Theoretical Computer Science 290 (2003) 1947 – 1986
Theoretical Computer Science
www.elsevier.com/locate/tcs

# Turning back time in Markovian process algebra

Peter G. Harrison
*Department of Computing, Imperial College of Science, Technology and Medicine, University of London, 180 Queen's Gate, London SW7 2BZ, UK*

Received 2 April 2001; received in revised form 16 July 2001; accepted 28 March 2002
Communicated by M. Jerrum

**Abstract**

Product-form solutions in Markovian process algebra (MPA) are constructed using properties of reversed processes. The compositionality of MPAs is directly exploited, allowing a large class of hierarchically constructed systems to be solved for their state probabilities at equilibrium. The paper contains new results on both reversed stationary Markov processes as well as MPA itself and includes a mechanisable proof in MPA notation of Jackson's theorem for product-form queueing networks. Several examples are used to illustrate the approach.


## 1. Introduction

Stochastic process algebra (SPA)—see, for example, [10,15,2,11,7]—is a formalism developed over the last decade that can describe rigorously both the qualitative (functional) and quantitative (performance-related) behaviour of systems of interacting processes. The principal advantage of this algebraic approach to modelling is the property of *compositionality* possessed by all SPAs. This means that two or more fully specified systems can be combined together (as subsystems) into a more complex system in a simple way—both syntactically and semantically. The behaviours of the subsystems are not affected, except where they are explicitly connected to each other. Unfortunately, compositionality alone is, in general, insufficient to facilitate a hierarchical, inductive analysis of most properties. For example, absence of deadlock is not compositional in this sense. The practical advantage of compositionality is to precisely separate those properties which are local to the components of a system from those

*E-mail address:* pgh@doc.ic.ac.uk (P.G. Harrison).


PII: S0304-3975(02)00375-4

involved in interactions between them. Thus, although it offers the prospect of efficient performance models, these do not come automatically.

In a Markovian process algebra (MPA), such as [11,7], all actions have an exponential duration leading to a Markov model. In general the state transition graph of this model is computed, from which steady-state probabilities and other quantities can be calculated by standard methods. The role of the MPA in this procedure is to facilitate a concise specification language from which the transition graph can be generated mechanically and reliably. Hand-produced graphs for non-trivial systems are notoriously error-prone, it being all too easy to overlook certain state-transitions. However, very large state spaces can be generated for quite simple systems and the direct solution route therefore has limited value, although some important results have been obtained for state spaces of arbitrary structure [17,18], using compositional minimisation and Markovian bisimulation [14], and via the Kronecker algebra [19]. Some transition graphs have special structures that facilitate efficient solutions, such as those of *product-form*. In general, it is difficult to establish that such structures exist, but if models are constructed in a certain hierarchical way, compositionality can be used to preserve the product-form properties of the constituent components [8,20,12,4].

In this paper, we use the MPA of Hillston, PEPA [11], although our approach could also be applied to other MPAs, providing they can express the appropriate kind of interaction between components, described in Section 3. The crux of our analysis is the identification of reversed processes [16], from which product-form solutions follow immediately in suitably defined, separable systems such as queueing networks. Traditionally, this identification has been based on heuristics applied to state transition graphs but we derive a methodology directly in the PEPA syntax, combined with a compositional result for dealing with cooperating processes.

In Section 2 we consider reversed processes as described in [16]. We obtain a new result that determines the reversed process of a stationary Markov process solely in terms of the processes' instantaneous transition rates, which can be obtained directly from their PEPA specifications. This leads to an algorithm for calculating the generators of a reversed process directly. In Section 3, we introduce a methodology for constructing reversed processes from transition graphs given by PEPA agents. [1] We apply this methodology to, first, simple (with no cooperation combinators) and then compound agents, including our main result, the Reversed Compound Agent Theorem (RCAT). Several examples are given in the following two sections, relating to simple and compound agents, respectively. In the latter case, Jackson's theorem for product-form queueing networks [13,9] and results obtained on quasi-reversibility and the so-called QR-agents [16,9,8] follow. In fact, given a suitable support environment for PEPA, the derivation of such theorems could be mechanised in the new approach. The paper concludes in Section 7 where we outline some directions for further work.

[1] Agent is the term used for a process in PEPA and many other process algebras.

## 2. Reversed processes

The analysis of stationary Markov processes, i.e. those in a steady-state, can often be greatly simplified by considering a dual process which has the same state space but in which the direction of time is reversed, cf. viewing a video film backwards. This dual process is known as the reversed process, and if it happens that the reversed process is stochastically identical to the original process, that process is called *reversible*.

A reversible process satisfies—as a necessary and sufficient condition for reversibility—the detailed balance equations

$$\pi_i q_{ij} = \pi_j q_{ji} \quad \text{for all states } i \neq j,$$

where $Q=(q_{ij})$ is the process's instantaneous state transition rate matrix (its *generators*, apart from the diagonal) and $\boldsymbol{\pi}$ is its equilibrium probability mass function (distribution) vector.

Most Markov processes are not reversible but we can still define the *reversed process* $X_{\tau-t}$ when the Markov process $X_t$ is not reversible. Of course, since the Markov property can be expressed as 'the past and future are independent when conditioned on the present', a reversed Markov process must also be a Markov process. The instantaneous transition rates of the reversed process are related to those of the original process through the stationary distribution $\boldsymbol{\pi}$, which is the same for both processes. Indeed, the relationships can be so simple that vectors satisfying them can often be found by inspection. Then, by uniqueness, the normalised vector chosen for $\boldsymbol{\pi}$ must be the actual stationary distribution. However, we would like to find a methodology for achieving this solution which is independent of the solution itself.

The basic results relating a stationary Markov process to its reversed process are the following. We say that a stochastic process $\{X_t \,|\, -\infty < t < \infty\}$ is *stationary* if $(X_{t_1}, X_{t_2}, \ldots, X_{t_n})$ and $(X_{t_1+\tau}, X_{t_2+\tau}, \ldots, X_{t_n+\tau})$ have the same probability distribution for all times $t_1, t_2, \ldots, t_n$ and $\tau$. The reversed process of $\{X_t\}$ is the (necessarily) stationary process $\{X_{\tau-t}\}$ for any real number $\tau$.

A process is said to be *reversible* if its reversed process is stochastically identical. Formally, we have:

**Definition 1.** The process $\{X_t\}$ is reversible if

$$(X_{t_1}, X_{t_2}, \ldots, X_{t_n}) \quad \text{and} \quad (X_{\tau-t_1}, X_{\tau-t_2}, \ldots, X_{\tau-t_n})$$

have the same probability distribution for all times $t_1, t_2, \ldots, t_n$ and $\tau$.

Although a stationary process need not be reversible, every reversible process is stationary. This follows since if the process $\{X_t\}$ is reversible, both $(X_{t_1}, X_{t_2}, \ldots, X_{t_n})$ and $(X_{t_1+\sigma}, X_{t_2+\sigma}, \ldots, X_{t_n+\sigma})$ have the same distribution as $(X_{-t_1}, X_{-t_2}, \ldots, X_{-t_n})$ for all $\sigma$; take $\tau=0$ and $\sigma$, respectively, in the above. However, we will not be considering reversible processes *per se* here.

### 2.1. Determination of reversed processes

It is straightforward to find the reversed process of a stationary Markov process if the stationary state probabilities are known.

**Proposition 1.** *The reversed process of a stationary Markov process* $\{X_t\}$ *with state space* $S$, *generator matrix* $Q$ *and stationary probabilities* $\boldsymbol{\pi}$ *is a stationary Markov process with generator matrix* $Q'$ *defined by*

$$q'_{ij} = \pi_j q_{ji}/\pi_i \quad (i, j \in S)$$

*and with the same stationary probabilities* $\boldsymbol{\pi}$.

**Proof.** For $i \neq j$ and $h > 0$, $P(X_{t+h} = i)P(X_t = j | X_{t+h} = i) = P(X_t = j)P(X_{t+h} = i | X_t = j)$. Thus,

$$P(X_t = j | X_{t+h} = i) = \frac{\pi_j}{\pi_i} P(X_{t+h} = i | X_t = j)$$

by stationarity. Dividing by $h$ and taking the limit $h \to 0$ yields the required equation for $q'_{ij}$ when $i \neq j$. But, when $i = j$,

$$-q'_{ii} = \sum_{k \neq i} q'_{ik} = \sum_{k \neq i} \frac{\pi_k q_{ki}}{\pi_i} = \sum_{k \neq i} q_{ik} = -q_{ii}.$$

That $\boldsymbol{\pi}$ is also the stationary distribution of the reversed process now follows since

$$-\pi_i q'_{ii} = \pi_i \sum_{k \neq i} q_{ik} = \sum_{k \neq i} \pi_k q'_{ki}. \qquad \square$$

Sometimes, we can use this result directly to obtain the equilibrium distribution of a stationary Markov process in the following way: guess possible transition rates $\{q'_{ij} \mid i, j \in S\}$ for the reversed process and a collection of positive real numbers $\{\pi_i \mid i \in S\}$ with finite sum $G$ such that

- $q'_i = q_i$ for all $i \in S$ (where $q_i \equiv -q_{ii}$ is the total rate out of state $i$),
- $\pi_i q'_{ij} = \pi_j q_{ji}$ for all $i \neq j \in S$.

Then it follows immediately that $\boldsymbol{\pi}$ satisfies the Kolmogorov (balance) equations of the Markov process and so $\{\pi_i / G \mid i \in S\}$ is its equilibrium distribution by uniqueness. Of course, this approach depends crucially on the ability to make the right guess and it is the intuition associated with a reversed process, together with the fact that it has the same equilibrium state space distribution, that induces "good guesses". We can start out by literally imagining the physical system working backwards in time. For example, departures from a system become arrivals in the reversed process and it is plausible that the arrivals in the reversed process will have the same rate as the arrivals in the original process. Guessing like this for the $M/M/1$ queue with constant arrival rate $\lambda$ and service rate $\mu$, we can verify that the reversed process is the same $M/M/1$ queue. The usual steady-state solution then follows immediately since $\pi_i \lambda = \pi_{i+1} \mu$ for queue lengths $i \geqslant 0$ and we get Burke's theorem [3] as a bonus since arrivals of the reversed process are Poisson and identical to the departures of the original $M/M/1$ queue. Of course, the same argument applies when the arrival and service rates depend on the current queue length.

The $M/M/1$ queue is a very simple birth–death process, which is reversible. For non-trivial systems, we need a methodology to find reversed processes without reference to the sought-after stationary probabilities. This is provided by the following proposition, which places conditions only on the instantaneous transition rates of a Markov process. These rates are always available in the specification of a Markov process; in particular, they are explicit in a PEPA agent. The proposition is a generalisation of that relating to Kolmogorov's criteria for reversible Markov processes, given in [16,9].

**Proposition 2** (Kolmogorov's generalised criteria). *A stationary Markov process with state space $S$ and generator matrix $Q$ has reversed process with generator matrix $Q'$ if and only if*
1. $q'_i = q_i$ *for every state* $i \in S$.
2. *For every finite sequence of states* $i_1, i_2, \dots, i_n \in S$,

$$q_{i_1 i_2} q_{i_2 i_3} \dots q_{i_{n-1} i_n} q_{i_n i_1} = q'_{i_1 i_n} q'_{i_n i_{n-1}} \dots q'_{i_3 i_2} q'_{i_2 i_1} \tag{1}$$

**Proof.** If $Q'$ is the generator matrix of the reversed process, $\pi_j q_{jk} = \pi_k q'_{kj}$ for all $j,k \in S$. Taking $(j,k) = (i_1,i_2),(i_2,i_3),\dots,(i_{n-1},i_n),(i_n,i_1)$ in turn and multiplying then yields Eq. (1).

Conversely, suppose that Eq. (1) is satisfied. For all $j,k \in S$, we can find a chain $j \to j_1 \to \cdots \to j_{n-1} \to k$ (for $n \geqslant 1$) of one-step transitions since the Markov process is irreducible. Suppose that $q_{kj} > 0$. Then there is a chain $j \to j_1 \to \cdots \to j_{n-1} \to k \to j$ and so by Eq. (1) there is also a chain in the reversed process $j \to k \to j_{n-1} \to \cdots \to j_1 \to j$. Hence $q_{kj} > 0 \Rightarrow q'_{jk} > 0$ for all $j,k \in S$.

Now pick an arbitrary state $i_0 \in S$ as a reference state and for state $i \in S$, let $i \to i_{n-1} \to \cdots \to i_0$ $(n \geqslant 1)$ be a chain of one-step transitions in the forward process. Let

$$\pi_i = C \prod_{k=1}^{n} \frac{q'_{i_{k-1} i_k}}{q_{i_k i_{k-1}}},$$

where $i_n = i$ and $C$ is a positive constant. $\pi_i$ is well defined since if $i = j_m \to j_{m-1} \to \cdots \to j_0 = i_0$ is another chain, we can always find a chain $i_0 = h_0 \to h_1 \to \cdots \to h_l = i$. Eq. (1) then ensures that

$$\prod_{k=1}^{m} \frac{q'_{j_{k-1} j_k}}{q_{j_k j_{k-1}}} = \prod_{k=1}^{n} \frac{q'_{i_{k-1} i_k}}{q_{i_k i_{k-1}}},$$

since

$$\prod_{k=1}^{l} q_{h_{k-1} h_k} \prod_{k=1}^{n} q_{i_k i_{k-1}} = \prod_{k=1}^{n} q'_{i_{k-1} i_k} \prod_{k=1}^{l} q'_{h_k h_{k-1}}$$

and

$$\prod_{k=1}^{l} q_{h_{k-1}h_k} \prod_{k=1}^{m} q_{j_k j_{k-1}} = \prod_{k=1}^{m} q'_{j_{k-1}j_k} \prod_{k=1}^{l} q'_{h_k h_{k-1}}$$

Dividing then proves well-definedness. Now suppose that $q_{ji} > 0$ for $j \in S$. Then,

$$\pi_j = C \frac{q'_{ij}}{q_{ji}} \prod_{k=1}^{n} \frac{q'_{i_{k-1}i_k}}{q_{i_k i_{k-1}}},$$

so that $\pi_j q_{ji} = \pi_i q'_{ij}$. That the reversed process has transition rate matrix $Q'$ now follows from Proposition 1. □

Note that the same theorem holds if we consider only *minimal cycles* instead of all cycles. A cycle is minimal if it cannot be constructed as a union of two or more smaller cycles, where, in such a union, an arc from $i \to j$ nullifies an arc from $j \to i$, i.e. the pair can be omitted from the composite cycle.

