ON THE LINEAR GROWTH OF THE SPLIT-AND-MERGE SIMULATION TREE FOR A MULTICOMPONENT AGE REPLACEMENT MODEL

Felisa J. Vázquez-Abad, Pierre L'Ecuyer and Benoit Martin

Département d'informatique et de recherche opérationnelle, Université de Montréal
C.P. 6128, Succ Centre-Ville, Montréal, H3C 3J7.
Tel. (514) 343-6952, Fax (514) 343-5834
vazquez@iro.umontreal.ca lecuyer@iro.umontreal.ca

Keywords

Functional estimation, split-and-merge tree, optimization, maintenance models, age replacement policies

Abstract

We a consider a replacement policy based on age thresholds, for a multicomponent system. We want to estimate the expected cost of operating the system as a function of one of the thresholds, in functional form, by simulation, and eventually to optimize. We investigate the effectiveness of a Split-and-Merge approach for doing that.

1 Introduction

We focus on a system with M identical components subject to failures. Each time a component fails it must be replaced by a new one to keep the system under operating conditions. The component lifetimes are independent and identically distributed (i.i.d.) and the replacements are assumed instantaneous. We consider a particular class of replacement policies where an intervention occurs when a component fails or reaches an age threshold $\bar{\theta}$. At any of these interventions, we replace the failed component (if any) and also replace preventively each component whose age exceeds θ, where $\theta < \bar{\theta}$ is a second threshold. Given the failure rate function of the components and some cost structure which penalizes failures, interventions, and replacements, our aim is to optimize the thresholds θ and $\bar{\theta}$, to minimize the expected cost over some finite horizon, or the average cost over an infinite horizon. Of course, for this to make sense, we assume that the failure rate is eventually increasing with respect to the age of a component.

The Split-and-Merge approach is a recently proposed functional estimation method, to estimate a mathematical expectation as a function of a threshold parameter [7, 11] over a given interval. The idea of the method is as follows. Up up to the time of the first intervention, a single simulation is valid for the entire interval of values of the parameter of interest θ (in this paper, we focus on the second threshold parameter θ). The first time a replacement decision is not the same for all values of θ in the interval, this interval is split at the point(s) where the decision changes as a function of θ, and the simulation is continued in each of the subintervals (with a different sample trajectory for each subinterval). This continues recursively: each subinterval is split whenever a replacement decision differs for different values of θ in this subinterval, and so on. This gives rise to a growing *simulation tree*, where each node of the tree corresponds to an ongoing simulation. Merging can occur when adjacent nodes in the tree reach a common regeneration point (e.g., all components are replaced by new ones).

By replicating this experiment a number of times and averaging, one obtains an estimator of the cost function over the whole interval of values θ. This can be used to estimate the optimum, or for other purposes.

It is shown in [7, 11] that under the assumptions that all the decisions are binary and depend on the system's evolution only via a threshold parameter θ, the simulation tree grows only linearly as a function of the number of decisions taken during the simulation. In the age replacement problem considered here, these assumptions are not satisfied, since the decisions made at the interventions times are not binary and (much more importantly) depend of the component lifetimes. At any intervention, for any node of the tree, splitting can occur into a (random) number of nodes from 1 to M. So, in the worst case scenario the tree may grow exponentially fast. But is that really what typically happens in practice ?

In this work we study the growing speed of the Split-and-Merge simulation tree for our multicomponent replacement problem, as a function of the number of interventions n and of the number of compo-

proc. of the workshop on Discrete Event Systems

nents M, both empirically and theoretically. Good surprise: the empirical analysis suggests that the tree size is *linear* in both n and M. We prove a bound which is linear in n and quadratic in M for the case where the interval of interest contains only small values of θ (smaller than $\bar{\theta}/2$). The proof for the more general case is more involved and will be included in a longer version of this paper. The simulation results are from [8].

