2939

REPRINT

Tanja Mlinar and Philippe Chevalier

Dynamic admission control for two customer classes with stochastic demands and strict due dates

International Journal of Production Research, 54(20), 6156-6173, 2016

CORE CENTER FOR OPERATIONS RESEARCH AND ECONOMETRICS

CORE
Voie du Roman Pays 34, L1.03.01
Tel (32 10) 47 43 04
Fax (32 10) 47 43 01
Email: immaq-library@uclouvain.be
https://uclouvain.be/en/research-institutes/immaq/core/reprints.html

International Journal of Production Research, 2016
Vol. 54, No. 20, 6156–6173, http://dx.doi.org/10.1080/00207543.2015.1055846

Taylor & Francis
Taylor & Francis Group

Dynamic admission control for two customer classes with stochastic demands and strict due dates

Tanja Mlinar[a] and Philippe Chevalier[b*]

[a]*ESC Rennes School of Business, Rennes, France;* [b]*Louvain School of Management, Center for Operations Research and Econometrics, Université catholique de Louvain, Louvain-La-Neuve, Belgium*

(*Received 30 August 2014; accepted 4 May 2015*)

We study a dynamic capacity allocation problem with admission control decisions of a company that caters for two demand classes with random arrivals, capacity requirements and strict due dates. We formulate the problem as a Markov decision process (MDP) in order to find the optimal admission control policy that maximises the expected profit of the company. Such a formulation suffers a state-space explosion. Moreover, it involves an additional dimension arising from the multiple possible order sizes that customers can request which further increases the complexity of the problem. To reduce the cardinality of possible policies, and, thus, the computational requirements, we propose a threshold-based policy. We formulate an MDP to generate such a policy. To deal with the curse of dimensionality, we develop threshold-based approximate algorithms based on the state-reduction heuristics with aggregation proposed previously. Our results reveal that for the majority of instances considered the optimal policy has a threshold structure. We then demonstrate the superiority of the proposed threshold-based approximate algorithms over two benchmark policies in terms of the generated profits and the robustness of the solutions to changes in operational conditions. Finally, we show that our proposed policies are also robust to changes in actual demand from its estimation.

Keywords: demand and revenue management; admission control; Markov decision process; heuristics; due dates

1. Introduction

In this paper, we consider a company that possesses a single bottleneck capacity to process the requirements of two demand streams. Customers of each demand stream arrive dynamically and stochastically along an infinite time horizon and place their orders. The demand streams differ in order sizes (measured in terms of capacity requirements), profit margins and lead times. We assume that arrivals and order sizes of future customer requests are random variables and that lead times and profit margins are known. We suppose the company charges a premium for a demand stream with a tight lead time. In practice, the company charges such a price for faster delivery because of the willingness of customers to pay an extra price for quick responses and the reduced flexibility to allocate capacity within a short lead-time window. The stochastic nature of demand and the limited capacity of the company, however, lead to difficulties on meeting the due dates of all the customers. Thus, the problem of the company is how much capacity should be assigned for the high-margin demand in order to maximise the expected long-run profit. In other words, each time a customer request is placed, the company has to decide whether to accept/reject the coming customer request taking into account the available capacity, the size of the order and the future demand.

Capacity allocation problems with admission control decisions in similar manufacturing and service environments have already been addressed in the literature. Here, we stress the most related papers to our work.

A first stream of papers focuses on applying revenue management techniques for manufacturing environments without explicitly taking into account due dates of customers when modelling capacity allocation problems. Balakrishnan, Sridharan, and Patterson (1996) develop an effective heuristic admission control method for allocating capacity to two profit classes for which orders arrive stochastically during a single planning horizon. In Balakrishnan, Patterson, and Sridharan (1999), the authors show that the heuristic proposed by Balakrishnan, Sridharan, and Patterson (1996) is robust with the change in profit performance in case of the forecast errors in the demand parameters. Hung and Lee (2010) develop a static and a dynamic stochastic capacity rationing procedures for the order acceptance problem under the assumptions that the capacity requirement and the profit of an order are continuous random variables. The authors, further, assume that the arrival and production periods are not overlapped and that the production period occurs after the arrival process. Their results reveal superiority of the dynamic procedure over the static procedure and the first-come-first-served (FCFS) rule. Hung, Tsai, and Wu (2014) extend the dynamic stochastic capacity rationing procedure to different applications.

*Corresponding author. Email: philippe.chevalier@uclouvain.be

In contrast to this first stream, a second group of papers considers the admission control and scheduling problem taking into account due dates but without applying revenue management techniques (see e.g. Ivanescu, Fransoo, and Bertrand 2002, 2006; van Foreest, Wijngaard, and van der Vaart 2010). Ivanescu, Fransoo, and Bertrand (2006) consider the production control structure of batch process industries featuring complex job and resource structures. The authors assume that all arriving customer orders will generate the same revenue per unit of capacity consumed. Thus, the production control structure consists of an admission control problem that selectively accept/reject orders in order to maximise resource utilisation, and a scheduling problem that determines the sequence and timing of the execution of the accepted orders on the resources. The authors investigate the impact of different acceptance policies on different performance measures such as the resource utilisation, delivery reliability and the acceptance rate. van Foreest, Wijngaard, and van der Vaart (2010) develop several order acceptance policies for the customised stochastic lot scheduling problem (CSLSP) with large setups an strict order due dates. The authors use simulation to analyse the performance of the developed policies in terms of resource utilisation.

Papers belonging to a third group implement revenue management techniques to deal with the admission control and scheduling problem with due dates. Barut and Sridharan (2004, 2005) consider the multiperiod multiclass order acceptance problem with heterogeneous due dates and profit margins, and stochastic order sizes and arrival process. The authors develop a dynamic capacity apportionment procedure in order to obtain a protection-level-based (PLB) policy. The policy protects a portion of the available capacity for future orders of higher profitable classes. The remaining available capacity (unprotected capacity) can be used to allocate orders of any class. Thus, if an order of the low profitable class arrives, it will be accepted as long as the remaining available capacity is sufficient to allocate the order. To evaluate the efficiency of the policy, the profit achieved by the policy is compared with the profit obtained by the FCFS policy.

Under the assumption of fixed order sizes, Chevalier et al. (2015) obtain the optimal admission control policy by formulating the problem as a Markov decision process (MDP). Computing the optimal policy, however, can require high computational requirements. To reduce the complexity, the authors develop state reduction and aggregation heuristics. The admission control policies constructed by these heuristics are shown to be near-optimal and can be obtained very quickly.

Germs and van Foreest (2011, 2013) extend the CSLSP considered in van Foreest, Wijngaard, and van der Vaart (2010) for an order acceptance and scheduling problem with family dependent setups, profit margins and fixed order sizes. The authors use an MDP approach to find optimal order acceptance and scheduling policies. Given the curse of dimensionality involved with the MDP, a simple threshold heuristic policy is developed. The efficiency of such a policy is evaluated with respect to the optimal policy. Their analysis is, however, limited to small instances because the system state space of the considered problem explodes quickly.

In our paper, we consider an admission control problem with stochastic order sizes similar to Barut and Sridharan (2004, 2005). We present an MDP formulation in order to obtain the optimal admission control policy. Such a formulation compared to the ones presented in Germs and van Foreest (2011, 2013), and Chevalier et al. (2015), involves an additional dimension arising from the multiple possible sizes that orders of one class can request. Consequently, the time to compute the optimal policy when demand sizes are random increases even faster. Thus, we propose a threshold-based policy to reduce the cardinality of possible policies, and, therefore, the computational requirements. To this end, we derive an MDP formulation to generate such a policy. In order to reduce the system state space, we then extend the state reduction and aggregation heuristics provided by Chevalier et al. (2015) incorporating the additional dimension. We develop an extensive numerical study to examine the efficiency of the proposed formulations under different levels of demand variability. In particular, we investigate how different distributions of order sizes, number of possible order sizes and their frequencies of occurrence impact the profit achieved by the different policies. The numerical study provides the following interesting insights. Our results reveal that for the majority of instances considered the optimal steady-state policy has a threshold structure. We then demonstrate the superiority of the proposed threshold-based policies over the two policies mostly used in literature and practice: the PLB policy and the FCFS policy. In particular, the results show that our proposed policies are superior in terms of profits generated and the robustness of the solutions to changes in different parameters. We further discuss the efficiency of the proposed heuristic policies for large instances for which the optimal policy cannot be obtained in reasonable amount of time. Last but not least we show that our policies are robust to changes of the actual demand from its estimation. Previous works on this topic have given very little attention to this important aspect for any practical implementation.

The remainder of this paper is organised as follows. In Section 2, we provide a description of the admission control problem with stochastic order sizes. In Section 3, we introduce different methodologies to obtain admission control policies. In Section 4, we provide insights into the structure of the optimal policy (Section 4.1), we describe the numerical experiments conducted (Section 4.2), we discuss the efficiency of the proposed algorithms in terms of the computational time (Section 4.3) and the generated profit (Section 4.4), and finally, we examine the efficiency of the proposed algorithms in case of an error estimate in demands (Section 4.5). In Section 5, we provide our concluding remarks.

2. Model formulation

We consider the admission control problem for a firm serving two distinct customer classes with different profit margins, order sizes and lead times. The planning horizon is infinite and consists of discrete periods. In each period, there is a unit-processing capacity. This setting would correspond for example to a situation where an industrial equipment company has one installation crew. Each day the crew can only be dispatched to one site, either for (emergency) repair work (i.e. urgent order) or installations (normal order).

