Discrete-time analysis of the gated generalized multiple-vacation queue

Dieter Fiems, Joris Walraevens, Herwig Bruneel
SMACS Research Group, Vakgroep TELIN(TW07), Ghent University

Abstract—**We consider the discrete-time gated multiple-vacation queue. Vacations are modeled as independent random variables with distributions depending on the number of the immediately preceding vacations. Using a probability generating function approach, we focus on various performance measures such as moments of queue contents and customer delay in equilibrium. These measures are functions of a constant value which we obtain numerically.**

I. Introduction

Vacation models [1], [2] have proven to be a useful abstraction of server unavailability in cases where classes of customers contend for a single resource such as polling systems [3] and priority systems [4], or in cases where this resource is unreliable, e.g., maintenance models [5] and ARQ-systems [6].

The current contribution investigates the gated multiple-vacation queue in discrete time. Our generalized multiple-vacation queueing model allows to capture performance of a.o., the multiple-vacation, the single-vacation and the limited multiple-vacation gated queueing systems. The model under consideration extends the results from [7] both regarding arrival and vacation process and regarding the performance measures under consideration.

The outline is as follows. In the next section we describe the queueing system under consideration in more detail. The analysis is then presented in sections 3 and 4, whereas some special cases are considered in section 5. Numerical examples illustrate our results in section 6 and conclusions are drawn in section 7.

II. Model

We assume that time is divided into fixed length intervals (slots) and that service is synchronized with respect to slot boundaries, i.e., service of a customer cannot start during this customer's arrival slot. Both the number of customers arriving in the consecutive slots and the service times (in slots) of these consecutive customers constitute series of i.i.d. random variables with common probability mass functions a_n $(n \geq 0)$ and s_n $(n > 0)$ respectively and with corresponding probability generating functions $A(z)$ and $S(z)$ respectively.

There are 2 queues, separated by a gate. Arriving customers first wait in the queue before the gate and move in batch to the queue after the gate whenever the latter opens. This happens at the end of the last slot of a vacation period (see further). We refer to the queues before and after the gate as the secondary and the primary queue respectively, i.e., customers arrive in the secondary queue, move to the primary queue when the gate opens and are then served – in order of arrival (FIFO) – before leaving the system. Both primary and secondary queues have an infinite capacity.

A vacation starts when the primary queue empties, (i.e., since the end of the last vacation) and the gate opens at the end of each vacation. If there are no customers present in the primary queue upon returning from a vacation, the server immediately takes another vacation. This continues until the primary queue is no longer empty, i.e., we consider a multiple-vacation policy.

The consecutive vacation lengths (in slots) are modeled by means of a series of independent random variables with probability mass functions $v_i(n)$ $(n > 0)$ and corresponding probability generating functions $V_i(z)$ depending on i, the number of immediately preceding vacations. We assume there exists a finite upper bound L such that, $V_k(z) = V_l(z)$ for all $k, l \geq L$.

III. Queue contents

The system under consideration alternates between busy-periods – the system serves a customer – and vacation-periods – the system takes (possibly multiple) vacations. A cycle is defined as a busy period followed by a vacation period. During a busy period, all customers in the primary queue are being served, while the secondary queue is filling with the newly arriving customers. At the end of the busy period, the server enters the vacation period and at the end of the latter, all customers in the secondary queue are transferred to the primary queue. Clearly, the number of customers present in the system – i.e., in the primary and secondary queue – immediately after a cycle, equals the number of customers that arrived during this cycle as a vacation is only taken when all customers that arrived before the start of the cycle, are served.

