REVUE FRANÇAISE D'AUTOMATIQUE, D'INFORMATIQUE ET DE RECHERCHE OPÉRATIONNELLE. RECHERCHE OPÉRATIONNELLE

T. SATOW

K. YASUI

T. NAKAGAWA

Optimal garbage collection policies for a database in a computer system

Revue française d'automatique, d'informatique et de recherche opérationnelle. Recherche opérationnelle, tome 30, nº 4 (1996), p. 359-372.

<http://www.numdam.org/item?id=RO_1996__30_4_359_0>

Recherche opérationnelle/Operations Research
(vol. 30, n° 4, 1996, pp. 359-372)

OPTIMAL GARBAGE COLLECTION POLICIES FOR A DATABASE IN A COMPUTER SYSTEM (*)

by T. Satow ([1]), K. Yasui ([1]) and T. Nakagawa ([1])

Communicated by Naoto Kaio

Abstract. – *It has recently become necessary to maintain a database periodically and economically for the requirements of continuous operation of a computer system. To use memory areas effectively and to improve the processing efficiency, garbage collections for a database are made at suitable times according to the number of updates and the amount of garbages. This paper considers that additive garbages arise according to Cdf G (x) when a database is updated, and that a database is useless if total garbages exceed a threshold level K. To prevent this, we make a garbage collection at periodic time T or at N-th update, whichever occurs first. Using the theory of cumulative processes, the expected cost is obtained, and the optimal T* and N* which minimize it are discussed. Finally, numerical examples are given when G (x) is exponential.*

Keywords: Garbage collection, Database, Expected cost, Optimal policy.

Résumé. – *Il est apparu récemment nécessaire de mettre à jour périodiquement une base de données de façon économe. Pour utiliser efficacement les zones de mémoires et pour améliorer l'efficacité du processus, des ramassages de déchets d'une base de données sont effectuées à des instants appropriés selon le nombre de mises à jour et la quantité de déchets. On considère dans cet article que les déchets s'accumulent selon une loi de probabilité cumulée G (x) lorsque la base de donnée est mise à jour, et que celle-ci est inutilisable si les déchets accumulés dépassent un seuil K. Pour prévenir cela, nous effectuons un ramassage de déchets soit avec une période de temps T, soit à la N-ième mise à jour, en choisissant la méthode qui se présente en premier. Utilisant la théorie des processus cumulatifs, nous obtenons le coût moyen, et nous examinons T* et N* qui le minimise. Nous terminons avec un exemple numérique où G (x) représente la loi exponentielle.*

Mots clés : Ramassage des déchets, base données, coût moyen, politique optimale.

1. INTRODUCTION

A database is in optimal storage according to the schema defined in the data structures. However, after some operations, storage areas are not in good order due to additions and deletions of data. Such updating procedures reduce the size of continuous memory areas and make processing efficiency worse. To use storage areas effectively and to improve processing efficiently, garbage

(*) Received June 1994.

([1]) Department of Industrial Engineering, Aichi Institute of Technology, Toyota 471-03, Japan.

Recherche opérationnelle/Operations Research, 0399-0559/96/04/$ 7.00

collections have to be made at suitable times. Many garbage collections to reclaim the storage and rearrange a database are used in most large list processing systems [1], [2]. Cohen [3] reviewed algorithms for performing garbage collection of linked data structures. Recently, several authors [4], [5], [6] have studied "real" time garbage collections to avoid suspension of the application program in its execution. Almost all problems have been concerning with how to introduce garbage collection methods.

When a database is updated from several online terminals, it would be necessary to set up a desired response time. If response times become comparatively long, processing efficiency becomes worse, and at last, it would be impossible to update data. Response times may depend on the amount of garbages.

This paper proposes when to make garbage collection for a database with a threshold level K of total garbages. An amount of garbages with Cdf $G(x)$ arises from each update and is additive. A cost and a time for a garbage collection are higher if total garbages are greater than K. To prevent such an event, a garbage collection is made at periodic time T or at N-th update, whichever occurs first.

