INFORMS Journal on Computing
Vol. 23, No. 2, Spring 2011, pp. 268–283
ISSN 1091-9856 | EISSN 1526-5528 | 11 | 2302 | 0268

DOI 10.1287/ijoc.1100.0399

An Efficient and Numerically Stable Method for Computing Bounds for the Interval Availability Distribution

Juan A. Carrasco
Departament d'Enginyeria Electrònica, Universitat Politècnica de Catalunya, 08028 Barcelona, Spain,
carrasco@eel.upc.edu

This paper is concerned with the computation of the interval availability (proportion of time in a time interval in which the system is up) distribution of a fault-tolerant system modeled by a finite (homogeneous) continuous-time Markov chain (CTMC). General-purpose methods for performing that computation tend to be very expensive when the CTMC and the time interval are large. Based on a previously available method (regenerative transformation) for computing the interval availability complementary distribution, we develop a method called bounding regenerative transformation for the computation of bounds for that measure. Similar to regenerative transformation, bounding regenerative transformation requires the selection of a regenerative state. The method is targeted at a certain class of models, including both exact and bounding failure/repair models of fault-tolerant systems with increasing structure function, with exponential failure and repair time distributions and repair in every state with failed components having failure rates much smaller than repair rates (F/R models), with a "natural" selection for the regenerative state. The method is numerically stable and computes the bounds with well-controlled error. For models in the targeted class and the natural selection for the regenerative state, computational cost should be traded off with bounds tightness through a control parameter. For large models in the class, the version of the method that should have the smallest computational cost should have small computational cost relative to the model size if the value above which the interval availability has to be guaranteed to be is close to 1. In addition, under additional conditions satisfied by F/R models, the bounds obtained with the natural selection for the regenerative state by the version that should have the smallest computational cost seem to be tight for all time intervals or not small time intervals, depending on whether the initial probability distribution of the CTMC is concentrated in the regenerative state or not.

Key words: engineering; applications; probability; Markov processes; reliability: availability
History: Accepted by Winfried Grassmann, Area Editor for Computational Probability and Analysis; received January 2008; revised August 2009, March 2010; accepted May 2010. Published online in *Articles in Advance* September 24, 2010.

1. Introduction

Consider a finite (homogeneous) continuous-time Markov chain (CTMC) $X = \{X(t); t \geq 0\}$ modeling a fault-tolerant system that can be up or down. Let U denote the subset of "up" states. Let $\mathbf{1}_c$ denote the indicator function returning the value 1 when condition c is satisfied and the value 0 otherwise. Then, the interval availability at time t is the random variable

$$\mathrm{IAV}(t) = \frac{1}{t}\int_0^t \mathbf{1}_{X(\tau)\in U}\, d\tau,$$

i.e., the fraction of time in the time interval $[0, t]$ in which the fault-tolerant system is up. The distribution of the interval availability is of practical interest because it quantifies the probability with which a certain interval availability level can be guaranteed to the user or users of the fault-tolerant system. The interval availability distribution is

$$\mathrm{IAVD}(t, p) = P[\mathrm{IAV}(t) \leq p] = P\left[\frac{1}{t}\int_0^t \mathbf{1}_{X(\tau)\in U}\, d\tau \leq p\right].$$

Sometimes, the interval availability complementary distribution

$$\mathrm{IAVCD}(t, p) = P[\mathrm{IAV}(t) > p] = P\left[\frac{1}{t}\int_0^t \mathbf{1}_{X(\tau)\in U}\, d\tau > p\right]$$

is used. Obviously, $\mathrm{IAVCD}(t, p) = 1 - \mathrm{IAVD}(t, p)$. To illustrate a typical behavior of the measure, Figure 1 plots $\mathrm{IAVCD}(t, p)$ of a fault-tolerant system using the pair-and-spare technique (Johnson 1989) in which active modules fail with a rate $\lambda_M = 10^{-3}\ \mathrm{h}^{-1}$, the spare module does not fail, the failure of an active module is covered with probability $C_M = 0.95$, failed modules are repaired by a single repairman with rate $\mu_M = 1\ \mathrm{h}^{-1}$, and modules do not fail when the system is down, for several values of t and values of p around the steady-state availability $\mathrm{SSA} = 0.9999$, assuming that initially all modules are unfailed. Figure 1 also gives the state diagram of the CTMC modeling the system. The up states are the states 1, 3, and 5. Note that the horizontal axis of the plot is

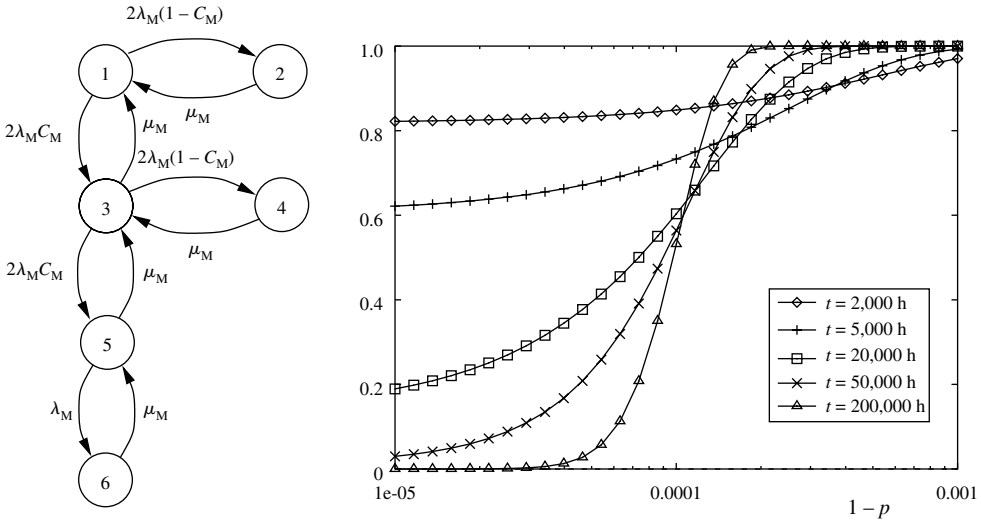

Figure 1 State Diagram of Continuous-Time Markov Chain Modeling a Repairable Fault-Tolerant System Using the Pair-and-Spare Technique (Left) and Behavior of IAVCD(t, p) (Right)

labelled with $1 - p$, so that IAVCD(t, p) decreases as p increases, as it should happen. Being irreducible and finite, the CTMC is ergodic (see, for instance, Kulkarni 1995), and as predicted by renewal reward process and regenerative process theories (see, for instance, Ross 1983), for $t \to \infty$, IAVCD(t, p) has an asymptotic shape with IAVCD$(t, p) = 1$ for $p <$ SSA and IAVCD$(t, p) = 0$ for $p \geq$ SSA, but the convergence to that asymptotic shape is very slow, making meaningful the computation of the measure for very large values of t and stressing the need for methods that have a small computational cost for large t.

Computing IAVD(t, p) or IAVCD(t, p) of a system modeled by a CTMC has been proved to be a challenging problem (Carrasco 2004; de Souza e Silva and Gail 1986; Goyal and Tantawi 1988; Ross 1983; Rubino and Sericola 1992, 1993, 1995; Sericola 1990; Takács 1957). The first effort is reported in Takács (1957), where a closed-form integral expression for IAVCD(t, p) is given for a two-state CTMC. In Ross (1983), randomization is used to obtain a closed-form expression for IAVD(t, p) for the same two-state CTMC. The first method able to deal with arbitrary finite CTMCs is presented in de Souza e Silva and Gail (1986), who develop a method for computing IAVD(t, p) using randomization. Goyal and Tantawi (1988) develop an approximate finite-differences method without error bounds for computing $P[\text{IAV}(t) \geq p]$ for arbitrary finite CTMCs. We note that, for $0 < p < 1$, $P[\text{IAV}(t) \geq p] = \text{IAVCD}(t, p)$, since the event IAV$(t) = p$ has probability measure 0. Sericola (1990) obtains a closed-form solution for IAVD(t, p) in terms of growing size matrices for arbitrary finite CTMCs. Rubino and Sericola (1992) develop an efficient method for the computation of IAVD(t, p) for finite CTMCs having a particular structure so that initially the system is up, consecutive up and down periods are independent, all up periods (except perhaps the first one) are identically distributed, and all down periods are identically distributed. Rubino and Sericola (1993) develop two randomization-based algorithms for arbitrary finite CTMCs reducing the computational requirements of the method developed by de Souza e Silva and Gail (1986). The first such algorithm reduces the time requirements and computes IAVD(t, p); the second one reduces the storage requirements and computes IAVCD(t, p). This second algorithm is reviewed in Rubino and Sericola (1995) as Algorithm A, where it is taken as a starting point to develop another algorithm (Algorithm B) for computing IAVCD(t, p), which is competitive when the number of up states of the model is small and furthermore can deal with some class of CTMCs with denumerable infinite state spaces.

Finally, a method that we will call *regenerative transformation*, has been developed by Carrasco (2004). The method covers finite CTMC models with a particular structure. The method computes IAVCD(t, p), requires the selection of a regenerative state, and is targeted at a class of finite CTMC models, class C_1. This class includes both exact and bounding failure/repair models of fault-tolerant systems with increasing structure function (see, for instance, Barlow and Proschan 1981), with exponential failure and repair time distributions and repair in every state with failed components having failure rates much smaller than repair rates (F/R models in the following), with a "natural" selection for the regenerative state. In this method, a truncated transformed CTMC is built that, with an appropriate subset of up states, has the same interval availability distribution as the original model with an arbitrarily

small error, and that truncated transformed CTMC is solved using Algorithm A of Rubino and Sericola (1995) (Algorithm A in the following). As with all other randomization-based methods, regenerative transformation is numerically stable and computes IAVCD(t, p) with well-controlled error. For large class C_1 models, large t, and p close to 1, which is the interesting case for fault-tolerant systems, the method can have significantly less computational cost than Algorithm A, which can be considered the current randomization-based general-purpose state-of-the-art method for computing IAVCD(t, p) for finite CTMCs.

In this paper, we take the regenerative transformation method as starting point to develop a method, called *bounding regenerative transformation*, for computing bounds for IAVCD(t, p). Similar to regenerative transformation, bounding regenerative transformation requires the selection of a regenerative state and is targeted at a class of models, class C_1', which is a subclass of model class C_1, including F/R models, with a given natural selection for the regenerative state.

The rest of this paper is organized as follows. Section 2 discusses the computational cost of Algorithm A and reviews the regenerative transformation method at the detail required by the developments to follow. Section 3 describes the bounding regenerative transformation method, specifying the CTMC models covered by the method, describing the model class C_1' at which the method is targeted, motivating the method, and showing that it indeed obtains bounds. The method depends on a control parameter D_C that for class C_1' models with the natural selection for the regenerative state should trade off computational cost with bounds tightness. In §3, it is also argued that for class C_1' models with the natural selection for the regenerative state, bounding regenerative transformation can have a smaller computational cost than regenerative transformation, and, for large C_1' models with the natural selection for the regenerative state, large t, p close to 1, and D_C set so that the computational cost should be the smallest one, it should have small computational cost relative to the model size, much smaller than that of Algorithm A. Section 3.2 justifies and describes a more efficient implementation of the method for an important particular case. Section 4 analyzes the performance of the method using a representative large class C_1' model and compares using that example the computational cost of the method with those of Algorithm A and regenerative transformation for p close to 1. We also illustrate that, under additional conditions satisfied by F/R models, the bounds seem to be tight for any time interval or not small time intervals, depending on whether the initial probability distribution of the model is concentrated in the regenerative state or not. The Online Supplement (available at http://joc.pubs.informs.org/ecompanion.html) includes all proofs.

2. Preliminaries

Let $X = \{X(t); t \geq 0\}$ be a CTMC with state space Ω partitioned into the set of up states U and the set of down states D. In this paper, we target the computation of bounds for IAVCD(t, p), where $t > 0$ and $0 < p < 1$.

Algorithm A is based on the randomization construct. In that construct (see, for instance, Kijima 1997), the given CTMC X is interpreted in terms of a discrete-time Markov chain subordinated to a Poisson process with arrival rate $\Lambda \geq \max_{i\in\Omega} \lambda_i$, where λ_i is the output rate of X from state i. For not too small X, large Λt, and p close to 1, the method has an approximate flop count $NC'(2T + 2|\Omega|)$, where N and C' are (Rubino and Sericola 1993) truncation parameters and T is the number of transitions of X. An important feature of the method is that it is numerically stable, the only important error source being the truncation error. The truncation parameters N and C' increase with Λ, making $\Lambda = \max_{i\in\Omega} \lambda_i$ the best selection for Λ. Using the well-known result (see, for instance, Theorem 3.3.5 in Ross 1983) that the number of arrivals in the time interval $[0, t]$ of a Poisson process with arrival rate Λ has for $\Lambda t \to \infty$ an asymptotic normal distribution with mean and variance Λt, for large Λt and a truncation error requirement $\varepsilon \ll 1$, the required N will be $\approx \Lambda t$, and, then, the method will be very costly if the model is large. As an example, for the model considered in §4, which has 646,646 states, 15,578,290 transitions, and $\Lambda = \max_{i\in\Omega} \lambda_i \approx 2.25\ \text{h}^{-1}$, we can estimate a flop count of 8.25×10^{13} when the method is run with a single target (t, p) pair with $t = 20{,}000$ h and $p = 0.9995$ and a truncation error requirement $\varepsilon = 10^{-8}$, which yields $N = 46{,}241$ and $C' = 55$.

The regenerative transformation method developed in Carrasco (2004) was an effort to reduce the high computational cost of Algorithm A for large CTMCs, large t, and p close to 1. The method requires the selection of a regenerative state r and is targeted at a particular class of models, class C_1, including F/R models, with a natural selection for the regenerative state. Because the method developed in this paper for computing bounds for IAVCD(t, p) is based on regenerative transformation, in the remainder of this section we will review the regenerative transformation method at the detail required by the developments to follow. Let $\alpha_i = P[X(0) = i]$ and let $\lambda_{i,j}$ denote the transition rate of X from state i to state j. Given $B \subset \Omega$, let $\alpha_B = \sum_{i\in B} \alpha_i$ denote the initial probability of X in subset B, and given $i \in \Omega$ and $B \subset \Omega - \{i\}$, let $\lambda_{i,B} = \sum_{j\in B} \lambda_{i,j}$ denote the transition rate of X from state i

to subset B. With the notation $S' = S - \{r\}$, $U_S = U \cap S$, $D_S = D \cap S$, $U'_S = U_S - \{r\}$, and $D'_S = D_S - \{r\}$, the method will cover CTMCs X and selections for r satisfying the following conditions (the set S is implicitly defined by Conditions 2, 4, and 8; see the following discussion).