Let c_k denote the slot following the k-th cycle and let U_i denote the number of customers in the primary queue at the beginning of slot i, then, one easily establishes,

$$U_{c_{k+1}} = \sum_{i=1}^{U_{c_k}} \sum_{j=1}^{S_i} A_{ij} + W_{k+1}, \tag{1}$$

where S_i denotes the service time of the i-th customer served during the k-th cycle, where A_{ij} denotes the number of arrivals during the j-th service slot of this customer and where W_{k+1} denotes the number of arrivals during the vacation period in the $(k+1)$-th cycle. Note that there are no customers present in the secondary queue at the beginning of a cycle. Let $U_{c_k}(z)$ denote the probability generating function of the number of customers in the system at the end of the k-th cycle, some standard z-transform manipulations then yield,

$$\begin{aligned} U_{c_{k+1}}(z) =& U_{c_k}\left(S\left(A\left(z\right)\right)\right) \overline{W_0}(z) \\ &+ U_{c_k}\left(S\left(x\, a_0\right)\right) \left(W_0(z) - \overline{W_0}(z)\right), \end{aligned} \tag{2}$$

All authors are with the SMACS Research Group, Vakgroep TELIN(TW07), Ghent University, St-Pietersnieuwstraat 41, 9000 Gent, Belgium, Phone: +32 9 264 89 02, Fax: +32 9 264 42 95, E-mails:{df,jw,hb}@telin.rug.ac.be.

where $W_0(z)$ and $\overline{W_0}(z)$ denote the probability generating functions corresponding to the number of arrivals during the vacation period of the $(k+1)$-th cycle given that there are no customers or given that there is at least one customer in the system at the end of the slot preceding the vacation period respectively.

As the presence of customers before the start of the vacation period implies the presence of customers at the end of the first vacation, the server takes only a single vacation, i.e., $\overline{W_0}(z) = V_0(A(z))$. If there are no customers in the system, the server keeps on taking vacations until there is at least one arrival. Conditioning on the number of necessary vacations then yields,

$$\begin{aligned} W_0(z) =& \sum_{i=0}^{L-1} \left(V_i(A(z)) - V_i(a_0)\right) \prod_{j=0}^{i-1} V_j(a_0) \\ &+ \frac{V_L(A(z)) - V_L(a_0)}{1 - V_L(a_0)} \prod_{j=0}^{L-1} V_j(a_0). \end{aligned} \tag{3}$$

Let $U_c(z) \triangleq \lim_{k\to\infty} U_{c_k}(z)$ denote the probability generating function corresponding to the number of customers at the end of a cycle in equilibrium. Similarly as in [7], one can prove that the latter exists whenever,

$$\rho = S'(1)\, A'(1) < 1, \tag{4}$$

where ρ denotes the system load. Under the assumption of equilibrium, equation (2) yields,

$$U_c(z) = U_c(S(A(z)))\, \overline{W_0}(z) + K\, (W_0(z) - \overline{W_0}(z)), \tag{5}$$

where $K = U_c(S(a_0))$ denotes the probability that the secondary queue is empty at the beginning of a vacation period. The former functional equation allows implicit determination of the various derivatives of $U_c(z)$ evaluated in $z = 1$ once K is determined.

The unknown K can be determined numerically as follows. Consider the series $z_i = S(A(z_{i-1}))$, $z_0 = 0$, $i > 0$. Given $\rho < 1$, one easily proves that this series converges to 1. Let $y_i \triangleq \frac{K}{U_c(z_i)}$, substitution of $z = z_i$ in equation (5) then yields,

$$y_{i+1} = \frac{\overline{W_0}(z_i)\, y_i}{1 + \left(\overline{W_0}(z_i) - W_0(z_i)\right)\, y_i}. \tag{6}$$

Further, note that $y_1 = 1$. Starting the recursion with $y_1 = 1$, allows us to determine $K = \lim_{i\to\infty} y_i$ numerically.

Given the system contents at the end of a random cycle, we can now easily retrieve the joint probability generating functions of the numbers of customers in the primary and secondary queue at other epochs. Let $U_d(z_1, z_2)$ denote the joint probability generating function of the number of customers in the primary and secondary queue at the beginning of a slot following a departure, then,

$$\begin{aligned} U_d(z_1, z_2) &= \mathrm{E}\left[z_1^{U_{d,1}}\, z_2^{U_{d,2}}\right] \\ &= \frac{1}{U_c'(1)}\, \mathrm{E}\left[\sum_{k=1}^{U_c} z_1^{U_c - k}\, z_2^{\sum_{i=1}^{k} \sum_{j=1}^{S_i} A_{ij}}\right] \\ &= S(A(z_2))\, \frac{U_c(S(A(z_2))) - U_c(z_1)}{U_c'(1)\, (S(A(z_2)) - z_1)}, \end{aligned} \tag{7}$$

where $U_{d,1}$ and $U_{d,2}$ denote the number of customers in the primary and secondary queue at a random departure epoch respectively and where U_c denotes the number of customers in the system at the end of a random cycle.

