

Stochastic Modeling of Database Backup Policy for a Computer System

Syouji Nakamura¹, Xufeng Zhao², Toshio Nakagawa²

¹Department of Human Life and Information, Kinjo Gakuin University, Nagoya, Japan; ²Department of Business Administration, Aichi Institute of Technology, Toyota, Japan.

Email: snakam@kinjo-u.ac.jp, g09184gg@aitech.ac.jp, toshi-nakagawa@aitech.ac.jp

Received January 3rd, 2013; revised February 1st, 2013; accepted February 9th, 2013

ABSTRACT

As the computer system has developed much in this highly information-oriented society, database security has become a very important problem and its backup strategies need to be made more efficiently and safety. The image copy method has been used as the most simple and dependable recovery mechanism for media failure. However, this method spends high overhead costs for massive data transmission and much processing time in the normal operation of the database. To cover such weak points, incremental and full backup methods are adopted before updated trucks reach a predetermined level. Moreover, when the number of full backup files exceeded a predetermined level, we stop incremental and full backups and switch it to the image copy. This paper applies cumulative damage model to backup of files in a database system, by putting damage shock by update, failure shock by database failure and damage by dumped files, and considers the tradeoff among overhead costs of image copy and incremental, full backup methods, and discusses analytically an optimal policy for the image copy backup interval. Finally, numerical examples are given in the case of Poisson process and exponential distributions.

Keywords: Database Management System; Image Copy; Full Backup; Incremental Backup; Cumulative Process

1. Introduction

In this highly information-oriented society, database security [1] in computer systems has become a very important problem. Some recovery techniques in a database management system have to be prepared previously for emergency of troubles, using backup, check pointing, reorganization, fault-tolerant technologies [2,3]. The backup service and its protection techniques have been studied and programmed by many researchers and engineers, e.g., the recent work Peer-to-Peer (P2P) backup system [4-6], backup and protection schemes for a distributed system [7,8].

When we refer to the backup techniques for the database systems, image copy [9] is the most simple and dependable method to ensure the safety of data and is always to take the backup copies of all files in other places and to take out them if files in the original secondary media are broken. However, this method takes many hours, storages and costs when files become large because it stores all files. Frequent image copy backup seems not be so reasonable so that it is often restricted to a weekly or monthly schedule, although the increasing speed and capacity of backup media could make overnight backup to be a more realistic proposition. To make

the backup copies efficiently, we might dump only files that have changed since the last backup, which is called an incremented backup [9]. This would lessen significantly both time and size of backup. Further, the recovery techniques for database failures and the backup schemes for broken hard disks were studied [2,10,11].

However, there are only a few studies that focus to the scheduling problems of backup and recovery methods, although many related techniques are available. That is, we propose that backup schedule is a stochastic decision making process from the viewpoints of management and can be analyzed by the theory of stochastic processes. Especially, with regarding to backup modeling, there have been very few research papers that studied analytically optimal policies for a database system. Most problems were concerned with several ways to introduce backup methods in techniques. Optimal full backup policies and incremental backup schedules have been studied [12-14]. Even so, as referred as in [9], image copy backup is necessary to any database system for its whole security strategy, although it is restricted to be performed weekly or monthly until now. Such a strict periodic image copy backup policy is unreasonable and could be optimized from the following two points: 1) this backup technique has its superiority that it could copy all files

without any compression and make backup simpler and reliable; 2) its performance needs to be combined with incremental and full backup schedules, *i.e.*, when the total cost for incremental and full backups exceeds some level, it is reasonable to do such an image copy backup to renew the database.

Thus, we propose the following backup policy which ensures the safety of data and saves hours: The image copy is carried out at scheduled times, and between these backups, the incremental backup or full backup at each files, which takes all copies of newly updated files since the image copy, is done. That is, the image copy with large overhead is done at long interval and the incremental and full backups with small overhead are done at short interval.