Condition 1. Ω is finite.

Condition 2. Either $\Omega = S$ or $\Omega = S \cup \{f\}$, f being an absorbing state.

Condition 3. $|S| \geq 2$.

Condition 4. Either all states in S are transient or X has a single recurrent class of states $F \subset S$.

Condition 5. All states are reachable (from some state with non-null initial probability).

Condition 6. $U \neq \varnothing$ and $D \neq \varnothing$.

Condition 7. $\max_{i \in U} \lambda_i > 0$ and $\max_{i \in D} \lambda_i > 0$.

Condition 8. $r \in S$ and, if X has a single recurrent class of states $F \subset S$, $r \in F$.

Condition 9. If $U'_S \neq \varnothing$, $\lambda_{r, U'_S} > 0$.

Condition 10. If $U'_S \neq \varnothing$, $\alpha_{D'_S} > 0$, and $\alpha_{U'_S} = 0$, then $\lambda_{i, U'_S} > 0$ for some $i \in D'_S$ with $\alpha_i > 0$.

Note that Conditions 2, 4, and 8 and the required specification of the regenerative state r "force" the subset S and, if existent, the state f for which the remaining conditions have to be checked to determine whether the given X with the given r is covered by regenerative transformation. More specifically, if X does not have any absorbing state, by Condition 2, S must be Ω and f does not exist; if X has a single absorbing state a and $r \neq a$, then S must be $\Omega - \{a\}$ and f must be a, since $S = \Omega$ would make, by Condition 4, $F = \{a\}$, and, by Condition 8, r should be a; if X has a single absorbing state a and $r = a$, then S must be Ω and f does not exist, since $S = \Omega - \{a\}$ and $r = a$ would contradict $r \in S$ (Condition 8); if X has two absorbing states a and b, and r is one of them, say, a, then S must be $\Omega - \{b\}$ and f must be b, since $S = \Omega$ would make S to have at least two recurrent classes, in contradiction with Condition 4, and $S = \Omega - \{a\}$, $f = a$ would imply $r \notin S$, in contradiction with Condition 8; if X has two absorbing states, say, a and b, but none of them is r, then no selections for S and f exist satisfying the conditions, since $S = \Omega$ would imply that S has at least two recurrent classes, in contradiction with Condition 4, and, say, $S = \Omega - \{a\}$ would imply that S has at least the recurrent class $\{b\}$ and, by Condition 8, we should have $r = b$; finally, if X has more than three absorbing states, $S = \Omega$ makes S to include at least three recurrent classes, in contradiction with Condition 4, and $S = \Omega - \{f\}$, where f is one of the absorbing states, makes S to include at least two recurrent classes, also in contradiction with Condition 4.

Conditions 3, 6, and 7 are mild, in the sense that when they are not satisfied, computation of IAVCD(t, p) either is trivial or can be reduced to a simpler problem. Thus, assuming $U \neq \varnothing$ and $\max_{i \in U} \lambda_i = 0$, all up states would be absorbing, and we would have IAVCD$(t, p) = P[X((1-p)t) \in U]$. Similarly, assuming $D \neq \varnothing$ and $\max_{i \in D} \lambda_i = 0$, all down states would be absorbing, and we would have IAVCD$(t, p) = P[X(pt) \in U]$. Condition 5 can be trivialized by deleting unreachable states. Finally, Conditions 9 and 10 can be circumvented by adding to X a tiny transition rate $\lambda \leq 10^{-10}\varepsilon/(2t_{\max})$ between an appropriate pair of states, where ε is the allowed error and $t_{\max}$ is the largest time t at which IAVCD(t, p) has to be computed, with a negligible impact on IAVCD(t, p) no greater than $10^{-10}\varepsilon$ (see Carrasco 2004).

That X can have an absorbing state f is allowed so as to cover bounding CTMCs (de Souza e Silva and Ochoa 1992), which are useful for systems for which an exact CTMC would have a state space of unmanageable size. A bounding CTMC would have a state space $\Omega = S \cup \{f\}$, where S is a subset of the state space of the exact CTMC and f is an absorbing state in which the bounding CTMC is whenever the exact CTMC has visited some state outside S. The initial probability distribution in S would be as in the exact CTMC and the initial probability of f would be the probability that the exact CTMC is initially outside S. Transition rates between states in S would be as in the exact CTMC and transition rates from states $i \in S$ to f would be equal to the transition rates from states $i \in S$ to outside S in the exact CTMC. The up states in S would be as in the exact CTMC. Considering f to be a down (up) state results in an IAVCD(t, p) measure for the bounding CTMC that bounds from below (above) the IAVCD(t, p) measure of the exact CTMC.

The model class C_1 at which the regenerative transformation method is targeted includes all CTMCs X satisfying Conditions 1–7 and the following condition:

Condition 11. A partition $U_0 \cup U_1 \cup \cdots \cup U_{N_C}$ for U_S exists satisfying the following properties:

Property 1. $U_0 = \{o\}$ (i.e., $|U_0| = 1$).

Property 2. If X has a single recurrent class of states $F \subset S$, $o \in F$.

Property 3. If $|U_S| \geq 2$, $\lambda_{o, U_1 \cup \cdots \cup U_{N_C}} > 0$.

Property 4. If $|U_S| \geq 2$, $\alpha_{D_S} > 0$, and $\alpha_{U_1 \cup \cdots \cup U_{N_C}} = 0$, $\lambda_{i, U_1 \cup \cdots \cup U_{N_C}} > 0$ for some $i \in D_S$ with $\alpha_i > 0$.

Property 5. If $N_C > 0$,

$$\max_{0 \leq k \leq N_C} \max_{i \in U_k} \lambda_{i, U_k - \{i\} \cup U_{k+1} \cup \cdots \cup U_{N_C} \cup D_S}$$

is significantly smaller than

$$\min_{0 < k \leq N_C} \min_{i \in U_k} \lambda_{i, U_0 \cup \cdots \cup U_{k-1}} > 0 \quad \text{if } \Omega = S$$

or

$$\min_{0 < k \leq N_C} \min_{i \in U_k} \lambda_{i, U_0 \cup \cdots \cup U_{k-1} \cup \{f\}} > 0 \quad \text{if } \Omega = S \cup \{f\}.$$

The natural selection for the regenerative state for class C_1 models is $r = o$. With that natural selection, Properties 2, 3, and 4 imply the fulfillment of, respectively, Conditions 8, 9, and 10. Model class C_1 includes F/R models. A partition for U_S showing that those models are in class C_1 would be the partition in which U_k includes up states with a given number of failed components, with the subsets U_k sorted following increasing numbers of failed components. That the structure function of the fault-tolerant system be increasing is required because otherwise there could be fast repair transitions from up states to down states, and, then, Property 5 of the partition for U_S might not be satisfied. Properties 2, 3, and 4 were not mentioned in Carrasco (2004), but they are implicitly enforced for the natural selection $r = o$ by Conditions 8, 9, and 10. The condition $|U_S| \geq 2$ was enforced for class C_1 models in Carrasco (2004), but the particular case $|U_S| = 1$ taken out from the definition of class C_1 models in Carrasco (2004) was discussed there, and we have decided to include it here.

The regenerative transformation method includes two phases. In the first one, a truncated transformed CTMC, V_T, is built by analyzing the given CTMC X, which, with an appropriate subset of up states, has the same IAVCD(t, p) measure as X with error $\leq \varepsilon/2$. In the second one, the IAVCD(t, p) measure of V_T is computed with truncation error $\leq \varepsilon/2$ using Algorithm A. This results in the computation of IAVCD(t, p) for the given CTMC X with an error that can be estimated to be $\leq \varepsilon$. V_T is obtained by first characterizing the behavior of X from $S' = S - \{r\}$ until either hit of state r or, if existing, hit of the absorbing state f, and from r until either next hit of state r or, if existing, hit of the absorbing state f, while keeping track of the amount of time spent in U_S. This gives a CTMC with infinite state space with the same IAVCD(t, p) measure. Finally, up to two truncations on that CTMC with infinite state space are performed to obtain V_T. The reader is referred to Carrasco (2004) for details.

We now start describing V_T as a black box and how can it be built from X at the detail required by the developments to follow in §3. X can be interpreted as an (homogeneous) discrete-time Markov chain (DTMC) $\widehat{X} = \{\widehat{X}_n, n = 0, 1, 2, \ldots\}$ randomized with rate $\Lambda_U = (1+\theta)\max_{i\in U}\lambda_i > 0$ in the states in U and rate $\Lambda_D = (1+\theta)\max_{i\in D}\lambda_i > 0$ in the states in D, where θ is a small quantity >0, say, $\theta = 10^{-4}$. The DTMC $\widehat{X}$ has same state space and initial probability distribution as X and transition matrix $\mathbf{P} = (P_{i,j})_{i,j\in\Omega}$, where $P_{i,j} = \lambda_{i,j}/\Lambda_U$, $i \in U$, $j \neq i$, $P_{i,i} = 1 - \lambda_i/\Lambda_U$, $i \in U$, $P_{i,j} = \lambda_{i,j}/\Lambda_D$, $i \in D$, $j \neq i$, and $P_{i,i} = 1 - \lambda_i/\Lambda_D$, $i \in D$. Let $\widehat{X}'$ denote a version of $\widehat{X}$ with initial state r, and, given a DTMC Y, let $Y_{m_1:m_2}c$, $m_1, m_2 \geq 0$, denote the predicate that is true when Y_n satisfies condition c for all n, $m_1 \leq n \leq m_2$ (by convention, $Y_{m_1:m_2}c$ is true for $m_2 < m_1$) and let $\#(Y_{m_1:m_2}c)$ denote the number of indices n, $m_1 \leq n \leq m_2$, for which Y_n satisfies condition c. Let $\boldsymbol{\pi}(n,k) = (\pi_i(n,k))_{i\in S}$, $n \geq 0$, $0 \leq k \leq n+1$, denote row vectors where $\pi_i(n,k) = P[\widehat{X}'_n = i \wedge \widehat{X}'_{1:n} \neq r \wedge \#(\widehat{X}'_{0:n} \in U) = k]$, and let $\boldsymbol{\pi}'(n,k) = (\pi'_i(n,k))_{i\in S'}$, $n \geq 0$, $0 \leq k \leq n+1$, denote row vectors where $\pi'_i(n,k) = P[\widehat{X}_n = i \wedge \widehat{X}_{0:n} \neq r \wedge \#(\widehat{X}_{0:n} \in U) = k]$. In other words, $\pi_i(n,k)$ is the probability that in the first n steps $\widehat{X}'$ will not have entered state r, has visited k up states, and at step n is in state i, $i \in S$; and $\pi'_i(n,k)$ is the probability that in the first n steps $\widehat{X}$ has not visited state r, has visited k up states, and at step n is in state i, $i \in S'$. Let $a(n,k) = \sum_{i\in S}\pi_i(n,k)$, $a_m(k) = \sum_{n=k-1}^{k+m-1} a(n,k)$, $k \geq 2$, $m \geq 1$, $a'(n,k) = \sum_{i\in S'}\pi'_i(n,k)$, and $a'_m(k) = \sum_{n=k-1}^{k+m-1} a'(n,k)$, $k \geq 2$, $m \geq 1$.

The truncated transformed CTMC V_T is defined by up to three truncation parameters, K, L, and C. The truncation parameter C is given by

$$C = \min\left\{c \geq 1 \colon \sum_{m=c+1}^{\infty} \frac{(\Lambda\, tq_{\max})^m}{m!}\, e^{-\Lambda\, tq_{\max}} \leq \varepsilon_1\right\}, \quad (1)$$

where $\Lambda = \max\{\Lambda_U, \Lambda_D\}$, $tq_{\max}$ is the largest value of $tq = t(1-p)$ at which IAVCD(t, p) has to be computed, and $\varepsilon_1 = \varepsilon/4$ if $U'_S \neq \varnothing$ and $\varepsilon_1 = \varepsilon/2$ if $U'_S = \varnothing$. For the case $U'_S \neq \varnothing$ and $\alpha_{S'} > 0$, the truncation parameters K and L are given by

$$K = \min\left\{k \geq 2 \colon \alpha_S a_C(k) \sum_{m=k}^{\infty}(m-k+2) \cdot \frac{(\Lambda_U t_{\max})^m}{m!}\, e^{-\Lambda_U t_{\max}} \leq \frac{\varepsilon}{8}\right\}, \quad (2)$$

$$L = \min\left\{k \geq 2 \colon a'_C(k) \sum_{m=k}^{\infty} \frac{(\Lambda_U t_{\max})^m}{m!}\, e^{-\Lambda_U t_{\max}} \leq \frac{\varepsilon}{8}\right\}, \quad (3)$$

and, for the case $U'_S \neq \varnothing$ and $\alpha_{S'} = 0$, the truncation parameter K is given by

$$K = \min\left\{k \geq 2 \colon \alpha_S a_C(k) \sum_{m=k}^{\infty}(m-k+2) \cdot \frac{(\Lambda_U t_{\max})^m}{m!}\, e^{-\Lambda_U t_{\max}} \leq \frac{\varepsilon}{4}\right\}, \quad (4)$$

where $t_{\max}$ is the largest value of t at which IAVCD(t, p) has to be computed.

