Computation of Absorption Probability Distributions
of continuous-time Markov Chains
using Regenerative Randomization

Angel Calderón and Juan A. Carrasco

Departament d'Enginyería Electrònica

Universitat Politécnica de Catalunya

Diagonal 647, plta. 9, 08028 Barcelona, Spain

Abstract

Randomization is a popular method for the transient solution of continuous-time Markov models. Its primary advantages over other methods (i.e., ODE solvers) are robustness and ease of implementation. It is however well-known that the performance of the method deteriorates with the "stiffness" of the model: the number of required steps to solve the model up to time t tends to Λt for $\Lambda t \to \infty$, where Λ is the maximum output rate. For measures like the unreliability Λt can be very large for the t of interest, making the randomization method very unefficient. In this paper we consider such measures and propose a new solution method called regenerative randomization *which exploits the regenerative structure of the model and can be far more efficient. Regarding the number of steps required in regenerative randomization we prove that: 1) it is smaller than the number of steps required in standard randomization when the initial distribution is concentrated in a single state, 2) for $\Lambda t \to \infty$, it is upper bounded by a function $O(\log(\Lambda t/\epsilon))$, where ϵ is the desired approximation error bound. Using a reliability example we analyze the performance and stability of the method.*

Keywords: Randomization, continuous-time Markov chains, regenerative models, reliability, transient solution.

1 Introduction

Continuous-time Markov chains (CTMC's) are often used for performance, dependability and performability modeling. The transient analysis of these models is usually significantly more costly than the steady-state analysis, and very costly in absolute terms when the CTMC model is large. This makes the development of efficient transient analysis techniques for CTMC's a research topic of great interest. Commonly used methods are ODE solvers and randomization. Good recent reviews can be found in [Rei88] and [Mal94]. The randomization method (also called uniformization) is numerically very stable and easy to

implement. It was first proposed by Jensen [Jen53] and has been applied to analyze performance models [Gra77], [Gro84], dependability models [Koh82], [Rei88], and performability models [Qur93]. The randomization method is based on the following result. Let $X = \{X(t); t \geq 0\}$ be a CTMC with state space Ω; let λ_{ij}, $i, j \in \Omega$ be the transition rates of X and let $\lambda_i = \sum_{j \in \Omega} \lambda_{ij}$, $i \in \Omega$ be the output rates of X. Consider any $\Lambda \geq \max_{i \in \Omega} \lambda_i$ (usually Λ is taken to be $\max_{i \in \Omega} \lambda_i$), and define the discrete-time Markov chain (DTMC) $Y = \{Y_k; k = 0, 1, 2, \ldots\}$ with same state space and initial distribution as X and jump probabilities $q_{ij} = \lambda_{ij}/\Lambda$, $i \neq j$, $q_{ii} = 1 - \lambda_i/\Lambda$. Let $N = \{N(t); t \geq 0\}$ be a Poisson process with arrival rate Λ. Then, $X(t) = Y_{N(t)}$. A recent, short proof of the result with very general conditions on X can be found in [Dij90]. These conditions are satisfied if, as we assume, X is finite.

The randomization equation gives immediately an esqueme for the computation of the transient probabilities of X, but it can also be exploited to compute more complex measures such as the distribution of the interval availability [Sou86], [Rub93] and the distribution of performability [Sou89]. In this paper we assume that X is a transient CTMC with state space $\Omega^t \cup \{a\}$, where a is an absorbing state and all states in Ω^t are transient, and consider the absorption probability by time t, $AP(t) = P[X(t) = a]$. The unreliability is an example of such a measure. Using the randomization equation:

$$AP(t) = \sum_{k=0}^{\infty} P[X(t) = a] \frac{(\Lambda t)^k}{k!} e^{-\Lambda t}.$$

In a practical implementation of the randomization method, the infinite series is approximated by truncating the series up to a given number of randomization steps, K, and the approximation error is bounded:

$$AP^a(t) = \sum_{k=0}^{K} P[X(t) = a] \frac{(\Lambda t)^k}{k!} e^{-\Lambda t},$$

92

$$AP(t) - AP^a(t) \le \sum_{k=K+1}^{\infty} \frac{(\Lambda t)^k}{k!} e^{-\Lambda t} = AP^\epsilon(t),$$

where $AP^a(t)$ is the approximation and $AP^\epsilon(t)$ upper bounds the approximation error. Taking a large enough M, $AP^\epsilon(t)$ can be computed using a finite number of terms with arbitrary accuracy as:

$$AP^\epsilon(t) \approx \sum_{k=K+1}^{M-1} \frac{(\Lambda t)^k}{k!} e^{-\Lambda t} + \frac{1}{1 - \frac{\Lambda t}{M}} \frac{(\Lambda t)^M}{M!} e^{-\Lambda t}.$$

Using the well known result that $N(t)$ has an asymptotic normal distribution with mean and variance Λt, it is easy to show that, for large Λt, the number of steps required in the randomization method is $\approx \Lambda t$, almost independent of the required approximation error. Often we are interested in solving the model for values of t for which Λt is large, and in such cases randomization is unefficient. Consider for instance a reliability model of a repairable fault-tolerant system. For such a model Λ is of the order of the maximum repair rate while the t of interest can be large. With a model including hot restarts, Λ can be as large as $1 \, \text{min}^{-1}$, while the t of interest could be 1 year, yielding $\Lambda t = 525,600$. Using the randomization method we are bound to solve Y for more than half a million steps! If X is large this may be *extremely* expensive.