2 Problem Formulation

2.1 Age Replacement Policies

Several age-replacement-type maintenance models have been treated in the literature; see, e.g., [2, 3, 4, 9]. The particular model considered in this paper is taken from [3, 4]. The system is comprised of M identical components, subject to failures, and with i.i.d. lifetimes. Each time a component fails, it must be replaced immediately. Preventive replacements of other components may also be performed at the same occasion, or at any other time. Replacements are instantaneous. We call an *intervention* the action (or event) of replacing one or several components (simultaneously), whether forced or preventive. Each intervention has a fixed cost c_1, plus a variable cost c_2 for each component replaced. An additional cost c_3 is incurred if the intervention is forced by a failure.

The objective function is the expected operating cost of the system, either over a finite horizon, or an infinite horizon with discounting, or as a time-average over an infinite horizon. For this model, we define the *state* of the system as a vector that contains the ages of all the components, plus the current time if the horizon is finite. A control policy is a rule which, for each state, tells us which decision to make: to stop the system for a preventive intervention, or to let it go, and which component(s) to replace when the system is stopped. This gives rise to an extremely rich class of policies. Unless the number of components is very small (say, less than 3 or 4), an optimal control policy turns out to be extremely complicated to compute and implement. L'Ecuyer [4] and Haurie and L'Ecuyer [3] compute the optimal policy via dynamic programming combined with finite element approximation. They give a numerical illustration for a three-component system and show that in general (for $M \geq 3$), the optimal policy is not even monotone with respect to the component ages. The computational effort also renders their method impractical as the number of components increases, say beyond half a dozen.

As a more practical alternative, L'Ecuyer [4] proposes to restrict the control to a class of (very simple) suboptimal policies determined by two thresholds $0 < \theta < \bar{\theta}$, as explained in the introduction. His experiments with three-component systems showed (for the examples he examined) that the best policy of that class was almost as good as the optimal one. Then comes the issue of how to optimize these two thresholds. For example, for a fixed value of $\bar{\theta}$, frequent replacements (a small θ) tend to reduce failure costs but to increase the fixed costs of the interventions as well as the replacement costs. Infrequent replacements (a large θ) do the opposite.

One way of estimating the optimum is to estimate the objective as a function of the parameters, in functional form, then optimize this sample function. This corresponds to the general idea of the *stochastic counterpart* method [10] (although these authors use a different method to obtain the functional estimation; see also [6]). Another optimization approach could be *stochastic approximation* combined with some simulation-based gradient estimation method. But an efficient solution to this problem has so far eluded the main gradient estimation methods such as the likelihood ratio and perturbation analysis (PA), mainly because the sample performance is generally discontinuous with respect to the threshold parameters [5]. Fu, Hu, and Shi [1] developed a PA-based derivative estimator for a simplified version of the problem (there was no failure cost). To estimate the gradient of the expected cost over a time horizon T, at a single value of θ, his method requires $\mathcal{O}(T)$ auxiliary simulations, in addition to the main simulation run.

In this work, we focus on obtaining a functional estimator only for the parameter θ (for $\bar{\theta}$ fixed). We assume that the component lifetimes have distribution G, with density g, reliability function $\bar{G}$, defined by $\bar{G}(t) = 1 - G(t)$, and failure rate function λ, defined by $\lambda(t) = g(t)/\bar{G}(t)$, for $t \geq 0$. We assume that the failure rate is bounded over the interval $[0, \bar{\theta}]$; that is, $\lambda(t) \leq \bar{\lambda}$ for some constant $\bar{\lambda}$.

For a given value of $(\theta, \bar{\theta})$, for $n = 0, 1, 2, \ldots$, let T_n be the time of the nth intervention, and $Z_n = (Z_{n,1}, \ldots, Z_{n,M})$ be the M-dimensional age vector of the components at time T_n, immediately after the replacement(s). Each vector $Z_n = (Z_{n,1}, \ldots, Z_{n,M})$ has all its components in the interval $[0, \theta)$ and at least one component equal to zero. We take $T_0 = 0$ and assume that initially, all the components are new, so $Z_0 = (0, \ldots, 0)$. The process $\{(T_n, Z_n), n \geq 0\}$ is a Markov renewal process, as pointed out in [3, 4]. Its evolution depends of course on the value of θ. The total cost incurred for the first N interventions is

$$C_N(\theta) = \sum_{n=1}^{N} [c_1 + c_2\rho_n + c_3\delta_n],$$

where ρ_n is the number of components replaced at intervention n, δ_n is the indicator function that the nth intervention was forced, and the parameter θ indicates that the value of $C_N(\theta)$ varies as a function of θ. In this paper, we consider a fixed time horizon T. If $N(T,\theta)$ denotes the number of interventions up to time T, then $\mathcal{C}(T,\theta) = E[C_{N(T,\theta)}(\theta)]$ is the total expected cost, and we aim to estimate the function $\mathcal{C}(T,\cdot)$ over some interval $[a,b)$. The problem of estimating the average cost over an infinite-horizon is the subject of on-going research and will be discussed elsewhere.

