Operations Research Letters 45 (2017) 592–597

Contents lists available at ScienceDirect

Operations Research Letters

journal homepage: www.elsevier.com/locate/orl

Reservation, a tool to reduce the balking effect and the probability of delay

Benjamin Legros

EM Normandie, Laboratoire Métis, 64 Rue du Ranelagh, 75016, Paris, France

ARTICLE INFO

Article history:
Received 16 May 2017
Received in revised form 10 September 2017
Accepted 10 September 2017
Available online 28 September 2017

Keywords:
Queue
Balking
Reservation
Threshold policy

ABSTRACT

We investigate a threshold reservation policy implemented within a single customer's class. From an explicit performance analysis, we prove that the potential of reservation is into the reduction of the balking effect together with a higher server's utilization. However, it also may result in a higher expected waiting time and more abandonment. We conclude that reservation can be efficiently implemented in large systems, under high workload situations, with a low waiting aversion and a low impatience.

1. Introduction

In a system where all resources are occupied, the service of new arriving customers has to be delayed. The waiting aversion leads some customers to reconsider asking for service in case of a non-immediate service. This phenomenon is referred to as the *balking effect*. Balking can be significant in situations where customers know that they can easily find a similar service elsewhere, eventually without wait. For instance in a street with many restaurants, a significant proportion of customers may refuse waiting for an available table if they can find another one elsewhere. Another example is call centers; [4] shows that the balking phenomenon results in a significant loss of customers.

Since the loss of customers is undesirable, it may be useful to develop strategies which allow the service provider to maintain an amount of resource availability if preemption is not possible. In a context where a unique group of servers has to serve urgent and non-urgent customers, it has been shown that a reservation strategy can significantly improve the service level of urgent customers. The idea of a reservation strategy is to force some servers to remain available for new arriving urgent customers even if some non-urgent customers are waiting. This strategy is sometimes implemented in health care systems (see [6]) and in call centers with inbound and outbound calls (see [9]).

An important stream of literature is devoted to the analysis of such reservation strategies with two customer's classes. These strategies are referred to as *blending policies*. Some papers focus on performance evaluation, and others address the analysis of blending policies or staffing decisions. [5] develops various continuous Markov chain models for a call center with inbound and outbound calls. The authors consider a threshold policy and characterize the rate of outbound calls and the waiting time distribution of inbound calls. Other papers address the analysis of reservation policies. [7] and [2] prove that a threshold policy on the number of idle agents is optimal to maximize the outbound throughput under a service level constraint on the inbound waiting time. Similar results are also found in [13], for a non-stationary model where inbound calls arrive according to a non-homogeneous Poisson process or in [14] in a setting with a callback option. [15] considers a large call center with a reservation model and propose a logarithmic safety staffing rule, combined with a threshold control policy to ensure that agents' utilization is always close to one with always idle agents present.

In a broader perspective, reservation is the idea that some resources should remain idle when some work is available. For instance in a context with heterogeneous servers and the objective to minimize the time spent in the system, it is optimal to consider non-work conserving policies where slow servers are used only if the queue length exceeds some given thresholds. A large literature investigates this question through the so-called "slow-server" problem (e.g., see [11,12,16]). Other illustrations are overflow policies where agents are initially reserved for only one customer type but can treat another customer type in case of high congestion (e.g., see [1,10]). When there is a switching time or a switching cost to change the job type in service, it may also be preferred for a server to remain idle and ready to serve the same job

E-mail address: benjamin.legros@centraliens.net.

http://dx.doi.org/10.1016/j.orl.2017.09.003
0167-6377/© 2017 Elsevier B.V. All rights reserved.

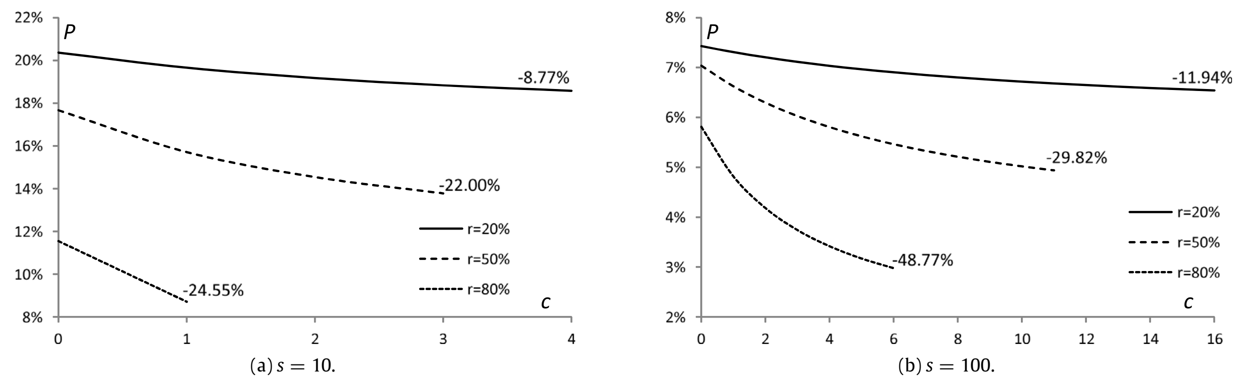

Fig. 1. P as a function of c ($\mu = 1, \lambda = s\mu$).

type instead of serving the other job type. For this purpose, cycling reservation strategies are studied (e.g., see [3,17]).

