Availability Analysis of Repairable Computer Systems and Stationarity Detection

Bruno Sericola

Abstract—Point availability and expected interval availability are dependability measures respectively defined by the probability that a system is in operation at a given instant and by the mean percentage of time during which a system is in operation over a finite observation period. We consider a repairable computer system and we assume, as usual, that the system is modeled by a finite Markov process. We propose in this paper a new algorithm to compute these two availability measures. This algorithm is based on the classical uniformization technique in which a test to detect the stationary behavior of the system is used to stop the computation if the stationarity is reached. In that case, the algorithm gives not only the transient availability measures, but also the steady state availability, with significant computational savings, especially when the time at which measures are needed is large. In the case where the stationarity is not reached, the algorithm provides the transient availability measures and bounds for the steady state availability. It is also shown how the new algorithm can be extended to the computation of performability measures.

Index Terms—Repairable computer systems, dependability, availability, performability, Markov processes, stationarity detection.

1 Introduction

In the dependability analysis of repairable computing systems, there is an increasing interest in evaluating transient measures, in particular, the point availability and the availability over a given period. This paper deals with the computation of the point availability and of the expected interval availability respectively defined by the probability that the system is in operation at a given instant and by the mean percentage of time during which the system is in operation over a finite observation period. Formally, the system is modeled by a Markov process. Its state space is divided into two disjoint sets which represent the *up* states in which the system delivers the specified service and the *down* states in which there is no more service delivered. Transitions from the *up* (resp. *down*) states to the *down*(resp. *up*) states are called *failures* (resp. *repairs*). The interval availability over $(0, t)$ is then the fraction of the interval $(0, t)$ during which the process is in the *up* states. This random variable has been studied in previous papers as, for instance, in [1], [2], and [3], where its distribution is evaluated using the uniformization technique. This approach is interesting because it has good numerical properties and it allows the user to perform the computation with an error as small as desired.

An approach to detect the stationarity of Markov processes has been proposed in [4], [5]. This approach is based on the uniformization method. The state probability vectors of the uniformized Markov chain are successively computed and the iterates that are spaced m iterations apart are compared. When the difference between two such iterates is small enough, the computation is stopped. The main problem with this method is that, unlike the standard uniformization, there is no ability to specify error bounds easily computable.

In this paper, we develop a new method to compute the point availability and the expected interval availability which is also based on the uniformization technique and on the stationary regime detection. In practice, one usually does not know whether the time horizon he/she is considering is large enough for a steady state analysis. The main advantage of our algorithm is that the computation is stopped when the steady state availability of the system is reached, giving both transient and steady state measures with an error tolerance specified in advance. When the stationarity is not reached, the algorithm gives the transient measures and bounds for the steady state availability.

The remainder of the paper is organized as follows: In the following section, we recall the classical way to compute the point availability and we derive new results to stop the computation when the stationary regime is reached. We also give in this section the pseudocode of both algorithms. In Section 3, we consider the expected interval availability and we show how it can be computed using the stationarity detection. In Section 4, we show, by means of a numerical example, that our new algorithm can considerably reduce the computation time of the availability measures considered here when the time at which measures are needed is sufficiently large. It is also shown that computational savings can be obtained even when the time horizon is small. In Section 5, we show how the results obtained for the availability measures can be easily extended to the corresponding performability measures. The last section is devoted to some conclusions.

2 Point Availability Analysis

Consider an irreducible continuous-time homogeneous Markov process $X = \{X_t, t \geq 0\}$, over a finite state space

• *The author is with IRISA-INRIA, Campus de Beaulieu, 35042 Rennes Cédex, France. E-mail: sericola@irisa.fr.*

For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number 110237.

denoted by S. The states of S are divided into two disjoint subsets: U, the set of the operational states (or the up states), and D, the set of the unoperational states (or the down states). For a system modeled by such a process, the point availability at time t is denoted by $PAV(t)$ and defined by

$$PAV(t) = \Pr\{X_t \in U\}.$$

The process X is, as usual, given by its infinitesimal generator, denoted by A, in which the ith diagonal entry $A(i,i)$ verifies $A(i,i) = -\sum_{j \neq i} A(i,j)$. Its initial probability distribution is denoted by the row vector α.

The uniformized Markov chain associated to the process X is characterized by its uniformization rate ν and by its transition probability matrix P [6]. The uniformization rate ν verifies $\nu \geq \max(-A(i,i); i \in S)$ and P is related to A by $P = I + A/\nu$, where I denotes the identity matrix. Using this notation, we get

$$PAV(t) = \alpha e^{At} \mathbf{1}_U = \sum_{n=0}^{+\infty} e^{-\nu t} \frac{(\nu t)^n}{n!} \alpha P^n \mathbf{1}_U, \tag{1}$$

where $\mathbf{1}_U$ is a column vector whose ith entry is 1 if $i \in U$ and 0 if $i \in D$. We denote by V_n the column vector defined by $V_n = P^n \mathbf{1}_U$. It follows that, for every $n \geq 0$, we have $V_n = PV_{n-1}$ and $V_0 = \mathbf{1}_U$. In the following, we define for every $n \geq 0$, $v_n = \alpha P^n \mathbf{1}_U = \alpha V_n$.