Several approaches have been proposed to alleviate the problem. Miller has used selective randomization to solve reliability models with detailed representation of error handling activities [Mil83]. Reibman and Trivedi [Rei88] have proposed a more general approach based on the multistep concept which works very well when the transition probability matrix is dense. Denoting by Q the transition probability matrix of Y and by $\pi(k)$ the probability distribution row vector of Y at step k, we have $\pi(k + S) = \pi(k)Q^S$, where S is the length of the multistep. Then, computing Q^S explicitely, the number of vector-matrix multiplications can be reduced significantly exploiting the fact that for large Λt the number of $\pi(k)$'s with significant contributions to the randomization formula is of the order of $\sqrt{\Lambda t}$. However, if Q is sparse Q^S can be much denser than Q, and the number of floating point operations can still be large. Adaptive uniformization [Moo93] is a recent method in which the randomization rate is adapted depending on the states in which the randomized DTMC can be in at a given step. For some models, adaptive randomization can be faster than the standard method. In addition, it can be used to solve models with infinite state spaces and not uniformly bounded transition rates. Another recent proposal to speed up the randomization method is steady-state detection [Mal94]. This technique will be efficient if the steady-state is reached fast, but for for many models (for instance, reliability models of repairable fault-tolerant systems) the steady state is reached very slowly and the technique will not help. More recently, we have proposed [Car94] a method

called *regenerative randomization* for the computation of measures like the steady-state availability and the expected interval availability which can be expressed as the transient reward rate or mean interval reward rate of rewarded regenerative CTMC models. Unlike standard randomization, our method is measure specific. In this paper we extend the regenerative randomization method to the computation of $AP(t)$ and investigate its theoretical properties. We show that when the initial distribution of X is concentrated in a single state (a very common case) regenerative randomization requires at most the same number of steps than standard randomization. Furthermore, for large Λt, the number of steps required to achieve an approximation error $\le \epsilon$ is $O(\log(\Lambda t/\epsilon))$. In Section 2 we derive the regenerative randomization method for $AP(t)$. Section 3 includes the proofs of the theoretical properties of the method. Section 4 analyzes the method compared to standard randomization using a reliability model. Section 5 concludes the paper.

2 Regenerative randomization

Regenerative randomization requires the selection of a transient state $u \in \Omega^t$. Let $\alpha_i = P[X(0) = i] = P[Y_0 = i]$, $i \in \Omega$, and assume $P[X(0) = a] = 0$. Let $\Omega_u^t = \Omega^t - \{u\}$. Let $Z = \{Z_k; k = 0, 1, 2, \ldots\}$ be the transient DTMC following Y from u till reentry in u. The state space of Z is $\Omega^t \cup \{a, b\}$, where both a and b are absorbing states. $Z_0 = u$ and Z enters b when Y enters u. The transition probabilities of Z are:

$$
\begin{aligned}
P[Z_{k+1} = j | Z_k = i] &= P[Y_{k+1} = j | Y_k = i], \\
P[Z_{k+1} = b | Z_k = i] &= P[Y_{k+1} = u | Y_k = i], \\
P[Z_{k+1} = a | Z_k = a] &= P[Y_{k+1} = b | Y_k = b] = 1,
\end{aligned}
$$

with $i \in \Omega^t$, $j \in \Omega_u^t \cup a$.

Let $Z' = \{Z_k'; k = 0, 1, 2, \ldots\}$ be the transient DTMC with state space $\Omega_u^t \cup \{a, b\}$, where both a and b are absorbing states, following Y except that Z' remains in b if Y has been in u. The initial distribution of Z' is $P[Z_0' = i] = \alpha_i$, $i \in \Omega_u^t$, $P[Z_0' = b] = \alpha_u$, and its transition probabilities are:

$$
\begin{aligned}
P[Z_{k+1}' = j | Z_k' = i] &= P[Y_{k+1} = j | Y_k = i], \\
P[Z_{k+1}' = b | Z_k' = i] &= P[Y_{k+1} = u | Y_k = i], \\
P[Z_{k+1}' = a | Z_k' = a] &= P[Y_{k+1} = b | Y_k = b] = 1,
\end{aligned}
$$

with $i \in \Omega_u^t$, $j \in \Omega_u^t \cup \{a\}$.

In the following let $\pi_i(k) = P[Z_k = i]$, $\pi_i'(k) = P[Z_k' = i]$.

The regenerative randomization formula is obtained considering a discrete-time stochastic process $W = \{W_k; k = 0, 1, 2, \ldots\}$, with state space $\{0, 1, 2, \ldots\} \cup \{0', 1', 2', \ldots\} \cup \{a\}$, defined from Y as follows:

$$
W_k = \begin{cases}
l & \text{if } Y_{k-l} = u \wedge Y_m \in \Omega_u^t, k - l < m \le k \\
k' & \text{if } Y_m \in \Omega_u^t, 0 \le m \le k \\
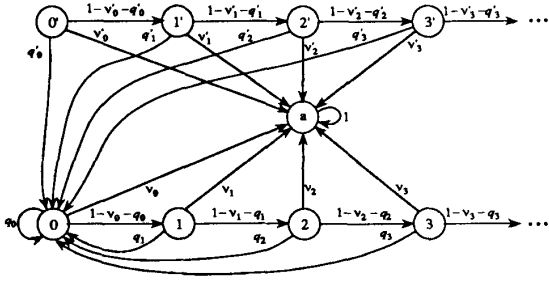a & \text{if } Y_k = a
\end{cases}
$$

Figure 1: State transition diagram of the DTMC W.

Informally, $W_k = l$ if Y has visited u by time k, has not been absorbed, and has made its last visit to u l steps before the current step, k; $W_k = k'$ if Y has not visited u by time k and has not been absorbed; and $W_k = a$ if Y has been absorbed.

Proposition 1 *Let* $a(k) = \sum_{i \in \Omega^t} \pi_i(k)$, $a'(k) = \sum_{i \in \Omega^t_u} \pi'_i(k)$, $v_k = (\pi_a(k+1) - \pi_a(k))/a(k)$, $q_k = (\pi_b(k+1) - \pi_b(k))/a(k)$, $v'_k = (\pi'_a(k+1) - \pi'_a(k))/a'(k)$, $q'_k = (\pi'_b(k+1) - \pi'_b(k))/a'(k)$. *Then, W is an homogeneous discrete-time Markov chain with initial probability distribution* $P[W_0 = 0] = \alpha_u$, $P[W_0 = 0'] = 1 - \alpha_u$, $P[W_0 = i] = 0$, $i \notin \{0, 0'\}$ *and the state transition diagram shown in Figure 1.*

Proof The initial distribution of W follows inmediately from the definition. It is also clear that: 1) $W_k \in \{0, 1, \ldots, k, k', a\}$, 2) from a state l, W can only jump to states a, 0, or $l+1$, and 3) from a state k', W can only jump to states a, 0, or $(k+1)'$. We will compute all transition probabilities at a given step k and show that they only depend on the "from" state.