In this paper, we apply the cumulative damage model [15,16] to the backup of files in a database system, by putting *damage shock* by *update*, *failure shock* by *database failure* and *damage* by *dumped files*. These models, which play an important role in reliability theory, are considered as a sequence of shocks that occur randomly in time and give some amount of damage to a unit. The damage is accumulated to the current damage level, weakens the unit gradually, and makes it failure when the total damage exceeds a failure level [15]. As applications of cumulative damage processes in computer science, such models have been applied successfully to backup policies for a database system [12-14] and garbage collection policies in memory management [17,18].

The following sections are organized as: Section 2 introduces working schemes of incremental, full and image copy backups, by taking a database system with n files as an example. Section 3 formulates the stochastic backup model, which combines the incremental, full and image copy backup, *i.e.*, the incremental backup is done when the number of updated trucks does not exceed a certain threshold value to an individual file; the full backup is done when the number of updated trucks exceeds a certain threshold value to an individual file; and the image copy backup is done at a planned time and when the number of full backup files exceeds a certain threshold value in a database. Then, we introduce costs suffered for the overheads of three backups and obtain the expected cost rate between the image copies. Section 4 discusses an optimal interval of the image copy that minimizes the expected cost in Section 3, and in Section 4, we compute the model and its policies as a numerical example in the case of Poisson process and exponential distribution. Finally, in Section 5, concluding remarks and further studies are given.

2. Backup Schedule

We consider a database system with n files that are composed of the same size of trucks. In this database system,

we consider an incremental backup that takes all copies of newly updated trucks since the previous image copy and its overhead is increasing with the number of newly updated trucks. For example, if all updated trucks are included in the previous updated ones, then the number of transferred data is the same as the previous one. However, if the updated trucks have some different ones from the previous ones, then the number of transferred data is increasing by their differences. Taking out copies of previous backups can make the recovery of a database easily and rapidly, when some errors have occurred in storage media.

For example, we consider a database with 6 files: In **Figure 1**, when the updated track exceeds a threshold value Z at each file, e.g., (a), (e) and (f), we do full backup to these files. Moreover, files (b)-(d) which do not exceed the threshold value Z execute the incremental backup. In **Figure 2**, when the number of full backup files in the database exceeds a threshold value, we do the image copy backup for this database. In this figure, when the number of 5/6 files in the database is targeted, e.g., files (a)-(c), (e) and (f) are needed to be done by full backup, we stop the full and incremental backups and switch it to the operation of image copy backup.

It is well known that when the number of updated trucks exceeds a threshold level Q in a file, the overhead of incremental backup is larger than that of full backup [10]. The value of Q/m is about 60% [10] in a database system, where m is the total trucks in a file. Thus, if the number of updated trucks exceeds a level Z ($0 < Z \leq Q$), we should make the full backup instead of the incremental backup. Moreover, when the number of updated files exceeds a threshold value in a database, the overhead of full backup is larger than that of image copy backup [9].

Figure 3 shows in the incremental backup method that a usual backup storage volume is small, however, it is necessary to preserve all backed up files. Therefore, the

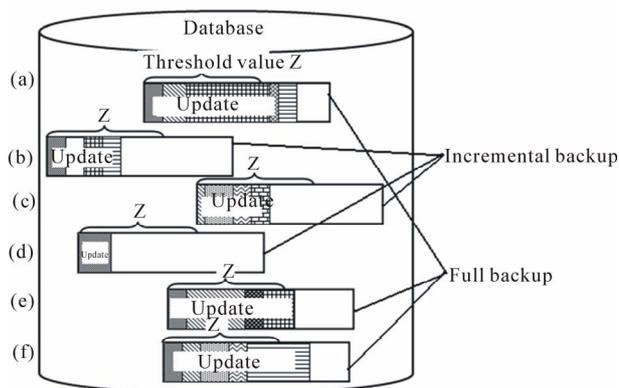


Figure 1. Execution of full and incremental backups in a database.

