Dept. of Math. University of Oslo
Statistical Research Report No. 11
ISSN 0806–3842 November 2008

ADVANCED DISCRETE EVENT SIMULATION METHODS WITH APPLICATION TO IMPORTANCE MEASURE ESTIMATION

Arne B. Huseby [*], Kristina A. Eide [†], Stefan L. Isaksen [‡],
Bent Natvig [§], Jørund Gåsemyr [¶]

Abstract

In the present paper we use discrete event simulation in order to analyze a *binary monotone system* of repairable components. Asymptotic statistical properties of such a system, e.g., the asymptotic system availability and component criticality, can easily be estimated by running a *single* discrete event simulation on the system over a sufficiently long time horizon, or by working directly on the stationary component availabilities. Sometimes, however, one needs to estimate how the statistical properties of the system evolve over time. In such cases it is necessary to run many simulations to obtain a stable curve estimate. At the same time one needs to store much more information from each simulation. A *crude* approach to this problem is to sample the system state at fixed points of time, and then use the mean values of the states at these points as estimates of the curve. Using a sufficiently high sampling rate a satisfactory estimate of the curve can be obtained. Still, all information about the process between the sampling points is thrown away. To handle this issue, we propose an alternative sampling procedure where we utilize process data between the sampling points as well. This simulation method is particularly useful when estimating various kinds of component importance measures for repairable systems. As explained in [11] such measures can often be expressed as weighted integrals of the time-dependent Birnbaum measure of importance. By using the proposed simulation methods, stable estimates of the Birnbaum measure as a function of time are obtained. Combined with the appropriate weight function the importance measures of interest can be estimated.

KEYWORDS: Discrete event simulation; Repairable systems; Importance measures

1 INTRODUCTION

Discrete event models are typically used in simulation studies to model and analyze pure jump processes. For an extensive introduction to discrete event models we refer to [3].

[*] Department of Mathematics, University of Oslo, P.O. Box 1053 Blindern, N-0316 Oslo, Norway. *arne@math.uio.no*

[†] FFI, P.O. Box 25, N-2027 Kjeller, Norway. *Kristina-Aalvik.Eide@ffi.no*

[‡] DNV Energy, Veritasveien 1, N-1322 Høvik, Norway. *Stefan.Isaksen@dnv.com*

[§] Department of Mathematics, University of Oslo, P.O. Box 1053 Blindern, N-0316 Oslo, Norway. *bent@math.uio.no*

[¶] Department of Mathematics, University of Oslo, P.O. Box 1053 Blindern, N-0316 Oslo, Norway. *gaasemyr@math.uio.no*

A discrete event model can be viewed as a system consisting of a collection of stochastic processes, where the states of the individual processes change as results of various kinds of *events* occurring at random points of time. Between these events the states of the processes are considered to be constant. We refer to the processes included in the collection, as the *elementary processes* of the system. In our context we always assume that each event only affects *one* of the elementary processes.

More formally we consider a pure jump process S, and let $S(t)$ denote the state of the process at time $t \geq 0$. Moreover, we let $T_1 < T_2 < \cdots$ denote the points of time of the events affecting the process, and let $T_0 = 0$. In our context a pure jump process is a process where the state function, $S(t)$, can be written in the following form:

$$S(t) = S(0) + \sum_{j=1}^{\infty} I(T_j \leq t) J_j, \qquad t \geq 0, \tag{1}$$

where $I(\cdot)$ denotes the indicator function, and J_j denotes the change in the state of the process at time T_j. The representation (1) implies that the state function $S(t)$ is piecewise constant and right-continuous in t, with jumps at $T_1 < T_2 < \cdots$. In particular, for $j = 0, 1, \ldots$, we have $S(t) = S(T_j)$ for all $t \in [T_j, T_{j+1})$, implying that $\lim_{t \to T_j^+} S(t) = S(T_j)$.

The fact that a pure jump process is right-continuous and piecewise constant in t is convenient during simulations. Hence, in order to keep track of how the process evolves and update the value of the state function, only the points of time where the events happen need to be considered.

The infinite sum in (1) indicates that the number of events occurring in the interval $[0, t]$ is unbounded. The possibility of having an infinite number of events in $[0, t]$, however, may cause various technical difficulties. In particular, this may cause simulations to break down since an infinite number of events need to be generated and handled. See [4] for a further discussion of this issue. To avoid these difficulties, we always assume that the number of events occurring in any finite interval is finite with probability one. A pure jump process satisfying this assumption is said to be *regular*. Some basic results on regularity are included in the appendix. See also [6].

Stationary statistical properties of a system, can easily be estimated by running a *single* discrete event simulation on the system over a sufficiently long time horizon, or by working directly on the stationary probability distributions of the elementary processes. Sometimes, however, one needs to estimate how the statistical properties of the system evolve over time. In such cases it is necessary to run many simulations to obtain stable results. Moreover, one must store much more information from each simulation. A *crude* approach to this problem is to sample the system state at fixed intervals of time, and then use the mean values of the states at these points as estimates of the corresponding statistical properties. Using a sufficiently high sampling rate, i.e., short intervals between sampling points, a satisfactory estimate of the full curve can be obtained. Still, all information about the process between the sampling points is thrown away. Thus, we propose an alternative sampling procedure where we utilize process data between the sampling points as well.

In order to illustrate the main ideas we use discrete events in order to analyze a *multi-component binary monotone system* of repairable components. In a companion paper [10] the simulation technology developed in the present paper, is used to estimate the Natvig measures of component importance in repairable systems and applied to an offshore oil and gas production system. For nonrepairable systems the Natvig measure is treated in [7], [8] and [9].

2 BASIC RELIABILITY THEORY

We start out by briefly reviewing basic concepts of reliability theory. See [1]. A *binary monotone system* is an ordered pair (C, ϕ) where $C = \{1, \dots, n\}$ is a nonempty finite set, and ϕ is a binary function. The elements of C are interpreted as components of some technological system. Each component, as well as the system itself can be either functioning or failed. We denote the state of component i at time $t \geq 0$ by $X_i(t)$, where $X_i(t) = 1$ if i is functioning at time t, and zero otherwise, $i = 1, \dots, n$. We also introduce the component state vector $\boldsymbol{X}(t) = (X_1(t), \dots, X_n(t))$. The function ϕ is called the *structure function* of the system, and expresses the state of the system as a function of the component state vector, and is assumed to be non-decreasing in each argument. Thus, $\phi = \phi(\boldsymbol{X}(t)) = 1$ if the system is functioning at time t and zero otherwise.