### 2.2. Asymptotic irreducible subsets of states

It may be that the Markov process of interest is given by the limit of a family of stationary, ergodic processes. Moreover, this limiting process may be reducible (even though every process in the said family is irreducible) and the process of interest be defined on an irreducible subset of the so-called 'essential states'. This situation sometimes arises when the cooperation combinator is applied in PEPA, i.e. in a compound agent; see Section 3.1. Under appropriate conditions, a result of Anisimov [1] shows that the reversed rates in the irreducible sub-processes are the limits of the corresponding reversed rates in the family of processes defined on the whole state space.

## 3. MPA agents

We conduct our analysis using an abbreviated PEPA syntax, the full syntax and semantics of the PEPA language being given in [11]. We have just three constructions:

1. The prefix combinator defines an agent $(a, \lambda).P$ that carries out action $(a, \lambda)$ of *type* $a$ at *rate* $\lambda$ and subsequently behaves as agent $P$.
2. The agent describing the cooperation of two agents $P$ and $Q$ which synchronise over actions with types in a specified set $L$ is written $P \underset{L}{\bowtie} Q$.
3. A new *constant* agent $A$ is defined by the assignment combinator $A \stackrel{\text{def}}{=} P$ to have the same behaviour as $P$.

Using this syntax, a *choice* (available in the full PEPA syntax) is expressed by multiple assignments:

$$A \stackrel{\text{def}}{=} P,$$
$$A \stackrel{\text{def}}{=} Q.$$

This is equivalent to $A \stackrel{\text{def}}{=} P + Q$ in conventional PEPA. Action hiding could easily be added to this syntax but will not be used in this paper.

In a cooperation $P \underset{L}{\bowtie} Q$, the agents $P$ and $Q$ proceed independently with any actions whose types do not occur in the cooperation set $L$. However, actions with types in $L$ are only enabled in $P \underset{L}{\bowtie} Q$ when they are enabled in both $P$ and $Q$. In standard PEPA, the shared action occurs at the rate of the slowest participant. Here, however, we require that for each action type in a cooperation set $L$, exactly one agent (either $P$ or $Q$) is *passive* and effectively its synchronising action has rate $\top = \infty$. This means that the passive agent does not influence the rate at which a shared action occurs, essentially waiting for the other agent. We often distinguish the instances of $\top$ that may occur in a cooperation, writing $\top_a, \top_b, \dots$ for action types $a, b, \dots$, for example.

We introduce *relabelling*, which preserves the semantics but will be useful in defining the reversed processes of cooperations: $P\{y \leftarrow x\}$ denotes the process $P$ in which all occurrences of the symbol $y$ have been changed to $x$, which may be an expression. $y$ may be an action, of the form $(y_1, y_2)$, an action-type or a rate. Thus, for example, $((a, \lambda).P)\{\lambda \leftarrow \mu\}$ denotes the agent $(a, \mu).P\{\lambda \leftarrow \mu\}$. Relabelling also applies to symbols in cooperation sets; these are not considered 'hidden' through tight binding. Thus, $(P \underset{L}{\bowtie} Q)\{y \leftarrow x\} = P\{y \leftarrow x\} \underset{L\{y \leftarrow x\}}{\bowtie} Q\{y \leftarrow x\}$.

We call an agent defined using only assignments and prefixes *simple* and *compound* if it contains at least one instance of the cooperation combinator.

### 3.1. The underlying Markov process

The set of actions which an agent $P$ may next engage in—the *current actions* of $P$—is denoted by $\mathscr{A}ct(P)$, which can be defined inductively over the structure of $P$. When the system is behaving as agent $P$, these are the actions that are enabled. The states thus resulting from $P$ are called the *derivatives* of $P$. If $P$ can perform the action $(a, \lambda)$ and then become $P'$, we write $P \xrightarrow{(a,\lambda)} P'$ and say that $P'$ is an *a-derivative* of $P$. The *derivative set*, denoted $ds(P)$, of an agent $P$ is the transitive closure of all its derivatives and is defined by recursion. This defines a labelled transition system as a semantic model for PEPA.

The *derivation graph*, formed by syntactic PEPA terms at the nodes, with arcs representing the transitions between them, determines the underlying Markov process of an agent $P$. The *transition rate* between two agents $C_i$ and $C_j$, denoted $q(C_i, C_j)$, is the sum of the action rates labelling arcs connecting node $C_i$ to node $C_j$. It is this graph that will form the basis of our analysis, but we will find that its explicit construction is often unnecessary for compound agents. In fact, a PEPA agent is associated not only with the continuous-time Markov chain defined by its derivation graph but also with one of the states in that graph—the *initial state*. If the derivation graph is irreducible, i.e. defines an irreducible Markov process, the choice of initial state is arbitrary at equilibrium.

### 3.2. Reducible PEPA cooperations

It may be that an agent describes a reducible Markov chain and we are interested in the steady-state behaviour of an irreducible sub-chain. The initial state will then determine into which of the irreducible subchains, if any, the process can enter. In particular, this situation may arise when the cooperation combinator is applied, i.e. in a compound agent. Consider, for example, a Markovian network of two doubly interconnected queues with one arrival stream and one departure stream, easily defined in PEPA, see Section 6.3. This network is irreducible over the Cartesian product of the state spaces of the constituent queues considered in isolation, i.e. over the set of all pairs of non-negative integers, for all non-zero arrival rates and non-zero departure probabilities. However, as the arrival rate and departure probability both tend to zero, the network becomes closed and its state space becomes reducible with one irreducible subset of states for every value of the population. In the limit, the underlying Markov process partitions into disjoint irreducible sub-chains; every state belongs to exactly one of these sub-chains. The same applies to arbitrary (irreducible) closed queueing networks and, in fact, we will see that Jackson's theorem [13] for closed networks follows from that for open networks by application of our Theorem 1; see Section 6.4.

More generally, a state in a PEPA cooperation between agents with irreducible derivation graphs can be either (a) in an irreducible sub-chain; (b) absorbing; or (c) transient. The second possibility is a special case of the first, i.e. an irreducible sub-chain with only one state. To illustrate this, consider the specification $P_1 \underset{\{a,b,c\}}{\bowtie} Q_1$ where

$$
\begin{aligned}
P_1 &= \alpha.P_2, & Q_1 &= \alpha.Q_2,\\
P_1 &= \gamma.P_2, & Q_2 &= \alpha.Q_3,\\
P_2 &= \alpha.P_3, & Q_3 &= \beta.Q_1,\\
P_3 &= \beta.P_1, & Q_3 &= \gamma.Q_2
\end{aligned}
$$

and $\alpha=(a,\lambda)$, $\beta=(b,\mu)$, $\gamma=(c,v)$. Then, in the derivation graph of $P_1 \underset{\{a,b,c\}}{\bowtie} Q_1$, shown in Fig. 1, the states $\{(P_1,Q_1),(P_2,Q_2),(P_3,Q_3)\}$ form an irreducible sub-chain, $(P_2,Q_3)$, $(P_3,Q_2),(P_3,Q_1)$ are absorbing (singleton irreducible sub-chains), $(P_1,Q_2),(P_2,Q_1)$ are transient, leading to an absorbing state, and $(P_1,Q_3)$ is transient, leading to the first irreducible sub-chain.

Note that it is not necessary for a transient state to lead into an irreducible sub-chain. Instead there may exist an infinite sequence of transient states, as in the cooperation $P'_1 \underset{\{a,b,c\}}{\bowtie} Q_1$ where $Q_1$ is as above and

$$
\begin{aligned}
P'_1 &= \alpha.P'_2,\\
P'_1 &= \gamma.P'_2,\\
P'_2 &= \alpha.P'_3,\\
P'_n &= \beta.P'_1 \qquad (n\geqslant 3),\\
P'_n &= \delta.P'_{n+1} \quad (n\geqslant 3).
\end{aligned}
$$

### 3.3. Arrow inversion

Consider an agent $X$ defined by $X \stackrel{\text{def}}{=} E$ for some agent-expression $E$, together with a number of assignments to constants occurring in $E$, defined in the above syntax.

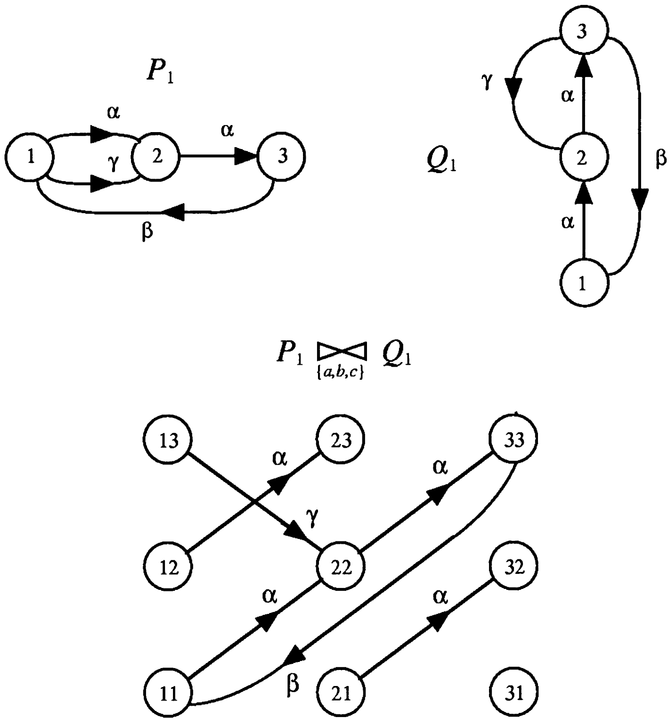


Fig. 1. Reducible derivation graph.

Without loss of generality we may assume that all assignments (including $X \stackrel{\text{def}}{=} E$) are of the form $A \stackrel{\text{def}}{=} (a, \lambda).P$ or $A \stackrel{\text{def}}{=} Q \underset{L}{\bowtie} R$ where $P, Q, R$ are constants appearing on the left of other assignments. If this property were not to hold, it would be trivial to transform the PEPA specification into one where it did by introducing new constants and assignments.

With this assumption, an agent $\bar{X}$ with state transitions (in the derivation graph) in the reverse direction to those of $X$ is easily defined recursively:

1. For each assignment in the agent of the form

   $$A \stackrel{\text{def}}{=} (a, \lambda).P$$

   (for constant $P$), the reversed agent has an assignment of the form

   $$\bar{P} \stackrel{\text{def}}{=} (\bar{a}, \bar{\lambda}).\bar{A}.$$

2. For each assignment in the agent of the form

   $$A \stackrel{\text{def}}{=} Q \underset{L}{\bowtie} R$$

(for constants $Q, R$), the reversed agent has an assignment of the form

$$\bar{A} \stackrel{\text{def}}{=} \overline{Q} \underset{\bar{L}}{\bowtie} \bar{R},$$

where $\bar{L} = \{\bar{a} \mid a \in L\}$.

Thus, an action $\alpha = (a, \lambda)$ induces the reversed action $\bar{\alpha} = (\bar{a}, \bar{\lambda})$ for some rate $\bar{\lambda}$ to be determined. The symbol $\bar{a}$ is arbitrary and simply labels the reversed arc corresponding to the action type $a$ in the original derivation graph.

Note that this definition gives the correct reversed derivation graph with the initial states of the (reversed) agents unchanged. Since initial states are arbitrary in stationary, irreducible Markov chains, there is no problem with correctness. However, it may sometimes be that a reversed agent has a non-intuitive initial state.

## 4. Rates of reversed actions

Although it is straightforward to find a PEPA agent definition $\bar{X}$ that has a derivation graph with arrows in the opposite direction to those of a given agent $X$, the challenge is to find the appropriate rates to render $\bar{X}$ the reversed process of $X$ in the sense of Section 2. Here, we describe a methodology to find these rates for simple agents—essentially by examining the state transition graph of the agent $X$—and prove a theorem that identifies the reversed process of a cooperation in terms of the reversed processes of its constituents.

### 4.1. Simple agents

Simple agents can be used to define an arbitrary Markov process. Hence, investigation of their reversed processes is tantamount to analysing the state transition graph of the Markov process directly, e.g. by using Kolmogorov's (extended) criteria of Proposition 2. We therefore propose the following methodology which we illustrate in Section 5. Given a graph $G$ with state space (set of nodes) $S$, suppose the transition rate from state $i$ to state $j$ is $q_{ij}$, with the total rate out of state $i$ being $q_i$. As in Section 2, we denote reversed rates with primes. We then determine the reversed graph $\bar{G}$ thus:

1. Use the conservation of outgoing rate to equate $q'_i = q_i$ for all states $i \in S$.[2]
2. Find a covering set of cycles, $C$, in the sense that every cycle in $G$ is a composition of cycles in $C$. For example, we may choose the set of all cycles or the set of all the minimal cycles in $G$. $C$ may be infinite (if $S$ is infinite) but often cycles will repeat in a parameterised way and so only finitely many need be considered. Each parameterised cycle may represent cycles for arbitrarily many values of one or more parameters.
3. For each (parameterised) cycle in $C$, apply Kolmogorov's criteria (Proposition 2). That is, equate the *known* product of the rates around the cycle in $G$ with the

[2] For every node $i$ in $G$ with only *one incoming arc*, from node $j$ say, we can set $q'_{ij} = q_i$. This often results in a significant number of rates being set in the reversed process immediately, simplifying the algorithm.

symbolic product of the rates around the reversed cycle in $\bar{G}$. Some of the reversed rate variables $q'_{ij}$ may be known or constrained from steps 1 and 2 or from previous cycles in this step, simplifying the resulting equations.

The result will be a system of non-linear equations that uniquely determine the reversed rates in $\bar{G}$. This must be so because of the necessity and sufficiency of Kolmogorov's criteria and the uniqueness of the equilibrium state probabilities in an ergodic time Markov chain. Algorithms for identifying cycles, minimal cycles and parameterised minimal cycles are not the focus of this paper, but their significance should become more apparent in the examples of Section 5. In general, it is not clear whether or not determining the reversed rates in this way is more efficient than calculating the equilibrium state probabilities in the usual way (by solving the linear balance equations) and using Proposition 1. Recall that one reason for analysing simple agents is to provide base cases for a compositional analysis of larger Markov chains; either approach should be numerically tractable.

### 4.2. Multiple actions

Any continuous time Markov chain can be described using only simple agents. However, a bundle of multiple actions performable by an agent, which all lead to the same derivative, cause multiple arcs in the derivation graph and so between two states in the transition graph of the underlying Markov chain. As already noted, this does not cause a problem in the semantics since the instantaneous transition rate between two states in the Markov chain is the sum of the rates on all the arcs between them. Moreover, we can always determine the *total* reversed rate between any two such states with multiple arcs between them using any of the methods described previously—including solving the balance equations and using Proposition 1.