2.2 Functional Estimation

Let (Ω, Σ, P) denote the probability space underlying our stochastic model, and $\omega \in \Omega$ a sample point. We assume that the probability space is independent of the threshold parameters. For instance, ω can be viewed as the successive lifetimes of all the components, or as the sequence of random numbers used to drive the simulation, assuming that these are i.i.d. uniform over the interval $(0,1)$. Thus, the evolution of the Markov renewal process depends on $(\theta, \bar{\theta})$ only through the replacement decisions.

The functional estimator will be the average of R i.i.d. copies of the sample cost function $\{C_{N(T,\theta)}(\theta),\ \theta \in [a,b)\}$, where $[a,b)$ is the interval of interest and $0 \le a < b \le \bar{\theta}$. Call this estimator

$$\hat{C}(\theta) = \frac{1}{R}\sum_{i=1}^{R} \tilde{C}_i(\theta), \qquad \theta \in [a,b),$$

where $\tilde{C}_i(\cdot)$ is the ith copy of $C_{N(T,\cdot)}(\cdot)$. Under appropriate conditions (see [10]), if $\theta^* \in (a,b)$ is the optimal value of θ, then $\hat{\theta}^*(N) = \arg\min_\theta \hat{C}(\cdot)$ converges to θ^* at rate $\mathcal{O}(R^{-1/2})$.

We now explain how to compute the sample function $C_{N(T,\cdot)}(\cdot)$ via a Split-and-Merge simulation tree. This approach was introduced in [7, 11]. The sample cost function is based on *common random numbers*, in the sense that the same set of component lifetimes is used for all values of θ.

Each node of the simulation tree corresponds to a simulation valid for a particular subinterval of $[a,b)$. As we shall explain in more detail in a moment, at each *level* of the tree, this interval is partitioned into a number of subintervals and the nodes at that level are in one-to-one correspondence with the subintervals of the partition. As the level increases, there are more and more subintervals, so the number of nodes per level increases with the level. For $n \ge 0$, let X_n be the number of nodes at level n, and label these nodes (n,i), $i = 1,\ldots,X_n$. Node (n,i) corresponds to some interval $(a_n(i-1), a_n(i)]$, where $a_n(0) = a$ and $a_n(X_n) = b$.

At time 0, we generate M component lifetimes and determine the time T_1 of the first intervention, which is the minimum between $\bar{\theta}$ and the shortest component lifetime. Note that up to time T_1^-, the evolution is independent of θ and a single simulation is thus valid for all $\theta \in [a,b)$. This is level 0 of the simulation tree. At this level, we have a single node ($X_0 = 1$), labeled $(0,1)$. When the first intervention is about to take place, at least one component is replaced and all the remaining components have lived $\tau_0 = T_1$ units of time. If $\tau_0 \ge b$, then we replace all components and the process starts afresh. If $\tau_0 \le a$, then this is a forced intervention and we replace only the failed component. In these two situations, the system's evolution is again independent of $\theta \in [a,b)$ until the next intervention time T_2, so there is still a single node at level 1 of the tree, i.e., $X_1 = 1$. But if $\tau_0 \in (a,b)$, then this interval splits in two subintervals $(a,\tau_0]$ and $(\tau_0,b]$, giving rise to nodes $(1,1)$ and $(1,2)$ at level 1. For $\theta \in (a,\tau_0]$, all components are replaced at time T_1, while for $\theta \in (\tau_0,b]$, only the failed component is replaced. In this case we have $X_1 = 2$ and the trajectory of the Markov renewal process differs, at these two nodes, from time T_1 onwards. From that point, we thus have to maintain two "parallel" simulations. For node $(1,1)$, where all components are replaced, we generate new lifetimes, determine the time $T_1 + \tau_1(1)$ of the next intervention, where $\tau_1(1)$ is the minimum between $\bar{\theta}$ and the smallest lifetime, and proceed as we did above, with $(a,\tau_0]$ instead of $[a,b)$. For node $(1,2)$, the time $\tau_1(2)$ until the next intervention is the minimum between the lifetime of the new component, the residual lifetimes of the other components (of age τ_0), and $\bar{\theta} - \tau_0$. Each of these subintervals (or nodes) may then split again, at level 2, if the age of one or more component(s) immediately before the intervention lies in the subinterval.

