

ON THE ASSIGNMENT OF CUSTOMERS TO PARALLEL QUEUES

Arie Hordijk & Ger Koole

*Dept. of Mathematics and Computer Science, University of Leiden
P.O. Box 9512, 2300 RA Leiden, the Netherlands*

ABSTRACT

This paper considers routing to parallel queues in which each queue has its own single server, and service times are exponential with nonidentical parameters. We give conditions on the cost function such that the optimal policy assigns customers to a faster queue when that server has a shorter queue. The queues may have finite buffers, and the arrival process can be controlled and can depend on the state and routing policy. Hence our results on the structure of the optimal policy are also true when the assigning control is in the "last" node of a network of service centers. Using dynamic programming we show that our optimality results are true in distribution.

Published as *Probability in the Engineering and Informational Sciences* **6**:495–511, 1992.

1. INTRODUCTION

This paper considers routing to parallel queues. There are m queues, each queue has its own single server, and service times are exponential. The queues may have a finite capacity, say queue j has capacity B_j , with B_j equal to infinity if the capacity is unlimited. Arriving customers are routed to one of the queues. In our model it is not allowed to reject a customer unless all queues are full. The server at queue j has service intensity μ_j . We assume that the queues are indexed in decreasing order of their server intensities, i.e. $\mu_1 \geq \dots \geq \mu_m$.

Most results on the assignment of customers to parallel queues are derived for a general arrival process. This means that the arrival process may be any stochastic process which is stochastically independent of the numbers of customers in the queues. If we consider a network of queues, for example two nodes in tandem with in each node parallel queues, then it is of interest to study policies for which the assignment decisions in the first node do depend on the numbers of customers in the queues of the second node. In this case the arrival process in the second node and its state are not independent any longer. In order

to study assignment problems with this type of dependent arrival processes we introduce a Markov Decision Arrival Process (MDAP). It is a generalization of the Markov Arrival Process (MAP) which we used in Hordijk & Koole [6] and [7]. The MDAP is a controlled Markov Process that generates customers with intensities which do not depend on the state of the queues. However, actions can be taken in the MDAP. The control, consisting of an action in the MDAP and the selection of the queue to which the next arrival is routed, may depend on the whole state of the system. This induces the stochastic dependence of the arrival process on the number of customers in the parallel queues.

The main result of this paper gives that an arriving customer should be assigned to the queue with a faster server when that server has a shorter queue. We call a policy of this type: Shorter Queue Faster Server Policy (SQFSP).

It is surprising that in the assignment of customers to parallel queues the optimal policy often dominates all other policies in stochastic sense. For example, if all queues have an exponential server with the same rate then the shortest queue policy minimizes the number of customers at any time point not only in expectation but also in distribution. The standard proof methods for the stronger optimality criterion, dominance in stochastic sense, are the coupling method (cf. Ross [19] and Walrand [26]) and the analytic method (cf. Stoyan [24] and Menich & Serfozo [14]). For finding the optimal policy with respect to expected costs one usually applies dynamic programming (dp). In this paper we use dp to show dominance in expectation as well as in stochastic sense. This gives not only a unification in method but also provides insight when the stronger criterion of stochastic dominance holds.

In Hordijk & Koole [9] we apply our method to the counterpart model in which the server(s) are assigned to the queues. It will there become clear why stochastic dominance is exceptional in that case.

In this paper we study a continuous-time model. However, we use the uniformization method and our main theorem in section 3 is shown for the discrete-time stochastic dynamic recursion. Through continuity arguments our results extend to the continuous-time model.

In section 2 we introduce our Markov Decision Arrival Process and we derive the discrete-time dynamic programming recursion. Also in section 2 we show how we can apply our results to the last node of a tandem system. Similar results are true for a labelled node in a general network as long as there is no feedback from the labelled node

to the network. The assignment decisions of the arriving customers to the parallel queues in the labelled node may depend on the complete information of all the nodes in the network. Also the routing and or service control in the other nodes may depend on the numbers of customers in the queues of the labelled node, and the cost functions at different nodes may be different. Our main theorem gives that in the class of policies which depend on the information of the whole network there exists an optimal policy such that the decisions in the labelled node are of SQFSP-type.

Incidentally, the analogous result when servers are assigned to queues is not true. The μc -rule is not optimal in the second node of the tandem queue (see Hordijk & Koole [8]).

In section 3 we derive our main theorem which contains a set of three inequalities (3.1), (3.2) and (3.3). Using dp we show that if the cost function v^0 satisfies this set of inequalities then so does v^n , the minimal expected cost at time n , for all n .

The inequality (3.1) gives that in state $i = (i_1, \dots, i_m)$, with i_j the number of customers in queue j , and $i_{j_1} \leq i_{j_2}$, with $j_1 < j_2$, $i_{j_1} < B_1$ and $i_{j_2} < B_2$, it is better to assign the customer to queue j_1 than to queue j_2 , regardless of the action in the arrival process. Note that the server of queue j_1 is faster. This inequality determines the optimal actions in a subset of the state space. It says that an arriving customer should be assigned to the queue with a faster server when that server has a shorter queue. Hence it is of SQFSP-type. Note that the inequality (3.1) does not specify the optimal assignment policy completely.

The inequality (3.2) says that a state with less customers is more favourable.

The inequality (3.3) shows that a state with the larger number of customers in the queue with the faster server, is better.

It turns out that the indicator function of the cost exceeding a specified value, does satisfy this type of inequalities. This is the reason why dominance in stochastic sense follows from our results.

A policy of the SQFSP-type is completely determined in the two special cases of section 4. In the first of the special cases there is no waiting room, i.e. we assume $B = (B_1, \dots, B_m) = (1, \dots, 1)$. Hence the class of SQFSP policies contains only one policy which assigns an arriving customer to the fastest free server. Let us call this policy, Fastest Server Policy (FSP). The FSP is optimal in expectation and in stochastic sense for a large class of cost functions, for any time horizon.

In the second special case, the symmetric case, we assume symmetric costs and equal service rates. Here the optimal policy assigns arriving customers to the shortest nonfull

queue. The Shortest Queue Policy (SQP) is optimal in expectation and in distribution. In this paper we allow that queues have a finite buffer. Note that the symmetric case may be asymmetric in the buffer sizes.

