SERIE RESEARCH MEMORANDA

A NON-EXPONENTIAL QUEUEING SYSTEM WITH BATCH SERVICING

Nico M. van Dijk
Eric Smeitink

Researchmemorandum 1988-13

April '88

VRIJE UNIVERSITEIT
Faculteit der Economische Wetenschappen en Econometrie
A M S T E R D A M

A NON-EXPONENTIAL QUEUEING SYSTEM WITH BATCH SERVICING

Nico M. van Dijk and Eric Smeitink
Free University, Amsterdam,
The Netherlands

Abstract. A finite source queueing system is studied in which jobs generated by a source arrive independently but are served in a batch manner. The servicing includes source interdependencies. The input and service distributions are allowed to be generally distributed. A non-standard product form expression is obtained for the steady state joint queue length distribution and shown to be insensitive (i.e. to depend on only mean input and service times). The result is of both practical and theoretical interest as an extension of more standard batch service systems.

Keywords. Batch servicing * Product Form * Insensitivity * Source balance.

1. Introduction

Queueing models have been extensively applied to telecommunication analysis and computer performance evaluation over the last decades while presently they also enjoy an increasing popularity in flexible manufacturing and stochastic Petri nets.

Generally, the assumption is made that jobs can leave a service station one at a time. However, in various present-day applications it appears more realistic that jobs depart in batches. For instance, in voice-data communication along digitized channels a number of time slots (to be seen as servers) are released at the same time. In parallel programming various program modules may have to be run simultaneously. In flexible manufacturing, parts are often worked upon (e.g. coated, heated, polished) or transported (e.g. by automated guided vehicles) grouped at pallets. In Petri nets, finally, the firing of a transition often requires various tokens to be released (completed) at the same time.

The literature on systems with synchronous servicing seems to be restricted to systems in which jobs to be served simultaneously all arrive at the same time. A distinction is then to be made between systems in which jointly used servers are released independently (independent case) or at the same time (concurrent case). For the independent case some analytical results are available (cf. Green [5], [6], Seila [12]), but generally a numerical or approximative approach seems to be needed (cf. Federgruen and Green [3], Fletcher et.al. [4]). For the concurrent case the celebrated product form has been established under the assumptions of exponential inputs and lost arrivals upon blocking (cf. Arthurs and Kaufman [1], Kaufman [8], Schwartz and Kraimeche [11], Whitt [13]).

The system under study in this paper can be regarded as just the opposite of the independent service case in that jobs to be served simultaneously all depart at the same time but arrive one after the other. This leads to an essentially different complication as jobs have to wait for other jobs to arrive before service can be started.

In our analysis we allow the input to be of a state dependent multi-source type while both the interarrival and service times are generally distributed. Moreover, a state dependent service allocation is included so as to model source interdependencies.

This system cannot be analyzed as a more standard system with simultaneous servicing as the jobs arrive one after the other, while it cannot be seen as a more standard state dependent restricted finite source system as the presence of jobs not in service and the size of batch departures influence both the arrival and source interdependent service rates.

A product form expression will be derived. This expression is insensitive to the distributional forms of the input and service distributions (i.e. it depends on only their means). At first glance this product form may seem standard, but it is not, as the marginal terms per source have no geometric or birth-death type form (see remark 3.1). Moreover, insensitivity is generally known to fail when waiting is involved. Here, however this is not the case. Finally, it is to be emphasized that no conditions are imposed upon the source interdependent service rates. The technique of the proof is of interest in itself as it requires a more general notion of partial balance than is standardly known to be responsible for insensitive product form expressions. (see remark 3.3).

The result can also be regarded as a first step towards queueing networks with batch departures. Such networks are currently of increasing interest for applications such as voice-data transmission analysis, packet switching and flexible manufacturing. Most notably also, the feature of batch departures seems of interest for the recently developing area of stochastic Petri nets (cf. Molloy [9]), as the firing of a transition is sometimes initiated by simultaneously released tokens. The results of this paper may motivate further ivestigation in these directions.

