Rejoinder on: Queueing Models for the Analysis of Communication Systems

Herwig Bruneel, Dieter Fiems, Joris Walraevens and Sabine Wittevrongel

Ghent University
Department of Telecommunications and Information Processing (TELIN)
*SMACS Research Group**
Sint-Pietersnieuwstraat 41, B-9000 Gent, Belgium
E-mail: {hb,df,jw,sw}@telin.UGent.be

First of all, we want to express our gratitude to the former Editor-in-Chief of the journal TOP, our fondly-remembered colleague Professor Jesús Artalejo, who invited us about one year ago to write a review paper for TOP on the topic of queueing models for the analysis of communication systems. Also, we want to thank the three discussants for the effort of carefully reading our paper and adding interesting information. We mention in particular the insightful discussion of Onno Boxma on the role of transform methods in queueing analysis, their strengths and weaknesses. With respect to scheduling disciplines, Onno Boxma also adds the topic of dynamic priority scheduling schemes. As far as traffic modeling is concerned, Alexander Dudin not only refers to the well-known D-BMAP model as an alternative description of bursty, correlated traffic streams, but he also points to several related papers on continuous-time models with session-based arrivals. Harry Perros emphasizes the practical applications of discrete-time queueing models in the performance evaluation of various subsystems of modern telecommunication networks. In particular, he discusses the restriction introduced by the assumption in our paper that packets be of fixed length, which makes the results applicable in the context of ATM, in optical burst switching, and in HTTP adaptive video streaming, but not in a general IP network where packets typically are of variable length.

A very striking observation is that all three discussants explicitly comment on the use of *discrete-time* queueing models in much of our work of the last thirty years (including the current paper), as opposed to the much more frequently encountered *continuous-time* models in literature. As these remarks are of a very generic nature, we isolate this topic from the more specific comments in this rejoinder. Therefore, our responses are structured around two main topics: (1) the aspect of modeling and queueing analysis in the discrete time domain and (2) possible extensions of and remarks on our presented analysis of the two-class priority queue with train arrivals.

*SMACS: Stochastic Modeling and Analysis of Communication Systems

1 Discrete-time modeling

Let us start with the commonly accepted remark that, in many cases, discrete-time queueing models arise in a very natural way in the context of modern (and, therefore, digital) communication systems. The reason for this is that digital systems are slotted systems (at least, at the microscopic time scale), where a system clock defines a discrete series of clock instants. The fixed time intervals in between consecutive clock instants are usually called (time) slots and the operation of the system is synchronized with respect to the slot structure of the time axis. In slotted systems, all the relevant quantities are discrete random variables. Not only the number of arrivals in a given time period or the number of packets in a buffer (which would be discrete in continuous-time models as well), but also all other quantities of interest, such as inter-arrival times, service times, unfinished work, busy and idle periods, etc. It is this distinct feature of discrete-time models that makes the use of probability generating functions (pgfs) so beneficial and generic in this context. When analyzing continuous-time models with transform methods, one needs both pgfs (for the discrete random variables) and Laplace-Stieltjes transforms (for the continuous random variables). This circumstance is a definite disadvantage, as we firmly believe that Laplace transforms, which are basically integrals, are much more complicated concepts than pgfs, which are merely (infinite) sums.

We therefore do not completely agree with Harry Perros' remark that "A crucial assumption in the use of generating functions is that time is slotted". This is true as far as the characterization of time periods is concerned, but not for other quantities such as the number in the system, which is a discrete quantity in continuous-time queueing models as well. Pgfs have also been very successfully used, for instance, in the analysis of the continuous-time $M/G/1$-model. It is also not true that discrete-time models can only deal with fixed-length packets (and, therefore, fixed-length service times). It is perfectly possible to consider variable-length service times with some given discrete probability distribution (or pgf). The only restriction within the discrete time setting is that service times must be integer multiples of the slot length, but this is usually not very important. Instead of being a restriction of discrete-time models, we feel, on the contrary, that dealing with constant service times of 1 slot each, is much simpler in a discrete time setting than in a continuous time setting. In continuous time, the simplest distribution to work with is very often the exponential distribution, because of its memoryless nature. The discrete analog of the exponential distribution is the geometric distribution which is also memoryless, but the deterministic distribution with value 1 is just a special case of the geometric distribution, notably with parameter 0, and can therefore be worked with equally easily. This is not the case in continuous time, where deterministic (constant) random variables are in many respects equally hard to deal with as completely arbitrarily distributed random variables. We must admit, however, that deterministic random variables with other values than 1, e.g., fixed service times consisting of multiple slot lengths, are also "difficult" in discrete time.

