Optimal Repairman Assignment in Two Symmetric Maintenance Models

Ger Koole
CWI
P.O. Box 94079, 1090 GB Amsterdam
The Netherlands

Abstract

Two maintenance models (a warm and a cold stand-by model) consisting of m parallel groups of identical components are considered. Components have exponentially distributed times to failure. There is a single repairman who can be assigned to one of the failed components. It is shown for both models that the repairman should be assigned to an item of the group with the smallest number of functioning components. The optimality results are obtained by adapting an iterative method which proved to be successful in the study of routing models.

1991 Mathematics Subject Classification: 90B22, 90B25

Keywords and Phrases: parallel components, repairman scheduling, smallest group policy, dynamic programming, delayed information

Published in: European Journal of Operations Research 82:295-301, 1995

1. Introduction

In this paper we consider the assignment of a repairman for two different models. In both models there are m groups of components, each consisting in perfect state of B functioning components. In section 3 we look at a warm stand-by model, in which each functioning component has an, equally distributed, exponential time to failure. This system is up if at least k groups have one or more functioning components, or, alternatively, if each group has at least k' components. It is shown that the repairman should always be assigned to the group with the smallest number of components at the beginning of each repair period. This policy is called the Smallest Group Policy (SGP).

In section 4 we look at a cold stand-by model. Here the system is up if each group has at least one functioning component. Only those components necessary for the system to be up can fail, all with equal rates, and all other functioning components stay as good as new. Thus, as long as the system is up, one component in each group has an exponentially distributed time to failure. If the system is down (meaning that at least one group has no functioning components), no components can fail. Again, the SGP maximizes the expected time that the system is up.

Of special interest is the operation of the repairman. The times at which the repairman completes service are generated by a *Markov decision arrival process* (MDAP). This process is introduced and discussed in section 2. With the MDAP we can model each sequence of completion times which does not depend on the numbers of functioning components. But also if the repair times depend on the state of the groups of components, an MDAP can sometimes be used to model the repair completion times. An example of such a repairman process is one in which the times at which the repairman goes on vacation can be controlled, depending on the numbers of failed components.

Our models are strongly related to routing models (which are queueing models with multiple queues in which customers on arrival have to be assigned to one of the queues). Indeed, if we let the completion of a repair in group j correspond to an arrival of a customer in queue j, we get a routing model, which is equivalent to the maintenance model in the sense that a repair completion and an arrival both increase the jth component of the state vector by 1. However, in the resulting routing model the decision on where to route an arriving customer to is not taken at the arrival instant, as in standard routing models. Because the repairman is assigned to a group at the beginning of a repair period, modeling the repair taking place on the spot, the decision on where to route the corresponding arrival is taken just after the previous arrival. Thus our maintenance model is equivalent to a routing model where the decisions have to be taken some time in advance. Note that, if the decision is to route a customer to the shortest queue (at the time of the previous arrival), this queue need not be the shortest any more at the moment of arrival, due to the possible departures from the queues.

This routing model with early decisions can also be seen as a model in which the information on the state of the system is delayed. Usually, models with delayed information are discrete time models in which the decisions have to be taken a fixed number of epochs in advance. See Kuri & Kumar [6] and Altman & Koole [1] for recent results on routing models with delayed information. Our model differs from their type of model in that the length of the delay is not fixed (and not known to the controller).

The equivalence of routing models and repairman assignment models was first observed by Smith [8] for a routing model consisting of m heterogeneous exponential servers, without waiting places. Derman et al. [3] show for this model that sending each arriving customer to the fastest (slowest) free server minimizes (maximizes) the probability that an arriving customer finds all servers busy. For the routing model, minimizing the probability that the system is full is of interest. For the equivalent repairman assignment problem (without early decisions) the objective is to maximize the number of functioning components (which is equivalent to maximizing the number of busy servers).

To prove the results of sections 3 and 4 we use dynamic programming on an embedded discrete time model, which proved to be a successful method for routing models without delay. We generalize this method to include the early decisions, by adding an extra variable to the state space denoting the assignment of the repairman.

The warm stand-by model of section 3, without the early decisions, has already been studied by Katehakis & Melolidakis [5].

2. The repairman process

In this section we start with introducing the MDAP, which generates the repairman completion times. Then we will give several examples of repairman processes which can be modeled with an MDAP. The results of the next sections will be shown to hold for each MDAP. Thus a repairman should be assigned according to the SGP if its repair completion times can be modeled as an MDAP (and if the reward function satisfies the given conditions).