Let $U_s(z_1, z_2)$ denote the joint probability generating function of the number of customers in primary and secondary queue at the beginning of a slot where a customer starts service, then, one easily verifies,

$$U_s(z_1, z_2) = \frac{z_1}{S(A(z_2))}\, U_d(z_1, z_2), \tag{8}$$

as the customers arriving during this customer's service are stored in the secondary queue and as the customer itself leaves the primary queue after being served.

The joint probability generating function of these quantities at the beginning of random busy slots $U_b(z_1, z_2)$, is then given by,

$$\begin{aligned} U_b(z_1, z_2) &= \frac{1}{S'(1)}\, \mathrm{E}\left[\sum_{k=1}^{S} z_1^{U_{s,1}}\, z_2^{U_{s,2} + \sum_{i=1}^{k-1} A_i}\right] \\ &= U_s(z_1, z_2)\, \frac{S(A(z_2)) - 1}{S'(1)\, (A(z_2) - 1)}, \end{aligned} \tag{9}$$

where $U_{s,1}$ and $U_{s,2}$ denote the number of customers in primary and secondary queue at the beginning of a slot where a random customer starts service and where A_i denotes the number of arrivals in the system during the i-th slot of this customer's service time S.

Let $U(z_1, z_2)$ denote the joint probability generating function of primary and secondary queue contents at random slot boundaries, then, as the primary queue is empty during vacations,

$$U(z_1, z_2) = \frac{U_c'(1)\, S'(1)}{C'(1)}\, (U_b(z_1, z_2) - U_v(z_2)) + U_v(z_2) \tag{10}$$

where $U_v(z_2)$ denotes the probability generating function of the number of customers in the secondary queue at the beginning of a random vacation slot and where $C'(1)$ denotes the mean cycle length. The cycle length C equals the sum of the lengths of the busy and vacation periods,

$$C = \sum_{j=1}^{U_c} S_j + V, \tag{11}$$

where V denotes the vacation length. Some standard z-transform manipulations then yield,

$$C(z) = U_c(S(z))\, \overline{N_0}(z) + U_c(S(a_0\, z))(N_0(z) - \overline{N_0}(z)), \tag{12}$$

where $C(z)$ denotes the probability generating function corresponding to the cycle length and where $\overline{N_0}(z)$ and $N_0(z)$ denote the probability generating functions of the length of the vacation period given that there is at least a customer in the system before the start of the vacation period or given that this is not the case respectively. Similarly as for the number of arrivals during a vacation period we get, $\overline{N_0}(z) = V_0(z)$, and,

$$\begin{aligned} N_0(z) = &\sum_{i=0}^{L-1} \left(V_i(z) - V_i(a_0\, z)\right) \prod_{j=0}^{i-1} V_j(a_0\, z) \\ &+ \frac{V_L(z) - V_L(a_0\, z)}{1 - V_L(a_0\, z)} \prod_{j=0}^{L-1} V_j(a_0\, z). \end{aligned} \tag{13}$$

The first derivative of equation (12) for $z = 1$ then yields an explicit expression for the mean cycle length.

As the system under consideration is a single-server one with an i.i.d. arrival process, system contents at departure epochs and at random slot boundaries are related as follows [8],

$$U_d(z, z) = \frac{A(z) - 1}{A'(1)\,(z-1)}\, U(z, z). \tag{14}$$

Equations (10) – evaluated for $z_1 = z_2 = z$ – and (14) then allow determination of $U_v(z)$,

$$\begin{aligned} U_v(z) = &\frac{C'(1)\, A'(1)}{C'(1) - U_c'(1) S'(1)}\, \frac{z-1}{A(z)-1}\, U_d(z, z) \\ &- \frac{U_c'(1) S'(1)}{C'(1) - U_c'(1) S'(1)}\, U_b(z, z). \end{aligned} \tag{15}$$

Substitution of the former equation in equation (10) then yields an expression for the joint probability generating function of steady-state the number of customers before and after the gate at random slot boundaries.

