An Algorithm to Calculate Transient Distributions of Cumulative Reward

Edmundo de Souza e Silva[1]
UCLA Computer Science Department
Los Angeles, CA 90024

H. Richard Gail
IBM Thomas J. Watson Research Center
Yorktown Heights, NY 10598

Abstract

Markov reward models have been widely used to solve a variety of problems. In these models, reward rates are associated to the states of a continuous time Markov chain, and impulse rewards are associated to transitions of the chain. Reward rates are gained per unit time in the associated state, and impulse rewards are instantaneous values that are gained each time certain transitions occur. We develop an efficient algorithm to calculate the distribution of the total accumulated reward over a given interval of time when both rate and impulse rewards are present. As special cases, we obtain an algorithm which is used when only rate rewards occur and another algorithm to handle the case of models for which only impulse rewards are present. The development is based purely on probabilistic arguments, and the recursions obtained are simple and have a low computational cost.

[1]This work was done while E. de Souza e Silva was on leave from the Federal University of Rio de Janeiro partially supported by grants from CNPq(Brazil) and NSF CCR-9215064.

1 Introduction

We consider the calculation of the distribution of cumulative reward over a finite interval of time for Markov models in which one or both of the following cases occur: (a) rate based rewards are associated with states; (b) impulse based rewards are associated with transitions between states. Such *performability* models have been studied thoroughly in the past few years. Meyer [10] introduced the general framework for performability modeling, unifying performance and dependability measures to obtain a new measure of the system's "ability to perform." In general, performability evaluation involves the construction of a "base" model whose states indicate that the system is performing at a particular level of accomplishment, and a separate model whose solution provides the reward levels to be associated with the states of the base model. Many measures can be mapped to Meyer's definition of performability, and there has a substantial literature on the subject in the last few years. In [11] a retrospective of past developments in the area is presented together with a comprehensive discussion of many concepts related to performability. In [3] formal definitions of many performability measures are presented together with a survey of solution techniques for their calculation. In [14] several examples which illustrate the use of performability modeling are considered.

The calculation of the distribution of cumulative reward is an interesting theoretical problem that has applications in many areas. For instance, assume that a Markov process is used to model the changes in the structure of a computer system that is subject to failures and repairs. The states of the model may represent, for example, the operational components and the mode of failure of the failed units (see [5]). Assume also that the system can be reconfigured once a failure occurs, so that it is still capable of performing useful work, but perhaps at a lower accomplishment level. Associate a reward rate to a state to represent the amount of work per unit time that can be performed while in that state. The measure to be calculated is then the distribution of total work accumulated over a finite observation period, averaged over time.

The example above illustrates the case when reward rates are associated with states of the base model. However, rewards associated with transitions are also useful in determining many measures of interest. For instance, assume that a reward of 1 is associated with transitions that indicate the occurrence of a given event, e.g., a particular type of system failure or the loss of a packet in a queueing model. All other transitions are associated with a reward of 0. Then the accumulated reward over an observation period is the total number of events that occurred in that period (e.g., the total number of the type of system failure or the total number of packets that were lost).

Certain measures require the solution of a reward model for which both impulse and rate rewards are present. For instance, assume that rewards are associated with the cost of repairing hardware or software components in a system. When a particular component is down, assume there is a cost per unit time that is incurred until the failed unit is brought back to operation. Furthermore, there is a fixed cost that must be paid each time a repairman needs to be called. This fixed cost may be modelled by assigning impulse rewards to certain types of transitions. The measure of interest is the total accumulated cost over a period of time.

In this paper we develop an algorithm for calculating numerically the distribution of the reward accumulated over an observation period for models in which both rate based and impulse based rewards are present. An algorithm which takes into account impulse and rate rewards was considered in [12], using the approach we developed in [2]. Since that algorithm uses the recursion of [2], its cost is combinatorial with the number of different rewards in the model. The algorithm presented here has a drastic reduction in cost compared to the one in [12]. We also show how the algorithm can be specialized to models for which only reward rates are present and models for which only impulse rewards are present. Thus we obtain the rate based algorithm of [4] and the impulse based algorithm of [15] as special cases. We also discuss implementation issues and present examples.

The remainder of the paper is organized as follows. In Section 2 we present the necessary background needed, including a brief description of the general methodology of [2] which gives the foundation for our approach. This section also serves to introduce the notation used throughout the paper. In Section 3 we present the algorithm for calculating the cumulative reward averaged over time, when both reward rates are associated with states and impulse rewards are associated with transitions. The case of only rate based rewards is covered in Section 4, while only impulse rewards are considered in Section 5. In Section 8 the computational complexity of the algorithm is discussed as well as implementation issues. Examples are presented in Section 9, and Section 10 concludes the paper.

2 Preliminaries

In this section we present the necessary background needed for the remaining sections and introduce the notation that is used. We first briefly introduce the uniformization technique, also called randomization or Jensen's method [9], which has been widely applied (see, for example, [1, 3, 6, 7, 8, 13]). Then we present the general methodology of [2] for calculating measures of interest.

Consider a homogeneous continuous time Markov process $\mathcal{X} = \{X(t) : t \geq 0\}$ with finite state space $\mathcal{S} = \{s_i : i = 1, \ldots, M\}$ and transition rate matrix $\mathbf{Q}$. The idea of the uniformization technique is to perform a simple transformation so that $\mathcal{X}$ can be considered as a discrete time Markov chain. Let $q_s = \sum_{s' \in \mathcal{S}} q_{s,s'}$ be the total rate out of state $s \in \mathcal{S}$. Since these output rates are uniformly bounded, a finite rate $\Lambda \geq \max_s\{q_s\}$ exists. We construct a new process such that, for any state $s \in \mathcal{S}$, transitions take place with rate Λ. With probability $q_{s,s'}/\Lambda$, a transition occurs to state $s' \neq s$, and with probability $1 - q_s/\Lambda$, a transition occurs to the same state s. Since all transitions take place with the same rate Λ, we may view $X(t) = Z_{N(t)}$, where $\mathcal{Z} = \{Z_n : n = 0, 1, \ldots\}$ is a discrete time Markov chain with finite state space $\mathcal{S}$ and transition matrix $\mathbf{P} = \mathbf{I} + \mathbf{Q}/\Lambda$ ($\mathbf{I}$ is the identity matrix) and where $\mathcal{N} = \{N(t) : t \geq 0\}$ is a Poisson process of rate Λ that is independent of $\mathcal{Z}$. Thus the transitions are governed by a Poisson process, and the probability of a transition from s to s' is given by the entry $p_{s,s'}$ of $\mathbf{P}$. Using this construction we can easily calculate $\mathbf{p}(t)$, the probability vector for $\mathcal{X}$ at time t, by conditioning on the number of transitions in $(0, t)$. This gives

$$\mathbf{p}(t) = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \boldsymbol{\pi}(n),$$

where $\boldsymbol{\pi}(n) = \boldsymbol{\pi}(0)\mathbf{P}^n$ is the probability vector for the chain $\mathcal{Z}$ after n transitions.

We now review the methodology developed in [2] (see also [3]) to calculate many measures of interest. We assume that there are $K+1$ distinct reward rates $r_1 > r_2 > \cdots > r_{K+1}$, where r_i is the reward accumulated per unit time in any state associated with it. Define $c(s)$ to be the index of the reward associated with state s. Then the instantaneous reward at time τ is a random variable with value $r_{c(X(\tau))}$. The cumulative rate based reward during an observation period $(0, t)$ averaged over t can be expressed in terms of it as

$$\mathrm{ACR}(t) = \frac{1}{t} \int_0^t r_{c(X(\tau))} d\tau.$$

We also assume that there are $\widehat{K}+1$ distinct impulse rewards $\rho_1 > \rho_2 > \cdots > \rho_{\widehat{K}+1}$, where ρ_i is the reward that is gained when a particular transition occurs. The reward gained on a transition from state s to s' of the Markov process $\mathcal{X}$ is $\rho_{\widehat{c}(s',s)}$, where $\widehat{c}(s', s)$ is the index of the reward associated with transition (s', s). Let $N(t)$ be the number of transitions of $\mathcal{X}$ during the interval $(0, t)$, and let σ_n be the nth transition (i.e., $\sigma_n = (s', s)$ if the nth transition occurs from state s' to state s). The impulse reward accumulated during $(0, t)$ averaged over t is

$$\mathrm{ACI}(t) = \frac{1}{t} \sum_{n=0}^{N(t)} \rho_{\widehat{c}(\sigma_n)}.$$

The total cumulative reward during the observation period averaged over t is simply the sum of the accumulated rate based reward and the accumulated impulse based reward, i.e.,

$$\mathrm{ACIR}(t) = \mathrm{ACR}(t) + \mathrm{ACI}(t).$$

In order to make the impulse rewards commensurate with the rate based rewards, we use the normalization $\widehat{r}_i = \rho_i/t$ for $i = 1, \ldots, \widehat{K}+1$. We may assume that $r_{K+1} = 0$

and all other reward rates are positive, since otherwise the simple shift $r_i^* = r_i - r_{K+1}$ may be used to guarantee this. In this case, the total accumulated rewards are related by $\mathrm{ACR}(t) = \mathrm{ACR}^*(t) + r_{K+1}$. The similar assumption $\widehat{r}_{\widehat{K}+1} = 0$ can also be made by replacing $\widehat{r}_i$ with $\widehat{r}_i - \widehat{r}_{\widehat{K}+1}$.