Then, with $\mathbf{x}^B$ denoting the restriction of the row vector $\mathbf{x}$ to the subset of indices B, $\mathbf{0}$ denoting a row vector of appropriate dimension with all components null, f, a, and b being absorbing states, and $\bigcup_c$ denoting a union to be performed when condition c is satisfied, V_T has, for the case $\alpha_{S'} > 0$, state space (note that Conditions 2, 6, and 7 imply $U_S \neq \varnothing$ and $D_S \neq \varnothing$)

$$\Omega_T = \{s^u_{n,k} \colon (n,k) \in D_T \wedge \boldsymbol{\pi}(n,k)^{U_S} \neq \mathbf{0}\} \cup \{s^d_{n,k} \colon (n,k) \in D_T \wedge \boldsymbol{\pi}(n,k)^{D_S} \neq \mathbf{0}\}$$

$$\bigcup_{U'_S \neq \varnothing} \{s'^{u}_{n,k} : (n,k) \in D'_T \wedge \boldsymbol{\pi}'(n,k)^{U'_S} \neq \mathbf{0}\}$$

$$\bigcup_{D'_S \neq \varnothing} \{s'^{d}_{n,k} : (n,k) \in D'_T \wedge \boldsymbol{\pi}'(n,k)^{D'_S} \neq \mathbf{0}\}$$

$$\bigcup_{\Omega = S \cup \{f\}} \{f\} \cup \{a\} \bigcup_{U'_S \neq \varnothing} \{b\},$$

and, for the case $\alpha_{S'} = 0$, state space

$$\Omega_T = \{s^u_{n,k} : (n,k) \in D_T \wedge \boldsymbol{\pi}(n,k)^{U_S} \neq \mathbf{0}\} \cup \{s^d_{n,k} : (n,k) \in D_T \wedge \boldsymbol{\pi}(n,k)^{D_S} \neq \mathbf{0}\} \bigcup_{\Omega = S \cup \{f\}} \{f\} \cup \{a\} \bigcup_{U'_S \neq \varnothing} \{b\},$$

where, for $U'_S \neq \varnothing$,

$$D_T = \{(n,k) : 0 \le k \le K \wedge \max\{0, k-1\} \le n \le k + C - 1\}$$

and

$$D'_T = \{(n,k) : 0 \le k \le L \wedge \max\{0, k-1\} \le n \le k + C - 1\};$$

for $U'_S = \varnothing$ and $r \in U_S$,

$$D_T = \{(n,1) : 0 \le n \le C\};$$

for $U'_S = \varnothing$ and $r \in D_S$,

$$D_T = \{(n,0) : 0 \le n \le C - 1\};$$

and, for $U'_S = \varnothing$,

$$D'_T = \{(n,0) : 0 \le n \le C - 1\}.$$

Figure 2 depicts the domain D_T for the case $U'_S \neq \varnothing$. The domain D'_T for the case $U'_S \neq \varnothing$ is identical with K replaced by L.

The initial probability distribution of V_T is $P[V_T = s^{\cdot}_{0,\cdot}] = \alpha_r$, $P[V_T = s'^{u}_{0,1}] = \alpha_{U'_S}$, $P[V_T = s'^{d}_{0,0}] = \alpha_{D'_S}$, $P[V_T = f] = \alpha_f$, and $P[V_T = i] = 0$, $i \notin \{s^{\cdot}_{0,\cdot}, s'^{u}_{0,1}, s'^{d}_{0,0}, f\}$, where $s^{\cdot}_{0,\cdot}$ denotes state $s^u_{0,1}$ if $r \in U_S$ and state $s^d_{0,0}$ if $r \in D_S$. Note that, according to the definition of $\boldsymbol{\pi}(n,k)$ and Ω_T, for $r \in U_S$, the only state $s^u_{0,k}$ or $s^d_{0,k}$ present in Ω_T is state $s^u_{0,1}$, and, for $r \in D_S$, the only state $s^u_{0,k}$ or $s^d_{0,k}$ present in Ω_T is state $s^d_{0,0}$. It is that single state that is denoted by $s^{\cdot}_{0,\cdot}$.

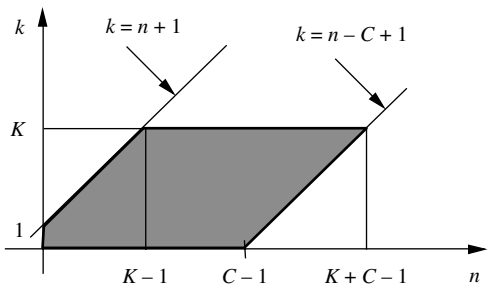

Figure 2 **Domain D_T for the Case $U'_S \neq \varnothing$ (the Domain Includes the Points in the Frontier)**

With $P_{i,B}$, $B \subset \Omega$ denoting $\sum_{j \in B} P_{i,j}$, the transition rates in V_T are as follows. Let

$$w^{uu}_{n,k} = \sum_{i \in U_S} \pi_i(n,k) P_{i,U'_S} \Big/ \sum_{i \in U_S} \pi_i(n,k),$$

$$w^{ud}_{n,k} = \sum_{i \in U_S} \pi_i(n,k) P_{i,D'_S} \Big/ \sum_{i \in U_S} \pi_i(n,k),$$

$$w^{du}_{n,k} = \sum_{i \in D_S} \pi_i(n,k) P_{i,U'_S} \Big/ \sum_{i \in D_S} \pi_i(n,k),$$

$$w^{dd}_{n,k} = \sum_{i \in D_S} \pi_i(n,k) P_{i,D'_S} \Big/ \sum_{i \in D_S} \pi_i(n,k),$$

$$q^{u}_{n,k} = \sum_{i \in U_S} \pi_i(n,k) P_{i,r} \Big/ \sum_{i \in U_S} \pi_i(n,k),$$

$$q^{d}_{n,k} = \sum_{i \in D_S} \pi_i(n,k) P_{i,r} \Big/ \sum_{i \in D_S} \pi_i(n,k),$$

$$v^{u}_{n,k} = \sum_{i \in U_S} \pi_i(n,k) P_{i,f} \Big/ \sum_{i \in U_S} \pi_i(n,k),$$

$$v^{d}_{n,k} = \sum_{i \in D_S} \pi_i(n,k) P_{i,f} \Big/ \sum_{i \in D_S} \pi_i(n,k),$$

$$w'^{uu}_{n,k} = \sum_{i \in U'_S} \pi'_i(n,k) P_{i,U'_S} \Big/ \sum_{i \in U'_S} \pi'_i(n,k),$$

$$w'^{ud}_{n,k} = \sum_{i \in U'_S} \pi'_i(n,k) P_{i,D'_S} \Big/ \sum_{i \in U'_S} \pi'_i(n,k),$$

$$w'^{du}_{n,k} = \sum_{i \in D'_S} \pi'_i(n,k) P_{i,U'_S} \Big/ \sum_{i \in D'_S} \pi'_i(n,k),$$

$$w'^{dd}_{n,k} = \sum_{i \in D'_S} \pi'_i(n,k) P_{i,D'_S} \Big/ \sum_{i \in D'_S} \pi'_i(n,k),$$

$$q'^{u}_{n,k} = \sum_{i \in U'_S} \pi'_i(n,k) P_{i,r} \Big/ \sum_{i \in U'_S} \pi'_i(n,k),$$

$$q'^{d}_{n,k} = \sum_{i \in D'_S} \pi'_i(n,k) P_{i,r} \Big/ \sum_{i \in D'_S} \pi'_i(n,k),$$

$$v'^{u}_{n,k} = \sum_{i \in U'_S} \pi'_i(n,k) P_{i,f} \Big/ \sum_{i \in U'_S} \pi'_i(n,k),$$

$$v'^{d}_{n,k} = \sum_{i \in D'_S} \pi'_i(n,k) P_{i,f} \Big/ \sum_{i \in D'_S} \pi'_i(n,k).$$

Then,

- If $U'_S \neq \varnothing$, each state $s^u_{n,k}$, $0 \le k < K$, has a transition rate $w^{uu}_{n,k}\Lambda_U$ to state $s^u_{n+1,k+1}$, a transition rate $w^{ud}_{n,k}\Lambda_U$ to state $s^d_{n+1,k}$ if $n \le k + C - 2$ and to state a otherwise, a transition rate $q^u_{n,k}\Lambda_U$ to state $s^{\cdot}_{0,\cdot}$ if $s^u_{n,k} \neq s^{\cdot}_{0,\cdot}$, and, if $\Omega = S \cup \{f\}$, a transition rate $v^u_{n,k}\Lambda_U$ to state f.
- If $U'_S \neq \varnothing$, each state $s^u_{n,K}$ has a transition rate Λ_U to state b.
- If $U'_S = \varnothing$, each state $s^u_{n,k}$ has a transition rate $w^{ud}_{n,k}\Lambda_U$ to state $s^d_{n+1,k}$ if $n \le k + C - 2$ and to state a otherwise, a transition rate $q^u_{n,k}\Lambda_U$ to state $s^{\cdot}_{0,\cdot}$ if

$s^u_{n,k} \neq s^{\cdot}_{0,\cdot}$, and, if $\Omega = S \cup \{f\}$, a transition rate $v^u_{n,k}\Lambda_U$ to state f.

• If $U'_S \neq \varnothing$, each state $s^d_{n,k}$, $0 \le k < K$, has a transition rate $w^{du}_{n,k}\Lambda_D$ to state $s^u_{n+1,k+1}$, a transition rate $w^{dd}_{n,k}\Lambda_D$ to state $s^d_{n+1,k}$ if $n \le k + C - 2$ and to state a otherwise, a transition rate $q^d_{n,k}\Lambda_D$ to state $s^{\cdot}_{0,\cdot}$ if $s^d_{n,k} \neq s^{\cdot}_{0,\cdot}$, and, if $\Omega = S \cup \{f\}$, a transition rate $v^d_{n,k}\Lambda_D$ to state f.

• If $U'_S = \varnothing$, each state $s^d_{n,k}$ has a transition rate $w^{dd}_{n,k}\Lambda_D$ to state $s^d_{n+1,k}$ if $n \le k+C-2$ and to state a otherwise, a transition rate $q^d_{n,k}\Lambda_D$ to state $s^{\cdot}_{0,\cdot}$ if $s^d_{n,k} \neq s^{\cdot}_{0,\cdot}$, and, if $\Omega = S \cup \{f\}$, a transition rate $v^d_{n,k}\Lambda_D$ to state f.

• If $U'_S \neq \varnothing$, each state $s'^u_{n,k}$, $0 \le k < L$ has a transition rate $w'^{uu}_{n,k}\Lambda_U$ to state $s'^u_{n+1,k+1}$, a transition rate $w'^{ud}_{n,k}\Lambda_U$ to state $s'^d_{n+1,k}$ if $n \le k + C - 2$ and to state a otherwise, a transition rate $q'^u_{n,k}\Lambda_U$ to state $s^{\cdot}_{0,\cdot}$, and, if $\Omega = S \cup \{f\}$, a transition rate $v'^u_{n,k}\Lambda_U$ to state f.

• If $U'_S \neq \varnothing$, each state $s'^u_{n,L}$ has a transition rate Λ_U to state b.

• If $U'_S \neq \varnothing$, each state $s'^d_{n,k}$, $0 \le k < L$ has a transition rate $w'^{du}_{n,k}\Lambda_D$ to state $s'^u_{n+1,k+1}$, a transition rate $w'^{dd}_{n,k}\Lambda_D$ to state $s'^d_{n+1,k}$ if $n \le k + C - 2$ and to state a otherwise, a transition rate $q'^d_{n,k}\Lambda_D$ to state $s^{\cdot}_{0,\cdot}$, and, if $\Omega = S \cup \{f\}$, a transition rate $v'^d_{n,k}\Lambda_D$ to state f.

• If $U'_S \neq \varnothing$, each state $s'^d_{n,L}$ has a transition rate Λ_D to state b.

• If $U'_S = \varnothing$, each state $s'^d_{n,k}$ has a transition rate $w'^{dd}_{n,k}\Lambda_D$ to state $s'^d_{n+1,k}$ if $n \le k + C - 2$ and to state a otherwise, a transition rate $q'^d_{n,k}\Lambda_D$ to state $s^{\cdot}_{0,\cdot}$, and, if $\Omega = S \cup \{f\}$, a transition rate $v'^d_{n,k}\Lambda_D$ to state f.

The states that have to be considered up in V_T are the states $s^u_{n,k}$, the states $s'^u_{n,k}$, and state f if $\Omega = S \cup \{f\}$ and f is an up state in X.

To illustrate the "structure" of V_T, Figure 3 gives a sketch of the state diagram of V_T for the case $\Omega = S \cup \{f\}$, $r \in U_S$, $U'_S \neq \varnothing$, $D'_S \neq \varnothing$, $\alpha_{U'_S} > 0$, and $\alpha_{D'_S} > 0$, with truncation parameters $K = 3$, $L = 3$, and $C = 2$. In that case, since $r \in U_S$, $s^{\cdot}_{0,\cdot} = s^u_{0,1}$, and state $s^d_{0,0}$ is not present. We include in the state space all possible candidate states $s^u_{n,k}$, $s^d_{n,k}$, $s'^u_{n,k}$, $s'^d_{n,k}$ subject to the considered particular case, taking into account the formal definition of Ω_T. States $s^u_{n,k}$, $(n,k) \in D_T$ and states $s^d_{n,k}$, $(n,k) \in D_T$ that are always (for the considered particular case) outside Ω_T are indicated by dotted circles. Similarly, states $s'^u_{n,k}$, $(n,k) \in D'_T$ and states $s'^d_{n,k}$, $(n,k) \in D'_T$ that are always outside Ω_T are indicated with dotted circles. The initial probability distribution of V_T is

$$P[V_T(0) = s^u_{0,1}] = \alpha_r, \quad P[V_T(0) = s'^u_{0,1}] = \alpha_{U'_S},$$
$$P[V_T(0) = s'^d_{0,0}] = \alpha_{D'_S}, \quad P[V_T(0) = f] = \alpha_f,$$
$$P[V_T(0) = i] = 0, \quad i \notin \{s^u_{0,1}, s'^u_{0,1}, s'^d_{0,0}, f\}.$$

For the sake of readability, we do not plot the arrows corresponding to the transition rates to states f and $s^{\cdot}_{0,\cdot} = s^u_{0,1}$. There is a transition rate with value $q^u_{n,k}\Lambda_U$ from every state $s^u_{n,k}$, $n > 0$, $k < K = 3$ to state $s^u_{0,1}$, a transition rate with value $q^d_{n,k}\Lambda_D$ from every state $s^d_{n,k}$, $k < K = 3$ to state $s^u_{0,1}$, a transition rate with value $q'^u_{n,k}\Lambda_U$ from every state $s'^u_{n,k}$, $k < L = 3$ to state $s^u_{0,1}$, and a transition rate with value $q'^d_{n,k}\Lambda_D$ from every state $s'^d_{n,k}$, $k < L = 3$ to state $s^u_{0,1}$. Finally, there is a transition rate with value $v^u_{n,k}\Lambda_U$ from every state $s^u_{n,k}$, $k < K = 3$ to state f, a transition rate with value $v^d_{n,k}\Lambda_D$ from every state $s^d_{n,k}$, $k < K = 3$ to state f, a transition rate with value $v'^u_{n,k}\Lambda_U$ from every state $s'^u_{n,k}$, $k < L = 3$ to state f, and a transition rate with value $v'^d_{n,k}\Lambda_D$ from every state $s'^d_{n,k}$, $k < L = 3$ to state f.

