ON THE OPTIMALITY OF THE GENERALIZED SHORTEST QUEUE POLICY

Arie Hordijk and Ger Koole

Department of Mathematics and Computer Science

University of Leiden

P.O.Box 9512, 2300 RA Leiden

The Netherlands

Probability in the Engineering and Informational Sciences **4**:477–487, 1990

1. Introduction

The model we consider consists of m parallel servers, each with their own queue. Customers arrive according to a general arrival process. The system is controlled by assigning each incoming customer to one of the queues. The control may depend on the numbers of jobs present at the m servers. The service times are exponentially distributed with equal mean. We consider models with finite buffers and batch arrivals.

In section 2 we present our models. In section 3 we characterize the policy that maximizes stochastically the number of customers served by any time t. This policy avoids blocking and assigns to the queue with the smallest number of customers. It is a generalization of the Shortest Queue (SQ)Policy.

In the seminal paper of Winston [13] the model with exponential arrivals and service times and infinite buffers is considered. Our model extends Winston's results to systems which allow finite buffers and batch arrivals. Also our arrival process is general. It can be seen as a generalization of the doubly stochastic Poisson process with a Markov environment (cf. Cox & Ingham [1]). Elsewhere we will show that any arrival process is the limit of arrival processes of the type we use in this note.

In the literature extensions of Winston's results to non-exponential service times can be found. Let us summarize these results. We distinguish between three cases. First we consider the case in which the amount of time that each customer has already been in service is known. The policy which selects amongst the shortest queues the one whose first customer has already been in service the longest is optimal for the model with increasing failure rate service times. This is shown in Weber [10]. By using a service time distribution with a U-shaped failure rate Whitt [12] showed that this result does not hold for general service times.

For the second case in which only the numbers of customers are known Whitt [12] showed that the SQPolicy is not always optimal.

Note that for exponential service times there is no sense in making difference between the cases discussed above.

Thirdly we consider the case with non-exponential service times and where the total amount of work of any queue is known. To this case our result with batch arrivals can be applied. Indeed, a batch of size k can also be regarded as one customer asking k phases of service. Customers are assigned regardless of their service request. In this way we do not consider the number of customers waiting in the queue but the actual amount of work to be done measured in exponential phases of work. By using limiting arguments (cf. Hordijk & Schassberger [5], Whitt [11]) we arrive at the following assertion: By sending each arriving customer to the queue with the smallest amount of work to be done we stochastically maximize the amount of work done at any time. The optimality of this policy has recently been proved, in a completely different way, by Daley [2].

We next summarize other related results.

In Ephremides et al. [4] the sum of the expected sojourn times of all customers which arrive before a certain time t is minimized by the SQP for 2 identical exponential servers.

In Menich & Serfozo [6] the arrival rate, cost function and service rate may depend on the number of customers present. Conditions on those functions are given which guarantee the SQPolicy to be optimal. Finite buffers are not included in their results.

Another related model is the one with different exponential service rates to the servers and no waiting places. Seth [8] proved that for the model with 2 queues the Fastest Queue Policy is optimal. He also shows that this result cannot be extended to non-exponential service time distributions: a counterexample is given where the optimal policy sends the arriving customer to the server with stochastically larger service time if both servers are idle. Derman et al. [3] have established the optimality of the FQP for exponential service times, general arrival streams and more than 2 servers. Sobel [9] extends Seth's result on the optimality of the FQP to m servers and a cost structure involving different types of customers.

2. Models

We want to choose our arrival process in such a way that we can model centres with parallel queues in tandem. In this case when a customer in the first centre leaves, the departure rate of the first queue, which is equal to the arrival rate of the second, can change. With a doubly stochastic Poisson process we cannot model this because in this process with probability one the transition of a state and the arrival of a customer do not occur simultaneously. The example above leads us to the following generalization.

Arrival process.

Let X be the—possibly countable—state space of a Markov process with transition rates λ_{xy}, $x, y \in X$. When this process moves from x to y a customer arrives with probability q_{xy}, $\sum_{y \in X} q_{xy} \leq 1 \; \forall x \in X$.