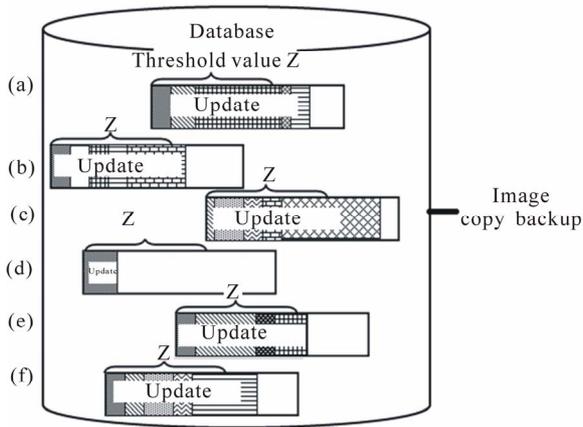


Figure 2. Execution of image copy backup in a database.

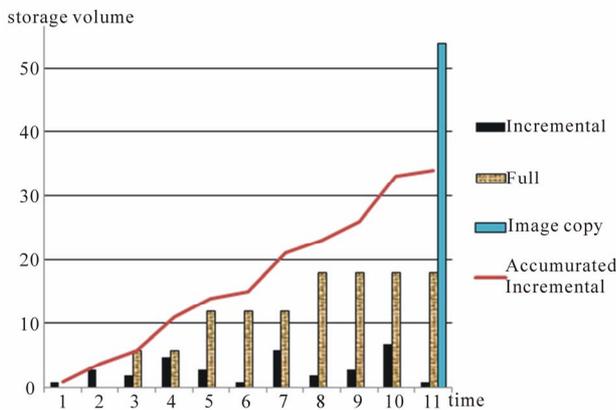


Figure 3. Storage volume every time of incremental, full and image copy backups.

accumulation of incremental backup files becomes large storage volume. In the full backup method, the storage volume is equal to the size of the object files. Moreover, in the image copy backup method, the storage volume is the same size of database.

From above discussions, it becomes an important problem in actual backup schemes when to create an image copy. We want to lessen the number of the image copy with large overhead, however, the overheads of the incremental and full backup increase adaptively with the number of newly updated trucks and files. From this point of view and we should decide the image copy interval, by comparing the overheads among three backups.

3. Expected Cost

We formulate the following stochastic backup models: The image copy backup is done at a planned time T and database initialization is made at such a scheduled time. The incremental backup is done when the volume of updated trucks does not exceed a certain threshold value Z to an individual file. The full backup is done when the

volume of updated trucks exceeds a certain threshold value Z to an individual file.

It is assumed that a database system with n files is updated according to a nonhomogeneous Poisson process with an intensity function $\lambda(t)$ and a mean-value function $R(t)$, i.e., $R(t) \equiv \int_0^t \lambda(u) du$ [15]. Then, the probability that j -th update occurs exactly during $(0, t]$ is

$$H_j(t) \equiv \frac{[R(t)]^j}{j!} e^{-R(t)} \quad (j = 0, 1, 2, \dots), \quad (1)$$

where $R(t) \equiv 0$.

Further, let W_j denote an amount of trucks of each file, which they are updated or are new created since the last backup at the j -th update. It is assumed that each W_j has an identical probability distribution $G(x) \equiv \Pr\{W_j \leq x\}$ ($j = 1, 2, \dots$). Then, the total amount of updated trucks $Z_j \equiv \sum_{i=1}^j W_i$ up to j -th update where $Z_0 \equiv 0$ has a distribution

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

and $G^{(0)}(x) \equiv 1$ for $x \geq 0$, 0 for $x < 0$, where $G^{(j)}(x)$ is the j -fold convolution of $G(x)$ with itself. Then, the probability that the total amount of updated trucks exceeds exactly a threshold level K at the j -th update is $G^{(j-1)}(K) - G^{(j)}(K)$ [15]. Let $Z(t)$ be the total amount of updated trucks at time t . Then, the distribution of $Z(t)$ is

$$\Pr\{Z(t) \leq x\} = \sum_{j=0}^{\infty} H_j(t) G^{(j)}(x). \quad (3)$$

Suppose that when the total amount of updated trucks exceeds a threshold level Z at time jT ($j = 1, 2, \dots$), we want to do the full backup for this file. When the total amount of updated trucks does not exceeds Z , we do the incremental backup, and this probability is

$$\bar{\Phi}_1(T) = \sum_{j=0}^{\infty} G^{(j)}(Z) H_j(T). \quad (4)$$

Oppositely, when the total amount of updated trucks exceeds Z , we do the full backup, and this probability is

$$\begin{aligned} \Phi_1(T) &= \sum_{j=0}^{\infty} [G^{(j)}(Z) - G^{(j+1)}(Z)] \int_0^T H_j(t) \lambda(t) dt \\ &= 1 - \sum_{j=0}^{\infty} G^{(j)}(Z) H_j(T). \end{aligned} \quad (5)$$