Case a $(W_k = l, 0 \leq l \leq k)$:

$$P[W_{k+1} = a | W_k = l] = \frac{P[W_k = l \wedge W_{k+1} = a]}{P[W_k = l]}$$

$$= \frac{P[Y_{k-l} = u \wedge Y_m \in \Omega^t_u, k-l < m \leq k \wedge Y_{k+1} = a]}{P[Y_{k-l} = u \wedge Y_m \in \Omega^t_u, k-l < m \leq k]}$$

$$= \frac{P[Y_m \in \Omega^t_u, k-l < m \leq k \wedge Y_{k+1} = a | Y_{k-l} = u]}{P[Y_m \in \Omega^t_u, k-l < m \leq k | Y_{k-l} = u]}$$

$$= \frac{P[Z_l \in \Omega^t \wedge Z_{l+1} = a]}{P[Z_l \in \Omega^t]} = \frac{P[Z_{l+1} = a] - P[Z_l = a]}{P[Z_l \in \Omega^t]}$$

$$= \frac{\pi_a(l+1) - \pi_a(l)}{a(l)} = v_l,$$

$$P[W_{k+1} = 0 | W_k = l] = \frac{P[W_k = l \wedge W_{k+1} = 0]}{P[W_k = l]}$$

$$= \frac{P[Y_{k-l} = u \wedge Y_m \in \Omega^t_u, k-l < m \leq k \wedge Y_{k+1} = u]}{P[Y_{k-l} = u \wedge Y_m \in \Omega^t_u, k-l < m \leq k]}$$

$$= \frac{P[Y_m \in \Omega^t_u, k-l < m \leq k \wedge Y_{k+1} = u | Y_{k-l} = u]}{P[Y_m \in \Omega^t_u, k-l < m \leq k | Y_{k-l} = u]}$$

$$= \frac{P[Z_l \in \Omega^t \wedge Z_{l+1} = b]}{P[Z_l \in \Omega^t]} = \frac{P[Z_{l+1} = b] - P[Z_l = b]}{P[Z_l \in \Omega^t]}$$

$$= \frac{\pi_b(l+1) - \pi_b(l)}{a(l)} = q_l,$$

$$P[W_{k+1} = l + 1 | W_k = l]$$
$$= 1 - P[W_{k+1} = a | W_k = l] - P[W_{k+1} = 0 | W_k = l]$$
$$= 1 - v_l - q_l.$$

Case b $(W_k = k')$:

$$P[W_{k+1} = a | W_k = k'] = \frac{P[W_k = k' \wedge W_{k+1} = a]}{P[W_k = k']}$$

$$= \frac{P[Y_m \in \Omega^t_u, 0 \leq m \leq k \wedge Y_{k+1} = a]}{P[Y_m \in \Omega^t_u, 0 \leq m \leq k]}$$

$$= \frac{P[Z'_k \in \Omega^t_u \wedge Z'_{k+1} = a]}{P[Z'_k \in \Omega^t_u]}$$

$$= \frac{P[Z'_{k+1} = a] - P[Z'_k = a]}{P[Z'_k \in \Omega^t_u]}$$

$$= \frac{\pi'_a(k+1) - \pi'_a(k)}{a'(k)} = v'_k,$$

$$P[W_{k+1} = 0 | W_k = k'] = \frac{P[W_k = k' \wedge W_{k+1} = 0]}{P[W_k = k']}$$

$$= \frac{P[Y_m \in \Omega^t_u, 0 \leq m \leq k \wedge Y_{k+1} = u]}{P[Y_m \in \Omega^t_u, 0 \leq m \leq k]}$$

$$= \frac{P[Z'_k \in \Omega^t_u \wedge Z'_{k+1} = b]}{P[Z'_k \in \Omega^t_u]}$$

$$= \frac{P[Z'_{k+1} = b] - P[Z'_k = b]}{P[Z'_k \in \Omega^t_u]}$$

$$= \frac{\pi'_b(k+1) - \pi'_b(k)}{a'(k)} = q'_k,$$

$$P[W_{k+1} = (k+1)' | W_k = k']$$
$$= 1 - P[W_{k+1} = a | W_k = k'] - P[W_{k+1} = 0 | W_k = k']$$
$$= 1 - v'_k - q'_k \; \bigcirc .$$

The regenerative randomization formula can now be obtained refining the state description of Y according to W and derandomizing W into a CTMC V. The result is expressed by the following theorem.

Theorem 1 *Let* $V = \{V(t); t \geq 0\}$ *be the CTMC with state space* $\{0, 1, \ldots\} \cup \{0', 1', \ldots\} \cup \{a\}$, *initial*
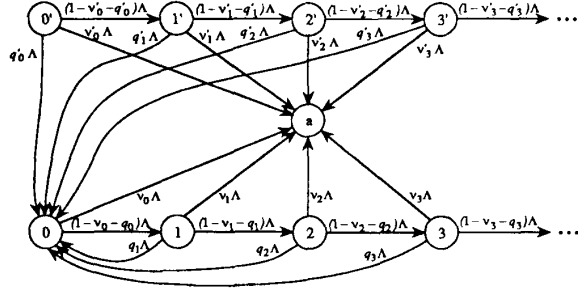
94

Figure 2: State transition diagram of the CTMC V.

Figure 3: State transition diagram of the CTMC $V_{K,L}$

distribution $P[V(0) = 0] = \alpha_u$, $P[V(0) = 0'] = 1 - \alpha_u$, $P[V(0) = i] = 0$, $i \notin \{0, 0'\}$, *and the state transition diagram shown in Figure 2. Then,* $\mathrm{AP}(t) = P[X(t) = a] = P[V(t) = a]$.