Each garbage collection restores computer resources such as response time, storage area and throughput to an initial state. This corresponds to one modification of replacements of shock models [7], replacing "update" by "shock" and "garbage" by "damage". Using the theory of cumulative processes [8], the expected cost is derived and optimal policies which minimize it are discussed. It is shown that optimal time T^* and number N^* exist uniquely in reasonable cases when the system is updated at a Poisson process. Numerical examples are given when an amount of damages is exponential.

2. EXPECTED COST

Suppose that the database is updated at a nonhomogeneous Poisson process with an intensity function $\lambda(t)$ and a mean-value function $R(t)$, *i.e.*, $R(t) \equiv \int_0^t \lambda(u)\,du$. Then, the probability of j updates of the database during $(0, t]$ is

$$H_j(t) \equiv \{[R(t)]^j / j!\}\, e^{-R(t)} \qquad (j = 0, 1, 2, \ldots).$$

Further, an amount W_j of garbages arises from the j-th update and has a probability distribution $G(x)$, independent of the number of updates. It is assumed that these garbages are additive. Then, total garbages $Z_j \equiv \sum_{i=1}^{j} W_i$ up to the j-th update have

$$\Pr\{Z_j \leq x\} = G^{(j)}(x) \quad (j = 0, 1, 2, ...),$$

where $G^{(j)}(x)$ is the j-fold convolution of $G(x)$ with itself, and $G^{(0)}(x) \equiv 0$ for $x < 0$, $\equiv 1$ for $x \geq 0$. If total garbages exceed an upper limit level K, then the database becomes useless for lack of storage areas or due to long response times.

A garbage collection is made, before the database is useless, at time T or at N-th update, whichever occurs first. For the above model, we introduce the following costs: let c_1 and c_3 be fixed costs for the respective garbage collections at time T and at N-th update, and c_2 be a fixed cost for garbage collection when total garbages exceed a level K, where $c_1 < c_2$ and $c_3 < c_2$. Further, let $c_0(x)$ be a variable cost for collections of an amount x of garbages.

The expected cost when a garbage collection is made at time T or N-th update is

$$\sum_{j=0}^{N-1} H_j(T) \int_0^K [c_1 + c_0(x)]\, dG^{(j)}(x) + \int_0^T H_{N-1}(t)\, \lambda(t)\, dt \int_0^K [c_3 + c_0(x)]\, dG^{(N)}(x), \qquad (1)$$

and the expected cost when total garbages exceed a level K is

$$[c_2 + c_0(K)] \sum_{j=0}^{N-1} [G^{(j)}(K) - G^{(j+1)}(K)] \int_0^T H_j(t)\, \lambda(t)\, dt. \qquad (2)$$

The mean time to a garbage collection is

$$T \sum_{j=0}^{N-1} H_j(T)\, G^{(j)}(K) + G^{(N)}(K) \int_0^T t\, H_{N-1}(t)\, \lambda(t)\, dt + \sum_{j=0}^{N-1} [G^{(j)}(K) - G^{(j+1)}(K)] \int_0^T t\, H_j(t)\, \lambda(t)\, dt = \sum_{j=0}^{N-1} G^{(j)}(K) \int_0^T H_j(t)\, dt. \qquad (3)$$

Therefore, the expected cost rate is

$$C(T, N) = \left\{ \sum_{j=0}^{N-1} H_j(T) \int_0^K [c_1 + c_0(x)]\, dG^{(j)}(x) + \int_0^T H_{N-1}(t)\,\lambda(t)\,dt \int_0^K [c_3 + c_0(x)]\, dG^{(N)}(x) + [c_2 + c_0(K)] \sum_{j=0}^{N-1} [G^{(j)}(K) - G^{(j+1)}(K)] \int_0^T H_j(t)\,\lambda(t)\,dt \right\} \Big/ \sum_{j=0}^{N-1} G^{(j)}(K) \int_0^T H_j(t)\,dt. \quad (4)$$