Suppose that there exist n files in the database. However, it would be useless to do separately backup policies for each file. It is assumed that if the total amount of updated trucks of j ($j = K, K+1, \dots, n$) files exceeds a threshold level Z at time T , then the full backup is done for such j files. In this case, the probability that the image copy backup is done for this database is

$$\begin{aligned}\Phi_n(T) &= \sum_{j=K}^n \binom{n}{j} [\Phi_1(T)]^j [\bar{\Phi}_1(T)]^{n-j} \\ &= \frac{n!}{(n-K)!(K-1)!} \int_0^T [\Phi_1(u)]^{K-1} [\bar{\Phi}_1(u)]^{n-K} d\Phi_n(u).\end{aligned}\quad (6)$$

If the total amount of updated trucks of j ($j=0,1,\dots,K-1$) files exceeds a threshold level Z at time T , then the full backup is done for such j files. In this case, the probability that full backup is done for the database is

$$\bar{\Phi}_n(T) = \sum_{j=0}^{K-1} \binom{n}{j} [\Phi_1(T)]^j [\bar{\Phi}_1(T)]^{n-j}. \quad (7)$$

Next, we introduce the following costs:

c_1 : Full and incremental backup costs per unit of time when the number of files whose updated trucks exceeds Z is less than K files.

c_2 : Image copy backup cost per unit of time when the number of files whose updated trucks exceeds Z is more than K files with $c_2 > c_1$.

c_3 : Cost of database backup and initialization at time T . The expected cost rate is

$$\begin{aligned}C(T) &= \frac{c_1 \left[T \bar{\Phi}_n(T) + \int_0^T t d\Phi_n(t) \right] + c_2 \int_0^T (T-t) d\Phi_n(t) + c_3}{T} \\ &= \frac{c_1 \int_0^T \bar{\Phi}_n(t) dt + c_2 \int_0^T \Phi_n(t) dt + c_3}{T}.\end{aligned}\quad (8)$$

The first item of right-hand side of (8) is the full and incremental backup cost and the second item is the image copy backup cost. Note that $C(0) = \infty$. If $T = \infty$, then the policy corresponds to the image copy only at the total amount of updated trucks exceeds the threshold level Z , and the expected cost rate is

$$C(\infty) \equiv \lim_{T \rightarrow \infty} C(T) = \frac{c_2}{\mu_n}, \quad (9)$$

where

$$\begin{aligned}\mu_n &\equiv \int_0^\infty \bar{\Phi}_n(t) dt \\ &= \sum_{j=0}^{K-1} \binom{n}{j} \int_0^\infty [\Phi_1(t)]^j [\bar{\Phi}_1(t)]^{n-j} dt,\end{aligned}\quad (10)$$

which is the mean time until the total updated trucks exceeds K .

4. Optimal Policy

We obtain an optimal time T^* ($0 < T^* < \infty$) which minimizes the expected cost $C(T)$ in (8). Letting $C'(T) = 0$,

$$\int_0^T [\Phi_n(T) - \Phi_n(t)] dt = \frac{c_3}{c_2 - c_1}, \quad (11)$$

i.e.,

$$\int_0^T t d\Phi_n(t) = \frac{c_3}{c_2 - c_1}, \quad (12)$$

whose left-hand side is strictly increasing from 0 to μ_n . Thus, if $\mu_n > c_3/(c_2 - c_1)$, then there exists a finite and unique T^* ($0 < T^* < \infty$) that satisfies (12) and the resulting cost rate is

$$C(T^*) = c_1 + (c_2 - c_1) \Phi_n(T^*). \quad (13)$$

5. Numerical Example

Suppose that $H_j(t) = \frac{(\lambda t)^j}{j!} e^{-\lambda t}$ and $G(Z) = 1 - e^{-\mu Z}$.