At the beginning of each period, an order of class $k = \{1, 2\}$ arrives with probability p_k and size D_k (measured in terms of time units of processing). The arrivals and order sizes D_k (non-negative and integer valued) are $i.i.d.$ random variables. We assume that when an order is placed, processing times are known with certainty i.e. D_k can take value $B_{k,v}$ with probability $q_{k,v}$ for $v = 1, 2, \dots, n_k$, where n_k is the number of possible demand sizes for class k and $\sum_{v=1}^{n_k} q_{k,v} = 1$. We denote by $B_k = \{B_{k,1}, B_{k,2}, \dots, B_{k,n_k}\}$ and $Q_k = \{q_{k,1}, q_{k,2}, \dots, q_{k,n_k}\}$ the sets of order sizes and probabilities of their occurrence for class k, respectively. We represent by vector $\mathbf{D} = (D_1, D_2)$ the amount of demand requested for class 1 and class 2 in a period. This assumption represents the fact that for the type of service we have in mind for this model, the work content is determined at the time of the order; if it is a new installation, the specifications are well known, and if it is a repair work, the components to be repaired have already been identified. In either case, the uncertainty remaining is rather low and is neglected in our model. The profit margin realised with an order is $r_k D_k$ where r_k is the margin per unit capacity for orders of class k. Lead time of customer class k is L_k. We assume that $r_1 > r_2$ and $L_1 < L_2$.

Our objective is to decide whether to accept or reject the coming customer request in order to maximise the company's long-run net profit. The admission decision of the coming customer request depends on the amount of available capacity to accommodate the arriving order, its characteristics and the future demands. If the customer request is accepted it has to be processed before its due date (no tardiness is allowed). This assumption is made for tractability reasons and it corresponds to the fact that no lateness should be planned when an order is accepted. If some residual uncertainty would delay an order, this would be handled by some recourse actions such as overtime and outsourcing.

In this work, we consider that the company assigns higher priority to order of class 1 because of higher level of emergency compared to class 2. Thus, we focus only on selective admission decisions regarding class 2 orders while class 1 orders will be accepted as long as there is a sufficient available capacity (see also Barut and Sridharan 2004, 2005; Gupta and Wang 2007; Chevalier et al. 2015). This assumption suits well for many applications, such as the example of the industrial equipment company. The company can face an urgent order for repair work on the previously installed equipment when the customer experiences a breakdown during the warranty period of the equipment. Such maintenance activities are often specified in the contractual agreements when the equipment is purchased. Thus, company assigns higher priority to such orders even-tough, there may be situations where it can be more beneficial to make selective admission control decisions regarding urgent orders.

For the modelling purposes, we consider the following sequence of events. We assume that at the beginning of a period an order of the low-margin class (class 2) arrives before an order of the high-margin class (class 1). Assuming any other sequence of arrivals does not complicate the model and has a marginal impact on our results as discussed in Appendix 1. Upon the arrival of each order, the company makes the admission control decisions, and, then produces in the current period if there is a backlog of not-processed orders. The production is carried out according to the earliest due date scheduling policy.

Since admission control decisions for class 2 orders are made at discrete points in time, it is reasonable to assume that the time horizon can be divided into enough small time intervals to have negligible probabilities that multiple arrivals of the same class occur. Such assumption is widely made in revenue management literature see e.g. Meissner and Strauss (2012), Adelman (2007) and Defregger and Kuhn (2007).

3. Analysis of the admission control problem

In Sections 3.1 and 3.2, we present how the optimal admission policy and the threshold-based admission policy are constructed. In Section 3.3, we propose the approximate MDP formulations based on the state reduction via aggregation, and in Section 3.4, we describe the myopic method for obtaining the PLB policy which serves as our benchmark policy.

3.1 *The optimal policy*

In this section, we describe the state of the system at the decision point, explain the unconstrained policy, and present the MDP formulation used to obtain the optimal admission policy.

3.1.1 *Reservation vectors*

In order to represent the usage of future capacity in the MDP formulation upon the order acceptance/refusal of the coming orders and production in the current period, we define three reservation and corresponding cumulative (available capacity) vectors. The characterisation and notations of the vectors correspond to the ones in the revenue management literature (see e.g. Chevalier et al. 2015).

We represent by reservation vector $\mathbf{x}$ the system state at the start of the current period, before arrivals of regular and urgent orders. Element $\mathbf{x}[0]$ denotes the amount of capacity already reserved for non-processed orders within L_1 ($\mathbf{x}[0] \leq L_1$). For each $j = 1, 2, \ldots, L_2 - L_1$, element $\mathbf{x}[j]$ denotes whether the capacity in $(L_1 + j)$th period is reserved ($\mathbf{x}[j] = 1$) or available ($\mathbf{x}[j] = 0$). The capacity in the last period of L_2 is always available at the start of the current period, because there is no tardiness allowed, i.e. $\mathbf{x}[L_2 - L_1] = 0$. Thus, the size of the system state space is $(L_1 + 1) \cdot 2^{L_2 - L_1 - 1}$.

We represent by reservation vector $\widetilde{\mathbf{x}}$ the state that the system will occupy upon the admission decision related to the class 2 demand denoted by $\mathbf{a}$. Thus, this vector is a function of $\mathbf{x}$, D_2 and $\mathbf{a}$, i.e. $\widetilde{\mathbf{x}} = \widetilde{\mathbf{x}}(\mathbf{x}, D_2, \mathbf{a})$.

We represent by reservation vector $\widehat{\mathbf{x}}$, the state that the system will occupy at the start of next period, after acceptance/rejection of demand of class 1 and production in the current period. This vector is updated from $\widetilde{\mathbf{x}}$ and, thus, is a function of $\mathbf{x}$, $\mathbf{D}$ and $\mathbf{a}$, i.e. $\widehat{\mathbf{x}} = \widehat{\mathbf{x}}(\mathbf{x}, \mathbf{D}, \mathbf{a})$.

With our reservation vectors, we aggregate the information about reserved capacity within L_1, represented by $\mathbf{x}[0]$, $\widetilde{\mathbf{x}}[0]$ and $\widehat{\mathbf{x}}[0]$, and accurately represent whether the capacity in each period from $L_1 + 1$ to L_2 is reserved or not. This is because of the following reasons. To accept/reject the coming order of class 1, the distribution of the available capacity within L_1 is not relevant because the sequential admission process ensures feasibility. The only information needed to accept the coming order of class 1 is the amount of available capacity within L_1. However, to accept/reject the coming order of class 2, it is insufficient only to calculate the total amount of available capacity within L_2 because the available capacity within L_2 consists of available capacity within L_1. Thus, it is important to keep track of distributional information in each period from $L_1 + 1$ to L_2 in order to accurately calculate the amount of available capacity within L_1 when transiting between states.[1]

The corresponding cumulative available capacity vectors of vectors $\mathbf{x}$, $\tilde{\mathbf{x}}$ and $\widehat{\mathbf{x}}$ are denoted by $\mathbf{y}$, $\widetilde{\mathbf{y}}$ and $\widehat{\mathbf{y}}$, respectively. For instance, $\mathbf{y}[j]$ for $j = 0, 1, \ldots, L_2 - L_1$ denotes the *total* available capacity until the $(L_1 + j)$th period, and can be obtained as follows $\mathbf{y}[j] = L_1 + j - \sum_{i=0}^{j} \mathbf{x}[i]$. We introduce the cumulative available vectors in order to facilitate the notations related to transitions between one system state to another in the MDP formulation presented in Section 3.1.3.

3.1.2 *Unconstrained acceptance policy*

Before describing the unconstrained policy for the acceptance/rejection of orders, lets first assume that the possible order sizes of class 2 are sorted in the non-descending order, i.e. $B_{2,1} \leq B_{2,2} \cdots \leq B_{2,n_2}$. We represent by vector $\mathbf{a}$ the admission decision for class 2, whose element $\mathbf{a}[v]$ denotes the admission decision for order size $B_{2,v}$, for $v = 1, 2, \ldots, n_2$. The order size $B_{2,v}$ is accepted if $\mathbf{a}[v] = 1$, and it is rejected if $\mathbf{a}[v] = 0$. Therefore, there are 2^{n_2} possible admission control policies per each state of the system. For example, consider that orders of class 2 require either small or high order size. Therefore, there are four possible admission policies per each state of the system: accept any order of class 2 ($\mathbf{a} = [1, 1]$), accept only a request for the small order size ($\mathbf{a} = [1, 0]$), accept only a request for the large order size ($\mathbf{a} = [0, 1]$), and reject any order of class 2 ($\mathbf{a} = [0, 0]$). The action space is, thus, $2^{n_2}(L_1 + 1) \cdot 2^{L_2 - L_1 - 1}$.