In section 5 we analyze the type of cost functions which satisfy the set of inequalities of the main theorem. In the two special cases we obtain a complete characterization. In the symmetric case it is the class of weak Schur convex functions. For the no waiting room case it is a closely related type of cost function.

Finally the proof of main theorem is given in the Appendix.

For the no waiting room case it is shown in Derman et al. [3] that the FSP is optimal for an arbitrary arrival process and the number of customers in the system as cost function. Our theorem is more general in that it allows very general cost functions for one node. Moreover, the node can be a labelled node of a network with no feedback from the node to the network. It follows from our results that there is an optimal policy which is FSP in the labelled node.

In Sobel [22] different customer classes with Poisson arrivals are considered. The paper contains an unrepairable error. In Sobel & Srivastava [23] a less general result is shown, namely the FSP is optimal with respect to the blocking costs. In Koole [11] the blocking cost function may depend on the class type. Also rejection is allowed, even when the system is not full. The counterexample in Seth [20] shows that the assumption of exponential service times can not be relaxed.

The first result for the symmetric case is due to Winston [27]. Hordijk & Koole [6] and Towsley et al. [25] extended Winston's result to a general arrival process and finite buffers. The result in this paper gives the extension to a "last" node in a network. In Hordijk & Koole [7] the optimality of SQP in the first node of the tandem queue is shown for a restricted class of policies. The paper also gives a more extensive review of related literature. It is not surprising that in the general case our main theorem does not completely determine the optimal policy. Indeed, it is very difficult to obtain the optimal policy in this case. Attempts to describe the optimal policy in more detail all failed, due to its complex structure. So far, only computational results have been achieved. They can be found in Van Moorsel & De Vries [15], Nobel & Tijms [16], Houck [10] and Shenker & Weinreb [21], mostly for $m = 2$ and $B_1 = B_2$. Van Moorsel & De Vries [15] and Nobel & Tijms [16] use successive approximation, in the other two papers simulation is used.

Nearly optimal policies are proposed, e.g. the policy that assigns each arriving customer to the queue where its expected delay is minimal. All policies studied in these papers are of SQFSP-type. Nobel & Tijms [16] also consider the case where there are more than one servers at each queue. If the number of servers and the service rates are equal one can prove, with a similar method as our main theorem, that the SQP is optimal. In Menich & Serfozo [14] this result is shown for the case without finite buffers.

2. THE ARRIVAL PROCESS AND THE DYNAMIC PROGRAMMING RECURSION

In this section we introduce the MDAP and we show that the arrival process in the second center of a tandem network is a special case of the MDAP. The MDAP is a generalization of the Markov Arrival Process (MAP) which we used in Hordijk & Koole [6] and [7]. In Asmussen & Koole [1] it is shown that any independent arrival process can be approximated arbitrarily closely by a MAP. From this result it follows that any dependent arrival process is the weak limit of a sequence of MDAPs.

In the dynamic programming recursion we use the uniformization method. We consider the jump epochs of a Poisson process with parameter γ , and we define $v_{(x,i)}^n$ as the minimal expected cost at the n th jump epoch of the Poisson process, given the starting state is (x, i) , where x is the state of the MDAP and i is the vector with components the numbers of customers in the queues. The cost function is denoted by v^0 . Using continuity arguments of Hordijk & Van der Duyn Schouten [5] and Van Dijk [4] (cf. chapter 5 in Koole [12]) as γ tends to infinity, our optimality results hold within the class of strongly regular policies in the continuous-time model.

2.1. Definition. (Markov Decision Arrival Process) Let Λ be the, possibly countable, state space of a Markov Decision Process with transition intensities λ_{xay} with $x, y \in \Lambda$ and a the action chosen in state x . $A(x)$ is the finite set of possible actions in x . An arrival occurs with probability q_{xay} when action a is chosen in x and a transition from x to y happens.

We recall that the MAP is the special case of the MDAP for which the λ 's and q 's do not depend on the actions a . If we think of a network of exponential nodes then the state x of the MDAP can be seen as the numbers of customers in all the queues of all nodes except the labelled node. Let $i = (i_1, \dots, i_m)$ give the numbers of customers in the labelled node.

The uniformization method requires that the rates in each state of the MDAP are bounded for each action. We assume that for some γ , $\sum_y \lambda_{xay} + \mu_1 + \dots + \mu_m \leq \gamma$ for all x and a . This is not a very restrictive assumption as most models do satisfy it. In order to have a more clear notation we take without restriction of generality γ equal to 1 in the following dp recursion. We want to find the optimal policy for a finite horizon and we suppose that there is only a cost incurred at the end of the horizon, say $v_{(x,i)}^0$ if the final state is (x, i) . Standard dynamic programming recursion gives for v^n , the minimal expected cost if the time horizon is n :

$$v_{(x,i)}^{n+1} = \sum_{j=1}^m \mu_j v_{(x,i-e_j \vee 0)}^n + \min_a \left\{ \sum_y \lambda_{xay} \left(q_{xay} \min_j \{ v_{(y,i+e_j \wedge B)}^n \} + (1 - q_{xay}) v_{(y,i)}^n \right) \right\}, \quad (2.1)$$

where \vee (\wedge) means componentwise maximum (minimum), and e_j is the vector with the j th component, as the only nonzero component, equal to one.