However, we need to consider multiple actions individually in cooperations. This is because an agent may have several actions leading to the same derivative that synchronise with distinct actions in a cooperating component. For example, departures from a queue may join a number of different queues or leave the system (with no cooperation). In the reversed cooperation, the portion of the total reversed rate allocated to each individual reversed arc is crucial. A rule is therefore required to decide how to apportion the rates on multiple reversed arcs: there will be one reversed arc for each of the forward arcs in a bundle. The rule we use is the following:

**Definition 2.** The reversed actions of multiple actions $(a_i, \lambda_i)$ for $1 \leqslant i \leqslant n$ that an agent $P$ can perform, which lead to the same derivative $Q$, are respectively

$$(\overline{a_i}, (\lambda_i/\lambda)\bar{\lambda})),$$

where $\lambda = \lambda_1 + \cdots + \lambda_n$ and $\bar{\lambda}$ is the reversed rate of the one-step, composite transition with rate $\lambda$ in the Markov chain, corresponding to all the arcs between $P$ and $Q$.

In other words, the total reversed rate, given by Proposition 1, is distributed amongst the reversed arcs in proportion to the forward transition rates. We use this rule in our main result, Theorem 1 in Section 4.3.2, which reverses compound agents.

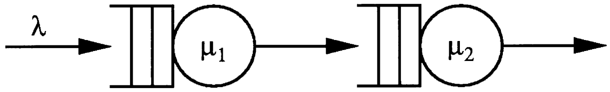


Fig. 2. A simple tandem queueing network with two nodes.

### 4.3. Compound agents

Under appropriate conditions, the reversed agent of a cooperation between two agents $P$ and $Q$ is a cooperation between the reversed agents of $P$ and $Q$, after some re-parameterisation. Before giving this result in its most general form, Theorem 1, we first preview our simplest example from Section 6 to provide some motivation.

#### 4.3.1. An illustration

Suppose we have two queues in tandem with Poisson arrivals to queue 1 at rate $\lambda$ (Fig. 2). On leaving queue 1, customers proceed immediately to queue 2 and on leaving queue 2 they depart the system. Customers are served one at a time without preemption in each queue and both queues' service times are exponential with parameters $\mu_1$ and $\mu_2$. The service rates may be state dependent, but we assume they are constant to simplify the notation.

This network (starting in the state with both queues empty) can be described by the PEPA expression $P_0 \underset{\{a\}}{\bowtie} Q_0$ where:

$$
\begin{aligned}
P_n &= (e, \lambda).P_{n+1} && (n \geqslant 0),\\
P_n &= (a, \mu_1).P_{n-1} && (n > 0),\\
Q_n &= (a, \top).Q_{n+1} && (n \geqslant 0),\\
Q_n &= (d, \mu_2).Q_{n-1} && (n > 0).
\end{aligned}
$$

Our aim is to find a reversed process of the form $X \underset{\{\bar{a}\}}{\bowtie} Y$ where $X$ and $Y$ are as close as we can get to the reversed processes of $P$ and $Q$, respectively. Now, the reversed process of $P$ is itself, but the rate of action type $\bar{a}$ is $\lambda$, which must be changed to $\top$ since it is now passive. Similarly, symbolically reversing $Q$ gives $Q$ again, but with unspecified arrival rate $\top$. Note that the rate of $\bar{a}$ is $\mu_2$, which is known to be correct for the synchronisation. All we need do is bind the correct value to the external arrivals' rate $\top$ in the reversed process; this is known to be $\lambda$, cf. [16]. A plausible argument that achieves this is the following. In the process $Q$, the passive action type $a$ is associated with the active action type $a$ in $P$. The rate $\top$ should therefore be bound to a rate associated with $a$ in $P$ before reversing $Q$. We choose its *reversed* rate, i.e. the rate of $\bar{a}$ in $\bar{P}$, namely $\lambda$, and then reverse $Q$. In this way, the action type $a$ is reversed twice, once in $P$ and once in $Q$. This is intuitively appealing since actions in cooperation sets participate in two components.

The situation becomes more complicated when there are more than one actions in a cooperation set, which are not all active in the same component. An example is a pair of queues, in which there are departures from each queue to the other in both directions. Then the rate $\lambda$ above is not a constant but a function of a passive rate (here corresponding to the internal arrivals from queue 2). The two passive rates must be determined simultaneously. The solution to this problem and the validity of the argument in general is established by Theorem 1 in the following section.

*4.3.2. The reversed compound agent theorem (RCAT)*

We are now in a position to state and prove the main result of this paper, the reversal of a compound PEPA agent under given conditions. First we define some new notation.

**Definition 3.** The subset of action types in a set $L$ which are *passive* with respect to a process $P$ (i.e. are of the form $(a,\top)$ in $P$) is denoted by $\mathscr{P}_P(L)$. The set of corresponding active action types is denoted $\mathscr{A}_P(L)=L\backslash\mathscr{P}_P(L)$.

Before reversing any agent, we first syntactically transform it so that every occurrence of a passive action $(a,\top)$ is relabeled $(a,\top_a)$. Consequently, we know that every passive action rate is uniquely identified with exactly one action type. Obviously this syntactic transformation is trivial to compile automatically. Henceforth, when referring to a reversed agent, we mean an agent that satisfies Kolmogorov's criteria: the agent will define a reversed process if and only if the original process was stationary. Stationarity issues are considered separately.

**Theorem 1** (Reversed compound agent). *Suppose that the cooperation $P \underset{L}{\bowtie} Q$ has a derivation graph with an irreducible subgraph $G$. Given that*

1. *every passive action type in $\mathscr{P}_P(L)$ or $\mathscr{P}_Q(L)$ is always enabled in $P$ or $Q$ respectively (i.e. enabled in all states of the transition graph);*
2. *every reversed action of an active action type in $\mathscr{A}_P(L)$ or $\mathscr{A}_Q(L)$ is always enabled in $\bar{P}$ or $\bar{Q}$, respectively;*
3. *every occurrence of a reversed action of an active action type in $\mathscr{A}_P(L)$ (respectively $\mathscr{A}_Q(L)$) has the same rate in $\bar{P}$ (respectively $\bar{Q}$).*

*the reversed agent $\overline{P \underset{L}{\bowtie} Q}$, with derivation graph containing the reversed subgraph $\bar{G}$, is*

$$\bar{R}\{(\bar{a},\overline{p_a}) \leftarrow (\bar{a},\top) \mid a \in \mathscr{A}_P(L)\} \underset{L}{\bowtie} \bar{S}\{(\bar{a},\overline{q_a}) \leftarrow (\bar{a},\top) \mid a \in \mathscr{A}_Q(L)\},$$

*where*

$$R = P\{\top_a \leftarrow x_a \mid a \in \mathscr{P}_P(L)\},$$

$$S = Q\{\top_a \leftarrow x_a \mid a \in \mathscr{P}_Q(L)\},$$

*$\{x_a\}$ are the solutions (for $\{\top_a\}$) of the equations*

$$\top_a = \overline{q_a}, \quad a \in \mathscr{P}_P(L),$$

$$\top_a = \overline{p_a}, \quad a \in \mathscr{P}_Q(L)$$

*and $\overline{p_a}$ (respectively $\overline{q_a}$) is the symbolic rate of action type $\bar{a}$ in $\bar{P}$ (respectively $\bar{Q}$).*

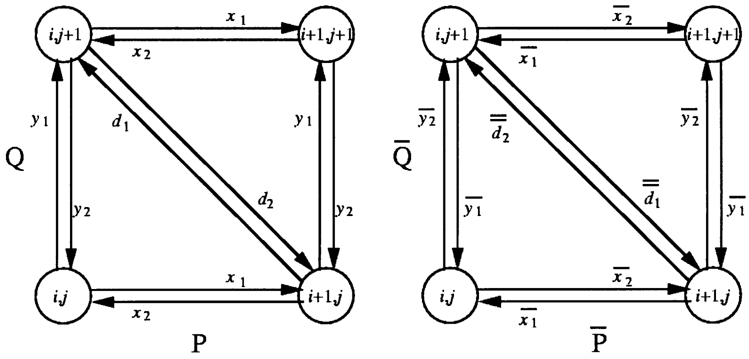


Fig. 3. Section of the state transition graphs for a cooperation.

**Remarks.** 1. The theorem essentially states that the reversed agent of a cooperation in which every passive action is always enabled is the cooperation of the reversed components with active synchronising actions made passive and vice-versa, with appropriate assignment of rates to the reversed actions thus made active.

2. In general, the reversed rates $\overline{p_a}$ (respectively $\overline{q_a}$) for active actions $a \in \mathscr{A}_P(L)$ (respectively $a \in \mathscr{A}_Q(L)$) will be functions of $\{\top_b \mid b \in \mathscr{P}_P(L)\}$ (respectively $\{\top_b \mid b \in \mathscr{P}_Q(L)\}$). They are then bound symbolically to the indeterminate rates $\top_a$ of the corresponding passive actions in the cooperation, i.e. in $Q$ (respectively $P$).

3. In $R$ and $S$, all actions have specified rates since the unspecified rates $\top_a$ are reassigned, in a well defined way by condition 3 of the theorem, by solving the specified equations.

4. We make the abbreviations

$$\bar{R}' = \bar{R}\{(\bar{a}, \overline{p_a}) \leftarrow (\bar{a}, \top) \mid a \in \mathscr{A}_P(L)\},$$

$$\bar{S}' = \bar{S}\{(\bar{a}, \overline{q_a}) \leftarrow (\bar{a}, \top) \mid a \in \mathscr{A}_Q(L)\}.$$

The proof of Theorem 1 consists of verifying that Kolmogorov's criteria hold. Conservation of outgoing rate is straightforward and uses conditions 1 and 2. The analysis of the cycles is considerably harder, however. Before giving the full proof in the next sub-section, we consider a special case of cycles, in order to clarify the theorem and to illustrate the proof technique. The simplest cycle with nodes involved in the cooperation set $L$ is a 'triangle' with the nodes $(i,j),(i+1,j),(i,j+1)$ at its vertices; see Fig. 3.

Suppose there exist pairs of one-step transitions with rates $(x_1,x_2)$ and $(y_1,y_2)$, as shown. The transitions on the diagonal, between nodes $(i+1,j)$ and $(i,j+1)$, are such that either

(a) $P$ is passive with respect to the transition with rate $d_2$ and $Q$ is passive with respect to the transition with rate $d_1$.

(b) $P$ is active and $Q$ is passive with respect to both $d_1$ and $d_2$.

Note that the vertical rates in the transition graph are identical in all columns, and similarly for the horizontal rates in the rows—and indeed the diagonal rates. This is because one process cannot influence another through the cooperation combinator except by synchronisation.

We first consider case (a) in which $Q$ is passive with respect to $d_1$ and $P$ is passive with respect to $d_2$. Here, there is a double transition in the active process $P$ from $(i+1,j)$ to $(i,j)$ with rates $x_2$ and $d_1$, and a double transition in the passive process $Q$ from $(i,j)$ to $(i,j+1)$ with rates $y_1$ and $\top$. Hence, by Definition 2, reversing in $R$ gives

$$\overline{d_1} = \left(\frac{d_1}{d_1 + x_2}\right)(\overline{d_1} + \overline{x_2}) \tag{2}$$

and similarly, considering the diagonal transition $d_2$ and reversing in $S$,

$$\overline{d_2} = \left(\frac{d_2}{d_2 + y_2}\right)(\overline{d_2} + \overline{y_2}). \tag{3}$$

The reversed rate of $\overline{d_1}$ in $S$, $\overline{\overline{d_1}}$ (in $\bar{S}$), is the claimed reversed rate of $d_1$ in the cooperation and satisfies

$$\overline{\overline{d_1}} = \left(\frac{\overline{d_1}}{\overline{d_1} + y_1}\right)(\overline{\overline{d_1}} + \overline{y_1}) \tag{4}$$

and similarly,

$$\overline{\overline{d_2}} = \left(\frac{\overline{d_2}}{\overline{d_2} + x_1}\right)(\overline{\overline{d_2}} + \overline{x_1}). \tag{5}$$

By definition of the reversed processes $\bar{R}$ and $\bar{S}$, we now have

$$(x_1 + \overline{d_2})(x_2 + d_1) = (\overline{x_1} + \overline{\overline{d_2}})(\overline{x_2} + \overline{d_1}), \tag{6}$$

$$(y_1 + \overline{d_1})(y_2 + d_2) = (\overline{y_1} + \overline{\overline{d_1}})(\overline{y_2} + \overline{d_2}). \tag{7}$$

We now need to verify the Kolmogorov criteria along each dimension, on the diagonal and round the triangle in both directions. For the diagonal:

$$\begin{aligned}
\overline{\overline{d_1}}\,\overline{\overline{d_2}} &= \frac{\overline{d_1}\,\overline{d_2}(\overline{\overline{d_1}} + \overline{y_1})(\overline{\overline{d_2}} + \overline{x_1})}{(\overline{d_1} + y_1)(\overline{d_2} + x_1)} \quad \text{by Eqs. (4) and (5)} \\
&= \frac{d_1 d_2(\overline{d_1} + \overline{x_2})(\overline{d_2} + \overline{y_2})(\overline{\overline{d_1}} + \overline{y_1})(\overline{\overline{d_2}} + \overline{x_1})}{(d_1 + x_2)(d_2 + y_2)(\overline{d_1} + y_1)(\overline{d_2} + x_1)} \quad \text{by (2) and (3)} \\
&= d_1 d_2 \quad \text{by (6) and (7)}
\end{aligned}$$

Using the complementary equations to (2)–(5), viz.

$$\overline{x_2} = \left(\frac{x_2}{d_1 + x_2}\right)(\overline{d_1} + \overline{x_2}),$$

$$\overline{y_2} = \left(\frac{y_2}{d_2 + y_2}\right)(\overline{d_2} + \overline{y_2}),$$

$$\overline{y_1} = \left(\frac{y_1}{\overline{d_1} + y_1}\right)(\overline{\overline{d_1}} + \overline{y_1}),$$

$$\overline{x_1} = \left(\frac{x_1}{\overline{d_2} + x_1}\right)(\overline{\overline{d_2}} + \overline{x_1})$$

it follows similarly that $\overline{x_1}\,\overline{x_2} = x_1 x_2$ and $\overline{y_1}\,\overline{y_2} = y_1 y_2$.

