robust value function decreases. Besides, if arrival rates are uncertain but the production rate is fixed, enlarging the uncertainty set part of the uncertainty set leads to lower optimal admission thresholds. Likewise, optimal production target levels decrease when the uncertainty set is enlarged. Finally, if the arrival rates are fixed, as the uncertainty set representing the production rates is enlarged, both the optimal admission thresholds and the production target levels increase.

Next, we focus on the S-robust formulation of the problem and investigate the structural properties under this formulation.

Corollary 4.3.2 *Consider two problems that are identical in their parameters except their reliance weight factors at stage t. Let δ^1 and δ^2 be two reliance weight factors, $w_t^1(x)$ and $w_t^2(x)$ be the corresponding value functions respectively. If ${\delta^1}_t \geq {\delta^2}_t$ then: t:*

1. *$w_t^1(x) \geq w_t^2(x)$ for all t, x,*

2. *(a) If only the arrival probabilities are uncertain, then $\Delta w_t^1(x) \geq \Delta w_t^2(x)$ for all*

Customer Class	Reward	Nominal Arrival Probability	Interval
1	$80 per item	0.075	(0.05, 0.10)
2	$35 per item	0.075	(0.05, 0.10)
3	$25 per item	0.15	(0.10, 0.20)

Table 4.1: Demand and Reward Parameters for the Numerical Example

t, x,

(b) If only the production rate is uncertain, $\Delta w_t^1(x) \leq \Delta w_t^2(x)$ for all t, x.

Proof:

The results follows from Corollary 4.3.1 by using the results of Corollary 4.2.4. □

Finally, let us briefly discuss the infinite horizon extension. Iyengar [16] and Nilim and El Ghaoui [25] establish that the respective controller and nature policies are stationary for the infinite horizon problem. Moreover, Nilim and El Ghaoui [25] show that the optimal value function of the infinite horizon problem with a discounted cost function can be obtained as the unique limit of the finite horizon problem. This suggests that the optimal policy structure can be extended to the infinite horizon case.

4.4 Numerical Results

In this section, we present some numerical results for the make-to-stock queue with multiple demand classes introduced in Section 4.3. Let us a consider system consisting of three customer classes. The holding cost is assumed to be $5 per item per year (approximately $0.0142 per item per day). The (daily) production probability is 0.2 and is certain. The (daily) demand arrival probabilities are assumed to be uncertain. In particular, we assume that there is best guess for the demand probability which we label as the nominal probability and an interval around this nominal probability. This data as well as the rewards of each class are presented in Table 4.1.

We experiment with the S-Robust Policy which includes the nominal policy and the pure robust policy as special cases. The nominal probabilities are taken as the concentration

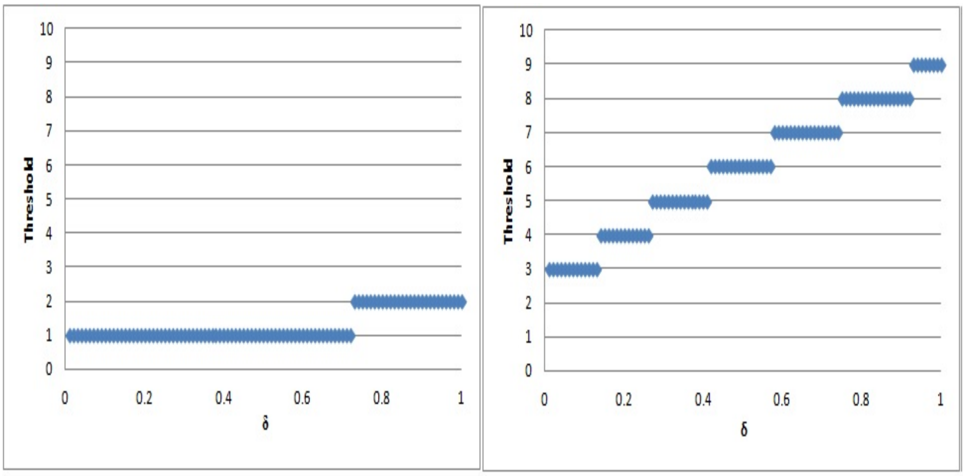

Figure 4.1: Optimal Admission Thresholds of Customer Classes 2 and 3 as a function of δ

set and the optimal policies are obtained for different δ values where δ reflects the weight of the concentration set. Hence, $\delta = 1.0$ designates the nominal solution whereas $\delta = 0$ designates the pure robust solution. We solve the problem for different δ values between $[0, 1]$ and compute the optimal $S-robust$ policy for different values of δ. Then we simulate the performance of these policies for demand data that is sampled from the uncertainty set. In particular, we generate the arrival probabilities to lie in their associated intervals uniformly, consequently with a mean equal to the nominal arrival probability.

In Figures 4.4 to 4.4 we present the long run results (using a 1 million stages) of the problem. In Figure 4.4 and Figure 4.4 the admission thresholds of Classes 2 and 3 and the target levels are depicted as a function of δ. Obviously, customer class 1 is the preferred customer in this problem and is always accepted to the system. Clearly, the admission thresholds increase as reliance on the nominal distribution increases as established in Corollary 4.3.2. Similarly, optimal target levels also increase as reliance on the nominal distribution increases. Therefore, at any given inventory level, the controller becomes less willing to sell and more willing to produce when δ increases. Robustness in this problem requires setting lower thresholds and lower target inventory levels.
Figure 4.4 depicts the average profit per stage as a function of δ. To better understand how the average profit increases in δ, we next investigate the fill rates (demand satisfaction

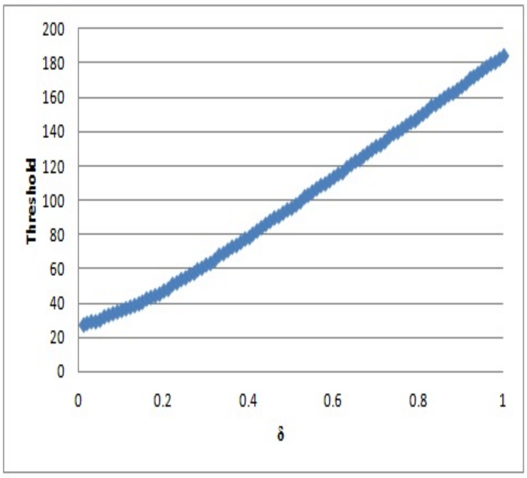

Figure 4.2: Optimal Base Stock Levels as a function of δ

probabilities) for each class as a function of δ. Apparently, increasing robustness (measured by δ) requires treating customers similarly in terms of demand admission in addition to keeping lower target inventory levels. Figure 4.4 reports the fill rates of class 1 and class 3, this shows that the fill rate of class 1 is increasing and the fill rate of Class 3 is decreasing in δ. Please note that decreasing δ, results in a decrease in the service quality of the class 1 customer as the controller produces uses a lower base stock level and admits more customers from other classes which increases the stock-out probability. On the other hand, class 3 has better access to the inventory and its fill rate improves when δ decreases.