Let us illustrate now the use of the MDAP by modelling the tandem of two service centers. Assume there are m_1 (m_2) queues in the first (second) center, with state $(i_1^1, \dots, i_{m_1}^1)$ ($(i_1^2, \dots, i_{m_2}^2)$) and service intensities $\mu_1^1 \geq \dots \geq \mu_{m_1}^1$ ($\mu_1^2 \geq \dots \geq \mu_{m_2}^2$). The arrival rate is λ . Then the dp recursion gives:

$$v_{(i^1, i^2)}^{n+1} = \sum_{j=1}^{m_2} \mu_j^2 v_{(i^1, i^2 - e_j \vee 0)}^n + \min_{a_1} \left\{ \lambda v_{(i^1 + e_{a_1} \wedge B, i^2)}^n + \sum_{j=1}^{m_1} \left(\delta_{i_j^1} \mu_j^1 \min_{a_2} \{ v_{(i^1 - e_j, i^2 + e_{a_2} \wedge B)}^n \} + (1 - \delta_{i_j^1}) \mu_j^1 v_{(i^1, i^2)}^n \right) \right\}.$$

Now if we take, $\lambda_{i^1, a, i^1 + e_a \wedge B} = \lambda$, $q_{i^1, a, i^1 + e_a \wedge B} = 0$, $\lambda_{i^1, a, i^1 - e_j} = \mu_j^1$ if $i_j^1 > 0$ and $q_{i^1, a, i^1 - e_j} = 1$ and 0 for all other transition rates then this recursive equation has the form of (2.1) where the decision variables a_1 and a_2 correspond to a and j , respectively.

Similarly as for the tandem queue, we can model any network in which we focus on the assignment of customers to parallel queues in a labelled node as an assignment problem with a MDAP as arrival process. The requirement on the network is that jobs leaving the labelled node do not enter the network. It is not difficult to extend the definition of the arrival process to include feedback to the network of jobs leaving the labelled node. However, our main theorem of the next section is generally not true in that case.

3. THE MAIN RESULT

In this section we consider a set of three conditions for the cost function, the inequalities (3.1), (3.2) and (3.3). We show in our main theorem that if v^0 satisfies this set of inequalities then does v^n , for all n . Let us discuss the set of inequalities of the main theorem. Inequality (3.1) below yields that an arriving customer should be routed to the faster server if this server has a shorter queue. For example if there are two queues with i_1 respectively i_2 customers and $i_1 \leq i_2$ then the customer should be assigned to queue 1. This result gives the structure of the optimal policy. It is of SQFSP-type as stated in corollary 3.3. Note that inequality (3.1) does not determine the optimal decision in all states. If $i_1 > i_2$ then the optimal decision is not specified. In the next section we give two special cases for which the optimal policy is completely determined.

Inequality (3.2) says that a state with less customers is more favourable. In our model we do not allow customers to be rejected if one of the buffers is not full. In Hordijk & Koole [6] we show that the number of served customers is stochastically maximized for the symmetric case. In that case rejecting or blocking customers can be included. A similar result is shown in Koole [12] for the model of this paper. In Koole [11] it is shown that the FSP is optimal in the no waiting room case with the rejection option.

Equation (3.3) says that is better to have the larger number of customers in the queue with the faster server.

3.1. Theorem. *If*

$$v_{(x, i+e_{j_1})} \leq v_{(x, i+e_{j_2})} \text{ for } i_{j_1} \leq i_{j_2}, j_1 \leq j_2, i + e_{j_1} + e_{j_2} \leq B \quad (3.1)$$

$$v_{(x, i)} \leq v_{(x, i+e_j)} \text{ for } i + e_j \leq B \quad (3.2)$$

$$v_{(x, i)} \leq v_{(x, i^*)} \text{ for } i_j^* = \begin{cases} i_j & \text{if } j \neq j_1, j_2 \quad i_{j_1} \vee i_{j_2} \leq B_{j_1} \wedge B_{j_2}, \\ i_{j_2} & \text{if } j = j_1 \\ i_{j_1} & \text{if } j = j_2 \quad i_{j_1} > i_{j_2} \text{ and } j_1 \leq j_2 \end{cases}, \quad (3.3)$$

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

The proof of theorem 3.1 can be found in the appendix.

In case the optimal policy is completely determined by the inequalities of theorem 3.1, it is myopic, i.e. the optimal decision rule at time 1, say f^* , is the same for any time horizon. In the standard dynamic programming problem there is an immediate cost function, say $c_{(x, i)}$. The total cost over time period $t = 1$ to $t = n$ is then $\sum_{t=1}^n c_{(X_t, I_t)}$,

where X_t and I_t are, respectively, the state and the node of the network at time t . If $c_{(x,i)}$ satisfies the set of inequalities then we may take $v^0 = c$ and we find that f^* minimizes $\mathbb{E}c_{(X_t, I_t)}$ for all t . Hence the policy which uses f^* at any decision epoch also minimizes the sum.

In case the optimal policy is not completely determined by the inequalities of theorem 3.1, it is in general not myopic. For the dynamic programming problem with immediate cost $c_{(x,i)}$ we then need an extension of the theorem, which guarantees that v^n satisfies the inequalities if v^0 and the immediate cost vector c do. The result is stated in theorem 3.2, but first we give the dynamic programming recursion with immediate cost vector c :

$$w_{(x,i)}^{n+1} = c_{(x,i)} + \sum_{j=1}^m \mu_j w_{(x, i - e_j \vee 0)}^n + \min_a \left\{ \sum_y \lambda_{xay} \left(q_{xay} \min_j \{ w_{(y, i + e_j \wedge B)}^n \} + (1 - q_{xay}) w_{(y,i)}^n \right) \right\}. \quad (3.4)$$

It is clear from this recursion that if $c_{(x,i)}$ satisfies (3.1), (3.2) and (3.3) then in the recursive proof of theorem 3.1 we can treat the term $c_{(x,i)}$ separately. Hence we have the following theorem:

3.2. Theorem. *If*

$$w_{(x, i + e_{j_1})} \leq w_{(x, i + e_{j_2})} \text{ for } i_{j_1} \leq i_{j_2}, j_1 \leq j_2, i + e_{j_1} + e_{j_2} \leq B \quad (3.5)$$

$$w_{(x,i)} \leq w_{(x, i + e_j)} \text{ for } i + e_j \leq B \quad (3.6)$$

$$w_{(x,i)} \leq w_{(x, i^*)} \text{ for } i_j^* = \begin{cases} i_j & \text{if } j \neq j_1, j_2 \\ i_{j_2} & \text{if } j = j_1 \\ i_{j_1} & \text{if } j = j_2 \end{cases}, \quad \begin{matrix} i_{j_1} \vee i_{j_2} \leq B_{j_1} \wedge B_{j_2}, \\ i_{j_1} > i_{j_2} \text{ and } j_1 \leq j_2 \end{matrix} \quad (3.7)$$

hold for the cost functions c and w^0 , then they hold for w^n , for all n .