In the present paper we consider systems with repairable components. Thus, for $i = 1, \dots, n$ and $j = 1, 2, \dots$ let:

U_{ij} = The jth lifetime of the ith component.

D_{ij} = The jth repair time of the ith component.

We assume that U_{ij} has an absolutely continuous distribution with a positive mean value $\mu_i < \infty$, while D_{ij} has an absolutely continuous distribution with a positive mean value $\nu_i < \infty$, $i = 1, \dots, n$, $j = 1, 2, \dots$. All lifetimes and repair times are assumed to be independent. Thus, in particular the component processes $\{X_1(t)\}, \dots, \{X_n(t)\}$ are independent of each other.

Let $A_i(t)$ be the availability of the ith component at time t, i.e., the probability that the component is functioning at time t. That is, for $i = 1, \dots, n$ we have:

$$A_i(t) = \Pr(X_i(t) = 1) = E[X_i(t)].$$

The corresponding stationary availabilities are given by:

$$A_i = \lim_{t \to \infty} A_i(t) = \frac{\mu_i}{\mu_i + \nu_i}, \qquad i = 1, \dots, n. \tag{2}$$

Introduce $\boldsymbol{A}(t) = (A_1(t), \dots, A_n(t))$ and $\boldsymbol{A} = (A_1, \dots, A_n)$. The system availability at time t is given by:

$$A_\phi(t) = \Pr(\phi(\boldsymbol{X}(t)) = 1) = E[\phi(\boldsymbol{X}(t))] = h(\boldsymbol{A}(t)),$$

where h is the system's reliability function. The corresponding stationary availability is given by:

$$A_\phi = \lim_{t \to \infty} A_\phi(t) = h(\boldsymbol{A}). \tag{3}$$

The component i is said to be *critical* at time t if $\psi_i(\boldsymbol{X}(t)) = \phi(1_i, \boldsymbol{X}(t)) - \phi(0_i, \boldsymbol{X}(t)) = 1$. We will refer to $\psi_i(\boldsymbol{X}(t))$ as the *criticality state* of component i at time t. The Birnbaum measure of importance of component i at time t, is defined as the probability that component i is critical at time t, and denoted $I_B^{(i)}(t)$. See [2]. Thus,

$$\begin{aligned} I_B^{(i)}(t) &= \Pr(\psi_i(\boldsymbol{X}(t)) = 1) = E[\psi_i(\boldsymbol{X}(t))] \\ &= h(1_i, \boldsymbol{A}(t)) - h(0_i, \boldsymbol{A}(t)). \end{aligned} \tag{4}$$

The corresponding stationary measure is given by:

$$I_B^{(i)} = \lim_{t \to \infty} I_B^{(i)}(t) = h(1_i, \boldsymbol{A}) - h(0_i, \boldsymbol{A}). \tag{5}$$

3 DISCRETE EVENT SIMULATION

Let (C, ϕ) be a binary monotone system with component state processes $\{X_1(t)\}, \ldots, \{X_n(t)\}$. For $i = 1, \ldots, n$ we denote the events affecting the process $\{X_i(t)\}$ by $E_{i1}, E_{i2}, \ldots$, listed in chronological order. Since we assumed that all lifetimes and repair times have absolutely continuous distributions, all these events happen at distinct points of time almost surely. We let $T_{i1} < T_{i2}, \ldots$ be the corresponding points of time for these events. We also let $T_{i0} = 0$, $i = 1, \ldots, n$. As in (1) the component state processes can then be expressed as:

$$X_i(t) = X_i(0) + \sum_{j=1}^{\infty} I(T_{ij} \leq t) J_{ij}, \qquad t \geq 0, \;\; i = 1, \ldots, n, \tag{6}$$

where the jumps J_{ij} are either -1 if E_{ij} is a failure event, or $+1$ if E_{ij} is a repair event. We assume that all components start out by being functioning. Thus, we have $X_i(0) = 1$, and $J_{ij} = (-1)^j$, for $i = 1, \ldots, n$ and $j = 1, 2, \ldots$. Finally, for $i = 1, \ldots, n$ we introduce the times between the events defined as:

$$\Delta_{ij} = T_{ij} - T_{ij-1}, \qquad i = 1, \ldots, n, \;\; j = 1, 2, \ldots. \tag{7}$$

Then for $i = 1, \ldots, n$ we have:

$$\Delta_{i1} = U_{i1}, \quad \Delta_{i2} = D_{i1}, \quad \Delta_{i3} = U_{i2}, \quad \ldots \tag{8}$$

Since $U_{i1}, U_{i2}, \ldots$ are independent and identically distributed with positive mean value μ_i, it follows by Proposition A.1 that X_i is a regular pure jump process, $i = 1, \ldots, n$. Hence, by Proposition A.4 the system state $\phi = \phi(\boldsymbol{X})$ as well as the criticality states $\psi_1(\boldsymbol{X}), \ldots, \psi_n(\boldsymbol{X})$ are regular pure jump processes.

At the system level the event set is the *union* of all the component event sets. Let $E^{(1)}, E^{(2)}, \ldots$ denote these events sorted with respect to their respective points of time, and let $T^{(1)} < T^{(2)} < \cdots$ be the corresponding points of time. Note that since we assumed that all lifetimes and repair times have absolutely continuous distributions, each system event corresponds almost surely to a unique component event.

In order to simulate such a system, we use an *object oriented approach* where the components as well as the system are represented as *objects*. The component objects are equipped with methods for generating failure and repair events according to their respective life- and repair time distributions. The system object determines the state of the system as a function of the component states. To keep track of the events and process them in the correct order, they are organized in a dynamic queue sorted with respect to the points of time of the events. The component processes place their upcoming events into the queue where they stay until they are processed.

More specifically, at time zero each component starts out by being functioning, and places its first failure event into the queue. As soon as all these failure events have been placed into the queue, the first event in the queue is processed. That is, the *system time* is set to the time of the first event, and the event is taken out of the queue and passed on to the component responsible for handling this event. The component then updates its state, generates a new event, in this case a repair event, which is placed into queue, and notifies the system about its new state so that the system state can be updated as well. Then the next event in the queue is processed in the same fashion, and so forth until the system time reaches a certain predefined point of time. Note that since the component events are generated as part of the event processing, the number of events in the queue stays constant.

3.1 *Sampling events*

Although the system state and component states stay constant between events, it may still be of interest to log the state values at predefined points of time. In order to facilitate this, we introduce yet another type of event, called a *sampling event*. Such sampling events will typically be spread out evenly on the timeline. Thus, if $e_1, e_2, \ldots$ denote the sampling events, and $t_1 < t_2 < \cdots$ are the corresponding points of time, we would typically have $t_j = j \cdot \Delta$ for some suitable number $\Delta > 0$.