IV. Customer delay

Customer delay is defined as the number of slots between the end of the slot a customer arrives in a queue and the end of the slot where that customer leaves the queue. As the customer only leaves the primary queue after being served, the delay in the primary queue includes the customer's service time.

As in [9], we first consider an alternative system where all arrivals in a slot are grouped to form a "batch-customer", i.e., we consider a system with Bernoulli "batch-customer" arrivals. Probability generating functions of the number of batch-customer arrivals per slot $A^*(z)$ and their service times $S^*(z)$ are then given by,

$$\begin{aligned} A^*(z) &= a_0 + (1 - a_0)\, z, \\ S^*(z) &= \frac{A(S(z)) - a_0}{1 - a_0}. \end{aligned} \tag{16}$$

Let $U_d^*(z_1, z_2)$ denote the probability generating function of the primary and secondary system contents at departure epochs for this alternative system. I.e., the latter is given by equation (7) given the arrival and service probability generating functions of (16). Now, consider a random batch-customer and let $D^*(z_1, z_2)$ denote the joint probability generating function of its delay in primary and secondary queue. All batch-customers that arrive during its delay in the secondary queue are moved to the primary queue along with the batch-customer under consideration, i.e., when the gate opens. Furthermore, all batch-customers that arrive during its delay in the primary queue are present in the secondary queue at its departure. As a result the probability generating functions of batch-customer delay and queue contents at batch-customer departure epochs, are easily related,

$$D^*(A^*(z_2), A^*(z_1)) = U_d^*(z_1, z_2). \tag{17}$$

We can now relate the delay of a customer to the delay of its batch. Clearly, delay in the secondary queue of a customer equals the delay of its batch as they enter and leave this queue at the same time. Let waiting time of a customer denote the number of slots between the end of its arrival slot and the beginning of the slot where this customer starts service. Waiting time in the primary queue of a customer then equals the sum of the waiting time of its batch, and the service times of all customers that arrived during the same slot as, but prior to the customer. Finally customer delay in the primary queue equals its waiting time augmented by its service time. Keeping these observations in mind, one gets,

$$D(z_1, z_2) = \frac{S(z_1)\,(A(S(z_1)) - 1)}{A'(1)\,(S(z_1) - 1)}\, \frac{U_d^*\left(\frac{z_2 - a_0}{1 - a_0}, \frac{z_1 - a_0}{1 - a_0}\right)}{S^*(z_1)}. \tag{18}$$

Taking the appropriate derivatives of (18) and (10), one easily confirms the discrete-time equivalent of Little's result [10], [11] for both the primary and secondary queue and for the complete system.

V. Special cases

As noted in the introduction, the model under consideration allows to capture performance of more specific models, a.o., the gated single- and multiple-vacation systems as well as the limited-multiple gated vacation system.

Upon returning from a vacation, the single-vacation system does not take a new vacation but waits for the following customer to arrive. Substituting $L = 1$ and $V_1(z) = z$ in our analysis – $V_0(z)$ denotes the probability generating function of the single vacation – easily gives the result for this system. Note that in this case, we consider the idle-period (the system waits for the first arrival batch following a vacation) as a part of the vacation-period.

The system with multiple vacations keeps on taking vacations (which are mutually independent) until there is at least one customer in the system upon returning from a vacation. Putting $L = 0$ ($V_0(z)$ denotes the probability generating function of all vacations) in our analysis, we get the results for the multiple-vacation system. One can show that in the case of multiple vacations, expressions for the moments of queue contents at various epochs and for the moments of the customer delay are independent of K, implying that one does not need numerical determination of K.

In the system with limited-multiple vacations, the server keeps on taking vacations when there are no customers in the system upon returning from a vacation and as long as a maximal number M of vacations is not reached. After this maximum number of vacations, the server waits for the first arrival, similarly as the single vacation system. We get $L = M$, and $V_M(z) = z$ whereas $V_i(z) = V_0(z)$ for $0 \leq i < M$, i.e., $V_0(z)$ is the probability generating function of the first M vacation periods. Clearly, the single-vacation policy corresponds to $M = 1$, whereas the multiple-vacation policy corresponds to $M = \infty$.