In the following corollaries we conclude from this theorem that for the finite time horizon and the infinite time horizon under mild conditions there is an optimal policy of SQFSP-type.

3.3. Corollary. *Consider the total costs over a finite horizon. A policy of SQFSP-type is optimal for all cost functions satisfying (3.5), (3.6) and (3.7).*

If $c_{(x,i)} \geq 0$, for all (x,i) then $w_{(x,i)}^n$ is nondecreasing in n , for all n . Hence the limit, say $w_{(x,i)}$ does exist for all (x,i) . It is easily seen that the limit also exists if the immediate

cost is bounded from below and the discount factor is strictly smaller than one. From (3.6) it follows that $c_{(x,i)}$ is bounded from below if $\inf_x c_{(x,0)} > -\infty$. Clearly if w^n satisfies the set of inequalities for all n then also the limit w . Now applying a standard result in negative dynamic programming (cf. Theorem 1.2 in Ross [18]) we get the following corollary:

3.4. Corollary. *Consider the discounted costs for a discount factor $\alpha \in [0, 1)$. A policy of SQFSP-type is optimal for all cost functions satisfying (3.5), (3.6), (3.7) and $\inf_x c_{(x,0)} > -\infty$.*

4. SPECIAL CASES AND STOCHASTIC DOMINANCE

Let us assume that v^0 satisfies (3.1), (3.2) and (3.3). From the main theorem it then follows that v^n satisfies (3.1). This means that the optimal assignment decision is known for all states such that the fastest server has less customers than the other servers. For the general case it is very difficult to specify the optimal policy completely. However, there are two interesting cases for which (3.1) gives the optimal policy for all states. It is in these cases that also the stronger optimality result on stochastic dominance follows from the main theorem. The special cases are $B = e$ (e is the vector with all components equal to 1) i.e. there are no waiting places in the buffers, and the symmetric case for which $\mu_1 = \dots = \mu_m$ and the final cost vector is symmetric, i.e. $v_{(x,i)}^0 = v_{(x,i^*)}^0$ if i^* is a permutation of i .

Before we will discuss stochastic dominance, these special cases will be treated in more detail.

4.1. *No waiting room.*

In the case $B = e$ the inequalities of theorem 3.1 simplify to:

$$v_{(x,i+e_{j_1})} \leq v_{(x,i+e_{j_2})} \text{ if } 0 \leq i \leq e, i_{j_1} = i_{j_2} = 0 \text{ and } j_1 < j_2, \quad (4.1)$$

$$v_{(x,i)} \leq v_{(x,i+e_j)} \text{ if } 0 \leq i \leq e \text{ and } i_j = 0. \quad (4.2)$$

It is clear from (4.1) that the optimal assignment decision is to route an arriving customer to the fastest free server. This policy is called the Fastest Server Policy (FSP). Note that our main theorem does not give the optimal actions in the arrival process. On the other hand if we assume the inequalities (4.1) and (4.2) for immediate cost vector $c_{(x,i)}$ there is no restriction on its dependence on x . Only the dependence on i is relevant. Applying this result to a network of service centers with a "last" node which has no feedback to the network gives:

4.1. Corollary. *The FSP is optimal in the "last" node of a network when this node has no waiting room and its cost vector satisfies (4.1) and (4.2).*

4.2. *The symmetric case.*

In the symmetric case we assume $\mu_1 = \dots = \mu_m$ and a symmetric final cost vector, i.e. $v_{(x,i)}^0 = v_{(x,i^*)}^0$ if i^* is a permutation of i . Note that the symmetric case may still be asymmetric in the buffer sizes. The set of inequalities can be simplified in the symmetric case. In order to do this we need the following lemma:

4.2. Lemma. *Assume in the symmetric case buffer sizes B and let \tilde{B} be a permutation of B , say $\tilde{B} = B\Pi$. Let v^n respectively \tilde{v}^n be the solution of the dp recursion for buffer sizes B respectively \tilde{B} . Then $v_{(x,i)}^n = \tilde{v}_{(x,\tilde{i})}^n$ with $\tilde{i} = i\Pi$ for all n , $x \in \Gamma$ and $i \leq B$.*

Proof. With induction. □

With this lemma 4.2 we can rewrite the set of inequalities of theorem 3.1:

$$v_{(x,i+e_{j_1})} \leq v_{(x,i+e_{j_2})} \text{ if } i_{j_1} \leq i_{j_2}, i + e_{j_1} + e_{j_2} \leq B \quad (4.3)$$

$$v_{(x,i)} \leq v_{(x,i+e_j)} \text{ if } i + e_j \leq B \quad (4.4)$$

$$v_{(x,i)} = v_{(x,i^*)} \text{ if } i_j^* = \begin{cases} i_j & \text{if } j \neq j_1, j_2 \\ i_{j_2} & \text{if } j = j_1 \\ i_{j_1} & \text{if } j = j_2 \end{cases}, i_{j_1} \vee i_{j_2} \leq B_{j_1} \wedge B_{j_2} \quad (4.5)$$

Indeed, (4.3) follows from (3.1) by indexing the nonfull queues such that the numbers of customers are increasing with the index of the queue; (4.4) is equal to (3.2) and the symmetry relation (4.5) follows by interchanging the numbers of customers in the queues j_1 and j_2 in (3.3). The inequality (4.3) gives that an arriving customer should be routed to the shortest queue. From the equality (4.5) it follows that ties may be broken in an arbitrary way if there two or more shortest queues. This is the shortest queue policy (SQP). It is well known that SQP is optimal for one node with a general arrival process and finite or infinite buffer sizes (cf. Hordijk & Koole [6]). This result also follows from the main theorem of this paper if we take as arrival process a MAP, which is a special case of a MDAP. For a network of service centers with a "last" node which has no feedback to the network we use the MDAP and find:

4.3. Corollary. *The SQP is optimal in the "last" node of a network when this node is symmetric and its cost function satisfies (4.3), (4.4) and (4.5) .*

4.3. Stochastic dominance.

Let us assume in this subsection that the set of inequalities of the main theorem completely determines an optimal assignment policy, say R_0 . The inequalities are true for all states of the MDAP. Therefore they can never specify optimal actions for the arrival process. So we assume in this section that for any time horizon and any cost function $c_{(x,i)}$ satisfying (3.1), (3.2) and (3.3) there exists an optimal policy such that the assignment decisions are given by R_0 . The optimal policy of the arrival process may depend on the time horizon and the cost function. Note that the assumption of this subsection is valid for the special cases of the preceding subsections. Clearly, under our assumption the optimal assignment policy R_0 is myopic.

