Response time distributions via reversed processes

Peter G. Harrison[*] Maria G. Vigliotti[*]

Abstract

Response time calculations in stochastic networks – e.g. queueing networks – are usually developed in terms of sample path analyses beginning in an equilibrium state. We consider the joint probability distribution of the sojourn times of a tagged task at each node in a network and observe that this is the same in both the forward and reversed processes. Therefore if the reversed process is known, each node-sojourn time can be taken from either process. In particular, the reversed process can be used for the first node in a path and the forward process for the other nodes in a recursive analysis. This approach derives, quickly and systematically, existing results for response time probability densities in tandem, open and closed tree-like, and overtake-free Markovian networks of queues. We also show how to apply the method in stochastic networks that are more general than queueing systems. An example is constructed to illustrate this, which has separable equilibrium state probabilities, a new product-form result in its own right.

1 Introduction

Response times, or sojourn times, are an important quality of service (QoS) metric in many operational systems such as computer networks, logistical systems and emergency services. For example, ambulances in the United Kingdom are under contract to arrive at the scene of a life-threatening emergency within 8 minutes at least 75% of the time. For on-line transaction processing (OLTP) and other real-time systems, quantiles are often specified in Service Level Agreement contracts and industry benchmarks such as TPC-C, which specifies the 90th percentile of response time [12].

The response time of a particular, 'tagged' task along a path in a network of nodes of some kind may be defined as the sum of the sojourn times of the task (i.e. its delays) at those nodes that constitute the path. More generally, the response time probability distribution follows directly from the joint probability distribution of the node-sojourn times. For a path comprising the sequence of nodes $(1, 2, \ldots, m)$, let the response time $R = T_1 + T_2 + \ldots + T_m$, where T_i is the sojourn time at node $i, (1 \leq i \leq m)$, with probability distribution function $T_i(t)$. Then the joint sojourn time distribution is $J(t_1, \ldots, t_m) = \mathbb{P}(T_1 \leq t_1, \ldots, T_m \leq t_m)$ and, denoting Laplace-Stieltjes transforms (LSTs) by asterisks,

[*] Department of Computing, Imperial College London, {pgh,mgv98}@doc.ic.ac.uk

the m-dimensional LST of the joint sojourn time distribution is

$$J^*(\theta_1,\ldots,\theta_m) = \int_0^\infty \ldots \int_0^\infty e^{-(\theta_1 t_1+\ldots+\theta_m t_m)} \mathbf{d}J(t_1,\ldots,t_m)$$

The response time distribution then has LST $R^*(\theta) = J^*(\theta,\ldots,\theta)$. When the sojourn times T_i are independent, this simplifies to $R^*(\theta) = \Pi_{i=1}^m T_i^*(\theta)$.

If the sojourn time at each node i depends solely on the state, N_i say, existing at the node *immediately prior to the arrival of the tagged task*, the conditional joint sojourn time LST is $J^*(\theta_1,\ldots,\theta_m \mid \mathbf{n}) = \Pi_{i=1}^m T_i^*(\theta_i \mid n_i)$ where $T_i^*(\theta_i \mid n_i) = \int_0^\infty e^{-\theta_i t}\mathbf{d}\mathbb{P}(T_i \leq t \mid N_i = n_i)$[1]. In such networks, response time distributions can be computed iteratively through their LSTs using the result that:

$$J^*(\theta_1,\ldots,\theta_m \mid \mathbf{l}) = \Pi_{i=1}^m T_i^*(\theta_i \mid n_i)\mathbb{P}(\mathbf{N} = \mathbf{n} \mid \mathbf{L}(0) = \mathbf{l})$$

where bold type indicates vectors and the random variable $L_i(t)$ is the state of node i at time t, so that the initial state is $\mathbf{L}(0)$ and $N_i = L_i(T_i^-)$ when the tagged task arrives at node i at time T_i. Of course, if the N_i are independent for $i = 1,\ldots,m$, this reduces to the above result that $J^*(\theta_1,\ldots,\theta_m) = \Pi_{i=1}^m T_i^*(\theta_i)$.

In queueing networks it is often the case that the node sojourn times depend only on the queue length at the arrival instant, for example in the overtake-free networks of [11], but the computation of the transient probabilities $\mathbb{P}(\mathbf{N} = \mathbf{n} \mid \mathbf{L}(0) = \mathbf{l})$ is problematic; see [5] for example. If these probabilities can be found (or avoided), the method applies in both open and closed networks; see the above citations, for example.

We apply a completely different approach to the computation of the LSTs of response times in Markovian networks at equilibrium, via joint sojourn time distributions and using reversed processes. The idea is based on the observation that sojourn times are the same whether one considers the forward process or its reversed process. When sojourn times depend only on the state existing at a node at the arrival instant and the reversed process is separable, i.e. a pairwise synchronising network of m reversed nodes, we can use the forward sojourn time at the nodes $2,\ldots,m$ in the 'tail' of a path and the reversed sojourn time at the first node 1, the 'head' of the path; a recursive analysis allows us to consider only the case $m = 2$, the tail-nodes $2,\ldots,m$ constituting a single aggregate 'super-node' in the recursion.

In the next section, we define our method and apply it to a range of queueing networks, providing greatly simplified derivations that hold the potential of automation through the reversed compound agent theorem (RCAT). Response times in simple two-node G-networks, which are actually non-queueing networks with very different response time characteristics [8], follow immediately, and an alternative methodology is revealed in more complex cases. Generalisations are considered in section 5, focusing on a tandem pair of first-come-first-served (FCFS) queues with Erlangian service times. Note that such networks do not even possess a product-form solution for their equilibrium queue lengths and so we first construct a variant that does. The paper concludes in section 7, where future potential of the method is evaluated.

[1] For example, when $m = 2$, $J^*(\theta_1,\theta_2 \mid \mathbf{N} = \mathbf{n}) = \mathbb{E}\,[\,\mathbb{E}\,[e^{-(\theta_1 T_1+\theta_2 T_2)} \mid T_1, \mathbf{N} = \mathbf{n}] \mid \mathbf{N} = \mathbf{n}] = \mathbb{E}\,[e^{-\theta_1 T_1}\,\mathbb{E}\,[e^{-\theta_2 T_2} \mid T_1, \mathbf{N} = \mathbf{n}] \mid \mathbf{N} = \mathbf{n}] = \mathbb{E}\,[e^{-\theta_1 T_1}\,\mathbb{E}\,[e^{-\theta_2 T_2} \mid N_2 = n_2] \mid N_1 = n_1]$.