2.1 The Classical Uniformization Method

The classical way to compute the point availability at time t is based on (1). Let ε be a given specified error tolerance and N be defined as

$$N = \min\left\{ n \in \mathbb{N} \;\middle|\; \sum_{j=0}^{n} e^{-\nu t} \frac{(\nu t)^j}{j!} \geq 1 - \varepsilon \right\}. \tag{2}$$

Then, we obtain

$$PAV(t) = \sum_{n=0}^{N} e^{-\nu t} \frac{(\nu t)^n}{n!} v_n + e(N),$$

where the rest of the series $e(N)$ verifies

$$\begin{aligned} e(N) &= \sum_{n=N+1}^{\infty} e^{-\nu t} \frac{(\nu t)^n}{n!} v_n \\ &\leq \sum_{n=N+1}^{\infty} e^{-\nu t} \frac{(\nu t)^n}{n!} \\ &= 1 - \sum_{n=0}^{N} e^{-\nu t} \frac{(\nu t)^n}{n!} \leq \varepsilon. \end{aligned}$$

The computation of integer N can be made without any numerical problems, even for large values of νt, by using the method described in [7].

The truncation level N is, in fact, a function of t, say N_t. For a fixed value of ε, N_t is an increasing function of t. It follows that if we want to compute $PAV(t)$ for J distinct values of t, denoted by $t_1 < \cdots < t_J$, we only need to compute v_n for $n = 1, \ldots, N_{t_J}$ since the values of v_n are independent of the parameter t.

The pseudocode of the classical uniformization method can then be written as shown in Table 1.

TABLE 1
Classical Algorithm for the Computation of $PAV(t)$

input : $\varepsilon, t_1 < \cdots < t_J$

output : $PAV(t_1), \ldots, PAV(t_J)$

Compute N from Relation (2) with $t = t_J$

$V_0 = \mathbf{1}_U;\, v_0 = \alpha V_0$

for $n = 1$ **to** N **do**

  $V_n = PV_{n-1};\, v_n = \alpha V_n$

endfor

for $j = 1$ **to** J **do**

  $PAV(t_j) = \sum_{n=0}^{N} e^{-\nu t_j} \frac{(\nu t_j)^n}{n!} v_n$

endfor

2.2 Stationarity Detection

The stationarity detection that we consider is based on the control of the sequence of vectors $V_n = P^n \mathbf{1}_U$. Let the row vector π denote the stationary probability distribution of the Markov process X. This vector verifies $\pi A = 0$ and $\pi P = \pi$. The steady state availability is given by $PAV(\infty) = \pi \mathbf{1}_U$. To ensure the convergence of the sequence of vectors V_n, we require that the uniformization rate ν verifies $\nu > \max(-A(i,i); i \in S)$ since this guarantees that the transition probability matrix P is aperiodic. We then have, for every $i \in S$,

$$\lim_{n \to \infty} V_n(i) = \pi \mathbf{1}_U.$$

We describe now the test used to detect that, for a given value of n, the entries of vector V_n are close to $\pi \mathbf{1}_U$. For every $n \geq 0$, we define

$$m_n = \min_{i \in S} V_n(i) \;\text{ and }\; M_n = \max_{i \in S} V_n(i).$$

Note that, since $V_0 = \mathbf{1}_U$, we have $M_0 = 1$ and $m_0 = 0$. The following result gives bounds of the steady state availability $PAV(\infty) = \pi \mathbf{1}_U$.

Lemma 2.1. *The sequences m_n and M_n are, respectively, nondecreasing and nonincreasing and, for every $n \geq 0$, we have*

$$\left| v_n - \frac{M_n + m_n}{2} \right| \leq \frac{M_n - m_n}{2} \;\; \text{and}$$
$$\left| \pi \mathbf{1}_U - \frac{M_n + m_n}{2} \right| \leq \frac{M_n - m_n}{2}.$$

Moreover, both sequences m_n and M_n converge to $\pi \mathbf{1}_U$.

Proof. For every $i \in S$, we have $V_{n+1}(i) = \sum_{j \in S} P(i,j) V_n(j)$. It follows that $m_n \leq V_{n+1}(i) \leq M_n$ and, so, we get $m_n \leq m_{n+1}$ and $M_{n+1} \leq M_n$, which shows that the sequences m_n and M_n are, respectively, nondecreasing and nonincreasing.

