# Iterative Convergence of Passage-time Densities in Semi-Markov Performance Models

Jeremy T. Bradley [a] and Helen J. Wilson [b,1]

[a] *Department of Computing, Imperial College London, 180 Queen's Gate, London SW7 2BZ, United Kingdom*

[b] *Department of Applied Mathematics, University of Leeds, Woodhouse Lane, Leeds LS2 9JT, United Kingdom*

**Abstract**

Passage-time densities are important for the detailed performance analysis of distributed computer and communicating systems. We provide a proof and demonstration of a practical iterative algorithm for extracting complete passage-time densities from expressive semi-Markov systems. We end by showing its application to a distributed web-server cluster model of 15.9 million states.

*Key words:* passage-time density, iterative algorithm, analytic performance modelling, semi-Markov process, stochastic Petri nets

## 1 Introduction

Passage-time densities yield important performance metrics for communication and distributed computer systems. An individual passage-time can be used to represent the elapsed time over a mission-critical sequence of system operation. For instance, we might require the time-to-failure of a satellite communications subsystem or the response-time of a typical search-engine query.

By deriving the full passage-time density for these quantities, we can further calculate probabilistic quality-of-service guarantees from the cumulative dis-

*Email addresses:* `jb@doc.ic.ac.uk` (Jeremy T. Bradley), `h.j.wilson@leeds.ac.uk` (Helen J. Wilson).

[1] *Current address:* Department of Mathematics, University College London, Gower Street, London WC1E 6BT, United Kingdom

tribution. For example, we might need to establish that there is a probability of less than 0.01 that the time-to-call-loss for a mobile phone base-station is less than a minute.

In this paper, we look at an iterative technique for calculating passage-time densities [1] in semi-Markov systems. Techniques for evaluating passage-times in Markovian systems have existed for some time [2–4], however the practical calculation of passage-times in a semi-Markov environment, where arbitrary distributions of atomic events are permitted, is a recent development. In previous work [1,5] we described the iterative algorithm presented here, but had no direct proof that the iterative scheme converged to the correct passage-time. Here, we present a complete proof of convergence to the correct result.

We also give graphical demonstrations of convergence of individual iterates to the analytically correct form. Finally, we show the algorithm's practical applicability by showing analysis of a parallel web cluster which requires manipulation of a system of 15.9 million states.

## 2 Background Theory

### *2.1 Semi-Markov Processes*

Consider a Markov renewal process $\{(X_n, T_n) : n \geq 0\}$ where $T_n$ is the time of the $n$th transition ($T_0 = 0$), $X_n \in \mathcal{S}$ is the state at the $n$th transition and $\mathcal{S}$ is the set of states in the Markov renewal process. Let the kernel of this process be:

$$R(n, i, j, t) = \mathbb{P}(X_{n+1} = j, T_{n+1} - T_n \leq t \mid X_n = i) \tag{1}$$

for $i, j \in \mathcal{S}$. The continuous time semi-Markov process (SMP), $\{Z(t), t \geq 0\}$, defined by the kernel $R$, is related to the Markov renewal process by:

$$Z(t) = X_{N(t)} \tag{2}$$

where $N(t) = \max\{n : T_n \leq t\}$, i.e. the number of state transitions that have taken place by time $t$. Thus $Z(t)$ represents the state of the system at time $t$. We consider time-homogeneous SMPs, in which $R(n, i, j, t)$ is independent of any previous state except the last. Thus $R$ becomes independent of $n$:

$$\begin{aligned} R(i, j, t) &= \mathbb{P}(X_{n+1} = j, T_{n+1} - T_n \leq t \mid X_n = i) \text{ for any } n \geq 0 \\ &= p_{ij} H_{ij}(t) \end{aligned} \tag{3}$$

where $p_{ij} = \mathbb{P}(X_{n+1} = j \mid X_n = i)$ is the state transition probability from state $i$ to state $j$ and $H_{ij}(t) = \mathbb{P}(T_{n+1} - T_n \leq t \mid X_{n+1} = j, X_n = i)$, is the sojourn time distribution in state $i$ when the next state is $j$.

### *2.2 First passage-times*

Consider a finite, irreducible, continuous-time semi-Markov process with $N_s$ states $\{1, 2, \ldots, N_s\}$. Recalling that $Z(t)$ denotes the state of the SMP at time $t$ ($t \geq 0$), the first passage-time from a source state $i$ at time $t$ into a non-empty set of target states $\vec{j}$ is:

$$P_{i\vec{j}}(t) = \inf\{u > 0 : Z(t+u) \in \vec{j}, N(t+u) > N(t), Z(t) = i\} \tag{4}$$

For a stationary time-homogeneous SMP, $P_{i\vec{j}}(t)$ is independent of $t$ and we have:

$$P_{i\vec{j}} = \inf\{u > 0 : Z(u) \in \vec{j}, N(u) > 0, Z(0) = i\} \tag{5}$$

The $N(u) > 0$ condition ensures that at least one transition has taken place before the passage can be counted, as is conventional. This formulation of the random variable $P_{i\vec{j}}$ applies to an SMP with no immediate (that is, zero-time) transitions. If zero-time transitions are permitted in the model then the passage-time can be stated as:

$$P_{i\vec{j}} = \inf\{u > 0 : N(u) \geq M_{i\vec{j}}\} \tag{6}$$

where $M_{i\vec{j}} = \min\{m \in \mathbb{Z}^+ : X_m \in \vec{j} \mid X_0 = i\}$ is the transition which marks the terminating state of the passage.

$P_{i\vec{j}}$ has an associated probability density function $f_{i\vec{j}}(t)$ such that the passage-time quantile is given as:

$$\mathbb{P}(t_1 < P_{i\vec{j}} < t_2) = \int_{t_1}^{t_2} f_{i\vec{j}}(t)\ \mathrm{d}t \tag{7}$$

In general, the Laplace transform of $f_{i\vec{j}}$, $L_{i\vec{j}}(s)$, can be computed by solving a set of $N_s$ linear equations:

$$L_{i\vec{j}}(s) = \sum_{k \notin \vec{j}} r^*_{ik}(s) L_{k\vec{j}}(s) + \sum_{k \in \vec{j}} r^*_{ik}(s) \qquad \text{for } 1 \leq i \leq N_s \tag{8}$$

where $r^*_{ik}(s)$ is the Laplace-Stieltjes transform (LST) of $R(i,k,t)$ from Section 2.1 and is defined by:

$$r^*_{ik}(s) = \int_0^\infty \mathrm{e}^{-st}\ \mathrm{d}R(i,k,t) \tag{9}$$

Equation (8) has a matrix-vector form, $A\tilde{x} = \tilde{b}$, where the elements of $A$ are general complex functions; care needs to be taken when storing such functions for eventual numerical inversion (see Section 4). For example, when $\vec{j} = \{1, 2\}$, Equation (8) yields:

$$\begin{pmatrix} 1 & 0 & -r^*_{13}(s) & \cdots & -r^*_{1N_s}(s) \\ 0 & 1 & -r^*_{23}(s) & \cdots & -r^*_{2N_s}(s) \\ 0 & 0 & 1-r^*_{33}(s) & \cdots & -r^*_{3N_s}(s) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & -r^*_{N_s3}(s) & \cdots & 1-r^*_{N_sN_s}(s) \end{pmatrix} \begin{pmatrix} L_{1\vec{j}}(s) \\ L_{2\vec{j}}(s) \\ L_{3\vec{j}}(s) \\ \vdots \\ L_{N_s\vec{j}}(s) \end{pmatrix} = \begin{pmatrix} r^*_{11}(s) + r^*_{12}(s) \\ r^*_{21}(s) + r^*_{22}(s) \\ r^*_{31}(s) + r^*_{32}(s) \\ \vdots \\ r^*_{N_s1}(s) + r^*_{N_s2}(s) \end{pmatrix} \tag{10}$$