For the anticlockwise triangle, we have

$$\begin{aligned}\overline{\overline{d_1}}\,\overline{y_2}\,\overline{x_1} &= \frac{d_1 y_2 x_1(\overline{d_1} + \overline{x_2})(\overline{\overline{d_1}} + \overline{y_1})(\overline{d_2} + \overline{y_2})(\overline{\overline{d_2}} + \overline{x_1})}{(d_1 + x_2)(\overline{d_1} + y_1)(d_2 + y_2)(\overline{d_2} + x_1)} \\ &= d_1 y_2 x_1\end{aligned}$$

as required. The clockwise triangle is similar.

For case (b), with $P$ being active with respect to both $d_1$ and $d_2$, we have:

$$\overline{d_1} = \left(\frac{d_1}{d_1 + x_2}\right)(\overline{d_1} + \overline{x_2}),$$

$$\overline{d_2} = \left(\frac{d_2}{d_2 + x_1}\right)(\overline{d_2} + \overline{x_1}),$$

$$\overline{\overline{d_1}} = \left(\frac{\overline{d_1}}{\overline{d_1} + y_1}\right)(\overline{\overline{d_1}} + \overline{y_1}),$$

$$\overline{\overline{d_2}} = \left(\frac{\overline{d_2}}{\overline{d_2} + y_2}\right)(\overline{\overline{d_2}} + \overline{y_2}),$$

$$(x_1 + d_2)(x_2 + d_1) = (\overline{x_1} + \overline{d_2})(\overline{x_2} + \overline{d_1}),$$

$$(y_1 + \overline{d_1})(y_2 + \overline{d_2}) = (\overline{y_1} + \overline{\overline{d_1}})(\overline{y_2} + \overline{\overline{d_2}}).$$

The analysis now mirrors that of case (a).

#### 4.3.3. Proof of Theorem 1

Let the instantaneous transition rate in the Markov chain of $P$ (respectively $Q, R, S$) between states $i$ and $j$ be $p_{ij}$ (respectively $q_{ij}, r_{ij}, s_{ij}$) and let $p_i = \sum_{j \neq i} p_{ij}$ $(q_i, r_i, s_i$

similarly). To prove the theorem, we consider the conservation of outgoing rate at every node and the Kolmogorov criteria on the cycles. Every node in the derivation graph can be identified as $(i,j)\in G$[3] where $i,j$ are states in the chains corresponding to $P,Q$, respectively—although not necessarily every pair of states taken from $P$ and $Q$ (i.e. in the Cartesian product) will be valid in the cooperation.

1. *Outgoing rate*: In $P \underset{L}{\bowtie} Q$, the total rate out of any node $(i,j)\in G$ is $p_i\{\top\leftarrow 0\}+q_j\{\top\leftarrow 0\}$ where the relabelling $\{\top\leftarrow 0\}$ is an abbreviation for $\{\top_a\leftarrow 0\mid a\in L\}$; i.e. every occurrence of an unspecified rate is set to zero. Now,

$$r_i = p_i\{\top_a \leftarrow x_a \mid a \in \mathscr{P}_P(L)\},$$

$$s_j = q_j\{\top_a \leftarrow x_a \mid a \in \mathscr{P}_Q(L)\}$$

are the total rates out of states $i$ and $j$ in $R$ and $S$, respectively, and hence, by definition, in $\bar{R}$ and $\bar{S}$, respectively. By condition 1, we may write

$$r_i = p_i\{\top \leftarrow 0\} + \sum_{a\in\mathscr{P}^i_P(L)} x_a,$$

$$s_j = q_j\{\top \leftarrow 0\} + \sum_{a\in\mathscr{P}^j_Q(L)} x_a,$$

where $\mathscr{P}^i_P(L)$ denotes the set of types of the actions that are passive in $P$ and correspond to transitions *out of* state $i$ in the Markov chain of $P$. Hence, the total rate out of state $(i,j)$ in $\bar{R}' \underset{\bar{L}}{\bowtie} \bar{S}'$ is $p_i\{\top\leftarrow 0\}+q_j\{\top\leftarrow 0\}$ since the summands in the above expressions for $r_i$ and $s_j$ are precisely the rates corresponding to those actions that are made passive and so must be removed from the Markov graph—by condition 2 their actions are always enabled and so their transitions are present in the graph.

2. *Cycles*: To complete the proof, we establish the Kolmogorov criteria for all cycles in $G$. Any cycle originates from two separate cycles, one in each of $P$ and $Q$, corresponding to an empty cooperation set $L$. We call these cycles the *base cycles*. Each 'diagonal' arc, i.e. one that causes a change of state in both $P$ and $Q$, contributes to bace cycles in both $P$ and $Q$. For a cycle with arcs $a_1,\dots,a_n$ in $P \underset{L}{\bowtie} Q$, let the total rates in the (horizontal) base $P$-cycle be $h_1,\dots,h_n$ and in the (vertical) base $Q$-cycle be $v_1,\dots,v_n$. Arcs corresponding to passive actions carry zero rate, so any horizontal (say) arc $i$ comprising only passive actions carries total rate $h_i=0$ in the base $P$-cycle. Suppose the $i$th arc (arc $i$) in a base cycle participates as an active action in $c_i\geqslant 0$ cooperations in $L$ and let the rates on the corresponding arcs be $\psi_{ii'}h_i$ if the active arc is horizontal or $\psi_{ii'}v_i$ if it is vertical, where $i'$ is the passive arc corresponding to $i$ in the vertical or horizontal cycle, respectively, $1\leqslant i'\leqslant c_i$, $0\leqslant\psi_{ij}\leqslant 1, \sum_{j=1}^{c_i}\psi_{ij}\leqslant 1$. Define $\psi_{i0}=1-\sum_{j=1}^{c_i}\psi_{ij}$ to be the fraction of the total rate that does not participate in the cooperation.

Consider a cycle of arcs $C=a_1,\dots,a_n$ in $P \underset{L}{\bowtie} Q$ with base $P$-cycle $H=N_H\cup A_H\cup P_H$ and base $Q$-cycle $V=N_V\cup A_V\cup P_V$, where $N_H,N_V$ are arcs of actions that are not

[3] Strictly speaking, we are abusing notation somewhat: $(i,j)$ is a state in the underlying Markov process that has state space isomorphic to the nodes in $G$.

in the cooperation set $L$, $A_H, A_V$ are arcs of active actions in $L$ and $P_H, P_V$ are arcs of passive actions in $L$. Thus, the arcs in $A_H$ and $P_V$ have the same types and similarly for $A_V$ and $P_H$.

The product of the transition rates round cycle $C$ is therefore

$$\prod_{i\in N_H} \psi_{i0} h_i \prod_{i\in A_H} \psi_{ii'} h_i \prod_{i\in N_V} \psi_{i0} v_i \prod_{i\in A_V} \psi_{ii'} v_i,$$

where $i'$ is the arc containing the passive action corresponding to the active action of $i$.

Now, in $R$, arc $i \in H$ has rate $h_i^*$ which is the sum of its original rate in $P$ (with passive actions assigned rate 0) and the reversed rates of the active actions in $S$ that are bound to the passive actions of $i$ in $P$. Similarly, in $S$, arc $i \in V$ has rate $v_i^*$ and we have, using Definition 2,

$$h_i^* = h_i + \sum_{j\in Ac_i} \frac{\psi_{ji} v_j}{v_j^*}\overline{v_j^*},$$

$$v_i^* = v_i + \sum_{j\in Ac_i} \frac{\psi_{ji} h_j}{h_j^*}\overline{h_j^*},$$

where $Ac_i$ is the set of arcs having an active action type in the cooperation set $L$ the same as a passive action in arc $i$—i.e. actively cooperating with $i$. Hence, the claimed product of the reversed rates round the cycle $C$, is, using Definition 2 again,

$$\prod_{i\in N_H} \frac{\psi_{i0} h_i}{h_i^*}\overline{h_i^*} \prod_{i'\in P_H} \frac{\psi_{ii'} v_i \overline{v_i^*}}{v_i^* h_{i'}^*}\overline{h_{i'}^*} \prod_{i\in N_V} \frac{\psi_{i0} v_i}{v_i^*}\overline{v_i^*} \prod_{i'\in P_V} \frac{\psi_{ii'} h_i \overline{h_i^*}}{h_i^* v_{i'}^*}\overline{v_{i'}^*}.$$

Because of the 1–1 correspondence between arcs in $A_H$ and $P_V$ and between arcs in $A_V$ and $P_H$, this may be written

$$\prod_{i\in N_H} \frac{\psi_{i0} h_i \overline{h_i^*}}{h_i^*} \prod_{i\in A_H} \frac{\psi_{ii'} h_i \overline{h_i^*}}{h_i^*} \prod_{i\in P_H} \frac{\overline{h_i^*}}{h_i^*} \prod_{i\in N_V} \frac{\psi_{i0} v_i \overline{v_i^*}}{v_i^*} \prod_{i\in A_V} \frac{\psi_{ii'} v_i \overline{v_i^*}}{v_i^*} \prod_{i\in P_V} \frac{\overline{v_i^*}}{v_i^*}$$
$$= \prod_{i\in N_H} \psi_{i0} h_i \prod_{i\in A_H} \psi_{ii'} h_i \prod_{i\in N_V} \psi_{i0} v_i \prod_{i\in A_V} \psi_{ii'} v_i$$

by the Kolmogorov criteria applied to the cycles $H$ in $R$ and $V$ in $S$. This is equal to the product of the rates around the forward cycle $C$, so completing the proof. □

*4.3.4. Conditions of the RCAT*

The three conditions of the RCAT, Theorem 1, are not difficult to check since they relate to each of the cooperating agents separately. It is assumed that these agents have already been fully analysed, either directly or through recursive applications of the RCAT. For condition 1, it is a straightforward syntactic check to see if the passive actions are enabled in every state of the two agents $P$ and $Q$. Recalling from Section 3.3 that arrow-inversion is simple, it is just as easy to check syntactically if the reversed active actions are always enabled: condition 2.

Condition 3 involves a little more work, being essentially a semantic constraint. However, the reversed processes $\bar{R}$ and $\bar{S}$ are known and hence every rate. If these reversed processes are known in parameterised form, as in the case of the $M/M/1$ queue, for example, it is straightforward to check if the reversed rates associated with the same action type are constant—again, essentially syntactically. The same will apply in a recursive analysis, for example in networks of queues, cf. Section 6.4. In the worst case, if a symbolic calculation is not possible, the rate on every reversed active arrow in $\bar{R}$ and $\bar{S}$ can, in principle, be checked numerically. However, this could become prohibitively expensive, as state spaces grow in a recursive analysis, and would require truncation to approximate infinite state spaces.

In general, a state space exploration of the cooperation's derivation graph must be done anyway to find the irreducible subchains, which the theorem assumes to exist. We assume here a finite state space, possibly the result of truncation. During the exploration, the states can be enumerated and non-normalised equilibrium probabilities allocated using Proposition 1, possibly on a separate pass. Although potentially costly, note that this approach avoids the need to solve the balance equations explicitly, which is by far the most computationally expensive part and main source of inaccuracy in direct solution methods. During the state space exploration, the normalising constant can also be accumulated. Note that this would have to be done in any analysis: 'clever' normalising constant algorithms need special mathematical structure in the state space and in the product-form solution. Such structures are not the subject of this paper.

Finally, note that a separate check for ergodicity is unnecessary: if (and only if) a normalising constant can be found, the chain is ergodic, being irreducible, by the steady-state theorem for Markov chains.

#### 4.3.5. *Invisible actions*

When conditions 1 and 2 of the RCAT do not hold, they can sometimes be secured by adding 'invisible actions' to the cooperating agents $P$ and $Q$. An invisible action $i$ does not cause an agent $A$ to change and may be specified as $A = i.A$. It represents a transition with arbitrary rate from a state to itself in the underlying Markov process. The reversed transition is identical since the total outgoing rate from the state will be increased by the same amount in the forward and reversed processes and the product of rates in the single arc cycle is also the same in both.

It is easy to verify that the RCAT still holds with invisible actions, and their introduction can ensure that the action types referred to in conditions 1 and 2 are indeed always enabled. Obviously, this will change the cooperation concerned: when an invisible passive action in one agent synchronises, the active action in the other agent is allowed to proceed independently, whereas otherwise it would be blocked. Invisible passive actions can be introduced anywhere but an invisible active (cooperating) action must be chosen with care. To maintain condition 3, the rate of a new invisible action of type $a$ must be equal to the (common) rate of the *reversed actions* of the other active actions of type $a \in L$. Then its own reversed rate will be that common rate.

Consider, for example, the cooperation $P_1 \underset{\{a\}}{\bowtie} Q_1$ where $P_1 = (b, \lambda).P_2$, $P_2 = (a, \top).P_1$; $Q_1 = (c, v).Q_2$, $Q_2 = (a, \mu).Q_1$. Condition 1 of the RCAT does not hold

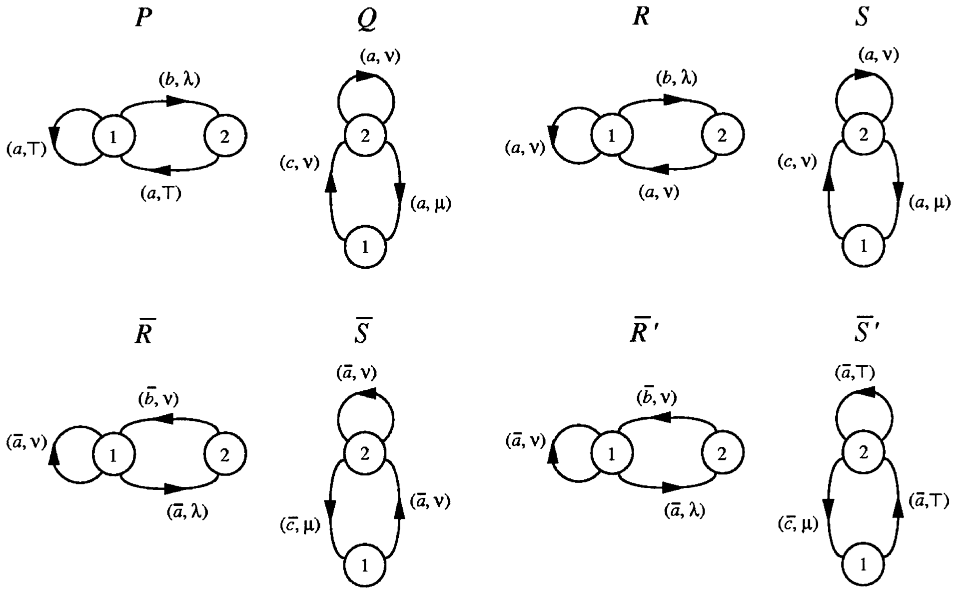


Fig. 4. The $P, Q, R, S, \bar{R}, \bar{S}, \bar{R}', \bar{S}'$ processses as used in the RCAT.