2 Node-sojourn times and reversed processes

First, let us consider the sojourn times spent by a task in a pair of nodes that are connected in the sense that the task first sojourns in node 1, for time T_1, after which it proceeds to node 2 and sojourns there, for time T_2, before departing from the system. We define the *middle state* $\mathbf{s}_0$ of the network to be that which excludes the tagged task itself at the instant when it passes from node 1 to node 2. The first component of the middle state is therefore the queue length at node 1 existing just after the instant of departure there, and the second component is the queue length existing just before the arrival instant at node 2. In many cases, e.g. a pair of tandem queues, the state $\mathbf{s}$ is a pair, $\mathbf{s} = (s_1, s_2)$, where s_i describes the state of node i only, $i = 1, 2$. We call such a state *separable*.

The sojourn time at node 1, T_1 say, can be calculated as the first passage time from the *initial state*, existing at the task's arrival instant, to exit from the state in which the task departs node 1. In general, this can involve arbitrary transitions in the whole system, i.e. be influenced by the evolution of node 2 as well as node 1. However, often, T_1 is determined solely by the initial state and the evolution of node 1, as in the case of constant rate queues, for example. In this case, the conventional approach to sojourn time analysis is to consider the state of the system at the instant of the task's departure from node 1 and use this as the initial state for the sojourn at node 2; this may also (or may not, of course) then depend solely on the evolution of node 2.

The properties we need to use this technique are therefore:

- The state of the system is separable, i.e. $\mathbf{s} = (s_1, s_2)$, where s_i describes the state of node i only, $i = 1, 2$;

- The sojourn time of the tagged task at each node depends *solely* on the node's state at its arrival instant − implying that the node has the 'overtake-free' property of [11] which requires that the passage of the tagged task through the node is not influenced by tasks at any other node;

- The sojourn time at each node can be characterised as a first passage time in a Markov chain describing the node's behaviour during that sojourn *insofar as it affects the tagged task.*

Notice that the last point does not necessarily require the Markov chain describing the whole system or even the node: for example a transient chain representing a queue with no arrivals is sufficient if the first property holds. This is a traditional approach that was used to obtain the Laplace transform of response time distributions in cyclic, tree-like and overtake-free networks in the 1980s [11, 2, 5].

Our alternative approach uses the observation that sojourn times are the same whether one considers the forward process or its reversed process. For example, given initial state $\mathbf{i}_0 = (i_{0;1}, i_{0;2})$ in a two-node network, we might take the sojourn time at the first node in the forward process (conditioned on $i_{0;1}$) and the *reversed sojourn time* at the second node in the reversed process, conditioned on the state existing at the end of the two sojourns. Notice that the reversed sojourn time is not necessarily dependent on only the initial state pertaining to the second node (final state in the forwards process). Indeed, the reversed process itself may depend on the joint state of the whole system, even if the forward node was overtake-free. In fact, this approach turns out to be no easier than the naive, purely 'forwards' one and a better method is as follows.

Let the reversed sojourn time at node i be denoted by $\tilde{T}_i$ and suppose the middle state is $\mathbf{S}(T_1) = \mathbf{s}_0 = (s_{0;1}, s_{0;2})$, where the random variable $\mathbf{S}(t)$ denotes the state of the system at time t. Then the LST of the joint sojourn time distribution can be written

$$\begin{aligned}
J^*(\theta_1, \theta_2) &= \mathbb{E}\,[\,\mathbb{E}\,[e^{-(\theta_1 \tilde{T}_1 + \theta_2 T_2)} \mid \mathbf{S}(T_1)]] \\
&= \mathbb{E}\,[\,\mathbb{E}\,[\,\mathbb{E}\,[e^{-(\theta_1 \tilde{T}_1 + \theta_2 T_2)} \mid T_2, \mathbf{S}(T_1)] \mid \mathbf{S}(T_1)]] \\
&= \mathbb{E}\,[\,\mathbb{E}\,[e^{-\theta_2 T_2}\,\mathbb{E}\,[e^{-\theta_1 \tilde{T}_1} \mid T_2, \mathbf{S}(T_1)] \mid \mathbf{S}(T_1)]] \qquad (1)
\end{aligned}$$

Now suppose that the state vector is separable, so that $\mathbf{S}(t) = (S_1(t), S_2(t))$ as described above, and that the reversed sojourn time at node 1 depends only on the state existing at the arrival instant of the tagged task there in the reversed process. Then we have

$$J^*(\theta_1, \theta_2) = \mathbb{E}\,[\,\mathbb{E}\,[e^{-\theta_1 T_1}\,\mathbb{E}\,[e^{-\theta_2 \tilde{T}_2} \mid S_2(T_1)] \mid \mathbf{S}(T_1)]]$$

If further the (forward) sojourn time at node 1 depends only on its initial state S_1, we find

$$J^*(\theta_1, \theta_2) = \mathbb{E}_{S_1, S_2}[T_1^*(\theta_1 \mid S_1(T_1)) \tilde{T}_2^*(\theta_2 \mid S_2(T_1))] \qquad (2)$$

To summarise, the conditions we need to apply equation 2 to a two-node network are:

1. The state of the system is separable;

2. The reversed sojourn time at node 1 depends *solely* on the state existing at node 1 just after the tagged task completes service at node 1, i.e. on the first component of the middle state;

3. The sojourn time at node 2 depends *solely* on the state existing at node 2 just before the arrival of the tagged task there, i.e. on the second component of the middle state;

4. The sojourn, respectively reversed sojourn, time at nodes 2 and 1 can be characterised as first passage times in Markov chains describing the respective nodes' behaviour during that sojourn.

Conditions 3 and 4 are aided by a specification of the reversed process for node 1. This may be provided by the Reversed Compound Agent Theorem (RCAT), which induces a systematic way to construct the reversed process of a separable synchronisation between two Markov chains [3]; see the next two sections. Note that the reversed response time in a node is not, in general, a response time in the same sense and may be hard to determine even if the reversed process of the node is known, e.g. by RCAT. The above conditions can be relaxed, according to equation 1, but the ensuing analysis is very much more complex, involving the evolution of the joint state. We will look at an example of this when we consider a pair of G-queues in section 4.