When there are multiple source states, denoted by the vector $\vec{i}$, we can weight the passage-time density with some initial distribution for the system. For instance the Laplace transform of the passage-time density at steady-state is:

$$L_{\vec{i}\vec{j}}(s) = \sum_{k \in \vec{i}} \alpha_k L_{k\vec{j}}(s) \tag{11}$$

where the weight $\alpha_k$ is the probability at equilibrium that the system is in state $k \in \vec{i}$ at the starting instant of the passage. If $\tilde{\pi}$ denotes the steady-state vector of the embedded discrete-time Markov chain (DTMC) with one-step transition probability matrix $D = [p_{ij}, 1 \leq i, j \leq N_s]$, then $\alpha_k$ is given by:

$$\alpha_k = \begin{cases} \pi_k / \sum_{j \in \vec{i}} \pi_j & \text{if } k \in \vec{i} \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

The vector with components $\alpha_k$ is denoted by $\tilde{\alpha}$.

### *2.3 Iterative Algorithm for Evaluating Passage-times*

In this section, we outline the algorithm from Bradley *et al.* [1] for generating passage-time densities, that creates successively better approximations to the SMP passage-time quantity of Equation (8). The technique considers the $r$th transition passage-time of the system, $P^{(r)}_{i\vec{j}}$. This is the conditional passage-time of the system having reached any of the specified target states within $r$ state-transitions. The unconditioned passage-time random variable, $P_{i\vec{j}}$, is then obtained in the limit as $r \to \infty$. We calculate the Laplace transform of $P^{(r)}_{i\vec{j}}$, $L^{(r)}_{i\vec{j}}(s)$, and pick a sufficiently high value of $r$ to give an approximation to $L_{i\vec{j}}(s)$ to within a specified degree of accuracy. $L_{i\vec{j}}(s)$ can then be numerically inverted to obtain the desired passage-time density.

This iterative method bears a loose resemblance to the well-known *uniformization* technique [6–8] which can be used to generate transient-state distributions and passage-time densities for Markov chains. However, as we are working with semi-Markov systems, there can be no *uniformizing* of the general distributions in the SMP. The general distribution information has to be maintained as precisely as possible throughout the process, which we achieve using the general distribution representation technique described in Section 4.

Recall the semi-Markov process, $Z(t)$, of Section 2.1, where $N(t)$ is the number of state transitions that have taken place by time $t$. We formally define the $r$th transition first passage-time to be:

$$P_{i\vec{j}}^{(r)} = \inf\{u > 0 : Z(u) \in \vec{j}, 0 < N(u) \leq r, Z(0) = i\} \tag{13}$$

which is the time taken to enter a state in $\vec{j}$ for the first time having started in state $i$ at time 0 and having undergone up to $r$ state transitions[2].

Again, if we have immediate transitions in our SMP model then the $r$th transition first passage-time becomes (cf. Equation (6)):

$$P_{i\vec{j}}^{(r)} = \inf\{u > 0 : M_{i\vec{j}} \leq N(u) \leq r\} \tag{14}$$

Its Laplace transform, $L_{i\vec{j}}^{(r)}(s)$, is in turn the $i$th component of the vector:

$$\tilde{L}_{\vec{j}}^{(r)}(s) = (L_{1\vec{j}}^{(r)}(s), L_{2\vec{j}}^{(r)}(s), \ldots, L_{N_s\vec{j}}^{(r)}(s)) \tag{15}$$

representing the passage-time which terminates in $\vec{j}$ from each possible start state. We compute this vector as:

$$\tilde{L}_{\vec{j}}^{(r)}(s) = U(I + U' + U'^2 + \cdots + U'^{(r-1)})\, \tilde{\delta}^{\vec{j}} \tag{16}$$

where $U$ is a matrix with elements $u_{pq} = r_{pq}^*(s)$ and $U'$ is a modified version of $U$ with elements:

$$u'_{pq} = \begin{cases} u_{pq} & \text{if } p \notin \vec{j} \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

[2] This is well-defined even in the case when the set $\vec{j}$ is not reached from state $i$ in $r$ transitions, when the set of $u$ in (13) is empty, that is:

$$\{u > 0 : Z(u) \in \vec{j}, 0 < N(u) \leq r, Z(0) = i\} = \emptyset$$

and since $\inf \emptyset = \infty$ by definition, the random variable, $P_{i\vec{j}}^{(r)} = \infty$.
This covers the case where the set $\vec{j}$ is unreachable by *any* realisation of the system in $r$ transitions, which makes the cumulative distribution function, $\mathbb{P}(P_{i\vec{j}}^{(r)} \leq t) = 0$, for any finite $t$. So if no route exists to the target states in $r$ transitions (or at all) the resulting distribution function of $P_{i\vec{j}}^{(r)}$ is 0 for all $t$, as required.

where states in $\vec{j}$ have been made absorbing. The column vector $\tilde{\delta}^{\vec{j}}$ has entries:

$$\delta_k^{\vec{j}} = \begin{cases} 1 \text{ if } k \in \vec{j} \\ 0 \text{ otherwise} \end{cases} \tag{18}$$

The physical justification behind Equation (16) is that it represents the sum of contributions to the overall passage-time $\tilde{L}_{\vec{j}}^{(r)}(s)$ from each of the $n$-transition matrices [3], $1 \leq n \leq r$. In Section 3, we will show that this sequence converges as $r \to \infty$, and that its limiting value, $L_{\vec{i}\vec{j}}^{(\infty)}(s) = \lim_{r\to\infty} L_{\vec{i}\vec{j}}^{(r)}(s)$ is a solution of the central passage equation, Equation (8).

We can generalise to multiple source states $\vec{i}$ using, for instance, the normalised steady-state vector, $\tilde{\alpha}$, of Equation (12):

$$L_{\vec{i}\vec{j}}^{(r)}(s) = \tilde{\alpha}\tilde{L}_{\vec{j}}^{(r)}(s) = \sum_{k=0}^{r-1} \tilde{\alpha} U U'^k \, \tilde{\delta}^{\vec{j}} \tag{19}$$

or any such vector which represents some initial distribution for the system.

The worst-case time complexity for this sum is $O(N_s^2 r)$ versus the $O(N_s^3)$ of typical matrix inversion techniques. In practice, for a sparse matrix with constant bandwidth (number of non-zeros per row), this can be as low as $O(N_s r)$.

Further details of efficient distributed implementation of Equation (19) can be found in [5].