3.1.3 *MDP formulation for the optimal steady-state policy*

We formulate the maximisation of the average profit per period as an MDP by using the standard average reward criteria presented by the optimality Equation (1), and the constraints (2)–(6).

$$V(\mathbf{x}) + g_s = \mathbb{E}_{\mathbf{D}} \left\{ \max_{\mathbf{a}} \left\{ R_2(\mathbf{x}, D_2, \mathbf{a}) + R_1(\tilde{\mathbf{x}}(\mathbf{x}, D_2, \mathbf{a}), D_1) + V\left(\widehat{\mathbf{x}}(\mathbf{x}, \mathbf{D}, \mathbf{a})\right) \right\} \right\}, \quad \forall \mathbf{x}, \tag{1}$$

$$\widetilde{\mathbf{y}}[j] = \begin{cases} \mathbf{y}[j] - (D_2 - (\mathbf{y}[L_2 - L_1] - \mathbf{y}[j]))^+, \text{ if} \\ D_2 = B_{2,v}, \mathbf{a}[v] = 1 \text{ and } \mathbf{y}[L_2 - L_1] \geq D_2, \\ \mathbf{y}[j], \text{ otherwise,} \end{cases} \tag{2}$$
$$\text{for } j = 0, 1, \ldots, L_2 - L_1, \text{ and } v = 1, \ldots, n_2,$$

$$\widehat{\mathbf{y}}[j] = \begin{cases} \min\left\{\widetilde{\mathbf{y}}[j+1] - D_1, L_1 + j\right\}, \\ \quad \text{if } D_1 = B_{1,v}, \ \widetilde{\mathbf{y}}[0] \geq D_1, \\ \min\left\{\widetilde{\mathbf{y}}[j+1], L_1 + j\right\}, \text{ otherwise,} \end{cases} \tag{3}$$
$$\text{for } j = 0, 1, \ldots, L_2 - L_1 - 1, \text{ and } v = 1, \ldots, n_1,$$

$$\widehat{\mathbf{y}}[L_2 - L_1] = \widehat{\mathbf{y}}[L_2 - L_1 - 1] + 1, \tag{4}$$

$$R_2(\mathbf{x}, D_2, \mathbf{a}) = \begin{cases} r_2 \cdot D_2, & \text{if } D_2 = B_{2,v}, a[v] = 1 \\ & \quad \text{and } \mathbf{y}[L_2 - L_1] \geq D_2, \\ 0, & \text{otherwise}, \end{cases} \tag{5}$$
$$\text{for } v = 1, \ldots, n_2,$$

$$R_1(\tilde{\mathbf{x}}(\mathbf{x}, D_2, \mathbf{a}), D_1) = \begin{cases} r_1 \cdot D_1, & \text{if } D_1 = B_{1,v}, \\ & \quad \text{and } \tilde{\mathbf{y}}[0] \geq D_1, \\ 0, & \text{otherwise}, \end{cases} \tag{6}$$
$$\text{for } v = 1, \ldots, n_1.$$

In the solution of Equation (1), g_s represents the steady-state expected profit per period for the admission control problem with stochastic order sizes, $V(\mathbf{x})$ represents the value function if the system is in state $\mathbf{x}$, $R_2(\mathbf{x}, D_2, \mathbf{a})$ denotes the profit from class 2 demand when the system is in state $\mathbf{x}$ and the admission decision is $\mathbf{a}$, and $R_1(\tilde{\mathbf{x}}(\mathbf{x}, D_2, \mathbf{a}), D_1)$ denotes the profit from class 1 demand upon updating to state $\tilde{\mathbf{x}}$.

Equation (2) updates the cumulative available capacity by subtracting the class 2 order if the available capacity is sufficient to accommodate the order and the admission decision is to accept the order. The equation characterises the transition from system state $\mathbf{x}$ to $\tilde{\mathbf{x}}$ with the help of $\mathbf{y}$ to $\tilde{\mathbf{y}}$.[2] This is done by reserving the latest available periods within L_2 to accommodate the accepted order so that the maximum flexibility will be ensured when allocating future orders.

Equations (3) and (4) characterise the transition from system state $\tilde{\mathbf{x}}$ to the system state at the start of next period $\widehat{\mathbf{x}}$, with the help of their corresponding cumulative vectors $\tilde{\mathbf{y}}$ and $\widehat{\mathbf{y}}$. The equations update the reserved capacity by adding demand of class 1, subtracting a unit that is processed in the current period and shifting units of reserved capacities to one period ahead when updating to the next period. The minimum between two values in Equation (3) is used to impose that, in the new state of the system, at most $L_1 + j$ periods can be available until the $(L_1 + j)$th period [3]. Equation (4) is used to set the capacity in the last period of L_2 to be available when updating to $\hat{\mathbf{y}}$.

Equations (5) and (6) calculate the benefits obtained from the transitions.

Here, we provide an example to demonstrate the transitions presented by Equations (2)–(4) and the calculation of the benefits by using Equations (5) and (6). Consider that $L_1 = 3$, $L_2 = 5$, $r_1 = \$1$, $r_2 = \$0.5$ and that all capacity is available within L_2 at the beginning of the current period, i.e. $\mathbf{y} = [3, 4, 5]$. If the class 2 order arrives with size equal to 3 units of capacity and the decision is to accept it, we reserve the latest periods to accommodate the order which, consequently, leads to two units of available capacity within L_1. Thus, the cumulative available capacity upon accepting the class 2 order is updated to $\tilde{\mathbf{y}} = [2, 2, 2]$ (Equation (2)). If demand of class 1 arrives with size equal to 2 units of capacity, the order will be accepted (because $\tilde{\mathbf{y}}[0] \geq D_1$), the remaining two units of available capacity will be reserved, and one unit of order in L_1 will be processed with the capacity in the current period. Upon the update to the next period, one unit of reserved capacity is shifted from $L_1 + 1$ to L_1 and one unit from L_2 to $L_1 + 1$, leading to $\widehat{\mathbf{y}}[0] = 0$ and $\widehat{\mathbf{y}}[1] = 0$, respectively (Equation (3)). Finally, period $L_2 + 1$ in the current period will be period L_2 in the next period, thus the capacity in period L_2 will be available i.e. $\widehat{\mathbf{y}}[2] = 1$ (Equation (4)). Therefore, the available capacity at the beginning of the next period is $\widehat{\mathbf{y}} = [0, 0, 1]$. The corresponding reservation vectors are $\mathbf{x} = [0, 0, 0]$, $\tilde{\mathbf{x}} = [1, 1, 1]$, and $\widehat{\mathbf{x}} = [3, 1, 0]$. The total profit in the current period calculated by using Equations (5) and (6) is \$3.5.

As the goal is to maximise the expected profit, we find the optimal policy by using the following standard linear programming (LP) model:

$$\max \quad g_s \tag{7}$$
$$\text{s.t.} \quad V(\mathbf{x}) + g_s \leq \mathbb{E}_{\mathbf{D}} \left\{ R_2(\mathbf{x}, D_2, \mathbf{a}) + R_1(\tilde{\mathbf{x}}(\mathbf{x}, D_2, \mathbf{a}) + V\left(\widehat{\mathbf{x}}(\mathbf{x}, \mathbf{D}, \mathbf{a})\right) \right\}, \quad \forall \mathbf{x}, \text{ and } \forall \mathbf{a} \in A. \tag{8}$$

We stress some important remarks about the LP model. The variables in the LP formulation are $\{V(\mathbf{x})\}$ and g_s. The number of constraints is equal to $2^{n_2} \cdot (L_1 + 1) \cdot 2^{L_2 - L_1 - 1}$ and increases exponentially with $L_2 - L_1$ and the number of order sizes of class 2. Thus, obtaining the optimal admission policy from the MDP formulation described in (1) is a challenging task.

In our formulation, we consider only the admission decision for class 2; however, we can easily extend the MDP formulation to deal with the admission decision for class 1. Nevertheless, such modifications will increase significantly the complexity of the formulation. The action state space will increase to $2^{n_1 + n_2}(L_1 + 1)2^{L_2 - L_1 - 1}$ and consequently, the number of constraints in the LP model will correspond to this value. This is because when incorporating the admission decision for class 1, the number of elements of vector $\mathbf{a}$ will be equal to $n_1 + n_2$, where elements $1 \leq i \leq n_1$ and $n_1 + 1 \leq i \leq n_1 + n_2$

represent the admission decision for $B_{1,i}$ and $B_{2,i-n_1}$, respectively. Thus, the number of possible policies per each state of the system will be $2^{n_1+n_2}$.

3.2 *The threshold-based policy*

There are two main reasons to implement the threshold policy t in our approach. Constructing the threshold policy enables us to obtain additional insights about the operational conditions under which the optimal policy is of a threshold structure. Another reason stems from the simplicity of the threshold policy which facilitates the implementation of the approximate formulations. Instead of having 2^{n_2} admission policies for each state of the system, there are only $n_2 + 1$ threshold policies. For a given state $\mathbf{x}$, if $t = 0$ the threshold policy rejects all demand of class 2; and if $t = 1, \ldots, n_2$ the policy is accept any order size $B_{2,v}$ for $v \leq t$, and reject any order size for $v > t$.

3.2.1 *The MDP formulation for the threshold-based policy*

It is easy to modify the exact MDP formulation proposed in the previous subsection in order to provide the optimal policy t^* under the constraint that it has to be of a threshold type. For that reason we omit the presentation of the model. We denote by g_t^* the long-run average reward under the threshold policy obtained from the MDP formulation. This policy reduces the complexity of the exact MDP formulation by decreasing the number of constraints and the action space to $(n_2 + 1) \cdot (L_1 + 1) \cdot 2^{L_2-L_1-1}$.

3.3 *Heuristic policies based on state reduction and aggregation*

In order to reduce the system state space and thus, the complexity in obtaining the two proposed admission policies when the difference between lead times increases, we extend the two state reduction and aggregation algorithms proposed in Chevalier et al. (2015) to deal with the considered problem with stochastic order sizes. Here, we briefly describe the algorithms and we refer to Chevalier et al. (2015) for more details. Namely, the first algorithm, Full Aggregation Heuristic (FAH), aggregates all information within the interval from $L_1 + 1$ to L_2 (called Interval II) completely ignoring any distributional information. Thus, the system state represented in the FAH is defined by the aggregated reservation vector $\mathbf{x}_f = (\mathbf{x}_f[0], \mathbf{x}_f[1])$ in which $\mathbf{x}_f[0] = \mathbf{x}[0]$ corresponds to the total reserved capacity in Interval I and $\mathbf{x}_f[1] = \sum_{j=1}^{L_2-L_1} \mathbf{x}[j]$ corresponds to that in Interval II. The second algorithm, partial aggregation heuristic (PAH), keeps track of some distributional information from period $L_1 + 1$ and aggregates information of last periods. The parameter $z \in \{0, \ldots, L_2 + L_1 - 1\}$ controls the tracking accuracy of the distributional information in the disaggregated interval, i.e. it represents the number of periods for which we know whether a unit of capacity in each period starting from $L_1 + 1$ is reserved or not. For example, if $z = 0$ the resulting formulation is identical to the full aggregation, and if $z = 1$ we keep track of information about the availability of capacity in period $L_1 + 1$ and aggregate information of periods from $L_1 + 2$ to L_2.