The construction of V_T requires the computation of $\boldsymbol{\pi}(n,k)$, $(n,k) \in D_T$ and, if $\alpha_{S'} > 0$, $\boldsymbol{\pi}'(n,k)$, $(n,k) \in D'_T$. Taking into account the definition of the row vectors $\boldsymbol{\pi}(n,k)$, and with $\mathbf{P}_{B,C}$ denoting the submatrix of $\mathbf{P}$ collecting the elements with index pairs in $B \times C$, the required row vectors $\boldsymbol{\pi}(n,k)$ can be obtained, for increasing k and for each k for increasing n, using the recurrences

$$\boldsymbol{\pi}(n,k)^{U'_S} = \boldsymbol{\pi}(n-1,k-1)\,\mathbf{P}_{S,U'_S}, \quad n \ge 1,\ 1 \le k \le n+1, \tag{5}$$

$$\boldsymbol{\pi}(n,k)^{D'_S} = \boldsymbol{\pi}(n-1,k)\,\mathbf{P}_{S,D'_S}, \quad n \ge 1,\ 0 \le k \le n, \tag{6}$$

and

$$\pi_r(n,k) = 0, \quad n \ge 1,\ 0 \le k \le n+1, \tag{7}$$
$$\pi_r(0,0) = \mathbf{1}_{r \in D_S}, \tag{8}$$
$$\pi_r(0,1) = \mathbf{1}_{r \in U_S}, \tag{9}$$
$$\boldsymbol{\pi}(0,k)^{U'_S} = \mathbf{0}, \quad 0 \le k \le 1, \tag{10}$$
$$\boldsymbol{\pi}(n,0)^{U'_S} = \mathbf{0}, \quad n \ge 1, \tag{11}$$
$$\boldsymbol{\pi}(0,k)^{D'_S} = \mathbf{0}, \quad 0 \le k \le 1, \tag{12}$$
$$\boldsymbol{\pi}(n,n+1)^{D'_S} = \mathbf{0}, \quad n \ge 1. \tag{13}$$

Similarly, with $\boldsymbol{\alpha}$ denoting the row vector $(\alpha_i)_{i \in \Omega}$, the required row vectors $\boldsymbol{\pi}'(n,k)$ can be obtained, for increasing k and for each k for increasing n, using the recurrences

$$\boldsymbol{\pi}'(n,k)^{U'_S} = \boldsymbol{\pi}'(n-1,k-1)\,\mathbf{P}_{S',U'_S}, \quad n \ge 1,\ 1 \le k \le n+1, \tag{14}$$

$$\boldsymbol{\pi}'(n,k)^{D'_S} = \boldsymbol{\pi}'(n-1,k)\,\mathbf{P}_{S',D'_S}, \quad n \ge 1,\ 0 \le k \le n, \tag{15}$$

and

$$\boldsymbol{\pi}'(0,0)^{U'_S} = \mathbf{0}, \tag{16}$$
$$\boldsymbol{\pi}'(0,1)^{U'_S} = \boldsymbol{\alpha}^{U'_S}, \tag{17}$$
$$\boldsymbol{\pi}'(n,0)^{U'_S} = \mathbf{0}, \quad n \ge 1, \tag{18}$$

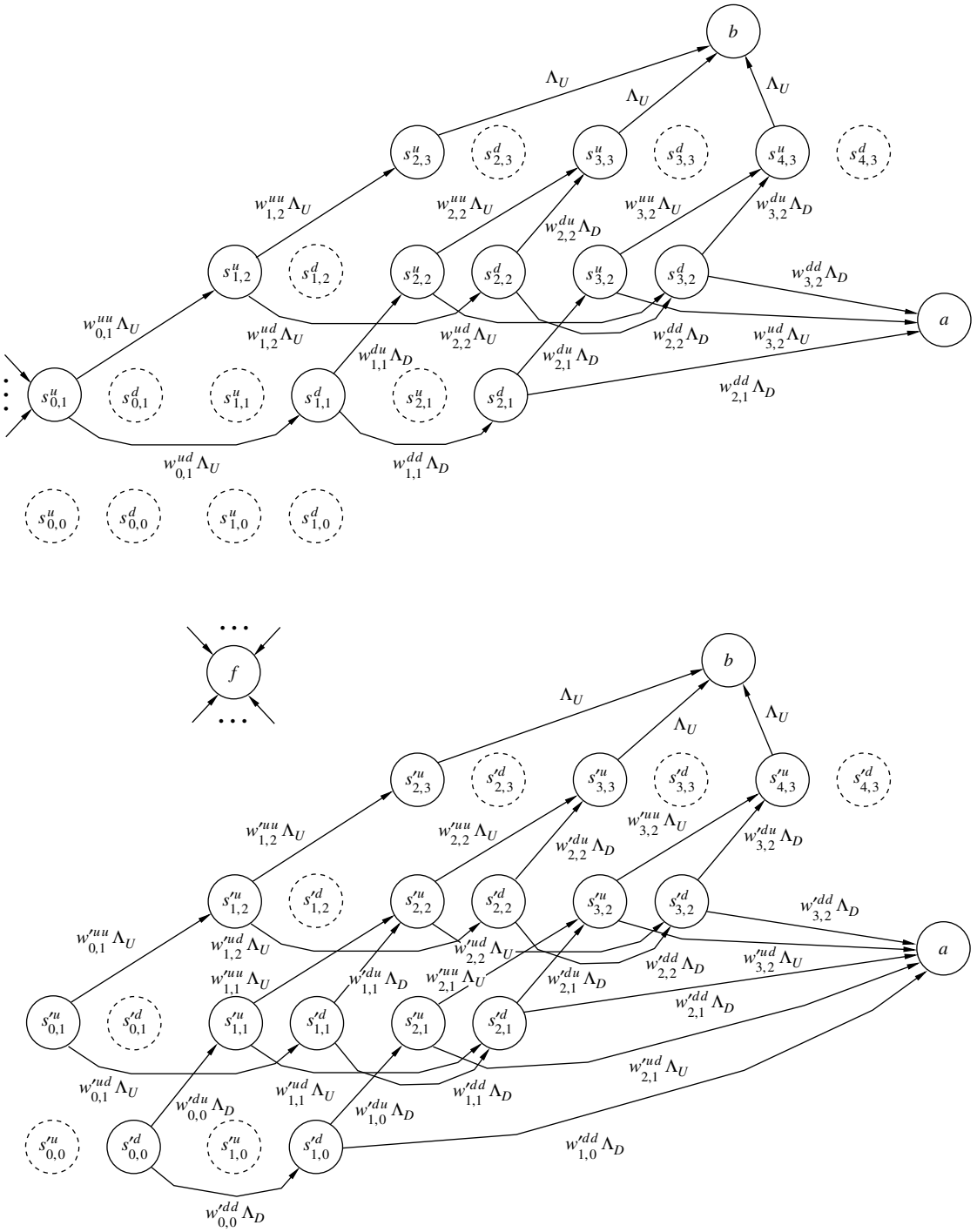

Figure 3 **Sketch of the State Diagram of V_T for the Case $\Omega = S \cup \{f\}$, $r \in U_S$, $U'_S \neq \varnothing$, $D'_S \neq \varnothing$, $\alpha_{U'_S} > 0$, and $\alpha_{D'_S} > 0$, with Truncation Parameters $K = 3$, $L = 3$, and $C = 2$**

$$\boldsymbol{\pi}'(0,0)^{D'_S} = \boldsymbol{\alpha}^{D'_S}, \tag{19}$$

$$\boldsymbol{\pi}'(0,1)^{D'_S} = \mathbf{0}, \tag{20}$$

$$\boldsymbol{\pi}'(n,n+1)^{D'_S} = \mathbf{0}, \quad n \geq 1. \tag{21}$$

Those orderings allow the computation of $a_C(k)$ and $a'_C(k)$ for increasing k and thus the determination of the truncation parameters K ((2) and (4)) and L (3).

For the case $U'_S \neq \varnothing$ and not too small models, the model transformation phase of regenerative transformation (construction of V_T) has an approximate flop count $CK(2T + M|\Omega|) + \mathbf{1}_{\alpha_{S'}>0} CL(2T + M|\Omega|)$, where T is the number of transitions of X, $M = 11$ if $\Omega = S \cup \{f\}$, and $M = 9$ if $\Omega = S$. For class C_1 models with $|U_S| \geq 2$ and the selection $r = o$, we have

the following additional result (Carrasco 2004), where $c(n) \sim d(n)$ for $n \to \infty$ denotes $\lim_{n\to\infty} c(n)/d(n) = 1$.

THEOREM 1. *For class* C_1 *models with* $|U_S| \geq 2$ *and* $r = o$, $a_C(n) \leq (C+1)h(n)$ *and* $a'_C(n) \leq \alpha_{S'}(C+1) \cdot h'(n)$, *where, for* $n \to \infty$, $h(n) \sim B\binom{n-1}{p-1}\rho^n$ *and* $h'(n) \sim B'\binom{n-1}{p'-1}\rho'^n$, *with* $B > 0$, $B' > 0$, p, p' *integers* ≥ 1, $\rho, \rho' \approx 1 - 1/R'$, *and* $R' = \max_{i\in U_S} \lambda_i / \min_{i\in U_S - \{o\}} \lambda_i$.

3. The Bounding Regenerative Transformation Method

We will start by identifying the CTMC models covered by bounding regenerative transformation and the model class C'_1 at which the method is targeted. Then, we will motivate and justify the method and will describe it in the general case. The method allows to compute a lower bound for IAVCD(t, p), an upper bound for IAVCD(t, p), or both, and depends on a control parameter D_C that for class C'_1 models should trade off computational cost with bounds tightness. A separate subsection will be dedicated to justify and describe a more efficient implementation of the method that is available for the case $D_C = 1$ when both bounds have to be computed and an additional condition is satisfied.

3.1. Motivation and General Case

Similar to regenerative transformation, the bounding regenerative transformation method requires the selection of a regenerative state r and covers the same class of CTMC models X and selections for the regenerative state r as the regenerative transformation method (Conditions 1–10) with the additional condition:

Condition 12. $U'_S \neq \varnothing$.

The additional condition is imposed because bounding regenerative transformation can be described as a succession of a phase in which transition rates of X from states in U'_S are scaled and an application of regenerative transformation to the CTMCs with scaled transition rates; when $U'_S = \varnothing$, there are no transition rates to be scaled.

The method is targeted at a model class C'_1 with a natural selection for the regenerative state r. Model class C'_1 is a subclass of model class C_1 defined by Conditions 1–7 and the conditions:

Condition 13. $|U_S| \geq 2$.

Condition 14. A partition $U_0 \cup U_1 \cup \cdots \cup U_{N_C}$ for U_S exists satisfying the following properties:

Property 1. $U_0 = \{o\}$ (i.e., $|U_0| = 1$).

Property 2. If X has a single recurrent class of states $F \subset S$, $o \in F$.

Property 3. $\lambda_{o,\, U_1\cup\cdots\cup U_{N_C}} > 0$.

Property 4. If $\alpha_{D_S} > 0$ and $\alpha_{U_1\cup\cdots\cup U_{N_C}} = 0$, $\lambda_{i,\, U_1\cup\cdots\cup U_{N_C}} > 0$ for some $i \in D_S$ with $\alpha_i > 0$.

Property 5. $\max_{0\leq k\leq N_C} \max_{i\in U_k} \lambda_{i,\, U_k - \{i\}\cup U_{k+1}\cup\cdots\cup U_{N_C}\cup D_S}$ is significantly smaller than

$$\min_{0<k\leq N_C} \min_{i\in U_k} \lambda_{i,\, U_0\cup\cdots\cup U_{k-1}} > 0 \quad \text{if } \Omega = S$$

or

$$\min_{0<k\leq N_C} \min_{i\in U_k} \lambda_{i,\, U_0\cup\cdots\cup U_{k-1}\cup\{f\}} > 0 \quad \text{if } \Omega = S \cup \{f\}.$$

Property 6. $\lambda_o \leq \min_{i\in U_1\cup\cdots\cup U_{N_C}} \lambda_i$.

Class C'_1 models obviously includes F/R models, the only additional condition being that there must be at least two up states apart from the absorbing state f, if existent. The natural selection for the regenerative state for class C'_1 models is $r = o$. Since class C'_1 is a subclass of class C_1, and for any model in class C'_1, $|U_S| \geq 2$ and $\max_{i\in U_S} \lambda_i = \max_{i\in U_S - \{o\}} \lambda_i$ because of Property 6 of the partition for U_S, we have, from Theorem 1, the following result.

THEOREM 2. *For class* C'_1 *models and* $r = o$, $a_C(n) \leq (C+1)h(n)$ *and* $a'_C(n) \leq \alpha_{S'}(C+1)h'(n)$, *where, for* $n \to \infty$, $h(n) \sim B\binom{n-1}{p-1}\rho^n$ *and* $h'(n) \sim B'\binom{n-1}{p'-1}\rho'^n$, *with* $B > 0$, $B' > 0$, p, p' *integers* ≥ 1, $\rho, \rho' \approx 1 - 1/R''$, *and* $R'' = \max_{i\in U_S - \{o\}} \lambda_i / \min_{i\in U_S - \{o\}} \lambda_i$.