Paths of more than two nodes can be handled recursively, building a path by adding one node at a time – at each stage, a two-node path is considered comprising the current (partial) path as the second node and a new node added as the first. This method derives all the known results on response time distributions in overtake-free queueing networks, as we discuss in section 3. Furthermore, it opens the door to a variety of non-queueing applications, but it must be remembered that the above conditions can be quite tricky to apply, especially the third and fourth.

In the next section we illustrate the new technique in queueing networks, obtaining a concise explanation of several previous results. We then consider further separable (in the sense of RCAT)applications in section 5, checking the required conditions against the specified forward process (for node 2) and the reversed process obtained from RCAT (for node 1).

3 Queueing networks

Queueing networks are relatively tractable since the M/M/1 queue is *reversible*, i.e. its reversed process is the same M/M/1 queue – a result routinely derivable by RCAT, but a well known fact anyway [10, 5, 3]. Moreover, the queue left behind by any departing task comprises precisely the tasks that arrived during its sojourn. Therefore, we have the following result:

Proposition 3.1 *At equilibrium, the reversed sojourn time in an M/M/1 queue has the same probability distribution as the forward sojourn time.*

Proof In the reversed process, the initial state is the number of tasks that arrived during the (forward) sojourn of the tagged task, n say. Consequently, the reversed sojourn time is the sum of $n + 1$ service times.[2] Conditioned on their respective initial states, therefore, the reversed sojourn time is equal in distribution to the (forward) sojourn time. The result now follows since the equilibrium probability distribution of the queue length immediately before an arrival is the same in both processes, these both being an M/M/1 queue. ♠

The result of the previous section is now easy to apply, in both open and closed queueing networks. We begin with a tandem pair and a cycle of two M/M/1 queues.

3.1 Tandem and cyclic pairs of queues

Consider first the tandem pair of queues depicted in figure 1 – the cyclic counterpart is simply obtained by connecting the departures of the second queue to the arrivals of the first.

The forward and reversed nodes are both shown; correspondingly, the forward and reversed sojourn times are illustrated for both nodes, as per section 2. Possible sample paths for the node 1 and node 2 forward processes are shown in figure 2, during the passage of the tagged task through the network. This task leaves behind a queue of length 4 (including the task in service) at node 1

[2] This uses the fact that the residual service time of the task being served on arrival (i.e. reversed departure) of the tagged task is distributed as a full (exponential) service time.

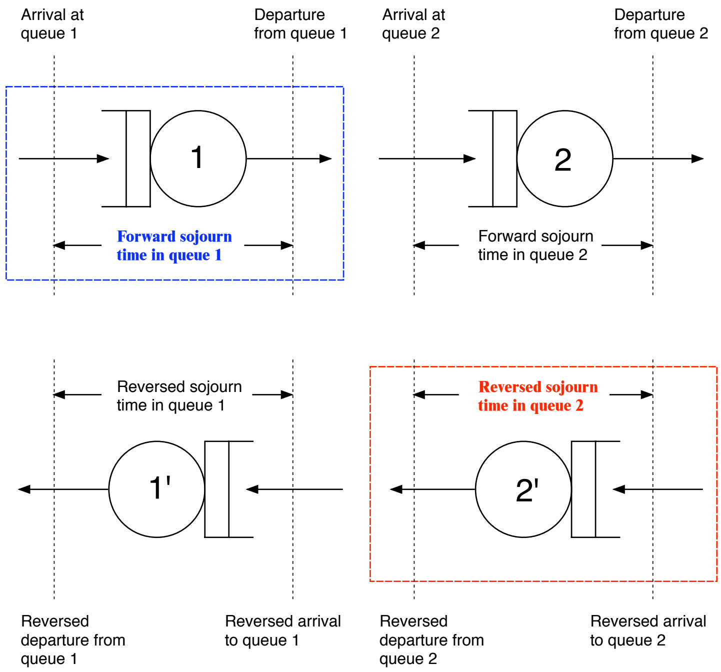
Arrival at queue 1
Departure from queue 1
Arrival at queue 2
Departure from queue 2
1
2
Forward sojourn time in queue 1
Forward sojourn time in queue 2
Reversed sojourn time in queue 1
Reversed sojourn time in queue 2
1'
2'
Reversed departure from queue 1
Reversed arrival to queue 1
Reversed departure from queue 2
Reversed arrival to queue 2

Figure 1: Two M/M/1 queues in tandem and the reversed process

on departure and finds a queue of length 3 just before its arrival at node 2, i.e. at the same instant, so that the middle state is (4,3). The traditional method of analysis investigates only forward sample paths and needs to consider the (transient) probability distribution of the node 2 queue length, starting with the middle state existing at the departure instant of the tagged task from node 1.

In our alternative approach, we consider the joint sample paths in the forward node 2 and reversed node 1 processes, beginning in a given middle state – (4,3) in the sample paths shown in figure 3. For the forward response time at node 2, we look to the right of the vertical axis and for the reversed response time at node 1, we look to the left. However, by equation 2, we only need to

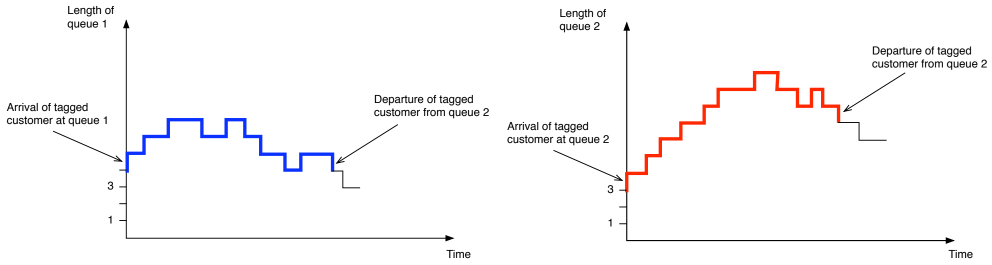

Figure 2: Possible sample paths for the queue lengths at each queue during the sojourn of the tagged task