In order to know whether with each update to the next period a unit of reserved capacity will be shifted from the aggregated interval (the aggregated interval refers to the interval with aggregated information in Interval II) or not, Chevalier et al. (2015) make three assumptions about how the reserved capacity is distributed within the aggregated interval. The two scenarios: the optimistic and pessimistic, assume that all reserved capacity within the aggregated interval is distributed as late as possible and as early as possible within this interval, respectively. For instance, under the FAH with the pessimistic scenario, with each update to the next period, one unit of reserved capacity will be shifted from the aggregated interval to Interval I (the lead time of class 1) as long as there is at least one unit of the reserved capacity in the aggregated interval. The third scenario, uniform scenario, assumes that a unit of reserved capacity in the aggregated interval will be shifted to the first interval with a probability assuming uniformly distributed units of reserved capacity.

We extend these heuristics with the three distributional scenarios for the considered admission control problem with stochastic order sizes. Thus, we modify these heuristics to obtain the unconstrained policy $\mathbf{a}$ and the threshold-based policy t for each aggregated state of the system. Acceptance policies can be obtained by the aggregated MDP formulations in a similar way as for the exact MDP formulation described in Section 3.1. We name the two modified heuristics that provide vector $\mathbf{a}$: the Stochastic Full Aggregation Heuristic (SFAH) and the Stochastic Partial Aggregation Heuristic (SPAH). We name the two modified algorithms that provide policy t: the threshold-based full aggregation heuristic (T-SFAH) and the threshold-based partial aggregation heuristic (T-SPAH).

3.4 *PLB policy*

In order to compare the efficiency of the proposed admission policies with the existing policies in the literature, we implement the myopic method which provides a PLB policy commonly applied for similar problems (see e.g. Barut and Sridharan 2004, 2005). The myopic method determines the portion of capacity to protect for high-margin class, q, that maximises the expected net profit within L_2. The expected net profit within L_2 is computed by the following expression: $r_1 \cdot E[\min(\bar{D}_1, \max(q, y[L_2]-\bar{D}_2))] + r_2 \cdot E[\min(\bar{D}_2, y[L_2]-q)]$, where the random variable $\bar{D}_k$ represents the amount of demand of class $k = \{1, 2\}$ during L_2 periods.

The myopic method requires the computation of the expected demand of each class. In order to calculate the expected demand, we first make some additional definition. Let $a_k = (a_{k,1}, \ldots, a_{k,n_k})$ be the vector of arrivals within L_2 such that $a_{k,v}$ is the number of arrivals of order size v of class k within L_2, and $A_k = (A_{k,1}, \ldots, A_{k,n_k})$ be the vector of random variables related to the number of arrivals of order size v of class k within L_2. Given the vector $a_k = (a_{k,1}, \ldots, a_{k,n_k})$, we can calculate $P(A_k = a_k)$, by expression (9).

$$P(A_k = a_k) = \frac{L_2!}{(L_2 - \sum_{v=1}^{n_k} a_{k,v})! \prod_{v=1}^{n_k} (a_{k,v}!)} \cdot p_k^{\sum_{v=1}^{n_k} a_{k,v}} \cdot \\ (1 - p_k^{L_2 - \sum_v^{n_k} a_{k,v}}) \cdot \prod_{v=1}^{n_k} q_{k,v}{}^{a_{k,v}}. \tag{9}$$

Thus, we can write the cumulative distribution function of demand of class k within L_2, $P(\bar{D}_k \leq d)$ as given by expression (10).

$$P(\bar{D}_k \leq d) = \sum_{a_{k,1}=0}^{f_{k,1}} \cdots \sum_{a_{k,n_k}=0}^{f_{k,n_k}} P(A_k = a_k), \tag{10}$$

where $f_{k,v} = \left\lfloor \dfrac{d - \sum_{j=1}^{v-1} a_{k,j} \cdot B_{k,j}}{B_{k,v}} \right\rfloor$.

4. Numerical study

The main objective of this section is to investigate whether our proposed formulations for the admission control problem with stochastic order sizes are efficient in terms of computational time and relative profits. Specifically, we focus on the performance of the threshold policy implemented by the MDP and the aggregation heuristics. Firstly, we compare the performance of the threshold-based policy obtained from the MDP with the optimal policy under the various system parameters (Section 4.1). We will show that the threshold policy provides the optimal performance for most instances. Given the promising insights obtained, we develop a numerical experiment to investigate the efficiency of the threshold aggregation algorithms (Section 4.2). In Section 4.3, we measure the reduction in the computational time to find the threshold policies with respect to the unconstrained policies obtained by the exact (the unconstrained policies correspond to optimal policies) and approximate formulations. In Section 4.4, we compare the different policies in terms of the expected profits. In Section 4.5, we discuss the robustness of solutions with respect to changes in actual demand from its estimation.

We first generate the acceptance policies from the different formulations by using the standard LP models. In order to determine the performance of a heuristic policy in terms of average gain per period, we then simulate the long-run net profit for the policy by generating demand realisations. The initial warm-up interval in our simulation consists of 100,000 periods. We then incorporate additional 100,000 periods for which we accumulate recorded net profits. This process is repeated until the average net profit converges with a precision of 0.001%. The simulation study was implemented in the Java language and LP models are solved using Gurobi 4.6.1 (http://www.gurobi.com/). The computer used is a 6-Core Intel Xeon 2 × 2.66 GHz with 48 GB of RAM.

Here, we also provide additional notations related to heuristic policies. $g^*_{s,i}$ and $\hat{g}_{s,i}$ denote the expected and simulated long-run profits, respectively, achieved by the SFAH under the distributional scenario $i = \{o, p, r\}$, where subscripts o, p and r refer to the optimistic, pessimistic and uniform scenario. $g^*_{s,i}(z)$ and $\hat{g}_{s,i}(z)$ denote the expected and simulated long-run profits achieved by the SPAH under the distributional scenario $i = \{o, p, r\}$ and level of disaggregation z. To indicate the long-run profits achieved by the T-SFAH and T-SPAH, we use subscript t instead of s in the above notations.

4.1 *Structure of the optimal policy*

In order to obtain insights into the structure of the optimal policy, we perform a full-factorial design of instances with parameters related to class 2: $L_2 \in \{3, 4, 5, 6\}, n_2 \in \{2, 3, \ldots, L_2-1\}, D_2 \in \{1, 2, \ldots, L_2-1\}, p_2 \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$, $q_{2,v} \in \{0.0, 0.1, \ldots, 1 - \sum_{l=1}^{v-1} q_{2,l}\}$ for $v \in \{1, \ldots, L_2 - 1\}$ and $r_2 \in \{0.1, 0.2, \ldots, 0.9\}$; and parameters related to class 1: $L_1 \in \{1, 2, \ldots, L_2 - 1\}, n_1 = 1, D_1 \in \{1, 2 \ldots L_1\}, p_1 \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ and $r_1 = 1$.

The results show that the optimal policy is always a threshold when $L_1 = 1$, $D_1 = 1$, $r_2 < 0.3$, $p_1 = 0.9$ or $p_2 = 0.9$. Clearly, when r_2 is very small the decision is to reject class 2 because the benefits of accepting class 2 are much lower than its opportunity cost, thus, the optimal and the threshold policy are the same. For the other instances, the threshold structure cannot be guaranteed. The optimal policy, however, has a threshold structure in 98.99% of instances and the optimality gap of the best threshold policy never exceeds 0.133%.

Analysing in more detail the structure of optimal policies, we note that the optimal policy is a threshold policy for almost every state. For a very small number of states, there is a combinatorial effect involved based on the sizes of orders of class 2 that would make some order sizes more or less attractive. For instance, the policy may refuse a medium order size and accept the small and large order sizes in order to reduce the probability of having idle periods in the future (i.e. accepting the medium size will lead to losing the opportunity to accept a high-margin order in the current period and a large size order in the next period).

4.2 *Experiment design*

We develop two numerical experiments to examine the efficiency of the proposed formulations under various operational conditions. Our objectives are the following:

- To investigate how the number of different order sizes influences the quality of the acceptance policies (Experiment 1).
- To investigate how heterogeneity in order sizes and probability of their occurrence influence the performance of the different policies (Experiment 2).
- To examine the efficiency of the proposed heuristics for large instances of the problem (Experiment 1).

We measure the performance achieved by each policy in terms of the CPU time and the profit generated with respect to the optimal profit (g_s^*). For large instances, for which the optimal profit cannot be obtained in a reasonable amount of time, the profits achieved by different policies are compared with $g_{s,p}^*(z)$ and $g_{s,o}^*(z)$ which represent the lower and upper bounds on the optimal profit (see Chevalier et al. 2015). We test the aggregation heuristic policies for different disaggregation levels $z \in \{0, 2, 4, 6\}$.