The particularity of our article compared to the existing ones on the same subject is to consider a threshold reservation strategy with a *unique class* of customers who may balk. The choice for a reservation threshold policy follows from the knowledge that this type of policies outperforms randomized policies since they take into account the system state. In addition, threshold policies are deterministic; for each state there is a unique action. This makes these policies easy to implement in practice. Finally, threshold policies are controlled by only one parameter which also makes them possible to analyze and implement.

We choose to adapt the threshold policy developed in the call blending references for a unique class of customers. We consider a multi-server single queue with infinite capacity and s identical, parallel servers. The arrival process of customers is Poisson with rate λ. Service times are independent and exponentially distributed with rate μ. At a customer's arrival, if at least one server is available then this customer is directly served, otherwise with probability r he/she accepts waiting and is routed to a first-come-first-served queue. The reservation policy is of threshold type. We define a threshold c on the number of servers. After a service completion, if the number of busy servers is higher than or equal to $s - c$ then no customer is routed to service. Otherwise, if the queue is non-empty then the first customer in line is routed to service. In other words, c servers are reserved for new arriving customers ($0 \le c < s$).

The objective of this article is to evaluate the performance of this system and determine the effect of reservation on the proportion of lost customers and on the waiting time. The idea is to show the motivations and the risks in implementing such a policy. In Section 2, we present the Markov chain associated to this system and develop a method to obtain the explicit expressions of the stationary probabilities. Section 3 is devoted to the impact of reservation on balking. We prove that reservation can reduce the proportion of lost customers. This might be unexpected by service providers. By forcing some servers not to work, the overall productivity may increase. The improvement can be significant in large systems, under high workload situations and with a low customers' waiting aversion. In Section 4, we develop a method to compute the Laplace–Stieltjes Transform (LST) of the waiting time distribution of an arriving customer. This allows us to obtain explicit expressions of the first and second moments of the waiting time. As expected, although reservation helps to reduce the probability of delay, it deteriorates the expected waiting time due to the existence of situations where a server may remain idle while a customer is waiting. In addition, in Section 5 we include abandonment in the model. From a numerical analysis, we show that reservation may be counterproductive if customers' impatience is high.

2. Stationary probabilities

In this section, we derive explicitly the stationary probabilities. The system is modeled using a two dimensional continuous-time Markov chain. We denote by (x, y) a state of the system for $0 \le x \le s$ and $y \ge 0$, where x represents the number of busy servers and y represents the number of customers in the queue. The state space S is $S = \{(0,0), (0,1), \dots, (0,s)\} \cup \{s-c, s-c+1, \dots, s\} \times \mathbb{N}^*$. We denote by $p_{x,y}$ the stationary probability to be in state (x, y) and by a the ratio λ/μ.

We next describe the 4 possible transitions in the Markov chain.

1. An arrival with rate λ while a server is available ($0 \le x < s$), which changes the state to $(x + 1, y)$. The number of busy server is increased by one.
2. An arrival with rate $r\lambda$ while all servers are busy ($x = s$), which changes the state to $(x, y + 1)$. The number of customers in the queue is increased by 1.
3. A service completion with rate $(s - c)\mu$ while the queue is not empty ($y > 0$) and the number of busy servers is equal to $s - c$ ($x = s - c$), which changes the state to $(x, y - 1)$. The number of customers in the queue is reduced by 1.
4. A service completion with rate $\min(s, x)\mu$ while the queue is empty ($y = 0$) or the queue is not empty but the number of busy servers is strictly higher than $s - c$ ($x > s - c, y > 0$), which changes the state to $(x - 1, y)$. The number of busy servers is reduced by 1.

The Markov chain is depicted in Fig. 1 of Section 1 of the online supplement. In Theorem 1, we give the stationary probabilities and the stability condition. The proof of Theorem 1 is given in Section 2 of the online supplement.

Theorem 1. *Under the stability condition $r\frac{(s-c-1)!}{s!}a^{c+1} < 1$, we have*

$$p_{x,0} = \frac{a^x}{x!}p_{0,0}, \text{ for } 0 \le x \le s - c, \tag{1}$$

$$p_{x,0} = \frac{a^x}{x!}\frac{1 + r\sum_{k=0}^{s-x-1}\frac{(x+k)!}{s!}a^{s-k-x}}{1 + r\sum_{k=0}^{c-1}\frac{(s-c+k)!}{s!}a^{c-k}}p_{0,0}, \text{ for } s - c \le x \le s, \tag{2}$$

$$p_{x,y} = \frac{a^x\left(1 - a^{s-(x+c+1)}\left[1 - r\frac{(s-c-1)!}{s!}a^{c+1}\right]\frac{\sum_{k=0}^{s-x-1}(x+k)!a^{-k}}{\sum_{k=0}^{c}(s-c-1+k)!a^{-k}}\right)}{rx!\sum_{k=0}^{c}\frac{(s-c-1+k)!}{s!}a^{c+1-k}} \times \left(\frac{r\sum_{k=0}^{c}\frac{(s-c-1+k)!}{s!}a^{c+1-k}}{1 + r\sum_{k=1}^{c}\frac{(s-c-1+k)!}{s!}a^{c+1-k}}\right)^{y+1} p_{0,0}, \tag{3}$$

for $s - c \le x \le s, y > 0$, with