Centrum voor Wiskunde en Informatica

REPORT*RAPPORT*

Optimal repairman assignment in two maintenance models which are equivalent to routing models with early decisions

G. Koole

Department of Operations Reasearch, Statistics, and System Theory

BS-R9301 1993

Optimal Repairman Assignment in Two Maintenance Models Which are Equivalent to Routing Models With Early Decisions

Ger Koole
CWI
P.O. Box 4079, 1009 AB 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 the failed components. It is shown for both models that the repairman should be assigned to the group with the smallest number of functioning components. Our maintenance models are equivalent to routing models with delayed state information. The optimality results are obtained by adapting a method which proved to be successful in the study of routing models.

1991 Mathematics Subject Classification: 90B22, 90B25

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

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 of at most B functioning components. In section 2 we look at a warm stand-by model, in which each functioning component has an, equally distributed, exponential time to failure. The system is up if at least k groups have one or more functioning components, or if each group has at least k' components. The times at which the repairman completes service are independent of the numbers of functioning components, but otherwise arbitrary. (The repair completion times are generated by a Markov Arrival Process, which will be introduced in section 2.) It is shown that the repairman should 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 3 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. The operation of the repairman is somewhat more general than in the warm stand-by model, for example also allowing rescheduling of the server at times other than repair completion times. (The repairman is governed by a Markov Decision Arrival Process, see section 3.) Again, the SGP maximizes the expected time that the system is up.

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 Altman & Nain [1] and Kuri & Kumar [6] for recent results on routing models with delayed information. Our model differs from this 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 [7] 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 2 and 3 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 2, without the early decisions, has already been studied by Katehakis & Melolidakis [5].

2. 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 a Markov Arrival Process (MAP).

2.1. Definition. (Markov Arrival Process) Let Λ be the finite state space of a Markov process with transition intensities λ_{xy} with $x, y \in \Lambda$. When this process moves from x to y, with probability q_{xy} an arrival occurs.

Thus an arrival corresponds to the completion of a repair. It can be shown that with this type of arrival process we can approximate any arrival process arbitrarily close (Asmussen & Koole [2]). Without restricting generality we can assume that $\sum_y \lambda_{xy} = \gamma$ for all x.

We assume that B, the maximum number of functioning components in each group, is finite. The state is denoted with (l, x, i), where l is the group the repairman is working on, x is the state of the arrival process and $i = (i_1, \ldots, i_m)$ denotes the number of functioning components in each group. Because we can only assign the repairman to a group in which not all components are functioning, we assume that $i_l < B$. We use the following notation: $e = (1, \ldots, 1)$, $e_j = (0, \ldots, 0, 1, 0, \ldots, 0)$ with the 1 in jth position, $e_0 = (0, \ldots, 0)$. 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:

$$\begin{aligned} v^{n+1}_{(l,x,i)} = \sum_y \lambda_{xy} \Big(q_{xy} \max_j \{ v^n_{(j,y,i+e_l)} \} + (1 - q_{xy}) v^n_{(l,y,i)} \Big) + \\ \mu \sum_{j=1}^{m} i_j v^n_{(l,x,i-e_j)} + (1 - \gamma - (i_1 + \cdots + i_m)\mu) v^n_{(l,x,i)}. \end{aligned} \tag{2.1}$$

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.

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

2.2. Lemma. *If*

$$v^n_{(l_1,x,i)} \geq v^n_{(l_2,x,i)} \quad \text{for } i_{l_1} \leq i_{l_2},\ i + e_{l_1} + e_{l_2} \leq Be, \tag{2.2}$$

$$v^n_{(l,x,i+e_{j_1})} \geq v^n_{(l,x,i+e_{j_2})} \quad \text{for } i_{j_1} \leq i_{j_2},\ i + e_{j_1} + e_{j_2} + e_l \leq Be, \tag{2.3}$$

and

$$v^n_{(l,x,i)} = v^n_{(l^*,x,i^*)} \quad \text{for } i^* \text{ a permutation of } i,\ i^*_{l^*} = i_l,\ i + e_l \leq Be \tag{2.4}$$

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 (2.2). Assume $i_{l_1} < i_{l_2}$. The case $i_{l_1} = i_{l_2}$ can be done with (2.4). 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. 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 (2.3) 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 (2.3) can be used.

For proving (2.3) we can also assume that $i_{j_1} < i_{j_2}$. The case $i_{j_1} = i_{j_2}$ can be done again with (2.4). 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 (2.2) 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 (2.3).

Equation (2.4) follows easily. □

From (2.2) it follows that the repairman should be assigned to the group with the smallest number of functioning components, that is, according to the SGP. Thus the lemma gives conditions on v^0, the reward function, for the SGP to be optimal. It is interesting to note that for the model without delay in the assignment the buffer size can depend on the group. Observe that 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.

2.3. Theorem. *The SGP maximizes the rewards at T (from 0 to T) for all reward functions satisfying (2.2) to (2.4).*

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 much 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. These results were, for the model without early decisions, also obtained by Katehakis & Melolidakis [5].