Suppose that the process $\mathcal{X}$ has been uniformized, and that n transitions occur in a given interval $(0, t)$. These transitions divide $(0, t)$ into $n + 1$ subintervals, and the state of $\mathcal{X}$ in these intervals is given by the discrete chain $\mathcal{Z}$ from the random vector $\mathbf{Z}_n = \langle Z_0, \ldots, Z_n \rangle$. The measure of interest $M(t)$ will depend on the rate and impulse rewards attached to states and to transitions, i.e., to pairs of states, in some way. Let $M(t, n, \mathbf{z})$ be the measure we wish to calculate conditioned on $N(t) = n$ transitions in $(0, t)$ and a function $\gamma(\mathbf{Z}_n) = \mathbf{z}$ of the sequence of states of the chain $\mathcal{Z}$. Since the transitions of the uniformized process $\mathcal{X}$ take place according to a Poisson process of rate Λ, we have

$$M(t) = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \sum_{\mathbf{z}} \Delta[n, \mathbf{z}] M(t, n, \mathbf{z}) \tag{1}$$

where

$$\Delta[n, \mathbf{z}] = P[\mathbf{z} | n \text{ transitions}]. \tag{2}$$

We now apply this methodology to the determination of $P[\mathrm{ACIR}(t) \le r]$, the distribution of the cumulative reward averaged over t when both impulse and rate based rewards are present. Given n transitions, recall that the interval $(0, t)$ is split into $n + 1$ subintervals, and we assign a reward rate to each of them according to the state Z_i associated with that interval. Let $\mathbf{k} = \langle k_1, \ldots, k_{K+1} \rangle$ be a vector whose ith component represents the number of intervals in $(0, t)$ associated with reward r_i, and define $\|\mathbf{k}\| = k_1 + \cdots + k_{K+1}$. A specific vector $\mathbf{k}$ is referred to as a *rate coloring*. Similarly, an *impulse coloring* is a vector $\widehat{\mathbf{k}} = \langle \widehat{k}_1, \ldots, \widehat{k}_{\widehat{K}+1} \rangle$ whose ith component represents the number of transitions with reward $\widehat{r}_i$. The total accumulated reward $\mathrm{ACIR}(t)$ depends on an impulse coloring $\widehat{k}$ only through the linear combination

$$\widehat{r} = \widehat{r}(\widehat{\mathbf{k}}) = \frac{1}{t} \sum_{i=1}^{\widehat{K}+1} \rho_i \widehat{k}_i = \sum_{i=1}^{\widehat{K}+1} \widehat{r}_i \widehat{k}_i, \tag{3}$$

i.e., two impulse colorings with identical values $\widehat{r}$ yield the same cumulative impulse reward.

For $n = 0, 1, \ldots$, let $\mathcal{K}_n = \{\mathbf{k} : \|\mathbf{k}\| = n + 1\}$ be the set of rate colorings corresponding to the case of n transitions, and similarly let $\widehat{\mathcal{K}}_n = \{\widehat{\mathbf{k}} : \|\widehat{\mathbf{k}}\| = n\}$ be the corresponding set of impulse colorings. Define $\mathcal{L}_n = \{\widehat{r}(\widehat{\mathbf{k}}) : \widehat{\mathbf{k}} \in \widehat{\mathcal{K}}_n\}$ to be the set of impulse rewards that can be gained from n transitions. From (1) with $M(t) = P[\mathrm{ACIR}(t) \le r]$, we may write

$$P[\mathrm{ACIR}(t) \le r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \sum_{\widehat{r} \in \mathcal{L}_n} \sum_{\mathbf{k} \in \mathcal{K}_n} \Theta[n, \mathbf{k}, \widehat{r}] P[\mathrm{ACIR}(t) \le r | n, \mathbf{k}, \widehat{r}], \tag{4}$$

where $\Theta[n, \mathbf{k}, \hat{r}]$ is the probability, given n transitions, of a rate coloring $\mathbf{k}$ and an average accumulated impulse reward of $\hat{r}$. Now note that

$$\mathrm{ACIR}(t)|n, \mathbf{k}, \hat{r} = \mathrm{ACR}(t)|n, \mathbf{k}, \hat{r} + \mathrm{ACI}(t)|n, \mathbf{k}, \hat{r} = \mathrm{ACR}(t)|n, \mathbf{k} + \hat{r}.$$

If a total average impulse reward of $\hat{r}$ has been accumulated, then the rate rewards must contribute no more than $r - \hat{r}$ in order for the total average reward to be at most r, i.e.,

$$P[\mathrm{ACIR}(t) \leq r|n, \mathbf{k}, \hat{r}] = P[\mathrm{ACR}(t) \leq r - \hat{r}|n, \mathbf{k}]. \tag{5}$$

Since $P[\mathrm{ACR}(t) \leq r - \hat{r}|n, \mathbf{k}] = 0$ for $\hat{r} > r$, (4) becomes

$$P[\mathrm{ACIR}(t) \leq r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \sum_{\hat{r} \leq r} \sum_{\mathbf{k} \in \mathcal{K}_n} \Theta[n, \mathbf{k}, \hat{r}] P[\mathrm{ACR}(t) \leq r - \hat{r}|n, \mathbf{k}]. \tag{6}$$

We now discuss the determination of $P[\mathrm{ACR}(t) \leq r|n, \mathbf{k}]$, the conditional distribution of the total reward accumulated during $(0, t)$ averaged over t when only rate based rewards are present. Given n transitions in the observation period $(0, t)$, let Y_i be the length of the ith subinterval, $i = 1, \ldots, n+1$. It is well-known that the Y_i are exchangeable random variables. As a consequence of the exchangeability property of the Y_i and the independence of the Poisson process $\mathcal{N}$ and the chain $\mathcal{Z}$, it is shown in [2] that intervals with the same reward may be grouped in sequence when finding $P[\mathrm{ACR}(t) \leq r|n, \mathbf{k}]$.

Let $U_{(1)}, U_{(2)}, \ldots, U_{(n)}$ be the order statistics of n independent and identically distributed random variables on $(0, 1)$. It is known that, conditioned on n events during $(0, t)$, the time of the ith transition has the same distribution as $tU_{(i)}$. Therefore, we may identify $Y_i = t(U_{(i)} - U_{(i-1)})$ (where we define for convenience $U_{(0)} = 0$ and $U_{(n+1)} = 1$). If the first nonzero entry in the coloring vector $\mathbf{k}$ is $k_{\xi(1)}$, the second nonzero entry is $k_{\xi(2)}$, etc., and the last nonzero entry is $k_{\xi(L+1)}$ we may make the identification

$$\mathrm{ACR}(t)|n, \mathbf{k} = \\ \frac{1}{t}\left[r_{\xi(1)} t U_{(n_1)} + r_{\xi(2)} t (U_{(n_2)} - U_{(n_1)}) + \cdots + r_{\xi(L)} t (U_{(n_L)} - U_{(n_{L-1})}) + r_{\xi(L+1)} t (1 - U_{(n_L)})\right].$$

where $n_j = \sum_{l=1}^{j} k_{\xi(l)}$ for $j = 1, \ldots, L$. This leads to the equation

$$P[\mathrm{ACR}(t) \leq r|n, \mathbf{k}] = P\left[\sum_{j=1}^{L} (r_{\xi(j)} - r_{\xi(j+1)}) U_{(n_j)} \leq r - r_{\xi(L+1)}\right]. \tag{7}$$

Thus, in order to find $P[\mathrm{ACR}(t) \leq r|n, \mathbf{k}]$, we need to evaluate the distribution of a linear combination of uniform order statistics on $(0, 1)$. The results of [16] enable this to be carried out when the coloring vector $\mathbf{k}$ has at least two nonzero entries, and the largest and smallest rewards corresponding to nonzero entries satisfy $r_{\xi(1)} \geq r \geq r_{\xi(L+1)}$. However, we require an expression for the conditional distribution $P[\mathrm{ACR}(t) \leq r | n, \mathbf{k}]$ for general $\mathbf{k}$.

For $n \geq 0$, $\mathbf{k} \in \mathcal{K}_n$, $l \geq 0$, $i = 1, \ldots, K+1$, define the functions

$$f_i[x, n, \mathbf{k}, r, l] = \frac{1}{l!} \cdot \frac{d^l}{dx^l} \left\{ \frac{(x-r)^n}{\prod_{\substack{j=1 \\ j \neq i}}^{K+1} (x - r_j)^{k_j}} \right\}. \tag{8}$$

Also define $f_i[x, n, \mathbf{k}, r, l] = 0$ for $l < 0$. An expression for the conditional distribution of $\mathrm{ACR}(t)$ is given in terms of the functions f_i in the following lemma, which is proved in [4].

Lemma 1 *For $n \geq 0$ and $\mathbf{k} \in \mathcal{K}_n$, we have*

$$P[\mathrm{ACR}(t) \leq r | n, \mathbf{k}] = \sum_{i: r_i \leq r} f_i[r_i, n, \mathbf{k}, r, k_i - 1]. \tag{9}$$

3 Impulse and Rate Rewards

In this section we derive a recursive expression for calculating the cumulative reward averaged over t when both rate based and impulse based rewards occur in the same model. The approach for obtaining the results generalizes the development of [4] for the case of rate based rewards. In fact, we will show how to obtain the algorithm for the rate based case by specializing the recursion developed here.

From (6) and (9), we may express the distribution of total accumulated reward during $(0, t)$ averaged over t as

$$P[\mathrm{ACIR}(t) \leq r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \sum_{\widehat{r} \leq r} \sum_{\mathbf{k} \in \mathcal{K}_n} \Theta[n, \mathbf{k}, \widehat{r}] \sum_{i: r_i \leq r - \widehat{r}} f_i[r_i, n, \mathbf{k}, r - \widehat{r}, k_i - 1]. \tag{10}$$

First consider the calculation of $\Theta[n, \mathbf{k}, \widehat{r}]$, and let $\Theta_s[n, \mathbf{k}, \widehat{r}]$ be the probability, given n transitions, of a rate coloring $\mathbf{k}$, an average accumulated impulse reward of $\widehat{r}$, *and* the state visited after the last transition is s. Then, we obtain

$$\Theta_s[n, \mathbf{k}, \widehat{r}] = \sum_{s' \in \mathcal{S}} p_{s's} \Theta_{s'}[n-1, \mathbf{k} - \mathbf{1}_{c(s)}, \widehat{r} - \widehat{r}_{\widehat{c}(s', s)}]. \tag{11}$$

That is, if the nth transition of the chain $\mathcal{Z}$ occurs from s' to s, then the entry of the rate coloring vector corresponding to state s is incremented by 1, while the reward associated to the transition (s', s) is added to the total accumulated impulse reward. The initial conditions are

$$\Theta_s[0, \mathbf{1}_i, 0] = \begin{cases} \pi_s(0) & \text{if } i = c(s) \\ 0 & \text{if } i \neq c(s) \end{cases} \tag{12}$$

From the definition of $\Theta_s[n, \mathbf{k}, \widehat{r}]$, we have

$$\Theta[n, \mathbf{k}, \widehat{r}] = \sum_{s \in \mathcal{S}} \Theta_s[n, \mathbf{k}, \widehat{r}]. \tag{13}$$