The instances for the two experiments are characterised by the following attributes:

- *Profit structure* (ρ), it is the ratio of profit margins of the two classes, $\rho = r_2/r_1$. We fix $r_1 = 1$ and we test the following values for ρ i.e. $\rho \in \{0.25, 0.50, 0.75\}$ in order to explore high, moderate and low differences in the profit structure, respectively.
- *Ratio of total demands* (β). Ratio of total demands is obtained as follows $\beta = (p_2 \cdot TB_2)/(p_1 \cdot TB_1)$, where $TB_k = \sum_{v=1}^{n_k} q_{k,v} \cdot B_{k,v}$ and $k = \{1, 2\}$. We choose the following values of $\beta \in \{1/2, 1, 2/1\}$ (i.e. the expected demand of one class is two times greater than the other, or the expected demand rates of both classes are equal).
- *Global demand rate* (τ). The global demand rate is obtained by $\tau = p_1 \cdot TB_1 + p_2 \cdot TB_2$. We choose the following values for $\tau \in \{1.2, 1.6, 2.0\}$.
- *Lead-time structure* (L_1 and L_2). Given that increasing the difference between the lead times leads to higher computational complexity and higher operational flexibility (i.e. higher difference between L_2 and the average order size of class 2), we fix $L_1 = 6$ and vary L_2. Therefore, we chose two values of lead time of class 2: $L_2 = \{3 \cdot L_1, 5 \cdot L_1\}$ to reflect smaller and higher difference between lead times.
- *Demand structure* (n_k, D_k, $q_{k,v}$, where $k = \{1, 2\}$ and $v = 1, \ldots, n_k$). The number of order sizes of each class, their volumes and frequencies of occurrence are set depending on each objective of the experimental design.

The profit structure, ratio of total demand and global demand rate are common experimental conditions (CEC) for all objectives of the experimental design. There are in total 27 (i.e. $3 \times 3 \times 3$) CEC.

- *Experiment 1*: In order to investigate the efficiency of the proposed heuristics when the number of order sizes vary, we consider that order sizes of class k have the same probability of occurrence, and that the average order size, the smallest and the largest order size have the same values for different n_k. In Table 1, we present how we set the number and volumes of order sizes of each class. We consider in total 216 instances.

Table 1. Setting the number and values of order sizes and the lead time of each class in Experiment 1.

Attributes	No. of levels	Values
n_1, D_1	2	3, {1, 3, 5}
		5, {1, 2, 3, 4, 5}
n_2, D_2	2	3, {6, 9, 12}
		7, {6, 7, 8, 9, 10, 11, 12}
L_1, L_2	2	6, 18
		6, 30
Total conditions:	Total $CEC \times 2 \times 2 \times 2 = 216$	

- *Experiment 2*: In order to investigate how heterogeneity in the demand of each class influences the efficiency of the proposed heuristics, we fix $n_k = 3$ and $L_2 = 18$, and consider different levels of heterogeneity in order sizes and their probabilities of occurrence. We consider two sets of order sizes of class 1: set L with low differences in order sizes (the difference between successive order sizes is 1), and set M with medium differences in order sizes (the difference is 2). Similarly, for class 2, we consider the following sets: set L with low difference in order sizes (the difference between successive order sizes is 1), set M with medium difference in order sizes (the difference is 4) and set H with high difference (the difference is 7). Regarding the probabilities of occurrence of each class, we choose the following sets: a set with equal probabilities of occurrence (set E) and three sets in which one of the order sizes has four times higher probability of occurrence than the other two (sets H^1, H^2 and H^3, where set H^v denotes that the v-th order size of a class has the highest probability of occurrence). In Table 2, we present values of the order sizes and probabilities of their occurrence. There are in total 2592 instances in Experiment 2.

Table 2. Setting the values of order sizes and probability of their occurrence in Experiment 2.

Set of attributes	Number of levels	Sets	Values
Q_k	4	E	(2/6, 2/6, 2/6)
		H^1	(4/6, 1/6, 1/6)
		H^2	(1/6, 4/6, 1/6)
		H^3	(1/6, 1/6, 4/6)
B_1	2	L	(2, 3, 4)
		M	(1, 3, 5)
B_2	3	L	(8, 9, 10)
		M	(5, 9, 13)
		H	(2, 9, 16)
Total conditions:		Total $CEC \times (4 \times 2) \times (4 \times 3) = 2592$	

Note that, the values of p_1 and p_2 are functions of β, τ, $B_{k,v}$ and $q_{k,v}$ ($k = 1, 2$ and $v = 1, \dots, n_2$). Their values are derived from these parameters.

4.3 *Computational times*

In this section, we compare the average CPU times for determining the optimal policy, the threshold policies obtained by the MDP and the heuristic policies under the uniform scenario.

As already explained, the number of order sizes and the difference between lead times has a direct impact on the size of the state space of the system and, thus, the computational time. To explore this dependency, we show the average CPU times in Table 3 and the number of constraints in Table 4 for each combination of L_2 and n_2 of Experiment 1.[4]

Table 3. Average CPU (in seconds) time for the unconstrained (**a**) and the threshold (t) policies obtained by the MDP, and the aggregation heuristics under the uniform scenario.

		Exact	Aggregation heuristics			
(L_2, n_1)	(n_2, p^a)	MDP	$z = 0$	$z = 2$	$z = 4$	$z = 6$
(18, 3)	(3, t)	2.4	0.01	0.07	0.29	1.4
	(3, a)	4.7	0.02	0.10	0.42	1.8
	(7, t)	8.2	0.02	0.15	0.59	2.5
	(7, a)	134	0.30	1.09	6.18	23
(18, 5)	(3, t)	3.4	0.01	0.09	0.35	1.5
	(3, a)	6.6	0.02	0.13	0.51	2.0
	(7, t)	12	0.03	0.15	0.79	3.0
	(7, a)	194	0.40	1.57	8.44	32
(30, 3)	(3, t)	$*^b$	0.04	0.23	3.83	42
	(3, a)	$*$	0.07	0.35	4.01	49
	(7, t)	$*$	0.08	0.45	8.17	100
	(7, a)	$*$	0.74	4.56	49	276
(30, 5)	(3, t)	$*$	0.04	0.35	5.18	46
	(3, a)	$*$	0.07	0.49	6.27	60
	(7, t)	$*$	0.09	0.71	8.61	175
	(7, a)	$*$	1.12	5.51	58	360

[a] p denotes policy $p = \{a, t\}$.
[b] '$*$' symbolises instances for which the exact formulation cannot provide the admission policy within 1 h.

Table 4. Dimension of the admission problem (in 10^3) for the unconstrained (**a**) and the threshold (t) policies obtained by the MDP and the aggregation heuristics with the uniform scenario.

		Exact	Aggregation heuristics		
L_2	(n_2, p^a)	MDP	$z = 0$	$z = 4$	$z = 6$
18	(3, t)	57.3	0.3	3.6	10.7
	(3, a)	114.7	0.7	7.2	21.5
	(7, t)	114.7	0.7	7.2	21.5
	(7, a)	1835.0	10.7	114.7	344.1
30	(3, t)	234,881.0	0.7	9.0	32.3
	(3, a)	469,762.0	1.3	17.9	64.5
	(7, t)	469,762.0	1.3	17.9	64.5
	(7, a)	7,516,192.8	21.5	286.7	1032.2

[a] p denotes policy $p = \{a, t\}$.

We made the following observations based on Table 3. The SFAH and the T-SFAH deal well with the increase in the number of order sizes. In particular, the T-SFAH finds the best threshold policy in less than 0.09 s on average. As expected, the time needed to obtain the policy t is considerably shorter than the time to obtain the policy **a** by either the exact or approximate formulation. This is because the complexity of the T-SPAH is considerably reduced compared to the SPAH, especially for large instances. For example, when $L_2 = 30$, $n_2 = 7$ and $z = 6$ the number of constraints decreases from 1,032,192 to 64,512 (see Table 4). This consequently, leads to at least a halving of the average CPU time (see Table 3).

4.4 *Comparison among different policies*

In the following subsections, we compare the relative profits achieved by the threshold aggregation heuristics and the two benchmark policies: the PLB policy and the FCFS policy with respect to the optimal policy or upper and lower bounds. Note that the difference in results achieved by the aggregation heuristics with different policies is insignificant; therefore, here we present only results for the T-SFAH and the T-SPAH.

4.4.1 *Efficiency of the T-SFAH*

We compute the optimality gap of an acceptance policy obtained with method i as follows: $Gap_i = (g_s^* - \hat{g}_{t,i})/g_s^* \times 100\%$, for the T-SFAH under the scenario $i = \{o, p, r\}$, and $Gap_i = (g_s^* - \hat{g}_i)/g_s^* \times 100\%$ for $i = \{f, m\}$, where f refers to the FCFS policy, and m to the PLB policy. In order to evaluate the efficiency of the proposed policies with respect to the optimal steady-state policy, we fix $L_2 = 18$. As shown in Table 3, under this setting the construction of the optimal steady-state policy is computationally tractable.

First, we study the average optimality gap and the reliability of the different policies when the values of the parameters vary. Figure 1 shows the dispersion of the optimality gap of the analysed methods for the instances of Experiment 2.[5] As illustrated in Figure 1, the overall average optimality gaps of the $T - SFAH$ with the uniform and pessimistic scenarios are very similar and significantly lower than the overall average optimality gaps of the other policies, e.g. $Gap_r = 0.39\%$ compared to $Gap_m = 6.35\%$. We also observe that the implementation of the T-SFAH with the uniform and the pessimistic scenario is less sensitive to the different levels of asymmetry in the parameters.

We then study how the average optimality gaps of the T-SFAH under the two distributional scenarios change for different demand structures sets. Table 5 report on the average optimality gaps for the different sets of probabilities of occurrence and order sizes of Experiment 2.

We note from Table 5 that the uniform scenario is in average superior over the pessimistic scenario ($Gap_r = 0.344\%$ and $Gap_p = 0.439\%$). Moreover, the uniform scenario is less sensitive on different levels of heterogeneity in order sizes and probabilities of occurrence than the pessimistic scenario. Under the uniform scenario the average optimality gap is the smallest when the smallest size is the most frequent (i.e. $Gap_r = 0.290\%$ and $Gap_r = 0.301\%$ when $Q_2 = H^1$ and $Q_1 = H^1$, respectively).