It is easily verified that if $c_{(x,i)}$ satisfies the set (3.1), (3.2) and (3.3) then the indicator function $\mathbf{I}_{\{c_{(x,i)} > s\}}$ satisfies them as well. Hence R_0 is an optimal assignment policy for the cost function $c_{(x,i)}$ and for all the cost functions $\mathbf{I}_{\{c_{(x,i)} > s\}}$, for all s .

Using that the probability of an event is the expectation of its indicator function we find:

4.4. Theorem. *If the set of inequalities of the main theorem uniquely determines an*

assignment policy, say R_0 , and cost function $c(x, i)$ satisfies this set then R_0 minimizes

$$\mathbb{E}(c_{(X_t, I_t)}) \quad \text{and} \quad \mathbb{P}(c_{(X_t, I_t)} > s_t) \quad \forall t, s_t$$

This theorem then gives that R_0 not only minimizes the marginal costs at time t in expectation but also in distribution.

Note however that the theorem does not guarantee that expressions like

$$\mathbb{P}(c_{(X_1, I_1)} > s_1, \dots, c_{(X_n, I_n)} > s_n)$$

are minimized by R_0 . In this case, a multidimensional version of the set of inequalities of the main theorem must be used. We have not pursued this any further.

5. COST FUNCTIONS

In this section we characterize the cost functions which satisfy the set of inequalities (3.1), (3.2) and (3.3) for the symmetric case and in the case of no waiting room. In the general case we did not succeed in doing so. However, for the class of additive cost functions we have a sufficient condition.

Let us first consider the cost function, $\mathbb{I}_{\{(x, i) \neq 0\}}$ which trivially satisfies the inequalities of the main theorem. Then v^n , with this indicator function as v^0 , gives the probability that the "system" is not empty at time n . Consider again a network with a labelled node which has no feedback to the network. Suppose there are no arrivals to the network, then the probability that the "system" is not empty at time n means that the makespan is at least n . In this case it follows from our main theorem that for any n , there is a policy of SQFSP-type which minimizes the probability that the makespan is at least n . Note that for different n , the optimal policy may be different.

When the main theorem completely determines the assignment policy (e.g. in the special cases of section 4), then this policy is the same for any time horizon and hence it minimizes the makespan in stochastic sense. Note that the optimal control policy of the arrival process may, and in general will depend on the time horizon, also in the special cases.

In this section we will use c instead of v^0 . Since the set of inequalities has the same x in all inequalities, the dependence of the cost structure on the state of the MDAP may be arbitrarily defined. Therefore we consider cost functions c in this section which only depend on i .

5.1. The general case, additive costs.

We consider in this subsection the nonsymmetric additive functions, i.e. $c_{(x,i)} = f_1(i_1) + \dots + f_m(i_m)$. Define $\Delta f_j(i) = f_j(i+1) - f_j(i)$. Then the following conditions are sufficient: f_j increasing, $\Delta f_1(i) \leq \dots \leq \Delta f_m(i)$ for all i (note that this condition in the terminology of Pinedo & Rammouz [17] requires that f_{j+1} is steeper than f_j) and $m - 1$ of the m functions are convex. Let us show that these conditions imply (3.1). Suppose $j_1 < j_2$ and $i_{j_1} \leq i_{j_2}$. When f_{j_1} is convex we have that $\Delta f_{j_1}(i_{j_1}) \leq \Delta f_{j_1}(i_{j_2})$. The fact that f_{j_2} is steeper than f_{j_1} gives $\Delta f_{j_1}(i_{j_2}) \leq \Delta f_{j_2}(i_{j_2})$. These inequalities together give (3.1). Equation (3.2) is immediate. To prove (3.3) we use that f_{j_2} is steeper than f_{j_1} . Hence $f_{j_1}(i_{j_1}) - f_{j_1}(i_{j_2}) \leq f_{j_2}(i_{j_1}) - f_{j_2}(i_{j_2})$, which gives (3.3). Consequently, the main theorem implies that these conditions are sufficient for the existence of an optimal policy of SQFSP-type.

5.2. The case $B = e$.

Define a partial ordering \prec as follows:

$i \prec i^*$ if there are i^0, \dots, i^n with $i^0 = i$ and $i^n = i^*$ such that

$$\begin{aligned} i^k &= i^{k-1} - e_{j_1} + e_{j_2}, \quad i_{j_1}^{k-1} = 1, i_{j_2}^{k-1} = 0, j_1 < j_2, \text{ or} \\ i^k &= i^{k-1} + e_j, \quad i_j^{k-1} = 0, k = 1, \dots, n. \end{aligned} \tag{5.1}$$

It is easily seen that the class of cost functions satisfying (4.1) and (4.2) is precisely the class of functions preserving the ordering \prec .

We will show that the ordering \prec is equivalent to an ordering \prec' , defined by:

$i \prec' i^*$ if $\sum_{j=k}^m i_j \leq \sum_{j=k}^m i_j^*$ for $k = 1, \dots, m$.

5.1. Theorem. *The orderings \prec and \prec' are equivalent.*

Proof. $i \prec i^* \Rightarrow i \prec' i^*$. Take $i = i^0, \dots, i^n = i^*$ as in (5.1). It is easy to see that $i^{k-1} \prec' i^k$ for $k = 1, \dots, n$. Due to the transitivity of \prec' we have $i \prec' i^*$.