Proof Note that $W_k = a$ if and only if $Y_k = a$. Also, $V(t) = W_{N(t)}$. Then:

$$P[X(t) = a] = \sum_{k=0}^{\infty} P[Y_k = a]P[N(t) = k]$$

$$= \sum_{k=0}^{\infty} P[W_k = a]P[N(t) = k] = P[V(t) = a] \bigcirc .$$

The infinite summations of the exact regenerative randomization formula will be truncated assuming that Z has been solved up to step K and Z' has been solved up to step L. The parameters K, L control the computational effort and the approximation error. This is done considering the CTMC $V_{K,L} = \{V_{K,L}(t); t \geq 0\}$ with state space $\{0, 1, \ldots, K\} \cup \{0', 1', \ldots, L'\} \cup \{a, b\}$ defined as follows. $V_{K,L}(t) = V(t)$ if $V(\tau) \in \{0', 1', \ldots, (L-1)'\} \cup \{0, 1, \ldots, K-1\} \cup \{a\}$ for $\tau \in [0, t]$, or $V(t) = L'$ and L' has been entered only once in $[0, t]$ or $V(t) = K$ and K has been entered only once in $[0, t]$; and $V_{K,L}(t) = b$ otherwise.

The state transition diagram of $V_{K,L}$ is given in Figure 3. The absorbing state b represents the behavior of V non captured by the states $\{0, 1, \ldots, K\} \cup \{0', 1' \ldots, L'\} \cup \{a\}$ of $V_{K,L}$. Clearly, from the definition of $V_{K,L}$:

$$P[V_{K,L}(t) = i] \leq P[V(t) = i],$$

$$i \in \{0, 1, \ldots, K\} \cup \{0', 1' \ldots, L'\} \cup \{a\}.$$

Then, a lower bound to $AP(t) = P[X(t) = a] = P[V(t) = a]$ is given by:
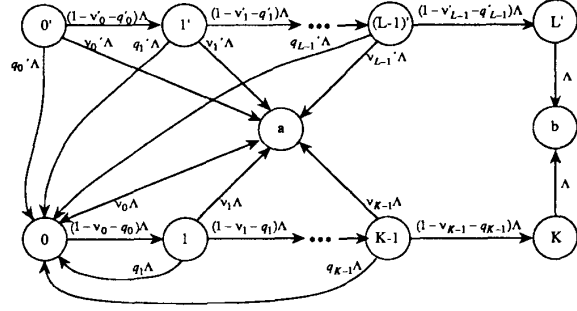
$$AP^a(t) = P[V_{K,L}(t) = a], \quad (1)$$

and the approximation error can be bounded as follows:

$$AP(t) - AP^a(t) = P[V(t) = a] - P[V_{K,L}(t) = a]$$
$$\leq P[V_{K,L}(t) = b] = AP^\epsilon(t) \quad (2)$$

The $V_{K,L}$ model for the transient probabilities of the states a, b can be solved in the Laplace domain (see [Cal94] for details). Then, using (1),(2) and antitransforming analytically whenever possible:

$$AP^a(t) = \sum_{k=0}^{L-1} v'_k a'(k) \sum_{l=k+1}^{\infty} \frac{(\Lambda t)^l}{l!} e^{-\Lambda t}$$

$$+ \mathcal{L}^{-1} \left\{ \frac{\tilde{p}_0(s)}{s} \sum_{k=0}^{K-1} v_k a(k) \frac{\Lambda^{k+1}}{(s+\Lambda)^k} \right\} \quad (3)$$

$$AP^\epsilon(t) = a'(L) \sum_{k=L+1}^{\infty} \frac{(\Lambda t)^k}{k!} e^{-\Lambda t}$$

$$+ a(K)\Lambda \mathcal{L}^{-1} \left\{ \frac{\tilde{p}_0(s)}{s} \left(\frac{\Lambda}{s+\Lambda} \right)^K \right\} \quad (4)$$

$$\tilde{p}_0(s) = \frac{N(s)}{D(s)}, \quad (5)$$

with

$$N(s) = 1 - \sum_{k=0}^{L-1} a'(k)(s + v'_k \Lambda) \frac{\Lambda^k}{(s+\Lambda)^{k+1}}$$

$$- a'(L) \left(\frac{\Lambda}{s+\Lambda} \right)^L$$

$$D(s) = \sum_{k=0}^{K-1} a(k)(s + v_k \Lambda) \left(\frac{\Lambda}{s+\Lambda} \right)^k$$

$$+ a(K)\Lambda \left(\frac{\Lambda}{s+\Lambda} \right)^{K-1}$$

The infinite series in (3), (4) can be approximated with arbitrary accuracy taking a large enough $M > L + 1$ as:

$$\sum_{l=k+1}^{\infty} \frac{(\Lambda t)^l}{l!} e^{-\Lambda t} \approx \sum_{l=k+1}^{M-1} \frac{(\Lambda t)^l}{l!} e^{-\Lambda t} + \delta(M),$$

$$\sum_{k=L+1}^{\infty} \frac{(\Lambda t)^k}{k!} e^{-\Lambda t} \approx \sum_{k=L+1}^{M-1} \frac{(\Lambda t)^k}{k!} e^{-\Lambda t} + \delta(M),$$

$$\delta(M) = \frac{1}{1 - \dfrac{\Lambda t}{M}} \frac{(\Lambda t)^M}{M!} e^{-\Lambda t}.$$

where the $\delta(M)$ terms upper bound the chopped summations.