Along the same lines, we are also not convinced that the remark of Alexander Dudin "It is well known that continuous-time Markov chains are a bit more complicated subject of research than the discrete-time Markov chains due to the necessity of account of time spent by the chain at each state until the next transition" is completely correct. We do agree that in the specific model dealt with in our paper, the description of the session-based arrival process was relatively simple, because actually the special case of train arrivals was

considered, i.e., where an active session generates exactly one packet in each consecutive slot, which means that there are no gaps in between subsequent packet arrivals within a session. But the session lengths (train lengths, message lengths) are also arbitrarily distributed in our model (with pgfs $L_1(z)$ and $L_2(z)$ respectively for class 1 and 2), and therefore in our analysis we must also keep track of the remaining message lengths. If we were to model gaps in between the subsequent packets of a session, our discrete-time analysis would also encounter the same difficulties as in a continuous time setting. We note, however, that on the contrary, there is no real continuous-time equivalent of the discrete-time model where subsequent packets arrive back-to-back during a session.

About the general remark, heard occasionally, that continuous-time models are more difficult to deal with than their discrete-time counterparts, we definitely have very mixed feelings. We believe that much depends on the exact nature of the models at hand. As exemplified already above, most continuous-time models have their discrete-time analogs. For instance, continuous probability distributions for inter-arrival times and service-times can be easily replaced by corresponding discrete distributions. The analysis techniques which are customary in continuous time can usually be applied in a discrete time setting as well. We think, for instance, about balance-equation approaches, embedded Markov chain approaches (e.g., at customer departure times, at customer arrival epochs, etc.), supplementary-variable approaches, matrix-analytic techniques, etc. In all these cases, we believe that the technical difficulties encountered in the solution are basically of the same complexity. However, we have observed time and again that discrete-time models are not always attacked with the same techniques as their continuous-time analogs. Some observations in this respect are the following.

- In a discrete time setting, a very natural sequence of observation epochs for the system (embedding points) is the collection of consecutive slot boundaries. There is no analog for this in continuous time. Consequently, other types of equations arise than the ones usually encountered in continuous time.

- In the continuous-time queueing literature, transform methods are very often applied in two consecutive steps. The first step consists of establishing balance equations for the steady-state probabilities of the Markov chain at hand, a process which may be very error prone as soon as the complexity of the model is not so low. The second step then amounts to transforming the balance equations from the "probability domain" to the "transform domain", by using either Laplace transforms or z-transforms. In discrete time settings, it is more customary to start the analysis with the definition of a number of relevant discrete random variables describing the system state at the embedding points at hand (slot boundaries, service completion times, arrival times, etc.). The next step then consists in the establishment of a limited number of "system equations", very often recursive equations, which relate the system state at the next embedded point to the state at the previous point. In these system equations, no probabilities are involved; only simple relationships between relevant random variables need to be established. Then, instead of translating the system equations from the "time domain" into corresponding (time dependent or balance) equations in the probability domain, one immediately goes from the random variables to their respective pgfs, using the definition of the pgf of a random variable X as the expected value $X(z) \triangleq \mathrm{E}\left[z^X\right]$. Computing expected values is usually much simpler than establishing equations between probabilities.

- Establishing equations in the "time domain" is less intuitive in a continuous time setting than in a discrete time setting. In discrete time, one can easily go from the previous time instant (slot boundary) to the next, because the consecutive slot boundaries are discrete points in time, one time unit apart from each other, i.e., they are countable. In continuous time, on the contrary, there is no such thing as the next time instant, since the set of time instants forms a continuum. This is usually solved by going from a time t to the time $t + \mathrm{d}t$ where $\mathrm{d}t$ is a positive infinitesimal time increment. In some way, one could say that a finite interval in discrete time (the "slot") is replaced by an infinitesimal interval (the increment $\mathrm{d}t$) in continuous time. We are convinced that infinitesimal quantities are not as natural or intuitive to deal with as finite quantities, which points into the direction that continuous-time models are harder to deal with than discrete-time models. On the other hand, in many cases only one system state transition can occur in an infinitesimal interval, while multiple events may take place during a slot. The latter observation may then lead to the conclusion that continuous-time models are easier than their discrete-time analogs.