2. Model

Consider a system with M sources that generate jobs to be served by a multiple processor in the following manner. A source remains generating new jobs as long as service upon its jobs has not started. Upon a new job generation by a source a batch service upon all its jobs can be initiated as one of the following happens:

(i) the job is rejected and lost after which the source starts a new job generation.

(ii) the job is accepted but no service upon the jobs from this source is started.

(iii) the job is accepted and initiates a service upon all jobs from that source simultaneously. That is they all begin and end service concurrently. During this service no other jobs are generated by the source. The job generation by this source is restarted upon completion of this service.

More precisely, when a newly arriving job from source i raises the number of source i jobs requiring service to n_i, a batch service upon these jobs is started with probability

$$b_i(n_i) \qquad (i=1,\ldots,M) \tag{2.1}$$

Let $[\bar{n},\bar{s}] = ((n_1,s_1),\ldots,(n_M,s_M))$ denote that n_i jobs from source i are currently waiting for service when $s_i=1$ while in service when $s_i=2$, where we always assume $s_i=1$ for $n_i=0$, $i=1,\ldots,M$. Then the n_i jobs from a source i with $s_i=2$ are served at a state dependent positive service rate:

$$\phi_i(n_i|[\bar{n},\bar{s}]) \qquad (s_i=2\ ;i=1,\ldots M). \tag{2.2}$$

The required service amount for a batch of n_i jobs from source i is random and may depend on the batch size as given by the distribution function $S_i^{n_i}(.)$ with mean $\tau_i(n_i)$.

Conversely, also the job generation times of a source are allowed to

have a general random distribution depending upon its number of jobs waiting for service. With n_i the number of jobs already generated by source i and waiting for service, the distribution function for a next generation by source i is given by $G_i^{n_i}(.)$ and assumed to have a mean $\gamma_i(n_i)$ for all $n_i \leq B_i$ while for $n_i > B_i$ no new jobs are generated, where B_i is some given constant (possibly infinite).

Without loss of generality we assume that the distributions $G_i^{n_i}(.)$ and $S_i^{n_i}(.)$ are continuously differentiable with densities $g_i^{n_i}(.)$ and $\sigma_i^{n_i}(.)$ repectively. Also, in order to guarantee that generated jobs are ever served we impose the natural conditions:

$$\Sigma_{k=1}^{B_i} b_i(k)[1-b_i(1)]\ldots[1-b_i(k-1)] = 1, \text{ and}$$

$$b_i(B_i) = 1, \quad i=1,\ldots,M. \tag{2.3}$$

Remarks.

2.1. The assumption that a source stops generating jobs while its jobs are being served may for instance reflect that some device such as a carrier or transporter is needed for either generating or servicing jobs. In stochastic Petri nets it naturally arises as a source directly empties itself by firing a batch of tokens.

2.2. As a particular example of probabilities $b_i(n)$ initiating a batch service of size n, we may have

$$b_i(n) = \begin{cases} 0 & n<B_i \\ 1 & n=B_i \end{cases}$$

which corresponds to batch services of a fixed size B_i for jobs from source i. This example is typically involved in manufacturing applications.

2.3. The state dependent service rates (2.2) can be used to model for instance a single-server or multi-server discipline for the jobs of a given source. This however can also be achieved by appropriate number dependent service distributions S_i^n. More importantly, however, they may function as delay or acceleration factors upon particular sources to reflect source interdependencies such as an excess of a common threshold, a joint processor-sharing server, or a source preference. For example, with $n_2[\bar{s}]$ the number of sources with jobs in service, we may have

$$\phi_i(n_i|[\bar{n},\bar{s}]) = \begin{cases} 1 & \text{if } n_2[\bar{s}] < L \\ 2 & \text{if } n_2[\bar{s}] \geq L \end{cases} \tag{2.5}$$