To find a recursion for the right hand side of (10), it is convenient to define the functions

$$\varphi_i[x, \mathbf{k}, l] = \frac{1}{l!} \cdot \frac{d^l}{dx^l} \left\{ \frac{1}{\prod_{\substack{j=1 \\ j \neq i}}^{K+1} (x - r_j)^{k_j}} \right\} \tag{14}$$

and

$$\psi_n[x, r, l] = \frac{1}{l!} \cdot \frac{d^l}{dx^l} \left\{ (x - r)^n \right\}. \tag{15}$$

We also define $\varphi_i[x, \mathbf{k}, l] = 0 = \psi_n[x, r, l]$ for $l < 0$. Differentiating repeatedly, we find

$$f_i[r_i, n, \mathbf{k}, r, k_i - 1] = \sum_{l=0}^{k_i - 1} \psi_n[x, r, l] \varphi_i[x, \mathbf{k}, k_i - 1 - l]. \tag{16}$$

Therefore, from (10) the distribution of $\mathrm{ACIR}(t)$ is given by

$$P[\mathrm{ACIR}(t) \leq r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \sum_{\widehat{r} \leq r} \sum_{\mathbf{k} \in \mathcal{K}_n} \Theta[n, \mathbf{k}, \widehat{r}] \sum_{i : r_i \leq r - \widehat{r}} \sum_{l=0}^{k_i - 1} \psi_n[x, r - \widehat{r}, l] \varphi_i[x, \mathbf{k}, k_i - 1 - l]. \tag{17}$$

The following lemmas give recursions for the functions $\varphi_i[x, \mathbf{k}, l]$ and $\psi_n[x, r, l]$.

Lemma 2 *Let j be an index such that $k_j \geq 1$. For $\|\mathbf{k}\| = n + 1 > 1$ $(n \geq 1)$*

$$\varphi_i[x, \mathbf{k}, l] = \begin{cases} \dfrac{1}{x - r_j} \left(\varphi_i[x, \mathbf{k} - \mathbf{1}_j, l] - \varphi_i[x, \mathbf{k}, l - 1] \right) & i \neq j \\ \\ \varphi_i[x, \mathbf{k} - \mathbf{1}_j, l] & i = j \end{cases} \tag{18}$$

For $\|\mathbf{k}\| = n + 1 = 1$ $(n = 0,\ k_j = 1)$ and $l > 0$

$$\varphi_i[x, \mathbf{k}, l] = \begin{cases} -\left(\dfrac{1}{x - r_j} \right) \varphi_i[x, \mathbf{k}, l - 1] & i \neq j \\ \\ 0 & i = j \end{cases} \tag{19}$$

The initial conditions are (for $n = 0$, $k_j = 1$, $l = 0$)

$$\varphi_i[x, \mathbf{k}, 0] = \begin{cases} \dfrac{1}{x - r_j} & i \neq j \\ \\ 1 & i = j \end{cases} \tag{20}$$

Proof: First consider the case $n \geq 1$. We prove (18) by induction on l. We have from the definition of $\varphi_i[x, \mathbf{k}, l]$ in (14) that

$$\varphi_i[x, \mathbf{k}, 0] = \begin{cases} \left(\dfrac{1}{x - r_j}\right) \varphi_i[x, \mathbf{k} - \mathbf{1}_j, 0] & i \neq j \\ \\ \varphi_i[x, \mathbf{k} - \mathbf{1}_j, 0] & i = j \end{cases} \tag{21}$$

which is (18) for $l = 0$.

Now we assume that (18) is valid for $l - 1$ and prove it for l. Rewriting (18) for the case $i \neq j$ we have

$$(x - r_j)\varphi_i[x, \mathbf{k}, l - 1] + \varphi_i[x, \mathbf{k}, l - 2] = \varphi_i[x, \mathbf{k} - \mathbf{1}_j, l - 1]. \tag{22}$$

Differentiating both sides of (22) gives

$$\varphi_i[x, \mathbf{k}, l - 1] + (x - r_j) l \varphi_i[x, \mathbf{k}, l] + (l - 1)\varphi_i[x, \mathbf{k}, l - 1] = l \varphi_i[x, \mathbf{k} - \mathbf{1}_j, l]. \tag{23}$$

Combining terms and dividing the resulting equation by $l(x - r_j)$, we obtain (18) for $i \neq j$.

Equation (18) for $i = j$ is proved in a similar manner.

Now consider the case $n = 0$. The initial conditions (20) are obtained directly from (14). Then equation (19) is trivially obtained by taking the derivatives of (20). $\square$

Lemma 3 *For $n \geq 1$*

$$\psi_n[x, r, l] = \psi_{n-1}[x, r, l - 1] + (x - r)\psi_{n-1}[x, r, l]. \tag{24}$$

The initial conditions are

$$\psi_0[x, r, l] = \begin{cases} 1 & \text{for } l = 0 \\ 0 & \text{for } l > 0 \end{cases} \tag{25}$$

Proof: We show (24) by induction on l. From (15) we have $\psi_n[x, r, 0] = (x - r)\psi_{n-1}[x, r, 0]$ for $n \geq 1$, which is equation (24) for $l = 0$. Now assume that (24) is valid for $l - 1$.

Differentiating both sides of the resulting equation and combining terms yields equation (24) for l. The initial conditions are obtained directly from the definition. $\square$

Although the recursions for Θ_s, ψ_n, and φ_i are sufficient to evaluate $P[\mathrm{ACIR}(t) \leq r]$ using (17), they would lead to computational requirements that are combinatorial in the number of rewards. However, we can find a simple recursion that does not require evaluating a combinatorial number of terms. We show that the recursions for φ_i and Θ_s can be combined into a single recursion by grouping the rate colorings for each n in a certain manner. To this end, we introduce partitions of $\mathcal{K}_n = \{\mathbf{k} : \|\mathbf{k}\| = n+1\}$ as indicated below. These partitions, which were first defined in [4], are the key to obtaining an efficient recursion for $P[\mathrm{ACIR}(t) \leq r]$.

Definition 1 *For $n \geq 0$, $i = 1, \ldots, K+1$, let*

$$G_g[i,n] = \{\mathbf{k} \in \mathcal{K}_n : k_i = g\}, \qquad g = 0, \ldots, n+1. \tag{26}$$

We find recursions over the sets $G_g[i,n]$ instead of recursions that consider each individual vector $\mathbf{k}$ separately. We first define a new quantity based on the partitions defined above.

Definition 2 *For $n \geq 0$, $i = 1, \ldots, K+1$, $s \in \mathcal{S}$, $l \geq 0$, $g = 0, \ldots, n+1$, $\hat{r} \geq 0$, let*

$$\Phi_s[g,i,n,l,\hat{r}] = \sum_{\mathbf{k} \in G_g[i,n]} \Theta_s[n,\mathbf{k},\hat{r}]\varphi_i[r_i,\mathbf{k},l]. \tag{27}$$

For $n \geq 0$, $i = 1, \ldots, K+1$, $s \in \mathcal{S}$, let

$$F_{g,s}[i,n] = \{\mathbf{k} \in G_g[i,n] : k_{c(s)} > 0\}, \qquad g = 0, \ldots, n+1. \tag{28}$$

It was shown in [4] that

$$\{\mathbf{k} - \mathbf{1}_{c(s)} : \mathbf{k} \in F_{g,s}[i,n]\} = \begin{cases} G_g[i,n-1] & i \neq c(s), g \geq 0 \\ G_{g-1}[i,n-1] & i = c(s), g > 0 \end{cases} \tag{29}$$

Since $\Theta_s[n,\mathbf{k},\hat{r}] = 0$ if $k_{c(s)} = 0$, we may write

$$\Phi_s[g,i,n,l,\hat{r}] = \sum_{\mathbf{k} \in F_{g,s}[i,n]} \Theta_s[n,\mathbf{k},\hat{r}]\varphi_i[r_i,\mathbf{k},l]. \tag{30}$$

Let $\omega_{ij} = r_i - r_j$ for $i \neq j$. The following lemma gives a simple recursion for $\Phi_s[g,i,n,l,\hat{r}]$.

Lemma 4 *Let $s \in \mathcal{S}$, $i = 1, \ldots, K+1$, $g = 0, \ldots, n+1$, $\hat{r} \geq 0$. For $n \geq 1$, $l \geq 0$ and for $n = 0$, $l > 0$*

$$\Phi_s[g,i,n,l,\hat{r}] =$$
$$\begin{cases} \left(\frac{1}{\omega_{i,c(s)}}\right)\left(\sum_{s' \in \mathcal{S}} p_{s's}\Phi_{s'}[g,i,n-1,l,\hat{r}-\hat{r}_{\widehat{c}(s',s)}] - \Phi_s[g,i,n,l-1,\hat{r}]\right) & i \neq c(s) \\ \sum_{s' \in \mathcal{S}} p_{s's}\Phi_{s'}[g-1,i,n-1,l,\hat{r}-\hat{r}_{\widehat{c}(s',s)}] & i = c(s) \end{cases} \tag{31}$$

The initial conditions are (for $n = 0$, $l = 0$)

$$\Phi_s[g,i,0,0,0] = \begin{cases} \pi_s(0)/\omega_{i,c(s)} & i \neq c(s), g = 0 \\ 0 & i \neq c(s), g = 1 \\ 0 & i = c(s), g = 0 \\ \pi_s(0) & i = c(s), g = 1 \end{cases} \tag{32}$$

Proof: First, we assume that $n \geq 1$ and $i \neq c(s)$. From the definition of Φ_s and using (11) and (18), we have

$$\begin{aligned} \Phi_s[g,i,n,l,\hat{r}] = & \\ & \sum_{\mathbf{k} \in F_{g,s}[i,n]} \sum_{s' \in \mathcal{S}} p_{s's}\Theta_{s'}[n-1,\mathbf{k}-\mathbf{1}_{c(s)},\hat{r}-\hat{r}_{\widehat{c}(s',s)}]\left(\frac{1}{\omega_{i,c(s)}}\right)\varphi_i[r_i,\mathbf{k}-\mathbf{1}_{c(s)},l] \\ & - \sum_{\mathbf{k} \in F_{g,s}[i,n]} \Theta_s[n,\mathbf{k},\hat{r}]\left(\frac{1}{\omega_{i,c(s)}}\right)\varphi_i[r_i,\mathbf{k},l-1]. \end{aligned} \tag{33}$$

Exchanging the order of summation in the first term of (33) and using (29), we obtain

$$\begin{aligned} \Phi_s[g,i,n,l,\hat{r}] &= \left(\frac{1}{\omega_{i,c(s)}}\right)\sum_{s' \in \mathcal{S}} p_{s's} \sum_{\mathbf{k} \in G_g[i,n-1]} \Theta_{s'}[n-1,\mathbf{k},\hat{r}-\hat{r}_{\widehat{c}(s',s)}]\varphi_i[r_i,\mathbf{k},l] \\ &\quad - \left(\frac{1}{\omega_{i,c(s)}}\right)\sum_{\mathbf{k} \in F_{g,s}[i,n]} \Theta_s[n,\mathbf{k},\hat{r}]\varphi_i[r_i,\mathbf{k},l-1]. \end{aligned} \tag{34}$$

From (27), recognizing the terms Φ_s, we obtain (31) for $i \neq c(s)$.