3. OPTIMAL GARBAGE COLLECTION TIME

Suppose that a garbage collection is made at only time T. Then, from (4), the expected cost is given by

$$C_1(T) \equiv \lim_{N \to \infty} C(T, N) = \left\{ \sum_{j=0}^{\infty} H_j(T) \int_0^K [c_1 + c_0(x)]\, dG^{(j)}(x) + [c_2 + c_0(K)] \sum_{j=0}^{\infty} [G^{(j)}(K) - G^{(j+1)}(K)] \int_0^T H_j(t)\,\lambda(t)\,dt \right\} \Big/ \sum_{j=0}^{\infty} G^{(j)}(K) \int_0^T H_j(t)\,dt. \quad (5)$$

We seek an optimal time T^* which minimizes $C_1(T)$ in (5). Differentiating $C_1(T)$ with respect to T and setting it equal to zero imply

$$(c_2 - c_1) \left\{ \lambda(T) Q_1(T) \sum_{j=0}^{\infty} G^{(j)}(K) \int_0^T H_j(t)\,dt - \sum_{j=0}^{\infty} H_j(T)[1 - G^{(j)}(K)] \right\} + \left\{ \frac{\lambda(T) \sum_{j=0}^{\infty} H_j(T) \int_0^K [G^{(j)}(x) - G^{(j+1)}(x)]\, dc_0(x)}{\sum_{j=0}^{\infty} H_j(T)\, G^{(j)}(K)} \right\}$$

$$\times \sum_{j=0}^{\infty} G^{(j)}(K) \int_0^T H_j(t)dt - \sum_{j=0}^{\infty} H_j(T) \int_0^K [1 - G^{(j)}(x)] dc_0(x) = c_1, \quad (6)$$

where

$$Q_1(T) \equiv \frac{\sum_{j=0}^{\infty} H_j(T) \left[G^{(j)}(K) - G^{(j+1)}(K)\right]}{\sum_{j=0}^{\infty} H_j(T) \, G^{(j)}(K)}.$$

It is very difficult to discuss an optimal T^* analytically. In particular, we assume that the database is updated at a Poisson process with rate λ and $c_0(x)$ is proportional to an amount of garbages, *i.e.*, $\lambda(t) = \lambda$ and $c_0(x) = c_0 \, x$. Then, equations (5) and (6) are rewritten as, respectively,

$$C_1(T) = \left\{c_2 - (c_2 - c_1) \sum_{j=0}^{\infty} H_j(T) \, G^{(j)}(K) + c_0 \sum_{j=0}^{\infty} H_j(T) \int_0^K [1 - G^{(j)}(x)] \, dx \right\} \Big/ \sum_{j=0}^{\infty} G^{(j)}(K) \int_0^T H_j(t) \, dt, \quad (7)$$

$$(c_2 - c_1) \left\{\lambda Q_1(T) \sum_{j=0}^{\infty} G^{(j)}(K) \int_0^T H_j(t) \, dt - \sum_{j=0}^{\infty} H_j(T) [1 - G^{(j)}(K)] \right\} + c_0 \left\{\lambda Q_2(T) \sum_{j=0}^{\infty} G^{(j)}(K) \int_0^T H_j(t) \, dt - \sum_{j=0}^{\infty} H_j(T) \int_0^K [1 - G^{(j)}(x)] \, dx \right\} = c_1, \quad (8)$$

where

$$Q_2(T) \equiv \frac{\sum_{j=0}^{\infty} H_j(T) \int_0^K [G^{(j)}(x) - G^{(j+1)}(x)]\, dx}{\sum_{j=0}^{\infty} H_j(T)\, G^{(j)}(K)},$$

$$H_j(t) \equiv \frac{(\lambda t)^j}{j!} e^{-\lambda t} \quad (j = 0, 1, 2, \ldots).$$