## 3 Proof of Convergence and Correctness

### *3.1 Technical Summary*

In this section, we prove the convergence of the iterative scheme Equation (16):

$$\tilde{L}_{\vec{j}}^{(r)}(s) = U(I + U' + U'^2 + \cdots + U'^{(r-1)}) \, \tilde{\delta}^{\vec{j}}$$

[3] The $n$th transition matrix is defined to be $UU'^{n-1}$ so as to ensure that the passage measures at least one transition.

and show that the limiting quantity $\tilde{L}^{(\infty)}_{\vec{j}}(s)$ does, as required, satisfy Equation (8):

$$L^{(\infty)}_{i\vec{j}}(s) = \sum_{k \notin \vec{j}} r^*_{ik}(s) L^{(\infty)}_{k\vec{j}}(s) + \sum_{k \in \vec{j}} r^*_{ik}(s) \qquad : \text{for } 1 \leq i \leq N_s \tag{20}$$

### *3.2 Preliminary observations*

It is clear from the definition of $p_{ij}$ below Equation (3) that $p_{ij}$ are non-negative real numbers with:

$$\sum_k p_{jk} = 1 \text{ for all } j \tag{21}$$

and the definition of $r^*_{ij}(s)$ in Equation (9) immediately yields:

$$r^*_{ij}(s) = p_{ij} h^*_{ij}(s) \tag{22}$$

in which $h^*_{ij}(s)$ is the Laplace-Stieltjes transform of the sojourn time distribution $H_{ij}(t)$.

We will use the vector $\tilde{\delta}^i$ (similar to Equation (18)), whose elements are:

$$\delta^i_k = \begin{cases} 1 \text{ if } k = i \\ 0 \text{ otherwise} \end{cases} \tag{23}$$

### *3.3 Conjectures*

In this section, we state and prove several facts about our system, culminating in the convergence of $\tilde{L}^{(r)}_{\vec{j}}(s)$ as defined above. Throughout, the term *eigenvector* will be used as shorthand for eigenvector of a given matrix under right-multiplication: thus $\tilde{v}$ is an eigenvector of matrix $U$ if $\tilde{v}U = \lambda\tilde{v}$.

**Lemma 1** *For any given value of $s$ with non-negative real part, the modulus of each complex number $h^*_{ij}(s)$ is at most 1.*

**Proof** We know that $\int_0^\infty H_{ij}(t)\ \mathrm{d}t = 1$, because $H_{ij}(t)$ is a distribution function. Let $s = x + \mathrm{i}y$. Then $x \geq 0$ and:

$$\begin{aligned} \left|h^*_{ij}(s)\right| &= \left|\int_0^\infty \mathrm{e}^{-xt}(\cos yt - \mathrm{i}\sin yt) H_{ij}(t)\ \mathrm{d}t\right| \\ &\leq \int_0^\infty \mathrm{e}^{-xt}\, H_{ij}(t)\ \mathrm{d}t \leq \int_0^\infty H_{ij}(t)\ \mathrm{d}t = 1 \end{aligned}$$

**Lemma 2** *Right-multiplication by $U'$ does not increase the $L_1$-norm of any vector $\tilde{v}$:*

$$\|\tilde{v}U'\|_1 \leq \|\tilde{v}\|_1 \quad \textit{and hence} \quad \|\tilde{v}U'^n\|_1 \leq \|\tilde{v}\|_1 \ \textit{ for all } n \geq 0$$

*Note that this is a stronger statement than the spectral radius of $U'$ being at most 1, which follows trivially from evaluation of the infinity-norm of $U'$.*

**Proof**

$$\begin{aligned}
\|\tilde{v}U'\|_1 &= \sum_k |(\tilde{v}U')_k| = \sum_k \left|\sum_j v_j U'_{jk}\right| \\
&\leq \sum_k \sum_j |v_j|\,|U'_{jk}| = \sum_k \sum_{j \notin \vec{j}} |v_j|\,|U_{jk}| \\
&\leq \sum_k \sum_{j \notin \vec{j}} |v_j|\, p_{jk} = \sum_{j \notin \vec{j}} |v_j| \sum_k p_{jk} \\
&= \sum_{j \notin \vec{j}} |v_j| \leq \sum_j |v_j| = \|\tilde{v}\|_1
\end{aligned} \tag{24}$$

The extension to $n$ multiplications by $U'$ is clear.

**Lemma 3** *Any eigenvalue $\lambda$ of $U'$ which has modulus 1 is non-defective, that is, there is no vector $\tilde{w}$ for which $\tilde{w}(U' - \lambda I)^2 = 0$ but $\tilde{w}(U' - \lambda I) \neq 0$.*

**Proof** Assume the converse. Then let $\tilde{w}$ be a principal vector of stage 1 associated with the eigenvalue in question. Then:

$$\tilde{w}(U' - \lambda I)^2 = 0 \quad \text{and} \quad \tilde{w}(U' - \lambda I) = \tilde{v} \neq 0$$

It follows that $\tilde{v}$ is an eigenvector corresponding to $\lambda$ and:

$$\tilde{w}U' = \lambda\tilde{w} + \tilde{v}, \quad \text{and hence that} \quad \tilde{w}U'^n = \lambda^n\tilde{w} + n\lambda^{n-1}\tilde{v}$$

Then $\|\tilde{w}U'^n\| \geq n|\lambda|^{n-1}\|\tilde{v}\| - |\lambda|^n\|\tilde{w}\|$, and if $|\lambda| = 1$ then:

$$\|\tilde{w}U'^n\| \geq n\|\tilde{v}\| - \|\tilde{w}\|$$

which grows without bound for large $n$ since $\|\tilde{v}\|$ cannot be 0. However, $\|\tilde{w}U'^n\|$ is bounded above by $\|\tilde{w}\|$ for any $n$, from Lemma 2 so we have a contradiction.

**Lemma 4** *The eigenvectors corresponding to all eigenvalues of modulus 1 have zero components corresponding to elements of $\vec{j}$.*

**Proof** Consider the proof of Lemma 2. For the norm of $\tilde{v}$ to be maintained under multiplication with $U'$ (which must happen for an eigenvector whose eigenvalue has modulus 1), every inequality in the proof must be a strict equality. In particular, from (24) we have:

$$\sum_{j \notin \vec{j}} |v_j| = \sum_j |v_j| \text{ which requires } v_j = 0 \text{ if } j \in \vec{j}$$

Now we consider two subspaces of the complex vector space on which $U'$ acts. $\mathcal{V}_0$ is the vector subspace spanned by the eigenvectors of $U'$ corresponding to eigenvalues of modulus 1, and $\mathcal{V}_1$ is the vector subspace spanned by the eigenvectors and principal vectors corresponding to eigenvalues of modulus strictly less than 1. Since the spectral radius of $U'$ is at most 1, these two subspaces span $\mathbb{C}^{N_s}$, and each of them is invariant under right-multiplication by $U'$ from their definitions.