$i \prec' i^* \Rightarrow i \prec i^*$. We will construct $i = i^0, \dots, i^n = i^*$ such that $i^0 \prec \dots \prec i^n$. First add $\sum_{j=1}^m i_j^* - \sum_{j=1}^m i_j$ customers to the free servers with smallest indices and call this state i^1 . Clearly, if $i^1 = i^*$, we are ready. If not, construct $i^2 = i^1 - e_{j_1} + e_{j_2}$, with j_1 such that $i_{j_1}^1 = 1$ and $i_{j_1}^* = 0$ and j_2 such that $i_{j_2}^1 = 0$ and $i_{j_2}^* = 1$. Since $\sum_{j=k}^m i_j^1 \leq \sum_{j=k}^m i_j^*$ for $k = 1, \dots, m$ we can choose j_1 and j_2 such that $j_1 < j_2$. Repeat this construction until we have $i^n = i^*$.

Then $i^1 \prec \dots \prec i^n$ and transitivity gives $i \prec i^*$. \square

Let us look at the \prec' -preserving functions. Allowed cost functions are $c_i = \sum_{j=k}^m i_j$, for all k , the total number of customers in the $m - k$ queues with slowest servers. Hence the FSP minimizes the total number of customers in the $m - k$ queues with slowest servers stochastically for all $k = 1, \dots, m$. It is clear that there are other interesting \prec' -preserving functions, e.g. the weighted total number of customers with agreeable weights.

5.3. The symmetric case.

As in the case $B = e$ the inequalities (4.3), (4.4) and the equality (4.5) define an ordering \prec with $i \prec i^*$ if there are $i^0, \dots, i^n, i^0 = i$ and $i^n = i^*$ such that

$$\begin{aligned} i^k &= i^{k-1} - e_{j_1} + e_{j_2}, & 0 < i_{j_1}^{k-1} < i_{j_2}^{k-1}, i^{k-1} + e_{j_2} \leq B \text{ or} \\ i^k &= i^{k-1} + e_j, & i^k \leq B \text{ or} \\ i^k &\text{ is a permutation of } i^{k-1} \text{ such that } i^{k-1}, i^k \leq B \end{aligned} \quad (5.2)$$

Now consider the *weak majorization ordering* \prec_w (see Marshall & Olkin [13]). $i \prec_w i^*$ if $\sum_{j=1}^k i_{[j]} \leq \sum_{j=1}^k i_{[j]}^*$ for all k , with $i_{[1]} \geq \dots \geq i_{[m]}$ the decreasing rearrangement of i . Thus, the sum of the k th largest components of i is smaller than that of i^* .

5.2. Theorem. *The orderings \prec and \prec_w are equivalent.*

Proof. $i \prec i^* \Rightarrow i \prec_w i^*$. Take i^0, \dots, i^n as in (5.2). It is easy to see that $i^{k-1} \prec_w i^k$ for all k . As \prec_w is a preordering transitivity holds and $i \prec_w i^*$.

$i \prec_w i^* \Rightarrow i \prec i^*$. We will construct $i = i^0, \dots, i^n = i^*$ such that $i^0 \prec \dots \prec i^n$.

Take the largest component of i^0 , say queue j_1 , and interchange it with the component of i^0 with the index of the longest queue of i^* , say queue j_2 . Call the resulting vector i^1 . Then, as $i_{[1]} \leq i_{[1]}^*$, $i_{[1]}$ fits in the buffer of queue j_2 . Because $i_{j_1} \geq i_{j_2}$, i_{j_2} fits in the buffer of queue j_1 , thus $i^1 \leq B$. We have by symmetry $i^0 \prec i^1$ and trivially $i^1 \prec_w i^*$.

The second step is to transfer a customer from $i_{[2]}^1$ to $i_{[1]}^1$. We assume $i_{[2]}^1 > 0$, the case $i_{[2]}^1 = 0$ is easy. Call the resulting vector i^2 . By the definition (5.2) we have $i^1 \prec i^2$. To show $i^2 \prec_w i^*$ we distinguish the following 2 cases.

Case 1, $i_{[2]}^1 > i_{[3]}^1$, then $i_{[1]}^1 + i_{[2]}^1 = i_{[1]}^2 + i_{[2]}^2$ and $i^2 \prec_w i^*$ follows immediately.

Case 2, $i_{[2]}^1 = \dots = i_{[k]}^1 > i_{[k+1]}^1$ then $i_{[1]}^2 + \dots + i_{[l]}^2 = 1 + i_{[1]}^1 + \dots + i_{[l]}^1$, $l < k$. However, it is straightforward to see $i_{[1]}^1 + \dots + i_{[l]}^1 < i_{[1]}^* + \dots + i_{[l]}^*$. Thus $i^2 \prec_w i^*$.

Repeating this we find a i^{n_1} with either $i_{[2]}^{n_1} = 0$ or $i_{[1]}^{n_1} = i_{[1]}^*$. The first case is easy. In the second case, omit the longest component of i^{n_1} and i^* and repeat the same arguments, etc. □

The equivalence of theorem 5.2 gives that the class of functions satisfying (4.3), (4.4) and (4.5) is precisely the class of functions preserving weak majorization. These functions are called the *weak Schur convex functions*, cf. Marshall & Olkin [13]. A continuous version of theorem 5.2 without the constraint of finite buffers can be found in [13]. Define $\Delta_j c_i = c_{i+e_j} - c_i$, then (4.3) and (4.4) yield $0 \leq \Delta_{j_1} c_i \leq \Delta_{j_2} c_i$ if $i_{j_1} \leq i_{j_2}$. This is, together with the symmetry, the discrete counterpart of the weak majorization version of theorem 3.A.4 of Marshall & Olkin [13].

Note that the class of functions given by Daley [2], the increasing convex symmetric functions, is strictly smaller than the class of weak Schur convex functions. For example the indicator functions of cost functions satisfying (4.3), (4.4) and (4.5) are in general not convex. However, as is easily verified these indicator functions are weak Schur convex.

Acknowledgement. A part of the research for this paper was done while the first author was on sabbatical leave and the second author was visitor at the Department of Operations Research of the University of North Carolina at Chapel Hill. The authors would like to thank V. Kulkarni, R. Nobel, F. Spieksma and S. Stidham for stimulating discussions. We are grateful to Rhonda Righter and an anonymous referee, whose comments improved the presentation of this paper.