Example 1. *Doubly stochastic Poisson process*

This arrival process includes the doubly stochastic Poisson process with a Markov environment: Let X^* be the state space of the governing Markov process with transition rates λ^*_{xy}. For simplicity assume $\lambda^*_{xx} = 0 \; \forall x \in X^*$. When this process is in state $x \in X^*$ customers arrive at rate μ_x. The equivalent process with a structure as defined above is the following: Take $X = X^*$, $\lambda_{xy} = \begin{cases} \mu_x, & \text{if } x = y; \\ \lambda^*_{xy}, & \text{otherwise,} \end{cases}$ and $q_{xy} = \begin{cases} 1, & \text{if } x = y; \\ 0, & \text{otherwise.} \end{cases}$

Example 2. *Renewal process*

As proven in Schassberger [7] we can approximate any distribution with nonnegative support with mixtures of Erlang distributions of the form $\sum_{x=1}^{\infty} \beta_x \mathsf{E}^x_\lambda$, where E^x_λ denotes the x-fold convolution of the exponential distribution with mean $\frac{1}{\lambda}$ and β_x is its weight. We can model an arrival process which is a renewal process with this as interarrival distribution in the following way: Let $X = \mathbb{N}_0$, take the starting probability in state $x \in X$ equal to β_x

and as transition intensities $\lambda_{xx-1} = \lambda$ if $x \geq 2$ and $\lambda_{1x} = \beta_x \lambda$. All other intensities are 0. The time until departure from state 1 is the lifetime wanted, thus $q_{xy} = \begin{cases} 1, & \text{if } x = 1 \\ 0, & \text{otherwise} \end{cases}$ $\forall y$.

For our arrival process we make the following assumption.

Assumption. $\sum_y \lambda_{xy} \leq M \quad \forall x$.

Furthermore we assume that all service times are exponentially distributed with mean $\frac{1}{\mu}$.

We consider 2 models.

Model A.

Each server has a buffer of size $B_i \leq \infty$, $B = (B_1, \ldots, B_m)$. Customers arrive one at a time.

Model B.

All buffers have infinite size. Customers arrive in batches of size k with probability β_k. All arriving customers must be assigned to the same queue and on arrival the size of the batch is not known to the controller.

Let i_j be the number of customers in queue j, $1 \leq j \leq m$.

Definition. *The Generalized SQPolicy assigns an arriving customer to the queue with index*

$$\min_{1 \leq j \leq m} \left\{ i_j = \min_{1 \leq j^* \leq m} \{ i_{j^*} | i_{j^*} < B_{j^*} \} \right\}.$$

This policy determines the queue to which an arriving batch of customers is sent uniquely. The optimality of the policy which assigns to any queue j with $i_j = \min_{1 \leq j^* \leq m} \{ i_{j^*} | i_{j^*} < B_{j^*} \}$ is shown in the next section. Hence the GSQPolicy is optimal.

Remark. If under model A an arriving customer is assigned to a queue which has a full buffer then it is lost. This provides a slight generalization of the model by adding a queue $m + 1$ with zero buffer size. Rejecting a customer is then included in the model through sending the customer to queue $m + 1$.

3. Optimality of the Generalized Shortest Queue Policy

We take the numbers of customers present and the state of the arrival process as part of the state of our model. This would be sufficient if the reward is equal for each departing customer, e.g. when we maximize the expected number of served customer. However, our technique to maximize the number of served customers stochastically only gives a reward to the sth customer leaving, $s \geq 1$. Therefore we introduce an extra variable and give it the initial value s. When a customer leaves the variable is lowered by 1 and the reward is earned when it is equal to 1 and a departure occurs.

Now we describe our model more formally. First we focus on model A. We take a semi-Markov Decision Process with states (i, s, x) where
$i = (i_1, \ldots, i_m)$ are the numbers of customers at the queues;
s is the number of departures before the reward is earned;
x is the state of the arrival process.

The possible actions are $a = 1, \ldots, m$, corresponding to assigning an (potentially) arriving customer to server a.

Define e_j the m-vector with a 1 in the jth component and 0 elsewhere; let $\mathbb{1}_{\{\ldots\}}$ denote the indicator-function and take δ_j equal to $\mathbb{1}_{\{j>0\}}$.

By introducing fictitious transitions we may assume that the time between decision epochs is exponentially distributed with mean $\frac{1}{M+m\mu}$ for all states and actions.

We then find that $p = \frac{\mu}{M+m\mu}$ is the probability that a customer who is in front of a queue leaves and $p_{xy} = \frac{\lambda_{xy}}{M+m\mu}$ is the probability that the state of the arrival process changes from x to y.