If $c_0 = 0$ then (8) is

$$\lambda Q_1(T) \sum_{j=0}^{\infty} G^{(j)}(K) \int_0^T H_j(t)\, dt - \sum_{j=0}^{\infty} H_j(T)[1 - G^{(j)}(K)] = \frac{c_1}{c_2 - c_1}. \tag{9}$$

If $Q_1(T)$ is strictly increasing, then the left-hand side in (9) is also strictly increasing from 0 to $Q_1(\infty)[1+M(K)]-1$, where $Q_1(\infty) \equiv \lim_{T\to\infty} Q_1(T)$ and $M(K) \equiv \sum_{j=1}^{\infty} G^{(j)}(K)$ which represents the mean number of updates until total garbages exceed a level K. Thus, if

$$Q_1(\infty)[1 + M(K)] > c_2/(c_2 - c_1)$$

then there exists a finite and unique $\tilde{T}$ which satisfies (9).

Further, if $Q_2(T)$ is strictly decreasing, then the second bracket of the left-hand side in (8) is strictly decreasing from 0, and hence, $T^* > \tilde{T}$.

4. OPTIMAL UPDATE NUMBER

The expected cost rate when a garbage collection is made at only N-th update is, from (4),

$$C_2(N) \equiv \lim_{T \to \infty} C(T, N)$$
$$= \frac{[c_2 + c_0(K)][1 - G^{(N)}(K)] + \int_0^K [c_3 + c_0(x)]\, dG^{(N)}(x)}{\sum_{j=0}^{N-1} G^{(j)}(K) \int_0^\infty H_j(t)\, dt}. \tag{10}$$

Forming the inequality $C_2(N+1) - C_2(N) \geq 0$ to seek an optimal number N^* which minimizes $C_2(N)$ in (10), we have

$$(c_2 - c_3)\left\{\frac{G^{(N)}(K) - G^{(N+1)}(K)}{G^{(N)}(K)\int_0^\infty H_N(t)\, dt} \sum_{j=0}^{N-1} G^{(j)}(K) \times \int_0^\infty H_j(t)\, dt - [1 - G^{(N)}(K)]\right\}$$
$$+ \frac{\int_0^K [G^{(N)}(x) - G^{(N+1)}(x)]\, dc_0(x)}{G^{(N)}(K)\int_0^\infty H_N(t)\, dt}$$
$$\times \sum_{j=0}^{N-1} G^{(j)}(K)\int_0^\infty H_j(t)\, dt - \int_0^K [1 - G^{(N)}(x)]\, dc_0(x) \geq c_3. \tag{11}$$

Suppose that $\lambda(t) = \lambda$ and $c_0(x) = c_0\, x$. Then, equations (10) and (11) are, respectively,

$$C_2(N) = \frac{c_2 - (c_2 - c_3)\, G^{(N)}(K) + c_0 \int_0^K [1 - G^{(N)}(x)]\, dx}{(1/\lambda) \sum_{j=0}^{N-1} G^{(j)}(K)}, \tag{12}$$

$$(c_2 - c_3)\left\{\frac{G^{(N)}(K) - G^{(N+1)}(K)}{G^{(N)}(K)} \sum_{j=0}^{N-1} G^{(j)}(K) - [1 - G^{(N)}(K)]\right\}$$

$$+ c_0 \left\{ \frac{\int_0^K [G^{(N)}(x) - G^{(N+1)}(x)]\,dx}{G^{(N)}(K)} \times \sum_{j=0}^{N-1} G^{(j)}(K) - \int_0^K [1 - G^{(N)}(x)]\,dx \right\} \geq c_3. \tag{13}$$

If $c_0 = 0$ then (13) is

$$\frac{G^{(N)}(K) - G^{(N+1)}(K)}{G^{(N)}(K)} \sum_{j=0}^{N-1} G^{(j)}(K) - [1 - G^{(N)}(K)] \geq \frac{c_3}{c_2 - c_3}. \tag{14}$$