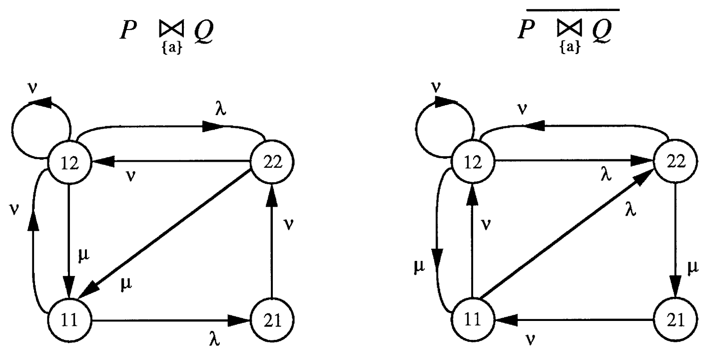


Fig. 5. The original cooperation and its reversed process.

because $a$ is not enabled in $P_1$, blocking $(a, \mu)$. Condition 2 does not hold because the reversed action of $a$ is not enabled in $Q_2$. Suppose we now add the assignments $P_1 = (a, \top).P_1$ and $Q_2 = (a, v).Q_2$, $v$ being the rate of the reversed action of $(a, \mu)$. The conditions of the RCAT now hold and applying it gives the transition graphs shown in Fig. 4.

The reversed cooperation $\overline{P_1 \underset{\{a\}}{\bowtie} Q_1}$ then follows as $\bar{R}' \underset{\{\bar{a}\}}{\bowtie} \bar{S}'$ and its transition graph is shown with that of $P_1 \underset{\{a\}}{\bowtie} Q_1$ in Fig. 5.

It is straightforward to verify that these graphs satisfy the Kolmogorov criteria. Moreover, they are irreducible and finite, and so are ergodic. Thus, the agent constructed for $\overline{P_1 \underset{\{a\}}{\bowtie} Q_1}$ by the RCAT does define the reversed process of $P_1 \underset{\{a\}}{\bowtie} Q_1$.

### 4.4. A weaker RCAT

For cooperations that do not satisfy condition 3 of the theorem, we only need to verify Kolmogorov's criteria for those cycles that involve shared actions. Hence we have the very much weaker theorem:

**Theorem 2** (Weak Reversed Compound Agent). *If conditions* 1 *and* 2 *of Theorem* 1 *are satisfied, the reversed agent* $\overline{P \underset{L}{\bowtie} Q}$ *of the agent* $P \underset{L}{\bowtie} Q$ *is*

$$\bar{R}\{(\bar{a}, \overline{p_a}) \leftarrow (\bar{a}, \top) \mid a \in \mathscr{A}_P(L)\} \underset{L}{\bowtie} \bar{S}\{(\bar{a}, \overline{q_a}) \leftarrow (\bar{a}, \top) \mid a \in \mathscr{A}_Q(L)\},$$

*where*

$$R = P\{\top_a \leftarrow x_a \mid a \in \mathscr{P}_P(L)\},$$

$$S = Q\{\top_a \leftarrow x_a \mid a \in \mathscr{P}_Q(L)\},$$

*provided there exist a set of positive real numbers* $\{x_a\}$ *such that the Kolmogorov criteria hold in all cycles involving nodes connected by a shared action.*

**Proof.** Since conditions 1 and 2 are still satisfied, the total outgoing rate is the same at all nodes in both the forward and reversed processes by the same arguments used in the proof of Theorem 1; condition 3 is not needed. Regarding cycles involving shared actions, the second of Kolmogorov's criteria is satisfied by hypothesis. We therefore only have to consider the remaining cycles.

In the Markov chain of $P \underset{L}{\bowtie} Q$, let $A$ be the set of nodes not included in the synchronisation, i.e. $n \in A$ iff there does not exist an arc $\alpha$ out of $n$ nor into $n$ such that $a \in L$, where $a$ is the type of the action represented by $\alpha$. Now, all nodes in a cycle entirely within $A$ must be of the form either $(i_0, j)$ or $(i, j_0)$ where $i_0, j_0$ are fixed. Note that the rates $p_{ii'}$ in $P$ are independent of $j_0$ and the rates $q_{jj'}$ in $Q$ are independent of $i_0$. Hence the cycle must be isomorphic to a cycle in either $P$ or $Q$—without loss of generality, suppose $P$. Since they do not contain any nodes in $L$, the rates of the transitions in this cycle, and hence their product, are the same in $R$ as in $P$. Therefore, the product of the reversed rates in $\bar{R}$ is the same, by Kolmogorov's criteria, and hence also in $\bar{R}' \underset{\bar{L}}{\bowtie} \bar{S}'$. $\square$

Although more general, it is not easy to see where the weak RCAT could be applied! The problem is that it is not sufficient to solve the equations for the rates $x_a$ relating to cycles involving shared actions. These will depend on other reversed rates which must also be found. Consequently, this route would lead to an explicit derivation of the reversed process, as we described for simple agents. Some heuristic is necessary to guess, under suitable conditions, the $x_a$, which can then be verified, bringing us almost full circle to the RCAT itself.

## 5. Simple applications

In this section, we show how the methodology described for simple agents in Section 4.1 can be applied to examples of increasing complexity. We begin with the classical birth–death process, the $M/M/1$ queue, which is known to be reversible, extend this to a slightly more complex queue which is not reversible and conclude with a model of an input–output buffer with more general cycles.

### 5.1. M/M/1 queue and reversible processes

The $M/M/1$ queue is a very simple birth–death process used to illustrate many concepts in stochastic modelling. The state of the system is the number of customers in the queue (including any in service), there is one server that has mean service time $1/\mu_n$ (service rate $\mu_n$) and arrivals are Poisson with rate $\lambda_n$ in state $n \geqslant 0$. A PEPA agent that describes this queue is defined as follows:

$$\begin{aligned} Q_0 &= (a_0, \lambda_0).Q_1, \\ Q_n &= (a_n, \lambda_n).Q_{n+1} \qquad (n > 0), \\ Q_n &= (d_n, \mu_n).Q_{n-1} \qquad (n > 0). \end{aligned}$$

Note that the first equation can be omitted if we say $n \geqslant 0$ in the second. This agent has the derivation graph shown in Fig. 6—isomorphic to the state transition graph of the underlying Markov process.

There are intuitive arguments that show that this process is reversible, i.e. its reverse process is the same as the original process describing the queue. Thus, for example, departures in one process are equivalent to arrivals in the other which are known to be Poisson. Moreover, the state at any time is independent of the departure process prior to that time. This result, the renowned 'Burke's theorem', follows since the arrival process after a given time is independent of the state existing at that time.

However, we can apply the methodology described in the previous section to obtain the reversed process mechanically. First, we can write down a PEPA definition of an agent with reversed arrows using the procedure of Section 3.3:

$$\begin{aligned} \overline{Q_{n+1}} &= (\overline{a_n}, \overline{\lambda_n}).\overline{Q_n} \qquad (n \geqslant 0), \\ \overline{Q_{n-1}} &= (\overline{d_n}, \overline{\mu_n}).\overline{Q_n} \qquad (n > 0). \end{aligned}$$

Relabelling the subscripts, we obtain:

$$\begin{aligned} \overline{Q_n} &= (\overline{a_{n-1}}, \overline{\lambda_{n-1}}).\overline{Q_{n-1}} \qquad (n \geqslant 1), \\ \overline{Q_n} &= (\overline{d_{n+1}}, \overline{\mu_{n+1}}).\overline{Q_{n+1}} \qquad (n > -1). \end{aligned}$$

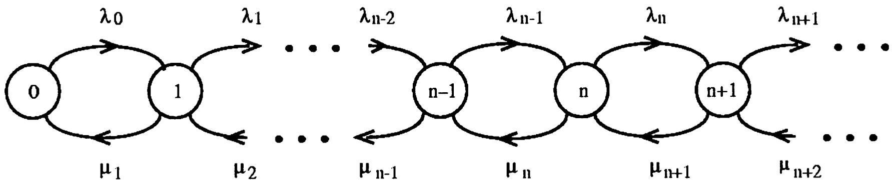


Fig. 6. State transition graph for a simple $M/M/1$ queue.

This has exactly the same structure as the agent describing the original queue, but we need to find the rates $\overline{\lambda_{n-1}}, \overline{\mu_{n+1}}$. We can do this using the method of Section 4.1 as follows, referring to Fig. 6.

1. There is one state, 0, with only one incoming arc and one outgoing arc, carrying rate $\lambda_0$. Hence, the transition rate from state 0 to state 1 in the reversed process is $\overline{\mu_1} = \lambda_0$.
2. The minimal cycles are the 2-cycles between adjacent states. That is, the single, generic, parameterised minimal cycle is $n \to (n+1) \to n$ for $n \geqslant 0$.

The conservation of total outgoing rate, together with Kolmogorov's criteria, then yield:

$$\overline{\mu_{n+1}} + \overline{\lambda_{n-1}} = \lambda_n + \mu_n \quad (n > 0),$$

$$\overline{\mu_{n+1}}\,\overline{\lambda_n} = \lambda_n \mu_{n+1} \quad (n \geqslant 0).$$

Now, for $n = 0$, we have $\overline{\mu_1} = \lambda_0$ already and so

$$\lambda_0 \overline{\lambda_0} = \lambda_0 \mu_1$$

so that $\overline{\lambda_0} = \mu_1$. The conservation of outgoing rate for $n = 1$ then gives $\overline{\mu_2} = \lambda_1$. Continuing this procedure for states $n = 1, 2, \ldots$ yields successively $\overline{\lambda_n} = \mu_{n+1}$ and $\overline{\mu_{n+2}} = \lambda_{n+1}$, completely determining the reversed process as identical to the original, as required.

### 5.2. An 'alternating server' $M/M/2$ queue

Now let us consider a non-reversible process, but only a small variation on the well-understood $M/M/1$ queue. This is a Markovian queue with two constant-rate servers that work together when there is enough work (queue length $n \geqslant 2$) but such that a customer arriving during an idle period (queue length 0) is served by the server that has been idle the longer (i.e. did not serve the most recent departure).

This system can be described by the following PEPA agent (isomorphic to the definition in terms of state transitions).

$$Q_{0A} = (a_{0A}, \lambda).Q_{1B},$$

$$Q_{0B} = (a_{0B}, \lambda).Q_{1A},$$

$$Q_{1A} = (a_{1A}, \lambda).Q_2,$$

$$Q_{1A} = (d_{1A}, \mu_1).Q_{0A},$$

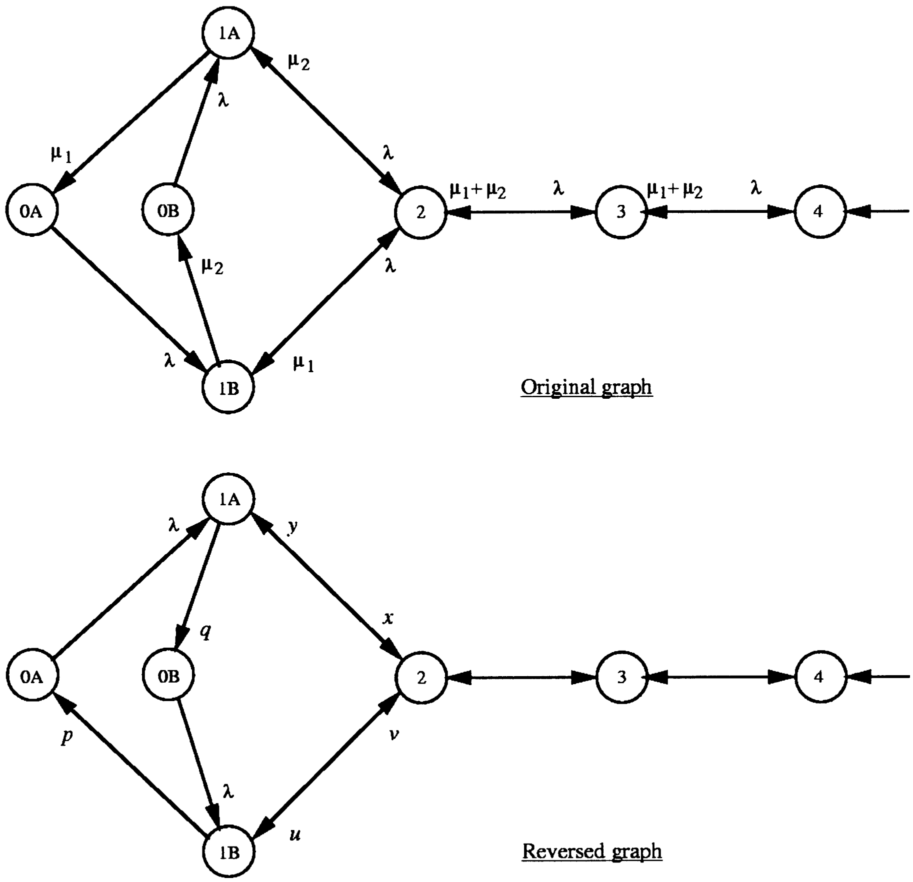


Fig. 7. State transition graphs for the alternating queue.

$$Q_{1B} = (a_{1B}, \lambda).Q_2,$$

$$Q_{1B} = (d_{1B}, \mu_2).Q_{0B},$$

$$Q_2 = (d_{21A}, \mu_2).Q_{1A},$$

$$Q_2 = (d_{21B}, \mu_1).Q_{1B},$$

$$Q_n = (a_n, \lambda).Q_{n+1} \qquad (n \geqslant 2),$$

$$Q_n = (d_n, \mu_1 + \mu_2).Q_{n-1} \quad (n \geqslant 3).$$

The state transition graph for this agent is shown in Fig. 7, along with a template (without rates) for the reversed graph. Clearly, we could mechanically construct a reversed-arrow agent, as we did for the $M/M/1$ queue, but this offers no more understanding. Instead, we apply the methodology for determining the reversed rates.