We can derive (31) for $n \geq 1$ and $i = c(s)$ in a similar way. In this case, the definition of Φ_s, (11) and (18) yield

$$\Phi_s[g,i,n,l,\hat{r}] = \sum_{\mathbf{k} \in F_{g,s}[i,n]} \sum_{s' \in \mathcal{S}} p_{s's}\Theta_{s'}[n-1,\mathbf{k}-\mathbf{1}_{c(s)},\hat{r}-\hat{r}_{\widehat{c}(s',s)}]\varphi_i[r_i,\mathbf{k}-\mathbf{1}_{c(s)},l]. \tag{35}$$

The result follows immediately after applying (29).

The initial conditions are easily obtained from (12) and (20). $\square$

We next define a further aggregation.

Definition 3 *For $n \geq 0$, $i = 1, \ldots, K+1$, $s \in \mathcal{S}$, $u \geq 0$, $\widehat{r} \geq 0$, let*

$$\Psi_s[i,n,u,\widehat{r}] = \sum_{g=0}^{n+1} \Phi_s[g,i,n,g+u-(n+1),\widehat{r}]. \tag{36}$$

We now find a recursion for $\Psi_s[i,n,u,\widehat{r}]$.

Lemma 5 *Let $s \in \mathcal{S}$, $i = 1, \ldots, K+1$. For $n \geq 1$, $u \geq 0$ and for $n = 0$, $u > 1$*

$$\Psi_s[i,n,u,\widehat{r}] =$$
$$\begin{cases} \left(\dfrac{1}{\omega_{i,c(s)}}\right)\left(\displaystyle\sum_{s' \in \mathcal{S}} p_{s's}\Psi_{s'}[i,n-1,u-1,\widehat{r}-\widehat{r}_{\widehat{c}(s',s)}] - \Psi_s[i,n,u-1,\widehat{r}]\right) & i \neq c(s) \\ \displaystyle\sum_{s' \in \mathcal{S}} p_{s's}\Psi_{s'}[i,n-1,u,\widehat{r}-\widehat{r}_{\widehat{c}(s',s)}] & i = c(s) \end{cases} \tag{37}$$

The initial conditions are (for $n = 0$, $u = 0, 1$)

$$\Psi_s[i,0,u,0] = \begin{cases} 0 & i \neq c(s), u = 0 \\ \pi_s(0)/\omega_{i,c(s)} & i \neq c(s), u = 1 \\ \pi_s(0) & i = c(s), u = 0 \\ 0 & i = c(s), u = 1 \end{cases} \tag{38}$$

Proof: For $i \neq c(s)$, applying (31) yields

$$\begin{aligned} \Psi_s[i,n,u,\widehat{r}] &= \left(\frac{1}{\omega_{i,c(s)}}\right)\left\{\sum_{s' \in \mathcal{S}} p_{s's}\sum_{g=0}^{n+1}\Phi_{s'}[g,i,n-1,g+u-(n+1),\widehat{r}-\widehat{r}_{\widehat{c}(s',s)}]\right. \\ &\quad \left. - \sum_{g=0}^{n+1}\Phi_s[g,i,n,g+u-(n+1)-1,\widehat{r}]\right\}. \end{aligned} \tag{39}$$

Since $\Phi_{s'}[n+1,i,n-1,l,\widehat{r}] = 0$ for all l, the first sum over g in (39) can be rewritten as a sum from 0 to n. Using $g+u-(n+1) = g+(u-1)-n$, this is equal to $\Psi_{s'}[i,n-1,u-1,\widehat{r}-\widehat{r}_{\widehat{c}(s',s)}]$ from (36). The second sum in (39) is clearly $\Psi_s[i,n,u-1,\widehat{r}]$ from (36).

The case $i = c(s)$ is simpler. Summing (31) from $g = 0$ to $n+1$ with $l = g+u-(n+1)$ gives

$$\Psi_s[i,n,u,\widehat{r}] = \sum_{s' \in \mathcal{S}} p_{s's}\sum_{g=0}^{n+1}\Phi_{s'}[g-1,i,n-1,g+u-(n+1),\widehat{r}-\widehat{r}_{\widehat{c}(s',s)}]. \tag{40}$$

Since the term for $g = 0$ is 0, using $g + u - (n+1) = g - 1 + u - n$ the above sum on g is equal to $\Psi_{s'}[i, n-1, u, \widehat{r} - \widehat{r}_{\widehat{c}(s',s)}]$.

The initial conditions follow immediately from (32). □

Define the vector $\boldsymbol{\Psi}[i,n,u,\widehat{r}] = \langle \Psi_{s_1}[i,n,u,\widehat{r}], \ldots, \Psi_{s_M}[i,n,u,\widehat{r}] \rangle$. The next theorem shows that the distribution of total reward when both impulse and rate rewards are present can be calculated from $\boldsymbol{\Psi}$ and ψ_n.

Theorem 1 *The distribution of the total reward accumulated during* $(0,t)$ *averaged over* t *is given by*

$$P[\mathrm{ACIR}(t) \le r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \sum_{\widehat{r} \le r} \sum_{i: r_i \le r - \widehat{r}} \sum_{u=0}^{n} \psi_n[r_i, r - \widehat{r}, u] \| \Psi[i, n, n-u, \widehat{r}] \|. \tag{41}$$

Proof: From (17) we have

$$P[\mathrm{ACIR}(t) \le r | n] = \sum_{\widehat{r} \le r} \sum_{\mathbf{k} \in \mathcal{K}_n} \sum_{s \in \mathcal{S}} \Theta_s[n, \mathbf{k}, \widehat{r}] \sum_{i: r_i \le r - \widehat{r}} \sum_{u=0}^{k_i - 1} \psi_n[r_i, r - \widehat{r}, u] \varphi_i[r_i, \mathbf{k}, k_i - 1 - u].$$

Since $G_g[i,n]$, $g = 0, \ldots, n+1$, is a partition of $\mathcal{K}_n$ for all i, we obtain

$$P[\mathrm{ACIR}(t) \le r | n] = \sum_{\widehat{r} \le r} \sum_{i: r_i \le r - \widehat{r}} \sum_{s \in \mathcal{S}} \sum_{g=1}^{n+1} \sum_{u=0}^{g-1} \psi_n[r_i, r - \widehat{r}, u] \sum_{\mathbf{k} \in G_g[i,n]} \Theta_s[n, \mathbf{k}, \widehat{r}] \varphi_i[r_i, \mathbf{k}, g - 1 - u].$$

Interchanging the sums on u and g and using (27) yields

$$P[\mathrm{ACIR}(t) \le r | n] = \sum_{\widehat{r} \le r} \sum_{i: r_i \le r - \widehat{r}} \sum_{s \in \mathcal{S}} \sum_{u=0}^{n} \psi_n[r_i, r - \widehat{r}, u] \sum_{g=u+1}^{n+1} \Phi_s[g, i, n, g - 1 - u, \widehat{r}].$$

Since $\Phi_s[g, i, n, l, \widehat{r}] = 0$ for $l < 0$, the sum on g above may be taken from 0 to $n+1$. Thus, using (36) and interchanging the sums on u and s, we have

$$P[\mathrm{ACIR}(t) \le r | n] = \sum_{\widehat{r} \le r} \sum_{i: r_i \le r - \widehat{r}} \sum_{u=0}^{n} \psi_n[r_i, r - \widehat{r}, u] \sum_{s \in \mathcal{S}} \Psi_s[i, n, n-u, \widehat{r}].$$

Equation (41) is obtained by unconditioning on the number of transitions. □

4 Rate Based Rewards

In this section we describe an algorithm to calculate the distribution of cumulative reward averaged over time when reward rates are assigned to states in the model and no impulse rewards are present. We will show that the algorithm derived in the recent report [4] can be obtained by specializing the results of Section 3 to the rate based case.

In the rate reward case, $\widehat{K}+1=1$ and $\hat{r}_1=0$. To obtain $P[\mathrm{ACR}(t)\le r]$, we simply set $\hat{r}=0$ in Theorem 1, since the total accumulated impulse reward is always 0. The result is the expression

$$P[\mathrm{ACR}(t)\le r]=\sum_{n=0}^{\infty}e^{-\Lambda t}\frac{(\Lambda t)^n}{n!}\sum_{i:r_i\le r}\sum_{u=0}^{n}\psi_n[r_i,r,u]\|\Psi[i,n,n-u,0]\|. \tag{42}$$

A further simplification is possible by aggregating the terms representing the convolution of ψ_n and Ψ_s.

Definition 4 *For $n\ge 0$, $i=1,\ldots,K+1$, $s\in\mathcal{S}$, $u\ge 0$, let*

$$\Upsilon_s[i,n,u]=\sum_{l=0}^{u}\psi_n[r_i,r,l]\Psi_s[i,n,u-l,0]. \tag{43}$$

Note that since the quantity r is fixed in (42) (unlike $r-\hat{r}$ in (41) for the rate plus impulse case), there is no need to indicate it explicitly in the definition of Υ_s.