Then, we have

$$G^{(j)}(Z) = \sum_{i=j}^{\infty} \frac{(\mu Z)^i}{i!} e^{-\mu Z} \quad (j=0,1,2,\dots).$$

Thus, (4) is rewritten as

$$\begin{aligned}\bar{\Phi}_1(t) &= \sum_{j=0}^{\infty} \left[\sum_{i=j}^{\infty} \frac{(\mu Z)^i}{i!} e^{-\mu Z} \right] \frac{(\lambda T)^j}{j!} e^{-\lambda T} \\ &= 1 - \sum_{j=0}^{\infty} \frac{(\lambda T)^j}{j!} e^{-\lambda T} \sum_{i=0}^{j-1} \frac{(\mu Z)^i}{i!} e^{-\mu Z},\end{aligned}\quad (14)$$

i.e.,

$$\Phi_1(T) = \sum_{j=0}^{\infty} \frac{(\lambda T)^j}{j!} e^{-\lambda T} \sum_{i=0}^{j-1} \frac{(\mu Z)^i}{i!} e^{-\mu Z}, \quad (15)$$

where $\sum_{i=0}^{-1} \equiv 1$. Because

$$\begin{aligned}\varphi_1(T) &\equiv \Phi_1'(T) \\ &= \lambda \sum_{j=0}^{\infty} \left[\frac{(\lambda T)^{j-1}}{(j-1)!} - \frac{(\lambda T)^j}{j!} \right] e^{-\lambda T} \sum_{i=0}^{j-1} \frac{(\mu Z)^i}{i!} e^{-\mu Z} \\ &= \lambda \sum_{j=0}^{\infty} \frac{(\lambda T)^j}{j!} e^{-\lambda T} \frac{(\mu Z)^j}{j!} e^{-\mu Z},\end{aligned}\quad (16)$$

from (6),

$$\begin{aligned}\Phi_n'(t) &= \frac{n!}{(n-K)!(K-1)!} [\Phi_1(t)]^{K-1} [\bar{\Phi}_1(t)]^{n-K} \varphi_1(t) \\ &= \frac{n!}{(n-K)!(K-1)!} \left[\sum_{j=0}^{\infty} \frac{(\lambda t)^j}{j!} e^{-\lambda t} \sum_{i=0}^{j-1} \frac{(\mu Z)^i}{i!} e^{-\mu Z} \right]^{K-1} \\ &\quad \times \left[1 - \sum_{j=0}^{\infty} \frac{(\lambda t)^j}{j!} e^{-\lambda t} \sum_{i=0}^{j-1} \frac{(\mu Z)^i}{i!} e^{-\mu Z} \right]^{n-K} \\ &\quad \times \lambda \sum_{j=0}^{\infty} \frac{(\lambda t)^j}{j!} e^{-\lambda t} \frac{(\mu Z)^j}{j!} e^{-\mu Z}.\end{aligned}\quad (17)$$

Therefore, (12) becomes

$$\int_0^T t d\Phi_n(t) = \frac{n!}{(n-K)!(K-1)!} \times \int_0^T \lambda t \left[\sum_{j=0}^{\infty} \frac{(\lambda t)^j}{j!} e^{-\lambda t} \sum_{i=0}^{j-1} \frac{(\mu Z)^i}{i!} e^{-\mu Z} \right]^{K-1} \times \left[1 - \sum_{j=0}^{\infty} \frac{(\lambda t)^j}{j!} e^{-\lambda t} \sum_{i=0}^{j-1} \frac{(\mu Z)^i}{i!} e^{-\mu Z} \right]^{n-K} \times \sum_{j=0}^{\infty} \frac{(\lambda t)^j}{j!} e^{-\lambda t} \frac{(\mu Z)^j}{j!} e^{-\mu Z} dt = \frac{c_3}{c_2 - c_1} \tag{18}$$

We obtain the optimal value T^* which satisfies (18), and the resulting expected cost rate is

$$C(T^*) = c_1 + (c_2 - c_1) \sum_{j=K}^n \binom{n}{j} [\Phi_1(T^*)]^j [\bar{\Phi}_1(T^*)]^{n-j} \tag{19}$$