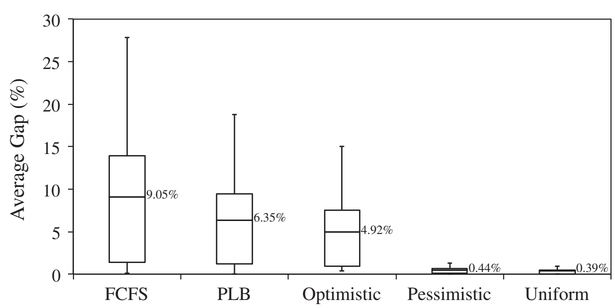

Figure 1. Dispersion of the optimality gap of different heuristic methods and distributional scenarios for the T-SFAH.

Table 5. Average optimality gap of the T-SFAH under the pessimistic and uniform scenarios for different demand structure sets.

Attribute		Class $k = 1$		Class $k = 2$	
class k	Sets	Gap_p (%)	Gap_r (%)	Gap_p (%)	Gap_r (%)
Q_k	E	0.432	0.342	0.437	0.358
	H^1	0.326	0.301	0.712	0.290
	H^2	0.413	0.351	0.373	0.308
	H^3	0.586	0.383	0.235	0.420
B_k	L	0.402	0.332	0.302	0.297
	M	0.476	0.356	0.503	0.397
	H			0.513	0.338
Overall		0.439	0.344	0.439	0.344

Table 6. The impact of the modeling assumption about the sequence of arrivals on the generated steady-state expected profit.

		Orders of class 1 arrives first			Order of any class can arrive first			# of instances with
D_2	ρ	Aver. gap (%)	Largest gap <0 (%)	Largest gap >0 (%)	Aver. gap (%)	Largest gap <0 (%)	Largest gap >0 (%)	the gap $\neq 0$
[1,5]	0.25	0.000	–	–	0.000	–	–	0
	0.50	0.000	–	–	0.000	–	–	0
	0.75	−0.007	−0.048	–	−0.003	−0.023	–	2
[5, 9]	0.25	0.000	–	–	0.000	–	–	0
	0.50	−0.035	−0.190	–	−0.017	−0.089	–	3
	0.75	−0.080	−0.168	–	−0.038	−0.081	–	8
[9, 13]	0.25	−0.067	−0.605	–	−0.034	−0.302	–	1
	0.50	−0.517	−1.219	–	−0.241	−0.591	–	6
	0.75	−0.223	−0.640	0.259	−0.102	−0.296	0.128	9
Total		−0.103	−1.219	0.259	−0.048	−0.591	0.128	29

The pessimistic scenario better represents the real situation for larger order sizes of class 2 (see Table 5). This can be explained as follows. If the large size of class 2 is accepted most of the capacity within Interval II will be reserved (especially, in case when demand exceeds capacity $\tau > 1$). Several periods will pass until the available capacity becomes sufficient to accommodate a new large order. During these periods, the already accepted orders will be moved several periods ahead leaving the last periods available. This means that with each update to the next period it is more likely that there will be a transfer of capacity from Interval II to Interval I, as assumed by the pessimistic scenario. Therefore, the pessimistic scenario tends to represents a real situation when $B_2 = L$ and $Q_2 = H^3$ (i.e. when the average order size is large). Therefore, only in these cases, the pessimistic scenario performs similarly or outperforms the uniform scenario (e.g. when $Q_2 = H^3$, $Gap_p = 0.235\%$ and $Gap_r = 0.420\%$, see Table 5).

As opposed to the pessimistic scenario, the uniform scenario better represents the real situation when order sizes of class 2 are smaller and/or more frequent. For example, $Gap_r = 0.290\%$ compared to $Gap_p = 0.712\%$ when $Q_2 = H^1$. This is because when order sizes of class 2 are small there is more possibility that available periods are not necessary the latest periods than in the case of large order sizes. Thus, the unit of capacity from the aggregated interval will not be always shifted to the first interval with each update to the next period.

4.4.2 *Efficiency of the T-SPAH*

In this section, we study how the systematic disaggregation of information about already accepted orders can improve the quality of the acceptance policies found. For this, we calculate the average optimality gap $Gap_i(z) = (g_s^* - \hat{g}_{t,i}(z))/g_s^* \times 100\%$ for the T-SPAH under the scenario $i = \{p, r\}$ and the level of disaggregation $z \in \{0, 2, 4, 6\}$.

In Figure 2, we present the average values of $Gap_i(z)$ for the 10% of instances with the highest gap and for all the instances of Experiment 2 (with $L_2 = 18$). The average optimality gap of the T-SFAH is greatly reduced by the T-SPAH while the computational time remains small (see Table 3). For example, when z increases from 0 to 6, the average optimality gap achieved under the uniform scenario for the worst instances decreases from 1.389% to 0.057% (see Figure 2).

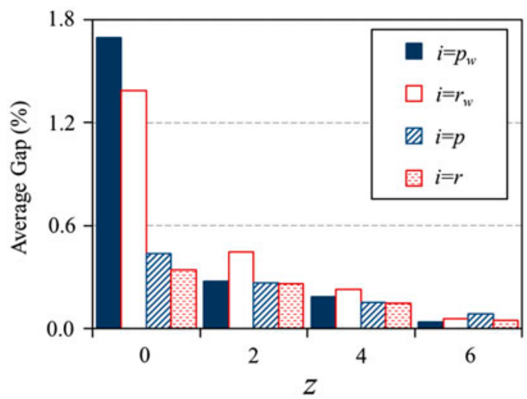

Figure 2. Average optimality gap of the admission policies under the two scenarios for all and the worst 10% of instances (indicated by subscript w) when z increases.

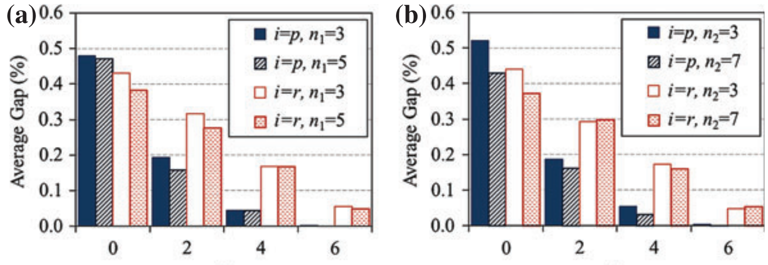

Figure 3. Average optimality gap of the admission policies under the two scenarios when z and n_k increases. (a) n_1 varies. (b) n_2 varies.

Finally, we investigate the accuracy of the approximation under the increased variety in order sizes. In Figure 3 we plot the average optimality gap of the T-SFAH and T-SPAH under the uniform and pessimistic scenario for Experiment 1 when the number of order sizes of each class and the level of disaggregation increase. We note that, in general, increasing the number of order sizes between the smallest and the largest order types of any class improves the performance of the aggregation heuristics. We also note from results presented in Figure 3 that the T-SFAH has the lower gap under the uniform scenario. On the other hand, the T-SPAH can better shrink the optimality gap under the pessimistic scenario when z increases. This is because when z increases the aggregated interval becomes smaller to accommodate class 2 orders (i.e. the difference between order sizes of class 2 and the aggregated interval become smaller). Thus, the same explanation provided in Section 4.4.1 for large order sizes of class 2 and T-SFAH follows.

4.4.3 *Efficiency of the T-SFAH for large instances*

As presented in Table 3, an advantage of the proposed heuristics is that they construct tractable policies for large instances for which the exact formulations suffer a state space explosion. Thus, we evaluate the efficiency of the studied heuristics when L_2 increases. The performance of the T-SFAH and T-SPAH is evaluated with respect to the lower and upper bound i.e. $g^*_{s,p}(z)$ and $g^*_{s,o}(z)$, respectively.

In Figure 4, we present the results from Experiment 1 with uniformly distributed demand, $n_1 = 5$ and $n_2 = 7$. The average net profits of the different policies are plotted in Figure 4(a) and (b) as a function of z for $L_2 = 18$ and $L_2 = 30$, respectively. We make several observations from the figures. The T-SFAH under the uniform scenario outperforms the pessimistic scenario when there is a higher operational flexibility (i.e. when $L_2 = 30$, see Figure 4(b)). Furthermore, the effect of increasing z seems much stronger on the quality of the bounds, and the performance of the T-SPAH under the pessimistic scenario than on the performance of the T-SPAH under the uniform scenario. We also note that the gap between the upper bound and the T-SPAH under the uniform scenario increases with L_2. Thus, when $L_2 = 30$ the average optimality gap of the T-SPAH under the uniform scenario will be less than 6.06% (see the average profits $g^*_{s,o}(z)$ and $g^*_{t,r}(z)$ for $z = 6$ in Figure 4(b)). Similar insights are obtained from the remaining instances.

4.5 *Robustness to changes in the actual demand from its estimation*

In the previous sections, we showed that our proposed approximate formulations are robust with respect to changes in parameters. Here, we study whether these formulations are robust with respect to changes in the realised demands from their estimations.

In practice, at the beginning of the planning horizon a company extracts its policy based on its estimation of arrivals. The resulting policy is then implemented for the realised demand. Given the stochastic nature of demand, some deviation may exist between the estimated and realised demands. In this section, we analyse how this deviation will affect the performance of the proposed heuristics.