REFERENCES

- [1] S. Asmussen & G. Koole (1991). Marked point processes as limits of Markovian arrival streams. To appear in *Journal of Applied Probability*.
- [2] D.J. Daley (1987). Certain optimality properties of the first-come first-served discipline for $G|G|s$ queues. *Stochastic Processes and their Applications* **25**: 301–308.
- [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] N.M. van Dijk (1988). On the finite horizon Bellman equation for controlled Markov jump models with unbounded characteristics: existence and approximations. *Stochastic Processes and their Applications* **28**: 141–157.
- [5] A. Hordijk & F. van der Duyn Schouten (1985). Markov decision drift processes; conditions for optimality obtained by discretization. *Mathematics of Operations Research* **10**: 160–173.
- [6] A. Hordijk & G. Koole (1990). On the optimality of the generalized shortest queue policy. *Probability in the Engineering and Informational Sciences* **4**: 477–487.
- [7] A. Hordijk & G. Koole (1992). On the shortest queue policy for the tandem parallel queue. *Probability in the Engineering and Informational Sciences* **6**: 63–79.
- [8] A. Hordijk & G. Koole (1992). The μc -rule is not optimal in the second node of the tandem queue: a counterexample. *Advances in Applied Probability* **24**: 234–237.
- [9] A. Hordijk & G. Koole (1992). On the optimality of LEPT and μc rules for parallel processors and dependent arrival processes. Technical report TW-92-01, University of Leiden. To appear in *Advances in Applied Probability*, Dec. 1993.
- [10] D.J. Houck (1987). Comparison of policies for routing customers to parallel queueing systems. *Operations Research* **35**: 306–310.
- [11] G. Koole (1991). Assigning multiple customer classes to parallel servers. Technical report TW-91-07, University of Leiden.
- [12] G. Koole (1992). Stochastic scheduling and dynamic programming. Ph.D. thesis, University of Leiden. (on request available from the author)
- [13] A.W. Marshall & I. Olkin (1979). *Inequalities: Theory of Majorization and its Applications*. Academic Press, New York.
- [14] R. Menich & R.F. Serfozo (1991). Monotonicity and optimality of symmetric parallel processing systems. *Queueing Systems* **9**: 403–418.
- [15] A. van Moorsel & E. de Vries (1989). Optimale toewijzingsstrategieën bij eenvoudige

- netwerken van wachtrijen. Master thesis, University of Leiden.
- [16] R. Nobel & H.C. Tijms (1990). Optimal routing of customers to parallel service groups. Memorandum 90-47, Vrije Universiteit Amsterdam.
 - [17] M. Pinedo & E. Rammouz (1988). A note on stochastic scheduling on a single machine subject to breakdown and repair. *Probability in the Engineering and Informational Sciences* **2**: 41–49.
 - [18] S.M. Ross (1983). *Introduction to Stochastic Dynamic Programming*. Academic Press, New York.
 - [19] S.M. Ross (1983). *Stochastic Processes*. Wiley, New York.
 - [20] K. Seth (1977). Optimal service policies, just after idle periods, in two-server heterogeneous queueing systems. *Operations Research* **25**: 356–360.
 - [21] S. Shenker & A. Weinrib (1989). The optimal control of heterogeneous queueing systems: a paradigm for load-sharing and routing. *IEEE Transactions on Computers* **38**: 1724–1735.
 - [22] M.J. Sobel (1990). Throughput maximization in a loss queueing system with heterogeneous servers. *Journal of Applied Probability* **27**: 693–700.
 - [23] M.J. Sobel & C. Srivastava (1991). Full-service policy optimality with heterogeneous servers. Working paper.
 - [25] D. Towsley, P.D. Sparaggis & C.G. Cassandras (1992). Optimal routing and buffer allocation for a class of finite capacity queueing systems. *IEEE Transactions on Automatic Control* **37**: 1446–1451.
 - [26] J. Walrand (1988). *An Introduction to Queueing Networks*. Prentice-Hall, Englewood Cliffs.
 - [27] W. Winston (1977). Optimality of the shortest line discipline. *Journal of Applied Probability* **14**: 181–189.

APPENDIX

Proof of theorem 3.1. Assume the inequalities hold for v^n . We prove the inequalities for each of the m terms corresponding to a departure at queue $j = 1, \dots, m$ and the term for a arrival of v^{n+1} separately. (Refer to equation (3.4).) By summing these terms we get the desired inequalities. We start with (3.1). Consider a departure at queue j , $j \neq j_1, j_2$:

$$\mu_j v_{(x, i+e_{j_1}-e_j \vee 0)}^n \stackrel{(3.1)}{\leq} \mu_j v_{(x, i+e_{j_2}-e_j \vee 0)}^n.$$

The terms corresponding to a departure from queue j_1 and j_2 will be considered together. Consider the following cases:

$$\begin{aligned} \underline{0 = i_{j_1} < i_{j_2}} : \mu_{j_1} v_{(x, i)}^n + \mu_{j_2} v_{(x, i+e_{j_1}-e_{j_2})}^n &\stackrel{(3.2)}{\leq} \mu_{j_1} v_{(x, i+e_{j_2})}^n + \mu_{j_2} v_{(x, i+e_{j_1}-e_{j_2})}^n \\ &\stackrel{(3.1)}{\leq} \mu_{j_1} v_{(x, i+e_{j_2})}^n + \mu_{j_2} v_{(x, i)}^n \end{aligned}$$

$$\begin{aligned} \underline{0 < i_{j_1} = i_{j_2}} : \mu_{j_1} v_{(x, i)}^n + \mu_{j_2} v_{(x, i+e_{j_1}-e_{j_2})}^n &\stackrel{(3.1)}{\leq} \\ \mu_{j_2} v_{(x, i)}^n + (\mu_{j_1} - \mu_{j_2}) v_{(x, i+e_{j_2}-e_{j_1})}^n + \mu_{j_2} v_{(x, i+e_{j_1}-e_{j_2})}^n &\leq \\ \mu_{j_1} v_{(x, i+e_{j_2}-e_{j_1})}^n + \mu_{j_2} v_{(x, i)}^n, & \end{aligned}$$

the last inequality follows by using (3.3) for $i - e_{j_1} - e_{j_2}$.

$$\begin{aligned} \underline{0 = i_{j_1} = i_{j_2}} : \mu_{j_1} v_{(x, i)}^n + \mu_{j_2} v_{(x, i+e_{j_1})}^n &\stackrel{(3.2)}{\leq} \mu_{j_2} v_{(x, i)}^n + \mu_{j_1} v_{(x, i+e_{j_1})}^n \stackrel{(3.1)}{\leq} \\ \mu_{j_1} v_{(x, i+e_{j_2})}^n + \mu_{j_2} v_{(x, i)}^n. & \end{aligned}$$

$\underline{0 < i_{j_1} < i_{j_2}}$: directly from (3.1).