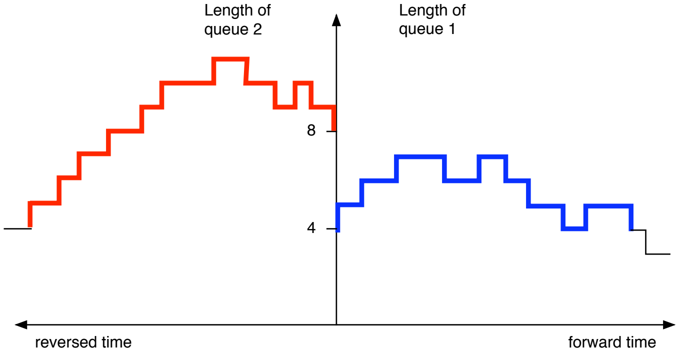

Figure 3: Forward and reversed sample paths given middle state (4,3)

condition on the middle state. Since forward and reversed sojourn times are identically distributed by proposition 3.1, we have:

$$\begin{aligned} J^*(\theta_1, \theta_2) &= \mathbb{E}_{S_1,S_2}[T_1^*(\theta_1 \mid S_1) T_2^*(\theta_2 \mid S_2)] \\ &= \sum_{n_1,n_2 \geq 0} \pi_{n_1 n_2} \left(\frac{\mu_1}{\mu_1 + \theta_1}\right)^{n_1+1} \left(\frac{\mu_2}{\mu_2 + \theta_2}\right)^{n_2+1} \end{aligned}$$

The equilibrium probabilities π are the standard product-form solution [9, 5], which is most easily derived by RCAT. In fact, an added advantage of RCAT is that it constructs the required reversed process for node 1, as well as providing the product-form. Here, we already know what this process is – the same M/M/1 queue – but we do not know this for general nodes, even G-queues (with negative customers) [1, 4].

The above result generalises inductively to overtake-free paths in both open and closed networks to give the following:

Proposition 3.2 *For overtake-free path $\mathbf{z} = (z_1, z_2, \ldots, z_m)$ in a queueing network of M nodes with state space $\mathcal{S}$ at equilibrium ($1 \leq m \leq M$), the LST of the joint sojourn time probability distribution is*

$$J^*(\theta_1, \ldots, \theta_m) = \sum_{(n_1,\ldots,n_M) \in \mathcal{S}} \pi_{n_1,\ldots,n_M} \prod_{j=1}^{m} \left(\frac{\mu_{z_j}}{\theta_j + \mu_{z_j}}\right)^{n_{z_j}+1}$$

where $\pi_{n_1,\ldots,n_M}$ is the equilibrium probability distribution of the network's state immediately prior to the instant of arrival of a task at any node.

Notice that $\pi_{n_1,\ldots,n_M}$ is well defined by the arrival theorem [5], being the same as an open network's steady state probabilities (at a random time point) or the steady state probabilities of a closed network with population reduced by one, depending on whether the network in question is open or closed, respectively.

In the case of open networks, $\pi_{n_1,\ldots,n_M}$ is a product of the form $\pi_1(n_1) \ldots \pi_M(n_M)$ where $\pi_i(n_i) = (1 - x_i)x_i^{n_i}$ for some constants x_i, and so the result simplifies to

$$J^*(\theta_1, \ldots, \theta_m) = \prod_{j=1}^{m} \frac{\mu_{z_j}(1 - x_{z_j})}{\theta_j + \mu_{z_j}(1 - x_{z_j})}$$

This is consistent with the fact that in a tandem series of stationary M/M/1 queues with fixed-rate servers and FCFS discipline, the sojourn times of a given task in each queue are independent. Interestingly, the proof of this result uses properties of reversibility and so we include it as an appendix, [10]. There is one obvious generalisation: the final queue in the series need not be M/M/1 since we are not concerned with its output. Also, the same result holds, by the same reasoning, when the final queue is M/G/c for $c > 1$. This contrasts with a similar result we get with our alternative approach in the next subsection.

In either approach, we observe that if service rates varied with queue length, we could not ignore tasks behind a given tagged task, even when they could not overtake, because they would influence the service rate received by the tagged task. Except in special cases, therefore, constant service rates are required.

4 G-queues and networks

4.1 Reversed sojourn time in a G-queue

The sojourn time probability distribution in a single G-queue at equilibrium can be determined by finding either

- the sojourn time probability distribution conditioned on the queue length faced on arrival and then deconditioning with respect to the equilibrium state probabilities – the same at arrival instants by the arrival theorem [**?**]; or
- the reversed sojourn time distribution conditioned on the queue length left behind on (forwards) departure and then deconditioning with respect to the equilibrium *departure state* probabilities. These are also the same as the equilibrium state probabilities in G-queues, since the reversed arrival process is Poisson.

The first method is detailed in [7]. Here we derive the same result using the second.

Consider a G-queue with positive arrival rate λ^+ of positive tasks, exponential service times with parameter μ and an additional Poisson arrival process of negative tasks, rate λ^-. This can be regarded as an M/M/1 queue with arrival rate λ^+ and service rate $\mu + \lambda^-$, with the departure stream split into normal service completions (rate μ) and killed tasks (rate λ^-). At equilibrium, the probability that the queue length is n is therefore $(1 - \rho)\rho^n$ where $\rho = \lambda^+/(\lambda^- + \mu)$. The reversed queue is therefore also an M/M/1 queue, with service rate $\mu + \lambda^-$ and two independent Poisson arrival streams with rates $\gamma_1 = \frac{\mu\lambda^+}{\lambda^- + \mu}$, for the reversed departures, and $\gamma_2 = \frac{\lambda^-\lambda^+}{\lambda^- + \mu}$, for the reversed killings.

For a tagged task arriving at the reversed queue, let R denote the (reversed) sojourn time random variable and $R^*(\theta)$ be the Laplace-Stieltjes transform (LST) of its probability distribution function. We also define the following random variables:

- N is the queue length just before the arrival of the tagged task (i.e. just after the departure of its forward counterpart);
- B, B_i for $i = 1, 2, \ldots$ are service times;

- W, W_i for $i = 1, 2, \ldots$ are busy periods arising from reversed negative killings.