An interesting question is how robustness affects overall performance. For this investigation, we consider two measures of performance: the expected total profit and the variance of the total profit obtained by simulation. Next, we report results for these performance measures as a function of δ. In order to explore the effects of variability on the expected profit, we explore the total expected profit over a short horizon. We consider the case where the total horizon is 55 stages and the initial inventory is 0. With these parameters the expected sales over the horizon is approximately 9 units. As a benchmark, we also consider the case where replenishment is not allowed. In this case, we assume that the starting inventory is 9 (corresponding to the average sales with the above case). In Figure 4.5, we present the expected profit versus the variance for both cases. It can be observed that there is a

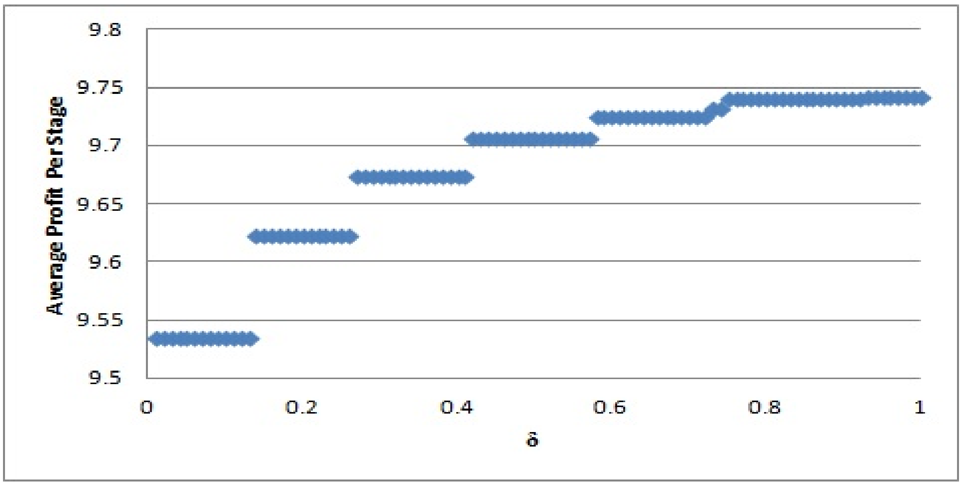

Figure 4.3: Average Profit as a function of δ

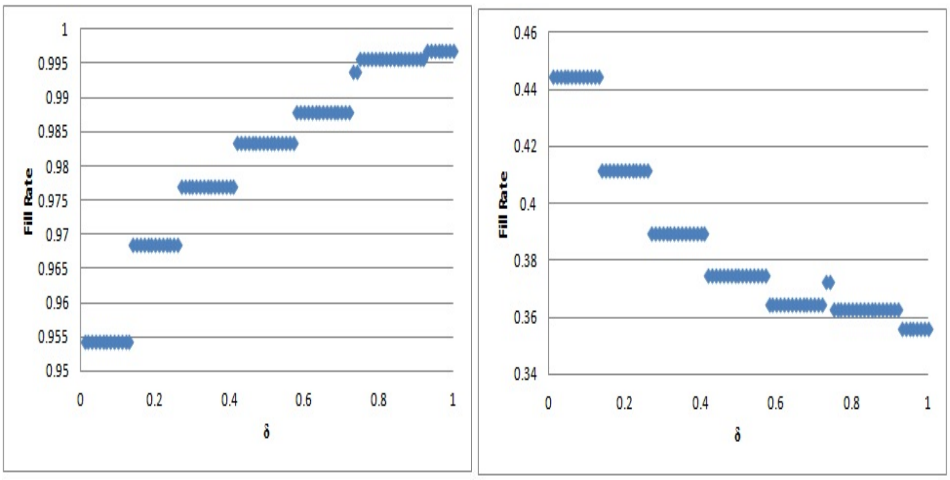

Figure 4.4: Fill Rates of Class 1 and Class 3 Customers wrt δ

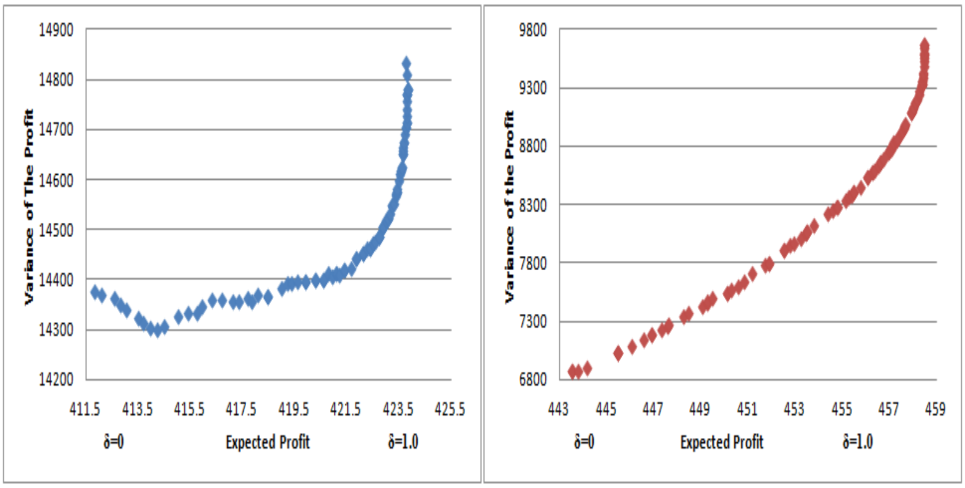

Figure 4.5: Expected Profit vs. Variance of The Profit with Production (left) and without Production (right)

significant trade-off between expected profit and variance of the profit.

Obviously the variance in the case without replenishment is less than the former case since the number of available items in the stock not affected by random production. Besides, in this case the opportunity to improve this variability is stronger. The total changes in the expected profit between the absolute robust and the nominal cases are nearly the same but the improvement in the variance is approximately 4% in the case with production and 30% without production. To further investigate these findings, let us consider a risk-sensitive objective function of the type $U(X) = E[X] - a\sigma_X$ where σ_X is the standard deviation of the random return X. Next, we assume that $a = 1$ and numerically compute the δ value that maximizes U(Profit). The utility maximizing reliance factor is denoted by δ^* and the corresponding maximum utility is denoted by U^*(Profit) respectively. As a benchmark $\tilde{U}$(Profit) designates the expected utility of the nominal policy. The results are reported in Table 4.2.

Finally, we briefly explore how the optimal δ changes with respect to the risk-sensitivity factor a. It is clear that $a = 0$ corresponds to the risk-neutral case and the optimizing $\delta^* = 1$. As we increase the value of a, we expect the δ^* to decrease. The computations reported in Figure 4.6 confirm that δ^* decreases as a increases. This preliminary exploration

	With Production	Without Production
U^*(Profit)	302.7	363.85
$\tilde{U}$(Profit)	302	360.11
Improvement	0.7	3.74

Table 4.2: Optimum U(Profit) values

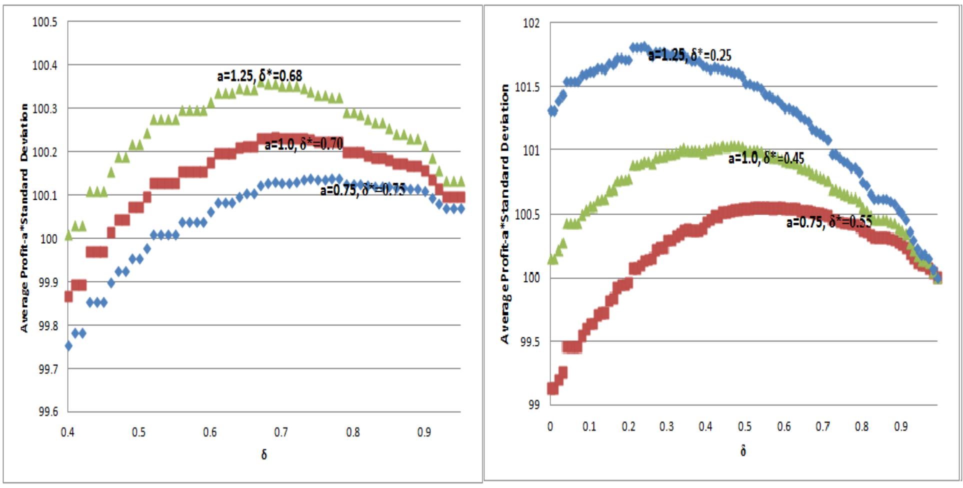

Figure 4.6: Optimum δ for Various Values of a for the cases with Production (left) and No Production (right)

suggests that there may be useful links between robust policies and their applications in a risk-sensitive decision making environment.