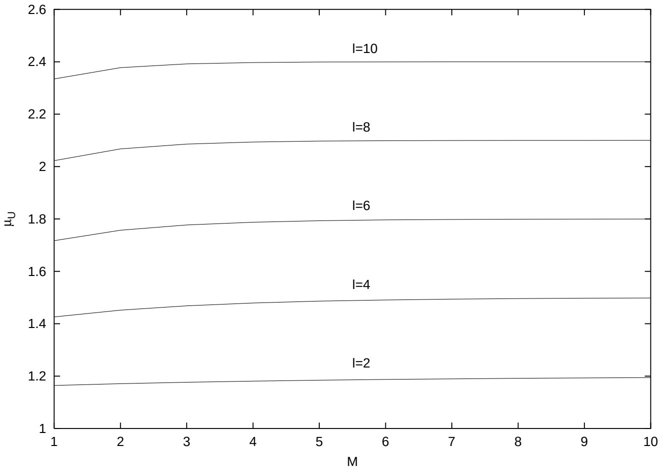

Fig. 1. Mean total system contents μ vs. vacation limit M

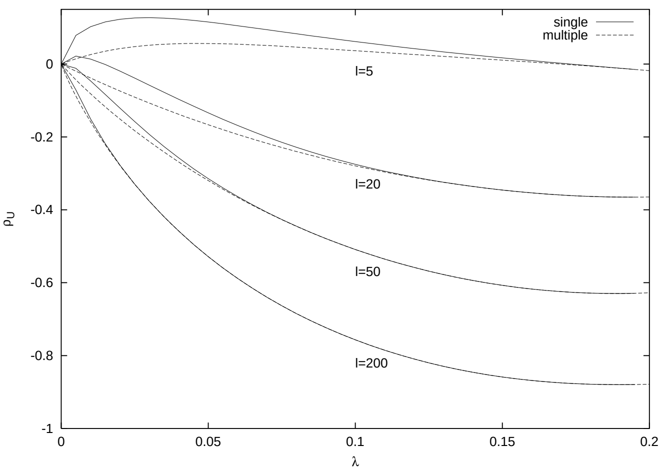

Fig. 2. Correlation between queues vs. arrival rate λ

VI. Numerical example

In some numerical examples, we show gated vacation systems with single, multiple and limited multiple vacations and compare them. We assume that the number of arrivals during the consecutive slots are a series of Poisson-distributed random variable whereas the service times of the consecutive customers are assumed to be a series of geometrically distributed random variables, i.e.,

$$\begin{aligned} A(z) &= e^{\lambda\,(z-1)}, \\ S(z) &= \frac{z}{\theta + (1-\theta)\,z}, \end{aligned} \tag{19}$$

where λ denotes the mean number of arrivals per slot, and where θ denotes the mean service time of a customer. We further assume deterministic vacation lengths of l slots,

$$V_0(z) = z^l, \tag{20}$$

and an upper limit M for the maximal number of vacations in case of the limited vacation policy.

Figure 1 depicts the mean total system contents – the number of customers in both queues – versus the maximal number of vacations M. Clearly, $M = 1$ corresponds to the single vacation system whereas $M = \infty$ corresponds to the multiple-vacation system. For all curves the arrival rate λ equals 0.1 whereas the mean customer service time θ equals 5 slots. The length of the vacation periods l varies from curve to curve as depicted. For all l, mean queue contents quickly converges for increasing M to the multiple-vacation value implying that performance gain by limiting the maximal number of vacations is small.

Figure 2 depicts the correlation between the number of customers in primary and secondary queue versus the arrival rate λ. The mean customer service time θ equals 5 slots whereas the vacation lengths vary for the different curves as depicted. We consider both the multiple-vacation as the single-vacation system. For more heavily loaded systems, correlation is negative, i.e., one can expect small secondary queue sizes if the primary queue is heavily loaded and vice versa. This is expected as for the system under consideration, the secondary queue size increases while the primary queue size decreases and vice versa. For smaller loads, and in particular for shorter vacation lengths, correlation is small and positive as both queues remain empty for longer periods.