The bounding regenerative transformation method is motivated by Theorem 2 and is based on the following result. See, for instance, Kijima (1997) for the definitions of conservative and uniformizable CTMCs with denumerable state space. In short, they are CTMCs with denumerable state space in which the output rate from any state i is equal to the sum of the transition rates from i and in which the output rates are uniformly bounded from above. Any finite CTMC is both conservative and uniformizable. Although we will only use the result for finite CTMCs, that restriction does not lead to a simpler proof.

THEOREM 3. *Let* W *be a conservative, uniformizable CTMC with denumerable state space* Ω, *subset of "up" states* U, *and transition rates* $\lambda_{i,j}$, $i, j \in \Omega$, $j \neq i$, *and let* W' *be another conservative, uniformizable CTMC with same state space, same initial probability distribution, same subset of up states, same transition rates from non-up states as* X, *and transition rates from up states* $\lambda'_{i,j} = \beta_i \lambda_{i,j}$, $i \in U$, $j \in \Omega$, $j \neq i$, $0 < \beta_i \leq 1$. *Let* IAVCD(t, p) *be the complementary interval availability distribution of* W; *i.e.,* IAVCD$(t, p) = P[(\int_0^t \mathbf{1}_{W(\tau)\in U}\, d\tau)/t > p]$, $t > 0$, $0 < p < 1$. *Let* IAVCD$'(t, p)$ *be the complementary interval availability distribution of* W'; *i.e.,* IAVCD$'(t, p) = P[(\int_0^t \mathbf{1}_{W'(\tau)\in U}\, d\tau)/t > p]$, $t > 0$, $0 < p < 1$. *Then,* IAVCD$'(t, p) \geq$ IAVCD(t, p). *Furthermore, it is enough that* $\beta_k < 1$ *for a reachable up state* k *from which a down state can be reached to have* IAVCD$'(t, p) >$ IAVCD(t, p).

PROOF. See the Online Supplement. □

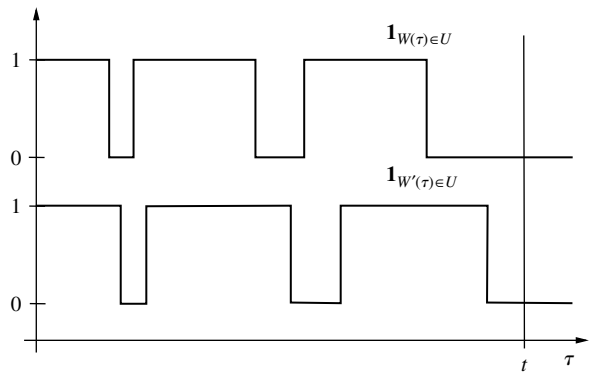

Figure 4 Comparison of Corresponding Realizations of W and W'

Essentially, the reason why Theorem 3 holds is that scaling transition rates from up states keeping their relative values will not modify the embedded DTMC Π of W. Because (see, for instance, Kijima 1997) W can be interpreted in terms of Π by associating with the states visited by Π independent exponential holding times with a parameter equal to the output rate from the visited state and W' can be interpreted in terms of Π by associating with the states visited by Π scaled up versions of those holding times, each realization of W will have a corresponding realization of W' differing from the former only in that the holding times in the up states will be non-smaller, and, as Figure 4 illustrates, this will cause the up time in the time interval $[0, t]$ of the realization of W' to be non-smaller than the up time in the same interval of the corresponding realization of W. As a consequence, $\text{IAVCD}'(t, p)$ will be non-smaller than $\text{IAVCD}(t, p)$. That, under the condition stated by the theorem, $\text{IAVCD}'(t, p) > \text{IAVCD}(t, p)$ follows from the fact that under that condition there will realizations of W' with a total non-null probability contributing to $\text{IAVCD}'(t, p)$ with corresponding realizations of W that do not contribute to $\text{IAVCD}(t, p)$.

According to Theorem 3, scaling up the transition rates from some up states will result in a CTMC model whose $\text{IAVCD}(t, p)$ measure will bound from below the $\text{IAVCD}(t, p)$ measure of the original model. Conversely, scaling down the transition rates from some up states will result in a CTMC model whose $\text{IAVCD}(t, p)$ measure will bound from above the $\text{IAVCD}(t, p)$ measure of the original model.

Let $\lambda_{\min} = \min_{i\in U'_S} \lambda_i$ and $\lambda_{\max} = \max_{i\in U'_S} \lambda_i$. The control parameter D_C is required to satisfy $1 \le D_C < \lambda_{\max}/\lambda_{\min}$. Note that $\lambda_{\min} > 0$. This follows, for the CTMC models and selections for the regenerative state r covered by bounding regenerative transformation, because S cannot include any absorbing state: by Condition 4 it can include at most one, and by Condition 8 that one should be r, in contradiction with $\lambda_{r,U'_S} > 0$ (Conditions 9 and 12). Then, $U'_S \subset S$ does not include any absorbing state and $\lambda_{\min} > 0$. The bounding regenerative transformation method allows the computation of a lower bound, $\text{IAVCD}^{\text{lb}}(t, p)$, for $\text{IAVCD}(t, p)$; an upper bound, $\text{IAVCD}^{\text{ub}}(t, p)$, for $\text{IAVCD}(t, p)$; or both. The lower bound $\text{IAVCD}^{\text{lb}}(t, p)$ is computed by solving by regenerative transformation with regenerative state r, subset of up states U, and error requirement ε a CTMC, X^{lb}, which differs from X only in that transition rates from states $i \in U'_S$ have been scaled up using $\lambda^{\text{lb}}_{i,j} = \lambda_{i,j}(\lambda^{\text{lb}}_i/\lambda_i)$, $\lambda^{\text{lb}}_i = \max\{\lambda_i, \lambda_{\max}/D_C\}$, where the superscript lb makes reference to quantities defining X^{lb}. The upper bound $\text{IAVCD}^{\text{ub}}(t, p)$ is computed by solving by regenerative transformation with regenerative state r, subset of up states U, and error requirement ε a CTMC, X^{ub}, which differs from X only in that transition rates from states $i \in U'_S$ have been scaled down using $\lambda^{\text{ub}}_{i,j} = \lambda_{i,j}(\lambda^{\text{ub}}_i/\lambda_i)$, $\lambda^{\text{ub}}_i = \min\{\lambda_i, D_C\lambda_{\min}\}$, where the superscript ub makes reference to quantities defining X^{ub}. Figure 5 illustrates bounding regenerative transformation (BRT), showing the two phases of regenerative transformation (RT). The truncated transformed CTMC obtained in the first phase of regenerative transformation applied to X^{lb} is denoted by V^{lb}_T, and the truncated transformed CTMC obtained in the first phase of regenerative transformation applied to X^{ub} is denoted by V^{ub}_T. We will also denote throughout this paper by the superscript lb the objects associated with the generation of V^{lb}_T in the first phase of regenerative transformation and by the superscript ub the objects associated with the generation of V^{ub}_T in the first phase of regenerative transformation.

Note that because larger values of D_C potentially yield smaller values for the output rates from some up states in X^{lb} and potentially yield larger values for the output rates from some up states in X^{ub}, according to Theorem 3, the larger D_C is, the tighter $\text{IAVCD}^{\text{lb}}(t, p)$ and $\text{IAVCD}(t, p)^{\text{ub}}$ can be. In fact, as D_C approaches $\lambda_{\max}/\lambda_{\min}$, both X^{lb} and X^{ub} approach X, and $\text{IAVCD}^{\text{lb}}(t, p)$ and $\text{IAVCD}^{\text{ub}}(t, p)$ become arbitrarily tighter. In addition, for class C'_1 models and $r = o$, X^{lb} and X^{ub} still belong to the class:

THEOREM 4. *For class* C'_1 *models and* $r = o$, X^{lb} *and* X^{ub} *belong to class* C'_1.

PROOF. See the Online Supplement. □

Furthermore, for class C'_1 models and $r = o$, the R'' parameter defined in Theorem 2 for both X^{lb} and X^{ub} is equal to D_C: $R''^{\text{lb}} = \max_{i\in U_S-\{o\}} \lambda^{\text{lb}}_i / \min_{i\in U_S-\{o\}} \lambda^{\text{lb}}_i = \max_{i\in U'_S} \lambda^{\text{lb}}_i / \min_{i\in U'_S} \lambda^{\text{lb}}_i = \lambda_{\max}/(\lambda_{\max}/D_C) = D_C$ and $R''^{\text{ub}} = \max_{i\in U_S-\{o\}} \lambda^{\text{ub}}_i / \min_{i\in U_S-\{o\}} \lambda^{\text{ub}}_i = \max_{i\in U'_S} \lambda^{\text{ub}}_i / \min_{i\in U'_S} \lambda^{\text{ub}}_i = D_C\lambda_{\min}/\lambda_{\min} = D_C$. Then, using Theorem 2, for class C'_1 models and $r = o$, the control parameter D_C should trade off computational cost with bounds tightness. As recently discussed,

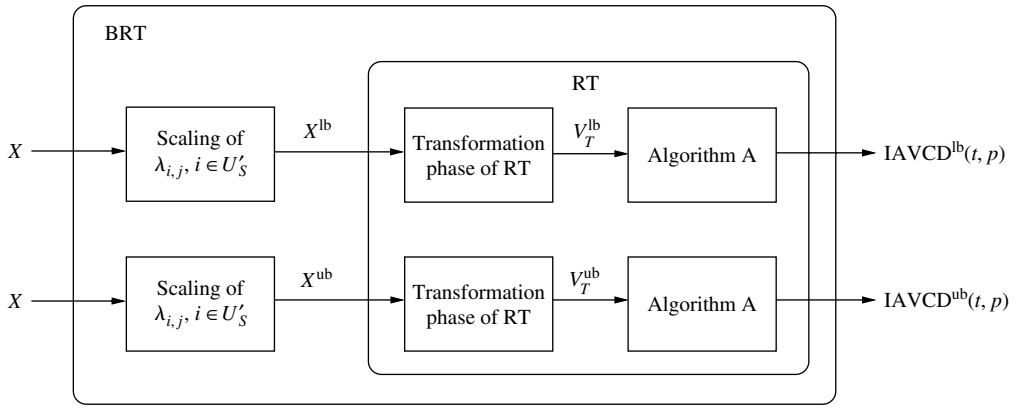

Figure 5 Schematic Representation of Bounding Regenerative Transformation

the larger D_C is, the tighter IAVCD$^{lb}(t, p)$ and IAVCD$^{ub}(t, p)$ can be, and the more costly bounding regenerative transformation should be, since (1) $\Lambda_U^{lb} = (1 + \theta)\max_{i\in U_S} \lambda_i^{lb} = (1 + \theta)\lambda_{\max} = (1 + \theta)\cdot \max_{i\in U_1\cup\cdots\cup U_{N_C}} \lambda_i = (1 + \theta)\max_{i\in U_S} \lambda_i = \Lambda_U$, implying $\Lambda^{lb} = \Lambda$, and $\Lambda_U^{ub} = (1 + \theta)\max_{i\in U_S} \lambda_i^{ub} = (1 + \theta) \cdot D_C\lambda_{\min} = (1 + \theta)D_C \min_{i\in U_1\cup\cdots\cup U_{N_C}} \lambda_i$, which increases with D_C, implying that Λ^{ub} cannot decrease with D_C, $C^{lb} = C(1)$ and C^{ub} can increase with $D_C(1)$; and (2) by Theorem 2, $a_{C^{lb}}^{lb}(n)$, $a_{C^{ub}}^{ub}(n)$ and $a_{C^{lb}}^{\prime lb}(n)$, $a_{C^{ub}}^{\prime ub}(n)$ should decrease asymptotically more slowly with n, implying that K^{lb} and K^{ub} ((2) and (4)) and L^{lb} and L^{ub} (3) should increase. Regenerative transformation can be looked at as the application of bounding regenerative transformation for the computation of either IAVCD$^{lb}(t, p)$ or IAVCD$^{ub}(t, p)$ for $D_C \to \lambda_{\max}/\lambda_{\min}$, but according to the previous reasoning, this implies that for class C'_1 models and $r = o$, bounding regenerative transformation can have a smaller computational cost than regenerative transformation.

For class C'_1 models, $r = o$ and $D_C = 1$, $R''^{lb} = R''^{ub} = 1$. Then, by Theorem 2, $a_{C^{lb}}^{lb}(n)$, $a_{C^{ub}}^{ub}(n)$ and $a_{C^{lb}}^{\prime lb}(n)$, $a_{C^{ub}}^{\prime ub}(n)$ will decrease asymptotically fast with n, and K^{lb}, K^{ub} ((2) and (4)) and L^{lb}, L^{ub} (3) should be small; and, if p is close to 1, so that $\max_{i\in\Omega} \lambda_i tq_{\max}$ is moderate ($\Lambda^{lb}\, tq_{\max} = \Lambda\, tq_{\max}$ and $\Lambda^{ub}\, tq_{\max} \le \Lambda\, tq_{\max}$, since $\Lambda^{lb} = \Lambda$ and Λ^{ub} cannot decrease with D_C and $\Lambda = \lim_{D_C\to\lambda_{\max}/\lambda_{\min}} \Lambda^{ub}$), C^{lb} and C^{ub} will be moderate; then, if X has a large size and $\max_{i\in\Omega} \lambda_i t$ is large, the computational cost of bounding regenerative transformation relative to the model size should be small and much smaller than that of Algorithm A. This is because the first phase of regenerative transformation applied to both X^{lb} and X^{ub} should have a computational cost relative to the size of X much smaller than that of Algorithm A applied to X (see the beginning and end of §2), and because $\Lambda^{lb} = \Lambda$ and $\Lambda^{ub} \le \Lambda$ and since the size of both V_T^{lb} and V_T^{ub} should be moderate, the second phase of regenerative transformation applied to both X^{lb} and X^{ub} should have a computational cost much smaller than the computational cost of Algorithm A applied to X (see the beginning of §2). In the case $\lambda_{\max} = \lambda_{\min}$, no selection for D_C is possible, but in that case regenerative transformation has for class C'_1 models and $r = o$ a parameter $R'' = 1$, and, according to the previous discussion, that method should have a small computational cost relative to the model size when the model is large and $\max_{i\in\Omega} \lambda_i tq_{\max}$ is moderate.