Since $v_n = \sum_{j \in S} \alpha(j) V_n(j)$, we get $m_n \leq v_n \leq M_n$, which is equivalent to

$$\left|v_n - \frac{M_n + m_n}{2}\right| \leq \frac{M_n - m_n}{2}.$$

Writing now $\pi\mathbf{1}_U = \pi P^n \mathbf{1}_U = \pi V_n = \sum_{j\in S} \pi(j)V_n(j)$, we get, in the same way, $m_n \leq \pi\mathbf{1}_U \leq M_n$, which is equivalent to

$$\left|\pi\mathbf{1}_U - \frac{M_n + m_n}{2}\right| \leq \frac{M_n - m_n}{2}.$$

The state space S being finite and the fact that, for every $i \in S$, $V_n(i)$ converges to $\pi\mathbf{1}_U$ show that both sequences m_n and M_n converge to $\pi\mathbf{1}_U$. □

Remark. *We have assumed that the Markov process X is irreducible. If the Markov process X is not irreducible, but contains an absorbing state denoted by a with $a \in D$, then we have $\pi\mathbf{1}_U = 0$ and, for every $i \in S$, we easily get $V_n(i) \longrightarrow 0$ when $n \longrightarrow \infty$. Now, since $a \in D$, we have $V_n(a) = 0$ for every $n \geq 0$ and so we also have $m_n = 0$ for every $n \geq 0$. Thus, in this case, it sufficies to consider the sequence M_n which is nonincreasing and converges to 0.*

This lemma shows that the difference $M_n - m_n$ converges to 0, that is, for a fixed error tolerance $\varepsilon > 0$, there exists an integer k such that, for $n \geq k$, we have $M_n - m_n \leq \varepsilon$. Since $m_n \leq M_n$, we have $m_n \leq m_{n+1} \leq M_{n+1} \leq M_n$ and, so, the sequence $(M_n - m_n)$ is nonincreasing. We can then define the following integer

$$K = \inf\{n \geq 0 | M_n - m_n \leq \varepsilon/2\}.$$

Using the integer K, (1) can be written as

$$\begin{aligned} PAV(t) = &\sum_{n=0}^{K} e^{-\nu t}\frac{(\nu t)^n}{n!} v_n \\ &+ \frac{M_K + m_K}{2}\left(1 - \sum_{n=0}^{K} e^{-\nu t}\frac{(\nu t)^n}{n!}\right) + e_1(K), \end{aligned} \tag{3}$$

where

$$e_1(K) = \sum_{n=K+1}^{\infty} e^{-\nu t}\frac{(\nu t)^n}{n!} v_n - \frac{M_K + m_K}{2}\sum_{n=K+1}^{\infty} e^{-\nu t}\frac{(\nu t)^n}{n!}.$$

Using Lemma 2.1, the rest $e_1(K)$ verifies

$$|e_1(K)| \leq \sum_{n=K+1}^{\infty} e^{-\nu t}\frac{(\nu t)^n}{n!}\left|v_n - \frac{M_K + m_K}{2}\right| \leq \varepsilon/4. \tag{4}$$

This last inequality follows from the fact that, for $n \geq K$, we have, from Lemma 2.1, $m_K \leq m_n \leq v_n \leq M_n \leq M_K$ and, so, $\left|v_n - \frac{M_K+m_K}{2}\right| \leq \frac{M_K - m_K}{2} \leq \varepsilon/4$.

The time K can be interpreted as the discrete time to stationarity with respect to the subset U.

For every $t \geq 0$ and for every integer $l \geq 0$, we denote by $F_l(t)$ the function defined by

$$F_l(t) = \sum_{n=0}^{l} e^{-\nu t}\frac{(\nu t)^n}{n!}(M_n - m_n).$$

It is easy to check that, for a fixed value of l, the function $F_l(t)$ decreases from 1 to 0 over the interval $[0, \infty[$. We can then define for every integer $l \geq 0$ and for every $\varepsilon > 0$, the time T_l as

$$T_l = \inf\{t \geq 0; F_l(t) \leq \varepsilon/4\}.$$

We then have the following theorem:

Theorem 2.2. *For every $\varepsilon > 0$, for every $t \geq T_K$ we have*

$$|PAV(t) - \pi\mathbf{1}_U| \leq 3\varepsilon/4 \tag{5}$$

$$\left|\pi\mathbf{1}_U - \frac{M_K + m_K}{2}\right| \leq \varepsilon/4 \tag{6}$$

$$\left|PAV(t) - \frac{M_K + m_K}{2}\right| \leq \varepsilon. \tag{7}$$

Proof. First note that, from Lemma 2.1, we have $m_n \leq v_n \leq M_n$ and $m_n \leq \pi\mathbf{1}_U \leq M_n$, for every $n \geq 0$. It follows that $|v_n - \pi\mathbf{1}_U| \leq M_n - m_n$ for every $n \geq 0$. We then have

$$\begin{aligned} |PAV(t) - \pi\mathbf{1}_U| &= \left|\sum_{n=0}^{\infty} e^{-\nu t}\frac{(\nu t)^n}{n!} v_n - \pi\mathbf{1}_U\right| \\ &\leq \sum_{n=0}^{\infty} e^{-\nu t}\frac{(\nu t)^n}{n!}|v_n - \pi\mathbf{1}_U| \\ &\leq \sum_{n=0}^{\infty} e^{-\nu t}\frac{(\nu t)^n}{n!}(M_n - m_n) \\ &= F_K(t) + \sum_{n=K+1}^{\infty} e^{-\nu t}\frac{(\nu t)^n}{n!}(M_n - m_n). \end{aligned}$$