An MDAP can be seen as an Markov decision process (without yet specifying the reward structure) which additionally generates arrivals at the transition times. We will use arrivals in 2 classes, one to model repair completions, the other to model possible rescheduling of the repairman.

2.1. Definition. (Markov decision arrival process) Let Λ be the finite state space of a Markov decision process with transition intensities λ_{xay} with $x, y \in \Lambda$ and $a \in A(x)$, the set of actions in x. When this process moves from x to y, while action a was chosen, then with probability q^k_{xay} an arrival in class $1 \leq k \leq 2$ occurs.

The arrivals in class 1 represent repairman completion times (after which the repairman can be rescheduled). At the arrival of a class 2 customer the repairman is just allowed to go to another group. The choice to which group to send the repairman is not part of the MDAP. Thus, in the models of sections 3 and 4 there are 2 actions; one in the MDAP, controlling the repair completion times, and one at the groups, modeling the assignment of the repairman to a failed item in one of the groups. In the next sections the optimal assignment is obtained, given that the MDAP is operated optimally.

In case there is a single action in each state, the MDAP simplifies to the well known *Markov arrival process* (for a practical introduction, see Lucantoni [7]). Then the repair completion times do not depend on the numbers of functioning components. It can be shown that with this type of arrival process any independent arrival process can be approximated arbitrarily close (Asmussen & Koole [2]).

In general, the actions chosen in the MDAP are allowed to depend both on the state of the MDAP and on the numbers of functioning components, thereby making the MDAP depend on the state of the groups. We will give two examples of repairman processes modeled by MDAP's.

Example 1. (Repairman with interrupts) Suppose each repair time is distributed according to a phase type distribution with starting state α and ending state β. During the repair, the repairman can be interrupted and restarted at a possibly different group. This can be modeled by taking as state space the states of the PH-distribution, and 2 actions in each state. Normally (if action 1 is chosen) the transitions are according to the PH-distribution, with state α and β

identified. A class 1 customer is generated once the repairman enters state β. So far we modeled general independent repair times. However, if action 2 is chosen, the repair man goes to state α (say with rate γ), and a class 2 customer is generated.

The results in the next sections do not state anything on when to interrupt the repairman; they just state that when the repair completion times are controlled optimally, the repairman should always be assigned according to the SGP.

Example 2. (Repairman with controllable vacations) Again assume that each repair time is a phase type distribution, but that after each repair completion the repairman can be sent on a vacation, depending on the number of functioning components. This can be modeled by two PH-distributions, one representing only one repair time, one a vacation plus a repair time. Now, after each successful repair (i.e., after the generation of a class 1 arrival), one chooses between the two PH-distributions.

Intuitively, one expects that, for an appropriate reward function, the repairman can go on vacation when there are few failed components.

3. Warm stand-by model

In this section we study the model in which each functioning component has an exponential lifetime, independent of the state of the other components. The repairman completion times are generated by an MDAP. Without restricting generality we can assume that $\sum_y \lambda_{xay} = \gamma$ for all x and a.

Also assume that B, the maximum number of functioning components in each group, is finite. The state is denoted with (l, x, i), where $0 \leq l \leq m$ is the group the repairman is working on, x is the state of the arrival process and $i = (i_1, \dots, i_m)$ denotes the number of functioning components in each group. If $l = 0$, the repairman is idling. Because we can only assign the repairman to a group in which not all components are functioning, we have $i_l < B$ for $l > 0$. We use the following notation: $e = (1, \dots, 1)$, $e_j = (0, \dots, 0, 1, 0, \dots, 0)$ with the 1 in jth position, $e_0 = (0, \dots, 0)$. Let μ be the rate at which each functioning component fails. Uniformize the model such that $\gamma + mB\mu \leq 1$. Let $v^n_{(l,x,i)}$ be the expected maximum reward in n steps of the uniformized discrete time model, starting in state (l, x, i). The dynamic programming recursion for v^n is:

$$\begin{aligned} v^{n+1}_{(l,x,i)} = \max_a \Big\{ \sum_y \lambda_{xay} \Big(& q^1_{xay} \max_j \{ v^n_{(j,y,i+e_l)} \} + q^2_{xay} \max_j \{ v^n_{(j,y,i)} \} + \\ & (1 - q^1_{xay} - q^2_{xay}) v^n_{(l,y,i)} \Big) \Big\} + \\ & \mu \sum_{j=1}^m i_j v^n_{(l,x,i-e_j)} + (1 - \gamma - (i_1 + \cdots + i_m)\mu) v^n_{(l,x,i)}. \end{aligned} \tag{3.1}$$

The maximization ranges over all j with $i_j < B$. Action 0 is only allowed if $i = Be$, thus the repairman idles for a whole repair period if he finds at the beginning of that period B components functioning in each group. Note that there are no direct rewards; the rewards are only earned at the end of the planning horizon, that is, v^0 is the reward function. This is done to be able to get results on the system for a fixed time T.