We introduce a new matrix $U_0$ satisfying:

$$\tilde{v}_0 U_0 = \tilde{v}_0 U' \text{ for all } \tilde{v}_0 \in \mathcal{V}_0 \quad \text{and} \quad \tilde{v}_1 U_0 = 0 \text{ for all } \tilde{v}_1 \in \mathcal{V}_1 \tag{25}$$

and define $U_1 = U' - U_0$. It is trivial to deduce that:

$$\tilde{v}_0 U_1 = 0 \text{ for all } \tilde{v}_0 \in \mathcal{V}_0 \quad \text{and} \quad \tilde{v}_1 U_1 = \tilde{v}_1 U' \text{ for all } \tilde{v}_1 \in \mathcal{V}_1 \tag{26}$$

**Lemma 5** *The matrix $U_1$ has spectral radius strictly less than 1.*

**Proof** Consider any eigenvector of $U_1$. By the first part of Equation (26), either it has eigenvalue 0 or it is a member of $\mathcal{V}_1$. In the latter case it is also an eigenvector of $U'$, whose eigenvalue has modulus less than 1 from the definition of $\mathcal{V}_1$. It follows that $U_1$ has spectral radius strictly less than 1.

**Lemma 6** *For any element $\tilde{v}_0 \in \mathcal{V}_0$, the scalar $\tilde{v}_0 U'^n \tilde{\delta}^{\vec{j}}$ is 0 for all $n \geq 0$.*

**Proof** Since $\mathcal{V}_0$ is an invariant subspace under multiplication by $U'$, we know that $\tilde{v}_0 U'^n \in \mathcal{V}_0$. Any element of $\mathcal{V}_0$ can be decomposed into a sum of eigenvectors of $U'$ corresponding to eigenvalues of modulus 1, and by Lemma 4 each such eigenvector has zeros corresponding to non-zero elements of $\tilde{\delta}^{\vec{j}}$.

**Lemma 7** *The vector quantity $U_0 U'^n \tilde{\delta}^{\vec{j}} = 0$ for all $n \geq 0$.*

**Proof** Write the elements of the vector $U_0 U'^n \delta^{\vec{j}}$ as $[U_0 U'^n \tilde{\delta}^{\vec{j}}]_i = \tilde{\delta}^i U_0 U'^n \tilde{\delta}^{\vec{j}}$. Then $\tilde{\delta}^i U_0 \in \mathcal{V}_0$ and the property follows from Lemma 6.

**Theorem 8** *The limits as $r \to \infty$ of the sequences:*

$$L^{(r)}_{i\vec{j}}(s) = \tilde{\delta}^i U(I + U' + U'^2 + \cdots + U'^{(r-1)})\, \tilde{\delta}^{\vec{j}}$$

*are finite and satisfy Equation (8):*

$$L^{(\infty)}_{i\vec{j}}(s) = \sum_{k \notin \vec{j}} U_{ik}(s) L^{(\infty)}_{k\vec{j}}(s) + \sum_{k \in \vec{j}} U_{ik}(s)$$

**Proof** Let us begin by defining an additional scalar quantity:

$$T^{(i,r)} = [(I + U' + U'^2 + \cdots + U'^{(r-1)})\tilde{\delta}^{\vec{j}}]_i = \tilde{\delta}^i (I + U' + U'^2 + \cdots + U'^{(r-1)})\tilde{\delta}^{\vec{j}}$$

with limit $\lim_{r\to\infty} T^{(i,r)} = \tau_i$. We note that we can calculate $L^{(r)}_{i\vec{j}}(s)$ simply via:

$$L^{(r)}_{i\vec{j}}(s) = \sum_k U_{ik} T^{(k,r)}, \text{ and so } L^{(\infty)}_{i\vec{j}}(s) = \sum_k U_{ik} \tau_k$$

Now we split the vector $\tilde{\delta}^i$ into two components, one from each vector subspace $\mathcal{V}_0$ and $\mathcal{V}_1$. Thus:

$$\tilde{\delta}^i = \tilde{\delta}^{i,0} + \tilde{\delta}^{i,1}$$

in the obvious notation, and we have:

$$\begin{aligned}
T^{(i,r)} &= (\tilde{\delta}^{i,0} + \tilde{\delta}^{i,1})(I + U' + U'^2 + \cdots + U'^{(r-1)})\tilde{\delta}^{\vec{j}} \\
&= \tilde{\delta}^{i,1}(I + U' + U'^2 + \cdots + U'^{(r-1)})\tilde{\delta}^{\vec{j}} \quad \text{by Lemma 6} \\
&= \tilde{\delta}^{i,1}(I + U_1 + U_1^2 + \cdots + U_1^{(r-1)})\tilde{\delta}^{\vec{j}} \quad \text{by Equations (25–26)} \\
&= \tilde{\delta}^{i}(I + U_1 + U_1^2 + \cdots + U_1^{(r-1)})\tilde{\delta}^{\vec{j}} \quad \text{by Lemma 6} \\
&= [(I + U_1 + U_1^2 + \cdots + U_1^{(r-1)})\delta^{\vec{j}}]_i
\end{aligned}$$

and since $U_1$ has spectral radius strictly less than 1, Theorem 6.2.8 of [9] implies:

$$\lim_{r\to\infty}(I + U_1 + U_1^2 + \cdots + U_1^{(r-1)}) = (I - U_1)^{-1} \quad \text{so} \quad \tau_i = [(I - U_1)^{-1}\delta^{\vec{j}}]_i$$

Denoting by $\tilde{\tau}$ the vector with elements $\tau_k$, we can left-multiply by $I - U_1$ to obtain:

$$\tilde{\tau} = \delta^{\vec{j}} + U_1\tilde{\tau} = \delta^{\vec{j}} + U'\tilde{\tau} - U_0\tilde{\tau},$$

and we note that $U_0\tilde{\tau} = \lim_{r\to\infty} U_0(1 + U' + U'^2 + \cdots + U'^{(r-1)})\delta^{\vec{j}} = 0$ by Lemma 7 to deduce $\tilde{\tau} = \delta^{\vec{j}} + U'\tilde{\tau}$. Finally, we left-multiply by $U$ to give:

$$\begin{aligned}
L^{(\infty)}_{i\vec{j}}(s) &= \sum_j U_{ij}[\delta^{\vec{j}}]_j + \sum_j \sum_k U_{ij}U'_{jk}\tau_k = \sum_{j\in\vec{j}} U_{ij} + \sum_{j\notin\vec{j}} \sum_k U_{ij}U_{jk}\tau_k \\
L^{(\infty)}_{i\vec{j}}(s) &= \sum_{j\in\vec{j}} U_{ij} + \sum_{j\notin\vec{j}} U_{ij} L^{(\infty)}_{j\vec{j}}(s)
\end{aligned}$$

We have shown that the iterative scheme of Equation (16) converges to a finite quantity $L^{(\infty)}_{i\vec{j}}(s)$ satisfying Equation (8), and thus $L^{(\infty)}_{i\vec{j}} = L_{i\vec{j}}$, as required.

## 4 Distribution Representation and Laplace Inversion

The key to practical analysis of semi-Markov processes lies in the efficient representation of their general distributions. Without care the structural complexity of the SMP can be recreated within the representation of the distribution functions. This is especially true with the manipulations performed in the iterative passage-time calculation of Section 2.3.