Define the vector $\boldsymbol{\Upsilon}[i,n,u]=\langle\Upsilon_{s_1}[i,n,u],\ldots,\Upsilon_{s_M}[i,n,u]\rangle$. Recall that $\omega_{ij}=r_i-r_j$ for $i\ne j$. Let $\omega_i=r_i-r$, $i=1,\ldots,K+1$, and let $\mathbf{P}[:s]$ be the sth column of the matrix $\mathbf{P}$. The following lemma gives a simple recursion for $\Upsilon_s[i,n,u]$.

Lemma 6 *Let $s\in\mathcal{S}$, $i=1,\ldots,K+1$. For $n\ge 1$, $u\ge 0$ and for $n=0$, $u>1$*

$$\Upsilon_s[i,n,u]=\begin{cases}\left(\dfrac{1}{\omega_{i,c(s)}}\right)\left(\{\boldsymbol{\Upsilon}[i,n-1,u-2]+\omega_i\boldsymbol{\Upsilon}[i,n-1,u-1]\}\,\mathbf{P}[:s]-\Upsilon_s[i,n,u-1]\right) & i\ne c(s)\\ \{\boldsymbol{\Upsilon}[i,n-1,u-1]+\omega_i\boldsymbol{\Upsilon}[i,n-1,u]\}\,\mathbf{P}[:s] & i=c(s)\end{cases} \tag{44}$$

The initial conditions are (for $n=0$, $u=0,1$)

$$\Upsilon_s[i,0,u]=\begin{cases}0 & i\ne c(s),\ u=0\\ \pi_s(0)/\omega_{i,c(s)} & i\ne c(s),\ u=1\\ \pi_s(0) & i=c(s),\ u=0\\ 0 & i=c(s),\ u=1\end{cases} \tag{45}$$

Proof: For $i \neq c(s)$, we have from (37) and (43)

$$\Upsilon_s[i,n,u] = \sum_{l=0}^{u} \psi_n[r_i,r,l] \left(\frac{1}{\omega_{i,c(s)}}\right) \left(\sum_{s'\in\mathcal{S}} p_{s's}\Psi_{s'}[i,n-1,u-1-l,0] - \Psi_s[i,n,u-1-l,0]\right).$$

Since $\Psi_s[i,n,l,0] = 0$ for $l < 0$, the terms corresponding to $l = u$ all vanish. Thus, using (24), we may write

$$\Upsilon_s[i,n,u] = \left(\frac{1}{\omega_{i,c(s)}}\right) \left\{ \sum_{s'\in\mathcal{S}} p_{s's} \sum_{l=0}^{u-1} \left(\psi_{n-1}[r_i,r,l-1] + \omega_i\psi_{n-1}[r_i,r,l]\right) \Psi_{s'}[i,n-1,u-1-l,0] - \sum_{l=0}^{u-1} \psi_n[r_i,r,l]\Psi_s[i,n,u-1-l,0] \right\}.$$

Since $\psi_n[r_i,r,l] = 0$ for $l < 0$, we have after rearranging terms

$$\begin{aligned}\Upsilon_s[i,n,u] &= \left(\frac{1}{\omega_{i,c(s)}}\right) \left\{ \sum_{s'\in\mathcal{S}} p_{s's} \left(\sum_{l=0}^{u-2} \psi_{n-1}[r_i,r,l]\Psi_{s'}[i,n-1,u-2-l,0] \right.\right. \\ &\quad \left. + \omega_i \sum_{l=0}^{u-1} \psi_{n-1}[r_i,r,l]\Psi_{s'}[i,n-1,u-1-l,0] \right) \\ &\quad \left. - \sum_{l=0}^{u-1} \psi_n[r_i,r,l]\Psi_s[i,n,u-1-l,0] \right\}.\end{aligned}$$

Recognizing the terms Υ_s, we have

$$\Upsilon_s[i,n,u] = \left(\frac{1}{\omega_{i,c(s)}}\right) \left\{ \sum_{s'\in\mathcal{S}} \left(\Upsilon_{s'}[i,n-1,u-2] + \omega_i\Upsilon_{s'}[i,n-1,u-1]\right) p_{s's} - \Upsilon_s[i,n,u-1] \right\}.$$

Writing the result in vector form gives (44) for $i \neq c(s)$.

The proof for the case $i = c(s)$ is similar.

The initial conditions follow directly from (25) and (38) for ψ_n and Ψ_s. □

It is interesting to observe that the recursion for calculating $\boldsymbol{\Upsilon}[i,n,u]$, can be done independently for each i and is independent of the number of rewards in the model. Theorem 2 below, which was proved in [4], shows that the distribution of ACR(t) can be obtained directly from the values of the vector $\boldsymbol{\Upsilon}[i,n,n]$.

Theorem 2 *The distribution of the total reward accumulated during $(0,t)$ averaged over t in the rate based case is given by*

$$P[\mathrm{ACR}(t) \le r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \sum_{i: r_i \le r} \|\boldsymbol{\Upsilon}[i,n,n]\|. \tag{46}$$

Proof: From (43) we have

$$\|\boldsymbol{\Upsilon}[i,n,n]\| = \sum_{u=0}^{n} \psi_n[r_i, r, u] \|\boldsymbol{\Psi}[i, n, n-u, 0]\|.$$

Thus (46) follows directly from (42). □

5 Impulse Rewards

When only impulse rewards are present, the measure of interest is the cumulative reward distribution $P[\mathrm{ACI}(t) \le r]$. In this case, $K+1 = 1$ and $r_1 = 0$, and Theorem 1 becomes

$$P[\mathrm{ACI}(t) \le r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \sum_{\widehat{r} \le r} \sum_{u=0}^{n} \psi_n[0, r - \widehat{r}, u] \|\boldsymbol{\Psi}[1, n, n-u, \widehat{r}]\|. \tag{47}$$

Since the model has only one rate reward, the case $i \neq c(s)$ never occurs, i.e., $c(s) = 1$ for any state s. Thus only the case $i = c(s)$ in the recursion for Ψ_s given in (37) is applicable. By examining (37) for $i = c(s)$, we observe that once the value of u is specified initially, it remains the same during the recursion. Now note from (38) that $\Psi_s[i, n, u, \widehat{r}] = 0$ when $u \neq 0$. Thus the only nonzero term in the sum over u on the right hand side of (47) is the term with $u = n$, so this equation simplifies to

$$P[\mathrm{ACI}(t) \le r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \sum_{\widehat{r} \le r} \psi_n[0, r - \widehat{r}, n] \|\boldsymbol{\Psi}[1, n, 0, \widehat{r}]\|. \tag{48}$$

This simplification may also be explained as follows. Since there is only one rate reward, the only rate coloring vector corresponding to n transitions is $\mathbf{k} = \langle n+1 \rangle$. Thus for $n \ge 0$, $\varphi_1[n, \mathbf{k}, 0] = 1$ and $\varphi_1[n, \mathbf{k}, l] = 0$ for $l \neq 0$ from (19) and (20) (using the case $i = j$), which causes the simplified behavior of the function Ψ_s in the impulse reward case.

Noting that $\psi_n[x, r, n] = 1$ for all n and r from (15), we obtain

$$P[\mathrm{ACI}(t) \le r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \sum_{\widehat{r} \le r} \|\boldsymbol{\Psi}[1, n, 0, \widehat{r}]\|. \tag{49}$$

Definition 5 *For $n \geq 0$, $s \in \mathcal{S}$, $r \geq 0$, let*

$$\widehat{\Upsilon}_s[n,r] = \sum_{\widehat{r} \leq r} \Psi_s[1,n,0,\widehat{r}]. \tag{50}$$

The following lemma shows that a simple recursion on n for $\widehat{\Upsilon}_s[n,r]$ can be found.

Lemma 7 *For $n \geq 1$*

$$\widehat{\Upsilon}_s[n,r] = \begin{cases} \sum_{s' \in \mathcal{S}} p_{s's} \widehat{\Upsilon}_{s'}[n-1, r - \widehat{r}_{\widehat{c}(s',s)}] & \text{if } r \geq 0 \\ 0 & \text{if } r < 0 \end{cases} \tag{51}$$

The initial conditions are

$$\widehat{\Upsilon}_s[0,r] = \begin{cases} \pi_s(0) & \text{if } r \geq 0 \\ 0 & \text{if } r < 0 \end{cases} \tag{52}$$

Proof: Using the case $i = c(s)$ of (37) in (50), we have for $r \geq 0$

$$\widehat{\Upsilon}_s[n,r] = \sum_{\widehat{r} \leq r} \sum_{s' \in \mathcal{S}} p_{s's} \Psi_{s'}[1, n-1, 0, \widehat{r} - \widehat{r}_{\widehat{c}(s',s)}].$$

Interchanging the order of summation and changing the variable of summation in the sum on $\widehat{r}$ gives

$$\widehat{\Upsilon}_s[n,r] = \sum_{s' \in \mathcal{S}} p_{s's} \sum_{\widehat{r} \leq r - \widehat{r}_{\widehat{c}(s',s)}} \Psi_{s'}[1, n-1, 0, \widehat{r}].$$

Recognizing the second sum as $\widehat{\Upsilon}_{s'}[n-1, r - \widehat{r}_{\widehat{c}(s',s)}]$ from (50), we obtain (51) as desired.

The initial conditions follow directly from (38) for $i = c(s)$. □

Define the vector $\widehat{\boldsymbol{\Upsilon}}[n,r] = \langle \widehat{\Upsilon}_{s_1}[n,r], \ldots, \widehat{\Upsilon}_{s_M}[n,r] \rangle$. The distribution of accumulated reward averaged over t in the case of impulse rewards can be calculated from the following.