The sampling events will be placed into the queue in the same way as for the ordinary events. As a sampling event is processed, the next sampling event will be placed into the queue. Thus, at any time only one sampling event needs to be in the queue.

3.2 *Updating system and criticality states*

In principle one must update the system state every time there is a change in the component states. For large complex systems, these updates may slow down the simulations considerably. Thus, whenever possible one should avoid computing the system state. Fortunately, since the structure function of a binary monotone system is non-decreasing in each argument, it is possible to reduce the updating to a minimum. To explain this in detail, we consider the event E_{ij} affecting component i. Let T_{ij} be the corresponding point of time, and let $\boldsymbol{X}(T_{ij}^-)$ denote the value of the component state vector immediately before E_{ij} occurs, i.e., $\boldsymbol{X}(T_{ij}^-) = \lim_{t \to T_{ij}^-} \boldsymbol{X}(t)$. Note that by Proposition A.2 these limits exist since the component state processes are regular.

If E_{ij} is a failure event of component i, i.e., $X_i(T_{ij}^-) = 1$ and $X_i(T_{ij}) = 0$, then the event cannot change the system state if the system is already failed, i.e., $\phi(\boldsymbol{X}(T_{ij}^-)) = 0$. Similarly, if E_{ij} is a repair event of component i, i.e., $X_i(T_{ij}^-) = 0$ and $X_i(T_{ij}) = 1$, this event cannot change the system state if the system is already functioning, i.e., $\phi(\boldsymbol{X}(T_{ij}^-)) = 1$. Thus, we see that we only need to recalculate the system state whenever:

$$\phi(\boldsymbol{X}(T_{ij}^-)) \neq X_i(T_{ij}). \tag{9}$$

Hence, the number of times we need to recalculate the system state is drastically reduced.

In cases where we keep track of the criticality state of each of the components, we can simplify the calculations even further by noting that the system state is changed as a result of the event E_{ij} if and only if component i is critical at the time of the event. Moreover, if i is critical, and E_{ij} is a failure event, it follows that the system fails as a result of this event, i.e., $\phi(\boldsymbol{X}(T_{ij})) = 0$. If on the other hand i is critical, and E_{ij} is a repair event, it follows that the system becomes functioning as a result of this event, i.e., $\phi(\boldsymbol{X}(T_{ij})) = 1$. Thus, we see that in this setup all the calculations we need to carry out, are related to the updating of the criticality states.

A similar technique can be used when updating the criticality states of the components. Thus, we consider the event E_{ij} affecting the state of component i. We first note that the criticality state function of component i, $\psi_i(\boldsymbol{X}(t)) = \phi(1_i, \boldsymbol{X}(t)) - \phi(0_i, \boldsymbol{X}(t))$ does not depend on the state of component i. Thus, the event E_{ij} does not have any impact on the criticality state of i. However, E_{ij} may still change the criticality state of other components in the system even when the system state remains unchanged. Thus, let $k \neq i$ be another component, and consider its criticality state function $\psi_k(\boldsymbol{X}(T_{ij}))$.

If $X_k(T_{ij}) = 1$ and $\phi(\boldsymbol{X}(T_{ij})) = 0$, it follows that $\phi(1_k, \boldsymbol{X}(T_{ij})) = \phi(0_k, \boldsymbol{X}(T_{ij})) = 0$. Thus, in this case we must have $\psi_k(\boldsymbol{X}(T_{ij})) = 0$. On the other hand, if $X_k(T_{ij}) = 0$ and

$\phi(\boldsymbol{X}(T_{ij})) = 1$, it follows that $\phi(1_k, \boldsymbol{X}(T_{ij})) = \phi(0_k, \boldsymbol{X}(T_{ij})) = 1$. Thus, we must have $\psi_k(\boldsymbol{X}(T_{ij})) = 0$ in this case as well. Hence, we see that a necessary condition for component k to be critical at time T_{ij} is that:

$$\phi(\boldsymbol{X}(T_{ij})) = X_k(T_{ij}). \tag{10}$$

Utilizing these observations reduces the need to recalculate the criticality states.

3.3 *Estimating availability and importance*

Stationary availability and importance measures are typically easy to derive. If the system under consideration is not too complex, these quantities can be calculated analytically using (2), (3) and (5). For larger complex systems one may estimate the availability and importance using Monte Carlo simulations. A fast simulation algorithm for this is provided in [5]. Alternatively, estimates can be obtained by running a *single* discrete event simulation on the system over a sufficiently long time horizon

Here, however, we focus on the problem of estimating the system availability $A_\phi(t)$ and the component importance measures $I_B^{(1)}(t), \dots, I_B^{(n)}(t)$ as functions of t. Ideally we would like to estimate these quantities for any $t \geq 0$. For practical purposes, however, we have to limit the estimation to a finite set of points. More specifically, we will estimate $A_\phi(t)$ for $t \in \{t_1, \dots, t_N\}$, i.e., the set of the N first sampling points. For the points of time between the sampling points, we just use linear interpolation to obtain the curve estimate.

A simple approach to this problem is to run M simulations on the system, where each simulation covers the time interval $[0, t_N]$. In each simulation we sample the values of ϕ and $\psi_1, \dots, \psi_n$ at each sampling point $t_1, \dots, t_N$. We denote the sth simulated value of the component state vector process at time $t \geq 0$ by $\boldsymbol{X}_s(t)$, $s = 1, \dots, M$, and obtain the following estimates for $j = 1, \dots, N$:

$$\hat{A}_\phi(t_j) = \frac{1}{M} \sum_{s=1}^{M} \phi(\boldsymbol{X}_s(t_j)), \tag{11}$$

$$\hat{I}_B^{(i)}(t_j) = \frac{1}{M} \sum_{s=1}^{M} \psi_i(\boldsymbol{X}_s(t_j)). \tag{12}$$

We will refer to these estimates as *pointwise estimates.* It is easy to see that for $j = 1, \dots, N$, $\hat{A}_\phi(t_j)$ and $\hat{I}_B^{(i)}(t_j)$ are unbiased and strongly consistent estimates of $A_\phi(t_j)$ and $I_B^{(i)}(t_j)$ respectively. In order to estimate $A_\phi(t)$ and $I_B^{(i)}(t)$ between the sampling points, one may use interpolation. Using a sufficiently high sampling rate, i.e., a small value of Δ, a satisfactory estimate of the full curve can be obtained. Still, all information about the process between the sampling points is thrown away.