Proposition 4.1 *Given queue length N on arrival, the reversed sojourn time probability distribution function has LST*

$$R^*_N(\theta) = w(\theta)^{N+1}$$

and the unconditional sojourn time LST is $R^*(\theta) = \frac{(1-\rho)w(\theta)}{1-\rho w(\theta)}$, where $w(\theta)$ is the smaller root of the equation

$$\gamma_2 w^2 - (\theta + \mu + \lambda^- + \gamma_2)w + (\mu + \lambda^-) = 0$$

Proof The derivation of $R^*(\theta)$ is based on the observation that the queue, of length N, left behind by a tagged task in the forwards process comprises precisely those tasks that arrived *after* that task and were not killed during its (forwards) sojourn time. This queue length is the same as that faced by the corresponding, arriving, reversed tagged task. The reversed sojourn time, conditional on the arrival queue length N, is therefore the sum of $N+1$ service times together with the service times of all those reversed killing departures that arrive during these service times. This is just the sum of $N+1$ busy periods in an M/M/1 queue with arrival rate γ_2 and service rate $\mu + \lambda^-$, i.e. $W_1 + \ldots + W_{N+1}$.

It is routine to show that the LST of the probability distribution of a generic busy period W is $B^*(\theta + \gamma_2(1 - W^*(\theta)))$.[3] For exponential service times, this implies that $W^*(\theta)$ is a solution for w of the quadratic equation $w = \frac{\mu+\lambda^-}{\theta+\mu+\lambda^-+\gamma_2(1-w)}$, i.e.

$$\gamma_2 w^2 - (\theta + \mu + \lambda^- + \gamma_2)w + (\mu + \lambda^-) = 0$$

Only the smaller root is valid, the larger one being greater than unity.

The conditional reversed sojourn time LST, given queue length N on arrival, now follows as $R^*_N(\theta) = \mathbb{E}\,[e^{-\theta(W_1+\ldots+W_{N+1})} \mid N] = W^*(\theta)^{N+1}$ and so the unconditional LST is

$$R^*(\theta) = \mathbb{E}\,[\,\mathbb{E}\,[W^*(\theta)^{N+1} \mid N]] = W^*(\theta) G_N(W^*(\theta))$$

where $G_N(z) = (1-\rho)z/(1-\rho z)$ is the probability generating function (pgf) of the queue length faced by a tagged customer on arrival in steady state, which is the same as the pgf of the equilibrium state at an arbitrary time by the Random Observer Property, see [6, **?**] for example. Thus, $R^*(\theta) = \frac{(1-\rho)w}{1-\rho w}$, where w is the smaller root of the above quadratic equation. ♠

This solution is easily shown to be the same as that originally derived in [7] by a completely different method, using an application of Rouche's theorem. Notice

[3] $W^*(\theta) = \mathbb{E}\,[e^{-\theta W}] = \mathbb{E}\,[e^{-\theta(B+W_1+\ldots+W_A}]$ where A is the number of arrivals in the service period B. Hence

$$\begin{aligned} W^*(\theta) &= \mathbb{E}\,[e^{-\theta B}\,\mathbb{E}\,[e^{-\theta(W_1+\ldots+W_A)} \mid B]] \\ &= \mathbb{E}\,[e^{-\theta B}\,\mathbb{E}\,[\,\mathbb{E}\,[e^{-\theta(W_1+\ldots+W_A)} \mid A, B] \mid B]] = \mathbb{E}\,[e^{-\theta B}\,\mathbb{E}\,[W^*(\theta)^A \mid B]] \\ &= \mathbb{E}\,[e^{-\theta B} e^{-\gamma_2(1-W^*(\theta))B}] = B^*(\theta + \gamma_2(1 - W^*(\theta))) \end{aligned}$$

that the reversed sojourn time depends solely on the queue length faced on arrival, the reversed node being an ordinary M/M/1 queue with two independent Poisson arrival streams, in contrast to a (forwards) G-queue. This property is crucial when analysing networks that include G-queues.

4.2 Joint sojourn times in a pair of G-queues

Suppose now that we have a tandem network comprising an M/M/1 queue and a G-queue that has an additional external arrival stream of negative tasks that remove the last task in the FCFS queue when it is non-empty [1]. If the G-queue is the first node, the conditions in section 2 are satisfied since the network is separable (by RCAT [4]), the second node is an M/M/1 queue with sojourn time depending only on the queue length existing on arrival, and the reversed sojourn time in the first queue depends only on the queue length at the (forward) departure instant – see proposition 4.1. The response time distribution therefore has LST which is the product of that for the G-queue and that for the M/M/1 queue with arrival rate equal to the positive throughput from queue 1, i.e. the product of the external positive arrival rate and the probability of a task not being 'killed'. This is precisely the result obtained in [7].

Now suppose the G-queue is the second node, node 1 being M/M/1. The network is still separable and the reversed sojourn time at the first node depends only on the state existing just after departure, as for any M/G/1 queue. However, the forwards sojourn time at the second node depends on the evolution of the first node since synchronised transitions with it influence the passage of the tagged task.[4] This leads to a complex, transient calculation, equivalent to that of [8]. The case in which both nodes are G-queues is even more complex, the actual solution involving a Fredholm integral equation of the second kind.

It is interesting to note that the response time LST is separable when node 1 is the G-queue, whereas an M/G/1 queue must be second if paired with an M/M/1, when there are no negative customers. This was highlighted as unexpected in [8] but it is clear in the new approach. Notice that if an M/G/1 queue were paired first with an M/M/1 queue, with FCFS queueing discipline, the network is not separable – it has long been known that no product-form then exists for the equilibrium queue length probabilities, and the conditions of RCAT correspondingly fail. However, separable solutions do exist with appropriately modified non-M/M/1 queues, as we shall see next.

5 Generalised networks

The most obvious route to finding a new separable response time distribution in a network of two nodes requires that the network satisfy RCAT (so the reversed process is separable) and that the reversed sojourn time is tractable at the first node. This is non-trivial since the first requirement itself implies a new product-form. We proceed by supposing that the first node is a modified queue with Erlang-2 service times (sum of two independent, identical exponential random variables) and the second is an M/M/1 queue. Note that a network with FCFS

[4] In the forwards process, arrivals from the first node offer 'protection' from the negative arrivals at node 2, in that such an arrival would be next to be killed; see [8].