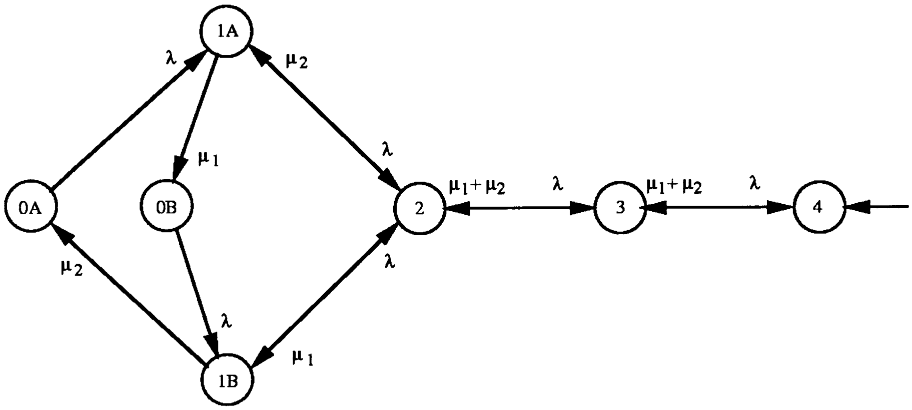


Fig. 8. Reversed graph with rates for the alternating queue.

By conservation of outgoing rate, we immediately know that the arcs $0A \to 1A$ and $0B \to 1B$ carry rate $\lambda$ in the reversed graph.

To calculate the six rates $p, q, x, y, u, v$, we can use seven equations arising from the conservation of outgoing rate at nodes 1A and 1B together with Kolmogorov's criteria on the five cycles $0A \to 1B \to 0B \to 1A \to 0A$, $0B \to 1A \to 2 \to 1B \to 0B$, $0A \to 1B \to 2 \to 1A \to 0A$, $1A \to 2 \to 1A$ and $1B \to 2 \to 1B$. This yields:

$$x + q = \lambda + \mu_1,$$

$$v + p = \lambda + \mu_2,$$

$$qp = \mu_1 \mu_2,$$

$$qvy = \lambda \mu_1 \mu_2,$$

$$xup = \mu_1 \mu_2 \lambda,$$

$$xy = \lambda \mu_2,$$

$$uv = \lambda \mu_1.$$

Any six of these equations are easy to solve symbolically—either by hand or computer—to give the solution (satisfied by them all) $p = \mu_2$, $q = \mu_1$, $x = \lambda$, $y = \mu_2$, $u = \mu_1$, $v = \lambda$. The remaining rates can be determined exactly as for the $M/M/1$ queue and the resulting reversed graph is shown in Fig. 8.

This graph can be obtained by making intuitive, educated 'guesses', considering that in the reversed process, some transitions are caused by arrivals with the same rate and some by service completions at the same servers (with same rates) as in the original process. However, here we have demonstrated a methodology that can be automated without intuition to provide reversed processes and thence product-form solutions.

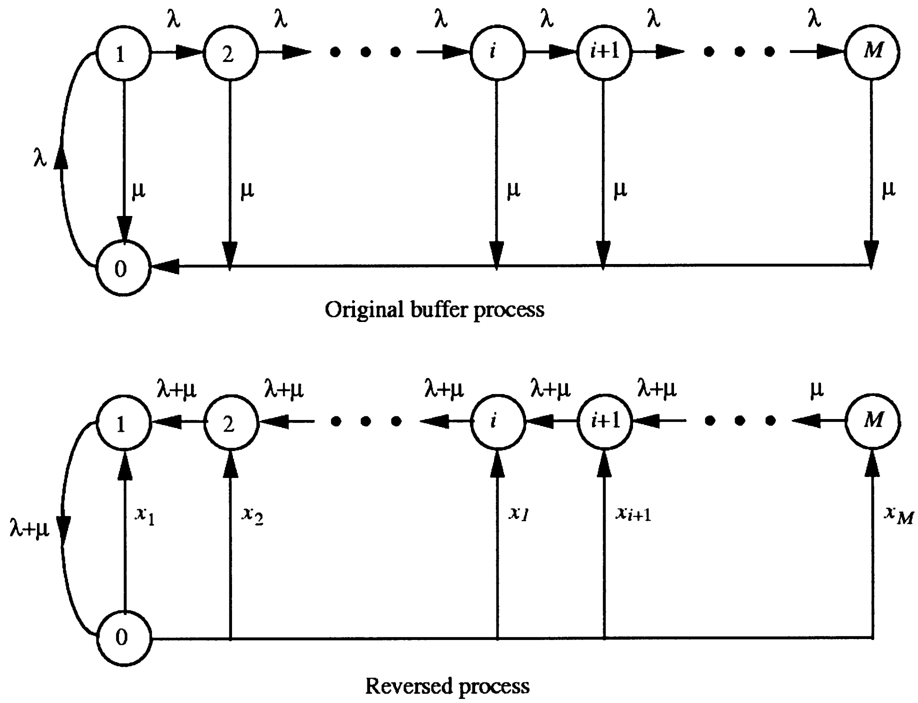


Fig. 9. State transition graphs for the buffer.

### *5.3. A buffer*

Next we consider a simple buffer in which messages are added according to a Poisson process with rate $\lambda$ and which is cleared at exponentially spaced instants, independently of the arrival process. The mean time between successive clearances is $\mu^{-1}$ and the buffer has capacity $M$; when full, arrivals are lost.

This buffer clearly follows a Markov process and can be specified in PEPA as:

$$B_n = (a_n, \lambda).B_{n+1} \qquad (0 \leqslant n \leqslant M-1),$$
$$B_n = (c_n, \mu).B_0 \qquad (1 \leqslant n \leqslant M).$$

The state transition graph for this agent is shown in Fig. 9, along with a template (without all rates) for the reversed graph. The process is clearly not reversible, since there are many uni-directional arcs, and it contains many non-trivial (minimal) cycles. By conservation of outgoing rate, we immediately know that the arcs $n \to n-1$ carry rate $\lambda + \mu$ in the reversed graph, for $n = 1, \dots, M-1$ and the arc $M \to M-1$ carries rate $\mu$. It remains to calculate the rates $x_n$ from node 0 to node $n$ $(1 \leqslant n \leqslant M)$. Applying Kolmogorov's criteria to the cycle $0 \to 1 \to 0$, we must have $x_1(\lambda + \mu) = \lambda\mu$

so that

$$x_1 = \mu\left(\frac{\lambda}{\lambda+\mu}\right).$$

Considering the cycles $0 \to 1 \to 2 \to \cdots \to n \to 0$ for $n = 2, 3, \ldots, M-1$ in the original graph, we obtain in exactly the same way $x_n(\lambda+\mu)^n = \lambda^n \mu$ so that

$$x_n = \mu\left(\frac{\lambda}{\lambda+\mu}\right)^n.$$

Finally, for cycle $0 \to 1 \to \cdots \to M \to 0$ we get

$$x_M = \lambda\left(\frac{\lambda}{\lambda+\mu}\right)^{M-1}.$$

It is simple to check that the total rate out of state 0 in the reversed graph is then $\lambda$ as required, and that the equilibrium probability for state $n$ is given by

$$\pi_n = \left(\frac{\lambda}{\lambda+\mu}\right)^n \pi_0 \quad (0 \leqslant n \leqslant M-1).$$

and

$$\pi_M = \frac{\lambda}{\mu}\left(\frac{\lambda}{\lambda+\mu}\right)^{M-1} \pi_0.$$

## 6. Compound applications

We now consider more complex examples which contain cooperation combinators in their PEPA specifications. The Reversed Compound Agent Theorem enables arbitrarily large networks to be analysed, provided they are constructed hierarchically in terms of the cooperation combinator and the theorem's conditions hold. We first apply the RCAT to the simple tandem queueing network introduced in Section 4.3.1, generalising to arbitrary interconnections between the two queues in Section 6.3 (after considering multiple input and output streams). This then naturally extends to arbitrary Markovian queueing networks. However, any composition of subnetworks with known reversed processes can be handled thus and we consider G-nets in Section 6.5.

### 6.1. Tandem queues

Consider again the two-queue tandem network of Section 4.3.1, with Poisson arrivals to queue 1, customers proceeding immediately to queue 2 after service at queue 1 and departures from the system after service at queue 2. The state space of the underlying Markov chain is $\{(i,j) \mid i,j \geqslant 0\}$ where $i$ and $j$ denote the numbers of customers at queues 1 and 2, respectively. Given external arrival rate $\lambda$ at queue 1 and service rates $\mu_i$ at queue $i$ $(i = 1,2)$, its state transition graph is as shown in Fig. 10.

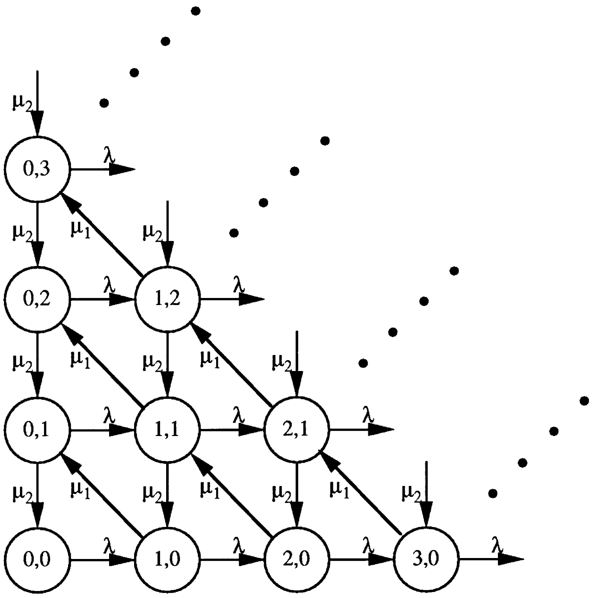


Fig. 10. State transition graph for the simple tandem-2 network.

The conditions of the RCAT are satisfied since both the passive action in $Q$ (*re* condition 1) and the reversed active action in $\bar{R}$ (*re* condition 2) are arrivals, which are always enabled in the $M/M/1$ queue. Regarding condition 3, the reversed rate of the active action in $R$ (the same as $P$ here) is constant, equal to the arrival rate $\lambda$ for all states of the first queue. Applying the RCAT, we need to solve the trivial single equation

$$\top_a = \lambda$$

giving the reversed PEPA agent: $X_0 \underset{\{\bar{a}\}}{\bowtie} Y_0$ where:

$$\begin{aligned} X_n &= (\bar{a}, \top).X_{n+1} \qquad (n \geqslant 0), \\ X_n &= (\bar{e}, \mu_1).X_{n-1} \qquad (n > 0), \\ Y_n &= (\overline{d}, \lambda).Y_{n+1} \qquad (n \geqslant 0), \\ Y_n &= (\bar{a}, \mu_2).Y_{n-1} \qquad (n > 0). \end{aligned}$$

Hence, we obtain the reversed graph shown in Fig. 11. Note that if the arrival rate to queue 1 had been dependent on the state (length) of queue 1, condition 3 of the RCAT would not hold since the reversed rate of the departure action (i.e. the arrival rate of the forward queue) would not be the same for all states of queue 1. This is consistent with the well-known restriction in product-form queueing networks that external arrival rates cannot be dependent on the state of the network or even the local queue arrived at.

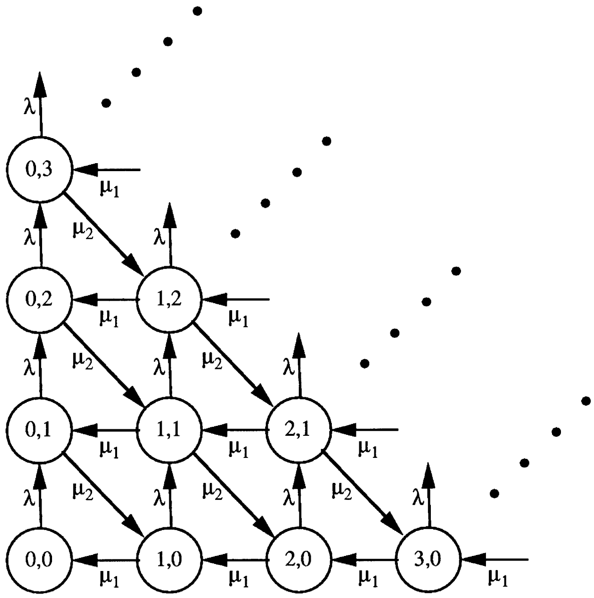


Fig. 11. Reversed graph for the simple tandem-2 network.

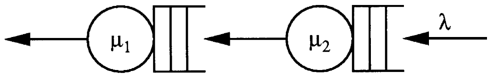


Fig. 12. The reversed tandem-2 network.

The reversed graph is readily seen to describe the tandem network comprising the same two queues, but with arrivals to queue 2 at rate $\lambda$, customer transitions from queue 2 to queue 1 on completion of service at queue 2 and departures from the network on completion of service at queue 1; see Fig. 12. This is quite consistent with intuition when one views the system 'running backwards in time'. However, the intuition is not so clear when we have arrivals and departures at both queues and arbitrary routing. We consider this more general problem in Section 6.3.

Note that, when an active and a passive agent cooperate in PEPA, it matters not which is the passive one—only the shared rate needs to be specified. Thus, the above tandem queues also have the following PEPA specification:

$$P_0 \underset{\{a\}}{\bowtie} Q_0,$$

where

$$P_n = (e, \lambda).P_{n+1} \qquad (n \geqslant 0),$$

$$P_n = (a, \top).P_{n-1} \qquad (n > 0),$$

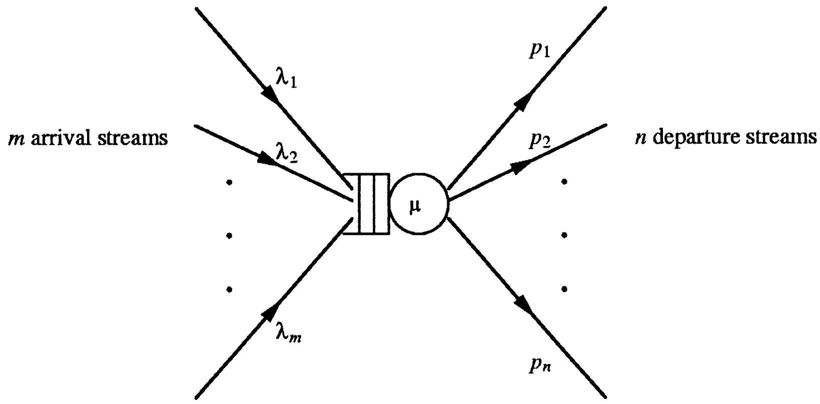


Fig. 13. Multiple input and output $M/M/1$ queue.

$$Q_n = (a, \mu_1).Q_{n+1} \qquad (n \geqslant 0),$$
$$Q_n = (d, \mu_2).Q_{n-1} \qquad (n > 0).$$

Here, the arrival process to the second queue has to wait if the first queue is empty, and has rate $\mu_1$ when it is non-empty. Although an equivalent specification, the passive cooperating action—departures from queue 1—is *not* always enabled. Hence condition 1 of Theorem 1 is not satisfied. Indeed, applying the theorem would lead to an erroneous reversed process; actually only because the outgoing rates are not preserved along the axes.