We now present an alternative approach where we utilize process data between the sampling points as well. As above we assume that the system is simulated M times over the interval $[0, t_N]$, and let $\boldsymbol{X}_s(t)$ denote the sth simulated value of the component state vector process at time $t \geq 0$, $s = 1, \dots, M$. Then let $E_s^{(1)}, E_s^{(2)}, \dots$ denote the events in the interval $[0, t_N]$ in the sth simulation, *including* sampling events at times $t_1, \dots, t_N$, and let $T_s^{(1)} < T_s^{(2)} < \cdots$ be the corresponding points of time, $s = 1, \dots, M$. In this case we also include an extra sampling event in each simulation at time $t_0 = 0$, denoted $E_s^{(0)}$, and let $T_s^{(0)} = 0$, $s = 1, \dots, M$.

The idea now is to use average simulated availability and criticalities from each interval $[t_{j-1}, t_j)$, $j = 1, \ldots, N$ as respective estimates for the availability and criticalities at the midpoints of these intervals. By using Proposition A.3, we obtain the following estimates for $j = 1, \ldots, N$:

$$\tilde{A}_\phi(\bar{t}_j) = \frac{1}{M\Delta} \sum_{s=1}^{M} \sum_{k \in \mathcal{E}_s^{(j)}} \phi(\boldsymbol{X}_s(T_s^{(k)}))(T_s^{(k+1)} - T_s^{(k)}), \tag{13}$$

$$\tilde{I}_B^{(i)}(\bar{t}_j) = \frac{1}{M\Delta} \sum_{s=1}^{M} \sum_{k \in \mathcal{E}_s^{(j)}} \psi_i(\boldsymbol{X}_s(T_s^{(k)}))(T_s^{(k+1)} - T_s^{(k)}), \tag{14}$$

where $\mathcal{E}_s^{(j)}$ denotes the index set of the events in $[t_{j-1}, t_j)$ in the sth simulation, and where we have introduced the interval midpoints $\bar{t}_j = (t_{j-1} + t_j)/2$, $j = 1, \ldots, N$. We will refer to these estimates as *interval estimates.*

By using the right-continuity of the component state processes, it is easy to see that for $j = 1, \ldots, N$, $\tilde{A}_\phi(\bar{t}_j)$ and $\tilde{I}_B^{(i)}(\bar{t}_j)$ are unbiased and strongly consistent estimates of the corresponding average availability and criticality in the intervals $[t_{j-1}, t_j)$ respectively. By choosing Δ so that the availabilities and criticalities are relatively stable within each interval, the interval estimates are approximately unbiased estimates for $A_\phi(\bar{t}_j)$ and $I_B^{(i)}(\bar{t}_j)$ as well. In fact the resulting interval estimates tend to stabilize much faster than the pointwise estimates. In order to estimate $A_\phi(t)$ and $I_B^{(i)}(t)$ between the interval midpoints, one may again use interpolation. Note that since all process information is used in the estimates, satisfactory curve estimates can be obtained for a much higher value of Δ than the one needed for the pointwise estimates. In the next section we will demonstrate this on some examples.

4 NUMERICAL RESULTS

In order to illustrate the methods presented in Section 3 we consider a simple bridge system shown in Figure 1. The components of this system are the five edges in the graph, labeled $1, \ldots, 5$. The system is functioning if the source node s can communicate with the terminal node t through the graph. All the components in the system have exponential lifetime and repair time distributions with mean values 1 time unit. The objective of the simulation is to estimate $A_\phi(t)$ and $I_B^{(1)}(t), \ldots, I_B^{(5)}(t)$ for $t \in [0, t_N]$, where $t_N = 1000$.

All the simulations were carried out using a program called *Eventcue*[1]. This program has an intuitive graphical user interface, and can be used to estimate availability and criticality of any undirected network system.

Since all the lifetimes and repair times are exponentially distributed with the *same mean*, it is easy to derive explicit analytical expressions for the component availabilities. To see this, we consider the ith component at a given point of time t and introduce $N_i(t)$ as the number of failure and repair events affecting component i in $[0, t]$. With times between events being independent and exponentially distributed with mean 1 it follows that $N_i(t)$ has a Poisson distribution with mean t. Moreover, component i is functioning at time t if

[1] Eventcue is a java program developed at the Department of Mathematics, University of Oslo. The program is freely available at http://www.riscue.org/eventcue/.

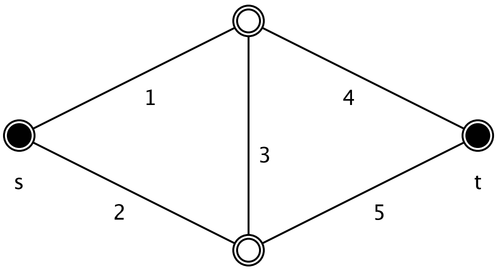

Figure 1: *A bridge system.*

and only if $N_i(t)$ is even. Thus, the ith component availability at time t is given by:

$$A_i(t) = \sum_{k=0}^{\infty} \Pr(N_i(t) = 2k) = \sum_{k=0}^{\infty} \frac{t^{2k}}{(2k)!} e^{-t}. \tag{15}$$

Using (15) one can verify numerically that all the component availabilities converge very fast towards their common stationary value, 0.5. As a result of this the system availability, $A_\phi(t)$, converges very fast towards its stationary value, 0.5, as well. In fact, for $t > 20$, numerical calculations show that $|A_\phi(t) - 0.5| < 10^{-15}$. Similarly, the Birnbaum measures of importance converges so that for $t > 20$, $|I_B^{(i)}(t) - 0.375| < 10^{-15}$, $i = 1, 2, 4, 5$, while $|I_B^{(3)}(t) - 0.125| < 10^{-15}$. Thus, for $t > 20$ the true values of all the curves are approximately constant. This makes it easy to evaluate and compare the quality of the different Monte Carlo estimates in this particular case.