- In continuous-time models, arrival streams are very often modeled as Poisson processes, which is equivalent to assuming that the inter-arrival times are independent and identically distributed (i.i.d.) with exponential distribution. This type of arrival process has the property that the numbers of arrivals in non-overlapping time intervals are independent random variables with a Poisson distribution. For other inter-arrival time distributions than the exponential, this property no longer holds, and in this respect the Poisson process and the Poisson distribution are unique. On the contrary, in discrete-time models, it is very natural to characterize the arrival process by specifying the numbers of arrivals during consecutive slots (instead of the discrete inter-arrival times). In this setting, it is often assumed that the numbers of arrivals during consecutive slots are i.i.d. with any given probability distribution, i.e., a general distribution is not "harder" to model than a Poisson distribution. This is a definite advantage of the discrete-time modeling paradigm.

Onno Boxma raised the question if it is somehow possible to obtain results for a discrete-time model from results for its continuous-time counterpart, and vice versa. Indeed, this is often possible, although properly identifying the counterpart is not always trivial. Before pointing to some issues with the continuous-time equivalent of the model in [1], we detail how a limiting operation can be performed for obtaining the mean queue content of a continuous-time $M/G/1$ queue from that of the discrete-time $M/G/1$ queue. We start with the continuous-time model and discretize the inter-arrival times and service times with respect to some slot length Δ. To simplify the limiting operation, assume that inter-arrival times are truncated to an integer number of slots, while service times are increased until they reach an integer number of slots. One now easily checks that the discrete-time model obtained is indeed a single-server queue with geometric inter-arrival times and general service times. Moreover, it should be clear that by such a construction, the queue content of the discretized queue is larger than or equal to the content of the continuous-time queue. Assuming a Loynes-type construction, the content of the discretized queue at time 0, denoted as u_0^{Δ}, will be decreasing for decreasing Δ. As the queue content is bounded, this implies that $u_0^{1/N}$ converges to the continuous-time queue content for $N \to \infty$, almost surely. Moreover, by the dominated convergence theorem

the moments of the discretized queue content converge to the moments of the continuous-time queue for $N \to \infty$, assuming that the moments of the discrete-time queue exist for some Δ. Note that if the moments of the continuous-time queue exist, there always exists a sufficiently small Δ such that the discretized queue is stable and the moments of the discretized queue exist as well. Related approaches to obtain results for continuous-time systems from discrete-time results can be found, e.g., in [2–5].

For the queueing model with train arrivals we considered in [1], we are not very hopeful that a similar approach will yield an equivalent continuous-time model. The described limit approach does not succeed by the lack of a continuous-time equivalent of two processes. Foremost, trains produce at least a single packet per slot. The obvious limit would be that trains produce packets at an infinite rate, which does not lead to an interesting model. More interesting limits can be obtained by assuming that trains only produce a packet with some probability. However, such an extension of the discrete-time model voids the discrete-time analysis of [1], as discussed further in the next section of this rejoinder. Secondly, the server processes a single packet per slot, the obvious limit being immediate service of packets. Here, we are more hopeful that the discrete-time analysis can be extended to include geometrically distributed service times, which lead to exponential service times after the limiting operation.

Summarizing, we feel that it is fair to say that continuous-time models and discrete-time models are partly overlapping and partly complementary. We believe that each continuous-time model has its own discrete-time counterpart. We are also convinced that most methodologies from the continuous time domain can be translated into equivalent discrete-time solution techniques, although this may not be the most efficient way to deal with the discrete-time model. It is not true, on the contrary, that each discrete-time model can be easily translated into an equivalent continuous-time model, for the simple reason that the continuous-time model does not dispose of the notion of a "slot". For instance, independent arrivals from slot to slot in discrete time cannot be easily translated into inter-arrival time characterizations in continuous time, unless their distribution is Poisson. If one allows multiple state transitions within infinitesimal intervals, as e.g., in batch-arrival models or batch-service models, then, of course, the distance between both types of models decreases. But still the discrete time setting has the advantage that it can deal in a very natural way with constant service times (or inter-arrival times, for that matter) equal to one time unit.

2 Further extensions of the queueing analysis

In this section, we describe some of the potential extensions to the model and analysis in [1] inquired about by the discussants in their comments about the paper. These extensions can be roughly classified into (i) refinements of the modeling assumptions and (ii) the calculation of additional performance measures.