Many techniques have been used for representing arbitrary distributions – two of the most popular being *phase-type distributions* (for example [10]) and *vector-of-moments* methods [11]. These methods suffer from, respectively, exploding representation size under composition, and containing insufficient information to produce accurate answers after large amounts of composition.

As all our distribution manipulations take place in Laplace-space, we link our distribution representation to the Laplace inversion technique that we ultimately use. Our tool supports two Laplace transform inversion algorithms, which are briefly outlined below: the Euler technique [12] and the Laguerre method [13] with modifications summarised in [2].

Both algorithms work on the same general principle of sampling the transform function $L(s)$ at $n$ points, $s_1, s_2, \ldots, s_n$ and generating values of $f(t)$ at $m$ user-specified $t$-points $t_1, t_2, \ldots, t_m$. In the Euler inversion case $n = km$, where $k$ can vary between 15 and 50, depending on the accuracy of the inversion required. In the modified Laguerre case, $n = 400$ and, crucially, is independent of $m$ (see Section 4.2).

The process of selecting a Laplace transform inversion algorithm is discussed later; however, whichever is chosen, it is important to note that calculating $s_i, 1 \leq i \leq n$ and storing all our distribution transform functions, sampled at these points, will be sufficient to provide a complete inversion. Key to this is the fact that matrix element operations, of the type performed in Equation (19), (which in time-space would be convolution and weighted sum) do not require any adjustment to the array of domain $s$-points required. In the case of a convolution, for instance, if $L_1(s)$ and $L_2(s)$ are stored in the form $\{(s_i, L_j(s_i)) \ : \ 1 \leq i \leq n\}$, for $j = 1, 2$, then the Laplace transform of the convolution, $L_1(s)L_2(s)$, can be stored using the same size array and using the same list of domain $s$-values, $\{(s_i, L_1(s_i)L_2(s_i)) \ : \ 1 \leq i \leq n\}$.

Storing our distribution functions in this way has three main advantages.

Firstly, the function has constant storage space, independent of the distribution type. Secondly, each distribution has, therefore, the same constant storage requirement even after composition with other distributions. Finally, the function has sufficient information about a distribution to determine the required passage-time (and no more).

### *4.1 Summary of Euler Inversion*

Our experience with numerical Laplace inversion algorithms has been very favourable [14]. If the function itself has large discontinuities, then occasionally small *Gibbs' phenomena* can be observed close to the discontinuity. However, for functions with discontinuous derivative, or discontinuous functions which have been convolved with *smooth* functions, numerical instabilities are not noticeable.

The Euler method is based on the Bromwich contour inversion integral, expressing the function $f(t)$ in terms of its Laplace transform $L(s)$. Making the contour a vertical line $s = a$ such that $L(s)$ has no singularities on or to the right of it gives:

$$f(t) = \frac{2\,\mathrm{e}^{at}}{\pi} \int_0^\infty \mathrm{Re}(L(a + \mathrm{i}u)) \cos(ut) \; \mathrm{d}u \tag{27}$$

This integral can be evaluated numerically using the trapezoidal rule with step-size $h = \pi/2t$ and $a = A/2t$ (where $A$ is a constant that controls the discretisation error), which results in the nearly alternating series:

$$f(t) \approx f_h(t) = \frac{\mathrm{e}^{A/2}}{2t} \mathrm{Re}(L(A/2t)) + \frac{\mathrm{e}^{A/2}}{2t} \sum_{k=1}^{\infty} (-1)^k \, \mathrm{Re}\left( L\left( \frac{A + 2k\pi\mathrm{i}}{2t} \right) \right) \tag{28}$$

We note that for each argument $s_i$ of $L$, $\mathrm{Re}(s_i) = A/2t > 0$, as required for convergence of Equation (16).

Euler summation is employed to accelerate the convergence of the alternating series infinite sum, so we calculate the sum of the first $n$ terms explicitly and use Euler summation to calculate the next $m$. To give an accuracy of $10^{-8}$ we set $A = 19.1$, $n = 20$ and $m = 12$ (compared with $A = 19.1$, $n = 15$ and $m = 11$ in [12]).

### 4.2 Summary of Laguerre Inversion

The Laguerre method [13] makes use of the Laguerre series representation:

$$f(t) = \sum_{n=0}^{\infty} q_n l_n(t) \qquad : t \geq 0 \tag{29}$$

where the Laguerre polynomials $l_n$ are given by:

$$l_n(t) = \left(\frac{2n-1-t}{n}\right) l_{n-1}(t) - \left(\frac{n-1}{n}\right) l_{n-2}(t) \tag{30}$$

starting with $l_0 = \mathrm{e}^{t/2}$ and $l_1 = (1-t)\,\mathrm{e}^{t/2}$, and:

$$q_n = \frac{1}{2\pi r^n} \int_0^{\pi} Q(r\,\mathrm{e}^{\mathrm{i}u})\,\mathrm{e}^{-iru}\ \mathrm{d}u \tag{31}$$

where $r = (0.1)^{4/n}$ and $Q(z) = (1-z)^{-1} L((1+z)/2(1-z))$.

The integral in Equation (31) can be approximated numerically by the trapezoidal rule, giving:

$$q_n \approx \bar{q}_n = \frac{1}{2nr^n} \left( Q(r) + (-1)^n Q(-r) + 2 \sum_{j=1}^{n-1} (-1)^j \,\mathrm{Re}\left(Q(r\,\mathrm{e}^{\pi j \mathrm{i}/n})\right) \right) \tag{32}$$

Again, we note that, since $r < 1$, all arguments passed to the Laplace function, $L$, have positive real part as required for convergence of Equation (16).

As described in [2], the Laguerre method can be modified by noting that the Laguerre coefficients $q_n$ are independent of $t$. This means that if the number of trapezoids used in the evaluation of $q_n$ is fixed to be the same for every $q_n$ (rather than depending on the value of $n$), values of $Q(z)$ (and hence $L(s)$) can be reused after they have been computed. Typically, we set $n = 200$. In order to achieve this, however, the scaling method described in [13] must be used to ensure that the Laguerre coefficients have decayed to (near) 0 by $n = 200$. If this can be accomplished, the inversion of a passage-time density for any number of $t$-values can be achieved at the fixed cost of calculating 400 truncated summations of the type shown in Equation (19). This is in contrast to the Euler method, where the number of truncated summations required is a function of the number of points at which the value of $f(t)$ is required.

## 5 Convergence Examples

We present two examples of our passage-time approximation in action. The first is a purely Markovian system, for compatibility with previous Markovian techniques. The second is a fully semi-Markovian example. In both cases, we present an analytically calculated result (using Equation (8)) with which to compare the approximations. Where we have needed to use a Laplace inversion algorithm, we have used the Euler method from Section 4.1.