Regenerative randomization involves in general two truncation parameters. This opens the issue of optimizing the distribution of the total number of steps. We adopt the optimization method which has worked succesfully for other measures [Car94]. In this method, $AP^{\epsilon}(t)$ is split into two terms; the first, $AP_1^{\epsilon}(t)$, accounts for the probability of b in $V_{K,L}$ resulting from entries from state L'; the second, $AP_1^{\epsilon}(t)$, accounts for the entries in b from K. These terms are:

$$AP_1^{\epsilon}(t) = a'(L) \sum_{k=L+1}^{\infty} \frac{(\Lambda t)^k}{k!} e^{-\Lambda t},$$

$$AP_2^{\epsilon}(t) = a(K) \Lambda \mathcal{L}^{-1} \left\{ \frac{\tilde{p}_0(s)}{s} \left(\frac{\Lambda}{s + \Lambda} \right)^K \right\}.$$

$AP_1^{\epsilon}(t)$ can be reduced increasing L; similarly, $AP_2^{\epsilon}(t)$ can be reduced increasing K. Then, starting with the distribution $(K = 1, L = 1)$ we increment K if $AP_2^{\epsilon}(t) \geq AP_1^{\epsilon}(t)$, and increment L otherwise. The step is repeated at each (K, L) pair obtained until $AP^{\epsilon}(t)$ satisfies the accuracy requirements. Using this strategy the contributions to the approximation error bound tend to equalize. Then, at worst, the total number of steps would be close to the minimum (with optimum distribution) number of steps required to achieve half the approximation error bound. Since the dependence of the number of steps on the imposed $AP^{\epsilon}(t)$ is typically smooth, this guarantees in practice a behavior very close to the optimum.

3 Theoretical properties

We start considering the case $P[X(0) = u] = 1$. In this case $v'_k = q'_k = a'(k) = 0$ and the equations (3-5) are simplified, requiring only the transient solution of the DTMC Z at the steps $k = 0, 1, \ldots, K$. We have the following result:

Theorem 2 *When $P[X(0) = u] = 1$ regenerative randomization requires at most the same number of steps than standard randomization.*

Proof Let $AP_S^a(t)$ and $AP_R^a(t)$ denote the approximations to $AP(t)$ given by, respectively, standard and regenerative randomization. Let $AP_S^{\epsilon}(t)$ and $AP_R^{\epsilon}(t)$ denote the respective error aproximation bounds. By a path analysis, we will show that for the same truncation parameter K (the truncation parameter L is not involved in the randomization formulae for the case considered) $AP_R^a(t) \geq AP_S^a(t)$ and $AP_R^{\epsilon}(t) \leq AP_S^{\epsilon}(t)$.

Consider the set of all paths P of the DTMC Y. A path $p \in P$ is a sequence of $L(p)+1$ states, where $L(p)$ denotes the length of the path. Let $p(i)$, $0 \leq i \leq L(p)$ be the ith state of the path p, and let $P[p]$ be the probability of the path p, i.e.:

$$P[p] = P[Y_0 = p(0) \wedge Y_1 = p(1) \wedge \ldots \wedge Y_{L(p)} = p(L(p))].$$

Let P_a^k be the subset of P including all paths of length k which end in state a. Clearly, $P[Y_k = a] = \sum_{p \in P_a^k} P[p]$. Then, we can write $AP_S^a(t)$ as:

$$AP_S^a(t) = \sum_{k=0}^{K} \sum_{p \in P_a^k} P[p] P[N(t) = k]. \qquad (6)$$

For $AP_S^{\epsilon}(t)$ we have:

$$AP_S^{\epsilon}(t) = \sum_{k=K+1}^{\infty} P[N(t) = k]. \qquad (7)$$

Let V_K be the CTMC $V_{K,L}$ particularized to the case considered $(P[X(0) = u] = 1)$, and let W_K be the randomized version of V_K, i.e., $V_K(t) = (W_K)_{N(t)}$. W_K can be defined as follows: $(W_K)_k = W_k$ if $W_m \in \{0, 1, \ldots, K-1\} \cup \{a\}$ for $0 \leq m \leq k$ or $W_k = K$ and K has been entered only once; $(W_K)_k = b$ otherwise.

We can write $AP_R^a(t)$ and $AP_R^{\epsilon}(t)$ as:

$$AP_R^a(t) = \sum_{k=0}^{\infty} P[(W_K)_k = a] P[N(t) = k],$$

$$AP_R^{\epsilon}(t) = \sum_{k=0}^{\infty} P[(W_K)_k = b] P[N(t) = k].$$

Let $P_a^{k,K}$ be the subset of P including the paths of length k which end in state a and do not give a realization of W_K entering state b. Clearly, $P[(W_K)_k = a] = \sum_{p \in P_a^{k,K}} P[p]$. Since W_K can enter b only after at least $K + 1$ steps, $P_a^{k,K} = P_a^k$ for $0 \leq k \leq K$. Then, using (6):

$$AP_R^a(t) = \sum_{k=0}^{\infty} P[(W_K)_k = a] P[N(t) = k]$$

$$= \sum_{k=0}^{\infty} \sum_{p \in P_a^{k,K}} P[p] P[N(t) = k]$$

$$= \sum_{k=0}^{K} \sum_{p \in P_a^k} P[p] P[N(t) = k]$$

$$\quad + \sum_{k=K+1}^{\infty} \sum_{p \in P_a^{k,K}} P[p] P[N(t) = k]$$

$$= AP_S^a(t) + \sum_{k=K+1}^{\infty} \sum_{p \in P_a^{k,K}} P[p] P[N(t) = k]$$

$$\geq AP_S^a(t).$$

Let $P_b^{k,K}$ be the subset of P including the paths of length k which give a realization of W_K entering state b. Clearly, $P[(W_K)_k = b] = \sum_{p \in P_b^{k,K}} P[p]$, and, since W_K can enter b only after at least $K+1$ steps, $P_b^{k,K} = \phi$ for $0 \leq k \leq K$. Then, using (7):

$$AP_R^{\epsilon}(t) = \sum_{k=0}^{\infty} P[(W_K)_k = b] P[N(t) = k]$$

$$= \sum_{k=0}^{\infty} \sum_{p \in P_b^{k,K}} P[p] P[N(t) = k]$$

$$= \sum_{k=K+1}^{\infty} \sum_{p \in P_b^{k,K}} P[p] P[N(t) = k]$$

$$\leq \sum_{k=K+1}^{\infty} P[N(t) = k] = AP_S^{\epsilon}(t) \bigcirc.$$

An immediate consequence of this result is the following corollary:

Corollary 1 *When X has an initial probability distribution concentrated in a single state, regenerative randomization with an appropriate selection of the regenerative state is better than standard randomization.*

Proof Let $r \in \Omega$ be the initial state of X. Take $u = r$ and apply Theorem 2 $\bigcirc$.