The most obvious extension is the generalization to more than 2 traffic classes. As Onno Boxma rightfully states, if one is interested in the delay of a particular class (say class M) in an N-class system, the higher-priority classes can be aggregated to one class, as can the lower-priority classes. As class M does not "see" lower-priority classes and as the scheduling within higher priority-classes has no impact on the delay of class M, the analysis of the multi-class system reduces to the analysis of a queueing system with

two priority classes, as in the present. The argument above however yields a model with *heterogeneous* trains (trains with different distributions for their lengths) within the high-priority class, whereas the model in [1] assumes homogeneous trains. Consequently, a direct application of the results of the two-class model is not possible. However, we suspect that the transition from homogeneous to heterogeneous trains (or to more than two priority classes) is not that hard. Without detailing such analysis here, we mention that heterogeneity requires the inclusion of more than two vectors of infinite dimension of the number of messages of which the n-th packet arrives during a random slot in, for instance, expression (10) for P_T in [1]. Note that the inclusion of more than one traffic class in one priority class is an interesting generalization in its own right. One can for example analyze the delay of a given traffic class within a given priority class and study different scenarios for mapping traffic classes to priority classes, extreme scenarios being full-FIFO (all traffic classes in one priority class) and full-priority (a one-to-one map between traffic classes and priority classes). We plan on studying generalizations along these lines in the near future.

Further extensions to the model include generally distributed service times (see also the discussion in the first section) and multiple servers. However, we expect the latter to be a hard task, as multi-server models with priority scheduling are complicated even for independent arrivals (see e.g., the analyses in [6] and [7] for single-slot service times and geometric service times, respectively). An interesting recent generalization of session-based arrival models is a model with possible gaps in between the subsequent packets of a session, i.e., where not necessarily a packet arrives in each slot the session is active. This can be seen as the discrete-time analog of the continuous-time models Alexander Dudin discussed in his comments. For such a model with gaps, the analysis becomes much harder than for the train arrival model we considered in our paper [1]. For instance, in [8] a FIFO queue with gaps in the arriving trains is approximately analyzed through the use of Taylor series approximations. Extension of this approach to priority queueing models would be an interesting and challenging research topic.

Finally, we comment on obtaining some additional performance measures that were *not* discussed (or not fully detailed) in the original analysis. In [1], we calculated the pgf $D_2(z)$ of the delay of class-2 packets and, from this, we found formulas for the mean delay $\mathrm{E}[d_2]$ as well as for asymptotics of the probability $\Pr[d_2 = m]$ that the class-2 delay equals m, for $m \to \infty$. First, we would like to reassure Harry Perros that percentiles of the delay can easily be obtained from the latter results. In particular, we recall that we encountered three types of asymptotics, summarized as

$$\Pr[d_2 = m] \sim c m^{\alpha} z_D^{-m}, \tag{1}$$

with c a constant depending on the system parameters, z_D the dominant singularity of $D_2(z)$ and $\alpha = 0$, $-3/2$ or $-1/2$. For large m, an increase in m has a negligible impact on the factor m^{α} in comparison with the impact of m on the factor z_D^{-m}. Therefore, we have

$$\begin{aligned} \Pr[d_2 > m] &\sim c m^{\alpha} \sum_{\ell=m+1}^{\infty} z_D^{-\ell}, \\ &= c m^{\alpha} \frac{z_D^{-m}}{z_D - 1}, \end{aligned} \tag{2}$$

for $m \to \infty$. From (2), one can then calculate the required percentiles. An alternative way to obtain (2) consists of calculating the z-transform of $\Pr[d_2 > m]$ and relating it to $D_2(z)$, as follows:

$$\sum_{m=0}^{\infty} \Pr[d_2 > m] z^m = \frac{D_2(z) - 1}{z - 1}$$

and then performing a singularity analysis on the latter formula; this yields

$$\Pr[d_2 > m] \sim \frac{\Pr[d_2 = m]}{z_D - 1}.$$

The final expression (26) of $D_2(z)$ in [1] contains the pgf $V(z)$ of the lengths of sub-busy periods, which is only implicitly defined. We showed that this is not a problem for neither the calculation of the moments of the delay nor for the calculation of the asymptotics. The former follows from the fact that $V(z)$ is a probability generating function and therefore takes the value 1 for $z = 1$, which in turn permits to calculate all derivatives of $V(z)$ in $z = 1$. With respect to the asymptotics, a numerical procedure to find the branchpoint z_V of $V(z)$ and the value $V(z_V)$ was explained in [1]. However, if one would want to numerically invert the probability generating function $D_2(z)$, by means of an Inverse Fast Fourier Transform, for instance, $V(z)$ needs to be calculated for a number of complex arguments z. As Onno Boxma rightfully notes, $V(z)$ bears a close resemblance to the functional equation for a busy period. In fact, for the model discussed in [1], the pgf of the length of a busy period of class 1 is given by

$$\frac{A_1(V(z)) - A_1(0)}{1 - A_1(0)},$$

which shows the connection with $V(z)$. This type of implicitly defined generating function can be determined iteratively for a desired argument z, namely by repeatedly calculating $V_i(z) = zA_1(V_{i-1}(z))$. It is shown in [9], by means of probabilistic arguments, that this procedure converges to the correct value, if the function is the transform of a (possibly defective) random variable, which is the case here.