The transition probabilities r of the embedded Markov Decision Process are (denote with $a \wedge b$ the componentwise minimum of the vectors a and b):

$$r_{(i,s,x)a(i-e_j,s-1,x)} = p \quad \forall (i,s,x), a \text{ with } i_j > 0 \text{ and } s > 0$$

$$r_{(i,s,x)a(i+e_j \wedge B,s,y)} = p_{xy}\, q_{xy} \quad \forall (i,s,x), a, y, j \text{ with } a = j$$

$$r_{(i,s,x)a(i,s,y)} = p_{xy}(1 - q_{xy}) \quad \forall (i,s,x), a, y$$

$$r_{(i,s,x)a(i,s,x)} = 1 - \sum_y p_{xy} - p \sum_j \delta_{i_j} \quad \forall (i,s,x), a$$

All other transitions have probability 0. Note that a customer assigned to a full queue is lost.

By taking the immediate reward in states with $s = 1$ to equal the probability that a customer leaves and to equal 0 in all other states, we see that, when we start in (i, s, x), we only earn a reward after s customers have left. Moreover, the reward after n periods, $v^n_{(i,s,x)}$, is equal to the probability that s customers have left, starting in (i, x) and using the optimal policy.

Thus we complete our description of the MDProcess by giving the immediate reward-function ϕ:

$$\phi_{(i,s,x)a} = \begin{cases} p\sum_j \delta_{i_j}, & \text{if } s = 1, \\ 0, & \text{otherwise.} \end{cases}$$

Set

$$v^0_{(i,s,x)} = 0 \quad \forall (i, s, x), \tag{3.1}$$

and take for convenience

$$v^n_{(i,0,x)} = 0 \quad \forall n, i, x. \tag{3.2}$$

We get the following recursive relation for the value-function:

$$\begin{aligned} v^{n+1}_{(i,s,x)} =& \mathbb{1}_{\{s=1\}}\, p \sum_{j=1}^{m} \delta_{i_j} + \\ & p \sum_{j=1}^{m} \Big(\delta_{i_j}\, v^n_{(i-e_j,s-1,x)} + (1-\delta_{i_j})\, v^n_{(i,s,x)} \Big) + \\ & \max_{1\le a\le m} \Big\{ \sum_y p_{xy} \Big(q_{xy} v^n_{(i+e_a\wedge B,s,y)} + (1-q_{xy}) v^n_{(i,s,y)} \Big) \Big\} + \\ & \Big(1 - m\,p - \sum_y p_{xy} \Big) v^n_{(i,s,x)} \end{aligned} \tag{3.3}$$

Denote with $a \le b$ for a and b vectors $a_j \le b_j \ \forall j$.

Before we prove the optimality of the GSQPolicy we derive some technical inequalities.

Lemma 3.1. $\forall n \ge 0 \quad \forall s \ge 0 \quad \forall x \quad \forall i \le B$

$$v^n_{(i+e_{j_1},s,x)} \ge v^n_{(i+e_{j_2},s,x)} \quad i_{j_1} \le i_{j_2}, i_{j_1} < B_{j_1}, i_{j_2} < B_{j_2} \tag{3.4}$$

$$\mathbb{1}_{\{s=0\}} + v^n_{(i,s,x)} \ge v^n_{(i+e_j,s+1,x)} \quad i_j < B_j \tag{3.5}$$

$$v^n_{(i+e_j,s,x)} \ge v^n_{(i,s,x)} \quad i_j < B_j \tag{3.6}$$

$$v^n_{(i,s,x)} = v^n_{(i^*,s,x)} \quad \forall i^* \le B \text{ with } \begin{cases} i^*_{j_1} = i_{j_2} \\ i^*_{j_2} = i_{j_1} \\ i^*_j = i_j \forall j \ne j_1, j_2 \end{cases} \quad \forall j_1, j_2 \quad (3.7)$$
$$\text{and } \max\{i_{j_1}, i_{j_2}\} \le \min\{B_{j_1}, B_{j_2}\}$$

The proof is tedious but not difficult. It can be found in the appendix.

The equations (3.4) to (3.7) have an intuitive explanation: Equation (3.5) says that for $s > 0$ the value-function in an arbitrary state is bigger than the value-function in the state with one customer extra and with the reward received after one more departure. Thus it is better to serve as many customers as is possible. When $s = 0$ equation (3.5) just states that the value-function of states with $s = 1$ is smaller than 1. Equation (3.4) and (3.6) establish the GSQPolicy: (3.6) gives us that for each queue the value function increases with the number of customers at that queue, hence it is optimal to accept customers if possible; (3.4) says it is better to send the arriving customer to the shorter queue. Equation (3.7) compares 2 states where the number of customers in 2 queues are interchanged. The value-function in these states are equal. Consequently, when assigning customers to queues, it is only relevant to know whether there are free places or not. The numbers of free places at the queues do not matter.