4.5 Conclusion

We investigated the robust versions of two single-product dynamic revenue management problems: a standard model where a fixed inventory is allocated over time to different classes of customers and a related model where inventory can be replenished by a finite capacity resource. We show that, under certain interval uncertainty models, the optimal policy structure extends to the robust case. Further, we characterized how the optimal policy changes with respect to the uncertainty set.

One drawback of a robust dynamic model is that the resulting policy may be too conservative. To alleviate this problem, we extended the analysis to a weighted optimization approach recently suggested by Xu and Mannor [34]. This approach can calibrate the level of robustness by choosing the appropriate weights between alternative objectives. We show that the optimal policy structure is not affected by this formulation.

Finally, we presented numerical results that explore how robustness affects optimal admission and production policies. While expected profits may be affected negatively by taking a robust approach, there are situations where the gain in the variance of the profit may be significant. This may suggest a computational link between risk-sensitive decision making and robust optimal policies.

In future research, we aim to explore the optimal policy structure for robust formulations of more general production/inventory control problems. Risk-sensitive optimization and computational approaches also appear to be fruitful avenues for further exploration.

Chapter 5

VALUE OF ROBUSTNESS IN BATCH RATIONING PROBLEM: AN EVALUATION ON SPOT MARKET CASE

5.1 Introduction

In this chapter, we consider the robust version of the single-item inventory management problem with replenishment where there is a setup cost of purchasing. Similar to Chapters 3 and 4, we use the maximin approach that has been used by various authors (Iyengar [16], Nilim and El Ghaoui [25], Gallego, Ryan and Simchi-Levi [10]). In that problem, we consider that the customer orders arrive in batches and partial demand acceptance is allowed. In order to model the spot market, we consider that the various batch sizes arrive with certain arrival probabilities. The batch sizes and the unit price of each item can change according to the class of the arriving demand. A similar problem has been studied by Gallego, Ryan and Simchi-Levi [10] where batch rationing is not allowed and all orders are accepted by the controller.

Perfect duality is discussed extensively in earlier Chapters. In this chapter, we illustrate a different case. We show that although the problem is not perfectly dual, the optimal robust policy and the classical policy have the same structure. Our contribution in this chapter mainly addresses the performance of the optimal policy. We propose a solution in order to improve the variance of the profit by employing the S-Robust approach suggested by Xu and Mannor [34]. In Section 5.2, we analyze the mathematical properties of the problem and in Section 5.3 we demonstrate a case where a significant improvement in variance of the profit (12%) is obtained by trading of 1% of the profit.

5.2 Model

5.2.1 Nominal Problem

We consider a finite horizon dynamic inventory control problem in this chapter. There are n classes of customers with deterministic rewards (per unit of item sold) R_i and $R_1 \geq R_2 \geq ... \geq R_n$ arriving as batches (please see the Section 2.2.3 for definition). The batch size B_i is a discrete random variable and order sizes are integer-valued which makes the state space discrete. However, the results can be easily extended to the continuous case. Time is denoted by t, where T denotes the end of the horizon. The probability of arrival $p_t^i(.)$ may be known with certainty or it may belong to an uncertainty set $\mathcal{P}_t$. We focus on the certain arrival probabilities first, then present the uncertain case later.

The state space is denoted by $x \in X$, consists of integers or may be continuous. The batch size B_i is a random variable with a discrete probability function, the controller is allowed to set the optimal rationing policy $m \in R^+$ upon observing the inventory status x for every arriving batch, stage and state independently, note that $m_t^i(y) \leq B_i$ and $\boldsymbol{B} = (B_1, B_2, ..., B_n)$. Likewise, the controller is allowed to set the ordering policy and $u_t(x) \in R^+$ for each stage and state independently.

The parameters are as follows; k (\$) is the fixed cost of the supplier and c (\$) is the unit cost per item. The holding cost is h $\$/item - time$ and it is assumed that the backlogs are not allowed and unsatisfied demand is lost.
Every batch has a specific size B_i and an arrival probability p_t^i. The cost of purchasing u units is defined as follows:

$$C(u) = k + cu \quad \text{if} \;\; u > 0$$
$$C(u) = 0 \quad \text{otherwise}$$

where $y = x + u$.
By using event-based approach, the value function $v_t(x)$ can be written as follows:

$$v_T(x) = 0$$

$$v_t(x) = \max_{m_t^i, u} \sum_{i=1}^{n} p_t^i \{ m_t^i(y) R^i + v_{t+1}(y - m_t^i(y)) - h(y - m_t^i(y)) - C(u) \},$$

The function $g_t(y)$ and the value function $v_t(x_t)$ is defined according to the following equation:

$$g_t(y) = \max_{m_t^i} \sum_{i=1}^{n} p_t^i \{ m_t^i(y) R^i + v_{t+1}(y - m_t^i(y)) - h(y - m_t^i(y)) \} \quad \text{if} \;\; y > 0$$

$$g_t(y) = v_{t+1}(0) \quad \text{if} \;\; y = 0.$$

$$v_t(x) = \max_{y \in X} \{ g_t(x_t) - cx, \max_{y \geq x} \{ k + g_t(y) - cy \} \} + cx$$

$$v_T(x) = 0. \tag{5.1}$$

The solution of the problem was provided by Scarf [29]. Scarf [29] shows that the s, S policy is optimal for a single customer class and lost sales case. Hence, the controller does not order unless the inventory status falls below the reorder point (s). In addition to this, there is a order up to level (S) if the controller decides for a purchase. In other words, it is only optimal to purchase if the inventory status falls below s_t and the optimal order is $S_t - x$.

These results will be extended to the multiple customer class case. First we give the necessary properties of the $k - concave$ functions (see Chapter 1) together with the following property from the Scarf's proof:

Remember that a function $f(x)$ is said to be $k - concave$ in x if it satisfies the following inequality:

$$f(y) \leq k + f(x) + (y - x) \frac{f(x) - f(x - b)}{b}. \tag{5.2}$$

for all $y \geq x \geq x - b \geq 0$.

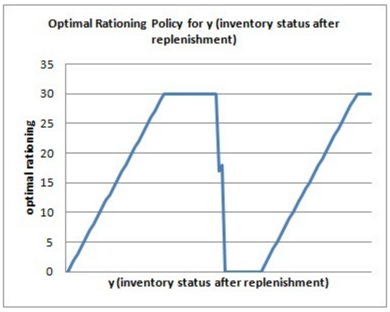

Figure 5.1: Optimal rationing decision per y

Property 5.2.1 *If $f(y)$ is $k-concave$ then $o(y) = \max_{0\leq q\leq d} f(y-q)$ is k-concave*

Property 5.2.2 *If $g_t(y) - cy$ is $k-concave$ in y then s, S policy is optimal*

Property 5.2.3 *If $g_1(y)$ and $g_2(y)$ are $k_1 - concave$ and $k_2 - concave$ respectively, then for $\alpha, \beta \geq 0$, $\alpha g_1(y) + \beta g_2(y)$ is $\alpha k_1 + \beta k_2 - concave$*

Based on these properties we define the function $g_t^i(y)$ as follows:

$$g_t^i(y) = \max_{m_t^i} \sum \{(m_t^i(y) - y)R^i + v_{t+1}(y - m_t^i(y)) - h(y - m_t^i(y))\} + R^i y. \tag{5.3}$$

By Property 5.2.3, it is clear that $g_t^i(y)$ is k-concave in y. Therefore, $g(y) = \sum p_t^i g_t^i(y) + \widehat{\mathbf{R}}y$ is k-concave and where $\widehat{\mathbf{R}} = \sum p_t^i R^i$ and is a constant, by rewriting the equation of $v_t(x)$:

$$v_t(x) = \max_{y\in X}\{g_t(x) - cx, \max_{y\geq x}\{-k + g_t(y) - cy\}\} + cx. \tag{5.4}$$

is k-concave, and Property 5.2.2 the (s, S) policy is optimal.