Since $t \geq T_K$, we have $F_K(t) \leq \varepsilon/4$. In the second term, since $n \geq K$, we have $M_n - m_n \leq M_K - m_K \leq \varepsilon/2$ and, so, we get (5).

Relation (6) is immediate from Lemma 2.1. Finally, combining (5) and (6), we get (7). □

The time T_K can be interpreted as the continuous time to stationarity with respect to the subset U.

2.3 The New Algorithm

Using these results, we obtain the following new algorithm (shown in Table 2). To simplify the writing of this algorithm, we define

$$G_l(t) = \sum_{n=0}^{l} e^{-\nu t}\frac{(\nu t)^n}{n!} v_n,$$

$$H_l(t) = 1 - \sum_{n=0}^{l} e^{-\nu t}\frac{(\nu t)^n}{n!},$$

$$S_l = \frac{M_l + m_l}{2}.$$

Note that it is not necessary to compute the continuous time to stationarity T_K with a high precision. It is sufficient to obtain an upper bound of T_K such as, for instance, $\lceil T_K \rceil$, which is the smallest integer greater or equal to T_K.

It must be also noted that, in this algorithm, the truncation step N is a function of the time t_J as in the classical unformization algorithm, but the times to

TABLE 2
Algorithm for the Computation of $PAV(t)$ Using Stationarity Detection

input : $\varepsilon, t_1 < \cdots < t_J$

output : $PAV(t_1), \ldots, PAV(t_J)$

Compute N from Relation (2) with $t = t_J$

$V_0 = 1_U; v_0 = \alpha V_0$

$M_0 = 1; m_0 = 0; K = N + 1$

for $n = 1$ **to** N **do**

  $V_n = PV_{n-1}; v_n = \alpha V_n$

  Compute M_n, m_n and S_n

  if $(M_n - m_n \leq \varepsilon/2)$

    $K = n$; **break**

  endif

endfor

if $(K = N + 1)$

  for $j = 1$ **to** J **do** $PAV(t_j) = G_N(t_j)$ **endfor**

  $m_N \leq PAV(\infty) \leq M_N$

endif

if $(K \leq N)$

  Compute $T_K = \inf\{t \geq 0; F_K(t) \leq \varepsilon/4\}$

  for $j = 1$ **to** J **do**

    if $(t_j \leq T_K)$ **then** $PAV(t_j) = G_K(t_j) + S_K H_K(t_j)$

    if $(t_j > T_K)$ **then** $PAV(t_j) = PAV(\infty) = S_K$

  endfor

endif

stationarity K and T_K are independent of the time parameter, when the discrete time K is reached.

The computational time complexity of both algorithms is essentially due to the computation of the vectors V_n. To compute these vectors, the classical algorithm requires N matrix-vector products and our new algorithm requires only $\min(K, N)$ matrix-vector products.

3 Expected Interval Availability Analysis

We show in this section how the new algorithm proposed above for the point availability computation can be adapted to compute the expected interval availability taking account of the stationarity detection.

The expected interval availability represents the mean percentage of time during which the system is in operation over a finite observation period $(0, t)$. The interval availability over $(0, t)$ is denoted by $IAV(t)$ and its expectation is given by

$$EIAV(t) = \frac{1}{t}\int_0^t PAV(s)ds.$$

Using (1) and by integration over $(0, t)$, we obtain

$$EIAV(t) = \sum_{n=0}^{+\infty} e^{-\nu t}\frac{(\nu t)^n}{n!}\frac{1}{n+1}\sum_{k=0}^{n}\alpha P^k \mathbf{1}_U.$$

We denote by V'_n the column vector defined by

$$V'_n = \frac{1}{n+1}\sum_{k=0}^{n} P^k \mathbf{1}_U,$$

and we define $v'_n = \alpha V'_n$. By definition of V_n and v_n in the previous section, we get, for every $n \geq 0$,

$$V'_n = \frac{1}{n+1}\sum_{k=0}^{n} V_k \quad \text{and} \quad v'_n = \frac{1}{n+1}\sum_{k=0}^{n} v_k.$$