The following lemma gives relations between the rewards in different states.

3.1. Lemma. *If*

$$v^n_{(l_1,x,i)} \geq v^n_{(l_2,x,i)} \qquad \textit{for } i_{l_1} \leq i_{l_2},\ i + e_{l_1} + e_{l_2} \leq Be, \tag{3.2}$$

$$v^n_{(l,x,i+e_{j_1})} \geq v^n_{(l,x,i+e_{j_2})} \ \textit{for } i_{j_1} \leq i_{j_2},\ i + e_{j_1} + e_{j_2} + e_l \leq Be, \tag{3.3}$$

and

$$v^n_{(l,x,i)} = v^n_{(l^*,x,i^*)} \qquad \textit{for } i^* \textit{ a permutation of } i,\ i^*_{l^*} = i_l,\ i + e_l \leq Be \tag{3.4}$$

hold for the reward function v^0, then they hold for all n.

Proof. By induction. Assume the lemma holds up to n. We prove each inequality by considering the terms concerning repair completions and the terms concerning failures separately. The inequality for the dummy term (i.e., the term concerning the transition from a state to itself) always follows immediately. We start with (3.2). Assume $i_{l_1} < i_{l_2}$. The case $i_{l_1} = i_{l_2}$ can be done with (3.4). Then, no matter which component fails, group l_1 has less working components than group l_2, and the failure terms follow easily using induction. This leaves the repair completion terms. First consider the outer maximization. Let a' be the maximizing action in (l_2, x, i). It is easily seen that it is sufficient to prove the inequality for a', i.e., for fixed a. Let l be the optimal action in state $(y, i + e_{l_2})$. If $l \neq l_1$, then l is also optimal in $(y, i + e_{l_1})$ and the repair completion terms follow by induction, using (3.3) with $j_1 = l_1$ and $j_2 = l_2$. If $l = l_1$, and there is no other queue with the same length (which would give the previous case), then l_1 is also optimal in $(y, i + e_{l_1})$. Note that $i_{l_1} + 1 < B$ (because $i_{l_1} < i_{l_2}$ and $i_{l_2} + 1 \leq B$), giving that l_1 is indeed an allowable action. Again (3.3) can be used.

For proving (3.3) we can also assume that $i_{j_1} < i_{j_2}$. The case $i_{j_1} = i_{j_2}$ can be done again with (3.4). The failure terms follow easily using induction (the terms for the extra customers in group j_1 and j_2 should be combined). As for (3.2) we can choose the optimal actions in $(l, x, i + e_{j_1})$ and $(l, x, i + e_{j_2})$ such that the same action is optimal in both states. Then the terms concerning repair completion follow using (3.3).

Equation (3.4) follows easily. □

From (3.2) it follows that the repairman should be assigned to the group with the smallest number of functioning components if v^0 yields more reward to states obtained by such actions. Thus the lemma gives conditions on v^0, the reward function, for the SGP to be optimal. Note that there are no inequalities relating states like (l, x, i) and (l, y, i). This means that the rewards can depend arbitrarily on the state of the MDAP; on the other hand it means that nothing can be said on the optimal action in the MDAP. It is necessary however for the optimality result that the actions taken in the MDAP are the optimal ones.

Remark 1. It is easily seen that the inequalities give no conditions on $v^n_{(0,x,Be)}$. This means that we can change $v^n_{(0,x,Be)}$ as we like, getting rid of the unnatural assumption that the repairman idles for a whole repair period if all components are functioning.