Having established the optimality of the GSQPolicy for the discrete MDP we return to the original continuous model. Denote with $\mathbb{P}^R_{(i,x)}(s \text{ leaves before } t)$ the probability that while starting in (i, x) and using policy R at least s customers leave before t.

Theorem 3.2. *The GSQPolicy is stochastically optimal for model A, i.e.* $\forall i, x, s, t, R$

$$\mathbb{P}^R_{(i,x)}(s \text{ leaves before } t) \le \mathbb{P}^{GSQP}_{(i,x)}(s \text{ leaves before } t).$$

Proof.

Noting that transitions occur independently of the policy, we obtain by conditioning on the number of transitions by time t:

$$\begin{aligned}
&\mathbb{P}^{R}_{(i,x)}(s \text{ leaves before } t) = \\
&\qquad = \sum_{n=0}^{\infty} \mathbb{P}^{R}_{(i,x)}(s \text{ leaves before transition } n+1)\mathbb{P}(n \text{ transitions at } t) \\
&\qquad \leq \sum_{n=0}^{\infty} \mathbb{P}^{GSQP}_{(i,x)}(s \text{ leaves before transition } n+1)\mathbb{P}(n \text{ transitions at } t) \\
&\qquad = \mathbb{P}^{GSQP}_{(i,x)}(s \text{ leaves before } t)
\end{aligned}$$

We continue with stating the optimality of the GSQPolicy for model B. First we give the value-function:

$$\begin{aligned}
v^{n+1}_{(i,s,x)} =& \mathbb{1}_{\{s=1\}}\, p \sum_{j=1}^{m} \delta_{i_j} + \\
& p \sum_{j=1}^{m} \Big(\delta_{i_j} v^{n}_{(i-e_j,s-1,x)} + (1-\delta_{i_j}) v^{n}_{(i,s,x)} \Big) + \\
& \max_{1 \leq a \leq m} \Big\{ \sum_{y} p_{xy} q_{xy} \sum_{k} \beta_k v^{n}_{(i+ke_a,s,y)} \Big\} + \\
& \sum_{y} p_{xy}(1-q_{xy}) v^{n}_{(i,s,y)} + \\
& \Big(1 - m\,p - \sum_{y} p_{xy} \Big) v^{n}_{(i,s,x)}
\end{aligned} \tag{3.8}$$

The counterpart of lemma 3.1 is:

Lemma 3.3. $\forall n \geq 0 \quad \forall s \geq 0 \quad \forall x \quad \forall i$

$$\sum_{k} \beta_k\, v^{n}_{(i+ke_{j_1},s,x)} \geq \sum_{k} \beta_k\, v^{n}_{(i+ke_{j_2},s,x)} \quad i_{j_1} \leq i_{j_2} \tag{3.9}$$

$$\mathbb{1}_{\{s=0\}} + v^{n}_{(i,s,x)} \geq v^{n}_{(i+e_j,s+1,x)} \tag{3.10}$$

The prove of equation (3.9) can be found in the appendix. For the prove of (3.10) we refer to the proof of the equivalent assertion for model A in lemma 3.1.

The optimality of the GSQPolicy can be achieved analogously to model A:

Corollary 3.4. *The GSQPolicy is stochastically optimal for model B, i.e.* $\forall i, x, s, t, R$

$$\mathbb{P}^{R}_{(i,x)}(s \text{ leaves before } t) \leq \mathbb{P}^{GSQP}_{(i,x)}(s \text{ leaves before } t).$$

Proof.

Analogously to the proof of theorem 3.2. The inequality in the proof is valid in model B too because of lemma 3.3.

Acknowledgement

We are grateful to the editor for his stimulating comments on this paper.