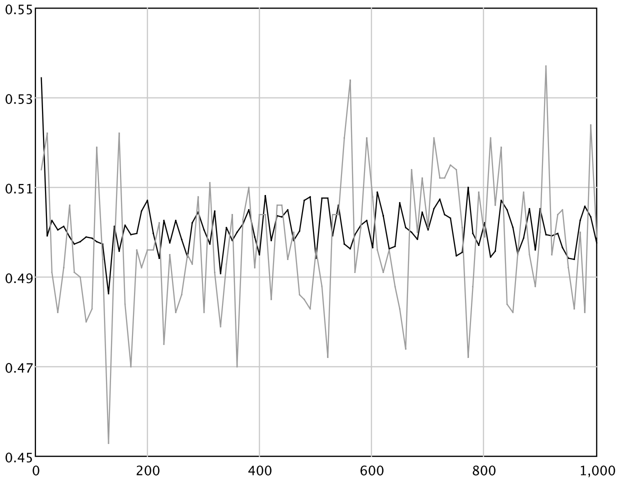

Figure 2: *Interval estimate (black curve) and pointwise estimate (gray curve) of the availability curve.*

Figure 2 and Figure 3 show respectively the availability curve and the criticality curve of component 1. The black curves are obtained using the interval estimates, while the

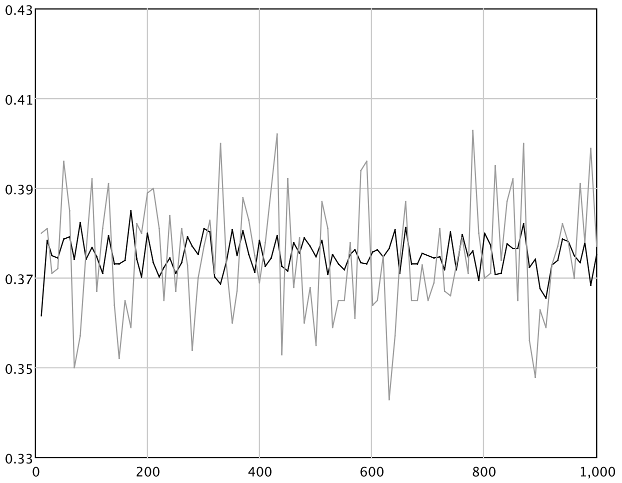

Figure 3: *Interval estimate (black curve) and pointwise estimate (gray curve) of the importance curve.*

gray curves show the corresponding pointwise estimate curves. In all cases we have used $M = 1000$ simulations and $N = 100$ sample points.

The plots clearly show the difference between the two methods. The black interval estimate curves are much more stable, and thus much closer to the true curve values, compared to the gray pointwise estimates.

One may think that increasing the number of sampling points would make the pointwise curve estimate better as more information is sampled. However, it turns out that the main effect of this is that the curve jumps more and more up and down. In fact with shorter intervals between sampling points the interval estimate becomes more unstable as well, and in the limit where the interval lengths go to zero, the two methods become equivalent. The only effective way of stabilizing the results for the pointwise curve estimate is to increase the number of simulations, i.e., M.

M	2000	4000	6000	8000
St.dev.	0.0121	0.0076	0.0062	0.0054

Table 1: Standard deviations for the pointwise curve estimates

In Table 1 we have listed estimated standard deviations for pointwise curve estimates for different values of M. We see that the standard deviation shows a steady decline as M increases. The corresponding numbers for $M = 1000$ are 0.0055 for the interval curve estimate and 0.0148 for the pointwise estimate. Thus, in this particular case we see that to obtain a pointwise curve estimate with a comparable stability to the interval curve estimate, one needs about eight times as many simulations.

For the interval curve estimate it is possible to obtain an even smoother curve simply by increasing Δ. Still, in general Δ should not be made too large, as this could produce a curve where important effects are obscured. Thus, in order to obtain optimal results, one should try out different values for Δ, and balance smoothness against the need of capturing significant oscillation properties of the curve.

Now, if smoothness is important, it is of course possible to apply some standard smoothing technique, such as moving averages or exponential smoothing, to the pointwise curve estimate. While such post-smoothing would clearly make the curve smoother, this technique does not add any new information to the estimate. The main advantage with the interval curve estimates is that such estimates actually use information about all events. Especially in cases where events occur at a very high rate, this turns out to be a great advantage.

5 APPLICATIONS TO IMPORTANCE MEASURE ESTIMATION

In this section we shall explain how the sampling methods developed in Section 3 can be used to estimate more advanced importance measures like e.g., those introduced in [11] and applied in [10]. In the context of the present paper the general idea can be explained as follows. As before we consider a binary monotone system (C, ϕ). Moreover, let $i \in C$ be a component in the system, and let $E_{i1}, E_{i2}, \ldots$ be the events affecting this component occurring respectively at $T_{i1} < T_{i2}, \cdots$. For each of these events we then introduce new *fictive* events $E'_{i1}, E'_{i2}, \ldots$ occurring respectively at $T'_{i1} < T'_{i2} < \cdots$. We assume that the fictive events always occur after their respective *real* events. That is, $T_{ij} < T'_{ij}$, $j = 1, 2, \ldots$. The fictive events could represent the results of some sort of fictive action altering how the state of the component interacts with the system throughout the interval between the real event and the corresponding fictive event. If E_{ij} is a failure event, then E'_{ij} could e.g., be a fictive failure event occurring as a result of the component being minimally repaired at T_{ij} and then functioning until T'_{ij}. Similarly, if E_{ij} is a repair event, one may consider fictive actions, such as e.g., a fictive minimal failure at T_{ij} that extends the repair interval until T'_{ij}. For a precise definition of the concept of minimal repairs and failures, we refer to [11]. The effect on the system of such fictive actions typically says something about the importance of the component. In any case, however, unless the component is critical at some point during the interval $[T_{ij}, T'_{ij})$, the system will not be affected by the fictive action. This motivates the definition of the following random variable ($i = 1, \ldots, n$):

$$Z_i(t_N) = \int_0^{t_N} \sum_{j=1}^{\infty} c_{ij} \cdot I(T_{ij} \leq t < T'_{ij}) \psi_i(\boldsymbol{X}(t)) dt, \tag{16}$$

where $c_{ij} = c_F$ if E_{ij} is a failure event, and $c_{ij} = c_R$ if E_{ij} is a repair event, and where c_F and c_R are suitable known constants, typically 0 or 1. Note that if $c_F = 1$, all the fictive minimal repairs occurring in $[0, t_N]$ will be included as contributions to $Z_i(t_N)$, while if $c_F = 0$, these fictive actions will be ignored. Similarly, if $c_R = 1$, all the fictive minimal failures occurring in $[0, t_N]$ will be included as contributions to $Z_i(t_N)$, while if $c_R = 0$, these fictive actions will be ignored. The expected values of the variables $Z_1, \ldots, Z_n$ now serve as a basis for an importance measure. In particular, it can be shown that the importance measures introduced in [11] can be obtained in this way. The so-called extended Natvig measure is for instance obtained by setting $c_F = c_R = 1$.