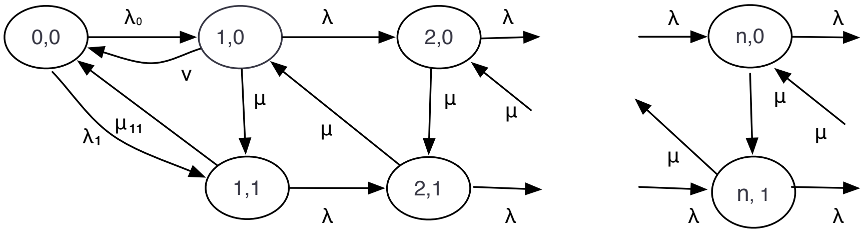

Figure 4: Transition diagram of the queue with Erlang-2 service

queueing discipline does not have a product-form when the Erlang-2 queue is unmodified. Clearly, then, the problems are to

- Find a modification for the Erlang-2 queue that produces a product-form – this can be done using RCAT;
- Define the reversed sojourn time in this modified queue and find its probability density function, conditioned on the initial state in the reversed process;
- Decondition.

5.1 Queue with Erlang-2 service

In this section we consider a queue which has Poisson arrivals λ and a two-phase Erlang-2 service with rate μ.

We carry out an analysis of the queue to find out the steady state probabilities and the reversed rates. From Figure 4 we write down directly the balance equations.

$$\begin{aligned}
\pi_{00}(\lambda_0+\lambda_1) &= \pi_{11}\mu_{11}+\pi_{1,0}\nu \quad (3)\\
\pi_{10}(\lambda+\mu+\nu) &= \pi_{21}\mu+\pi_{00}\lambda_0 \quad (4)\\
\pi_{11}(\mu_{11}+\lambda) &= \pi_{10}\mu+\pi_{00}\lambda_1 \quad (5)
\end{aligned}$$

The three equations written above represent the balance equation for the first three states: $(00), (10), (11)$.

For any state $n \geq 2$ the balance equations are written below:

$$\begin{aligned}
\pi_{n,0}(\lambda+\mu) &= \pi_{n+1,1}\mu+\pi_{n-1,0}\lambda \quad (6)\\
\pi_{n,1}(\lambda+\mu) &= \pi_{n-1,1}\lambda+\pi_{n,0}\mu \quad (7)
\end{aligned}$$

Assume that $\kappa\pi_{n,0} = \pi_{n+1,1}$ $(\forall n \geq 0)$, by substituting in 6 and 7 show that the ratio between successive states in the second phase of the queue is constant $\frac{\pi_{n+1,1}}{\pi_{n,1}} = K$ where:

$$\begin{aligned}
K &= \frac{\lambda}{(\lambda+\mu)-\kappa\mu} \quad (8)\\
K &= \kappa^2 \quad (9)
\end{aligned}$$

In order for the queue to reach steady state $K < 1$ which implies that $\kappa > 1$ as a constrain to our system. Thus by equating 8 and 9 we obtain a third order equation

$$\kappa^2(\lambda+\mu) - \kappa^3\mu - \lambda = 0 \tag{10}$$

There are three solutions the equation namely $\kappa_1 = 1, \kappa_{2,3} = \frac{-\lambda \pm \sqrt{\lambda^2+4\lambda\mu}}{2\mu}$, imposing that $\kappa_{2,3} > 1$ we obtain the condition that $4\lambda > \mu$.

We seek to express $\pi_{n,0}, \pi_{n,1}$ $(\forall n > 1)$ using the recurrent equation as follows:

$$\begin{aligned} \pi_{n,1} &= \kappa K^{n-1}\pi_{00} \\ \pi_{n,0} &= K^n \pi_{00} \end{aligned}$$

Using the normalising equation we can find the value for π_{00}.

$$\begin{aligned} \pi_{00} + \sum_{i=1}^{\infty} \pi_{i,0} + \sum_{i=1}^{\infty} \pi_{i,1} &= 1 \\ \pi_{00} + \sum_{i=1}^{\infty} K^i \pi_{00} + \sum_{i=1}^{\infty} \kappa K^{i-1}\pi_{00} &= 1 \\ \pi_{00} &= \frac{1-K}{(1+\kappa)} \\ \pi_{00} &= \frac{1-\kappa^2}{(1+\kappa)} \\ \pi_{00} &= 1-\kappa \end{aligned} \tag{11}$$

Proposition 5.1 *Assume that the Continous Time Markov Chain with transition diagram described in Figure 4 is ergotdic, then the steady state probabilities are the following:*

$$\begin{aligned} \pi_{00} &= 1-\kappa \\ \pi_{n,1} &= \kappa^{2n+1}(1-\kappa) \\ \pi_{n,0} &= \kappa^{2n}(1-k) \end{aligned}$$

for all $n > 1$ where $\kappa = \frac{-\lambda \pm \sqrt{\lambda^2+4\lambda\mu}}{2\mu}$.

To obtain the product from solution for the networks of queue, we need to find out the reversed rates of the queue Erlang-2 service. In any continuous time homogenous Markov chain we can calculate the reversed rate using the simple following result $\pi_{ij}q_{ij} = \overline{q}_{ji}\pi_{ji}$ where π_{ij} is the state state distribution [10].

$$\begin{aligned} \pi_{11}\mu_{11} &= \mu_{11}\kappa \\ \overline{\mu} &= \mu\kappa \\ \overline{\lambda} &= \tfrac{\lambda}{K} \\ \overline{\lambda_0} &= \tfrac{\lambda_0}{K} \end{aligned}$$

5.2 Product form solution

We consider the tandem of queue where the first queues has been described in the previous section and the second queue is an ordinary $M/M/1$ with rate δ