Theorem 3 *The distribution of the total reward accumulated during $(0,t)$ averaged over t in the impulse based case is given by*

$$P[\mathrm{ACI}(t) \leq r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \|\widehat{\boldsymbol{\Upsilon}}[n,r]\|. \tag{53}$$

Proof: This follows immediately from (49) and (50). □

A probabilistic interpretation of $\widehat{\boldsymbol{\Upsilon}}[n,r]$ can be given in the following way. From (36) we see that

$$\Psi_s[1,n,0,\widehat{r}] = \Phi_s[n+1,1,n,0,\widehat{r}],$$

since only the term $g = n+1$ is nonzero when $u = 0$. Also, since $\mathbf{k} = \langle n+1 \rangle$ is the only rate coloring vector given n transitions, and since $\varphi_1[r_1, \langle n+1 \rangle, 0] = 1$, we have from (27)

$$\Phi_s[n+1,1,n,0,\widehat{r}] = \Theta_s[n,\langle n+1 \rangle,\widehat{r}].$$

Define

$$\Gamma_s[n,\widehat{r}] = \Theta_s[n,\langle n+1 \rangle,\widehat{r}]. \tag{54}$$

Then $\Psi_s[1,n,0,\widehat{r}] = \widehat{\Gamma}_s[n,\widehat{r}]$ is the probability, given n transitions, of an accumulated reward of $\widehat{r}$ *and* the state visited after the last transition is s. Thus

$$P[\mathrm{ACI}(t) \le r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \sum_{\widehat{r} \le r} \widehat{\Gamma}[n,\widehat{r}], \tag{55}$$

where $\widehat{\Gamma}[n,\widehat{r}] = \sum_{s \in \mathcal{S}} \widehat{\Gamma}_s[n,\widehat{r}]$ is the probability, given n transitions, of an accumulated reward of $\widehat{r}$. Furthermore

$$\widehat{\Upsilon}_s[n,r] = \sum_{\widehat{r} \le r} \widehat{\Gamma}_s[n,\widehat{r}] \tag{56}$$

is the probability, given n transitions, that the total accumulated impulse reward averaged over t is at most r *and* the state visited after the last transition is s.

In [15] the distribution of ACI(t) was obtained directly as follows. Conditioning on n transitions and an accumulated reward of $\widehat{r}$, (1) becomes

$$P[\mathrm{ACI}(t) \le r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \sum_{\widehat{r} \in \mathcal{L}_n} \widehat{\Gamma}[n,\widehat{r}] P[\mathrm{ACI}(t) \le r | n,\widehat{r}]. \tag{57}$$

We clearly have

$$P[\mathrm{ACI}(t) \le r | n,\widehat{r}] = \begin{cases} 1 & \text{if } \widehat{r} \le r \\ 0 & \text{otherwise} \end{cases} \tag{58}$$

Substituting (58) into (57) gives (55). As noted in (56), this leads to (53) using the aggregation introduced in Definition 5.

The following result may also be derived directly using (1).

Theorem 4 *The probability mass function of the total reward accumulated during* $(0,t)$ *averaged over* t *in the impulse based case is given by*

$$P[\mathrm{ACI}(t) = r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \widehat{\Gamma}[n,r]. \tag{59}$$

The recursion for $\widehat{\Gamma}_s[n,r]$ is the same as that for $\widehat{\Upsilon}_s[n,r]$, except for the initial conditions.

Lemma 8 *For $n \geq 1$*

$$\widehat{\Gamma}_s[n,r] = \begin{cases} \sum_{s' \in \mathcal{S}} p_{s's} \widehat{\Gamma}_{s'}[n-1, r - \widehat{r}_{\widehat{c}(s',s)}] & \text{if } r \geq 0 \\ 0 & \text{if } r < 0 \end{cases} \tag{60}$$

The initial conditions are

$$\widehat{\Gamma}_s[0,r] = \begin{cases} \pi_s(0) & \text{if } r = 0 \\ 0 & \text{if } r \neq 0 \end{cases} \tag{61}$$

6 The Complementary Distribution

In certain cases, the upper tail of the distribution of the cumulative reward averaged over t is the measure of interest. In this section we will derive an expression for $P[\mathrm{ACIR}(t) > r]$ for the general case of both rate and impulse rewards, and then specialize to the rate reward case and to the impulse reward case. First note that the equation analogous to (4) is

$$P[\mathrm{ACIR}(t) > r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \sum_{\widehat{r} \in \mathcal{L}_n} \sum_{\mathbf{k} \in \mathcal{K}_n} \Theta[n, \mathbf{k}, \widehat{r}] P[\mathrm{ACIR}(t) > r | n, \mathbf{k}, \widehat{r}]. \tag{62}$$

Since the accumulated reward is greater than r with probability 1 if the impulse reward gained is itself greater than r, i.e., $P[\mathrm{ACR}(t) > r - \widehat{r} | n, \mathbf{k}] = 1$ for $\widehat{r} > r$, the equation corresponding to (6) is

$$P[\mathrm{ACIR}(t) > r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \sum_{\widehat{r} \leq r} \sum_{\mathbf{k} \in \mathcal{K}_n} \Theta[n, \mathbf{k}, \widehat{r}] P[\mathrm{ACR}(t) > r - \widehat{r} | n, \mathbf{k}] + P[\mathrm{ACI}(t) > r]. \tag{63}$$

To evaluate the first term of the right hand side of (63), we use the following result which was proved in [4] and complements Lemma 1.

Lemma 9 *For $n \geq 0$ and $\mathbf{k} \in \mathcal{K}_n$, we have*

$$P[\mathrm{ACR}(t) > r | n, \mathbf{k}] = \sum_{i : r_i > r} f_i[r_i, n, \mathbf{k}, r, k_i - 1]. \tag{64}$$

Substituting this into (63) yields

$$\begin{aligned} P[\mathrm{ACIR}(t) > r] &= \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \sum_{\widehat{r} \leq r} \sum_{\mathbf{k} \in \mathcal{K}_n} \Theta[n, \mathbf{k}, \widehat{r}] \sum_{i : r_i > r - \widehat{r}} f_i[r_i, n, \mathbf{k}, r - \widehat{r}, k_i - 1] \\ &\quad + P[\mathrm{ACI}(t) > r]. \end{aligned} \tag{65}$$

Proceeding as in Section 3 and using (59), we obtain

$$P[\mathrm{ACIR}(t) > r] = \sum_{n=0}^{\infty} e^{-\Lambda t}\frac{(\Lambda t)^n}{n!}\sum_{\widehat{r}\le r}\sum_{i:r_i>r-\widehat{r}}\sum_{u=0}^{n}\psi_n[r_i, r-\widehat{r}, u]\|\Psi[i,n,n-u,\widehat{r}]\| + \sum_{n=0}^{\infty} e^{-\Lambda t}\frac{(\Lambda t)^n}{n!}\sum_{\widehat{r}>r}\widehat{\Gamma}[n,\widehat{r}]. \tag{66}$$

Definition 6 *For $n \ge 0$, $s \in \mathcal{S}$, $r \ge 0$, let*

$$\widehat{\Omega}_s[n,r] = \sum_{\widehat{r}>r}\widehat{\Gamma}_s[n,\widehat{r}]. \tag{67}$$

Then $\widehat{\Omega}_s[n,r]$ is the probability, given n transitions, that the total accumulated impulse reward averaged over t is greater than r *and* the state visited after the last transition is s. The following recursion is similar to those for $\widehat{\Upsilon}_s[n,r]$ and $\widehat{\Gamma}_s[n,r]$ in (51) and (60).

Lemma 10 *For $n \ge 1$*

$$\widehat{\Omega}_s[n,r] = \begin{cases} \sum_{s'\in\mathcal{S}} p_{s's}\widehat{\Omega}_{s'}[n-1, r-\widehat{r}_{\widehat{c}(s',s)}] & \text{if } r \ge 0 \\ \pi_s(n) & \text{if } r < 0 \end{cases} \tag{68}$$

The initial conditions are

$$\widehat{\Omega}_s[0,r] = \begin{cases} 0 & \text{if } r \ge 0 \\ \pi_s(0) & \text{if } r < 0 \end{cases} \tag{69}$$

Define the vector $\widehat{\boldsymbol{\Omega}}[n,r] = \langle \widehat{\Omega}_{s_1}[n,r], \ldots, \widehat{\Omega}_{s_M}[n,r]\rangle$.

Theorem 5 *The complementary distribution of the total reward accumulated during $(0,t)$ averaged over t is given by*

$$P[\mathrm{ACIR}(t) > r] = \sum_{n=0}^{\infty} e^{-\Lambda t}\frac{(\Lambda t)^n}{n!}\sum_{\widehat{r}\le r}\sum_{i:r_i>r-\widehat{r}}\sum_{u=0}^{n}\psi_n[r_i, r-\widehat{r}, u]\|\Psi[i,n,n-u,\widehat{r}]\| + \sum_{n=0}^{\infty} e^{-\Lambda t}\frac{(\Lambda t)^n}{n!}\|\widehat{\boldsymbol{\Omega}}[n,r]\|. \tag{70}$$

When Theorem 5 is specialized to the case of only rate rewards, the second term on the right hand side of (70) vanishes, since all impulse rewards are 0. Thus, similar to the derivation of (42), we obtain

$$P[\mathrm{ACR}(t) > r] = \sum_{n=0}^{\infty} e^{-\Lambda t}\frac{(\Lambda t)^n}{n!}\sum_{i:r_i>r}\sum_{u=0}^{n}\psi_n[r_i, r, u]\|\Psi[i,n,n-u,0]\|. \tag{71}$$

Using the aggregation defined in (43) yields the following result.

Theorem 6 *The complementary distribution of the total reward accumulated during* $(0, t)$ *averaged over* t *in the rate based case is given by*

$$P[\mathrm{ACR}(t) > r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \sum_{i: r_i > r} \|\boldsymbol{\Upsilon}[i, n, n]\|. \tag{72}$$

When only impulse rewards are present, all rate rewards are set to 0. In this case, the first term on the right hand side of (70) vanishes, since there are no rate rewards that satisfy $r_i > r - \hat{r}$ for $\hat{r} \leq r$.

Theorem 7 *The complementary distribution of the total reward accumulated during* $(0, t)$ *averaged over* t *in the impulse based case is given by*

$$P[\mathrm{ACI}(t) > r] = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \|\widehat{\boldsymbol{\Omega}}[n, r]\|. \tag{73}$$

7 Computational Requirements

In Sections 3, 4 and 5 we presented the details of algorithms to calculate the distribution of cumulative reward averaged over the observation period. In this section we briefly present the computational requirements of the algorithms. Since the complexity of the algorithm for the combined rate and impulse case can be easily obtained from the individual complexities of the rate case and the impulse case, we first concentrate on the latter two algorithms.