In general, node (n,i) at level n corresponds to the nth intervention time, for θ in the subinterval $(a_n(i-1), a_n(i)]$. Let $T_n(i)$ denote the nth intervention time for this interval, $Z_n(i) = (Z_{n,1}(i),\ldots,Z_{n,M}(i))$ the age vector immediately after the intervention, and $\tau_n(i)$ the time until the next intervention. In our implementation, new lifetimes are generated at level n for each component that is replaced at one (at least) of the level-n nodes. If the same component is replaced for several nodes at level n, then the same lifetime is used across all these nodes for this component. Whenever a new node (n, i) is created, we use the new lifetimes generated for that level for the components that are replaced and compute the value of $\tau_n(i)$ for this node. If $\zeta_n(i)$ is the minimum among the new component lifetimes and the residual lifetimes of the non-replaced (old) components, then $\tau_n(i) = \min\{\zeta_n(i), \bar{\theta} - Z_{n,1}(i), \ldots, \bar{\theta} - Z_{n,M}(i)\}$. The age of component m at the next intervention, just before the replacements, will be $\tilde{Z}_{n,m}(i) = \infty$ (say) if the next intervention is triggered by a failure of this component, and $\tilde{Z}_{n,m}(i) = Z_{n,m}(i) + \tau_n(i)$ otherwise. The number of splits of the subinterval $(a_n(i-1), a_n(i)]$ at level $n+1$ will be equal to the number of *different* values of $\tilde{Z}_{n,m}(i)$ lying in the open interval $(a_n(i-1), a_n(i))$, and these values will determine the partition of the subinterval. The number of components replaced at intervention $n+1$ will be the same for all θ, within each of these new subintervals determined par the partition, but will differ between the subintervals (so the process trajectory will start to differ, from that point on, between these subintervals). For the components whose value of $\tilde{Z}_{n,m}(i)$ is outside this interval, the (replacement) decision at the next intervention time is the same for all values of θ in the interval.

When the current time $T_n(i)$ has reached the simulation horizon T for each leaf (node without current successor) of the tree, the function $C_{N(T,\cdot)}(\cdot)$ can be computed. This function is piecewise constant (constant over each of the subintervals of the partition of $[a, b)$ determined by the leaves of the tree).

Since the number of splittings of node (n, i) can be (in general) anywhere in the range 0 to $M-1$, the size of the simulation tree may, in the worst case, increase exponentially fast as a function of n. Fortunately, it turns out that $E[X_n]$ increases only *linearly* in n (see next section). As a consequence, the expected computational effort to obtain the functional estimator $\hat{C}(\theta)$ over the interval $[a, b)$ is $\mathcal{O}(RT^2)$.

3 Algorithmic Analysis via a Branching Process

We discuss in this section the growth rate (in expectation) of the process $\{X_n, n \geq 0\}$. The following proposition states that $E[X_n]$ grows only linearly with n. We give a proof only for the case where the interval $[a, b)$ satisfies $b \leq \bar{\theta}/2$. Without this assumption, the linear growth result still holds, but the proof is more involved and will be given elsewhere.

Proposition 1 *Under the assumptions of our model, if $b \leq \bar{\theta}/2$, one has*

$$E[X_n] \leq 1 + (b-a)(M-1)M\bar{\lambda}n$$

for all $n \geq 0$.