### 6.2. Multiple streams and channels

In order to build a network successively from its constituent queues—adding one queue at a time—we need a building block that consists of a queue with multiple arrival streams together with a server that has multiple output channels. Each arrival stream has its own rate and each output channel is selected with a fixed probability. This queue is depicted in Fig. 13.

Now, the length of this queue must have the same steady-state behaviour as that of a simple $M/M/1$ queue with the same service rate and arrival rate equal to the sum of all the rates of the arrival streams, viz. $\lambda = \lambda_1 + \cdots + \lambda_m$. The reversed process is given by Definition 2 and splits the total arrival rate $\lambda$ in proportion to the output channel selection probabilities. Similarly, the outputs of the reversed queue are selected with probabilities proportional to the arrival rates of the original queue. The reversed queue is shown in Fig. 14.

We now use this construction first in a tandem network with general routing and then in an arbitrary network of queues.

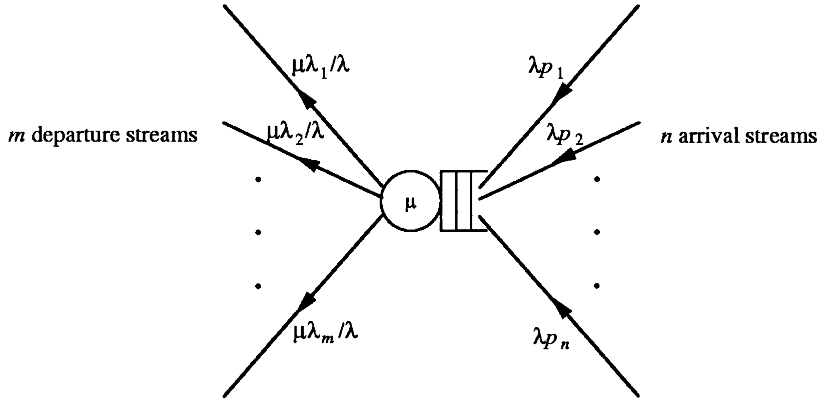


Fig. 14. Reversed multi-$M/M/1$ queue.

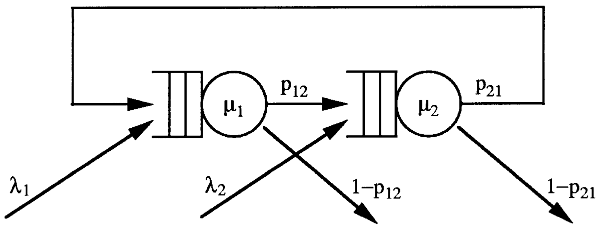


Fig. 15. General tandem-2 network.

### 6.3. Generally connected pair of queues

Consider two queues, 1 and 2, with respective external arrival rates $\lambda_1$ and $\lambda_2$, service rates $\mu_1$ and $\mu_2$ and routing probabilities $p_{12}$ from node 1 to node 2, $p_{21}$ from node 2 to node 1. Tasks leave the network from nodes 1 and 2 with probabilities $1 - p_{12}$ and $1 - p_{21}$, respectively. The network is shown in Fig. 15.

This network (starting in the state with both queues empty) can be described by the PEPA expression $P_0 \underset{\{a_1,a_2\}}{\bowtie} Q_0$ where:

$$\begin{aligned}
P_n &= (e_1, \lambda_1).P_{n+1} && (n \geqslant 0),\\
P_n &= (a_1, \top_1).P_{n+1} && (n \geqslant 0),\\
P_n &= (d_1, (1 - p_{12})\mu_1).P_{n-1} && (n > 0),\\
P_n &= (a_2, p_{12}\mu_1).P_{n-1} && (n > 0),\\
Q_n &= (e_2, \lambda_2).Q_{n+1} && (n \geqslant 0),\\
Q_n &= (a_2, \top_2).Q_{n+1} && (n \geqslant 0),
\end{aligned}$$

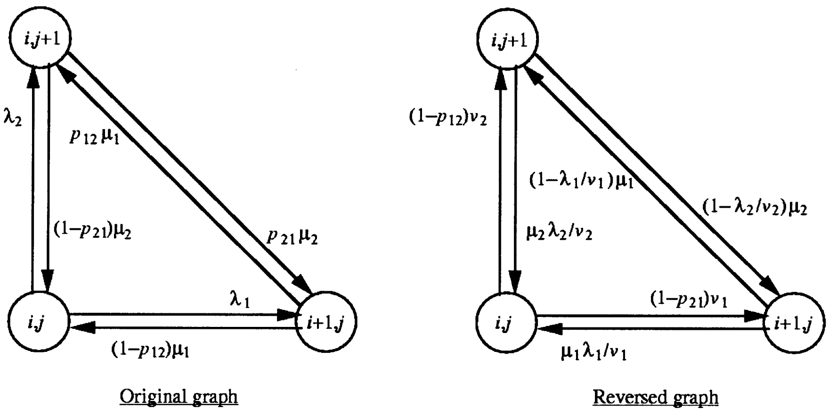


Fig. 16. Section of the general tandem network's state transition graphs.

$$Q_n = (d_2, (1 - p_{21})\mu_2).Q_{n-1} \qquad (n > 0),$$
$$Q_n = (a_1, p_{21}\mu_2).Q_{n-1} \qquad (n > 0).$$

This has state transition graph (fragment), in the neighbourhood of the state representing queue lengths $i, j$ at nodes $1, 2$, respectively, shown on the left of Fig. 16.

The first two conditions of the RCAT are satisfied because

1. the passive actions in $P$ and $Q$ are arrivals that are always enabled;
2. the passive actions in the reversed processes $\bar{R}$ and $\bar{S}$ (reversed actions of the respective active actions in $R$ and $S$) are also arrivals and always enabled.

Applying the RCAT, assuming for now that condition 3 holds, we solve the pair of equations:

$$\top_1 = (\lambda_2 + \top_2)p_{21},$$
$$\top_2 = (\lambda_1 + \top_1)p_{12}.$$

These are precisely the traffic equations for the internal flows. Thus, if the total average traffic entering (and leaving) node $i$ in steady-state in unit time is denoted $v_i$ $(i = 1, 2)$, also called the *visitation rate*, the solution of the equations is

$$\top_i = v_i - \lambda_i$$

i.e.

$$\top_1 = \frac{(\lambda_2 + \lambda_1 p_{12})p_{21}}{1 - p_{12}p_{21}}$$

or

$$v_1 = \frac{\lambda_1 + \lambda_2 p_{21}}{1 - p_{12}p_{21}}$$

and $\top_2, v_2$ similarly.

Because these equations have a solution, we see that condition 3 is indeed satisfied because the reversed rates of the active actions in $R$ and $S$ are constant fractions of the constant visitation rates, by Definition 2.

This gives the reversed PEPA agent: $X_0 \underset{\{\overline{a_1},\overline{a_2}\}}{\bowtie} Y_0$ where:

$$
\begin{aligned}
X_n &= \left(\overline{e_1}, \frac{\lambda_1}{v_1}\mu_1\right).X_{n-1} && (n \geqslant 0),\\
X_n &= \left(\overline{a_1}, \left(1 - \frac{\lambda_1}{v_1}\right)\mu_1\right).X_{n-1} && (n \geqslant 0),\\
X_n &= (\overline{d_1}, (1-p_{21})v_1).X_{n+1} && (n > 0),\\
X_n &= (\overline{a_2}, \top).X_{n+1} && (n > 0),\\
Y_n &= \left(\overline{e_2}, \frac{\lambda_2}{v_2}\mu_2\right).Y_{n-1} && (n \geqslant 0),\\
Y_n &= \left(\overline{a_2}, (1 - \frac{\lambda_2}{v_2})\mu_2\right).Y_{n-1} && (n \geqslant 0),\\
Y_n &= (\overline{d_2}, (1-p_{12})v_2).Y_{n+1} && (n > 0),\\
Y_n &= (\overline{a_1}, \top).Y_{n+1} && (n > 0).
\end{aligned}
$$

This agent has the state transition graph shown on the right of Fig. 16. It can be seen that the reversed process is that of the same pair of queues (i.e. with the same service rates, $\mu_1$ and $\mu_2$) with the traffic flows reversed. The routing probability for a departure from node $i$ $(=1,2)$ in the reversed network is the ratio of the external arrival rate to the total visitation rate at node $i$ in the forward network. The external arrival rate to node $i$ in the reversed network is equal to the product of the total visitation rate of node $i$ and the departure probability on the unreversed arc in the forward network.

### 6.4. General queueing networks

The conclusion of the previous subsection suggests the following property for general Markovian networks described by cooperations of queues in PEPA.

**Theorem 3.** *The reversed process of a Markovian queueing network, defined by a set of constant external arrival rates to specified nodes, a set of nodes' constant service rates and a set of constant routing probabilities, is that of the same network with the traffic flows reversed in which:*

- *The routing probabilities for (internal and external) departures from each node are proportional to the traffic rates of the corresponding arrivals in the forward network.*

- *The external arrival rate to each node is equal to the product of the total visitation rate at that node and the external departure probability from it in the forward network.*

**Proof.** The proof is by induction on the number $M$ of nodes in the network. For $M=1$ the result is true by Definition 2. In fact we already have the result for $M=2$ in the previous subsection and we extend that analysis to the cooperation of a single queue with service rate $\mu_0$ ('node 0'), with $m+1$ inputs and $n+1$ outputs, and a network ('the network') of $M$ nodes with at least $m$ departing arcs and $n$ arrival arcs, for $M \geqslant m, n \geqslant 1$. We assume without loss of generality that there are external arrivals to node 0, described by action $(a_0, \lambda_0)$, and one external departure stream from node 0, selected with routing probability $p_{0\infty} = 1 - \sum_{i=0}^{M} p_{0i}$ and described by action $(b_0, p_{0\infty}\mu_0)$. If there are no external arrivals to node 0, we set $\lambda_0 = 0$ and similarly for no external departures, we would set $p_{0\infty} = 0$.

Let the action types representing the passage of tasks from node 0 to the network be labeled $b_1, \dots, b_n$ and those representing passage of tasks from the network to node 0 be $a_1, \dots, a_m$. Without loss of generality, let the node in the network that corresponds to the destination of arc $b_i$ be numbered $i$ $(1 \leqslant i \leqslant n)$.

We now apply Theorem 1. Its conditions hold by the inductive hypothesis and the same arguments for node 0 that were used in Section 6.3. Condition 3 requires the existence of constant solutions for the $m+n$ bound variables $x_a$, which comes at the end of the proof. For node 0, by Definition 2,

- the reversed rate of action type $\overline{b_i}$ (in $\bar{R}$) is $(\lambda_0 + \sum_{j=1}^{m} \top_{a_j}) p_{0i}$ for $0 \leqslant i \leqslant n$ where $p_{0i}$ is the routing probability from node 0 to node $i$ in the network;
- the reversed rate of action type $\overline{a_0}$ is $\lambda_0 \mu_0 / (\lambda_0 + \sum_{j=1}^{m} \top_{a_j})$.

For the network, let the visitation rate at any node $j$, when the traffic corresponding to actions $b_i$ is $\top_{b_i}$ $(1 \leqslant i \leqslant n)$, be denoted $v_j(\top)$, where $\top = (\top_{b_1}, \dots, \top_{b_n})$. Then, by the inductive hypothesis,

- the reversed rate of action type $\overline{a_j}$ is $v_{j'}(\top) p_{j'0}$, where $j'$ is the source node corresponding to arc $a_j$, $1 \leqslant j \leqslant m$.

Following Theorem 1, we solve the equations:

$$\top_{b_i} = \left( \lambda_0 + \sum_{j=1}^{m} \top_{a_j} \right) p_{0i} \quad \text{for } 1 \leqslant i \leqslant n,$$

$$\top_{a_j} = v_{j'}(\top) p_{j'0} \quad \text{for } 1 \leqslant j \leqslant m.$$

But the solution to these equations is precisely the set of (internal) traffic rates on the arcs $b_1, \dots, b_n$ and $a_1, \dots, a_m$. Since all other arcs are not involved in the cooperation, their rates in the reversed process are as in the reversed process for the $M$-node network when the external arrival rates are $\top$. It is known that these traffic equations have a solution—unique in open networks, unique up to a multiplicative constant in closed networks. Condition 3 is thereby established and the theorem is proved by induction. □

The product-form solution for a Markovian queueing network's equilibrium state probabilities [13,9] now follows directly from Proposition 1.

**Corollary 1** (Jackson's Theorem). *An $M$-node, ergodic, Markovian queueing network with visitation rate $v_i$ and service rate $\mu_i$ at node $i$ $(1 \leqslant i \leqslant M)$ has equilibrium probability for state $(n_1,\ldots,n_M)$*

$$\pi(n_1,\ldots,n_M) \propto \prod_{i=1}^{M} \left(\frac{v_i}{\mu_i}\right)^{n_i}.$$

**Proof.** Suppose initially that there are external arrivals and departures at all nodes and choose as a reference state the empty network, state $(0,\ldots,0)$. To a given state $(n_1,\ldots,n_M)$, we choose the path from the reference state going along each of the $M$ dimensions in turn, from 0 to $n_i$ in dimension $i$ successively for $i=1,2,\ldots,M$. The product of the rates along dimension $i$ on this path in the reversed process is then $(v_i p_{i0})^{n_i}$. The product of the rates on the unreversed path in the forward process is $(\mu_i p_{i0})^{n_i}$. The result now follows from Proposition 1.