An obvious choice is freezing the arrival process until the next failure, at which time the repairman starts working on that component, i.e.,

$$v^{n+1}_{(o,x,Be)} = \mu B \sum_{j=1}^{m} v^n_{(j,x,Be-e_j)} + (1 - m\mu B) v^n_{(0,x,Be)}.$$

Remark 2. It is interesting to note that for the model without delay in the assignment the buffer size can depend on the group. Observe that in the current model there is no equation relating states like $i + e_j$ and i, like in the standard routing model with one server for each queue (e.g., equation (4.4) in Hordijk & Koole [4]). We do not need such an equation here: the total number of functioning components remains stochastically the same under each policy, due to the structure of the model and the fact that idleness is not allowed (unless all components are functioning). In fact, if we add such an equation, we cannot prove our results any more.

By uniformization (see e.g. [4]) we have the following result at T, for the original continuous time model. The result from 0 to T is obtained by integrating the rewards over the time horizon.

3.2. Theorem. *The SGP maximizes the rewards at T, or, alternatively, the (discounted) rewards from 0 to T, for all reward functions satisfying (3.2) to (3.4).*

Thus it remains to consider the allowable reward functions. First take $v^0_{(l,x,i)} = 1$ if there are more than k non-empty groups in i, and 0 otherwise. This function indeed satisfies the inequalities in the lemma, thus the SGP maximizes the probability that there are k or more groups with one or more functioning components, i.e. it maximizes the number of groups with at least one working component stochastically. A related reward function is $\mathrm{I}_{\{i_j \geq k\ \forall j\}}$. This is also an allowable choice, corresponding to a system in which each group must have at least k functioning components. This reward function is equivalent to $\mathrm{I}_{\{\min_j\{i_j\} \geq k\}}$, thus also $\min_j\{i_j\}$ is maximized stochastically by the SGP. These reward functions can be useful for systems with groups of parallel components in series, where the chain of groups performs as its worst part.

Katehakis & Melolidakis [5] obtained the same results for the model where the group to which the repaired component is added is selected at the repair completion time.

4. Cold stand-by model

In this section we consider the cold stand-by model. Again there are m groups of components of size B. The system is up whenever there is at least one functioning component in each group. When the system is down, that is, when there is a group without functioning components, no functioning components can fail. When the system is up only the m components required for the system to be up, one in each group, can fail, all with the same rate. Thus the working components have equally distributed exponential times to failure. Again the repairman process is governed by an MDAP.

If we want to maximize the probability that the system is up at T, the SGP might not be optimal, even in the model without early decisions, as the following example shows. Take $m = 2$, B large, assume that the repairman finishes repairs at times $1, 2, \ldots$. Further assume that in the starting state group 2 has no functioning components and that T is integer. If we assign the repairman according to the SGP there is a positive probability that the system is not functioning at T. However, if we assign the repairman to the first group up to $T-2$ and only to the second group at $T-1$, it is clear the system is up at T (but not between 0 and T). Thus the SGP is not optimal. However, if we take as objective the expected time the system is up between 0 to T, then the SGP is again optimal. To show this, we have to introduce immediate rewards. As we prefer to collect all rewards together at T, we will add an extra component to the state space, which is raised by 1 each time a component fails.

Assume, by uniformization, that $\gamma + m\mu \le 1$. The state is denoted by (l, x, i, k), which is the same as for the previous model, with one component added to count the number of failed components. The dynamic programming equation is:

$$v^{n+1}_{(l,x,i,k)} = \begin{cases} \max_a \Big\{ \sum_y \lambda_{xay} \Big(q^1_{xay} \max_j \{ v^n_{(j,y,i+e_l,k)} \} + q^2_{xay} \max_j \{ v^n_{(j,y,i,k)} \} + \\ \qquad (1 - q^1_{xay} - q^2_{xay}) v^n_{(l,y,i,k)} \Big) \Big\} + \\ \quad \mu \sum_{j=1}^{m} v^n_{(l,x,i-e_j,k+1)} + (1-\gamma-m\mu) v^n_{(l,x,i,k)} & \text{if } i_j > 0 \text{ for all } j, \\ \max_a \Big\{ \sum_y \lambda_{xay} \Big(q^1_{xay} \max_j \{ v^n_{(j,y,i+e_l,k)} \} + q^2_{xay} \max_j \{ v^n_{(j,y,i,k)} \} + \\ \qquad (1 - q^1_{xay} - q^2_{xay}) v^n_{(l,y,i,k)} \Big) \Big\} + \\ \quad (1-\gamma) v^n_{(l,x,i,k)} & \text{if } i_j = 0 \text{ for some } j. \end{cases} \tag{4.1}$$

Here we let the maximization over j range over all j with $i_j > 0$ and over 0, meaning that idleness is a possible action, even if not all components are functioning.