It is important to mention that the optimal rationing policy does not have a simple monotonic structure. In the following figure 5.1 the optimal rationing decisions per y is given.

5.2.2 Robust Problem

In the robust formulation, we follow the convention given in Chapters 3 and 4. We aim to show that the optimal robust replenishment policy is of (s, S) type. In order to do achieve, we use Properties 5.2.4 and 5.2.5. A formal proof for Property 5.2.4 can be found at Chapter 3.

Property 5.2.4 *The optimal action $m_t^i(y)$ does not depend on $p \in \mathcal{P}_t$ for the robust value function equation $w_t(x)$ given in the Equation as:*

$$w_t(x) = \max_{m_t^i, y \geq x} \{ \min_{p_i^t(x,a) \in \mathcal{P}_t} \{ \sum p_t^i(x) \{ m_t^i(x) R^i + v_{t+1}(x - m_t^i(x)) - h(x - m_t^i(x)) \} - cx, \\ \sum p_t^i(y) \{ m_t^i(y) R^i + v_{t+1}(x - m_t^i(y)) - h(x - m_t^i(y)) \} - cy - k \} + cx.$$

Proof: Property 5.2.4 is a direct consequence of Theorem 3.3.1 given in Section 3. □

Property 5.2.5 *The pointwise minimum of k-concave functions is k-concave*

Proof of this property can be found in several sources, however we provide an independent proof here.

Proof: First please note that the following inequality holds for every i:

$$\left[1 + \frac{a}{b}\right] f_i(x) \geq f_i(x + a) + \frac{a}{b} f_i(x - b) - k.$$

Hence, it is apparent that:

$$\left[1 + \frac{a}{b}\right] f_i(x) \geq \min_i f_1(x + a), f_2(x + a), ..., f_i(x + a).... + \frac{a}{b} \min_i f_1(x - b), f_2(x - b), ..., f_i(x - b).... - k$$

Suppose that $m = \arg\min_i f_1(x), f_2(x), ..., f_i(x).....$ Then:

$$\left[1 + \frac{a}{b}\right] f_m(x) \geq \min_i f_1(x + a), f_2(x + a), ..., f_i(x + a).... + \frac{a}{b} \min_i f_1(x - b), f_2(x - b), ..., f_i(x - b).... -$$

This completes the proof. □

Theorem 5.2.1 *The optimal replenishment policy is of (s, S) type.*

Proof According to Property 5.2.4 the optimal robust rationing policy $m_t^i(y)$ does not depend on the posteriori decision of the Nature. Hence, the optimal $m_t^i(y)$ is defined by:

$$m_t^i(y) = \arg \max_{m \leq \min(B_i, y)} \{ mR^i + w_{t+1}(y - m) - H(y - m) \}.$$

The robust dynamic programming equation is given in the following:

$$g_t^i(y) = \sum p_t^i(y) \max_{m_t^i \in M} \{(m_t^i(y) - y)R^i + w_{t+1}(y - m_t^i(y)) - h(y - m_t^i(y))\} + p_t^i(y)R^i y.$$

By Property 5.2.5, $g_t^i(y)$ is k-concave, and the following robust value function $w_t(x)$ preserves the k-concavity property. Therefore, an (s, S) policy is optimal for the robust counterpart of the problem.

$$w_t(x) = \max_{y \in X}\{g_t(x) - cx, \max_{y \geq x}\{-k + g_t(y) - cy\}\} + cx.$$

5.2.3 Counter Examples

Here, we present two counterexamples. The first is a counterexample for monotonicity of the optimal rationing policy. We show that the optimal rationing policy is not monotone and counterintuitive by an example. In the second counter example, we show that the problem is not perfectly dual and unlike to the optimal rationing decision, the optimal purchasing decision depends on the Nature's posteriori decision.

Counter Example for Monotonicity of the Batch Rationing Policy

The parameters of the counter example is as follows:

Number of Batches	6
Total Stages	10
Setup Cost k	\$300
Cost/Item	\$5
Holding Cost/Item-Stage	\$0.01

Batch No	Batch Size	Arrival Probability	Reward
1	30	0.70	30
2	10	0.06	20
3	20	0.06	15
4	30	0.06	10
5	40	0.06	7.50
6	50	0.06	3

Table 5.1: Optimal Rationing Decisions for Different Inventory Levels

Inventory Status y	Class 2	Class 3
43	20	30
44	14	30

In the Table 5.2.3 optimal rationing decisions are presented when there is 3 more stages in the horizon. As it is seen in the Table 5.2.3 all the incoming demand has been satisfied at state $y = 43$. However, when there is one more item in the inventory, i.e. 44 items, the controller only satisfies 14/20 of the demand. Moreover, the controller satisfies all the demand of a less valuable Class and satisfies all the incoming demand at both states. The reason of this situation depends on the purchasing decisions of the following stages, controller gives a purchase order when there are less than 28 items in the inventory when there are two more remaining stages. When a demand of 20 arrives at state 44, the controller concludes that to reserve 30 items in the inventory and earn an instant reward of \$280 and to not order in the preceding stages (and sell only to Class-1 customer) instead of earning an \$400 of instant reward and make a purchase order in the preceding stages. However, when a class with lower reward but higher demand comes controller earns \$450 instant reward and concludes to make purchasing decisions for the preceding stages.

Counter Example for the Perfect Duality

In this part, we show that the problem is not perfectly dual. Hence, the optimal purchasing decision and the optimal expected revenue change if the sequence of the game is changed. In the original robust formulation, Nature decides upon observing the controller's decision. The dual counterpart designates the case where controller decides upon observing the Nature's decision. In our example, the uncertainty set consists of three discrete points representing the arrival probabilities, $p_1 = (1, 0, 0)$, $p_2 = (0, 1, 0)$ and $p_3 = (0, 0, 1)$. The other problem parameters are as follows:

Number of Batches	3	Batch No	Batch Size	Reward
Setup Cost k	\$ 200	1	10	30
Cost/Item	\$ 5	2	20	20
Holding Cost/Item-Stage	\$ 0,01	3	30	15

Now suppose that the problem is a single stage problem and there is not any item in the inventory ($x = 0$) and controller decides after observing the arrival probabilities. The uncertainty set represents a situation that the arrival will be exactly from one of the classes. If controller knows that the arrival will be Class-1, then it decides to purchase 10 items, sell to the Customer and have an expected profit of \$50. If controller knows that the arrival will be Class-2, then it decides to purchase 20 items, sell to customer and have an expected profit of \$100 . Similarly, if the controller knows that the arrival will be Class-3 then it decides to purchase 30 items, sell to customer and have an expected profit of \$100. Obviously, by knowing these Nature will decide on the p_1, and the arrival will be $Class - 1$ and the controller will earn \$50.