3. 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. Besides the way in which components can fail, this model differs from the warm stand-by model in the operation of the repairman, as we will see when the arrival process is introduced.

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 $1, 2, \ldots$. Further assume that in the starting state group 2 has no functioning components and that T is integer. Then it is clear that the repairman should only be assigned to the second group at $T - 1$. However, if we look at the expected time the system is up from 0 to T the SGP is optimal. To show this, we have to introduce immediate rewards. We prefer to collect all rewards together at T. Therefore we will add an extra component to the state space, which is raised by 1 each time a component fails.

But first we generalize the MAP (which governs the lengths of the repair periods) to a Markov Decision Arrival Process (MDAP). We will use arrivals in 2 classes of customers, one to model repair completions, the other to model possible rescheduling of the repairman.

3.1. 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.

Before proceeding with the dynamic programming equation and a lemma similar to lemma

2.2, we give an example of such an MDAP. Suppose the 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 result below does not state anything on when to interrupt the repairman; it just states then when the repairman is operated optimally, then it should always be assigned according to the SGP.

Assume, by uniformization, that $\gamma + m\mu \leq 1$. The dynamic programming equation is:

$$v^{n+1}_{(l,x,i,k)} = \begin{cases} \max_a \Big\{ \sum_y \lambda_{xay} \Big(q^1_{xay} \max_j \{ v^n_{(j,y,i+e_l,k)} \} + q^2_{xay} \max_j \{ v^n_{(j,y,i,k)} \} + \\ \qquad (1 - q^1_{xay} - q^2_{xay}) v^n_{(l,y,i,k)} \Big) \Big\} + \\ \quad \mu \sum_{j=1}^m v^n_{(l,x,i-e_j,k+1)} + (1-\gamma-m\mu) v^n_{(l,x,i,k)} & \text{if } i_j > 0 \text{ for all } j, \\ \max_a \Big\{ \sum_y \lambda_{xay} \Big(q^1_{xay} \max_j \{ v^n_{(j,y,i+e_l,k)} \} + q^2_{xay} \max_j \{ v^n_{(j,y,i,k)} \} + \\ \qquad (1 - q^1_{xay} - q^2_{xay}) v^n_{(l,y,i,k)} \Big) \Big\} + \\ \quad (1-\gamma) v^n_{(l,x,i,k)} & \text{if } i_j = 0 \text{ for some } j. \end{cases} \tag{3.1}$$

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.

3.2. Lemma. *If*

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

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

$$\sum_{j_1=1}^m v^n_{(l,x,i-e_{j_1},k+1)} \geq m v^n_{(l,x,i,k)} \qquad \text{for } i \geq e,\ i + e_l \leq Be, \tag{3.4}$$

$$v^n_{(l,x,i,k)} \geq v^n_{(0,x,i,k)} \qquad \text{for } i + e_l \leq Be, \tag{3.5}$$

$$v^n_{(l,x,i+e_{j_1},k)} \geq v^n_{(l,x,i,k)} \qquad \text{for } i + e_{j_1} + e_l \leq Be, \tag{3.6}$$

$$v^n_{(0,x,i+e_{j_1},k)} \geq v^n_{(j_1,x,i,k)} \qquad \text{for } i + e_{j_1} \leq Be, \tag{3.7}$$

$$v^n_{(l,x,i,k+1)} \geq v^n_{(l,x,i,k)} \tag{3.8}$$

and

$$v^n_{(l,x,i,k)} = v^n_{(l^*,x,i^*,k)} \quad \textit{for } i^* \textit{ a permutation of } i,\ i^*_{l^*} = i_l,\ i + e_l \le Be \tag{3.9}$$

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

Proof. The proof is similar to that of lemma 2.2. The MDAP complicates the analysis somewhat. See the proof of theorem 3.1 in [4] for details.

For the proof of (3.2) we refer to that of (2.2). (Note that the SGP is optimal not only because of (3.2), but also because of (3.5).) Also (3.3) goes follows easily unless $i_{j_1} = 0$ and $i_{j_2} > 0$. Then, for the failure terms, (3.4) should be used first with $i + e_{j_1}$ instead of i. Concerning (3.4), 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 $i \ge Be$. Equation (3.5) gives that idling is suboptimal and was needed in the proof of (3.4), in case $i = Be$. Its proof is easy, using (3.6) in the repair terms. In the proof of (3.6) we need (3.7) (which proof also follows easily) in case $i + e_{j_1} = Be$. Equations (3.8) (which is needed to prove (3.6) and (3.7)) and (3.9) 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 (3.5) gives that idling is suboptimal.

3.3. 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 & P. Nain (1992). Closed-loop control with delayed information. *Performance Evaluation Review* **20**: 193–204.

[2] S. Asmussen & G. Koole (1991). Marked point processes as limits of Markovian arrival streams. To appear in *Journal of Applied Probability*.

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

[4] A. Hordijk & G. Koole (1992). On the assignment of customers to parallel queues. *Probability in the Engineering and Informational Sciences* **6**: 495–511.

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

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

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