to reflect that the servicing speed of a single server for each source is doubled if more than L sources are to be served at the same time. Or we can use,

$$\phi_i(n_i|[\bar{n},\bar{s}]) = n_i/[\Sigma_{j:s_j=2} n_j] \tag{2.6}$$

to represent that service is provided by a single processor-sharing server. Or we can model 2 sources with an "almost" service priority for source 2 by choosing an arbitrarily small $\varepsilon>0$ and letting

$$\phi_1(n_1|[\bar{n},\bar{s}]) = \begin{cases} 1 & , s_1=2 , s_2=1 \\ \varepsilon & , s_1=2 , s_2=2 \end{cases} \tag{2.7}$$

$$\phi_2(n_2|[\bar{n},\bar{s}]) = \begin{cases} 1 & , s_1=1 , s_2=2 \\ 1-\varepsilon & , s_1=2 , s_2=2 \end{cases}$$

2.4. By allowing the generation and service time distributions $G_i^{n_i}$ and $S_i^{n_i}$ respectively to depend on the number of current jobs n_i, we can model various input and service disciplines such as a Poissonian (with constant rate) or finite source input (with decreasing rate proportional to $[M_i-n_i]$ for some given M_i) and a single-server (with unit capacity) or multi-server discipline (with service capacity proportional to n_i). The details are left to the reader.

3. Product form results.

Let the sources be numbered 1,...,M and denote by

$$[\bar{n},\bar{s},\bar{r}] = \{(n_i,s_i,r_i)\ ;\ i=1,\ldots,M\}$$

the state in which n_i jobs from source i are waiting for service when $s_i=1$ while in service when $s_i=2$, and in which r_i is the corresponding residual time up to the next job generation (when $s_i=1$) or service completion (when $s_i=2$) respectively, where we assume that $s_i=1$ for $n_i=0$. For a vector $\bar{t} = (t_1,\ldots,t_M)$ and with $t_i=n$ for some given i, let $\bar{t} - (n)_i + (m)_i$ denote the same vector with $t_i=n$ replaced by $t_i=m$. (i.e. the i-th component is changed from n into m). Also, abbreviate $\bar{t} - (t_i)_i + (t_i-1)_i$ by $\bar{t}-e_i$.

Without loss of generality assume that the corresponding Markov process has a unique stationary probability density function $\pi(.)$ which is continuously differentiable in all its residual lifetime components r_i, i=1,...,M. The following key-result is then obtained.

Theorem 3.1 With c a normalizing constant, and for all states $[\bar{n},\bar{s},\bar{r}]$ with $n_i \le B_i$ (i=1,...,M), the equilibrium density function is given by

$$\pi([\bar{n},\bar{s},\bar{r}]) = c \ \times$$

$$\left\{ \prod_{\{i:s_i=1\}} [1-G_i^{n_i}(r_i)] \prod_{k=1}^{n_i} [1-b_i(k)] \right\}$$

$$\left\{ \left(\prod_{\{i:s_i=2\}} [1-S_i^{n_i}(r_i)] \right) [\varphi_i(n_i \mid [\bar{n},\bar{s}])]^{-1} \ b_i(n_i) \prod_{k=1}^{n_i-1} [1-b_i(k)] \right\} \quad (3.1)$$

Proof Due to the Markovian structure it suffices to verify the global balance or stationary forward Kolmogorov equations. Recalling that $\pi(.)$ is assumed to be continuously differentiable in its residual life time components r_i and writing 0^+ to indicate a right hand limit in 0, the global balance equations are given by

$$\Sigma_{\{i|s_i=1,\ n_i>0\}} \left[\frac{\partial}{\partial r_i}\,\pi([\bar{n},\bar{s},\bar{r}]) + \right.$$

$$\left. \pi([\bar{n}-e_i,\bar{s},\bar{r}-(r_i)_i+(0^+)_i])\ [1-b_i(n_i)]\ g_i^{n_i}(r_i)\right] +$$