Now suppose the original form of the game. Let $f_1(y)$, $f_2(y)$, $f_3(y)$ denotes the expected profit if the controller selects y item and the probability distribution is p_1, p_2, p_3 respectively. The equations are as follows:

$$f_1(y) = 25y - 200 \text{ if } 0 < y \leq 10$$
$$f_1(y) = 100.1 - 5.01y \text{ else,}$$

$$f_2(y) = 15y - 200 \text{ if } 0 < y \leq 20$$
$$f_2(y) = 200.2 - 5.01y \text{ else,}$$

$$f_3(y) = 10y - 200 \text{ if } 0 < y \leq 30$$
$$f_3(y) = 250.3 - 5.01y \text{ else.}$$

Since Nature's objective is to minimize the following resultant function is given as the solution of the game:

$f(y) = \min\{25y - 200, 15y - 200, 10y - 200\}$ if $0 < y \leq 10$
$f(y) = \min\{100.1 - 5.01y, 15y - 200, 10y - 200\}$ if $10 < y \leq 20$
$f(y) = \min\{100.1 - 5.01y, 200.2 - 5.01y, 10y - 200\}$ if $20 < y \leq 30$
$f(y) = \min\{100.1 - 5.01y, 200.2 - 5.01y, 250.3 - 5.01y\}$ else

and equivalently:

$$f(y) = 10y - 200 \; if \; 0 < y < 19.9933$$
$$f(y) = 100.1 - 5.01y \; else.$$

Apparently, the optimal value of $y \simeq 19.9933$ and Nature is indifferent between p_1 and p_3 in that case. Hence, the controller's best choice is to stay at inventory level 0 and not to order anything. In Figure 5.2 the expected profit of each decision under each probability distribution is presented.
This establishes that the problem is not perfectly dual. Table is given as illustration for the integer case.

5.2.4 S-Robust Policy

Xu and Mannor [34] and Paschalidis [27] proposed an approach modifying the absolute robust policy that are more adaptive to different realizations in the uncertainty set. We again discuss here the S-Robust Policy framework suggested by Xu and Mannor [34] that we illustrate in Chapter 3.
By using their approach, we suggest that the S-robust policy satisfies the following equation:

Table 5.2: Profit of Each Decision Under Different Probability Distribution

y	$Profit\,1$	$Profit\,2$	$Profit\,3$	minimum
1	-175	-185	-190	-190
2	-150	-170	-180	-180
3	-125	-155	-170	-170
4	-100	-140	-160	-160
5	-75	-125	-150	-150
6	-50	-110	-140	-140
7	-25	-95	-130	-130
8	0	-80	-120	-120
9	25	-65	-110	-110
10	50	-50	-100	-100
11	44.99	-35	-90	-90
12	39.98	-20	-80	-80
13	34.97	-5	-70	-70
14	29.96	10	-60	-60
15	24.95	25	-50	-50
16	19.94	40	-40	-40
17	14.93	55	-30	-30
18	9.92	70	-20	-20
19	4.91	85	-10	-10
20	-0.1	100	0	**-0.1**
21	-5.11	94.99	10	-5.11
22	-10.12	89.98	20	-10.12
23	-15.13	84.97	30	-15.13
24	-20.14	79.96	40	-20.14
25	-25.15	74.95	50	-25.15
26	-30.16	69.94	60	-30.16
27	-35.17	64.93	70	-35.17
28	-40.18	59.92	80	-40.18
29	-45.19	54.91	90	-45.19
30	-50.2	49.9	100	-50.2

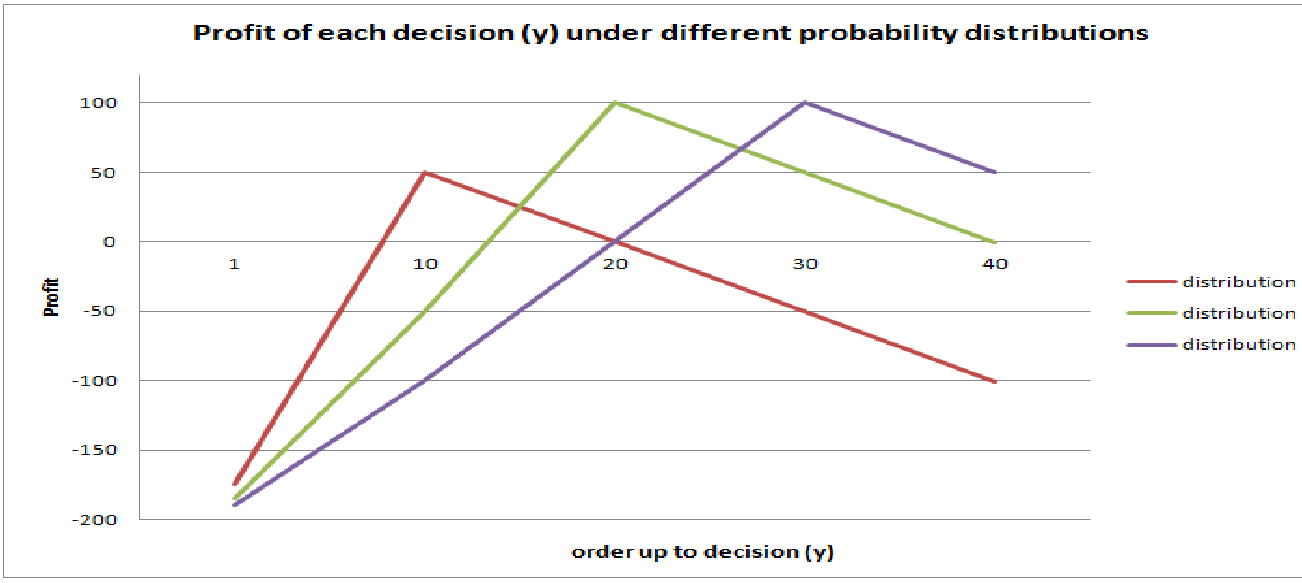

Figure 5.2: Profit of Each Decision Under Different Probability Distributions

$$w_t(x) = \max_{m_t^i \in M, y \geq x} \{ \min_{p_i^t(x,a) \in \mathcal{P}_t, q_i^t(x,a) \in \mathcal{P}_t, \mathcal{P}_t^{\mho}} w_t^s(m_t^i, x, y)\} + cx$$

$$\begin{aligned} w_t^s(m_t^i, x, y) = \delta_t \sum p_t^i(x)\{m_t^i(x)R^i + v_{t+1}(x - m_t^i(x)) - h(x - m_t^i(x))\} + \\ (1-\delta_t) \sum q_t^i(x)\{m_t^i(x)R^i + v_{t+1}(x - m_t^i(x)) - h(x - m_t^i(x))\} - cx, \\ \delta_t \sum p_t^i(y)\{m_t^i(y)R^i + v_{t+1}(x - m_t^i(y)) - h(x - m_t^i(y))\} + \\ (1-\delta_t) \sum q_t^i(y)\{m_t^i(y)R^i + v_{t+1}(x - m_t^i(y)) - h(x - m_t^i(y))\} - cy - k. \end{aligned}$$

By property 5.2.4, the optimal action m_t^i does not depend on the Nature's posteriori decisions $p_i^t(x,a)$ and $q_i^t(x,a)$. Therefore, Theorem 5.2.1 applies easily.

We use this approach in order to improve the variance of the profit. By constructing a fictitious uncertainty set around the probability distribution and selecting different δ values different sub-optimal polices are obtained. These policies are then simulated with respect to the actual probability distribution. In the next section we provide an example.