Proof: Let $\xi_n(i)$ be the number of splits of the subinterval $(a_n(i-1), a_n(i)]$ at level $n+1$. Observe that

$$X_{n+1} = X_n + \sum_{i=1}^{X_n} \xi_n(i).$$

Thus, it suffices to show that

$$\begin{aligned} & E[\xi_n(i) \mid X_n, a_n(i-1), a_n(i), Z_n(i)] \\ \leq\ & (M-1)M(a_n(i) - a_n(i-1))\bar{\lambda} \end{aligned} \quad (1)$$

for all (n, i), because this would imply

$$\begin{aligned} & E[X_{n+1}] = E[X_n] \\ & \quad + E\left[\sum_{i=1}^{X_n} E[\xi_n(i) \mid X_n, a_n(i-1), a_n(i), Z_n(i)]\right] \\ & \quad \leq E[X_n] + E\left[\sum_{i=1}^{X_n} (M-1)M(a_n(i) - a_n(i-1))\bar{\lambda}\right] \\ & \quad = E[X_n] + (b-a)(M-1)M\bar{\lambda}, \end{aligned}$$

which, together with the fact that $X_0 = 1$, implies the result. To prove (1), let us denote $Z_n(i) = (z_1, \ldots, z_M)$ and $(a_n(i-1), a_n(i)) = (a_0, a_1)$ to simplify the notation, and assume, without loss of generality, that $z_1 \geq z_m$ for all $m > 1$. Note that the lifetime density of the system is the product of the system's failure rate and reliability function. That is, at node (n, i), the residual lifetime ζ of the system has conditional density

$$f(\zeta \mid Z_n(i)) = \left(\prod_{m=1}^{M} \bar{G}(z_m + \zeta \mid z_m)\right) \sum_{m=1}^{M} \lambda(z_m + \zeta),$$

for $\zeta > 0$, where $\bar{G}(z_m + \zeta \mid z_m) = \bar{G}(z_m + \zeta)/\bar{G}(z_m)$ is the conditional reliability function of component m. The expectation of $\xi_n(i)$ is bounded above by the expected number of components m for which $z_m + \zeta$ belongs to the interval (a_0, a_1). This gives the following inequalities, explained below.

$$\begin{aligned} & E[\xi_n(i) \mid a_0, a_1, z_1, \ldots, z_M) \\ \leq\ & \int_0^{\bar{\theta} - z_1} \prod_{m=1}^{M} \bar{G}(z_m + \zeta \mid z_m) \sum_{j=1}^{M} \lambda(z_j + \zeta) \cdot \\ & \qquad \sum_{m=1,\, m \neq j}^{M} I[a_0 < z_m + \zeta < a_1]\, d\zeta \qquad (2) \\ & + \prod_{m=1}^{M} \bar{G}(z_m + \bar{\theta} - z_1 \mid z_m) \times \qquad (3) \end{aligned}$$

$$\sum_{m=2}^{M} I[a_0 < z_m + \bar{\theta} - z_1 < a_1] \quad (4)$$

$$\leq (M-1)M\bar{\lambda}(a_1 - a_0)$$

$$+ \sum_{m=2}^{M} I[a_0 < z_m - z_1 + \bar{\theta} < a_1], \quad (5)$$

where I denotes the indicator function. The integral above is with respect to the density of a failure before the next planned intervention, and the other term corresponds to the event of a preventive intervention because component 1 has reached age $\bar{\theta}$. The sum in (2) is the number of components m whose age $z_m + \zeta$ at the next intervention lies in (a_0, a_1), if the next intervention is triggered by a failure of component j in ζ units of time. The sum in (4) is the number of components whose age at the next intervention lies in (a_0, a_1), if the next intervention is preventive, due to component 1 reaching its age threshold $\bar{\theta}$. In the second inequality, we use the facts that the reliability function never exceeds 1 and the that each indicator function in (2) is nonzero only in an interval of values of ζ of length at most $a_1 - a_0$. If $b \leq \bar{\theta}/2$, then $a_1 \leq \bar{\theta}/2$ and $z_1 \leq b < \bar{\theta}/2$, so the sum in (5) vanishes and this completes the proof. ■

To prove the result for the more general case, we would need a proper bound on the sum (5). Our numerical experiments agree with the idea that the linear bound holds in general.