### *5.1 Markovian Example*

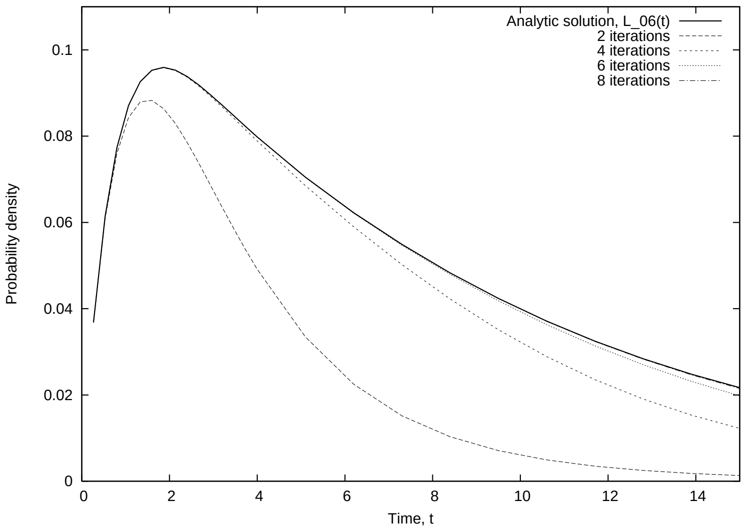


Fig. 1. Four iterative approximations of $L_{06}(t)$

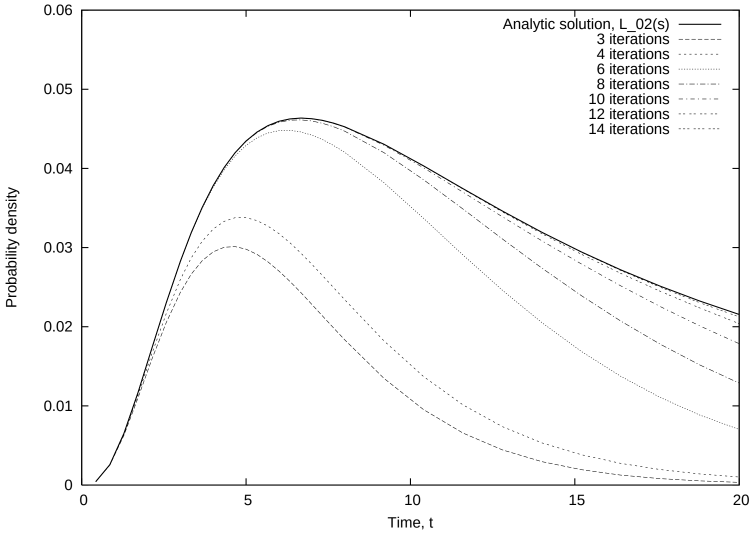


Fig. 2. Seven iterative approximations of $L_{02}(t)$

We set out a Markov system as a simple semi-Markov model with DTMC transition matrix, $D$, and rate matrix, $S$. We aim to find the passage-times

$L_{06}(t)$, from state 0 to state 6, and $L_{02}(t)$, from state 0 to state 2.

$$D = \begin{pmatrix} 0 & p_1 & 0 & 1-p_1 & 0 & 0 & 0 \\ 1-p_2 & 0 & 0 & 0 & p_2 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1-p_3 & 0 & 0 & p_3 \\ p_4 & 0 & 1-p_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \quad S = \begin{pmatrix} 0 & s_{01} & 0 & s_{03} & 0 & 0 & 0 \\ s_{10} & 0 & 0 & 0 & s_{14} & 0 & 0 \\ s_{20} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & s_{36} \\ 0 & 0 & 0 & s_{43} & 0 & 0 & s_{46} \\ s_{50} & 0 & s_{52} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & s_{65} & 0 \end{pmatrix} \tag{33}$$

We use values $p_1 = 2/3, p_2 = 1/2, p_3 = 1/4, p_4 = 1/3, s_{01} = 1/4, s_{03} = 1, s_{10} = 1/3, s_{14} = 2/3, s_{20} = 1, s_{36} = 1/2, s_{43} = 1, s_{46} = 4/5, s_{50} = 1/5, s_{52} = 2/5, s_{65} = 1$. $S$ is a matrix of exponential rates, so the distribution matrix $H_{ij}(t) = 1 - \exp(-s_{ij}t)$.

Figure 1 shows the passage-time results for $L_{06}(t)$. For such a small passage only 8 iterations are required for a good approximation to the analytic result to be achieved.

By comparison, Figure 2 shows the passage-time results for $L_{02}(t)$. As more of the system is exposed to the passage calculation, it takes more iterations (14 in total) to give good visual convergence.

### 5.2 Semi-Markov Example

Next, we look at a semi-Markov example of a system which employs different types of distribution (Figure 3).

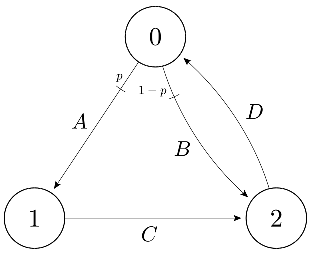


Fig. 3. A 3 state semi-Markov system with generally distributed sojourn times.

The distributions used in the transition system of Figure 3 are given by: $A \sim \Gamma(\alpha, \lambda)$, $B \sim \text{Hyper}\left(\frac{2}{3}, \frac{1}{3}; \lambda, \mu\right)$, $C \sim \text{U}\left[\frac{1}{\lambda}, \frac{1}{\mu}\right]$, $D \sim \delta\left[\frac{1}{\mu}\right]$, where:

$$
\begin{aligned}
r^*_{01}(s) &= p\left(\frac{\lambda}{\lambda+s}\right)^{\alpha} \\
r^*_{02}(s) &= (1-p)\left[\frac{2}{3}\frac{\lambda}{\lambda+s}+\frac{1}{3}\frac{\mu}{\mu+s}\right] \\
r^*_{12}(s) &= \frac{\lambda\mu}{(\lambda-\mu)s}\left[\mathrm{e}^{-s/\lambda}-\mathrm{e}^{-s/\mu}\right] \\
r^*_{20}(s) &= \mathrm{e}^{-s/\mu}
\end{aligned}
$$

The following parameter values are used: $\alpha = 2.5$, $\lambda = 1.5$, $\mu = 1.23$ and $p = 0.3$. Figure 4 shows the passage-time density between states 2 and 1 as calculated using a standard linear equation solver. Again, the iterative solutions are plotted on the same diagram and it can be seen that they approach the analytic solution with relatively few iterations. Note that even though the distribution has a discontinuous derivative (at time $t = 1/\mu$), the Euler Laplace inversion algorithm has no numerical difficulty inverting the function.

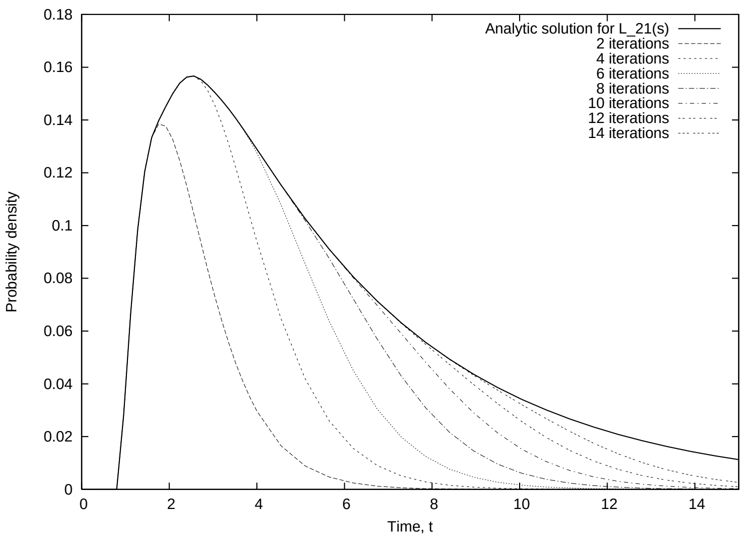


Fig. 4. Seven iterative approximations of $L_{21}(s)$, plotted with the analytic solution.