Because regenerative transformation is numerically stable and has good error control (Carrasco 2004), bounding regenerative transformation will compute the bounds with numerical stability and well-controlled error.

3.2. Particular Implementation

The particular case in which both IAVCD$^{lb}(t, p)$ and IAVCD$^{ub}(t, p)$ have to be computed, $D_C = 1$, and if $r \in U_S$, $\lambda_r \le \min_{i\in U'_S} \lambda_i$ allows a computationally more efficient implementation of the bounding regenerative transformation method than the one described in the previous subsection. That more efficient implementation is based on the fact that V_T^{ub} can be obtained from quantities associated with the generation of V_T^{lb} during the first phase of regenerative transformation applied to X^{lb}.

The justification of the particular implementation is elaborated. However, for class C'_1 models and $r = o$, the case $D_C = 1$ is an important one because it is in that case that, if X has a large size and p is close to 1, the method should have a small computational cost relative to the size of X, much smaller than that of Algorithm A. Also, the additional condition $\lambda_r \le \min_{i\in U'_S} \lambda_i$ will be satisfied because of Property 6 of the partition for U_S, and, often, both bounds will be of interest to "bracket" the exact solution of the model.

Recall that for the CTMC models and selections for r covered by bounding regenerative transformation, $U'_S \ne \varnothing$ (by Condition 12), and, according to the review of regenerative transformation in §2, V_T

is defined by the truncation parameters C, K, and, if $\alpha_{S'} > 0$, L; $D_T = \{(n,k)\colon 0 \le k \le K \wedge \max\{0, k-1\} \le n \le k + C - 1\}$; and, if $\alpha_{S'} > 0$, $D'_T = \{(n,k)\colon 0 \le k \le L \wedge \max\{0, k-1\} \le n \le k + C - 1\}$. Let

$$E_{T,u} = \{(n,k)\colon (n,k) \in D_T \wedge k < K \wedge \boldsymbol{\pi}(n,k)^{U_S} \neq \mathbf{0}\},$$

$$E_{T,d} = \{(n,k)\colon (n,k) \in D_T \wedge k < K \wedge \boldsymbol{\pi}(n,k)^{D_S} \neq \mathbf{0}\},$$

and, assuming $\alpha_{S'} > 0$, let

$$E'_{T,u} = \{(n,k)\colon (n,k) \in D'_T \wedge k < L \wedge \boldsymbol{\pi}'(n,k)^{U'_S} \neq \mathbf{0}\},$$

and, if $D'_S \neq \varnothing$, let

$$E'_{T,d} = \{(n,k)\colon (n,k) \in D'_T \wedge k < L \wedge \boldsymbol{\pi}'(n,k)^{D'_S} \neq \mathbf{0}\}.$$

Note that $E_{T,u}$ ($E_{T,d}$) collects the pairs (n,k) corresponding to the states $s^u_{n,k}$ ($s^d_{n,k}$) in Ω_T with $k < K$, and $E'_{T,u}$ ($E'_{T,d}$) collects the pairs (n,k) corresponding to states $s'^u_{n,k}$ ($s'^d_{n,k}$) in Ω_T with $k < L$.

Then, the quantities associated with the generation of V_T^{lb} during the first phase of regenerative transformation applied to X^{lb} that have to be saved are Λ_U^{lb}; Λ_D^{lb}; $a^{\mathrm{lb}}(n,k)$, $(n,k) \in D_T^{\mathrm{lb}}$, $k \ge 2$; if $\alpha_{S'} > 0$, $a'^{\mathrm{lb}}(n,k)$, $(n,k) \in D'^{\mathrm{lb}}_T$, $k \ge 2$; $w^{uu\,\mathrm{lb}}_{n,k}$, $w^{ud\,\mathrm{lb}}_{n,k}$, $(n,k) \in E^{\mathrm{lb}}_{T,u}$; $q^{u\,\mathrm{lb}}_{n,k}$, $(n,k) \in E^{\mathrm{lb}}_{T,u} - \{(0,1)\}$; if $\Omega = S \cup \{f\}$, $v^{u\,\mathrm{lb}}_{n,k}$, $(n,k) \in E^{\mathrm{lb}}_{T,u}$; $w^{du\,\mathrm{lb}}_{n,k}$, $w^{dd\,\mathrm{lb}}_{n,k}$, $(n,k) \in E^{\mathrm{lb}}_{T,d}$; $q^{d\,\mathrm{lb}}_{n,k}$, $(n,k) \in E^{\mathrm{lb}}_{T,d} - \{(0,0)\}$; if $\Omega = S \cup \{f\}$, $v^{d\,\mathrm{lb}}_{n,k}$, $(n,k) \in E^{\mathrm{lb}}_{T,d}$; if $\alpha_{S'} > 0$, $w'^{uu\,\mathrm{lb}}_{n,k}$, $w'^{ud\,\mathrm{lb}}_{n,k}$, $q'^{u\,\mathrm{lb}}_{n,k}$, $(n,k) \in E'^{\mathrm{lb}}_{T,u}$; if $\alpha_{S'} > 0$ and $\Omega = S \cup \{f\}$, $v'^{u\,\mathrm{lb}}_{n,k}$, $(n,k) \in E'^{\mathrm{lb}}_{T,u}$; if $\alpha_{S'} > 0$ and $D'_S \neq \varnothing$, $w'^{du\,\mathrm{lb}}_{n,k}$, $w'^{dd\,\mathrm{lb}}_{n,k}$, $q'^{d\,\mathrm{lb}}_{n,k}$, $(n,k) \in E'^{\mathrm{lb}}_{T,d}$; and, if $\alpha_{S'} > 0$, $D'_S \neq \varnothing$, and $\Omega = S \cup \{f\}$, $v'^{d\,\mathrm{lb}}_{n,k}$, $(n,k) \in E'^{\mathrm{lb}}_{T,d}$.

Construction of V_T^{ub} from those quantities is possible because (1) $C^{\mathrm{ub}} \le C^{\mathrm{lb}}$; $K^{\mathrm{ub}} \le K^{\mathrm{lb}}$; for $\alpha_{S'} > 0$, $L^{\mathrm{ub}} \le L^{\mathrm{lb}}$; $\boldsymbol{\pi}^{\mathrm{ub}}(n,k)^{U_S} \neq \mathbf{0}$ if and only if $\boldsymbol{\pi}^{\mathrm{lb}}(n,k)^{U_S} \neq \mathbf{0}$ and $\boldsymbol{\pi}^{\mathrm{ub}}(n,k)^{D_S} \neq \mathbf{0}$ if and only if $\boldsymbol{\pi}^{\mathrm{lb}}(n,k)^{D_S} \neq \mathbf{0}$; for $\alpha_{S'} > 0$, $\boldsymbol{\pi}'^{\mathrm{ub}}(n,k)^{U'_S} \neq \mathbf{0}$ if and only if $\boldsymbol{\pi}'^{\mathrm{lb}}(n,k)^{U'_S} \neq \mathbf{0}$; and, for $\alpha_{S'} > 0$ and $D'_S \neq \varnothing$, $\boldsymbol{\pi}'^{\mathrm{ub}}(n,k)^{D'_S} \neq \mathbf{0}$ if and only if $\boldsymbol{\pi}'^{\mathrm{lb}}(n,k)^{D'_S} \neq \mathbf{0}$, implying $D_T^{\mathrm{ub}} \subset D_T^{\mathrm{lb}}$, if $\alpha_{S'} > 0$, $D'^{\mathrm{ub}}_T \subset D'^{\mathrm{lb}}_T$, and $\Omega_T^{\mathrm{ub}} \subset \Omega_T^{\mathrm{lb}}$; (2) there exist simple relationships between Λ_U^{ub}, Λ_D^{ub}, Λ^{ub}, $a^{\mathrm{ub}}(n,k)$, $a'^{\mathrm{ub}}(n,k)$, $w^{uu\,\mathrm{ub}}_{n,k}$, $w^{ud\,\mathrm{ub}}_{n,k}$, $q^{u\,\mathrm{ub}}_{n,k}$, $v^{u\,\mathrm{ub}}_{n,k}$, $w^{du\,\mathrm{ub}}_{n,k}$, $w^{dd\,\mathrm{ub}}_{n,k}$, $q^{d\,\mathrm{ub}}_{n,k}$, $v^{d\,\mathrm{ub}}_{n,k}$, $w'^{uu\,\mathrm{ub}}_{n,k}$, $w'^{ud\,\mathrm{ub}}_{n,k}$, $q'^{u\,\mathrm{ub}}_{n,k}$, $v'^{u\,\mathrm{ub}}_{n,k}$, $w'^{du\,\mathrm{ub}}_{n,k}$, $w'^{dd\,\mathrm{ub}}_{n,k}$, $q'^{d\,\mathrm{ub}}_{n,k}$, $v'^{d\,\mathrm{ub}}_{n,k}$ and the corresponding quantities for X^{lb}. Using those relationships and noting that $a^{\mathrm{ub}}_{C^{\mathrm{ub}}}(k) = \sum_{n=k-1}^{k+C^{\mathrm{ub}}-1} a^{\mathrm{ub}}(n,k)$, $k \ge 2$, and, if $\alpha_{S'} > 0$, $a'^{\mathrm{ub}}_{C^{\mathrm{ub}}}(k) = \sum_{n=k-1}^{k+C^{\mathrm{ub}}-1} a'^{\mathrm{ub}}(n,k)$, $k \ge 2$, it is possible to determine (1)–(4) C^{ub}, K^{ub}, and, if $\alpha_{S'} > 0$, L^{ub}, and to build V_T^{ub}. In the remainder of this section we will prove (1) and will obtain the abovementioned relationships. The relationships are established in terms of the parameter $R = \lambda_{\max}/\lambda_{\min}$, with, we remember, $\lambda_{\min} = \min_{i \in U'_S} \lambda_i$ and $\lambda_{\max} = \max_{i \in U'_S} \lambda_i$. Note that $R > 1$, since $D_C < \lambda_{\max}/\lambda_{\min}$ and $D_C = 1$ for the particular implementation to apply.

We start by relating Λ_U, Λ_D, $\Lambda = \max\{\Lambda_U, \Lambda_D\}$ and the transition probabilities of the randomized DTMCs of X^{lb} and X^{ub}.

Theorem 5. *Assume $D_C = 1$, and, if $r \in U_S$, $\lambda_r \le \lambda_{\min}$. Then, $\Lambda_U^{\mathrm{ub}} = \Lambda_U^{\mathrm{lb}}/R$, $\Lambda_D^{\mathrm{ub}} = \Lambda_D^{\mathrm{lb}}$, $\Lambda^{\mathrm{ub}} = \max\{\Lambda_U^{\mathrm{ub}}, \Lambda_D^{\mathrm{ub}}\} \le \Lambda^{\mathrm{lb}}$, if $r \in U_S$, $P^{\mathrm{ub}}_{r,j} = RP^{\mathrm{lb}}_{r,j}$, $j \neq r$, if $r \in D_S$, $P^{\mathrm{ub}}_{r,j} = P^{\mathrm{lb}}_{r,j}$, $j \neq r$, and $P^{\mathrm{ub}}_{i,j} = P^{\mathrm{lb}}_{i,j}$, $i \in S'$.*

Proof. See the Online Supplement. □

Using Theorem 5, it is possible to prove the following result, which relates the vectors $\boldsymbol{\pi}(n,k)$ and $\boldsymbol{\pi}'(n,k)$.

Proposition 1. *Assume $D_C = 1$, and, if $r \in U_S$, $\lambda_r \le \lambda_{\min}$. Then, if $r \in U_S$, $\boldsymbol{\pi}^{\mathrm{ub}}(0,k) = \boldsymbol{\pi}^{\mathrm{lb}}(0,k)$, $0 \le k \le 1$, and $\boldsymbol{\pi}^{\mathrm{ub}}(n,k) = R\,\boldsymbol{\pi}^{\mathrm{lb}}(n,k)$, $n \ge 1$, $0 \le k \le n+1$; if $r \in D_S$, $\boldsymbol{\pi}^{\mathrm{ub}}(n,k) = \boldsymbol{\pi}^{\mathrm{lb}}(n,k)$, $n \ge 0$, $0 \le k \le n+1$; and, for $\alpha_{S'} > 0$, $\boldsymbol{\pi}'^{\mathrm{ub}}(n,k) = \boldsymbol{\pi}'^{\mathrm{lb}}(n,k)$, $n \ge 0$, $0 \le k \le n+1$.*

Proof. See the Online Supplement. □

Note that Proposition 1 implies, as required, that $\boldsymbol{\pi}^{\mathrm{ub}}(n,k)^{U_S} \neq \mathbf{0}$ if and only if $\boldsymbol{\pi}^{\mathrm{lb}}(n,k)^{U_S} \neq \mathbf{0}$, $\boldsymbol{\pi}^{\mathrm{ub}}(n,k)^{D_S} \neq \mathbf{0}$ if and only if $\boldsymbol{\pi}^{\mathrm{lb}}(n,k)^{D_S} \neq \mathbf{0}$, for $\alpha_{S'} > 0$, $\boldsymbol{\pi}'^{\mathrm{ub}}(n,k)^{U'_S} \neq \mathbf{0}$ if and only if $\boldsymbol{\pi}'^{\mathrm{lb}}(n,k)^{U'_S} \neq \mathbf{0}$, and, for $\alpha_{S'} > 0$ and $D'_S \neq \varnothing$, $\boldsymbol{\pi}'^{\mathrm{ub}}(n,k)^{D'_S} \neq \mathbf{0}$ if and only if $\boldsymbol{\pi}'^{\mathrm{lb}}(n,k)^{D'_S} \neq \mathbf{0}$. The following result, relating the quantities $a(n,k)$ and $a'(n,k)$, is an immediate consequence of Proposition 1, taking into account $a(n,k) = \sum_{i \in S} \pi_i(n,k)$ and $a'(n,k) = \sum_{i \in S'} \pi'_i(n,k)$.