For the calculation of $D_2(z)$, we also need to first analyze the joint pgf $P_T(\mathbf{x_1}, \mathbf{x_2}, z)$ of the numbers of messages of both classes of which the n-th ($n = 1, 2, \ldots$) packet arrives during a random slot and the *total* number of packets in the system at the beginning of the next slot. This joint pgf is actually independent of the scheduling discipline at hand, as long as it is work-conserving. Important to notice here is that we only need to know the *total* system content and not how the packets in the system are divided over both classes, which simplifies matters considerably. Indeed, calculation of the joint pgf of the numbers of packets of both types in the system by a similar approach turns out to be considerably more difficult. Nevertheless, analysis of the numbers of packets (or even messages) of the individual classes could be interesting as well, and remains an open problem for now. For the same reason, the calculation of higher moments of the number of packets of class 2 in the system (and also of the class-2 delay) in the more general model of zero-regenerative arrivals (see Section 6 of [1]) is still an open problem at the moment. For the calculation of the *mean* number of class-2 packets, we used the fact that the mean number of class-2 packets equals the difference between the mean total number of packets and the mean number of class-1 packets. This kind of reasoning obviously does

not hold for higher moments and it is not yet clear whether calculation of higher-order moments will require assumptions on the arrival process beyond those stated in [1].

Finally, Harry Perros inquired about the departure process from a priority queue. We did investigate this for independent arrivals, see our paper [10]. In this paper, the departure process is described by a three-state semi-Markov chain, where the states represent the situations of no departure, a departure of class 1 and a departure of class 2 in a slot. Note that the model of [10] assumes independence between consecutive sojourn times in the states, which is a simplification. In fact, this model leads to trains of outgoing packets, whereby class-1 and class-2 packets are transmitted in the same train, contrasting the assumptions we made in [1]. Nevertheless, we believe that the analyses of [1] and [10] can be combined and then allow one to analyze the performance of a network of queues by splitting up the network into single queues whereby the departure process of one queue is used as (part of) the input process of the next queue.

Acknowledgment. This research has been funded by the Interuniversity Attraction Poles Programme initiated by the Belgian Science Policy Office.

References

[1] H. Bruneel, D. Fiems, J. Walraevens and S. Wittevrongel, Queueing models for the analysis of communication systems, *TOP*, this issue.

[2] J.A.C. Resing, G. Hooghiemstra and M.S. Keane, The M/G/1 processor sharing queue as the almost sure limit of feedback queues, *Journal of Applied Probability*, vol. 27 (1990), pp. 913-918.

[3] J.R. Artalejo, I. Atencia and P. Moreno, A discrete-time $\text{Geo}^{[X]}/\text{G}/1$ retrial queue with control of admission, *Applied Mathematical Modelling*, vol. 29 (2005), pp. 1100-1120.

[4] J. Walraevens, J.S.H. van Leeuwaarden, O.J. Boxma, Power series approximations for two-class generalized processor sharing systems, *Queueing Systems*, vol. 66 (2010), pp. 107-130.

[5] A. Pacheco, S.K. Samanta and M.L. Chaudhry, A short note on the GI/Geo/1 queueing system, *Statistics & Probability Letters*, vol. 82 (2012), pp. 268-273.

[6] K. Laevens and H. Bruneel, Discrete-time multiserver queues with priorities, *Performance Evaluation*, vol. 33 (1998), pp. 249-275.

[7] P. Gao, S. Wittevrongel and H. Bruneel, Analysis of buffer behavior for a discrete-time multiserver preemptive priority queue with geometric service times, *Book of Abstracts of the Twelfth INFORMS Applied Probability Society Conference* (Beijing, June 2004), pp. 38-39.

[8] K. De Turck, D. Fiems, S. Wittevrongel and H. Bruneel, A Taylor series expansions approach to queues with train arrivals, *Proc. of VALUETOOLS 2011* (Cachan, May 2011).

[9] J. Abate and W. Whitt, Solving probability transform functional equations for numerical inversion, *Operations Research Lettters*, vol. 12 (1992), pp. 275-281.

[10] J. Walraevens, D. Fiems, S. Wittevrongel and H. Bruneel, Calculation of output characteristics of a priority queue through a busy period analysis, *European Journal of Operational Research*, vol. 198 (2009), pp. 891-898.