As we will see, for larger systems of millions of states, as generated by higher-level formalisms, we typically need only a few hundred iterations to achieve convergence.

## 6 System-size Passage-time Analysis

### 6.1 A Distributed Web-server Cluster Model

Having tested the iterative passage-time analysis technique on very small models, we demonstrate that it is scalable to large models by summarising analysis of a 15 million state SMP model of a distributed web server [5], shown in Figure 5.

The model is specified in a semi-Markov stochastic Petri net (SM-SPN) formalism [15] using an extension of the DNAmaca Markov chain modelling language [16]. Generally distributed transitions, if simultaneously enabled, are selected by probabilistic choice; in this way, we are guaranteed an underlying semi-Markov state space. Concurrent Markovian execution can, nevertheless, be implemented by using the `markov(s)` pragma. The distributions are specified directly as Laplace transforms with certain macros provided for popular distributions (e.g. uniform, gamma, deterministic) and can be made marking dependent by use of the $m(p_i)$ function (which returns the current number of tokens at place $p_i$). Support for inhibiting transitions is also provided.

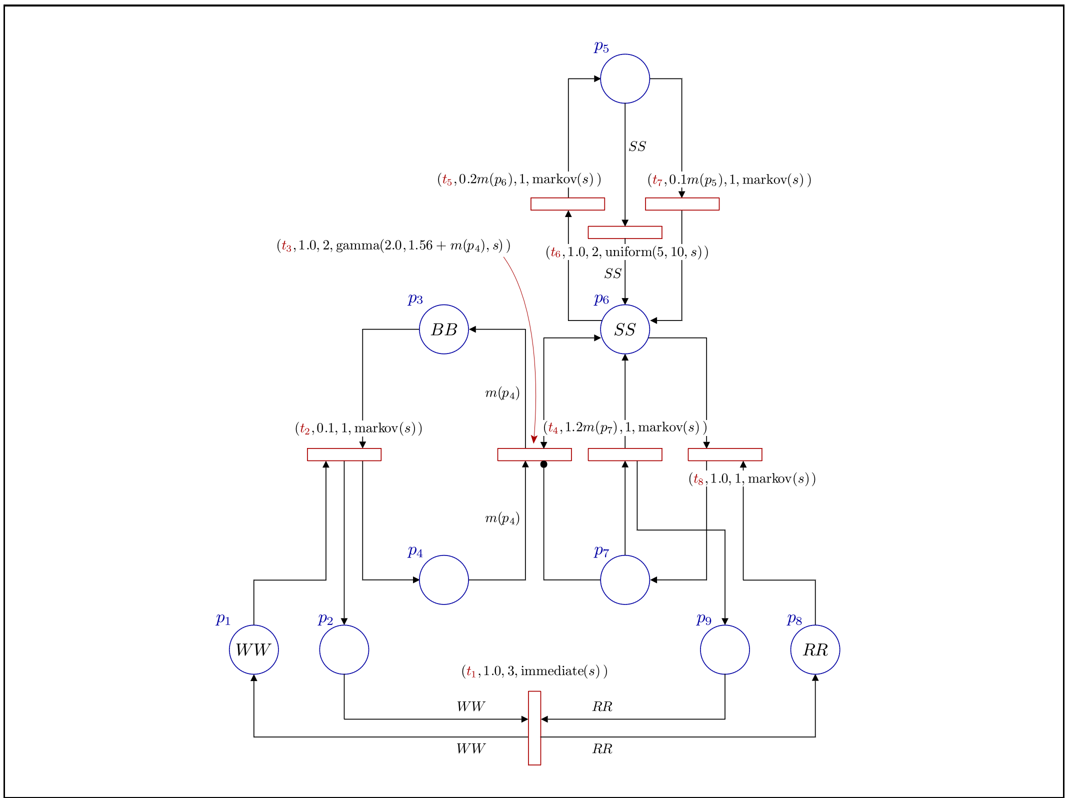


Fig. 5. A semi-Markov Petri Net of a parallel web server

Figure 5 represents a web server with $RR$ clients, $WW$ web content authors,

$SS$ parallel web servers and a write-buffer of $BB$ in size. A write is buffered by a token moving from $p_1$ to $p_2$ and the writes in the buffer, $p_4$, are not actually executed until all available servers have finished their respective read lookups (i.e. the alive servers are all in place $p_6$). Servers themselves can fail to and recover from place $p_5$, and while there they become unavailable for reads or writes. A read takes place when a token moves from place $p_8$ to $p_9$.

In the following, we consider the rate of convergence of the iterative passage-time algorithm and the extraction of passage-time densities and cumulative distribution functions, for the example semi-Markov systems.

### *6.2 Convergence of the Iterative Passage-time Algorithm*

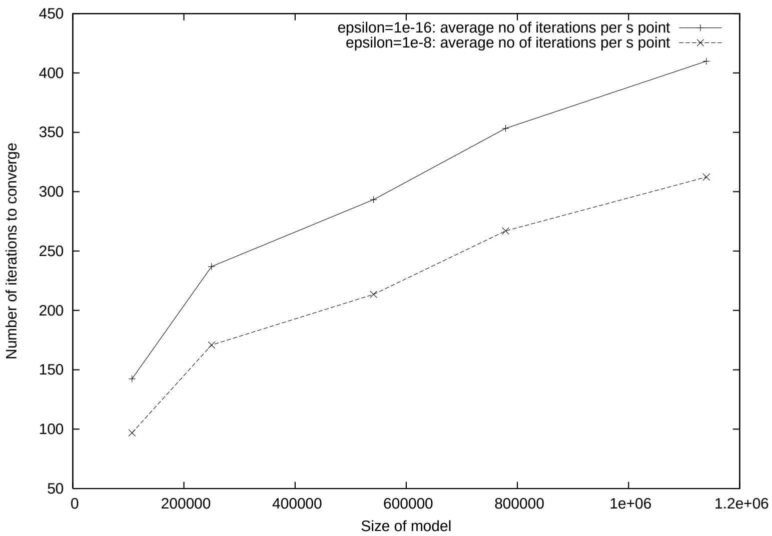


Fig. 6. Average number of iterations to converge per $s$ point for two different values of $\varepsilon$ over a range of model sizes.

Our iterative algorithm terminates when two successive iterates are less than $\varepsilon$ apart, for some suitably small value of $\varepsilon$. Figure 6 shows the average number of iterations the algorithm takes to converge per $s$-point for both models for two different values of $\varepsilon$: $10^{-8}$ and $10^{-16}$.