The main computational effort of using the algorithm of Section 4 is spent in the calculation of $\boldsymbol{\Upsilon}[i, n, u]$ for i such that $r_i > r$ (or $r_i < r$) for $n = 0, \ldots, N$ and $u = 0, \ldots, N$, where recall that N is the truncation point of the infinite series in Theorem 2. From equation (44) we see that a vector by vector multiplication is needed to calculate an entry of the vector $\boldsymbol{\Upsilon}[i, n, u]$. Therefore, for a given i, we need a total of $O(EN^2)$ multiplications, where E is the number of nonzero entries in the stochastic matrix $\mathbf{P}$. Note that $E = dM$, where recall that M is the dimension of the state space and $1 \leq d \leq M$ represents the average number of nonzero elements in a column (or row) of $\mathbf{P}$. Since $\mathbf{P}$ is usually sparse, in most cases we have $d << M$. Let $m = m(r)$ be the minimum of the number of rewards that are strictly greater than r and the number of rewards that are strictly less than r. From (46) or (71), only the values of $\boldsymbol{\Upsilon}$ for m indices i are needed, so that a total of $O(mEN^2)$ multiplications are required. As discussed in [4] in many cases one of the tails of the distribution is of interest, and so the value of m is close to or equal to 1. The algorithm has storage requirements of $O(EN^2)$, since the values of $\boldsymbol{\Upsilon}$ for different values of i can be calculated independently.

We now consider the computational requirements for the impulse reward algorithm of Section 5. The main computational cost of (53) is in the evaluation of $\widehat{\boldsymbol{\Upsilon}}$ using the recursion of Lemma 7. The computational complexity of that recursion can be estimated as follows. Let $\kappa(n, r)$ be the number of distinct values less than or equal to r that can be obtained by adding any combination of n of the $\widehat{K}+1$ impulse rewards. The number of operations necessary to evaluate the recursion is $O(M \sum_{n=0}^{N} \kappa(n, r))$, and the storage requirements are $O(M\kappa(N, r))$. It is not difficult to see that $\kappa(n, r)$ is related to the precision of the impulse reward values. If all the rewards are rational, and we multiply them by the minimum integer q that scales them to integer values, then $\kappa(n, r) < qr$ for any value of n. For example, let $\widehat{K} = 2$ and $\widehat{r}_1 = 1.5$, $\widehat{r}_2 = 1.0$ and $\widehat{r}_3 = 0$. Then $q = 2$ and $\kappa(n, r) < 2r$.

The computational cost for the algorithm of Section 3 is virtually the same as the cost of evaluating $\Psi_s[i, n, u, \widehat{r}]$. From the discussion above, it is easy to see that a total of $O(mEN^2 \sum_{n=0}^{N} \kappa(n, r))$ multiplications and storage requirements of $O(EN^2\kappa(N, r))$ are needed.

8 Implementation Issues

In this section we discuss some useful implementation details. Although the discussion is limited to the algorithm of Section 4 for conciseness, the issues are similar for the algorithm of Section 3.

The recursion for $\boldsymbol{\Upsilon}$ given in Lemma 6 involves the multipliers $1/\omega_{i,j}$ and ω_i. Depending on the values of the rewards r_i and the value of r, these multipliers may cause overflow problems for large values of N. In order to alleviate this problem, our objective is to find a scaling such that the multipliers in (44) always have absolute value less than or equal to 1. We consider only the case for which the index i in Theorem 2 satisfies $r_i > r$. The scaling for the case related to equation (71), when i is such that $r_i < r$, is similar.

We first scale the rewards r_i and the reward level r by dividing them by $\omega_1 = r_1 - r$. The distribution of cumulative reward averaged over t does not change under this scaling, and thus the recursion in (44) with ω_i and $\omega_{i,c(s)}$ replaced by ω_i/ω_1 and $\omega_{i,c(s)}/\omega_1$ can be used to obtain $P[\mathrm{ACR}(t) > r]$. Although $\omega_i/\omega_1 \le 1$, the multipliers $\frac{1}{\omega_{i,c(s)}/\omega_1}$ must be further scaled to ensure that they are at most 1 in absolute value. Define

$$\omega = \min_{i: r_i > r} \{1, \omega_{i,i+1}/\omega_1\}. \tag{74}$$

Set $\omega_i^* = \omega_i/\omega_1$, and let $\omega_{i,j}^* = \frac{\omega}{\omega_{i,j}/\omega_1}$ for $j \neq i$. Note that $0 \le \omega_i^* \le 1$ and $0 \le |\omega_{i,j}^*| \le 1$, for i such that $r_i > r$ and $j \neq i$, after this rescaling.

Now make the following definition.

Definition 7 *For i such that $r_i > r$, $n \geq 0$, $s \in \mathcal{S}$, $u \geq 0$, let*

$$\Upsilon^*_s[i,n,u] = \omega^u \Upsilon_s[i,n,u]. \tag{75}$$

We now show that the recursion in (44) can be rewritten so that all multipliers have absolute value at most 1.

Corollary 1

$$\Upsilon^*_s[i,n,u] = \begin{cases} \omega^*_{i,c(s)} \left(\{\omega \boldsymbol{\Upsilon}^*[i,n-1,u-2] + \omega^*_i \boldsymbol{\Upsilon}^*[i,n-1,u-1]\} \mathbf{P}[:s] - \Upsilon^*_s[i,n,u-1]\right) & i \neq c(s) \\ \{\omega \boldsymbol{\Upsilon}^*[i,n-1,u-1] + \omega^*_i \boldsymbol{\Upsilon}^*[i,n-1,u]\} \mathbf{P}[:s] & i = c(s) \end{cases} \tag{76}$$

The initial conditions are similar to those in equation (45), except that $1/\omega_{i,c(s)}$ is replaced by $\omega^*_{i,c(s)}$.

Proof: First consider the case $i \neq c(s)$. Multiplying both sides of (44) by ω^u and applying Definition 7 to the left hand side of the resulting equation, we can write

$$\begin{aligned} \Upsilon^*_s[i,n,u] &= \omega \left(\frac{\omega_1}{\omega_{i,c(s)}}\right) \left(\left\{\omega \omega^{u-2} \boldsymbol{\Upsilon}[i,n-1,u-2] + (\omega_i/\omega_1)\omega^{u-1} \boldsymbol{\Upsilon}[i,n-1,u-1]\right\} \mathbf{P}[:s] \right. \\ &\quad \left. - \omega^{u-1} \Upsilon_s[i,n,u-1]\right). \end{aligned} \tag{77}$$

Applying Definition 7 to the right hand side of (77) we obtain (76) for $i \neq c(s)$. The case $i = c(s)$ is proved in the same manner. □

It is clear that all the multiplicative factors in the recursion (76) are at most 1 in absolute value. Next the $\boldsymbol{\Upsilon}^*$ are scaled back to obtain the distribution of cumulative reward, but this must be done in a manner that prevents overflow. From the values of $\boldsymbol{\Upsilon}^*$, we obtain $P[\mathrm{ACR}(t) > r]$ as follows. Let α be the smallest positive integer such that $\omega^{-N/\alpha}$ is less than the overflow value. For $n = 0, \ldots, N$, let

$$\Theta(n) = \sum_{i: r_i > r} \|\boldsymbol{\Upsilon}^*[i,n,n]\|. \tag{78}$$

Then $P[\mathrm{ACR}(t) > r|n] = \omega^{-n}\Theta(n)$ by (46) and (75). Now define the function $\theta_n(k)$, for $k = 0, \ldots, \alpha$, recursively as $\theta_n(k) = \omega^{-n/\alpha}\theta_n(k-1)$ with initial condition $\theta_n(0) = \Theta(n)$. Clearly, $P[\mathrm{ACR}(t) > r|n] = \theta_n(\alpha)$, so that $\theta_n(\alpha) \leq 1$. Since $0 \leq \omega \leq 1$, $\theta_n(k)$ is an increasing function of k which is upper bounded by 1. Note that $1 \leq \omega^{-n/\alpha} \leq \omega^{-N/\alpha}$, and so all the quantities involved in the operations always remain below the overflow value.

Clearly, the value of ω can be made arbitrarily small (and so ω^{-n} can be made arbitrarily large) if the values of two distinct rewards that are greater than r approach each other. In this case it was observed that $\Theta(n)$ may lose precision as n approaches N. Such a loss of precision can be easily monitored by comparing the positive and negative values of the terms in the recursive expressions for $\boldsymbol{\Upsilon}^*$ in equation (78). It was also observed empirically that the nth root of $\Theta(n)$ converges in these cases, and this can be used to obtain the remaining terms in (46). However, rewards that are near to each other can also be merged into a single reward to obtain bounds for the final solution. Intuitively, if the reward values hardly differ, the bounds should be tight. In other words, suppose that the normalized values r_i^* and r_j^*, where $r_i^* > r$, $r_j^* > r^*$, differ by a few percent. If both rewards are set to r_i, we obtain an upper bound on the final solution. On the other hand, setting both rewards to r_j gives a lower bound. In fact, a procedure to automatically merge rewards can be easily implemented as part of the general algorithm, starting from the largest reward r_1 and continuing on to $r_2, r_3, \ldots$, stopping when the smallest reward greater than r is reached.

9 Examples

In this section we present examples to illustrate the application of the algorithm developed in Section 3. We consider a model of a system with two processors and three disks. These units fail independently of each other, and we assume that no failures can occur once the system is down. The processors fail at an exponential rate of 1 per 120 hours, and the disks fail at an exponential rate of 1 per 240 hours.

In the first example we assume that no repairs to the system are performed until the system is down. Once the system is inoperative it undergoes a repair procedure, and the system is brought back to full operation with no failed components. The repair lasts for an exponential amount of time with rate 0.125. The system is considered operational when at least one processor and one disk unit are operational. The model that captures the system failures and repairs over time has 11 states, and 6 of them represent an operational system.

We assume that each time the system stops working and a repair has to be performed, a fee of \$300.00 is paid. Furthermore, when the system is down, the company that owns the system loses \$40.00 per each hour the system remains inoperative. An extra repair cost of \$10.00 per hour is added when the system is down with no processors available. We assume the processors are helpful in the diagnosis and repair procedure, and so this extra fee covers the use of extra equipment needed for the repair.

The failure behavior for the second example is the same as for the first example. However, in this case, repairs are done to the failed components one at a time, with priority given to the repair of the processors. A fixed fee is paid each time a repair is performed on a unit, independent of whether or not the system is operational. When the system is down, the company incurs the same costs per unit time as in the first example.