References

1. Cox, D.R. & Isham, V. (1980). *Point processes.* London: Chapman and Hall.
2. Daley, D.J. (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. Derman, C., Lieberman, G.J. & Ross, S.M. (1980). On the optimal assignment of servers and a repairman. *Journal of Applied Probability* 17:577–581.
4. Ephremides, A., Varaiya, P. & Walrand, J. (1980). A simple dynamic routing problem. *IEEE transactions on Automatic Control* 25:690–693.
5. Hordijk, A. & Schassberger, R. (1982). Weak convergence for generalized semi-Markov processes. *Stochastic Processes and their Applications* 12:271–291.
6. Menich, R. & Serfozo, R.F. (1986). Optimality of shortest queue routing for dependent service stations. Working paper.
7. Schassberger, R. (1973). *Warteschlangen.* Wien: Springer.
8. Seth, K. (1977). Optimal service policies, just after idle periods, in two-server heterogeneous queueing systems. *Operations Research* 25:356–360.
9. Sobel, M.J. (1989). Throughput maximization in a loss queueing system with heterogeneous servers. Working paper.
10. Weber, R.R. (1978). On the optimal assignment of customers to parallel queues. *Journal of Applied Probability* 15:406–413.

11. Whitt, W. (1980). Continuity of generalized semi-Markov processes. *Mathematics of Operations Research* 5:494–501.
12. Whitt, W. (1986). Deciding which queue to join: some counterexamples. *Operations Research* 34:55–62.
13. Winston, W. (1977). Optimality of the shortest line discipline. *Journal of Applied Probability* 14:181–189.

Appendix

Proof of lemma 3.1.

We prove lemma 3.1 for each s and initial state by induction to n. For ease of notation we take $j_1 = 1$ and $j_2 = 2$. $n = 0$ is easy because of (3.1). Suppose the lemma holds for $1, \ldots, n$. We start with (3.4) with $i_1 \le i_2$.

$\underline{i_1 = i_2}$: see (3.7).

$\underline{i_1 \neq i_2}$. Take $i_2 > 0$.

$$\mathbb{1}_{\{s=1\}} + v^n_{(i,s-1,x)} \overset{(3.4)+(3.5)}{\geq} \delta_{i_1}(\mathbb{1}_{\{s=1\}} + v^n_{(i-e_1+e_2,s-1,x)}) + (1-\delta_{i_1}) v^n_{(i+e_2,s,x)} \tag{A.1}$$

$$\mathbb{1}_{\{s=1\}} + v^n_{(i+e_1-e_2,s-1,x)} \overset{(3.4)+(3.7) \text{ if } i_1+1=i_2}{\geq} \mathbb{1}_{\{s=1\}} + v^n_{(i,s-1,x)} \tag{A.2}$$

$$j > 2: \ \delta_{i_j}(\mathbb{1}_{\{s=1\}} + v^n_{(i+e_1-e_j,s-1,x)}) + (1-\delta_{i_j}) v^n_{(i+e_1,s,x)} \overset{(3.4)}{\geq} \delta_{i_j}(\mathbb{1}_{\{s=1\}} + v^n_{(i+e_2-e_j,s-1,x)}) + (1-\delta_{i_j}) v^n_{(i+e_2,s,x)} \tag{A.3}$$

If $\max_a \left\{ \sum_y p_{xy}\, q_{xy}\, v^n_{(i+e_2+e_a \wedge B,s,y)} \right\}$ is attained by:

$$a^* \geq 2: \quad \max_a \left\{ \sum_y p_{xy}\, q_{xy}\, v^n_{(i+e_1+e_a \wedge B,s,y)} \right\} \geq \sum_y p_{xy}\, q_{xy}\, v^n_{(i+e_1+e_{a^*} \wedge B,s,y)} \overset{(3.4)}{\geq} \max_a \left\{ \sum_y p_{xy}\, q_{xy}\, v^n_{(i+e_2+e_a \wedge B,s,y)} \right\}$$

$$a^* = 1: \quad \max_a \left\{ \sum_y p_{xy}\, q_{xy}\, v^n_{(i+e_2+e_a \wedge B,s,y)} \right\} \geq \sum_y p_{xy}\, q_{xy}\, v^n_{(i+e_1+e_2 \wedge B,s,y)} = \max_a \left\{ \sum_y p_{xy}\, q_{xy}\, v^n_{(i+e_2+e_a \wedge B,s,y)} \right\}$$

The two inequalities above give

$$\max_a \left\{ \sum_y p_{xy}\, q_{xy}\, v^n_{(i+e_1+e_a \wedge B,s,y)} \right\} \geq \max_a \left\{ \sum_y p_{xy}\, q_{xy}\, v^n_{(i+e_2+e_a \wedge B,s,y)} \right\} \tag{A.4}$$

Finally we need

$$v^n_{(i+e_1,s,x)} \overset{(3.4)}{\geq} v^n_{(i+e_2,s,x)} \tag{A.5}$$

Multiply (A.1), (A.2) and (A.3) with p, sum (A.3) from $j = 3$ to m; add (A.4); multiply (A.5) (with y instead of x) with $p_{xy}(1-q_{xy})$ and sum over y; multiply (A.5) with $1-mp-\sum_y p_{xy}$ and sum all to get $v^{n+1}_{(i+e_1,s,x)} \geq v^{n+1}_{(i+e_2,s,x)}$. We proceed with (3.5).