Theorem 6. *Assume $D_C = 1$, and, if $r \in U_S$, $\lambda_r \le \lambda_{\min}$. Then, if $r \in U_S$, $a^{\mathrm{ub}}(0,k) = a^{\mathrm{lb}}(0,k)$, $0 \le k \le 1$, and $a^{\mathrm{ub}}(n,k) = R\,a^{\mathrm{lb}}(n,k)$, $n \ge 1$, $0 \le k \le n+1$; if $r \in D_S$, $a^{\mathrm{ub}}(n,k) = a^{\mathrm{lb}}(n,k)$, $n \ge 0$, $0 \le k \le n+1$; and, for $\alpha_{S'} > 0$, $a'^{\mathrm{ub}}(n,k) = a'^{\mathrm{lb}}(n,k)$, $n \ge 0$, $0 \le k \le n+1$.*

The following theorem relates the truncation parameters associated with V_T^{ub} with the truncation parameters associated with V_T^{lb}.

Theorem 7. *Assume $D_C = 1$, and, if $r \in U_S$, $\lambda_r \le \lambda_{\min}$. Then, $C^{\mathrm{ub}} \le C^{\mathrm{lb}}$, $K^{\mathrm{ub}} \le K^{\mathrm{lb}}$, and, for $\alpha_{S'} > 0$, $L^{\mathrm{ub}} \le L^{\mathrm{lb}}$.*

Proof. See the Online Supplement. □

Finally, the following theorem relates the quantities $w^{uu}_{n,k}$, $w^{ud}_{n,k}$, $q^{u}_{n,k}$, $v^{u}_{n,k}$, $w^{du}_{n,k}$, $w^{dd}_{n,k}$, $q^{d}_{n,k}$, $v^{d}_{n,k}$, $w'^{uu}_{n,k}$, $w'^{ud}_{n,k}$, $q'^{u}_{n,k}$, $v'^{u}_{n,k}$, $w'^{du}_{n,k}$, $w'^{dd}_{n,k}$, $q'^{d}_{n,k}$, and $v'^{d}_{n,k}$.

Theorem 8. *Assume $D_C = 1$, and, if $r \in U_S$, $\lambda_r \le \lambda_{\min}$. Then, if $r \in U_S$, $w^{uu\,\mathrm{ub}}_{0,1} = R\,w^{uu\,\mathrm{lb}}_{0,1}$, $w^{ud\,\mathrm{ub}}_{0,1} = R\,w^{ud\,\mathrm{lb}}_{0,1}$, and, if $\Omega = S \cup \{f\}$, $v^{u\,\mathrm{ub}}_{0,1} = R\,v^{u\,\mathrm{lb}}_{0,1}$; if $r \in D_S$, $w^{du\,\mathrm{ub}}_{0,0} = w^{du\,\mathrm{lb}}_{0,0}$, $w^{dd\,\mathrm{ub}}_{0,0} = w^{dd\,\mathrm{lb}}_{0,0}$, and, if $\Omega = S \cup \{f\}$, $v^{d\,\mathrm{ub}}_{0,0} = v^{d\,\mathrm{lb}}_{0,0}$; for $(n,k) \in E^{\mathrm{ub}}_{T,u} - \{(0,1)\}$, $w^{uu\,\mathrm{ub}}_{n,k} = w^{uu\,\mathrm{lb}}_{n,k}$, $w^{ud\,\mathrm{ub}}_{n,k} = w^{ud\,\mathrm{lb}}_{n,k}$, $q^{u\,\mathrm{ub}}_{n,k} = q^{u\,\mathrm{lb}}_{n,k}$, and, if $\Omega = S \cup \{f\}$, $v^{u\,\mathrm{ub}}_{n,k} = v^{u\,\mathrm{lb}}_{n,k}$; and, for $(n,k) \in E^{\mathrm{ub}}_{T,d} - \{(0,0)\}$, $w^{du\,\mathrm{ub}}_{n,k} = w^{du\,\mathrm{lb}}_{n,k}$, $w^{dd\,\mathrm{ub}}_{n,k} = w^{dd\,\mathrm{lb}}_{n,k}$, $q^{d\,\mathrm{ub}}_{n,k} = q^{d\,\mathrm{lb}}_{n,k}$, and, if $\Omega = S \cup \{f\}$, $v^{d\,\mathrm{ub}}_{n,k} = v^{d\,\mathrm{lb}}_{n,k}$. Finally, if $\alpha_{S'} > 0$, for $(n,k) \in E'^{\mathrm{ub}}_{T,u}$, $w'^{uu\,\mathrm{ub}}_{n,k} = w'^{uu\,\mathrm{lb}}_{n,k}$, $w'^{ud\,\mathrm{ub}}_{n,k} = w'^{ud\,\mathrm{lb}}_{n,k}$, $q'^{u\,\mathrm{ub}}_{n,k} = q'^{u\,\mathrm{lb}}_{n,k}$, and, if $\Omega = S \cup \{f\}$, $v'^{u\,\mathrm{ub}}_{n,k} = v'^{u\,\mathrm{lb}}_{n,k}$; and, for $(n,k) \in E'^{\mathrm{ub}}_{T,d}$, $w'^{du\,\mathrm{ub}}_{n,k} = w'^{du\,\mathrm{lb}}_{n,k}$, $w'^{dd\,\mathrm{ub}}_{n,k} = w'^{dd\,\mathrm{lb}}_{n,k}$, $q'^{d\,\mathrm{ub}}_{n,k} = q'^{d\,\mathrm{lb}}_{n,k}$, and, if $\Omega = S \cup \{f\}$, $v'^{d\,\mathrm{ub}}_{n,k} = v'^{d\,\mathrm{lb}}_{n,k}$.*

Proof. See the Online Supplement. □

4. Numerical Analysis

In this section we illustrate, using a representative large model in the class, that, for large class C_1' models with $r = o$ when p is close to 1 so that $\max_{i\in\Omega} \lambda_i t q_{\max}$ is moderate, bounding regenerative transformation with the selection $D_C = 1$ can compute bounds for IAVCD(t, p) with a computational cost smaller than that of regenerative transformation and much smaller relative to the model size than that of Algorithm A when $\max_{i\in\Omega} \lambda_i t$ is large. We also discuss under which conditions the obtained bounds with the selection $D_C = 1$ seem to be tight. Finally, we illustrate the trade-off between bounds tightness and computational cost controlled by the parameter D_C. The experiments were performed taking the value 10^{-4} for the θ parameter defining Λ_U and Λ_D in regenerative transformation (see §2).

The example is a CTMC model of a fault-tolerant storage system made up of 10 five-level RAID subsystems, each one comprising eight disks, two redundant disk controllers (CONT), and two redundant power supplies (PS) (see Figure 6). The power supplies work in cold standby redundancy. The system is up if all RAID subsystems are up. A RAID subsystem is up if, ignoring coverage faults, at least one controller is unfailed, at least one power supply is unfailed, and at least seven disks have updated data (when a failed disk is repaired in an up subsystem, a reconstruction process fills the repaired disk with data consistent with the data stored in the remaining seven disks). Disks in up subsystems fail at a rate of 4×10^{-6} h^{-1} if no disk is under reconstruction and at a rate of 6×10^{-6} h^{-1} if one disk is under reconstruction. Controllers in up subsystems fail at a rate of 2×10^{-5} h^{-1} if the subsystem has two unfailed controllers and at a rate of 3×10^{-5} h^{-1} if the subsystem has one unfailed controller. The active power supply of an up subsystem fails at a rate of 2×10^{-5} h^{-1}. The coverage to controller failures is 0.95, and the coverage to power supply failures is 0.98. Disks are reconstructed at a rate of 0.125 h^{-1}. Components of down RAID subsystems do not fail. It is assumed the availability of an unlimited number of repairmen to repair failed components in up RAID subsystems. However, there is only a repairman to recover down RAID subsystems. The repair rate of failed components in up RAID subsystems is 0.05 h^{-1}, and down subsystems are brought up to a fully operational state with no component failed and all disks containing consistent data at a rate of 0.10 h^{-1}. In case several RAID subsystems are down, the repairman selects at random the one to be brought up. Advantage is taken of the fact that all RAID subsystems have identical behavior to reduce the size of the state space of the model. A more detailed description of the model can be found in Carrasco (2003). The model is quite large: 646,646 states and 15,578,290 transitions. The model has no absorbing state and, thus, illustrates the case $\Omega = S$. A partition for the subset of up states U_S showing that the model is in class C_1' is $U_S = U_0 \cup U_1 \cup \cdots \cup U_{40}$, $U_k = \{s \in U_S : N_C(s) + 2N_D(s) + N_P(s) + N_R(s) = k\}$, where $N_C(s)$ is the number of up RAID subsystems with one failed controller in state s, $N_D(s)$ is the number of up RAID subsystems with one failed disk in state s, $N_P(s)$ is the number of up RAID subsystems with one failed power supply in state s, and $N_R(s)$ is the number of up RAID subsystems with one disk under reconstruction in state s. We will start by assuming that the system is initially in the state where all RAID subsystems are in their fully operational state. That state is the single state o belonging to the subset U_0. The steady-state availability of the system is 0.99975425, making 0.9995 and 0.9999 reasonable choices that we will take for p. All methods were run with a single target (t, p) pair and an error requirement $\varepsilon = 10^{-8}$. The bounding regenerative transformation method is requested to compute both the lower and upper bounds. CPU times are measured (estimated) in (for) a workstation with a Sun-Blade 1000 processor and 4 GB of memory (significantly larger than the memory consumption for all methods). For large t, the CPU times for Algorithm A are enormous; thus, we estimated those CPU times using measured CPU times for smaller t and the approximate flop count of that method given at the beginning of §2.

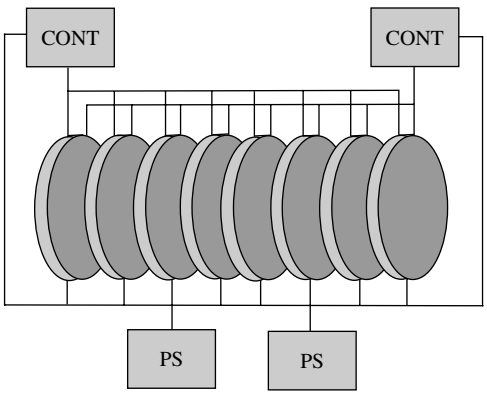

Figure 6 **Architecture of the RAID Subsystem**

We start by considering the selection $D_C = 1$ for bounding regenerative transformation with $r = o$, and, thus, bounding regenerative transformation will use the particular, more efficient implementation discussed in §3.2. For regenerative transformation we also took $r = o$. Table 1 gives the bounds obtained by bounding regenerative transformation (BRT), the values of the truncation parameters C^{lb}

Table 1 Results for BRT with $D_C = 1$, RT, and Algorithm A (A)

		BRT				RT		A	
t(h)	p	IAVCD$^{\text{lb}}(t,p)$	IAVCD$^{\text{ub}}(t,p)$	C^{lb}	K^{lb}	C	K	C'	N
1	0.9995	0.99997543	0.99997600	2	8	2	13	2	15
10	0.9995	0.99975017	0.99975927	3	13	3	47	3	55
100	0.9995	0.99751052	0.99757828	6	15	6	278	5	316
1,000	0.9995	0.97644748	0.97700453	12	16	12	884	11	2,528
10,000	0.9995	0.85732856	0.86048627	36	18	36	1,009	35	23,375
20,000	0.9995	0.81889809	0.82303294	55	18	55	1,041	55	46,241
1	0.9999	0.99997542	0.99997599	2	8	2	13	2	15
10	0.9999	0.99974996	0.99975907	2	13	2	47	2	55
100	0.9999	0.99749956	0.99755676	4	15	4	278	4	316
1,000	0.9999	0.97548885	0.97606827	7	16	7	884	6	2,528
10,000	0.9999	0.79696265	0.80124391	16	18	16	1,009	15	23,375
20,000	0.9999	0.66211670	0.66861207	22	18	22	1,041	21	46,241

and K^{lb} associated with V_T^{lb} (the truncation parameters C^{ub} and K^{ub} associated with V_T^{ub} are nongreater than, respectively, C^{lb} and K^{lb}; see Theorem 7), the values of the truncation parameters C and K associated with V_T in regenerative transformation (RT), and the values of the truncation parameters C' and N of Algorithm A (A), for $p = 0.9995$ and 0.9999 and increasing values of t. Figure 7 gives the CPU times consumed by the methods. As suggested theoretically, K^{lb} has small values. Since $\max_{i\in\Omega} \lambda_i t(1-p) = \max_{i\in\Omega} \lambda_i tq$ has moderate values (for $p = 0.9995$ and $t = 20{,}000$ h, $\max_{i\in\Omega} \lambda_i tq \approx 22.5$) and $\max_{i\in\Omega} \lambda_i^{\text{lb}} tq = \max_{i\in\Omega} \lambda_i tq$, C^{lb} has moderate values. All this makes the CPU times consumed by BRT relatively small: for the largest t considered, 5,494 s (about 92 minutes) for $p = 0.9995$ and 1,883 s (about 31 minutes) for $p = 0.9999$. Since C is identical to C^{lb} (this will always be the case, since, as discussed at the end of §3.1, $\Lambda^{\text{lb}} tq = \Lambda tq$; see (1)); as discussed at the end of §3.1, $\Lambda^{\text{lb}} = \Lambda$ (the maximum output rate in V_T^{lb} is Λ^{lb}), $C^{\text{ub}} \le C^{\text{lb}}$, and $K^{\text{ub}} \le K^{\text{lb}}$; and, as discussed also at the end of that section, $\Lambda^{\text{ub}} \le \Lambda$ (the maximum output rate in V_T^{ub} is Λ^{ub}); the CPU times for RT compared with those of BRT scale approximately as the truncation parameter K scales with K^{lb} and are, therefore, significantly larger for large t: for the largest t considered, 399,853 s (about 111 h) for $p = 0.9995$ and 103,290 s (about 29 h) for $p = 0.9999$. The values of K satisfy the rough upper bound $30R'$ mentioned in Carrasco (2004), since, for the example, $R' \approx 2.25/0.05 = 45$ and $30R' \approx 1{,}350$. Finally, since the model is large, the truncation parameter N is significantly larger than K^{lb} and K for large t, and the truncation parameter C' is very similar to C^{lb} and C (this will almost always be the case; see (1) and Rubino and Sericola 1993), the CPU times of Algorithm A are significantly larger than the CPU times of both BRT and RT: for the largest t considered, the estimated CPU time for Algorithm A is 1.158×10^7 s (about 134 days) for $p = 0.9995$ and 4.482×10^6 s (about 52 days) for $p = 0.9999$, and, thus, for large values of t and a conventional hardware platform, the example can be considered out of reach of Algorithm A.