4 Simulation Results

We now give some numerical results illustrating the growth rate of the simulation tree as a function of n and M. These results are from [8], where several more numerical examples can be found. For all our experiments, the components lifetime distribution was the same Weibull distribution as in [3], namely

$$G(t) = 1 - e^{-x^2/100},$$

with linearly increasing failure rate $\lambda(t) = 0.02t$, $t > 0$. We took $\bar{\theta} = 13$.

Figure 1 shows the results of the functional estimation for an example taken from [3]. Here, $M = 3$, $c_1 = c_2 = 1, c_3 = 2$, $T = 200$, and $R = 500$. In the figure, the solid line represents the value of $\hat{C}(\theta)$, over the interval $[a, b) = [0, \bar{\theta}) = [0, 13)$. The dotted lines are located two (empirical) standard deviations above and below the empirical mean and indicate a (heuristic) "confidence band" for our estimator.

In order to assess the empirical behavior of the growth of the tree, we performed several experiments. The first set of experiments focuses on the

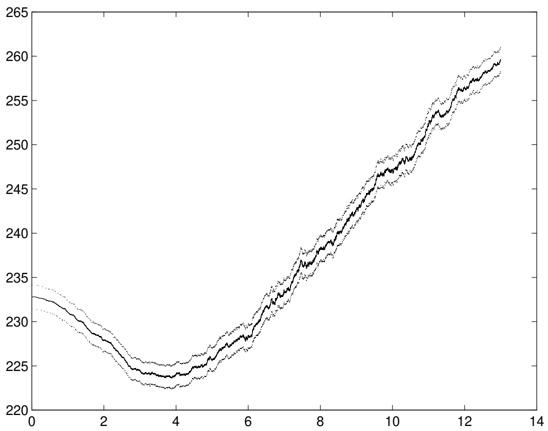

Figure 1: Estimation of the total expected cost.

behavior of the growth rate of $E[X_{n+1} - X_n]$ for a fixed number M of components. In Proposition 1 we proved that this expected difference is uniformly bounded over the left part of the interval: $\theta \in (0, \bar{\theta}/2)$. Figure 2 shows the simulation results for the consecutive values of $X_{n+1} - X_n$, as a function of n, performed for three different numbers of components: $M = 5$ (lower), $M = 10$ (middle) and $M = 20$ (upper). Each of these plots was obtained by running $R = 200$ independent runs (simulation trees) over the whole interval $\theta \in (0, \bar{\theta})$. The results suggest that the growth is also linear in the right half of the interval.

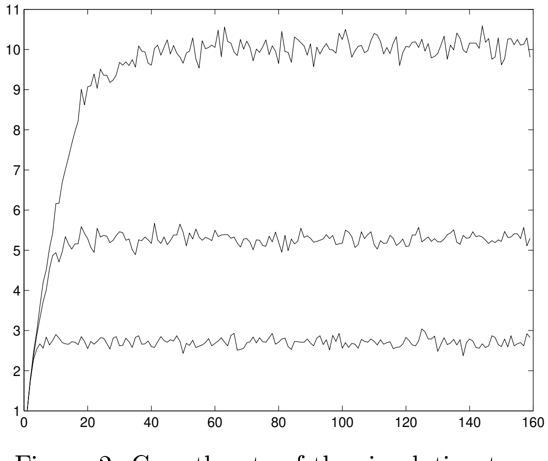

Figure 2: Growth rate of the simulation tree.

Finally, we focus on the growth rate as a function of the number of components. In Proposition 1, the upper bound is quadratic in M for $\theta \in (0, \bar{\theta}/2)$. We performed $R = 200$ simulation runs for each of 10 different values of M over the interval $(0, \bar{\theta})$. The growth rate was estimated in each case and plotted in Figure 3 as a function of M.