We now study the limiting behavior of regenerative randomization. We will show that the number of steps required to achieve an approximation error bound ϵ is, for $\Lambda t \to \infty$, $O(\log(\Lambda t/\epsilon))$. The proof is done finding a suitable upper bound for $AP^{\epsilon}(t)$ and using spectral properties. The bound is obtained in the following sequence of two propositions. The proofs of these propositions run in parallel with those of similar propositions which appear in [Car94] and are ommited here.
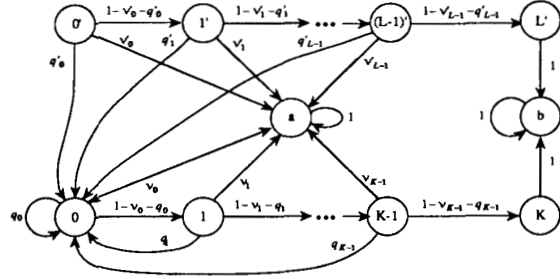

Figure 4: State transition diagram of $W_{K,L}$.

Proposition 2 *Let $W_{K,L} = \{(W_{K,L})_k; k = 0, 1, \ldots\}$ be the DTMC obtained from $V_{K,L}$ under randomization rate Λ, i.e., $V_{K,L}(t) = (W_{K,L})_{N(t)}$ (its state transition diagram is given in Figure 4). Then, $P[(W_{K,L})_k = b] \leq I(k > L)a'(L) + I(k > K)(k - K)a(K).$*

Proposition 3 *The error approximation error bound of regenerative randomization satisfies $AP^{\epsilon}(t) < a'(L) + a(K)\Lambda t.$*

Theorem 3 *The number of steps required by regenerative randomization to achieve an approximation error bound $AP^{\epsilon}(t) \leq \epsilon$ is, for $\Lambda t \to \infty$, upper bounded by a function which is $O(\log(\Lambda t/\epsilon))$.*

Proof The proof uses as a basic tool results for the spectrum of non-negative matrices (see, for instance, [Cin75]). Let Q_R be the restriction of the transition probability matrix of Z to its reachable transient states. In general, Q_R may be reducible. Let $Q_1, Q_2, \ldots, Q_M$ be the irreducible submatrices of the normal form of Q_R. Using Frobenius theorem, each Q_i has a real, positive and simple eigenvalue ρ_i such that any other eigenvalue ζ of Q_i satisfies $|\zeta| \leq \rho_i$. The equality can only be given in the case in which Q_i is periodic and, in that case, the eigenvalues ζ with $|\zeta| = \rho_i$ form with ρ_i a equally spaced rotated set of complex numbers. Also, each Q_i has rows which sum less than 1, and $\rho_i < 1$. Let $\rho = \max_{1 \leq i \leq M} \rho_i$. Using this characterization of the eigenvalues of the submatrices Q_i of Q_R, it is easy to prove that $a(k) = \sum_{i \in \Omega} P[Z_k = i]$ is, for $k \to \infty$, upper bounded by a function of the form $A\rho^k$, with $A > 0$. Similarly for Z', let ρ_i', $1 \leq i \leq M'$ be the "dominant" eigenvalues of the irreducible matrices of the normal form of Q_R', the restriction of the transition probability matrix of Z' to its reachable transient states. Let $\rho' = \max_{1 \leq i \leq M'} \rho_i'$, then, for $k \to \infty$, $a'(k)$ is upper bounded by $A'\rho'^k$, with $A' > 0$. Then, using Proposition 3, for $\Lambda t \to \infty$:

$$AP^{\epsilon}(t) < A'\rho'^L + A\rho^K \Lambda t.$$

Let $S = K + L$. Since $K = L$ may not be the optimal distribution, the approximation error for $K = L = S/2$ upper bounds the approximation error achieved with optimal distribution of the steps for a given S. Let $\rho_m = \max\{\rho, \rho'\}$. Then:

$$AP^\epsilon(t) < A'\rho'^{S/2} + A\rho^{S/2}\Lambda t < (A' + A\Lambda t)\rho_m^{S/2}.$$

The number of steps required to achieve an approximation error $\leq \epsilon$ is upper bounded by the S^* satisfying:

$$(A' + A\Lambda t)\rho_m^{S^*/2} = \epsilon.$$

Solving in S^*, considering $\rho_m < 1$:

$$S^* = \frac{2}{\log(1/\rho_m)} \log\left(\frac{A' + A\Lambda t}{\epsilon}\right) = O(\log \frac{\Lambda t}{\epsilon}) \bigcirc.$$

4 Numerical analysis

In this section we illustrate the properties of regenerative randomization and compare its performance with that of standard randomization using a relatively small but representative reliability model. We use the number of randomization steps in Y for SR and the number of steps in Z and Z' ($K+L$) for RR as comparison metric. RR requires the use of a Laplace inversion algorithm. We have used Crump's method [Cru76] with the parameter T of the inversion formula set to $2t$, other parameters set so that the relative antitransform error is smaller than 10^{-8}, and a maximum number of Laplace abscissae to achieve convergence equal to 50. In general, between 20 and 30 transform abscissae have been enough to achieve convergence. RR is implemented by advancing K, L according to the heuristic described at the end of section 2, till the goal relative error is satisfied. Crump's method adds a time complexity $O(m^2)$ per step, where m is the number of abscissae. The significance of this overhead depends on the size of the CTMC X. For the relatively small models of the examples (a few hundred states) the overhead is around 30%. For large models, the overhead would be insignificant.