Tables 1 and 2 indicate the cost rates $c_3/(c_2 - c_1)$ for optimal times T^* when $K = 6, 12$ and $n = 10, 20$. From **Tables 1 and 2**, when the value of $c_3/(c_2 - c_1)$ increases, the interval of T^* becomes long. This shows that we should lengthen the image copy backup interval, if the value of c_3 is large compared to the value of $c_2 - c_1$. From the tables, when $\mu Z = 5.10$, the value of $c_3/(c_2 - c_1)$ to the value of T^* becomes about twice. When $\mu Z = 5.0, K = 6, n = 10, \lambda = 0.05$ and $c_3/(c_2 - c_1) = 9.875, T^*$ becomes 25. When the unit of T is a day, we should execute the image copy backup

Table 1. Optimal T^* when $\mu Z = 5.0$.

T^*	$c_3/(c_2 - c_1)$			
	$K = 6, n = 10$		$K = 12, n = 20$	
	$\lambda = 0.05$	$\lambda = 0.08$	$\lambda = 0.05$	$\lambda = 0.08$
15	0.115	4.9726	0.001	1.150
20	1.654	32.613	0.128	23.581
25	9.875	93.454	2.653	104.720
30	33.862	166.324	19.410	202.318

Table 2. Optimal T^* when $\mu Z = 10.0$.

T^*	$c_3/(c_2 - c_1)$			
	$K = 6, n = 10$		$K = 12, n = 20$	
	$\lambda = 0.05$	$\lambda = 0.08$	$\lambda = 0.05$	$\lambda = 0.08$
15	0.240	10.457	0.003	2.487
20	3.473	68.794	0.278	32.346
25	20.773	197.659	5.735	223.785
30	71.369	352.425	41.837	428.991

every 25 days. In addition, when $\mu Z = 10.0, K = 12, n = 20, \lambda = 0.05$ and $c_3/(c_2 - c_1) = 41.837$ in **Table 2**, T^* becomes 30. When the unit of T is a day, we should execute the image copy backup every 30 days. In the real world, we can request the value of $c_3/(c_2 - c_1)$ comparatively easily. Then, when $c_3/(c_2 - c_1) = 41.837, \mu Z = 10.0, K = 12, n = 20$ and $\lambda = 0.05$, it is necessary to do the image copy back up every month.

6. Conclusions

We have considered the problem when to make the incremental, full and image copy backups, under the assumptions that the overhead of backups depends on the total amount of newly updated files in a database. We have obtained the expected cost rate until the image copy backup, and have discussed the optimal interval of the image copy backup that minimizes it. It has been shown that the optimal interval is given by a finite and unique solution of an equation.

As a further problem, it would be necessary to consider the model where the backup of only newly updated files is made, and to compare it with the model studied in this paper.

7. Acknowledgements

This work is partially supported by the Grant-in-Aid for Scientific Research (C) of Japan Society for the Promotion of Science under Grant No. 22500897 and No. 24530371.

REFERENCES

- [1] T. Ward, "Security of Backup Data," *Information Systems Security*, Vol. 15, No. 1, 2006, pp. 31-34. [doi:10.1201/1086.1065898X/45926.15.1.20060301/92683.6](https://doi.org/10.1201/1086.1065898X/45926.15.1.20060301/92683.6)
- [2] T. Haerder and J. Muckstadt, "Optimal Policy for Batch Operations: Backup, Checkpointing, Reorganization, and Updating," *ACM Transactionson Database Systems*, Vol. 2, No. 3, 1977, pp. 209-222. [doi:10.1145/320557.320558](https://doi.org/10.1145/320557.320558)
- [3] I. Lazaridis, *et al.*, "Fault Tolerant Evaluation of Continuous Selection Queries over Sensor Data," *International Journal of Distributed Sensor Networks*, Vol. 5, No. 4, 2009, pp. 338-360. [doi:10.1080/15501320701585600](https://doi.org/10.1080/15501320701585600)
- [4] H. Jarraya and M. Laurent, "A Secure Peer-to-Peer Backup Service Keeping Great Autonomy While under the Supervision of a Provider," *Computers & Security*, Vol. 29, No. 2, 2010, pp. 180-195. [doi:10.1016/j.cose.2009.10.003](https://doi.org/10.1016/j.cose.2009.10.003)
- [5] M. Gramaglia, M. Urueña and I. Martinez-Yelmo, "Off-Line Incentive Mechanism for Long-term P2P Backup Storage," *Computer Communications*, Vol. 35, No. 12, 2012, pp. 1516-1526. [doi:10.1016/j.comcom.2012.04.017](https://doi.org/10.1016/j.comcom.2012.04.017)
- [6] M. Fesci-Sayita, E. TurhanTunalib and A. Murat Tekalpc, "Resilient Peer-to-Peer Streaming of Scalable Video over