$$\Sigma_{\{i|s_i=2\}}\left[\frac{\partial}{\partial r_i}\,\pi([\bar{n},\bar{s},\bar{r}])\ \phi_i(n_i|[\bar{n},\bar{s}]) + \right.$$

$$\left. \pi([\bar{n}-e_i,\bar{s}-(2)_i+(1)_i,\bar{r}-(r_i)_i+(0^+)_i])\ b_i(n_i)\ \sigma_i^{n_i}(r_i)\right]+$$

$$\Sigma_{\{i|n_i=0\}}\left[\frac{\partial}{\partial r_i}\,\pi([\bar{n},\bar{s},\bar{r}]) + \right.$$

$$\Sigma_{n=1}^{B_i}\pi([\bar{n}-(0)_i+(n)_i,\bar{s}-(1)_i+(2)_i,\bar{r}-(r_i)_i+(0^+)_i])$$

$$\left.\phi_i(n|[\bar{n}-(0)_i+(n)_i,\bar{s}-(1)_i+(2)_i])g_i^0(r_i)\right] = 0$$

(3.2)

Herein the first sum reflects the out and inrate due to sources with one or more jobs waiting for service, the second sum corresponds to the out and inrate due to sources with their jobs in service while the last sum is concerned with the out and inrate due to sources currently without jobs. The natural assumption is made that this equation has a unique probability density solution. We thus need to show that it holds with (3.1) substituted. From (3.1) and recalling that $g_i^{n_i}(.)$ and $\sigma_i^{n_i}(.)$ are the density functions of $G_i^{n_i}(.)$ and $S_i^{n_i}(.)$ and noting that $1-G_i^{n_i}(0^+) = 1-S_i^{n_i}(0^+) = 1$, we obtain

$$\frac{\partial}{\partial r_i}\,\pi([\bar{n},\bar{s},\bar{r}]) = -g_i^{n_i}(r_i)\ [1-b_i(n_i)]$$
$$\pi([\bar{n}-e_i,\bar{s},\bar{r}-(r_i)_i+(0^+)_i])\ ,\ (i:s_i=1,\ n_i>0) \qquad (3.3)$$

$$\frac{\partial}{\partial r_i}\,\pi([\bar{n},\bar{s},\bar{r}]) = -\sigma_i^{n_i}(r_i)\ b_i(n_i)\ \phi_i(n_i|[\bar{n},\bar{s}])^{-1}$$
$$\pi([\bar{n}-e_i,\bar{s}-(2)_i+(1)_i,\bar{r}-(r_i)_i+(0^+)_i])\ ,\ (i:s_i=2) \qquad (3.4)$$

$$\pi([\bar{n}\text{-}(0)+(n)_i, \bar{s}\text{-}(1)_i+(2)_i, \bar{r}\text{-}(r_i)_i+(0^+)_i) =$$

$$\frac{\partial}{\partial r_i} \pi([\bar{n},\bar{s},\bar{r}])\ b_i(n) \left\{ \prod_{k=1}^{n-1} [1-b_i(k)] \right\}$$

$$\{\emptyset_i(n|[\bar{n}\text{-}(0)_i+(n)_i, \bar{s}\text{-}(1)_i+(2)_i])\ g_i^0(r_i)\}^{-1}, \quad (n_i=0) \tag{3.5}$$

By substitution of (3.3) and (3.4) we immediately observe that for each source i separately the term within [...] of the first and second sum in (3.2) is equal to 0. By substituting (3.5) in the third sum of (3.2) and recalling the boundary condition (2.3), equality to 0 is also concluded for the term within [...] for each source with $n_i=0$ separately. The proof is hereby completed

Let $\pi([\bar{n},\bar{s}])$ be the stationary probability of a state $[\bar{n},\bar{s}] = \{(n_i,s_i), i=1,\ldots,M\}$ denoting that source i is in status s_i with n_i jobs. Then the following corollary is an immediate consequence of (3.1) by integrating over values r_i and noting that

$$\int_0^\infty [1-G_i^{n_i}(r)]\ dr = \gamma_i(n_i)$$

$$\int_0^\infty [1-S_i^{n_i}(r)]\ dr = \tau_i(n_i) \quad . \tag{3.6}$$

It proves a product form expression that depends upon the state dependent input and service distributions only through their means. This is generally referred to as insensitivity.