The third example is similar to the second one, except that, instead of paying a fixed fee each time a repair is performed, a larger fee is paid each time the system goes down. Note that the difference between the first and third examples is that in the former repairs are performed only when system is down, and in the latter repairs are performed as soon as a component fails.

These examples illustrate reward models with combined rate and impulse rewards. Our goal is to calculate the distribution of the total company cost during a given interval of time, i.e., $P[t\mathrm{ACIR}(t) \leq C]$. Figure 1 plots the distribution of the total cost during a month of operation for the first and third examples. For the third example, we consider two values

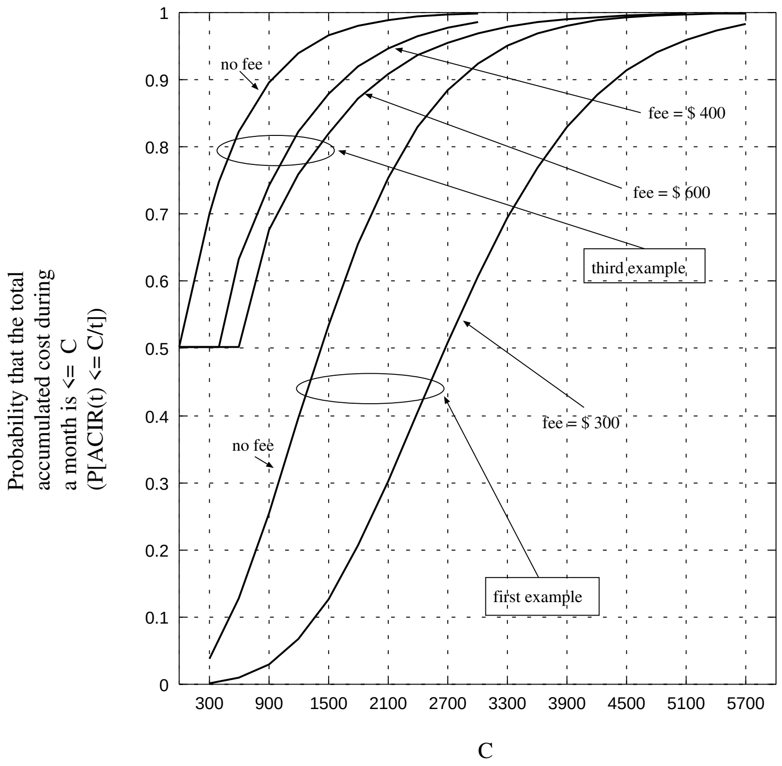

Figure 1: Distributions of cost (720 hours), first and third examples.

for the fee: \$400 and \$600, as indicated in the figure. Two sets of curves are plotted, one set for each example. In order to show the influence of the impulse reward on the distribution of the total cost, the corresponding distribution when no impulse fee is paid is also plotted.

In both examples we see that there is a considerable difference in the distribution of the cost when a fee is paid each time the system goes down. For instance, in the first example, the probability that the cost is less than or equal to $\$3,000$ is more than 0.92 if no fee is paid when the system goes down, but only 0.6 if the fee is paid. In the third example, the probabilities are approximately 0.9 when no fee is paid and approximately 0.7 when any of the two fees considered is paid, for a cost under \$900. Note that in Figure 1, the probability of incurring a cost less than the fee paid each time the system goes down is the probability that the system does not fail during the interval of time in study.

It also is interesting to observe that the limiting expected cost per month for the first example is a little less than $\$3,000$, but the probability of achieving a cost less than or equal to this value is only 0.61.

In Figure 2 we compare the first and second examples. It is not clear if it is more cost

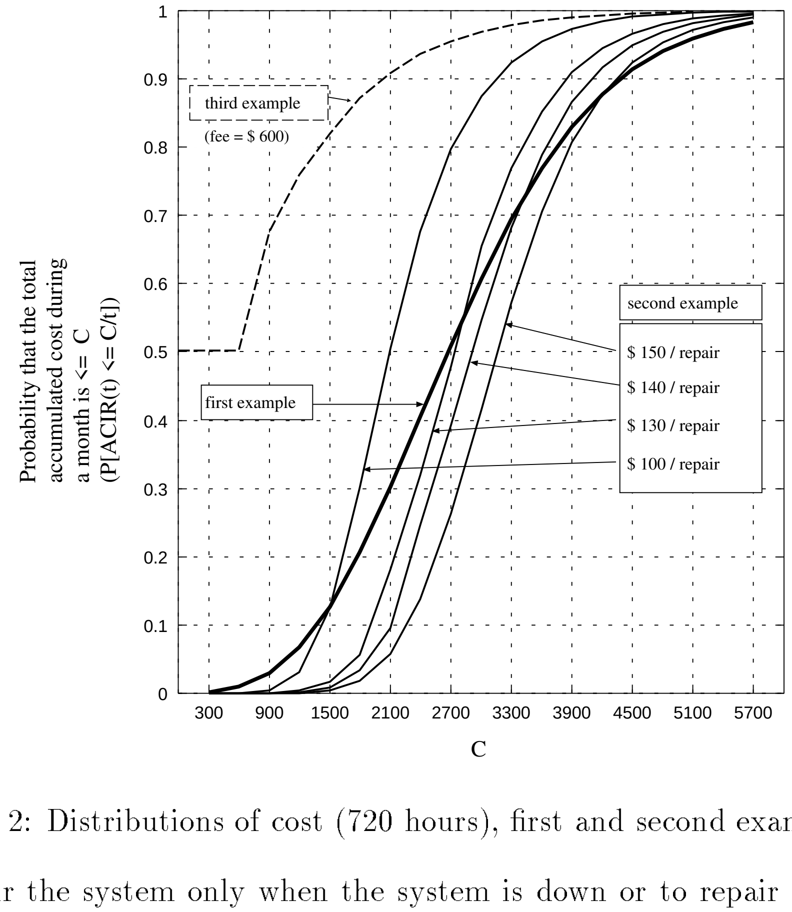

Figure 2: Distributions of cost (720 hours), first and second examples.

effective to repair the system only when the system is down or to repair each time a unit fails, paying a price for each repair performed. Intuitively, the answer will depend on the price paid for each repair performed, but it is not evident for which values one scheme will be better than the other. We plot the distribution of total cost when the fee paid for each repair performed (second example) is \$100, \$130, \$140 and \$150. For reference, we also plot the third example for a fee of \$600. From the figure we note that there is less variability in the distribution of the cost for the second example for the range of fees plotted than for the first example (recall that in both examples, there is also a cost per time unit when the system is down). By examining the figure it is clear that paying \$100 per repair performed is better than performing repairs only when the system goes down and paying a fee of \$300 when that occurs. However, the choice is not as clear when the fee is in the range of \$130 to \$150 per repair. For instance, consider the \$140 curve in Figure 2. The probability that the total cost over one month is under, say, $\$4,200$ is higher than the corresponding probability for the first example. As a consequence, if the expense budget has a hard limit, we may want to make repairs as components fail, since this policy implies a lower probability of exceeding the limited budget. On the other hand, the policy of the first example has a higher probability of a cost under, say, $\$2,700$ (e.g., below the limiting expected cost). In summary, the choice will depend on the amount of risk the company wants to take. A higher possibility for savings implies a higher probability of exceeding the budget limit.

10 Summary

We developed an algorithm to obtain the distribution of the reward accumulated during a given interval of time when both rate and impulse rewards are present. We then showed how to specialize this algorithm to the case of models with only rate based rewards and models with only impulse based rewards. Previous algorithms for these cases were shown to result. Only probabilistic arguments were applied throughout the development to obtain the main equations. The algorithm developed is simple to implement, robust, and has a low computational cost. Its computational complexity compares favorably with respect to previous methods.

References

[1] E. Çinlar. *Introduction to Stochastic Processes.* Prentice-Hall, 1975.

[2] E. de Souza e Silva and H.R. Gail. Calculating availability and performability measures of repairable computer systems using randomization. *Journal of the ACM*, 36(1):171–193, 1989.

[3] E. de Souza e Silva and H.R. Gail. Performability analysis of computer systems: from model specification to solution. *Performance Evaluation*, 14:157–196, 1992.

[4] E. de Souza e Silva and H.R. Gail. Calculating transient distributions of cumulative reward. Technical report, UCLA CSD-930033, UCLA Computer Science Department, Los Angeles, CA 90024, September 1993.

[5] A. Goyal, W.C. Carter, E. de Souza e Silva, S.S. Lavenberg, and K.S. Trivedi. The system availability estimator. In *Proceedings of FTCS-16*, pages 84–89, 1986.

[6] W.K. Grassmann. Numerical solutions for Markovian event systems. In P. Kall *et al.*, editor, *Quantitative Methoden in den Wirtschaftswissenschaften*, pages 73–87. Springer, 1989.

[7] W.K. Grassmann. Finding transient solutions in Markovian event systems through randomization. In W.J. Stewart, editor, *Numerical Solution of Markov Chains*, pages 357–371. Marcel Dekker, Inc., 1991.

[8] D. Gross and D.R. Miller. The randomization technique as a modeling tool and solution procedure for transient Markov processes. *Operations Research*, 32(2):343–361, 1984.

[9] A. Jensen. Markoff chains as an aid in the study of Markoff processes. *Skandinavsk Aktuarietidskrift*, 36:87–91, 1953.

[10] J.F. Meyer. On evaluating the performability of degradable computing systems. *IEEE Trans. on Computers*, C-29(8):720–731, 1980.

[11] J.F. Meyer. Performability: A retrospective and some pointers to the future. *Performance Evaluation*, 14:139–156, 1992.

[12] M.A. Qureshi and W.H. Sanders. Reward model solution methods with impulse and rate rewards: an algorithm and numerical results. Technical report, The University of Arizona, Tucson, AZ, 1993.

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

[14] K.S. Trivedi, J.K. Muppala, S.P. Woolet, and B.R. Haverkort. Composite performance and dependability analysis. *Performance Evaluation*, 14:197–215, 1992.

[15] R.A. Vallejos Campos. *Performability Models and Applications to Computing Systems.* PhD thesis, (in Portuguese), Federal University of Rio de Janeiro, Rio de Janeiro, Brazil, 1993.

[16] H. Weisberg. The distribution of linear combinations of order statistics from the uniform distribution. *Annals Math. Stat.*, 42(2):704–709, 1971.