To this end, we carry out additional numerical experiments. We consider that the probability of arrivals of class $k = \{1, 2\}$ is mistakenly expected to be $p^f_k = (1 + \epsilon_k) \times p^r_k$, where ϵ_k denotes the mean percentage error in demand estimation of class

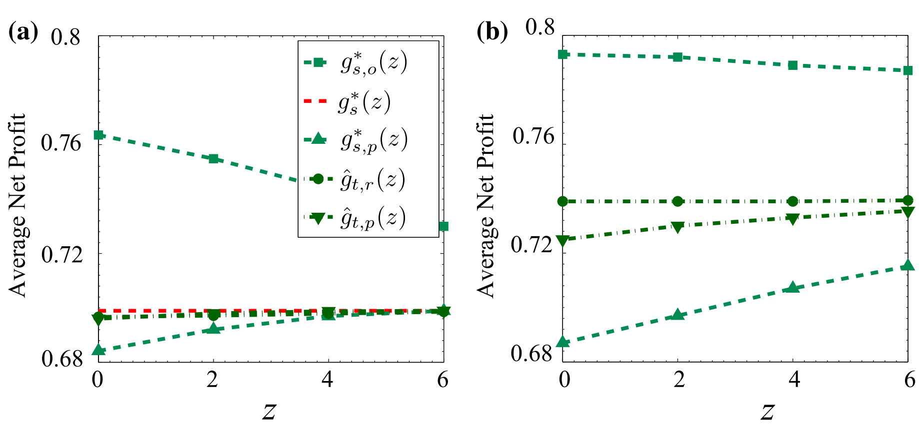

Figure 4. Effect of disaggregation on the average long-run profit for different L_2 and the uniformly distributed demand. (a) $L_2 = 18$. (b) $L_2 = 30$.

k, and p_k^r denotes the true (realised) probability of arrival. In order to explore different levels of the errors, we consider the following values $\epsilon_k \in [-0.5, 0.5]$, similar to Petrick et al. (2012) and Lee (1990). We then assume that the demand of each class follows a uniform distribution. Furthermore, we use the same common experiment conditions as in Section 4 for ρ, CT and β, and fix $L_1 = 7$, $L_2 = 15$, $D_1 \in \{2, 3, 4\}$ and $D_2 \in \{4, 5, 6\}$. Thus, p_k^r is obtained from these parameters.

In order to explain how we evaluate the robustness of the solutions in case of the demand deviation we need to introduce a few additional definitions. We denote by $g_s^{*,r} = g_s^*(p_1^r, p_2^r)$ the average long-run profit obtained by the optimal policy generated considering the true probabilities of arrivals; and $\breve{g}_{s,i}$ the average long-run profit obtained by simulation generating demand realisations with true probabilities (p_1^r and p_2^r) over the policy obtained by the SFAH under the scenario $i = \{r, p\}$ considering the estimated probabilities of arrivals (p_1^f and p_2^f). The efficiency of the SFAH under the scenario i for different errors in demand estimations is obtained as follows: $Gap_i(\epsilon_1, \epsilon_2) = (g_s^{*,r} - \breve{g}_{s,i})/g_s^{*,r} \times 100\%$. In other words, $Gap_i(\epsilon_1, \epsilon_2)$ represents the impact of the errors in estimations of demand on the performance of the SFAH under scenario $i = \{r, p\}$. We note that the smaller impact is, the more efficient the generated policy.

The average impact of the errors under the two scenarios and different combinations of errors are plotted in Figure 5. We note that in general the impact of errors for the aggregation heuristics is considerably smaller when $\epsilon_1 > 0$ (the forecast demand of class 1 is greater than its realisation) than when $\epsilon_1 < 0$ (the forecast demand class 1 is smaller than its realisation). For instance, when $\epsilon_1 > 0$, the largest average impact of the error achieved with the SFAH under the uniform scenario is 1.74% compared to the largest impact of 7.16% when $\epsilon_1 < 0\%$ (see Figure 5(a)). The results also show that errors related to class 1 have a higher impact on the efficiency of the heuristics than errors related to class 2.

The pessimistic scenario tends to protect more capacity for class 1 (i.e. it refuses more orders of class 2) than the uniform scenario because it underestimates the available capacity in Interval I (Chevalier et al. 2015). Therefore, the average performance degradation under the pessimistic scenario will be larger than under the uniform scenario when $\epsilon_1 > 0$ i.e. it will be lower when $\epsilon_1 < 0$ (see Figure 5(a) and (c)). This is because the SFAH under the pessimistic scenario will protect even more capacity when ϵ_1 increases for $\epsilon_1 > 0$ leading to excessive refusal of regular orders. This consequently causes greater average performance degradation. On the other hand, when $\epsilon_1 < 0$, i.e. the realised demand of class 1 is greater than the estimated, the pessimistic assumption will be preferable.

Given that the SPAH leads to a greater precision than the SFAH, the SPAH under the pessimistic leads to a larger performance degradation than the SFAH when the realised demand of class 1 is greater than its forecast value ($\epsilon_1 < 0$), and a smaller performance degradation for ($\epsilon_1 > 0$). On the opposite, when z increases the performance degradation under the uniform scenario will decrease when $\epsilon_1 < 0$ and will increase when $\epsilon_1 > 0$. The difference in performances between the two scenarios, however, decreases with z.

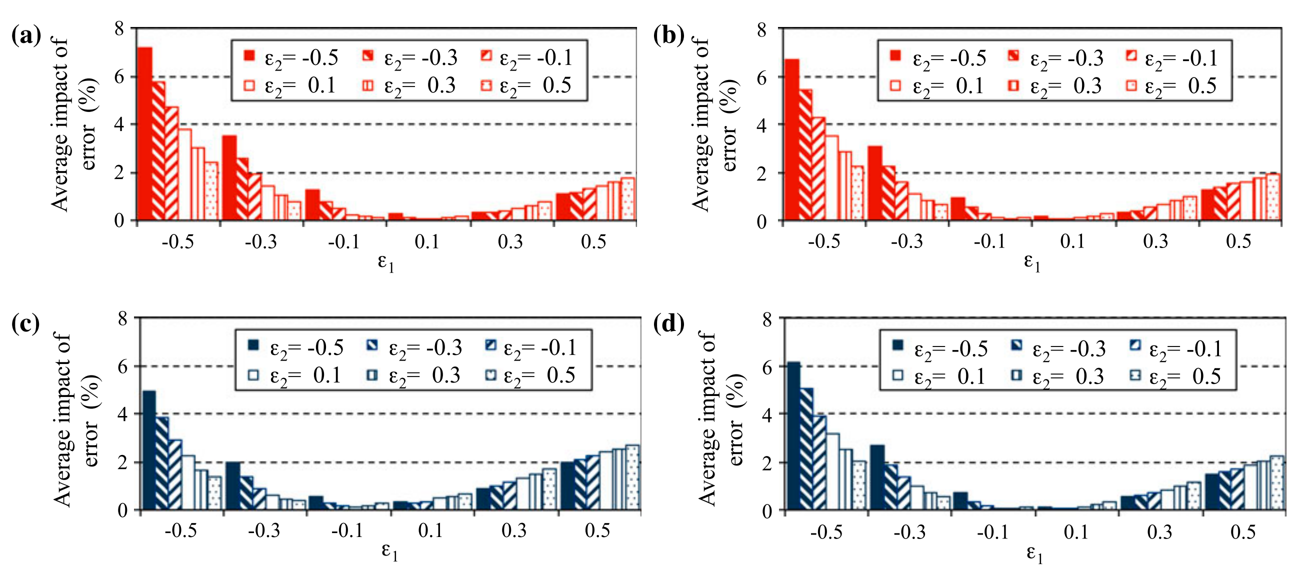

Figure 5. Robustness of the SFAH and the SPAH for different errors in demand estimation. (a) The SFAH under the uniform scenario. (b) The SPAH under the uniform scenario and $z = 4$. (c) The SFAH under the pessimistic scenario. (d) The SPAH under the pessimistic scenario and $z = 4$.

Finally, we can conclude that our results are quite robust with respect to changes in the realised demand from its estimation, especially when the estimated demand of class 1 is larger than the actual demand (in such a case the largest average impact of error is 2.68%, see Figure 5(c)). Regarding the decision related to the implementation of two heuristics in a real-business situation, we stress that the SPAH outperforms the SFAH under the uniform scenario when $\epsilon_1 < 0$ and the performance degradation of the SPAH for $\epsilon_1 > 0$ is insignificant.

5. Summary

In this paper, we present a revenue management formulation for the admission control and capacity allocation problem for a company facing random demands from two customer classes. The customer classes differ in net profits, lead times and variability in order sizes. The problem is formulated as a multidimensional MDP which involves high computational requirements for computing the optimal policy when the lead time of low-margin class and the number of possible order sizes increase.

In order to reduce the action state space we develop the MDP to obtain a threshold policy. Our results indicate that the structure of the optimal policy is a threshold for the vast majority of cases. The optimality gap of the threshold policy is quite small for the remaining instances.

In order to reduce the state space of the system when order sizes are stochastic, we extend the aggregation heuristics proposed in Chevalier et al. (2015). Our results show that the proposed heuristics provide near-optimal solutions very quickly. We further show that the solutions are robust with respect to changes in operational parameters and estimation errors for the actual demand.

In this paper, we consider that at most one order of each class can arrive in each period. The order can be entirely accepted or rejected depending on the admission control decision and the available capacity to accommodate the coming demand. A decision-maker can adapt our admission control problem to deal with multiple customer arrivals. If each customer requests a unit order size, the admission control problem becomes how many orders of class 2 to accept in each period. Thus, the admission policy will be of a threshold type. The new policy can take value $t' = 0, 1, \ldots, \max\{B_{2,n_2}, L_2\}$, where B_{k,n_k} now represents the maximum number of customers of class k that can arrive in a period. The amount of accepted orders of class 2 and class 1 per each state of the system is $\min\{t', D_2, y[L_2 - L_1]\}$ and $\min\{D_1, \tilde{y}[0]\}$, respectively, where D_1 and D_2 represent the amounts of total demand from class 1 and 2 in a period. Our dynamics of the MDP formulations presented by Equations (2)–(6) can be modified by taking into account the amounts of accepted orders of class of class 2 and class 1 provided above and considering policy t' instead of policy $\mathbf{a}$. As a result, the complexity of MDP and approximate formulations with multiple arrivals will not change significantly compared to our proposed formulations for the threshold-based policy t.