Since the variable $Z_i(t_N)$ involves both real and fictive events, estimating its expectation using standard discrete event simulation can be a complex task. While the real events represent a single possible sequence of changes in the states of the system and its components, each of the fictive events introduces an alternative sequence of state changes. Note in particular that it may happen that a fictive event, E'_{ij}, occurs after the *next* real event, E_{ij+1}, in which case the intervals $[T_{ij}, T'_{ij})$ and $[T_{ij+1}, T'_{ij+1})$ overlap. Hence, keeping track of all the different parallel sequences of events is indeed a challenge. Armed with the methods introduced in the present paper, however, the problem can easily be solved. In order to study this in further detail we first note that since the component processes are assumed to be independent of each other, we have:

$$\begin{aligned} E[Z_i(t_N)] &= \int_0^{t_N} [\sum_{j=1}^{\infty} c_{ij} \Pr(T_{ij} \leq t < T'_{ij})] I_B^{(i)}(t) dt \qquad (17) \\ &= \int_0^{t_N} \omega_i(t) I_B^{(i)}(t) dt, \end{aligned}$$

where we have introduced the weight function:

$$\omega_i(t) = E[\sum_{j=1}^{\infty} c_{ij} I(T_{ij} \leq t < T'_{ij})] = \sum_{j=1}^{\infty} c_{ij} \Pr(T_{ij} \leq t < T'_{ij}). \qquad (18)$$

Now, by running a separate discrete event simulation for each of the components the weight functions, $\omega_1, \ldots, \omega_n$, can easily be estimated using similar techniques as the ones discussed in the previous sections. More specifically, we introduce the processes $W_1(t), \ldots, W_n(t)$ defined by:

$$W_i(t) = \sum_{j=1}^{\infty} c_{ij} I(T_{ij} \leq t < T'_{ij}), \quad t \geq 0,\, i = 1, \ldots, n. \qquad (19)$$

To simplify the expressions it is convenient to introduce a common notation for *all* events, real or fictive, affecting the process $W_i(t)$, $i = 1, \ldots, n$. We sort these events in chronological order and denote them by $E_i^{(1)}, E_i^{(2)}, \ldots$. Moreover, we let $T_i^{(1)} < T_i^{(2)} < \cdots$ be the points of time corresponding to these events.

Since we have assumed that the fictive events always occur after their respective real events, it is easy to see that the processes $W_1(t), \ldots, W_n(t)$ are regular pure jump processes that can be written as:

$$W_i(t) = W_i(0) + \sum_{j=1}^{\infty} I(T_i^{(j)} \leq t) J_i^{(j)}, \quad t \geq 0,\, i = 1, \ldots, n, \qquad (20)$$

where $W_1(0) = \cdots = W_n(0) = 0$, and where the jumps are given by:

$$J_i^{(j)} = \begin{cases} +c_F & \text{if } E_i^{(j)} \text{ is a real failure event} \\ -c_F & \text{if } E_i^{(j)} \text{ is a fictive failure event} \\ +c_R & \text{if } E_i^{(j)} \text{ is a real repair event} \\ -c_R & \text{if } E_i^{(j)} \text{ is a fictive repair event} \end{cases} \qquad i = 1, \ldots, n,\ j = 1, 2, \ldots. \qquad (21)$$

From (18) and (19) we have $E[W_i(t)] = \omega_i(t)$, $t \geq 0$, $i = 1, \ldots, n$. Thus, in order to estimate the weight functions $\omega_1(t), \ldots, \omega_n(t)$ for $t \in [0, t_N]$, we run M simulations for each of the processes $W_1, \ldots, W_n$ over this interval. Unbiased and strongly consistent pointwise estimates of the weight functions at sample points $t_1, \ldots, t_N$ are then obtained using the following formula:

$$\hat{\omega}_i(t_j) = \frac{1}{M} \sum_{s=1}^{M} W_{is}(t_j), \quad j = 1, \ldots, N,\ \ i = 1, \ldots, n, \qquad (22)$$

where $W_{is}(t)$ denotes the value of $W_i(t)$ at time $t \geq 0$ in the sth simulation, $s = 1, \ldots, M$, $i = 1, \ldots, n$.

Alternatively, we can obtain interval estimates of the weight functions in the same way as we did for the reliability and criticality functions. As above we assume that the processes $W_1, \ldots, W_n$ are simulated M times over the interval $[0, t_N]$, and let $W_{is}(t)$ denote the value of $W_i(t)$ at time $t \geq 0$ in the sth simulation, $s = 1, \ldots, M$, $i = 1, \ldots, n$. Then for $i = 1, \ldots, n$, we let $E_{is}^{(1)}, E_{is}^{(2)}, \ldots$ denote the events affecting the process W_i in the interval $[0, t_N]$ in the sth simulation, *including* the sampling events, and let $T_{is}^{(1)} < T_{is}^{(2)} < \cdots$ be the corresponding points of time, $s = 1, \ldots, M$. As for the previous interval estimates we also include an extra sampling event in each simulation at time $t_0 = 0$, denoted $E_{is}^{(0)}$, and let $T_{is}^{(0)} = 0$, $s = 1, \ldots, M$.

The interval estimates for the weight functions are then obtained by using average simulated values of the processes $W_1, \ldots, W_n$ from each interval $[t_{j-1}, t_j)$, $j = 1, \ldots, N$ as estimates for the respective weight functions at the midpoints of these intervals. By using Proposition A.3 we obtain the following estimates for $j = 1, \ldots, N$:

$$\tilde{\omega}_i(\bar{t}_j) = \frac{1}{M\Delta} \sum_{s=1}^{M} \sum_{k \in \mathcal{E}_{is}^{(j)}} W_{is}(T_{is}^{(k)})(T_{is}^{(k+1)} - T_{is}^{(k)}), \tag{23}$$

where $\mathcal{E}_{is}^{(j)}$ denotes the index set of the events affecting the process W_i in $[t_{j-1}, t_j)$ in the sth simulation, and where the interval midpoints are $\bar{t}_j = (t_{j-1} + t_j)/2$, $j = 1, \ldots, N$.

In particular, one may use both pointwise curve estimates as well as interval estimates, denoted respectively by $\hat{\omega}_i$ and $\tilde{\omega}_i$. Combining these estimates with the respective estimates for $I_B^{(i)}$ we get the following estimates for $E[Z_i(t_N)]$, $i = 1, \ldots, n$:

$$E[Z_i\hat{(}t_N)] = \int_0^{t_N} \hat{\omega}_i(t) \hat{I}_B^{(i)}(t) dt, \tag{24}$$

$$E[Z_i\tilde{(}t_N)] = \int_0^{t_N} \tilde{\omega}_i(t) \tilde{I}_B^{(i)}(t) dt, \tag{25}$$

where the integrals are easily calculated numerically. Note that one should generally not mix pointwise curve estimates and interval curve estimates. The reason for this is that the pointwise curve estimates provide unbiased estimates for the curve values at the sampling points, while interval estimates provide unbiased estimates for the average curve values over the corresponding intervals. Thus, by mixing the two, the result may not be unbiased. In the stationary phase, this issue is negligible. However, in the initial phase, the error resulting from this may be significant.