5.3 Numerical Example

In order to demonstrate our solution approach, we present the following example. Suppose the main customer batch size is 100, arrival probability is 0.1 and reward is \$10. The setup

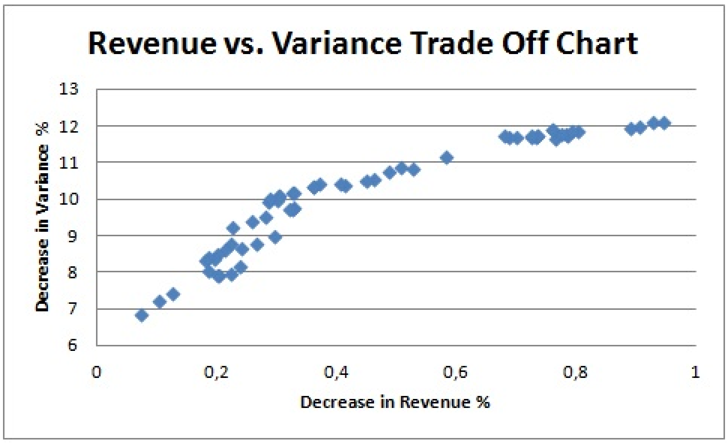

Figure 5.3: Revenue vs. Trade-Off Chart

cost k is \$250, and c is \$5 and the annual interest rate is 0.20. The batches coming from the spot market has the arrival probability of 0.90 and the batch sizes can be any value in the set (10, 20, 30, 40, 50) each having an equal probability. In order to reduce the variability of the solution, we construct a fictitious set around this distribution and suppose that arrival probability of the spot market classes changes between $0.18 \mp 20\%$. The suboptimal solutions obtained by selecting different δ values are then ranked according to their average revenue and variance results. Please note that the coefficient of variation (standard deviation/average revenue) of the profit in the nominal problem is 0.19.

The simulation is performed with 1000, 000 runs. Figure 5.3 depicts the trade off between the variance and the simulated revenue.
The worst 10% cases out of 1000000 simulations in terms of profit are obtained for the nominal policy. Then these cases are reevaluated by using the pure robust and S-robust policies, i.e. a simulation is carried out for these instances with robust and S-robust policies. In the following figures, the difference between the average robust revenue and the average nominal revenue are shown for each particular realization. Therefore, each point corresponds to the difference of simulated revenue for that instance. In Figure 5.4, the $robust\ revenue - \ nominal\ revenue$ are given. When the worst 1000 realization of the nominal policy are considered the 36.8% of these realizations are higher than the associated result obtained by pure robust strategy. If the same comparison is performed between the

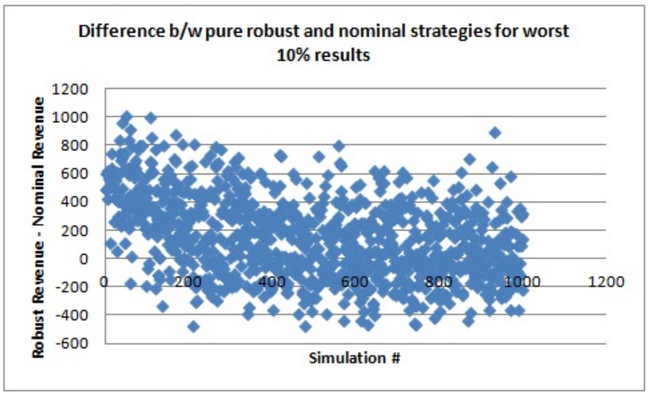

Figure 5.4: Difference b/w pure robust and nominal strategies for worst 10% results

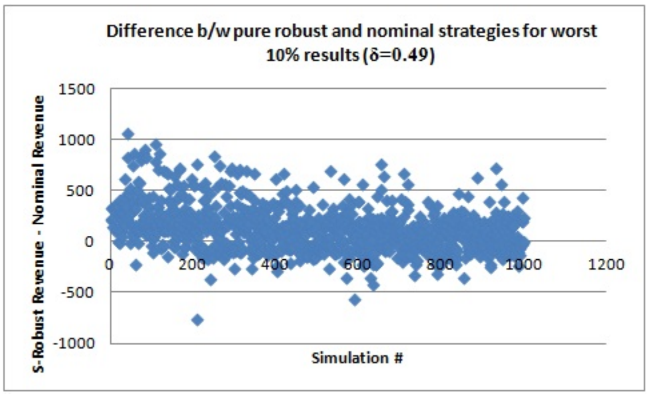

Figure 5.5: Difference b/w pure robust and nominal strategies for worst 10% results ($\delta = 0.49$)

$S-robust$ policy and the nominal policy the results are given in Figure 5.5, where 31.7% of the results obtained by nominal policy performs better than the associated result obtained by $S-robust$ policy. These figures demonstrate that the robust policies perform better for the worst case realizations. It is notable that the performance of the pure robust policy with respect to nominal policy is superior for the worst case realizations.

Chapter 6

REVENUE MANAGEMENT PROBLEMS WITH COMMITMENTS

6.1 Problem Definition

In this part, we consider a slightly different formulation of a revenue management problem. We suppose that there are financial commitments such as sales targets and installment dues in the horizon. A similar problem is first analyzed by Besbes and Maglaras [6] where they propose a discrete-review policy and dynamically track the commitments. In our approach, we track the provision of commitments but we employ a dynamic policy rather than a discrete policy. We consider the installments whose due dates are set to equal intervals over the horizon and whose amounts are equal to each other. However, our method can easily be extended to a case where the intervals and installment amounts change over the horizon.

Although the emphasis of this thesis is a set of problems with structural characteristics, we provide some numerical examples in order to demonstrate the usefulness of robust decision making when the problem does not possess any structure for its optimal policy. Before providing our examples, we first formulate the revenue management problem with periodic installments. We compare the numerical results with the dynamic programming solution and the approximate solution that is based on robust decision making.

As discussed in Chapter 4 the value function of the single-item inventory management problem without replenishment and its robust counterpart is already given. The objective function is to maximize the total expected profit. Let's recall the classical formulation of the problem:

$$v_t(x) = \sum_{i=1}^{n} \lambda_{i,t} \max\{R_i + v_{t+1}(x-1), v_{t+1}(x)\} + \lambda_{n+1,t}(x)v_{t+1}(x).$$

Please remember that $v_t(x)$ denotes the value function, $x \in Z^+$ denotes the inventory status,

i represents the customer class, R_i represents the associated reward of each customer class. There is a total of n classes and $n+1$ represents the fictitious class (class with 0 reward). The total arrival probability together with the fictitious class is 1 and each arrival probability is represented with $\lambda_{i,t}$. The total horizon is T stages and t denotes the current stage.

Now suppose that the company has to pay an installment at certain stages. Hence, there is an installment I_t for a given stage t. If the controller fails to pay the total installment amount, then a positive penalty proportional to the unpaid amount is borne, i.e. suppose the installment is \$1000 and the current cash position is \$900, then the cash position drops to \$$-100$ and after the introduction of the penalty (suppose 20%) the cash position is calculated as \$ -120, i.e. there is a penalty Pe for unsuccessful commitment. We introduce the cash position as CP, installment as I_t and penalty ratio as pr into the dynamic programming algorithm and formulate the dynamic programming algorithm as follows by treating the fictitious event as a special customer class with 0 reward:
Now let's define the following equations for penalties where A stands for admission and R stands for rejection:

$$\begin{aligned} Pe^i_A &= 0 \text{ if } CP + R_i - I_t > 0 \\ Pe^i_A &= pr(CP + R_i - I_t) \text{ else} \\ Pe^i_R &= 0 \text{ if } CP - I_t > 0 \\ Pe^i_R &= pr(CP - I_t) \text{ else.} \end{aligned}$$

If the controller admits any customer class at x, t the operator $Tv^A_{i,t+1}(x, CP)$ is equal to:

$$Tv^A_{i,t+1}(x, CP) = R_i + Pe^i_A - I_t + v_{t+1}(CP + R_i + Pe^i_A - I_t, x - 1).$$