In case of multiple customer arrivals with multiple order sizes the admission control problem becomes much more complex. The complexity of the exact MDP formulation will significantly increase given that the optimal admission policy should include information about what combinations of the potential arriving orders to accept. One way to decrease the complexity of such a formulation is to provide a threshold policy per each state. Threshold policies obtained from such formulations can be, then, applied over the realised demands. Depending on the real business situation, the decision-maker can, for instance, accept the coming orders in the current period according to their arrivals as long as the sum of total demand accepted is smaller than or equal to minimum between the threshold value and the available capacity. Alternatively, the decision-maker can check all received orders in the current period and chose the most suitable combinations of orders to accept. How different are the solutions obtained by such policies with respect to the solution provided by the optimal policy, calls for further investigation.

Another extension of the problem is to consider more than two customer classes. The complexity of such a formulation will increase with the number of different classes since there is a need to aggregate information between the lead times of successive classes.

Acknowledgements

The authors are thankful to J. Will M. Bertrand for his useful suggestions which enriched the contribution of the paper. The authors are also grateful to anonymous referees for their valuable comments which improved the quality of the paper.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

Research that the authors conducted at Center for Operations Research and Econometrics, Université catholique de Louvain was partially supported by the ARC project ‘Managing Shared Resources in Supply Chains’ [ARC 08-13-008].

ORCID

Philippe Chevalier http://orcid.org/0000-0001-6443-624X

Notes

1. Note that when updating to the next period, period t within L_2 will become period $t-1$ at the beginning of the next period. Thus, an unit of reserved/available capacity in period t will be shifted to period $t-1$ with an update to the next period. With our representation of the system states, we can properly capture whether an unit of reserved capacity is shifted or not from L_1+1 to L_1 with each update to the next period.
2. For a given cumulative vector $\mathbf{y}$, its corresponding reservation vector $\mathbf{x}$ can be calculated as follows: $\mathbf{x}[0] = L_1 - \mathbf{y}[0]$, and for $j = 1, 2, \ldots, L_2 - L_1$, $\mathbf{x}[j] = \mathbf{y}[j-1] - \mathbf{y}[j] + 1$.
3. For instance, consider that $L_1 = 3$ and $L_2 = 5$ and that there is no arrival of any class at the beginning of a period. If all the capacity is available within L_2, i.e. $\mathbf{y} = \tilde{\mathbf{y}} = [3, 4, 5]$ the Equation(4) ensures that also $\hat{\mathbf{y}} = [3, 4, 5]$.
4. We exclude the presentation of the average CPU time of Experiment 2 because different probabilities of occurrence and higher differences in the order sizes do not have a significant influence on the CPU time.
5. Note that we show only the results from Experiment 2 with equal probabilities of occurrence (sets E). This is due to the long computational times needed to obtain the PLB policy with the myopic approach (the average CPU time is 2212 s). Our main insights based on the results presented on Figure 1 still hold for all considered instances of Experiment 2 under the FCFS policy and the T-SFAH policies.

References

Adelman, D. 2007. “Dynamic Bid Prices in Revenue Management.” *Operations Research* 55 (4): 647–661.

Balakrishnan, N., J. W. Patterson, and V. Sridharan. 1999. “Robustness of Capacity Rationing Policies.” *European Journal of Operational Research* 115: 328–338.

Balakrishnan, N., V. Sridharan, and J. W. Patterson. 1996. “Rationing Capacity between Two Product Classes.” *Decision Sciences* 27 (2): 185–214.

Barut, M., and V. Sridharan. 2004. “Design and Evaluation of a Dynamic Capacity Apportionment Procedure.” *European Journal of Operational Research* 155: 112–133.

Barut, M., and V. Sridharan. 2005. “Revenue Management in Order-driven Production Systems.” *Decision Sciences* 36 (2): 287–316.

Chevalier, P., A. Lamas, L. Lu, and T. Mlinar. 2015. “Revenue Management for Operations with Urgent Orders.” *European Journal of Operational Research* 240 (2): 476–487.

Defregger, F., and H. Kuhn. 2007. “Revenue Management for a Make-to-Order Company with Limited Capacity.” *OR Spectrum* 29: 137–156.

van Foreest, N. D., J. Wijngaard, and T. van der Vaart. 2010. “Scheduling and Order Acceptance for the Customised Stochastic Lot Scheduling Problem.” *International Journal of Production Research* 48 (12): 3561–3578.

Germs, R., and N. D. van Foreest. 2011. “Admission Policies for the Customized Stochastic Lot Scheduling Problem with Strict Due-dates.” *European Journal of Operational Research* 213: 375–383.

Germs, R., and N. D. van Foreest. 2013. “Order Acceptance and Scheduling Policies for a Make-to-Order Environment with Family-dependent Lead and Batch Setup Times.” *International Journal of Production Research* 51 (3): 940–951.

Gupta, D., and L. Wang. 2007. “Capacity Management for Contract Manufacturing.” *Operations Research* 55 (2): 367–377.

Hung, Y.-F., and T.-Y. Lee. 2010. “Capacity Rationing Decision Procedures with Order Profit as a Continuous Random Variable.” *International Journal of Production Economics* 125: 125–136.

Hung, Y.-F., P.-H. Tsai, and G.-H. Wu. 2014. “Application Extensions from the Stochastic Capacity Rationing Decision Approach.” *International Journal of Production Research* 152 (6): 1695–1710.

Ivanescu, C. V., J. C. Fransoo, and J. W. M. Bertrand. 2002. “Makespan Estimation and Order Acceptance in Batch Process Industries when Processing Times are Uncertain.” *OR Spectrum* 24 (4): 467–495.

Ivanescu, V. C., J. C. Fransoo, and J. W. M. Bertrand. 2006. “A Hybrid Policy for Order Acceptance in Batch Process Industries.” *OR Spectrum* 28: 199–222.

Lee, A. O. 1990. “Airline Reservations Forecasting – Probabilistic and Statistical Models of the Booking Process.” PhD thesis, MIT, Cambridge.

Meissner, J., and A. Strauss. 2012. “Network Revenue Management with Inventory-sensitive Bid Prices and Customer Choice.” *European Journal of Operational Research* 216 (2): 459–468.

Petrick, A., C. Steinhardt, J. Gönsch, and R. Klein. 2012. “Using Flexible Products to Cope with Demand Uncertainty in Revenue Management.” *OR Spectrum* 34: 215–242.

Appendix 1. Robustness to changes in the sequence of arrivals

In our work, we model the admission control problem by considering that in each period an order of class 2 arrives before an order of class 1. In reality, we could expect that in some periods an order of class 1 arrives before an order of class 2. In this section, we perform an additional numerical experiment to compare optimal profits under different sequences of arrivals. Thus, we consider two other possible scenarios: (1) class 1 orders always arrive before class 2 orders; and (2) random sequence of arrivals i.e. if orders of both classes arrive there is a probability π that the class 1 order arrives first and probability $1 - \pi$ that the class 2 order arrives first. We modify our MDP formulation in order to deal with each sequence of arrivals by changing how transitions are made between states presented by Equations (2) and (3). In case of randomness in the sequence of arrivals, the transition probabilities will depend not only on p_k and $q_{k,v}$ but also on probability π.

In the numerical experiment, we assume that the demand of each class follows a uniform distribution. Furthermore, we use the same common experiment conditions as in Section 4 for ρ, CT and β, and fix $L_1 = 7$, $L_2 = 14$, and $D_1 \in \{1, 2, 3, 4, 5\}$. We consider three possible sets for D_2 to investigate the impact of the modelling assumptions when the average order size of class 2 is small, medium and large, i.e. $D_2 \in \{1, 2, 3, 4, 5\}$, $D_2 \in \{5, 6, 7, 8, 9\}$, and $D_2 \in \{9, 10, 11, 12, 13\}$, respectively. The values of p_k for $k = 1, 2$ are obtained from these parameters. In case of the random sequence of arrivals, we consider that there is an equal probability that an order of any class arrives first. Overall, there are 81 different instances considered in the numerical experiment.

To compare optimal profits obtained under different sequences of arrivals we use as base the profit achieved by the optimal policy assuming class 2 orders arrive first. Thus, we calculate the gap between profits as $(g_s^* - g_{s,j}^*)/g_s^* \times 100\%$, where $g_{s,j}^*$ represents the optimal long-run expected profit assuming sequence of arrivals $j = 1, 2$, where $j = 1$ refers to the sequence when class 1 orders always arrive before the class 2 orders, and $j = 2$ refers to the random sequence of arrivals.

In Table 6 we present the average, the largest negative and the largest positive gaps for the two scenarios, and the number of instances with the gap different to zero for different values of ρ and demand sets D_2. Our results show the robustness of the solutions under different assumptions on the sequence of arrivals. In particular, the average gaps of assuming class 1 orders arrive first and assuming the random sequence of arrivals are $-0, 103\%$ and $-0, 048\%$, respectively. The largest gaps occur when the difference in profit margins is small ($\rho = 0.75$) and when the order sizes of class 2 are large ($D_2 \in \{9, 10, 11, 12, 13\}$). However, for those operational conditions the difference in the expected profits achieved by revenue management and the FCFS method is the smallest.

Finally, our results reveal that our proposed admission control approach considerably outperforms the FCFS method for any sequence of arrivals. The average optimality gaps of an acceptance policy obtained with the FCFS assuming class 2 orders arrive first, class 1 order arrive first, and the random sequence of arrivals are 11.4%, 9.6% , and 10.5%, respectively.