In general, any diagonal arc $d$ corresponds to a departure from some node $a$ that passes to some other node $b$, $1 \leqslant a \neq b \leqslant M$. The arc then has a state $(n_1,\ldots,n_{a-1},n_a+1,n_{a+1},\ldots,n_M)$ as source and $(n_1,\ldots,n_{b-1},n_b+1,n_{b+1},\ldots,n_M)$ as destination, for vector $\mathbf{n}=(n_1,\ldots,n_M)$ with non-negative components. The forward rate on $d$ is $p_{ab}\mu_a$ and the reversed rate is $(p_{ab}v_a/v_b)\mu_b$. Consequently, for any path from the reference state that includes arc $d$, the ratio of these rates contributing to the product is $(v_a/\mu_a)(v_b/\mu_b)$. This path is therefore equivalent to the path in which the arc $d$ is replaced by the two arcs from $(n_1,\ldots,n_{a-1},n_a+1,n_{a+1},\ldots,n_M)$ to $\mathbf{n}$ (in dimension $a$ only) and from $\mathbf{n}$ to $(n_1,\ldots,n_{b-1},n_b+1,n_{b+1},\ldots,n_M)$ (in dimension $b$ only). Since $v_a$ and $v_b$ are positive real numbers, this property holds whether or not the two arcs in dimensions $a$ and $b$ exist, i.e. whether or not $p_{a0}>0$ and $\lambda_b>0$. Hence, our assumption in the first part of the proof loses no generality. □

This corollary is equally valid for open or closed queueing networks because of the argument about diagonal arcs in the proof. In the latter case, all the external arrival rates and departure routing probabilities are zero and the rates $v_i$ may be chosen up to an arbitrary multiplicative constant; recall that the RCAT calculates non-normalised state probabilities. Note that it is straightforward to generalise the theorem and its corollary to locally state-dependent service rates, so proving the general form of Jackson's theorem.

### 6.5. Networks with negative customers

Queueing networks with negative customers (G-nets) were introduced by Gelenbe around 1990, initially to model neural networks but with diverse applications following; see for example [5,6]. At the time, many considered it surprising that such networks had a product-form at all, in view of the irregularities introduced by the negative customers. Like service completions, these reduce queue lengths, but they appear from

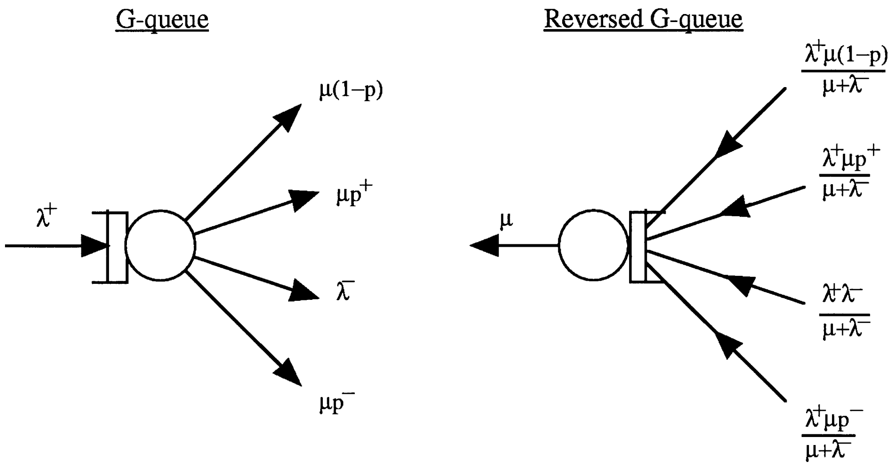


Fig. 17. G-queue network node and its reversed queue.

other nodes; in a Jackson network the service rate at a node cannot depend on the states of other queues.

We represent a single $M/M/1$ queue with negative customers (called a G-queue) by a normal $M/M/1$ queue with arrival rate $\lambda^+$ (that of the positive arrivals) and service rate $\mu+\lambda^-$ where $\mu$ is the actual rate of service and $\lambda^-$ is the negative arrival rate. In other words, we split the departure arc from a busy queue into a 'service' component and a 'negative arrival' component. The reversed queue is then the same $M/M/1$ queue. In this model, there are no negative arrivals to an empty queue. Equivalently, we may allow negative arrivals when the queue length is zero, but they have no effect and may be represented by an invisible transition on state 0 with arbitrary rate, e.g. $\lambda^-$. The same will apply in the reversed process, see Section 4.3.5, which remains identical to the forward process.

In a network of G-queues, customers completing service normally at a node $i$ may pass to a node $j$ as either a positive or a negative customer, with respective probabilities $p_{ij}^+$ or $p_{ij}^-$, or else leave the network. The generic single G-queue, for use in a network, and its reversed queue—identical apart from a redistribution of rates—is as shown in Fig. 17. Departures from the queue shown go to another queue with probability $p$ ($p^+$ if positive, $p^-$ if negative, $p^+ + p^- = p$) or leave the network with probability $1-p$. It is straightforward to further split the departures to allow for transitions to any number of nodes, as in Jackson networks.

Now let us try to apply the RCAT to a G-network of two nodes. Its conditions are satisfied since positive arrivals are always enabled in a single queue, as are negative arrivals if we use invisible transitions on empty queue states (condition 1). The reversed actions of the co-operating active actions are arrivals and also always enabled, as in a Jackson network (condition 2). The reversed rates of the active actions in each queue are more complex than in the standard $M/M/1$ queue, and again are determined by Definition 2, cf. Fig. 17. They are equated to the rates $x_a$ of Theorem 1 and we shall show that a solution for these equations does exist, validating condition 3. Suppose each node has positive and negative external arrivals with rates $\lambda_i^+$, $\lambda_i^-$, respectively, $i=1,2$. In the RCAT, we solve the following equations for the reversed rates of the active actions, bound to the passive rates in $P$ and $Q$ in order to define $R$ and $S$:

$$\top_2^+ = \frac{\mu_1 p_{12}^+(\lambda_1^+ + \top_1^+)}{\mu_1 + \lambda_1^- + \top_1^-},$$

$$\top_2^- = \frac{\mu_1 p_{12}^-(\lambda_1^+ + \top_1^+)}{\mu_1 + \lambda_1^- + \top_1^-},$$

$$\top_1^+ = \frac{\mu_2 p_{21}^+(\lambda_2^+ + \top_2^+)}{\mu_2 + \lambda_2^- + \top_2^-},$$

$$\top_1^- = \frac{\mu_2 p_{21}^-(\lambda_2^+ + \top_2^+)}{\mu_2 + \lambda_2^- + \top_2^-}.$$

Writing $v_i^+ = \top_i^+ + \lambda_i^+$ and $v_i^- = \top_i^- + \lambda_i^-$, $i=1,2$, we find

$$v_i^+ = \frac{\mu_{i'} p_{i'i}^+ v_{i'}^+}{\mu_{i'} + v_{i'}^-} + \lambda_i^+,$$

$$v_i^- = \frac{\mu_{i'} p_{i'i}^- v_{i'}^+}{\mu_{i'} + v_{i'}^-} + \lambda_i^-,$$

where $i'=1$ if $i=2$ and $i'=2$ if $i=1$.

These are exactly the traffic equations derived and solved by Gelenbe [6] and lead to the same product-form by the argument applied in Jackson networks. Note that the proof of existence of a solution to these equations is essential since otherwise condition 3 of the RCAT could not hold.

The result generalises to G-nets of $M \geqslant 2$ nodes in the same way as Theorem 3.

## 7. Conclusion

We have introduced a new approach to deriving the equilibrium state probabilities for continuous time Markov chains that does not require balance equations to be solved. Instead, the reversed process is determined using only the instantaneous transition rates of the Markov chain and a simple product-form solution ensues. Incorporating this into the framework of a Markovian process algebra—here, PEPA—leads to a new, mechanisable methodology for generating reversed processes and hence product-form solutions from specifications. For simple PEPA agents, defined without the cooperation combinator, the same heuristic approach for Markov chains is adapted to the PEPA syntax. However, for compound agents, only those nodes directly involved in cooperations need be analysed, specifically for conservation of outgoing flow and products of rates around cycles.

Moreover, under certain conditions, the reverse of a cooperation is the cooperation of the reversed processes with some remapping of transition rates. This result has far-reaching consequences. Firstly, it provides many product-form solutions automatically, and in particular, an automated proof (in PEPA) of Jackson's theorem by a completely different method—with no balance equations. Indeed, the same constraints arise as in the original theorem, but from a different source, viz. constant arrival rates to ensure unique reversed rates to bind with the unspecified rates of the same action type.

More generally, any composition of subnetworks with known reversed processes can be handled by this methodology, particularly effectively when the RCAT, Theorem 1, can be applied. The ease with which G-networks could be solved in the same vein as Jackson's theorem is a good illustration of the utility of this compositional approach.

It is interesting to look at the connection between the RCAT and results on the existence of product-forms. As noted already, the conditions required in Jackson's theorem arise quite naturally in our approach. Consider now networks with blocking. When a queue is blocked by a downstream queue that is full, there are various scenarios that may be implemented in a physical system. One is for the upstream queue to remain blocked until there has been a service completion at the downstream queue that makes space for the blocked customer. This is called 'blocking after service', there is no known product form for such networks and it is thought that none exists. If we try to apply the RCAT, we would find that (passive) arrivals to the downstream queue are not always enabled—i.e. not when the queue is full—and so condition 1 is violated. An alternative strategy is to discard the blocked customer immediately so that the upstream server can continue. Such is called a 'loss' network or 'system with losses' and there are product-forms. We still cannot apply the RCAT directly, but we can introduce an invisible passive transition at the downstream server when it is in the full-state, cf. Section 4.3.5. This describes losses perfectly and the RCAT can be applied, yielding a product-form solution, again in a natural way.

Under appropriate conditions, it may be that an explicitly solved, non-product-form subnetwork could be combined with a product-form subnetwork and the reversed process obtained using the RCAT—if its conditions are satisfied. This would then provide a product-form solution, using the blocking subnetwork as one of its primitive cooperating agents. If the conditions were not satisfied, a product-form might still follow using the Weak RCAT, but at greater computational complexity. This is clearly an area for further investigation.

The most obvious theoretical research direction is to relax the conditions of the RCAT to allow reversed agents for a wider range of compound agents to be determined. Condition 3 of the RCAT appears not unduly restrictive at first sight since, if some action type is associated with multiple reversed rates, we can always introduce a new action type for each distinct reversed rate. However, this could lead to a large number—perhaps infinity—of action types and hence equations to be solved for the set $\{\top_a\}$. Moreover, this process would in general invalidate condition 1 since the new action types would not be enabled in all states.

Many of the state spaces encountered in stochastic models and MPA are infinite and so the derivation of ergodicity conditions is important. The use of *ad hoc* truncation is often effective, but rarely rigorous, i.e. with precise error bounds. Moreover, truncation or the consideration of a realistic finite model, e.g. with finite capacity queues, often violates product-form conditions—in particular those of the RCAT. Another direction for theoretical research is therefore to prove ergodicity in certain classes of cooperation; recall it was assumed as a hypothesis in the RCAT. One possible route is the asymptotic analysis relating to [1] referred to in Section 2.2, where we are interested in sub-chains in the reducible limit of a family of Markov processes.

More rigorous heuristics would also be desirable for handling simple agents, which may themselves be complex: afterall, *any* Markov graph can be thus described. However, it is not necessary to use the Kolmogorov criteria and we can always fall back on the direct solution of the (linear) balance equations to compute the reversed rates, using Proposition 1.

However, the most pressing work in progress is the implementation of the theorems and methodologies introduced in this paper. Both the methodology proposed for simple agents and for analysing nodes and arcs involved in cooperations need to be mechanised, involving in particular efficient algorithms for finding the minimal cycles in a graph. Mechanisation of the RCAT is straightforward, but the automation of proof methods based on this is a greater challenge. A truly automatic proof of Jackson's theorem would represent a triumph of computer science theory in an engineering application.

## Acknowledgements

This work was supported by the EPSRC grant number GR/L10222. It is a pleasure to acknowledge my ex-PhD student and colleague Kamyar Kanani for many illuminating discussions and, not least, the title of this paper. Finally, thanks to Ann, my inspiration for the best parts.

## References

[1] V.V. Anisimov, Asymptotic aggregation of states for finite Markov processes, unpublished private communication, 2001.
[2] M. Bernardo, R. Gorrieri, A tutorial on EMPA: a theory of concurrent processes with nondeterminism, priorities, probabilities and time Theoret. Comput. Sci. 202 (1998) 1–54.
[3] P.J. Burke, The output of a queueing system, Oper. Res. 4 (1956) 699–704.
[4] G. Clark, J. Hillston, Product form solution for an insensitive stochastic process algebra structure, Performance Evaluat. (2002) to appear.
[5] E. Gelenbe, Random neural networks with positive and negative signals and product form solution, Neural Comput. 1 (4) (1989) 502–510.
[6] E. Gelenbe, Queueing networks with negative and positive customers, J. Appl. Probab. 28 (1991) 656–663.
[7] N. Goetz, U. Herzog, M. Rettelbach, Multiprocessor and distributed system design: the integration of functional specification and performance analysis using stochastic process algebras, in: Proc. PERFORMANCE '93, Rome, Lecture Notes in Computer Science, Vol. 229, Springer, Berlin, 1993, pp. 121–146.
[8] P.G. Harrison, J. Hillston, Exploiting quasi-reversible structures in Markovian process algebra models, Comput. J. 38 (7) (1995) 510–520.

[9] P.G. Harrison, N.M. Patel, Performance Modelling of Communication Networks and Computer Architectures, Addison-Wesley, Reading, MA, 1992.
[10] P.G. Harrison, B. Strulo, Spades–a process algebra for discrete event simulation, J. Logic Comput. 10 (1) (2000) 3–42.
[11] J. Hillston, A compositional approach to performance modelling, Ph.D. Thesis, University of Edinburgh, 1994.
[12] J. Hillston, N. Thomas, Product form solution for a class of PEPA models, Performance Evaluat. 35 (3–4) (1999) 171–192.
[13] J.R. Jackson, Jobshop-like queueing systems, Manag. Sci. 10 (1) (1963) 131–142.
[14] J.P. Katoen, Markovian bisimulation, Sci. Comput. Programming 202 (2000) 1–54.
[15] J.P. Katoen P. D'Argenio, E. Brinksma, General purpose discrete event simulation using ♠, in: Proc. PAPM 1998, Nice, July 1998, pp. 85–102.
[16] F.P. Kelly, Reversibility and Stochastic Networks, Wiley, New York, 1979.
[17] W. Knottenbelt, P. Harrison, M. Mestern, P. Kritzinger, A probabilistic dynamic technique for the distributed generation of very large state spaces, Performance Evaluat. J. 39 (1–4) (2000) 127–148.
[18] W. Knottenbelt, P. Harrison. Distributed disk-based solution techniques for large Markov models, in: Proc. 3rd Internat. Conf. Numerical Solution of Markov Chains, NSMC '99, Zaragoza, September 1999, pp. 58–75.
[19] B. Plateau, W. Stewart, Kronecker algebra, J. Assoc. Comput. Mach. 45 (2) (1998).
[20] M. Sereno, Towards a product form solution for stochastic process algebras, Comput. J. 38 (7) (1995) 622–632.