Similarly, if the controller rejects any customer class at x, t the operator $Tv^R_{i,t+1}(x, CP)$ is equal to:

$$Tv^R_{i,t+1}(x, CP) = Pe^i_R - I_t + v_{t+1}(CP + Pe^i_R - I_t, x).$$

Last, the fictitious operator is a special type where a customer with zero reward is rejected. Therefore, the value function is given in below equation:

$$v_t(x, CP) = \sum_{i=1}^{n} \lambda_{i,t} \max\{Tv^A_{i,t+1}(x, CP), Tv^R_{i,t+1}(x, CP)\} + \lambda_{n+1,t}(x) Tv^R_{n+1,t+1}(x, CP), \quad (6.1)$$

with the following boundary condition for $t = T$:

$$Pe = 0 \text{ if } CP - I_T > 0$$

$$Pe = pr(CP - I_T) \text{ if } CP - I_T < 0$$

$$v_T(x, CP) = -I_T - Pe.$$

Due the fact that the cash position CP is a continuous variable, the dynamic programming equation given in Algorithm 6.1 cannot be solved exactly. Therefore, an appropriate discretization is selected for CP in order to compute the optimal policy. Moreover, there does not appear to be simple structure that governs the optimal policy and it is possible to find examples for the following argument:

Rejecting a customer at x does not necessitate rejecting a customer at $x-1$ for a given cash position CP, i.e., there is not necessarily a threshold in terms of x at any stage at a given cash position CP.

Consider the following example with two customers where the arrival probability vector is $\boldsymbol{p} = (0.5, 0.5)$ and the reward vector is $\boldsymbol{R} = (\$64, \$16)$. Suppose that at every two stage there is an installment of 32\$ and at the initial stage there is no payment, the penalty ratio $pr = 1$. Hence, if cash position is negative at the end of any stage, it is multiplied by 2. In the initial stage when the inventory status $x = 4$ and the initial cash position is $\$-1$ it is optimal to reject the second class customer (i.e. \$16) on the contrary when inventory status $x = 3$ (and the initial cash position is $\$-1$) it is optimal to accept the second class customer.

We propose a myopic algorithm based on the classical dynamic programming formulation. The algorithm is developed in two phases, in the first phase the optimal threshold policy is obtained by using the dynamic programming equation given in Algorithm . In the second phase, the policy is reevaluated based on the current cash position CP and the next installment I_{t+n} at stage t where $I_{t+1} = ... = I_{t+n-1} = 0$.

$$
\begin{aligned}
k = the\ lowest\ class\ accepted\ at\ state\ C, x \\
isSurvive = false \\
do \\
if\ n \sum_{i=1}^{i=k} \lambda_{i,t} R_i + C \ \geq \ I_{t+n} \\
isSurvive = true \\
\text{else } k = k + 1 \\
until\ isSurvive\ or\ k = n.
\end{aligned} \tag{6.2}
$$

As clearly seen, the algorithm evaluates the sufficiency of the current policy and cash position for satisfying the next installment. In order to determine this the optimal threshold of the current inventory is checked and the lowest customer class that is admitted is obtained. Then by assuming that the this thresholds are maintained till the installment due the average collected reward is calculated. If the cash position CP is sufficient, i.e. $n \sum_{i=1}^{i=k} \lambda_{i,t} R_i + C \ \geq I_{t+n}$, then the optimal policy evaluated for the classical problem is applied otherwise the optimal policy is updated according to the Algorithm 6.2. In the next section, we provide some numerical examples in order to demonstrate the efficiency of the algorithm.

6.2 Numerical Illustrations

In order to assess the effectiveness of the algorithm we compare it with the semi-robust approach on a sample problem whose parameters are given in below:

Number of Stages	500
Starting Inventory	100
Number Of Installments	10
Installment Period	Every 50 stages
Number of Customer Classes	5
Customer Rewards ($)	$(40, 38, 35, 33, 30)$
Arrival Probability	$(0.10, 0.10, 0.10, 0.10, 0.10)$
Absolute Robust Arrival Parameters	$(0.05, 0.05, 0.05, 0.05, 0.05)$
Penalty ratio pr	0.01

Absolute robust arrival probabilities represent the lower bounds on the arrival probabilities. The semi-robust approach is given in Chapter 3 and 4 and the best performing S-robust policy is evaluated according to the method illustrated there. There are 10 installments on every $50th$ stage and the installment is $350 in the first case and $380 in the second case. The installment amounts are selected as approximately %90 and %98 of the optimal expected revenue obtained by Equation 6.1. The dynamic programming algorithm given in the Equation 6.1 can not solve the equation exactly since the cash position CP is a continuous variable. For this reason, we solve only for integer cash positions between $(-1200, 1200)$ interval. The results are as follows:

1. When installment is $350 the results are as follows:

Optimal revenue obtained by DP algorithm 6.1	$380.47
Simulation result of the 6.1	$376.75
Simulation result of 6.2	$379.52
Simulation result of the best semi-robust solution	$375.82

2. When installment is 380$ the results are as follows:

Optimal revenue obtained by DP algorithm 6.1	$76.78
Simulation result of the 6.1	$69.87
Simulation result of 6.2	$74.77
Simulation result of the best semi-robust solution	$74.22

As it is seen, the myopic algorithm proposed in 6.2 performs better than the policy obtained

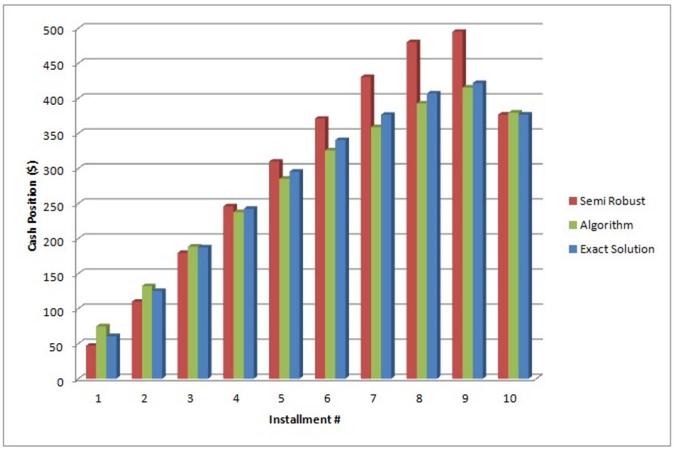

Figure 6.1: Cash Position After Each $350 Installment Payment

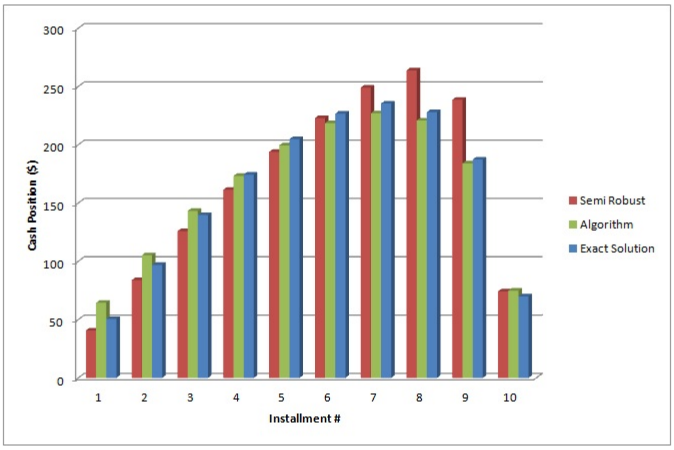

Figure 6.2: Cash Position After Each $380 Installment Payment

by the dynamic programming equation given in 6.1. The reason is that the equation is not exactly solved, rather it is solved for a discrete approximation values and the cash position is rounded to the nearest discrete value during simulation. Semi-robust solution is competitive to the 6.1 however at least 20 S-robust solutions have to evaluated in order to obtain the best performing S-robust policy. In the following figures 6.1 and 6.2, the cash positions at the end of each installment period are presented for both cases.