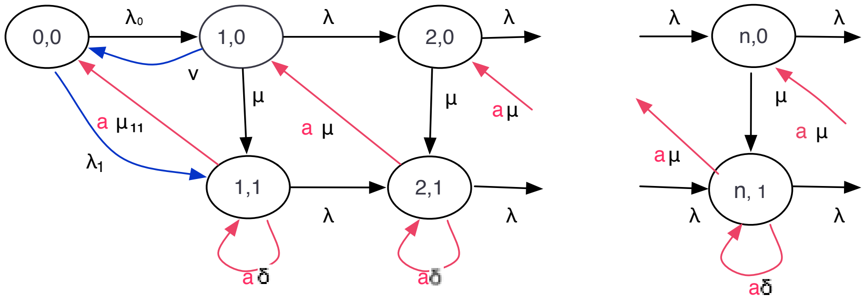

Figure 5: Transition diagram of Erlang-2 service in isolation

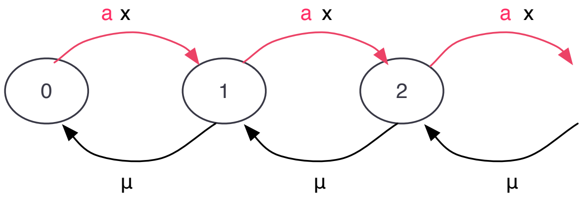

Figure 6: Transition diagram of the $M/M/1$ queue in isolation

for arrivals and rate γ for service. Arrivals in the second queue can happen in both phases of the service. We show the transition diagram of the queue with Erlang-2 in Figure 5. Note that this is different from Figure 4 as we have labelled with a the arcs that co-operate -red in the figure- with the $M/M/1$. There are two possible co-operation arcs: the ones that have rate μ represent customers leaving the queue to join the second queue, while the self loop corresponds to the independent arrival of a customer in the $M/M1$ which clearly does not change the state of the first queue i.e. state of the queue does not change. The queue describe in Figure 5 would have the same balance equations as the one analysed in Section 5.1 since the only new transitions are the self loops which do not modify the behaviour of the Markov Chain.

The second queue in the tandem is represented in Figure 6 and it is clearly a simple $M/M/1$ queue. Note that the rate of the arrival is passive (x is not a rate). The rate of x will be determined by the cooperation with the first queue. The rate of x will be the reversed rate of the rate of the co-operation arcs. Assuming that $x = \beta$ then the $M/M/1$ queue would be completely specified. Moreover if $\frac{\beta}{\gamma} < 1$ then we know that the steady state probabilities are $\pi_m \propto \frac{\beta}{\gamma}$ We can see the transition diagram of the joint state space of the tandem of queues in Figure 7.

It would be quite hard to find a product form solution. However, RCAT [3] simplify the job considerably. In fact it is easy to verify that:

1. Each passive action in the first queue in enabled;

2. Every reverse action of an active action type is always enabled in the reversed process in the joint space.

3. Each reversed action of type a is the same rate, which implies that $\kappa\mu = \overline{\mu} = \overline{\mu_{11}} = \kappa\mu$ which implies $\mu_{11} = \mu$ and $\delta = \kappa\mu$.

Thus, since these conditions are satisfied there is product form solution as stated in the next theorem.

Theorem 5.2 (Product form solution) *Assume a tandem of queues with transition diagram as in Figure 7 is ergodic then the equilibrium probability have a product form:*

$$\begin{array}{rcl} \pi_{[(0,0),m]} & \propto & \pi_{00}(\frac{\kappa\mu}{\gamma})^m \\ \pi_{[(n,0),m]} & \propto & \pi_{n,0}\frac{\kappa\mu}{\gamma})^m \\ \pi_{[(n,1),m]} & \propto & \pi_{n,1}(\frac{\kappa\mu}{\gamma})^m \end{array}$$

for all $n \geq 1$ and for all $m \geq 0$:

6 Observation on the rates

RCAT imposes that $\mu_{11} = \mu$. Moreover, in order to calculate the reverse response time in a simple way, we could assume that $\mu = 0$ and $\overline{\lambda_0} = \overline{\lambda}$ which implies that $\lambda_0 = \lambda$. With those constrains, the question is whether the system of linear equation provided by the balance equation can be satisfied. We observe under these assumptions just mentioned equation 4 becomes redundant. From 3 we obtain that $\lambda_1 = \kappa\mu - \lambda$ and from 5 we obtain that $\lambda_1 = \kappa(\mu + \lambda) - \kappa^2\mu$ then $\lambda_1 = \lambda(\kappa + 1) - \kappa^2\mu$.

6.1 Product-form tandem pair

creating additional external arrivals at node 1 by ensuring that all states in node 1 have an incoming, active, synchronising action type, using invisible actions [4].

7 Conclusion

Response times distributions – more generally, joint node-sojourn time distributions – can be derived much more simply and generally than previously using the reversed process of a separable network. In this way, most of the known separable solutions for the LSTs of response time distributions in queueing networks can be obtained. Moreover, many other special cases of product form solutions can be explained. The methodology provides a handle for such problems and certainly is conducive to automation. In fact, new product-forms for equilibrium state probabilities could provide a basis, since they would at least, via RCAT, provide the right, separable reversed node processes.

References

[1] E. Gelenbe. Random neural networks with positive and negative signals and product form solution. *Neural Computation*, 1(4):502–510, 1989.

[2] P.G. Harrison. The distribution of cycle times in tree-like networks of queues. *Computer Journal*, 27(1):27–36, 1984.