In cases where several different importance measures are used, each with its own weight function, the above technique allows us to reuse the curve estimate for $I_B^{(i)}$ when calculating each of the measures. This makes it easier and faster to compare the different measures.

We close this section by applying the proposed methods to the example considered in Section 4. That is, we consider once again the bridge system shown in Figure 1, and focus on component 1. Our first goal is to estimate the weight function $\omega_1(t)$ given in (18) for $t \in [0, t_N]$, where $t_N = 1000$. In this particular case we let $c_F = 1.0$ while $c_R = 0.0$. Thus, only effects of the fictive failure events are included. Moreover, if E_{1j} is a (real) failure event occurring at time T_{1j} then the corresponding fictive failure event, denoted E'_{1j} occurring at time T'_{1j} is a result of component 1 being minimally repaired at T_{1j} and then functioning until T'_{1j}. The time between the real and fictive events is easily generated using a standard rejection method.

In order to obtain an estimate of $\omega_1(t)$, we run $M = 1000$ simulations of the process W_1 defined in (19) with $N = 100$ sample points. The resulting curve estimates are shown in Figure 4. As before, the black curve is obtained using the interval estimates, while the gray curve shows the corresponding pointwise estimate curve. As for the availability and criticality curve estimates, the interval method produces more stable results.

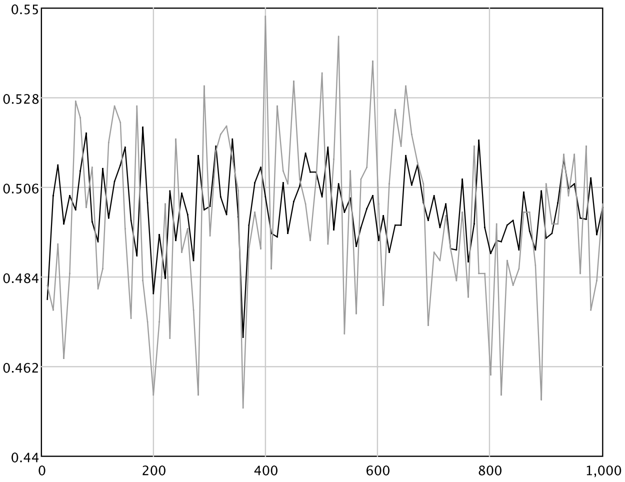

Figure 4: *Interval estimate (black curve) and pointwise estimate (gray curve) of the weight function* $\omega_1(t)$

Having estimates for both $\omega_1(t)$ and $I_B(t)$ we can the proceed to estimating $E[Z_1(t_N)]$ as defined in (17). This is done by calculating numerically the integrals (24) and (25). Since, however, $E[Z_1(t_N)]$ typically is an unbounded function of t_N it is often more convenient to work with a normalized version of the form $E[Z_1(t_N)/t_N]$. In Figure 5 we have plotted the resulting normalized estimates as functions of t_N. The black curve is derived using the interval estimates, while the gray curve is obtained using pointwise estimates. We observe that in this case the two methods produce almost identical results, although the interval estimates are slightly more stable, especially for small values of t_N. The reason for this is that the integrals tend to smoothen the curve estimates considerably. This effect makes the increased precision obtained by using interval estimates less significant.

For more examples of the use of this technique see the companion paper [10].

6 CONCLUSIONS

In the present paper we have discussed two different approaches to curve estimation in discrete event simulations. In particular, we have indicated that using interval estimates may produce more stable curve estimates compared to pointwise estimates. The proposed methods are particularly useful in relation to importance measure estimation, especially when several different importance measures are calculated and compared.

An important parameter used in the curve estimates is the distance between the sampling points, i.e., Δ. Finding a suitable value for this parameter, may be challenging as it depends

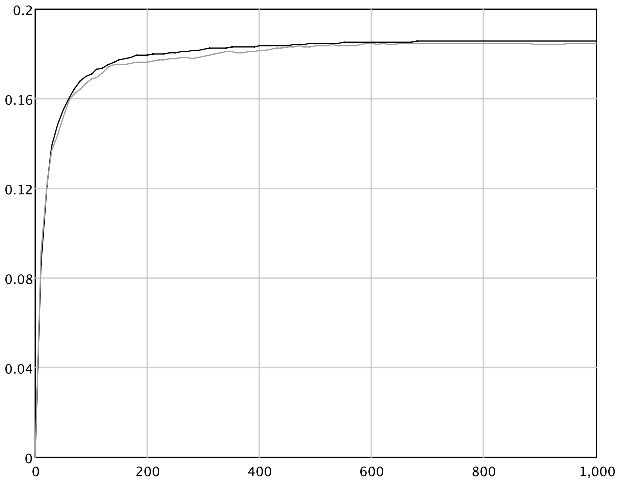

Figure 5: *Interval estimate (black curve) and pointwise estimate (gray curve) of $E[Z_1(t_N)]/t_N$ for different values of t_N.*

on how fast the underlying processes converge to a stationary state. Note, however, that it is not necessary to use the same distance between the sampling points throughout the sampling period. Instead it is possible to use shorter distances between the sampling points in the early stage, where the processes have not converged, and then use longer distances as soon as all the processes have entered an approximate stationary state. By studying this issue further, we think that the proposed methods can be improved considerably.

References

[1] R. E. Barlow and F. Proschan. *Statistical Theory of Reliability and Life Testing.* To Begin With – Silver Spring MD, 1981.

[2] Z. W. Birnbaum. On the importance of different components in a multicomponent system. In P. R. Krishnaia, editor *Multivariate Analysis - II*, pp. 581–592, 1969.

[3] P. Glasserman and D. D. Yao. *Monotone Structure in Discrete-event Systems.* John Wiley and Sons, Inc., 1994.

[4] P. Glasserman. *Monte Carlo Methods in Financial Engineering.* Springer Verlag, 2004.

[5] A. B. Huseby and M. Naustdal. Improved Simulation Methods for System Reliability Evaluation. In *Mathematical and Statistical Methods in Reliability.*, pp. 105–121. World Scientific Publishing Co. Pte. Ltd., 2003.

[6] F. C. Klebaner. *Introduction to stochastic calculus with applications.* Imperial College Press, 2005.

[7] B. Natvig. A suggestion of a new measure of importance of system components. *Stochastic Process. Appl.*, (9): 319–330, 1979.

[8] B. Natvig. On the reduction in remaining system lifetime due to the failure of a specific component. *J. Appl. Prob.*, (19): 642–652, 1982. Correction J. Appl. Prob. 20, 713, 1983.