Denote the left-hand side in (14) by $U(N)$. Then, we have

$$U(N) - U(N-1) = \left[\frac{G^{(N)}(K)}{G^{(N-1)}(K)} - \frac{G^{(N+1)}(K)}{G^{(N)}(K)}\right] \sum_{j=0}^{N-1} G^{(j)}(K).$$

Thus, if $G^{(j+1)}(x)/G^{(j)}(x)$ is strictly decreasing in j, then $U(N)$ is also strictly increasing, and

$$U(\infty) \equiv \lim_{N \to \infty} U(N) = Q_3(\infty)[1 + M(K)] - 1.$$

where

$$Q_3(\infty) \equiv \lim_{N \to \infty} \frac{G^{(N)}(K) - G^{(N+1)}(K)}{G^{(N)}(K)}.$$

Therefore, if $Q_3(\infty)[1 + M(K)] > c_2/(c_2 - c_3)$ then there exists a unique minimum $\tilde{N}$ which satisfies (14).

Further, if $\left\{\int_0^K [G^{(N)}(x) - G^{(N+1)}(x)]\,dx\right\}/G^{(N)}(K)$ is strictly decreasing in N, then the second bracket of the left-hand side in (13) is strictly decreasing from 0 when $N = 0$, and hence $N^* \geq \tilde{N}$.

5. NUMERICAL EXAMPLE

We compute the optimal policies numerically when $c_0(x) = c_0 x$, $\lambda(t) = \lambda$ and $G(x) = 1 - e^{-\mu x}$. In this case, equation (8) is

$$(c_2 - c_1)\left\{\lambda Q_1(T) \sum_{j=0}^{\infty} G^{(j)}(K) \int_0^T H_j(t)\,dt - \sum_{j=0}^{\infty} H_j(T)\,[1 - G^{(j)}(K)]\right\} + \frac{c_0}{\mu}\left\{\lambda\,[1 - Q_1(T)] \sum_{j=0}^{\infty} G^{(j)}(K) \int_0^T H_j(t)\,dt - \sum_{j=1}^{\infty} H_j(T) \sum_{i=1}^{j} G^{(i)}(K)\right\} = c_1, \tag{15}$$

where $G^{(j)}(K) \equiv \sum_{i=j}^{\infty} [(\mu K)^i / i!]\, e^{-\mu K}$, $H_j(t) \equiv [(\lambda t)^j / j!]\, e^{-\lambda t}$,

$$Q_1(T) \equiv 1 - \frac{\sum_{j=0}^{\infty} H_j(T)\, G^{(j+1)}(K)}{\sum_{j=0}^{\infty} H_j(T)\, G^{(j)}(K)}.$$

Denote the left-hand side in (15) by $L_1(T)$. Then, it is evident that

$$L_1'(T) = \lambda\left(c_2 - c_1 - \frac{c_0}{\mu}\right) Q_1'(T) \sum_{j=0}^{\infty} G^{(j)}(K) \int_0^T H_j(t)\,dt,$$

$$L_1(0) = 0,$$

$$L_1(\infty) \equiv \lim_{T \to \infty} L_1(T) = \mu K \left(c_2 - c_1 - \frac{c_0}{\mu}\right).$$

Note that $G^{(j+1)}(x)/G^{(j)}(x)$ is strictly decreasing in j when $G(x) = 1 - e^{-\mu x}$. Thus, from Appendix, $Q_1(T)$ is strictly increasing, and hence, $L_1(T)$ is also strictly increasing for $c_2 - c_1 - c_0/\mu > 0$. Therefore, if

$c_2 - c_1 - c_0/\mu > c_1/(\mu K)$ then there exists a finite and unique T^* which satisfies (15), and the resulting expected cost is

$$C_1(T^*) = \lambda\left[\frac{c_0}{\mu} + (c_2 - c_1)\, Q_1(T^*)\right]. \tag{16}$$