- Hierarchical Multicast Trees with Backup Parent Pools,” *Signal Processing: Image Communication*, Vol. 27, No. 2, 2012, pp. 113-125. [doi:10.1016/j.image.2011.11.004](https://doi.org/10.1016/j.image.2011.11.004)
- [7] G. Bella, C. Pistagna and S. Riccobene, “Distributed Backup through Information Dispersal,” *Electronic Notes in Theoretical Computer Science*, Vol. 142, No. 3, 2006, pp. 63-77. [doi:10.1016/j.entcs.2004.11.046](https://doi.org/10.1016/j.entcs.2004.11.046)
- [8] J. Ma, *et al.*, “A Novel Adaptive Current Protection Scheme for Distribution Systems with Distributed Generation,” *International Journal of Electrical Power & Energy Systems*, Vol. 43, No. 1, 2012, pp. 1460-1466.
- [9] <http://www.backup4all.com/>
- [10] K. Suzuki and K. Nakajima, “Storage Management Software,” *FUJITSU*, Vol.48, No. 4, 1997, pp. 389-397.
- [11] G. Lohman and A. Reuter, “Principles of Transaction Oriented Database Recovery,” *ACM Computing Surveys*, Vol. 15, No. 4, 1983, pp. 287-317. [doi:10.1145/289.291](https://doi.org/10.1145/289.291)
- [12] C. Qian, S. Nakamura and T. Nakagawa, “Cumulative Damage Model with Two Kinds of Shocks and Its Application to the Backup Policy,” *Journal of the Operations Research Society of Japan*, Vol. 42, No. 4, 1999, pp. 501-511. [doi:10.1016/S0453-4514\(00\)87116-1](https://doi.org/10.1016/S0453-4514(00)87116-1)
- [13] S. Nakamura, C. Qian, S. Fukumoto and T. Nakagawa, “Optimal Backup Policy for a Database System with Incremental and Full Backups,” *Mathematical and Computer Modelling*, Vol. 38, No. 11-13, 2003, pp. 1373-1379. [doi:10.1016/S0895-7177\(03\)90140-3](https://doi.org/10.1016/S0895-7177(03)90140-3)
- [14] C. Qian, Y. Huang, X. Zhao and T. Nakagawa, “Optimal Backup Interval for a Database System with Full and Periodic Increment Backup,” *Journal of Computers*, Vol. 5, No. 4, 2010, pp. 557-564. [doi:10.4304/jcp.5.4.557-564](https://doi.org/10.4304/jcp.5.4.557-564)
- [15] T. Nakagawa, “Shock and Damage Models in Reliability Theory,” Springer, London, 2007.
- [16] S. Nakamura and T. Nakagawa, “Stochastic Reliability Modeling, Optimization and Applications,” World Scientific, Singapore, 2010.
- [17] X. Zhao, S. Nakamura and T. Nakagawa, “Two Generational Garbage Collection Models with Major Collection Time,” *IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences*, Vol. E94-A, No. 7, 2011, pp.1558-1566.
- [18] X. Zhao, S. Nakamura and T. Nakagawa, “Optimal Tenuing and Major Collection Times for a Generation Garbage Collector,” *Asia-Pacific Journal of Operational Research*, Vol. 29, No. 3, 2012, p. 1240018 (17 pages).