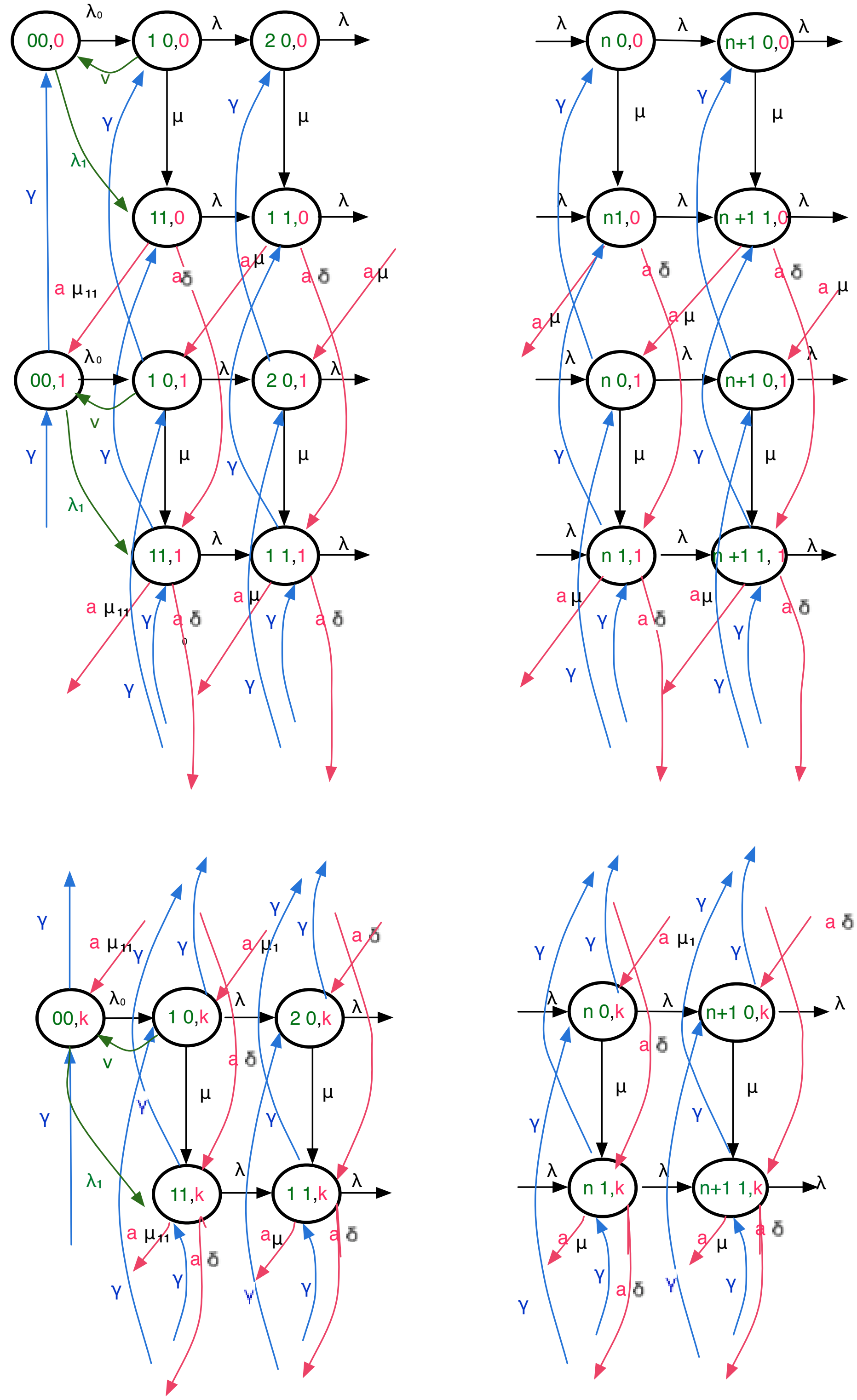

Figure 7: Transition diagram of the joint state space of the tandem of queues

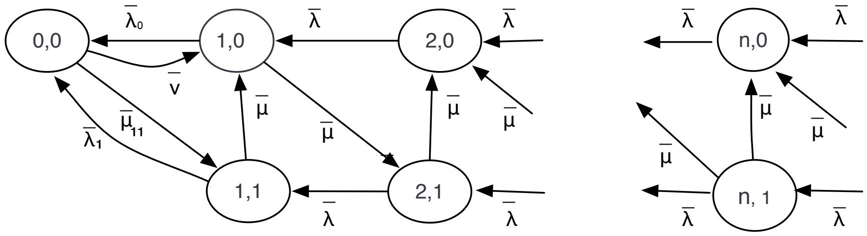

Figure 8: Transition diagram of the reversed queue with Erlang-2 service

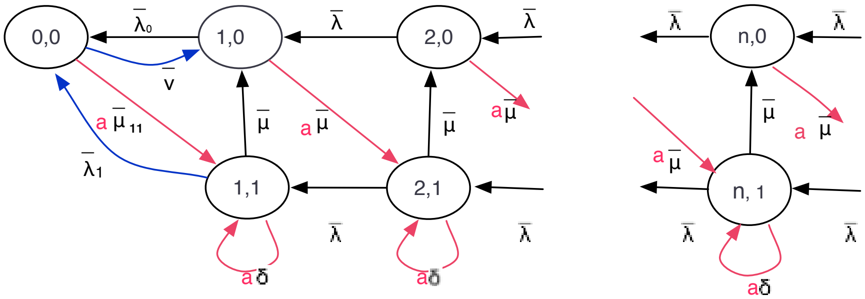

Figure 9: Transition diagram of the reversed queue with Erlang-2 service

[3] P.G. Harrison. Turning back time in markovian process algebra. *Theoretical Computer Science*, 290(3):1947–1986, January 2003.

[4] P.G. Harrison. Compositional reversed Markov processes, with applications to G-networks. *Performance Evaluation*, December 2004.

[5] P.G. Harrison and Naresh M. Patel. *Performance Modelling of Communication Networks and Computer Architectures.* Addison-Wesley, 1992.

[6] P.G. Harrison and N.M. Patel. *Performance Modelling of Communication Networks and Computer Architectures.* International Computer Science Series. Addison Wesley, 1993.

[7] P.G. Harrison and E. Pitel. Sojourn times in single server queues with negative customers. *Journal of Applied Probability*, 30:943–963, 1993.

[8] P.G. Harrison and E. Pitel. Response time distributions in tandem G-networks. *Journal of Applied Probability*, 32:224–246, 1995.

[9] J.R. Jackson. Jobshop-like queueing systems. *Management Science*, 10(1):131–142, 1963.

[10] F.P. Kelly. *Reversibility and stochastic networks.* Wiley, 1979.

[11] F.P. Kelly and P.K. Pollett. Sojourn times in closed queueing networks. *Advances in Applied Probability*, 15:638–656, 1983.

[12] Transaction Processing Performance Council. *TPC benchmark C: Standard specificationrevision 5.2.* 2003.

A Appendix: Proof of independence in tandem M/M/1 queues

Proposition A.1 *In a tandem series of stationary M/M/1 queues with fixed-rate servers and FCFS queueing discipline, the sojourn times in each queue of a tagged task are independent.*

Proof First we claim that the sojourn time of a tagged task, C say, in a stationary M/M/1 queue is independent of the departure process before the departure of C. This is a direct consequence of the reversibility of the M/M/1 queue.

To complete the proof, let A_i and T_i denote C's time of arrival and sojourn time respectively at queue i in a series of m queues ($1 \leq i \leq m$). Certainly, by our claim, T_1 is independent of the arrival process at queue 2 before A_2 and so of the queue length faced by C on arrival at queue 2. Thus, T_2 is independent of T_1. Now, we can ignore tasks that leave queue 1 after C since they cannot arrive at (nor influence the rate of) any queue in the series before C, again because all queues have single servers and FCFS discipline. Thus, T_1 is independent of the arrival process at queue i before A_i and so of T_i for $2 \leq i \leq m$. Similarly, T_j is independent of T_k for $2 \leq j < k \leq m$. ♠