Conversely, if $c_2 - c_1 - c_0/\mu \le c_1/(\mu K)$ then $T^* \to \infty$, *i.e.*, a garbage collection should not be made before total garbages exceed *K*, and

$$C_1(\infty) = \frac{c_2 + c_0 K}{(\mu K + 1)/\lambda}. \tag{17}$$

Next, we compute an optimal number N^* which minimizes $C_2(N)$ in (10). In this case, equation (13) is

$$(c_2 - c_3)\left\{\left[1 - \frac{G^{(N+1)}(K)}{G^{(N)}(K)}\right]\sum_{j=0}^{N-1} G^{(j)}(K) - [1 - G^{(N)}(K)]\right\} + \frac{c_0}{\mu}\left\{\frac{G^{(N+1)}(K)}{G^{(N)}(K)}\sum_{j=0}^{N-1} G^{(j)}(K) - \sum_{j=1}^{N} G^{(j)}(K)\right\} \ge c_3. \tag{18}$$

Denote the left-hand side in (18) by $L_2(N)$. Then,

$$L_2(N) - L_2(N-1) = \left(c_2 - c_3 - \frac{c_0}{\mu}\right)\left\{\frac{G^{(N)}(K)}{G^{(N-1)}(K)} - \frac{G^{(N+1)}(K)}{G^{(N)}(K)}\right\}\sum_{j=0}^{N-1} G^{(j)}(K), \tag{19}$$

$$L_2(0) = 0,$$
$$L_2(\infty) \equiv \lim_{N\to\infty} L_2(N) = \mu K\left(c_2 - c_3 - \frac{c_0}{\mu}\right).$$

Therefore, if $c_2 - c_3 - c_0/\mu > c_3/(\mu K)$ then there exists a unique minimum N^* which satisfies (18), and conversely, if $c_2 - c_3 - c_0/\mu \le c_3/(\mu K)$ then $N^* \to \infty$, since $G^{(j+1)}(x)/G^{(j)}(x)$ is strictly decreasing in j.

Table I gives the optimal times T^* for $\mu K = 150, 300, 500, 700$ and $c_2/c_1 = 100, 200, 500, 1000$ when $c_0 K/c_1 = 1$. For example, when $\lambda = 5$, $c_2/c_1 = 100$ and $\mu K = 700$, the optimal time λT^* is about 572. That is, when the database is updated at 5 times an hour and becomes useless after 700 updates on the average, a garbage collection should be made at $572/5 = 114.4$ hours, *i.e.*, at about $114.4/24 \approx 4.8$ days. Taking another view point, when total garbages exceed $(572/700) \times 100 \approx 81.7\%$ of the upper limit *K*, a garbage collection should be made.

TABLE I
Optimal times λT^* and $C_1(T^*)/\lambda$ when $c_0 K/c_1 = 1$

c_2/c_1	μK							
	150		300		500		700	
	λT^*	$C_1(T^*)/\lambda$	λT^*	$C_1(T^*)/\lambda$	λT^*	$C_1(T^*)/\lambda$	λT^*	$C_1(T^*)/\lambda$
100	98.1	0.017152	221.5	0.007904	394.5	0.004576	572.1	0.003191
200	95.3	0.017340	217.5	0.008026	389.2	0.004614	565.8	0.003213
500	92.0	0.017903	212.5	0.008115	382.6	0.004643	558.0	0.003244
1000	89.6	0.018084	209.0	0.008223	377.9	0.004663	552.4	0.003259

Similarly, Table II gives the optimal numbers N^* for $\mu K = 150, 300, 500, 700$ and $c_2/c_3 = 100, 200, 500, 1000$ when $\mu = 1.0$ and $c_0 K/c_3 = 1$. For example, when $c_2/c_3 = 100$ and $\mu K = 700$, the optimal number N^* is 605. That is, a garbage collection is made at $(605/700) \times 100 \approx 86.4\%$ of the upper limit *K*, the value of which is greater than that of the previous case when $c_1 = c_3$.