As it is seen Figures 6.1 and 6.2 the cash position in the semi-robust solution is the highest in the early installment periods. As already explained in Chapter 4, the semi-robust solution gradually decreases the acceptance thresholds. However, the proposed algorithm efficiently mimics the dynamic programming algorithm given in Algorithm 6.1 and does not decrease the thresholds unless it is necessary.

Chapter 7

CONCLUSION

In this thesis, we investigated the structural properties of the optimal policies of the robust counterparts of some stochastic dynamic programming problems. In order to achieve this, we explored the mathematical properties of the robust value function $w_t(x)$. By using a general model of uncertainty representing the transition probabilities, we succeeded to show that the mathematical properties of the robust value function $w_t(x)$ are inherited from the classical counterpart. This result implies that the structure of the optimal policy that governs the system is also the same. We also elaborated on the perfect duality property and effectively used this property in order to prove the independence of the optimal action from Natures posteriori decisions. This enabled us to propose semi robust solutions with better performance while maintaining computational tractability.

Imposing certain mathematical conditions on the uncertainty set enabled us to compare the optimal polices with respect to the perturbations in the uncertainty set. By using an interval uncertainty set, we showed that the optimal thresholds monotonically change when the uncertainty set is enlarged. This is a powerful feature that enables to construct and compare different policies systematically.

We provide some results for the cases where other problem parameters such as cost and rewards are uncertain. We demonstrate that the robust dynamic problem does not have a computationally tractable solution unless the problem parameters have certain properties.

Our research suggests some future areas of exploration. We provide some general knowledge on optimal policies of the robust counterparts of multi dimensional problems. However, there remains a lot of opportunity for this case. We also proposed to use S-robust approach in order to improve the variance of the expected profit for a case where all of the problem parameters are known with certainty. Different techniques may be adopted and compared for this purpose. Last, we demonstrated on a specific case that an approximate algorithm inspired by other economical theories has a good performance.

BIBLIOGRAPHY

[1] E. Altman and A. Hordijk. Zero-sum markov games and worst-case optimal control of queueing systems. *Queueing Systems*, 21(3):415–447, 1995.

[2] S. Aydin, Y. Akcay, and F. Karaesmen. On the structural properties of a discrete-time single product revenue management problem. *Operations Research Letters*, 37(4):273–279, 2009.

[3] M. Ball and M. Queyranne. Toward robust revenue management: Competitive analysis of online booking. *Operations research*, 57(4):950–963, 2009.

[4] A. Ben-Tal, L. El Ghaoui, and Nemirovskiĭ. *Robust optimization*. Princeton Univ Pr, 2009.

[5] D. Bertsekas. Dynamic programming and optimal control. 1995.

[6] O. Besbes and C. Maglaras. Dynamic pricing with financial milestones: Feedback-form policies. 2012.

[7] S. Birbil, J. Frenk, J. Gromicho, and S. Zhang. The role of robust optimization in single-leg airline revenue management. *Management Science*, 55(1):148, 2009.

[8] E. Cil, E. Ormeci, and F. Karaesmen. Effects of system parameters on the optimal policy structure in a class of queueing control problems. *Queueing Systems*, 61(4):273–304, 2009.

[9] E. Delage and S. Mannor. Percentile optimization for Markov decision processes with parameter uncertainty. *Operations research*, 2009.

[10] G. Gallego, J. Ryan, and D. Simchi-Levi. Minimax analysis for finite-horizon inventory models. *IIE Transactions*, 33(10):861–874, 2001.

[11] G. Gallego and G. Van Ryzin. Optimal dynamic pricing of inventories with stochastic demand over finite horizons. *Management science*, pages 999–1020, 1994.

[12] I. Gilboa. Theory of decision under uncertainty. *Cambridge Books*, 2009.

[13] S. H. The optimality of (s;s) policies in the dynamic inventory problem. *Mathematical Models in the Social Sciences, Stanford University Press*, 1960.

[14] A. Ha. Inventory rationing in a make-to-stock production system with several demand classes and lost sales. *Management Science*, pages 1093–1103, 1997.

[15] B. Huang and S. Iravani. A make-to-stock system with multiple customer classes and batch ordering. *Oper. Res*, 56(5):1312–1320, 2008.

[16] G. Iyengar. Robust dynamic programming. *Mathematics of Operations Research*, 30(2):257, 2005.

[17] A. Jain, A. Lim, and J. Shanthikumar. On the optimality of threshold control in queues with model uncertainty. *Queueing Systems*, 65(2):157–174, 2010.

[18] G. Koole. Structural results for the control of queueing systems using event-based dynamic programming. *Queueing Systems*, 30(3):323–339, 1998.

[19] G. Koole. *Monotonicity in Markov reward and decision chains: Theory and applications.* Now Publishers Inc., 2006.

[20] P. Kouvelis and G. Yu. *Robust discrete optimization and its applications.* Kluwer Academic Pub, 1997.

[21] Y. Lan, H. Gao, M. Ball, and I. Karaesmen. Revenue management with limited demand information. *Management Science*, 54(9):1594–1609, 2008.

[22] C. Lautenbacher and S. Stidham Jr. The underlying Markov decision process in the single-leg airline yield-management problem. *Transportation Science*, 33(2):136, 1999.

[23] A. Lim and J. Shanthikumar. Relative entropy, exponential utility, and robust dynamic pricing. *Operations research*, 55(2):198–214, 2007.

[24] S. Lippman. Applying a new device in the optimization of exponential queuing systems. *Operations Research*, 23(4):687–710, 1975.

[25] A. Nilim and L. Ghaoui. Robust control of Markov decision processes with uncertain transition matrices. *Operations Research*, 53(5):780–798, 2005.

[26] A. Nowak. On zero-sum stochastic games with general state space i. *Probability and Mathematical Statistics*, 4(1):13–32, 1984.

[27] I. Paschalidis and S. Kang. A robust approach to Markov decision problems with uncertain transition probabilities. In *Proceedings of the 17th IFAC World Congress*, pages 408–413, 2008.

[28] J. Satia and R. Lave Jr. Markovian decision processes with uncertain transition probabilities. *Operations Research*, 21(3):728–740, 1973.

[29] H. Scarf. Optimal inventory policies when sales are discretionary. *Cowles Foundation Discussion Papers*, 2000.

[30] M. Sion. On general minimax theorems. *Pacific Journal of Mathematics*, 8(1):171–176, 1958.

[31] S. Stidham Jr. Optimal control of admission to a queueing system. *Automatic Control, IEEE Transactions on*, 30(8):705–713, 2002.

[32] K. Talluri and G. Van Ryzin. *The theory and practice of revenue management*, volume 68. Springer Verlag, 2005.

[33] Z. Turgay, F. Karaesmen, and L. Ormeci. Robust Control of Stochastic Inventory and Queueing Problems. *Working Paper.*

[34] H. Xu and S. Mannor. The MaxMin Expected Utility Approach to Uncertain Markov Decision Processes, 2009.

VITA

Zeynep Turgay was born in Istanbul. She received her B.Sc. and M.Sc. degrees from METU (Ankara, Turkey) in 1998 and 2001. After gaining seven years of professional experience in Research and Development sector she joined to Koc University (Istanbul, Turkey) in 2008 in order to pursue her Ph.D. degree in Industrial Engineering and Operations Management and received her degree in 2012. Her research interests are stochastic dynamic programming and revenue management with an emphasis on economic theories.