Now we continue with the term corresponding to arrivals. Let j^* be the optimal assignment in state $(y, i + e_{j_2})$, a an arbitrary action. Then $j^* \neq j_2$ according to (3.1).

$$\begin{aligned} \underline{j^* \neq j_1} : \lambda_{xay} \left(q_{xay} \min_j \{ v_{(y, i+e_{j_1}+e_j \wedge B)}^n \} + (1 - q_{xay}) v_{(y, i+e_{j_1})}^n \right) &\leq \\ \lambda_{xay} \left(q_{xay} v_{(y, i+e_{j_1}+e_{j^*})}^n + (1 - q_{xay}) v_{(y, i+e_{j_1})}^n \right) &\stackrel{(3.1)}{\leq} \end{aligned}$$

$$\begin{aligned} & \lambda_{xay} \left(q_{xay} v_{(y, i+e_{j_2}+e_{j^*})}^n + (1 - q_{xay}) v_{(y, i+e_{j_2})}^n \right) = \\ & \lambda_{xay} \left(q_{xay} \min_j \{ v_{(y, i+e_{j_2}+e_j \wedge B)}^n \} + (1 - q_{xay}) v_{(y, i+e_{j_2})}^n \right). \end{aligned}$$

$$\begin{aligned} \underline{j^* = j_1} : \quad & \lambda_{xay} \left(q_{xay} \min_j \{ v_{(y, i+e_{j_1}+e_j \wedge B)}^n \} + (1 - q_{xay}) v_{(y, i+e_{j_1})}^n \right) \leq \\ & \lambda_{xay} \left(q_{xay} v_{(y, i+e_{j_1}+e_{j_2})}^n + (1 - q_{xay}) v_{(y, i+e_{j_1})}^n \right) \stackrel{(3.1)}{\leq} \\ & \lambda_{xay} \left(q_{xay} v_{(y, i+e_{j_1}+e_{j_2})}^n + (1 - q_{xay}) v_{(y, i+e_{j_2})}^n \right) = \\ & \lambda_{xay} \left(q_{xay} \min_j \{ v_{(y, i+e_{j_2}+e_j \wedge B)}^n \} + (1 - q_{xay}) v_{(y, i+e_{j_2})}^n \right). \end{aligned}$$

Now let a^* be optimal in state $(x, i + e_{j_2})$. Then we have

$$\begin{aligned} & \min_a \left\{ \sum_y \lambda_{xay} \left(q_{xay} \min_j \{ v_{(y, i+e_{j_1}+e_j \wedge B)}^n \} + (1 - q_{xay}) v_{(y, i+e_{j_1})}^n \right) \right\} \leq \\ & \sum_y \lambda_{xa^*y} \left(q_{xa^*y} \min_j \{ v_{(y, i+e_{j_1}+e_j \wedge B)}^n \} + (1 - q_{xa^*y}) v_{(y, i+e_{j_1})}^n \right) \leq \\ & \sum_y \lambda_{xa^*y} \left(q_{xa^*y} \min_j \{ v_{(y, i+e_{j_2}+e_j \wedge B)}^n \} + (1 - q_{xa^*y}) v_{(y, i+e_{j_2})}^n \right) = \\ & \min_a \left\{ \sum_y \lambda_{xay} \left(q_{xay} \min_j \{ v_{(y, i+e_{j_2}+e_j \wedge B)}^n \} + (1 - q_{xay}) v_{(y, i+e_{j_2})}^n \right) \right\}. \end{aligned}$$

The proof of (3.2), monotonicity, is straightforward, we omit it. We continue with (3.3). The $m - 2$ terms corresponding to departures from the queues different from queue j_1 and j_2 follow directly with induction. Consider departures from queue j_1 and j_2 . Remark that $i_{j_1} > 0$.

$$\begin{aligned} \underline{i_{j_2} > 0} : \quad & \mu_{j_1} v_{(x, i-e_{j_1})}^n + \mu_{j_2} v_{(x, i-e_{j_2})}^n \stackrel{(3.3)}{\leq} \mu_{j_1} v_{(x, i^*-e_{j_2})}^n + \mu_{j_2} v_{(x, i^*-e_{j_1})}^n \stackrel{(3.1)}{\leq} \\ & \mu_{j_2} v_{(x, i^*-e_{j_2})}^n + (\mu_{j_1} - \mu_{j_2}) v_{(x, i^*-e_{j_1})}^n + \mu_{j_2} v_{(x, i^*-e_{j_1})}^n. \end{aligned}$$

$$\begin{aligned} \underline{i_{j_2} = 0} : \quad & \mu_{j_1} v_{(x, i-e_{j_1})}^n + \mu_{j_2} v_{(x, i)}^n \stackrel{(3.2)}{\leq} \mu_{j_2} v_{(x, i-e_{j_1})}^n + \mu_{j_1} v_{(x, i)}^n \stackrel{(3.3)}{\leq} \\ & \mu_{j_2} v_{(x, i^*-e_{j_2})}^n + \mu_{j_1} v_{(x, i^*)}^n. \end{aligned}$$

Consider the optimal action a^* , j^* in state (x, i^*) . Then $j^* \neq j_2$ according to (3.1). If $j^* \neq j_1$, the term for an arrival goes directly. If $j^* = j_1$, choose action j_2 in state (x, i) . \square