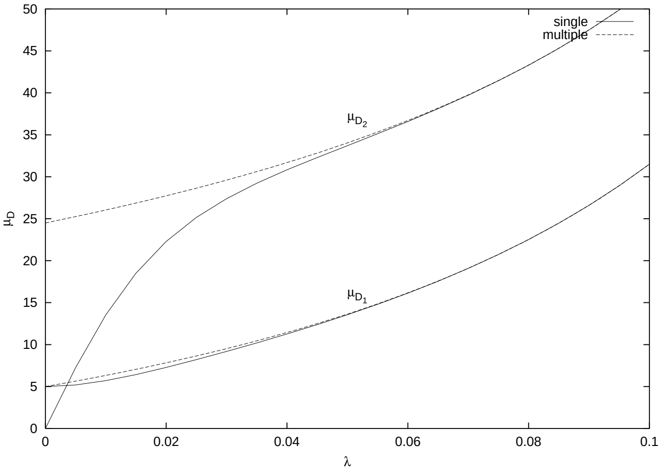

Fig. 3. Mean customer delay vs. arrival rate λ

Figure 3 depicts the mean customer delay in primary (μ_{D_1}) and secondary (μ_{D_2}) queue for the single- and multiple-vacation system. The mean customer service time θ equals 5 slots whereas the vacation length equals 50 slots. Consider in particular the mean secondary queue length for the single-vacation system. Clearly, for low load, the probability that a customer arrives during an idle period (the system waits for the first arrival after a vacation) increases. For such a customer, secondary delay equals 0 (since the gate opens at the end of its arrival slot) which explains the strong decrease of the secondary delay for decreasing λ. For low loads in the multiple-vacation system, the server is most probably on vacation, and therefore mean delay converges to $(l-1)/2$ for $\lambda \to 0$, i.e., the mean waiting time until the end of a vacation. For increasing load, the probability to find the system empty at the end of a vacation decreases, and as a consequence, curves for multiple-vacation and single-vacation systems converge.

VII. Conclusions

We considered the gated vacation system in discrete-time. We analyzed the joint system-contents and joint customer-delay in both queues of the system. The flexibility of the vacation process under consideration allowed to model a.o., the single- and multiple- vacation gated queueing systems as well as the gated limited multiple-vacation system.

REFERENCES

[1] B.T. Doshi. Queueing systems with vacations – a survey. *Queueing Systems*, 1:29 – 66, 1986.

[2] H. Takagi. *Queueing Analysis, A Foundation of Performance Evaluation, Volume 1: Vacation and Priority Systems, Part 1*. Elsevier Science Publishers, 1991.

[3] H. Takagi. A survey of queueing analysis of polling models. In *Proc. Third IFIP International Conference on Data Communication Systems and Their Performance*, Rio de Janeiro, Brazil, 22–25 June 1987.

[4] D. Fiems, B. Steyaert, and H. Bruneel. Discrete-time queues with general service times and general server interruptions. In *Internet Performance and Control of Network Systems, Proc. of SPIE, Vol 4211*, Boston, USA, 6-7 November 2000.

[5] F.A. Van der Duyn Schouten and S.G. Vanneste. Maintenance optimization of a production system with buffer capacity. *European Journal of Operational Research*, 82:232–338, 1995.

[6] D. Towsley and J.K. Wolf. On the statistical analysis of queue lengths and waiting times for statistical multiplexers with ARQ retransmission schemes. *IEEE Transactions on Communications*, COM-27(4):693 –702, April 1979.

[7] S. Sumita. Performance analysis of interprocessor communications in an electronic switching system with distributed control. *Performance Evaluation*, 9:83–91, 1988/89.

[8] H. Bruneel. Performance of discrete-time queuing systems. *Computers and Operations Research*, 20:303 – 320, 1993.

[9] H. Takagi. *Queueing Analysis; Volume 3: Discrete-Time Systems*. Elsevier Science Publishers, Amsterdam, 1993.

[10] D. Fiems and H. Bruneel. A note on the discretization of little's result. *Operations Research Letters*, (accepted for publication 2002).

[11] W. Whitt. A review of $L = \lambda W$ and extensions. *Queueing Systems*, 9:235–268, 1991.