TABLE II
Optimal number N^* and $C_2(N^*)/\lambda$ when $\mu = 1.0$ and $c_0 K/c_3 = 1$

c_2/c_3	μK							
	150		300		500		700	
	N^*	$C_2(N^*)/\lambda$	N^*	$C_2(N^*)/\lambda$	N^*	$C_2(N^*)/\lambda$	N^*	$C_2(N^*)/\lambda$
100	110	0.016000	241	0.007562	421	0.004406	605	0.003100
200	108	0.016175	238	0.007613	417	0.004428	600	0.003112
500	105	0.016403	234	0.007678	412	0.004455	594	0.003127
1000	103	0.016575	231	0.007727	409	0.004475	590	0.003139

In general, a garbage collection policy at N-th update is more economical than that at time T, however, they have almost the same values in case of $c_1 = c_3$. Further, it is of interest that both T^* and N^* depend little on costs c_2/c_1 and c_2/c_3, and are given approximately by μK.

6. CONCLUSIONS

We have studied when to make garbage collections in the operation of a database which is useless at an upper limit K of total garbages. When a database is updated at a nonhomogeneous Poisson process, and an amount of garbage due to each update can be estimated and has Cdf $G(x)$, we have considered the model where a garbage collection is made at time T or at N-th update.

Applying the theory of cumulative processes to this model, we have obtained the expected cost and have discussed the optimal T^* and N^* which minimize the expected cost. From numerical examples, it has been shown that the optimal policies are determined approximately by an upper limit K.

In this paper, we adopt the time T and the update number N as indicators of operation of a database. If total garbages or remaining storage and memory areas would be estimated from some methods, we could consider similar models where garbage collections are made at total garbages or remaining areas.

REFERENCES

1. H. G. Baker Jr., List processing in real time on a serial computer, *Communications of the ACM,* 1978, *21*, p. 280-294.
2. G. L. Steele Jr., Multiprocessing compactifying garbage collection, *Communications of the ACM,* 1975, *18*, p. 495-508.
3. J. Cohen, Garbage collection of linked data structures, *ACM Computing Surveys,* 1981, *13*, p. 341-367.
4. H. T. Kung and S. W. Song, *An efficient parallel garbage collection system and it correctness proof,* I.E.E.E. Science, 1977, p. 120-131.
5. H. Lieberman and C. Hewitt, A real-time garbage collector based on the lifetimes of objects, *Communications of the ACM*, 1983, *26*, p. 419-429.
6. T. Yuasa, Real-time garbage collection on general-purpose machines, *Journal of Systems and Software*, 1990, *11*, p. 181-198.
7. H. M. Taylor, Optimal replacement under additive damage and other failure models, *Naval Res. Logist. Quart.*, 1975, *22*, p. 1-18.
8. S. M. Ross, *Applied Probability Models with Optimization Applications,* Holden-Day, San Francisco, 1970.

APPENDIX

When $G^{(j+1)}(x)/G^{(j)}(x)$ is strictly decreasing in j, we prove that

$$1 - Q_1(T) = \sum_{j=0}^{\infty} \frac{(\lambda T)^j}{j!} G^{(j+1)}(x) \Big/ \sum_{j=0}^{\infty} \frac{(\lambda T)^j}{j!} G^{(j)}(x),$$

is also strictly decreasing in T for any $x > 0$.