The example is a fault-tolerant database including two front-ends, two databases and two processing subsystems, each one made up of a switch, a memory and two processors. The fault-tolerant database system is operational if at least one front-end and one database are unfailed and one processing subsystem is operational; a processing subsystem is operational if its switch, its memory and at least one processor are unfailed. Unfailed components of a non-operational processing subsystem and all unfailed components if the system is not operational become dormant. Dormant components do not fail. All components fail with constant rates. Front-ends and databases have failure rate 10^{-4}; switches and memories have failure rate 5×10^{-5}, and processors have failure rate 2×10^{-4}. When a processor fails it contaminates (fails) the operational databases with probability 0.01. For the repair rates we consider two sets of values. In the data

set 1 all components are repaired with rate 1. In the data set 2, databases have repair rate 0.5, front-ends, switches and memories have repair rate 1, and processors have repair rate 5. There is only one repairman which follows a preemptive resume priority strategy with random selection of the component to repair among the failed components with the same repair priority. Front-ends and databases have the highest repair priority, followed next by switches and memories, followed by processors. The measure of interest is the unreliability.

We will consider two initial states for the fault-tolerant database system: s_1 (the state in which all components are unfailed) and s_2 (the state in which only one front-end is failed). Both are operational states and $ur(0) = 0$. For data set 1, $\Lambda \approx 1$; for data set 2, $\Lambda \approx 5$. Table 1 gives the number of required steps under standard randomization (SR) and regenerative randomization (RR) with a *relative* approximation error goal $\epsilon_r = 10^{-5}$ for the data set 1 (repair rates equal) and initial state s_1, taking $u = s_1$ for RR. Table 2 gives the same results for the data set 2 (repair rates unequal). In both cases u is the initial state and RR is guaranteed to require at most the same number of steps as SR. This is confirmed by the numerical results, which show that RR can require significantly less steps than SR also for small and medium Λt. We can also see the benign behavior of RR when t increases. For data set 1, RR is remarkably efficient; for data set 2, RR is less efficient but still significantly more than SR, specially for large Λt. The difference in performance of RR in both cases can be explained by the dispersion of repair rates in the data set 2. In the first case the output rate of *all* the states with failed components is approximately equal to Λ, the probabilities q_{ii} of the DTMC Z are small and Z moves very "fast" to the absorbing state b; then, the quantities $a(k)$ decrease very rapidly with k, and very few steps are enough to achieve a small approximation error (see Proposition 3, which gives an asymptotic upper bound to $AP^\epsilon(t)$ for large Λt). For data set 2 the probabilities q_{ii} of Z are not negligible and more steps are required before $a(k)$ decreases significantly, causing RR to be slower.

The number of steps required under SR is independent on the approximation error for $\Lambda t \rightarrow \infty$. The asymptotic bound for the number of required steps under RR when $\Lambda t \rightarrow \infty$ given by Theorem 3 suggests a more sensitive behavior of RR in relation to ϵ. Figure 5 shows that behavior for the fault-tolerant database model, initial state s_1, $u = s_1$ and data set 2. We should note that here we fix the relative approximation error bound, whereas the asymptotic behavior of Theorem 3 is established in terms of the absolute approximation error bound. This explains why the number of required steps S^* seems to go asymptotically to a constant value instead of increasing logarithmically with t: for the relatively small values of Λt we consider in the figure, $AP(t)$ is approximately proportional to Λt, transforming the law $A + B\log((\Lambda t)/\epsilon)$ of Theorem 3 into a function independent of Λt, when

Table 1: Number of steps in SR and RR for the unreliability $ur(t)$ of the fault-tolerant database model for data set 1, initial state s_1, $u = s_1$, $\epsilon_r = 10^{-5}$ and several values of t.

t	$ur(t)$	steps	
		SR	RR
0.01	8.000×10^{-8}	5	3
0.1	8.003×10^{-7}	7	4
1	8.021×10^{-6}	12	5
10	8.052×10^{-5}	32	6
100	8.054×10^{-4}	162	6
10^3	8.025×10^{-3}	1,171	6
10^4	7.742×10^{-2}	10,497	6

Table 2: Number of steps in SR and RR with $u = s_1$ for the unreliability $ur(t)$ of the fault-tolerant database model for data set 2, initial state s_1, $u = s_1$, $\epsilon_r = 10^{-5}$ and several values of t.

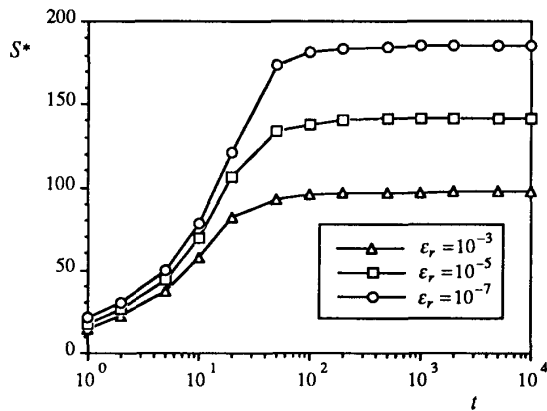| t	$ur(t)$	steps	
		SR	RR
0.01	8.000×10^{-8}	6	5
0.1	8.003×10^{-7}	10	8
1	8.023×10^{-6}	25	18
10	8.067×10^{-5}	98	69
100	8.074×10^{-4}	631	138
10^3	8.046×10^{-3}	5,376	141
10^4	7.761×10^{-2}	51,090	141

Figure 5: Number of steps in RR with $u = s_1$ for the unreliability $ur(t)$ of the fault-tolerant database model for data set 2, initial state s_1, as a function of the relative error approximation goal ϵ_r and t.

the relative error $\epsilon_r \approx B\Lambda t/\epsilon$ is considered instead of ϵ. The impact of the required relative approximation error bound is quite evident. Thus, it seems that, in RR, solution accuracy can be sensibly traded-off with computational effort, even for large Λt. This is in contrast with SR.