It follows that V'_n and v'_n are recursively given, for $n \geq 1$, by

$$V'_n = \frac{n}{n+1}V'_{n-1} + \frac{1}{n+1}V_n,$$

and

$$v'_n = \frac{n}{n+1}v'_{n-1} + \frac{1}{n+1}v_n, \tag{8}$$

with $V'_0 = V_0 = \mathbf{1}_U$ and, thus, $v'_0 = v_0$. For every $n \geq 0$, we have $0 \leq v'_n \leq 1$. It follows that, using the truncation step N defined in (2), we get the classical algorithm to compute the expected interval availability by writing

$$EIAV(t) = \sum_{n=0}^{N} e^{-\nu t}\frac{(\nu t)^n}{n!}v'_n + e'(N),$$

where

$$e'(N) = \sum_{n=N+1}^{\infty} e^{-\nu t}\frac{(\nu t)^n}{n!}v'_n \leq \sum_{n=N+1}^{\infty} e^{-\nu t}\frac{(\nu t)^n}{n!} = 1 - \sum_{n=0}^{N} e^{-\nu t}\frac{(\nu t)^n}{n!} \leq \varepsilon.$$

This algorithm is basically as the one depicted in Table 1. More precisely the computation of v_n in Table 1 must be followed by the the recursion (8), with $v'_0 = v_0$, and, in the last loop over j, v_n must be replaced by v'_n in order to get $EIAV(t_j)$ instead of $PAV(t_j)$.

3.1 Stationarity Detection for the Expected Interval Availability

Using the results obtained for the point availability, we can derive a new method to obtain the expected interval availability using the stationarity detection. This method is based on the two following theorems. Both theorems will be used in the case where the discrete time to stationarity K is such that $K \leq N$. The first theorem states that, in order to compute the expected interval availability, $EIAV(t)$, we only need the values of v'_n for $n \leq K$. The second theorem states that, in order to compute the expected interval availability, $EIAV(t)$, for $t \geq T_K$, we only need the value $EIAV(t')$ at a time t' such that $t \geq t' \geq T_K$.

We denote by $G'_K(t)$ the function

$$G'_K(t) = \sum_{n=0}^{K} e^{-\nu t} \frac{(\nu t)^n}{n!} v'_n,$$

and recall that

$$H_K(t) = 1 - \sum_{n=0}^{K} e^{-\nu t} \frac{(\nu t)^n}{n!} \text{ and } S_K = \frac{M_K + m_K}{2}.$$

Theorem 3.1. *For every $t \geq 0$, we have*

$$\left| EIAV(t) - \left[G'_K(t) + \frac{K+1}{\nu t} (v'_K - S_K) H_{K+1}(t) + S_K H_K(t) \right] \right| \leq \varepsilon/4. \tag{9}$$

Proof. For every $t \geq 0$, we have

$$EIAV(t) = G'_K(t) + \phi(t),$$

where

$$\phi(t) = \sum_{n=K+1}^{\infty} e^{-\nu t} \frac{(\nu t)^n}{n!} v'_n.$$

For $n \geq K+1$, we have

$$\begin{aligned} v'_n &= \frac{1}{n+1} \sum_{k=0}^{n} v_k \\ &= \frac{1}{n+1} \left[\sum_{k=0}^{K} v_k + \sum_{k=K+1}^{n} v_k \right] \\ &= \frac{K+1}{n+1} v'_K + \frac{1}{n+1} \sum_{k=K+1}^{n} v_k \\ &= \frac{K+1}{n+1} v'_K + \frac{1}{n+1} \sum_{k=K+1}^{n} (v_k - S_K) + \frac{(n-K)}{n+1} S_K \\ &= \frac{K+1}{n+1} v'_K + \frac{(n-K)}{n+1} S_K + x_n, \end{aligned}$$

where

$$\begin{aligned} |x_n| &= \left| \frac{1}{n+1} \sum_{k=K+1}^{n} (v_k - S_K) \right| \\ &\leq \frac{1}{n+1} \sum_{k=K+1}^{n} |v_k - S_K| \\ &\leq \frac{(n-K)}{n+1} \frac{\varepsilon}{4} \leq \varepsilon/4. \end{aligned}$$

The inequality $|v_k - S_K| \leq \varepsilon/4$, for $k \geq K$, follows from Lemma 2.1; it has already been used to bound the error $e_1(K)$ in (4). If $\psi(t)$ is the function defined by

$$\psi(t) = \sum_{n=K+1}^{\infty} e^{-\nu t} \frac{(\nu t)^n}{n!} x_n,$$

we obtain $|\psi(t)| \leq \varepsilon/4$. We then have

$$\phi(t) = \sum_{n=K+1}^{\infty} e^{-\nu t} \frac{(\nu t)^n}{n!} \left(\frac{K+1}{n+1} v'_K + \frac{(n-K)}{n+1} S_K \right) + \psi(t).$$