Figure 8 gives the breakdown of the CPU times consumed by BRT into its three main components: generation of V_T^{lb}, trans (lb); solution of V_T^{lb} by Algorithm A, sol (lb); and solution of V_T^{ub} by Algorithm A, sol (ub).

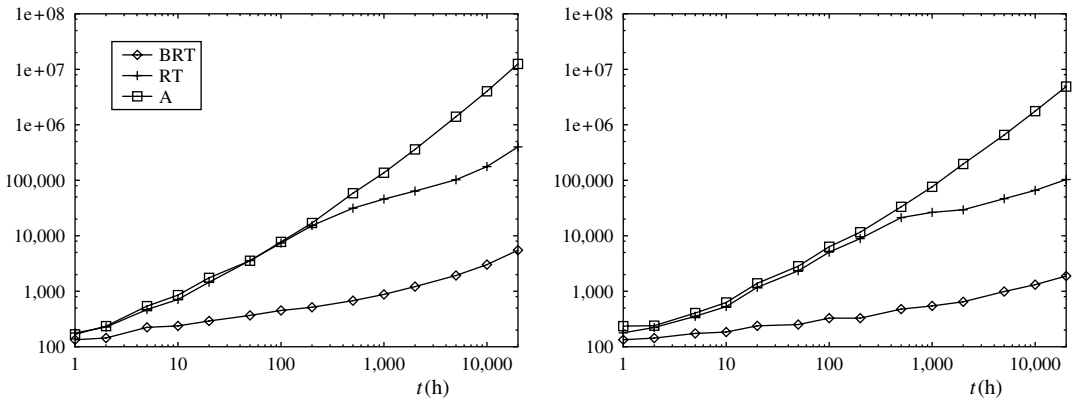

Figure 7 CPU Times of BRT with $D_C = 1$, RT, and Algorithm A (A) for $p = 0.9995$ (Left) and $p = 0.9999$ (Right)

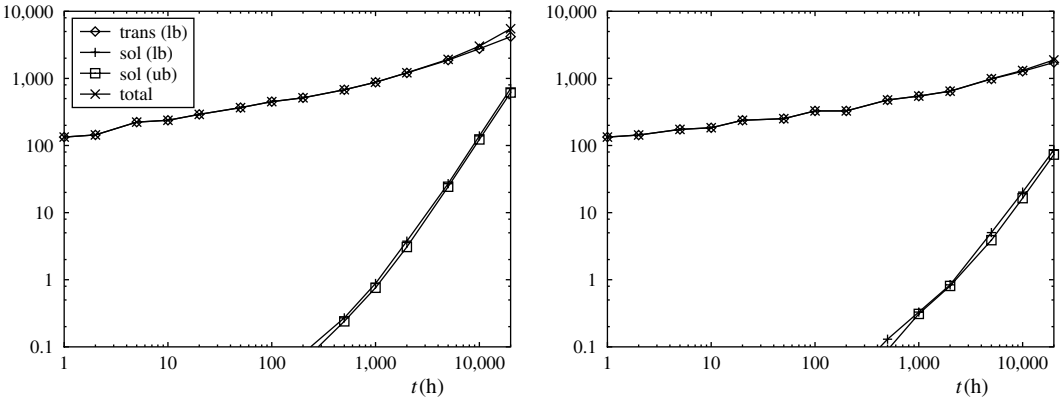

Figure 8 **Breakdown of CPU Times of BRT with $D_C = 1$ for $p = 0.9995$ (Left) and $p = 0.9999$ (Right)**

Scaling of transition rates and generation of V_T^{ub} from quantities associated with the generation of V_T^{lb} consumed negligible CPU times, and those CPU times are not shown. For the considered values of t, most of the CPU times consumed by the method were due to the generation of V_T^{lb}, but the CPU times resulting from the solution by Algorithm A of both V_T^{lb} and V_T^{ub} increase with t faster than the CPU time consumed in the generation of V_T^{lb} and, for large enough t, would dominate the computational cost of the method. As the figure clearly illustrates, the importance of those components also increases with $1-p$.

The bounds obtained by BRT with $D_C = 1$ are quite tight. Intuitively, for large t, this is because all X, X^{lb}, and X^{ub} spend most of the time in U_S in state o, and the three models only differ in the holding times in the states in $U_S - \{o\}$. This will be the case for any class C_1' model provided that the partition for U_S satisfies the additional properties:

Property 7. For each $i \in U_k$, $0 < k \le N_C$,

$$\lambda_{i,\, U_k - \{i\} \cup U_{k+1} \cup \cdots \cup U_{N_C} \cup D_S} \text{ if } \Omega = S$$

or

$$\lambda_{i,\, U_k - \{i\} \cup U_{k+1} \cup \cdots \cup U_{N_C} \cup D_S \cup \{f\}} \text{ if } \Omega = S \cup \{f\}$$

is significantly smaller than $\lambda_{i,\, U_0 \cup \cdots \cup U_{k-1}}$.

Property 8. $\lambda_o \ll \min_{i \in U_1 \cup \cdots \cup U_{N_C}} \lambda_i$.

The reason is that Property 7 implies that, from any state $i \in U_S - \{o\}$, the embedded DTMC will go toward state o with almost one probability and Property 8 implies that each holding time in a state $i \in U_S - \{o\}$ will be much smaller than each holding time in state o. Note that

$$\min_{i \in U_1 \cup \cdots \cup U_{N_C}} \lambda_i^{\text{lb}} = \max_{i \in U_1 \cup \cdots \cup U_{N_C}} \lambda_i > \min_{i \in U_1 \cup \cdots \cup U_{N_C}} \lambda_i \gg \lambda_o$$

and

$$\min_{i \in U_1 \cup \cdots \cup U_{N_C}} \lambda_i^{\text{ub}} = \min_{i \in U_1 \cup \cdots \cup U_{N_C}} \lambda_i \gg \lambda_o.$$

Properties 7 and 8 are satisfied moderately by the example, since for the partition for U_S previously discussed, $\max_{0<k\le 40} \max_{i \in U_k} \lambda_{i,\, U_k - \{i\} \cup U_{k+1} \cup \cdots \cup D_S} = 1.08 \times 10^{-3}\ \text{h}^{-1}$, $\min_{0<k\le 40} \min_{i \in U_k} \lambda_{i,\, U_0 \cup \cdots \cup U_{k-1}} = 0.05\ \text{h}^{-1}$, $\lambda_o = 9.2 \times 10^{-4}\ \text{h}^{-1}$, and $\min_{0<k\le 40} \min_{i \in U_k} \lambda_i \approx 0.05\ \text{h}^{-1}$. Class C_1' models with the additional Properties 7 and 8 for the partition for U_S include F/R models. The fact that the bounds are also tight for small t seems to have to do with the fact that the initial probability of the CTMC in U_S is concentrated in state o. Table 2 gives the bounds obtained by BRT with $D_C = 1$ when the initial state of the CTMC model is the state in which one RAID subsystem has one unfailed controller, no other component failed, and no disk under reconstruction, and the remaining RAID subsystems are in their fully operational state. In that case, the bounds are not so tight for small values of t.

Finally, we analyze the trade-off in BRT between computational cost and bounds tightness controlled by the parameter D_C. As D_C increases, the output rate in X^{lb} from the up state in $U_1 \cup \cdots \cup U_{N_C}$ with minimum output rate in X decreases. Furthermore, as D_C increases, the output rate in X^{ub} from the up state in $U_1 \cup \cdots \cup U_{N_C}$ with maximum output rate in X increases. Then, because X is irreducible, the condition of Theorem 3 ensuring $\text{IAVCD}'(t,p) > \text{IAVCD}(t,p)$ is satisfied, implying that $\text{IAVCD}^{\text{lb}}(t,p)$ will increase strictly as D_C increases and $\text{IAVCD}^{\text{ub}}(t,p)$ will decrease strictly as D_C increases. Table 3 gives the bounds obtained by BRT and the CPU times for $t = 10{,}000$ h, $p = 0.9995$,

Table 2 **Bounds Obtained by BRT with $D_C = 1$ for an Initial Probability Distribution Not Concentrated in State o**

	$p = 0.9995$		$p = 0.9999$	
t(h)	$\text{IAVCD}^{\text{lb}}(t,p)$	$\text{IAVCD}^{\text{ub}}(t,p)$	$\text{IAVCD}^{\text{lb}}(t,p)$	$\text{IAVCD}^{\text{ub}}(t,p)$
1	0.99905631	0.99994872	0.99905616	0.99994870
10	0.99870751	0.99954032	0.99870689	0.99953997
100	0.99647487	0.99703209	0.99645977	0.99701923
1,000	0.97547825	0.97648897	0.97448111	0.97553219
10,000	0.85677215	0.86018960	0.79620927	0.80084126
20,000	0.81853341	0.82283871	0.66154598	0.66830577

Table 3 Trade-off in BRT Between Bounds Tightness and Computational Cost for $t = 10{,}000$ h, $p = 0.9995$, and Initial State o

D_C	$\text{IAVCD}^{\text{lb}}(t,p)$	$\text{IAVCD}^{\text{ub}}(t,p)$	C^{lb}	K^{lb}	C^{ub}	K^{ub}	CPU time (s)
1	0.85732856	0.86048627	36	18	35	18	3,002
2	0.85740339	0.86005160	36	48	35	37	13,429
10	0.85799895	0.85996905	36	277	35	221	80,027
20	0.85869229	0.85996905	36	520	35	452	158,420

and increasing D_C, assuming that the initial state is state o. We also give C^{lb}, K^{lb}, C^{ub}, and K^{ub}. We can note that the bounds become moderately tighter as D_C increases, but, as a result of a significant increase of K^{lb} and K^{ub}, the computational cost of the method increases sharply.

5. Conclusions

We have developed a method for the computation of bounds for the interval availability complementary distribution. The method requires the selection of a regenerative state, is numerically stable, and computes the bounds with well-controlled error. For models belonging to a certain class, class C_1', and a particular, "natural" selection for the regenerative state, the method should trade off bounds tightness with computational cost through a control parameter D_C. For large class C_1' models, the version that should have the smallest computational cost should provide bounds at a small computational cost relative to the model size if the abscissa at which bounds are sought is close to 1, which is the interesting case for fault-tolerant systems. When the model satisfies additional conditions, the bounds obtained by the version of the method that should have the smallest computational cost seem to be tight for any time interval or not small time intervals, depending on whether the initial probability distribution of the model in up states different from the absorbing state, if existing, is concentrated in the natural selection for the regenerative state. Class C_1' models with those additional conditions include F/R models.

Acknowledgments

This research work has been supported by the Ministry of Science and Technology of Spain and FEDER ("Fondo Europeo de Desarrollo Regional") under Research Grant DPI2004-05077 and by the Ministry of Science and Innovation of Spain and FEDER under Research Grant TIN2008–06735–C02–02.

References

Barlow, R. E., F. Proschan. 1981. *Statistical Theory of Reliability and Life Testing: Probability Models*. McArdle Press, Silver Spring, MD.

Carrasco, J. A. 2003. Markovian dependability/performability modeling of fault-tolerant systems. H. Pham, ed. *Handbook of Reliability Engineering*. Springer-Verlag, London, 613–642.

Carrasco, J. A. 2004. Solving large interval availability models using a model transformation approach. *Comput. Oper. Res.* **31**(6) 807–861.

de Souza e Silva, E., H. R. Gail. 1986. Calculating cumulative operational time distributions of repairable computer systems. *IEEE Trans. Comput.* **35**(4) 322–332.

de Souza e Silva, E., P. M. Ochoa. 1992. State space exploration in Markov models. *ACM SIGMETRICS Performance Eval. Rev.* **20**(1) 152–166.

Goyal, A., A. N. Tantawi. 1988. A measure of guaranteed availability and its numerical evaluation. *IEEE Trans. Comput.* **37**(1) 25–32.

Johnson, B. W. 1989. *The Design and Analysis of Fault Tolerant Digital Systems*. Addison-Wesley, Boston.

Kijima, M. 1997. *Markov Processes for Stochastic Modeling*. Chapman & Hall, London.

Kulkarni, V. G. 1995. *Modeling and Analysis of Stochastic Systems*. Chapman & Hall, London.

Ross, S. M. 1983. *Stochastic Processes*. John Wiley & Sons, New York.

Rubino, G., B. Sericola. 1992. Interval availability analysis using operational periods. *Performance Eval.* **14**(3–4) 257–272.

Rubino, G., B. Sericola. 1993. Interval availability distribution computation. *IEEE Internat. Sympos. Fault-Tolerant Comput., Toulouse, France,* IEEE Computer Society Press, Los Alamitos, CA, 48–55.

Rubino, G., B. Sericola. 1995. Interval availability analysis using denumerable Markov processes: Application to multiprocessor subject to breakdowns and repair. *IEEE Trans. Comput.* **44**(2) 286–291.

Sericola, B. 1990. Closed-form solution for the distribution of the total time spent in a subset of a homogeneous Markov process during a finite observation period. *J. Appl. Probab.* **27**(3) 713–719.

Takács, L. 1957. On certain sojourn time problems in the theory of stochastic processes. *Acta Math. Hungar.* **8**(1–2) 169–191.

CORRECTION

In this article, "An Efficient and Numerically Stable Method for Computing Bounds" by Juan Carrasco (*INFORMS Journal on Computing, Articles in Advance*, September 24, 2010, DOI: 10.1287/ijoc.1100.0399), an error was found in the abstract. The second sentence from the end has been corrected to read as "For large models in the class, the version of the method that should have the smallest computational cost should have small computational cost relative to the model size if the value above which the interval availability has to be guaranteed to be is close to 1."