[9] B. Natvig. New light on measures of importance of system components. *Scand. J. Statist.*, (12): 43–54, 1985.

[10] B. Natvig and K. A. Eide and J. Gåsemyr and A. B. Huseby and S. L. Isaksen. The Natvig measures of component importance in repairable systems applied to an offshore oil and gas production system, 2008. Submitted.

[11] B. Natvig and J. Gåsemyr. New results on the Barlow-Proschan and Natvig measures of component importance in nonrepairable and repairable systems. *Methodol. Comput. Appl. Prob.*, 2009. To appear.

A Regular pure jump processes

In this appendix we present a few basic results on pure jump processes needed in the present paper. We consider a pure jump process S with jumps at $T_1 < T_2 < \cdots$. We also introduce the times between the events defined as:

$$\Delta_j = T_j - T_{j-1}, \qquad j = 1, 2, \ldots. \tag{26}$$

Using these quantities the event times can be expressed as:

$$T_k = \sum_{j=1}^{k} \Delta_j, \qquad k = 1, 2, \ldots. \tag{27}$$

Obviously, the process S is regular if and only if $T_\infty = \infty$ almost surely. Thus, it follows that a necessary and sufficient criterion for regularity is that the series $\sum_{j=1}^{\infty} \Delta_j$ is *divergent* with probability one. This condition can often be verified using the following simple result:

Proposition A.1 *Let S be a pure jump process with jumps at $T_1 < T_2 < \cdots$. Moreover, we let $T_0 = 0$ and introduce the non-negative random variables $\Delta_j = T_j - T_{j-1}$, j = 1, 2, ... Assume then that the sequence $\{\Delta_j\}$ contains an infinite subsequence $\{\Delta_{k_j}\}$ of independent, identically distributed random variables such that $E[\Delta_{k_j}] = d > 0$. Then S is regular.*

Proof: By the strong law of large numbers it follows that:

$$P(\lim_{n\to\infty} n^{-1} \sum_{j=1}^{n} \Delta_{k_j} = d) = 1.$$

This implies that the series $\sum_{j=1}^{\infty} \Delta_{k_j}$ is divergent with probability one. Hence, since obviously $\sum_{j=1}^{\infty} \Delta_{k_j} \leq \sum_{j=1}^{\infty} \Delta_j$, the result follows ■

The regularity property implies that the set of points where the process jumps does not have any accumulation points. The following result utilizes this to show the existence of left limits of the state function of a regular pure jump process.

Proposition A.2 *Let S be a regular pure jump process with jumps at $T_1 < T_2 < \cdots$. Then $\lim_{t\to s^-} S(t)$ exists for every $s > 0$ with probability one.*

Proof: Let $0 \leq t < s < \infty$. We then consider the set $\mathcal{T} = \{T_j : t \leq T_j < s\} \cup \{t\}$. Since S is assumed to be regular, the number of elements in $\mathcal{T}$ is finite with probability one. Moreover, $\mathcal{T}$ is non-empty since $t \in \mathcal{T}$. Thus, this set contains a maximal element, which we denote by t'. Moreover, since every element in $\mathcal{T}$ is less than s, then so is t'. From this it follows that the interval (t', s) is nonempty. At the same time (t', s) does not contain any jumps, so $S(t)$ is constant throughout this interval. Hence, $\lim_{t \to s^-} S(t)$ exists. Since s was arbitrary chosen, this holds for any $s > 0$ ■

Regularity is also of importance when considering the integral of a pure jump process:

Proposition A.3 *Let S be a regular pure jump process with jumps at $T_1 < T_2 < \cdots$, and let $0 \leq u < v < \infty$. Assume that $\{T_j : u < T_j < v\} = \{T^{(1)}, \ldots, T^{(k)}\}$, where $T^{(1)} < \cdots < T^{(k)}$. Moreover, we let $T^{(0)} = u$ and $T^{(k+1)} = v$. Then we have:*

$$\int_u^v S(t)dt = \sum_{j=0}^{k} S(T^{(j)})(T^{(j+1)} - T^{(j)}).$$

Proof: We first note that since S is assumed to be regular, the number of elements in the set $\{T_j : u < T_j < v\}$ is finite with probability one. Thus, this set can almost surely be written in the form $\{T^{(1)}, \ldots, T^{(k)}\}$, for some suitable $k < \infty$. Since S is right-continuous and piecewise constant, it follows that $S(t) = S(T^{(j)})$ for all $t \in [T^{(j)}, T^{(j+1)})$, $j = 0, 1, \ldots, k$. Thus, we have:

$$\int_{T^{(j)}}^{T^{(j+1)}} S(t)dt = S(T^{(j)})(T^{(j+1)} - T^{(j)}), \quad j = 0, 1, \ldots, k.$$

The result then follows by adding up the contributions to the integral from each of the $k+1$ intervals $[T^{(0)}, T^{(1)}), \ldots, [T^{(k)}, T^{(k+1)})$ ■

We then consider a system consisting of a collection of n regular pure jump processes, $S_1, \ldots, S_n$. The state of the system is then typically expressed as a function of the states of the elementary processes. It is easy to see that the system state also evolves as a regular pure jump process. That is, we have:

Proposition A.4 *Let $\boldsymbol{S} = (S_1, \ldots, S_n)$ denote a vector of regular pure jump processes, and let H be a process such that $H = H(\boldsymbol{S})$. Then H is a regular pure jump process as well. That is, $H(t) = H(\boldsymbol{S}(t))$ is piecewise constant and right-continuous in t, and the number of jumps in any finite interval is finite with probability one.*

Proof: Let $\mathcal{T}_i$ be the set of time points corresponding to the jumps of the process S_i, $i = 1, \ldots, n$, and let $\mathcal{T}$ be the set of time points corresponding to the jumps of the process H. Since the state value of H cannot change unless there is a change in the state value of at least one of the elementary processes, it follows that $\mathcal{T} \subseteq (\mathcal{T}_1 \cup \cdots \cup \mathcal{T}_n)$. Thus, $H(t)$ is piecewise constant and right-continuous in t. Moreover, for any finite interval $[t, s]$ we also have:

$$\mathcal{T} \cap [t, s] \subseteq [(\mathcal{T}_1 \cap [t, s]) \cup \cdots \cup (\mathcal{T}_n \cap [t, s])].$$

Since by regularity $(\mathcal{T}_i \cap [t, s])$ is finite for $i = 1, \ldots, n$, it follows that $\mathcal{T} \cap [t, s]$ is finite as well. Hence, we conclude that H is regular ■