We next consider the fault-tolerant database model with data set 2 and initial state s_2. s_2 is a state which is rarely visited. In such a situation taking the initial state as u may degrade significantly the typical performance of RR, making it very close to SR even though RR is still theoretically better. For the fault-tolerant database model which is highly skewed to state s_1, the choice $u = s_1$ is the reasonable one. This is clearly illustrated by the results obtained with both choices which are given in Table 3. We also give the number of steps required under SR. RR with $u = s_2$ requires almost the same number of steps as SR for all t. In addition, for $t = 10^3, 10^4$ and the choice $(u = s_2)$, Crump's method did not converge, apparently due to cancellation errors caused by the very small rate at which $a(k)$ decreases. This lead us to suspect about the numerical stability of the method. However, by comparing the results given by the Laplace inversion algorithm with the solution of the $V_{K,L}$ model with SR we checked that our implementation of Crump's method was robust whenever it converged (the relative error for both the approximation and the error bound was below 10^{-8} in all cases we have tested). RR with $u = s_1$ requires slightly more steps than SR for small Λt and significantly less for large Λt, exhibiting the "benign" behavior.

5 Conclusions

A new, recently proposed randomization method called *regenerative randomization* has been extended

99

Table 3: Number of steps in SR, RR with $u = s_1$ and RR with $u = s_2$ for the unreliability $ur(t)$ of the fault-tolerant database model for data set 2, initial state s_2, $\epsilon_r = 10^{-5}$, and several values of t.

t	$ur(t)$	steps		
		SR	RR (s_1)	RR (s_2)
0.01	1.075×10^{-6}	6	9	6
0.1	1.032×10^{-5}	10	15	10
1	7.123×10^{-5}	24	36	24
10	1.807×10^{-4}	97	138	96
100	9.074×10^{-4}	631	258	629
10^3	8.145×10^{-3}	5,376	258	*
10^4	7.770×10^{-2}	51,090	258	*

to the computation of measures of transient CTMC's of the type "absorption probability distribution". An example of such a measure is the unreliability. We have proved that in the common case in which the initial distribution of the model is concentrated in a single state, regenerative randomization (RR) is always better (requires at most the same number of steps) than standard randomization (SR). Using numerical examples it has been shown that RR can be orders of magnitude faster than SR, specially in dependability models which often have "rare" states and, at the same time, frequently visited states which are good candidates for the state u required by RR. The method described here can be also applied to models incorporating performance as well as failure/repair events, e.g., queued servers which can fail. We feel that the solution method we have proposed makes more accesible characteristics of such stiff models which depend on short as well as long term behavior, thus opening a way to explore interesting tradeoffs. We also want to point out that, at the price of an increase in the number of states, non-exponential distributions of very general classes can be accomadated in CTMC's using phase-type distributions [Bob92]. Thus, the application of the method proposed here is not restricted to models with exponential distributions.

References

[Bob92] A. Bobbio and A. Cumani, "ML estimation of the parameters of a PH distribution in triangular canonical form," in *Computer Performance Evaluation: Modelling Techniques and Tools*, Elsevier Science, 1992, pp. 33–46.

[Car94] J. A. Carrasco and A. Calderón, "Regenerative Randomization: Theory and applications," *technical report*, UPC, 1994.

[Cal94] A. Calderón and J. A. Carrasco, "Computation of Absorption Probability Distributions of continuous-time Markov Chains using Regener-ative Randomization," *technical report*, UPC, 1994.

[Cin75] E. Çinlar, *Introduction to Stochastic Processes*, Prentice-Hall, 1975, pp. 371–378.

[Cru76] K. S. Crump, "Numerical Inversion of Laplace Transforms Using a Fourier Series Approximation," *Journal of the ACM*, vol. 23, pp. 89–96, 1976.

[Dij90] N. M. Dijk, "On a Simple Proof of Uniformization for Continuous and Discrete-State Continuous-Time Markov Chains," *Adv. Appl. Prob.*, vol. 22, pp. 749–750, 1990.

[Gra77] W. K. Grassmann, "Transient solutions in Markovian queuing systems," *Comput. Operations Res.*, vol. 4, pp. 47–53, 1977.

[Gro84] D. Gross and D. R. Miller, "The randomization technique as a modelling tool and solution procedure for transient Markov processes," *Operations Res.*, vol. 32, pp. 343–361, 1984.

[Jen53] A. Jensen, "Markoff chains as an aid in the study of Markoff processes," *Skand. Akuarietidskrift*, vol. 36, pp. 87–91, 1953.

[Koh82] J. Kohlas, *Stochastic Methods of Operations Research*, Cambridge University Press, Cambridge, 1982.

[Mal94] M. Malhotra, J. K. Muppala and K. S. Trivedi, "Stiffness-Tolerant Methods for Transient Analysis of Stiff Markov Chains," *Microelectron. Reliab.*, vol. 34, no. 11, pp. 1825–1841, 1994.

[Mil83] D. R. Miller, "Reliability Calculation using Randomization for Markovian Fault-Tolerant Computing Systems," *in Proc. 13th IEEE Int. Symp. on Fault-Tolerant Computing, FTCS-13*, 1983, pp 284–289.

[Moo93] A. P. Moorsel and W. H. Sanders, "Adaptive Uniformization", Technical report, University of Arizona, 1993.

[Qur93] M. A. Qureshi and W. H. Sanders, "Reward Model Solution Methods with Impulse and Rate Rewards: An Algorithm and Numerical Results," Technical Report, University of Arizona, 1993.

[Rei88] A. Reibman and K. S. Trivedi, "Numerical Transient Analysis of Markov Models," *Comput. Operations Res.*, vol. 15, pp. 19–36, 1988.

[Rub93] G. Rubino and B. Sericola, "Interval Availability Distribution Computation," in *Proc. 23th Int. Symp. on Fault-Tolerant Computing FTCS-23*, Toulouse, pp. 48–55, June 1993.

[Sou86] E. de Souza e Silva and H. R. Gail, "Calculat-
         ing Cumulative Operational Time Distributions
         of Repairable Computer Systems," *IEEE Trans.
         on Computers*, vol. 35, pp. 322–332, 1986.

[Sou89] E. de Souza e Silva and H. R. Gail, "Calculating
         Availability and Performability Measures of Re-
         pairable Computer Systems using Randomiza-
         tion," *Journal of the ACM*, vol. 34, pp. 179–199,
         1989.