Corollary 3.2 With c a normalizing constant and for all $[\bar{n},\bar{s}]$ with $n_i \le B_i$ $(i=1,\ldots,M)$, we have

$$\pi([\bar{n},\bar{s}]) = c \left\{ \prod_{\{i:s_i=1\}} \gamma_i(n_i) \prod_{k=1}^{n_i} [1-b_i(k)] \right\}$$

$$\left\{ \prod_{\{i:s_i=2\}} \tau_i(n_i)[\emptyset_i(n_i|[\bar{n},\bar{s}])]^{-1} b_i(n_i) \prod_{k=1}^{n_i-1} [1-b_i(k)] \right\} \tag{3.7}$$

Remark 3.3. Note that (3.2) is actually verified by showing that the out and inrates are balanced per source. Insensitivity results are well-known to be related to notions of balance per individual component or job. (cf. Barbour [2], Schassberger [10], Hordijk en Van Dijk [7]). For the model of this paper however the job-local-balance notion is easily shown to fail. This notion of "source balance" therefore is of interest in itself.

Remark 3.4. The assumption of continuously differentiable interarrival and service distributions is made for presentational convenience. It excludes for instance deterministic times. However, by standard though complicated weak convergence limiting approaches (cf. [2]) the expressions (3.1) and (3.7) can be extended to generally distributed interarrival and service times.

References

[1] Arthurs, E. and Kaufman, J.S., "Sizing a message store subject to blocking criteria", in: M. Arato, A. Butrimenko and E. Gelenbe (Eds.), Performance of Computer Systems, *North-Holland*, Amsterdam, 547-564 (1979).

[2] Barbour, A., "Networks of queues and the method of stages", *Adv. Appl. Probl.* 8, 584-591 (1976).

[3] Federgruen, A. and Green L., "An M|G|c queue in which the number of servers required is random", Columbia Bus. Sch. Res. Paper 504A., Colombia University (1983).

[4] Fletcher, G.Y., Perros, H.G. and Stewart, W.J., "A queueing system where customers require a random number of servers simultaneously", *EJOR* 23, 331-342 (1986).

[5] Green, L., "A queueing system in which customers require a random number of servers", *Operations Res.* 28, 1335-1346 (1980).

[6] Green, L., "Comparing operating characteristics of queues in which customers require a random number of servers", *Management Sci.* 27, 65-74 (1981).

[7] Hordijk, A. and Van Dijk, N.M., "Adjoint process, job-local-balance and insensitivity of stochastic networks", *Bull. 44-th Session Int. Inst.*, Vol 50, 776-788 (1983).

[8] Kaufman, J., "Blocking in a shared resource environment", *IEEE Transactions on Comm.* 29, 1474-1481 (1981).

[9] Molloy, M.R., "discrete time Stochastic Petri Nets", *IEEE Transactions on Software Engineering*, No. 4, 417-423 (1985).

[10] Schassberger, R., "The insensitivity of stationary probabilities in networks of queues", *Adv. Appl. Prob.* 10, 906-912 (1978).

[11] Schwartz, M. and Kraimeche, B., "An analytic control model for an integrated mode", *Infocom* '83, San Diego (1983).

[12] Seila, A.F., "On waiting times for a queue in which customers require simultaneous service from a random number of servers", *Operations Res.* 32, 1181-1184 (1984).

[13] Whitt, W., "Blocking when service is required from several facilities simultaneously", *AT&T Technical Journal*, Vol. 64, No. 8, 1807-1856 (1985).