By writing $(n-K) = n+1-(K+1)$ in this last expression, we get

$$\phi(t) = \frac{K+1}{\nu t} (v'_K - S_K) H_{K+1}(t) + S_K H_K(t) + \psi(t).$$

We then obtain

$$\left| EIAV(t) - \left[G'_K(t) + \frac{K+1}{\nu t} (v'_K - S_K) H_{K+1}(t) + S_K H_K(t) \right] \right| = \psi(t),$$

which completes the proof since $|\psi(t)| \leq \varepsilon/4$. □

Theorem 3.2. *For every $\varepsilon > 0$, for every t and t' such that $t \geq t' \geq T_K$ we have*

$$\left| EIAV(t) - \left[\frac{t'}{t} EIAV(t') + \left(1 - \frac{t'}{t} \right) S_K \right] \right| \leq \varepsilon. \tag{10}$$

Proof. For every t and t' such that $t \geq t' \geq T_K$, we have

$$\begin{aligned} EIAV(t) &= \frac{1}{t} \int_0^t PAV(s) ds \\ &= \frac{1}{t} \left[\int_0^{t'} PAV(s) ds + \int_{t'}^{t} PAV(s) ds \right] \\ &= \frac{1}{t} \left[\int_0^{t'} PAV(s) ds + (t - t') S_K \right. \\ &\qquad \left. + \int_{t'}^{t} [PAV(s) - S_K] ds \right] \\ &= \frac{t'}{t} EIAV(t') + \left(1 - \frac{t'}{t} \right) S_K \\ &\qquad + \frac{1}{t} \int_{t'}^{t} [PAV(s) - S_K] ds. \end{aligned}$$

Using (7), we have, since $t' \geq T_K$

$$\begin{aligned} \left| \frac{1}{t} \int_{t'}^{t} [PAV(s) - S_K] ds \right| &\leq \frac{1}{t} \int_{t'}^{t} |PAV(s) - S_K| ds \\ &\leq \left(1 - \frac{t'}{t} \right) \varepsilon \leq \varepsilon, \end{aligned}$$

which completes the proof. □

Note that Theorem 3.2 is still valid if we replace T_K by $\lceil T_K \rceil$. So, as for the point availability, we can use $\lceil T_K \rceil$ instead of T_K to make easier the computation of the expected interval availability.

Using these two theorems, we obtain a new algorithm to compute the expected interval availability which is similar to the one described in Table 2 for the point availability. It suffices to perform the following changes in the algorithm given in Table 2: The computation of v'_n given by (8) must be added just after the computation of v_n, with $v'_0 = v_0$. The relation $PAV(t_j) = G_K(t_j)$ must be replaced by $EIAV(t_j) = G'_K(t_j)$ and the computations of $PAV(t_j)$ in the case where $K \leq N$ must be replaced by those of $EIAV(t_j)$ given in (9) for $t_j \leq T_K$ and in (10) for $t_j > T_K$. To use (10), we need $EIAV(t)$ for one value of t such that $t \geq T_K$. Such a value can be obtained by using (9) one more time for the smallest value of t_j such that $t_j \geq T_K$. Note that we have the well-known stationary relation $PAV(\infty) = EIAV(\infty)$.

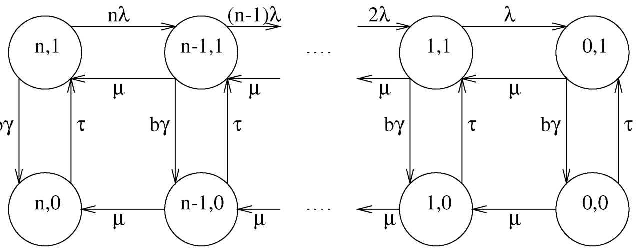

Fig. 1. State-transition diagram for an n-processor system.

4 Numerical Example

We consider a fault-tolerant multiprocessor system with finite buffer stages. This system was first considered in [8] for two processors without repair and has been extended in [9] to include repair for the computation of the moments of performability. Its has been also used in [10] to obtain the distribution of performability. We use the same model here for the computation of the point availability with our new method. It consists of n identical processors and b buffer stages. Processors fail independently at rate λ and are repaired singly with rate μ. Buffers stages fail independently at rate γ and are repaired with rate τ. Processor failures cause a graceful degradation of the system and the number of operational processors is decreased by one. The system is in a failed state when all the processors have failed or any of the buffer stages has failed. No additional processor failures are assumed to occur when the system is in a failed state. The model is represented by a Markov process with state transition diagram shown in Fig. 1. The state space of the system is $S = \{(i,j); 0 \le i \le n,\ j = 0,1\}$. The component i of a state (i,j) means that there are i operational processors and the component j is zero if any of the buffer stages fails; otherwise, it is one. It follows that the set U of operational states is $U = \{(i,1); 1 \le i \le n\}$.