Differentiating $1 - Q_1(T)$ with respect to T,

$$\frac{\lambda}{\left[\sum_{j=0}^{\infty} \frac{(\lambda T)^j}{j!} G^{(j)}(x)\right]^2} \times \left\{\sum_{j=0}^{\infty} \frac{(\lambda T)^j}{j!} G^{(j)}(x) \sum_{i=0}^{\infty} \frac{(\lambda T)^i}{i!} G^{(i+2)}(x) - \sum_{j=0}^{\infty} \frac{(\lambda T)^j}{j!} G^{(j+1)}(x) \sum_{i=0}^{\infty} \frac{(\lambda T)^i}{i!} G^{(i+1)}(x)\right\}. \tag{A.1}$$

The numerator is rewritten as

$$\begin{aligned}
&\sum_{j=0}^{\infty} \frac{(\lambda T)^j}{j!} \sum_{i=0}^{\infty} \frac{(\lambda T)^i}{i!} G^{(j)}(x)\, G^{(i+1)}(x) \left[\frac{G^{(i+2)}(x)}{G^{(i+1)}(x)} - \frac{G^{(j+1)}(x)}{G^{(j)}(x)}\right] \\
&= \sum_{j=0}^{\infty} \frac{(\lambda T)^j}{j!} \sum_{i=0}^{j-1} \frac{(\lambda T)^i}{i!} G^{(j)}(x)\, G^{(i+1)}(x) \left[\frac{G^{(i+2)}(x)}{G^{(i+1)}(x)} - \frac{G^{(j+1)}(x)}{G^{(j)}(x)}\right] \\
&\quad + \sum_{j=0}^{\infty} \frac{(\lambda T)^j}{j!} \sum_{i=j}^{\infty} \frac{(\lambda T)^i}{i!} G^{(j)}(x)\, G^{(i+1)}(x) \left[\frac{G^{(i+2)}(x)}{G^{(i+1)}(x)} - \frac{G^{(j+1)}(x)}{G^{(j)}(x)}\right].
\end{aligned} \tag{A.2}$$

Note that the second term in (A.2) is negative since $G^{(j+1)}(x)/G^{(j)}(x)$ is strictly decreasing.

Changing the summation of i and j, the first term in (A.2) is

$$\sum_{i=0}^{\infty} \frac{(\lambda T)^i}{i!} \sum_{j=i+1}^{\infty} \frac{(\lambda T)^j}{j!} G^{(j)}(x)\, G^{(i+1)}(x) \left[\frac{G^{(i+2)}(x)}{G^{(i+1)}(x)} - \frac{G^{(j+1)}(x)}{G^{(j)}(x)} \right]. \tag{A.3}$$

Changing i into j with each other, (A.3) is

$$\begin{aligned} &\sum_{j=0}^{\infty} \frac{(\lambda T)^j}{j!} \sum_{i=j+1}^{\infty} \frac{(\lambda T)^i}{i!} G^{(i)}(x)\, G^{(j+1)}(x) \left[\frac{G^{(j+2)}(x)}{G^{(j+1)}(x)} - \frac{G^{(i+1)}(x)}{G^{(i)}(x)} \right] \\ &= \sum_{j=1}^{\infty} \frac{(\lambda T)^{j-1}}{(j-1)!} \sum_{i=j}^{\infty} \frac{(\lambda T)^{i+1}}{(i+1)!} G^{(i+1)}(x)\, G^{(j)}(x) \\ &\qquad \times \left[\frac{G^{(j+1)}(x)}{G^{(j)}(x)} - \frac{G^{(i+2)}(x)}{G^{(i+1)}(x)} \right]. \end{aligned} \tag{A.4}$$

Consequently, (A.2) is

$$\begin{aligned} &\sum_{j=0}^{\infty} \frac{(\lambda T)^j}{j!} \sum_{i=j}^{\infty} \frac{(\lambda T)^{i+1}}{(i+1)!} G^{(j)}(x)\, G^{(i+1)}(x) \\ &\quad \times \left[\frac{G^{(i+2)}(x)}{G^{(i+1)}(x)} - \frac{G^{(j+1)}(x)}{G^{(j)}(x)} \right] (i+1-j) < 0, \end{aligned}$$

which completes the proof.