This figure (and other experiments we did) suggest that the growth rate is linear instead of

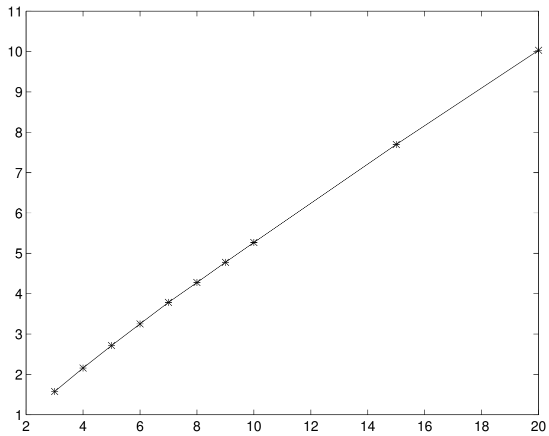

Figure 3: Growth rate vs the number of components.

quadratic in the number of components. Our estimates can perhaps be tightened to agree with the experimental results; this is a topic for further investigation.

5 Conclusion

We have studied the computational efficiency of the Split-and-Merge simulation tree for functional estimation for an age replacement problem. It is well known that this is a difficult problem and no analytical solution is available for $M > 2$. Several methods exist for finding the optimal replacement strategies, but they lack in efficiency as the number of components increases. The Split-and-Merge tree is a method that allows the estimation of functions of a threshold control over a whole interval of interest. While this method may yield a viable way to estimating the optimal threshold control, it could yield a geometric growth rate in the computational effort as the time horizon of the simulation increases. We have presented a theoretical proof that the expected growth rate of the tree is bounded linearly in the number of generations and quadratically in the number of components, for a subset of the control space. We have also presented experimental evidence that this bound is valid over the whole control space and that it is not tight as a function of the number of components. The growth of the tree seems to be linear both in the number of interventions as well as the number of components.

Acknowledgements

This work has been supported by NSERC-Canada grants # WFA-0139015 and # OGP-0110050, as well as FCAR-Québec grant # 94-ER-1654.

References

[1] M. C. Fu, J.-Q. Hu, and L. Shi, An Application of Perturbation Analysis to a Replacement Problem in Maintenance Theory, *Proceedings of the 1993 Winter Simulation Conference*, IEEE Press: 329–337, 1993.

[2] I. B. Gertsbakh, *Models of Preventive Maintenance*, North Holland, 1977.

[3] A. Haurie and P. L'Ecuyer, "Approximation and Bounds in Discrete Event Dynamic Programming", *IEEE Transactions on Automatic Control*, **AC-31**, No. 3: 227–235, 1986.

[4] P. L'Ecuyer, "Processus de décision markoviens à étapes discrètes : application à des problèmes de remplacement d'équipement", Ph.D. thesis, published in *Les cahiers du GERAD*, no. G-83-06, Ecole des H.E.C., Montreal, 1983.

[5] P. L'Ecuyer, "A Unified View of the IPA, SF, and LR Gradient Estimation Techniques", *Management Science*, **36**, 11:1364–1383, 1990.

[6] P. L'Ecuyer, "Two Approaches for Estimating the Gradient in Functional Form", *Proceedings of the 1993 Winter Simulation Conference*, IEEE Press: 207–346, 1993.

[7] P. L'Ecuyer and F. J. Vázquez-Abad, "Functional Estimation with Respect to a Threshold Parameter via Dynamic Split-and-Merge", submitted to *Journal of Discrete Event Dynamic Systems*, 1995.

[8] B. Martin, "Estimation fonctionnelle dans le cadre de politiques de remplacement pour un système à plusieurs composantes", M. Sc. Thesis, Département d'I.R.O., Université de Montréal, 1996.

[9] W. P. Pierskalla and A. Voelker, "A Survey of Maintenance Models: The Controls and Surveillance of Deteriorating Systems", *Naval Research Logistic Quarterly*, **23**, 3: 351–388, 1976.

[10] R. Y. Rubinstein and A. Shapiro, *Discrete Event Systems: Sensitivity Analysis and Stochastic Optimization by the Score Function Method*, Wiley, 1993.

[11] F.J. Vázquez-Abad and P. L'Ecuyer, "Simulation Trees for Functional Estimation via the Phantom Method", *Proceedings of the 11th International Conference on Analysis and Optimization of Systems: DES* in *Lecture Notes in Control and Information Systems*, Springer Verlag, **199**:449–455, 1994.