We evaluate the point availability given that the system started in state $(n,1)$. The number of processors is fixed to 16, each with a failure rate $\lambda = 0.01$ per week and a repair rate $\mu = 0.1666$ per hour. The individual buffer stage failure rate is $\gamma = 0.22$ per week and its repair rate is $\tau = 0.1666$ per hour. The error tolerance is $\varepsilon = 0.00001$.

In Fig. 2, we plot the point availability, $PAV(t)$, as a function of t for different values of the number of buffer stages b. The largest value of t, that is, the value of t_J in the algorithm, has been chosen equal to $10,000$ hours.

For the largest value of t, we show, in Fig. 3, the truncation step $N = N_{10000}$, the discrete time to stationarity K, and the continuous time to stationarity T_K (in fact, we give $\lceil T_K \rceil$) for different values of the number of buffer stages b. This figure shows, for example, that when $b = 16$ the classical algorithm needs $3,581$ matrix-vector products and our new algorithm needs only 18 matrix-vector products, the continuous time to stationarity being equal to 77. When $b = 1,024$ the classical algorithm needs $15,616$ matrix-vector products and our new algorithm needs only 86 matrix-vector products, the continuous time to stationarity being equal to 62. Moreover, our algorithm also computes the steady state point availability with a precision equal to $\varepsilon/4$. Fig. 3 also shows that both situations, $K < T_K$ and $K > T_K$, are possible.

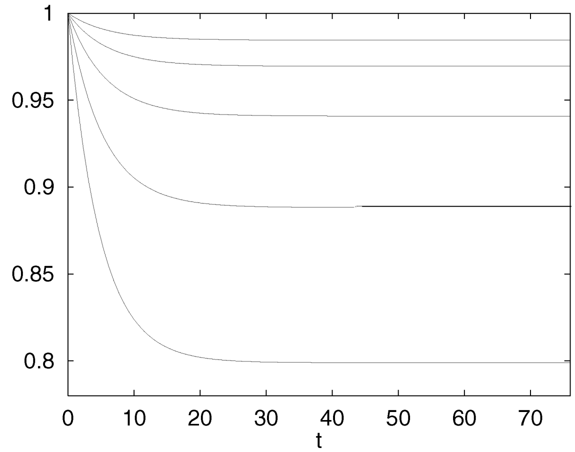

Fig. 2. From top to the bottom: $PAV(t)$ for $b = 2, 4, 8, 16, 32$.

We consider in Fig. 4 smaller values of t_J. The number of buffer stages is fixed to $b = 8$. For $t_J \le 10$, we get $N_{10} \le 14$ and the discrete time to stationarity K is not reached. This means that $K > 14$. For $t_J \ge 20$ we get $N_{t_J} \ge 20$ and the discrete time to stationarity is reached. Its value is $K = 18$ and the continuous time to stationarity is $\lceil T_K \rceil = 80$. Fig. 4 shows that, even for small values of t_J $(t_J < T_K)$, our algorithm can reduce the computation time with respect to the classical algorithm. For instance, when $t_J = 60$, the classical algorithm needs 42 matrix-vector products and our new algorithm needs only 18 matrix-vector products.

5 Extension to the Performability Analysis

The method proposed for the computation of the point availability and the expected interval availability using the steady state availability detection can be extended to more general measures such as the point performability and the expected interval performability.

b	2	4	8	16	32	64	128	256	512	1024
N	3581	3581	3581	3581	3581	3581	3602	5334	8776	15616
K	19	19	18	18	18	18	18	28	48	86
$\lceil T_K \rceil$	81	81	80	78	77	75	77	70	66	62

Fig. 3. Stationarity detection for different numbers of buffer stages.

t_J	10	20	30	40	50	60	70	80	90	100
N_{t_J}	14	20	26	32	37	42	47	51	56	61

Fig. 4. Stationarity detection for small values of the time.

In performability modeling (see, for instance, [8], [9], [10], [11], [12], [13], [14], [15] and the references therein), reward rates are associated with states of the model to quantify the ability of the system to perform in the corresponding states. We denote by $\rho(i)$ the reward rate associated to the state $i \in S$. The reward rates $\rho(i)$ are assumed to be nonnegative real numbers. The point performability at time t, denoted by $PP(t)$, and the expected interval performability, denoted by $EIP(t)$, are defined by

$$PP(t) = \sum_{i \in S} \rho(i) \Pr\{X_t = i\} \text{ and } EIP(t) = \frac{1}{t}\int_0^t PP(s)ds.$$

We define $\rho = \max_{i \in S} \rho(i)$ and $r(i) = \rho(i)/\rho$ and we denote by r the column vector whose ith entry is equal to $r(i)$. We then have $PP(t) = \rho f(t)$ and $EIP(t) = \rho g(t)$, where

$$f(t) = \alpha e^{At} r \text{ and } g(t) = \frac{1}{t}\int_0^t f(s)ds.$$

Since, for every $i \in S$, we have $0 \leq r(i) \leq 1$, all the results and algorithms obtained for the computation of the availability measures can be easily extended to the computation of $f(t)$ and $g(t)$. To do that, it suffices to replace the column vector $\mathbf{1}_U$ by the column vector r. The values M_0 and m_0 become $M_0 = \max_{i \in S} r(i)$ and $m_0 = \min_{i \in S} r(i)$. Moreover, we have $f(\infty) = g(\infty) = \pi r$.