4.1. Lemma. *If*

$$v^n_{(l_1,x,i,k)} \ge v^n_{(l_2,x,i,k)} \qquad \textit{for } i_{l_1} \le i_{l_2},\ i + e_{l_1} + e_{l_2} \le Be, \tag{4.2}$$

$$v^n_{(l,x,i+e_{j_1},k)} \geq v^n_{(l,x,i+e_{j_2},k)} \quad \text{for } i_{j_1} \leq i_{j_2},\ i+e_{j_1}+e_{j_2}+e_l \leq Be, \tag{4.3}$$

$$\sum_{j_1=1}^{m} v^n_{(l,x,i-e_{j_1},k+1)} \geq m v^n_{(l,x,i,k)} \quad \text{for } i \geq e,\ i+e_l \leq Be, \tag{4.4}$$

$$v^n_{(l,x,i,k)} \geq v^n_{(0,x,i,k)} \quad \text{for } i+e_l \leq Be, \tag{4.5}$$

$$v^n_{(l,x,i+e_{j_1},k)} \geq v^n_{(l,x,i,k)} \quad \text{for } i+e_{j_1}+e_l \leq Be, \tag{4.6}$$

$$v^n_{(0,x,i+e_{j_1},k)} \geq v^n_{(j_1,x,i,k)} \quad \text{for } i+e_{j_1} \leq Be, \tag{4.7}$$

$$v^n_{(l,x,i,k+1)} \geq v^n_{(l,x,i,k)} \tag{4.8}$$

and

$$v^n_{(l,x,i,k)} = v^n_{(l^*,x,i^*,k)} \quad \text{for } i^* \text{ a permutation of } i,\ i^*_{l^*} = i_l,\ i+e_l \leq Be \tag{4.9}$$

hold for the reward function v^0, then they hold for all v^n.

Proof. The proof is similar to that of lemma 3.1. For the proof of (4.2) we refer to that of (3.2). (Note that the SGP is optimal not only because of (4.2), but also because of (4.5).) Also (4.3) goes follows easily unless $i_{j_1} = 0$ and $i_{j_2} > 0$. Then, for the failure terms, (4.4) should be used first with $i + e_{j_1}$ instead of i. Concerning (4.4), note that the optimal action in (l, x, i, k) is allowed in all left hand states (even in case that action is idling). Then the repair term follows easily. Also the failure terms follow easily, as $i \geq Be$. Equation (4.5) gives that idling is suboptimal and was needed in the proof of (4.4), in case $i = Be$. Its proof is easy, using (4.6) in the repair terms. In the proof of (4.6) we need (4.7) (which proof also follows easily) in case $i + e_{j_1} = Be$. Equations (4.8) (which is needed to prove (4.6) and (4.7)) and (4.9) follow directly. □

Now we can take $v^0_{(l,x,i,k)} = k$, giving that the SGP maximizes the expected number of components that failed in any number of jumps of the embedded chain. However, we are interested in the time that the system is up. But, components only fail if the system is up, with rate $m\mu$. Thus the policy that maximizes the number of components that failed, also maximizes the time the system is up. Note that (4.5) gives that idling is suboptimal.

4.2. Theorem. *The SGP maximizes the expected time the system is up between 0 and T.*

Acknowledgment. I thank Onno Boxma and Pieter Wartenhorst for many valuable comments on earlier versions of this paper.

References

[1] E. Altman and G.M. Koole. Control of a random walk with noisy delayed information. *Systems and Control Letters*, 24:207–213, 1995.

[2] S. Asmussen and G.M. Koole. Marked point processes as limits of Markovian arrival streams. *Journal of Applied Probability*, 30:365–372, 1993.

[3] C. Derman, G.J. Lieberman, and S.M. Ross. On the optimal assignment of servers and a repairman. *Journal of Applied Probability*, 17:577–581, 1980.

[5] M.N. Katehakis and C. Melolidakis. On the optimal maintenance of systems and control of arrivals in queues. Technical report, Graduate School of Management, Rutgers University, 1989.

[6] J. Kuri and A. Kumar. Optimal control of arrivals to queues with delayed queue length information. In *Proceedings of the 31th IEEE Conference on Decision and Control*, 1992.

[7] D.M. Lucantoni. New results on the single server queue with a batch Markovian arrival process. *Stochastic Models*, 7:1–46, 1991.

[8] D.R. Smith. Optimal repair of a series system. *Operations Research*, 26:653–662, 1978.