The number of iterations required for convergence as the model size grows is sub-linear; that is, as the model size doubles the number of iterations less than doubles. This suggests the algorithm has good scalability properties for increasing model size, although this will be highly model dependent (according to the sparsity of the underlying transition matrices).

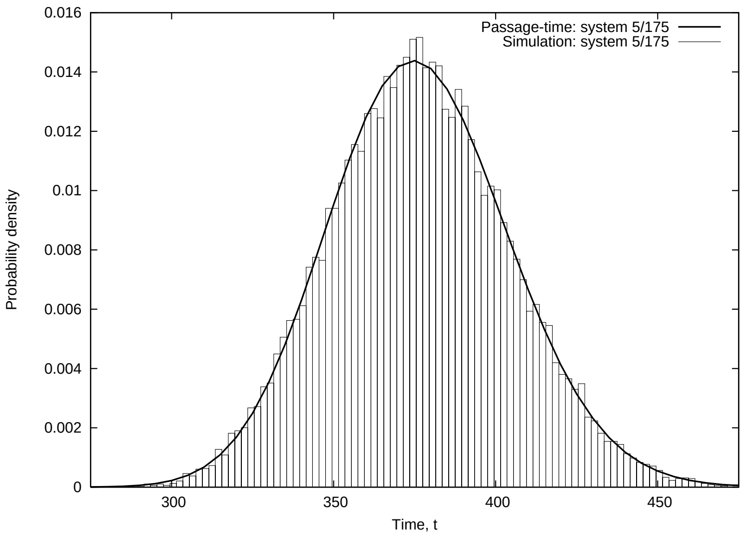


Fig. 7. Analytic and simulated density for the time taken to process 100 reads and 50 page updates in the web-server model (15.4 million states).

### *6.3 Passage-time Densities*

Here, we display passage-time densities produced by our iterative passage-time algorithm and also those produced by simulation to validate those results.

Figure 7 shows the density of the time taken to perform 100 reads and 50 page updates in the web server model. Calculation of the 35 $t$-points plotted required 2 days, 17 hours and 30 minutes using 64 slave processors [5]. Our algorithm evaluated $L_{\vec{i}\vec{j}}(s)$ at 1155 $s$-points, each of which involved manipulating sparse matrices of rank $15,445,919$. The analytical curve is validated against a simulation of 1 billion transition firings. We observe excellent agreement.

## 7 Conclusion

In this paper, we proved the convergence of an iterative algorithm for deriving passage-time densities. This algorithm is defined over semi-Markov processes, which are more descriptive than purely Markovian models. As such, a practically applicable algorithm with guaranteed convergence properties is a useful extension to the field.

After proving convergence, we demonstrated the convergence of the algorithm for some small sample systems. Finally, we showed the algorithm in action on a practical model of a web-server cluster, a semi-Markov model of over 15 million states, and were able to show that even for such large models only a few hundred iterations are required to get satisfactory convergence. The passage-time density for the web-server cluster was also verified by simulation.

## Acknowledgements

Jeremy Bradley is supported in part by the Nuffield Foundation under grant reference NAL/00805/G. The authors would especially like to thank Nick Dingle and Will Knottenbelt for their help and advice with the semi-Markov version of the DNAmaca tool.

## References

[1] J. T. Bradley, N. J. Dingle, P. G. Harrison, W. J. Knottenbelt, Distributed computation of passage time quantiles and transient state distributions in large semi-Markov models, in: PMEO'03, Performance Modelling, Evaluation and Optimization of Parallel and Distributed Systems, IEEE Computer Society Press, Nice, 2003, p. 281.

[2] P. G. Harrison, W. J. Knottenbelt, Passage-time distributions in large Markov chains, in: M. Martonosi, E. A. de Souza e Silva (Eds.), Proc. ACM SIGMETRICS, 2002, pp. 77–85.

[3] N. J. Dingle, P. G. Harrison, W. J. Knottenbelt, Response time densities in Generalised Stochastic Petri Net models, in: WOSP 2002, Proc. 3rd International Workshop on Software and Performance, Rome, 2002, pp. 46–54.

[4] N. J. Dingle, W. J. Knottenbelt, P. G. Harrison, HYDRA: HYpergraph-based Distributed Response-time Analyser, in: H. R. Arabnia, Y. Man (Eds.), PDPTA'03, Proc. 2003 International Conference on Parallel and Distributed Processing Techniques and Applications, Vol. 1, Las Vegas, NV, 2003, pp. 215–219.

[5] J. T. Bradley, N. J. Dingle, W. J. Knottenbelt, H. J. Wilson, Hypergraph-based parallel computation of passage time densities in large semi-Markov models, Journal of Linear Algebra and Applications 386 (2004) 311–334.

[6] J. K. Muppala, K. S. Trivedi, Numerical transient analysis of finite Markovian queueing systems, in: U. N. Bhat, I. V. Basawa (Eds.), Queueing and Related Models, Oxford University Press, 1992, pp. 262–284.

[7] B. Melamed, M. Yadin, Randomization procedures in the computation of cumulative-time distributions over discrete state Markov processes, Operations Research 32 (4) (1984) 926–944.

[8] G. Bolch, S. Greiner, H. de Meer, K. S. Trivedi, Queueing Networks and Markov Chains, Wiley, 1998.

[9] R. A. Horn, C. R. Johnson, Topics in Matrix Analysis, Cambridge University Press, 1986.

[10] H. C. Bohnenkamp, B. R. Haverkort, Stochastic event structures for the decomposition of stochastic process algebra models, in: J. Hillston, M. Silva (Eds.), Process Algebra and Performance Modelling Workshop, Centro Politécnico Superior de la Universidad de Zaragoza, Prensas Universitarias de Zaragoza, 1999, pp. 25–39.

[11] S. W. M. Au-Yeung, N. J. Dingle, W. J. Knottenbelt, Efficient approximation of response time densities and quantiles in stochastic models, in: WOSP 2004, Proc. 4th International Workshop on Software and Performance, ACM, Redwood City, 2004, pp. 151–155.

[12] J. Abate, W. Whitt, Numerical inversion of Laplace transforms of probability distributions, ORSA Journal on Computing 7 (1) (1995) 36–43.

[13] J. Abate, G. L. Choudhury, W. Whitt, On the Laguerre method for numerically inverting Laplace transforms, INFORMS Journal on Computing 8 (4) (1996) 413–427.

[14] J. T. Bradley, Report on the numerical inversion of Laplace tranforms of probabilistic functions: A Haskell implementation, Tech. Rep. 2001(6), Department of Computer Science, University of Durham, South Road, Durham, DH1 3LE, UK (July 2001).

[15] J. T. Bradley, N. J. Dingle, W. J. Knottenbelt, P. G. Harrison, Performance queries on semi-Markov stochastic Petri nets with an extended Continuous Stochastic Logic, in: G. Ciardo, W. Sanders (Eds.), PNPM'03, Proc. Petri Nets and Performance Models, IEEE Computer Society, University of Illinois at Urbana-Champaign, 2003, pp. 62–71.

[16] W. J. Knottenbelt, Generalised Markovian analysis of timed transitions systems, M.Sc. thesis, University of Cape Town, South Africa (July 1996).