6 Conclusions

A new algorithm has been developed to compute the point availability and the expected interval availability of repairable computer systems modeled by Markov processes. This new algorithm is based on the uniformization technique and on the detection of the steady state availability. It compares favorably with the classical uniformization algorithm when the time horizon is large and it is shown through a numerical example that computational savings can be obtained even when the time horizon is small. Moreover, our algorithm gives the steady state availability if the stationarity is reached and bounds of the steady state availability otherwise. Finally, this method can be easily extended to the computation of more general measures such as the point performability and the expected interval performability.

References

[1] E. de Souza e Silva and H.R. Gail, "Calculating Cumulative Operational Time Distributions of Repairable Computer Systems," *IEEE Trans. Computers,* vol. 35, no. 4, pp. 322-332, Apr. 1986.

[2] G. Rubino and B. Sericola, "Interval Availability Distribution Computation," *Proc. IEEE 23rd Fault-Tolerant Computing Symp.,* pp. 49-55, Toulouse, France, June 1993.

[3] G. Rubino and B. Sericola, "Interval Availability Analysis Using Denumerable Markov Processes: Application to Multiprocessor Subject to Breakdowns and Repair," *IEEE Trans. Computers,* vol. 44, no. 2, pp. 286-291, Feb. 1995.

[4] G. Ciardo, A. Blakemore, P.F. Chimento, J. K. Muppala, and K.S. Trivedi, "Automated Generation and Analysis of Markov Reward Model Using Stochastic Reward Nets," *Linear Algebra, Markov Chains, and Queueing Models,* C.D. Meyer and E.R.J. Plemmons, eds., pp. 145-191, Springer-Verlag, 1993.

[5] M. Malhotra, J.K. Muppala, and K.S. Trivedi, "Stiffness-Tolerant Methods for Transient Analysis of Stiff Markov Chains," *Microelectronics and Reliability,* vol. 34, no. 11, pp. 1,825-1,841, 1994.

[6] S.M. Ross, *Stochastic Processes.* John Wiley & Sons, 1983.

[7] P.N. Bowerman, R.G. Nolty, and E.M. Scheuer, "Calculation of the Poisson Cumulative Distribution Function," *IEEE Trans. Reliability,* vol. 39, pp. 158-161, 1990.

[8] J.F. Meyer, "Closed-Form Solutions for Performability," *IEEE Trans. Computers,* vol. 31, no. 7, pp. 648-657, July 1982.

[9] B.R. Iyer, L. Donatiello, and P. Heidelberger, "Analysis of Performability for Stochastic Models of Fault-Tolerant Systems," *IEEE Trans. Computers,* vol. 35, no. 10, pp. 902-907, Oct. 1986.

[10] V.G. Kulkarni, V.F. Nicola, R.M. Smith, and K.S. Trivedi, "Numerical Evaluation of Performability and Job Completion Time in Repairable Fault-Tolerant Systems," *Proc. IEEE 16th Fault-Tolerant Computing Symp.,* pp. 252-257, Vienna, Austria, July 1986.

[11] G. Ciardo, R. Marie, B. Sericola, and K.S. Trivedi, "Performability Analysis Using Semi-Markov Reward Processes," *IEEE Trans. Computers,* vol. 39, no. 10, pp. 1,251-1,264, Oct. 1990.

[12] R.M. Smith, K.S. Trivedi, and A.V. Ramesh, "Performability Analysis: Measures, an Algorithm, and a Case Study," *IEEE Trans. Computers,* vol. 37, no. 4, pp. 406-417, Apr. 1988.

[13] H. Nabli and B. Sericola, "Performability Analysis: A New Algorithm," *IEEE Trans. Computers,* vol. 45, no. 4, pp. 491-495, Apr. 1996.

[14] E. de Souza e Silva and H.R. Gail, "Calculating Availability and Performability Measures of Repairable Computer Systems Using Randomization," *J. ACM,* vol. 36, pp. 171-193, Jan. 1989.

[15] J.F. Meyer, "On Evaluating the Performability of Degradable Computing Systems," *IEEE Trans. Computers,* vol. 29, no. 8, pp. 720-731, Aug. 1980.

Bruno Sericola received the PhD degree in computer science from the University of Rennes I in 1988. He has been with INRIA (Institute National de Recherche en Informatique en Automatique, (a public French laboratory) since 1989. His main research activity is in computer and communication systems performance evaluation, dependability and performability of fault-tolerant architectures, and applied stochastic processes.