<u>$s = 0$</u>

$$v^{n+1}_{(i+e_1,1,x)} \overset{(3.2)+(3.3)+(3.6)}{\leq} m\,p + (1-m\,p) \max_y v^n_{(i+e_1+e_{a^*},1,y)} \overset{(3.5)}{\leq} m\,p + 1 - m\,p = 1$$

with a^* the maximizing action.

<u>$s > 0$</u>

$$\delta_{i_1}(\mathtt{1}_{\{s=1\}} + v^n_{(i-e_1,s-1,x)}) + (1-\delta_{i_1}) v^n_{(i,s,x)} \overset{(3.5)}{\geq} v^n_{(i,s,x)} \tag{A.6}$$

$$j > 1: \quad \delta_{i_j}(\mathtt{1}_{\{s=1\}} + v^n_{(i-e_j,s-1,x)}) + (1-\delta_{i_j}) v^n_{(i,s,x)} \overset{(3.5)}{\geq} \delta_{i_j} v^n_{(i+e_1-e_j,s,x)} + (1-\delta_{i_j}) v^n_{(i+e_1,s+1,x)} \tag{A.7}$$

$$\sum_y p_{xy}\, q_{xy}\, v^n_{(i+e_1,s,y)} \overset{(3.5)}{\geq} \sum_y p_{xy}\, q_{xy}\, v^n_{(i+e_1+e_{a^*},s+1,y)} \tag{A.8}$$

with a^* the maximizing action.

$$v^n_{(i,s,x)} \overset{(3.5)}{\geq} v^n_{(i+e_1,s+1,x)} \tag{A.9}$$

Multiply (A.6) and (A.7) with p, sum (A.7) from $j = 2$ to m; maximize the left side of (A.8); multiply (A.9) (with y instead of x) with $p_{xy}(1-q_{xy})$ and sum over y; multiply (A.9) with $1 - m\,p - \sum_y p_{xy}$ and sum all to get $\mathtt{1}_{\{s=0\}} + v^{n+1}_{(i,s,x)} \geq v^{n+1}_{(i+e_1,s+1,x)}$.

Now we prove (3.6).

$$\mathbb{1}_{\{s=1\}} + v^n_{(i,s-1,x)} \overset{(3.5)+(3.6)}{\geq} \delta_{i_1}(\mathbb{1}_{\{s=1\}} + v^n_{(i-e_1,s-1,x)}) + (1-\delta_{i_1})v^n_{(i,s,x)} \tag{A.10}$$

$$j > 1: \ \delta_{i_j}(\mathbb{1}_{\{s=1\}} + v^n_{(i+e_1-e_j,s-1,x)}) + (1-\delta_{i_j})v^n_{(i+e_1,s,x)} \overset{(3.6)}{\geq} \delta_{i_j}(\mathbb{1}_{\{s=1\}} + v^n_{(i-e_j,s-1,x)}) + (1-\delta_{i_j})v^n_{(i,s,x)} \tag{A.11}$$

Let a^* be the maximizing action in (i,s,x).

$$a^* = 1: \qquad \sum_y p_{xy}\, q_{xy}\, v^n_{(i+e_1+e_a\wedge B,s,y)} \overset{(3.6)}{\geq} \sum_y p_{xy}\, q_{xy}\, v^n_{(i+e_2+e_a\wedge B,s,y)} \tag{A.12}$$

$a^* > 1 \quad \Rightarrow \quad$ The maximizing action in $(i+e_1,s,x)$ is $a^* \Rightarrow$

$$\max_a \left\{ \sum_y p_{xy}\, q_{xy}\, v^n_{(i+e_1+e_a,s,y)} \right\} \overset{(3.6)}{\geq} \sum_y p_{xy}\, q_{xy}\, v^n_{(i+e_{a^*},s,y)} \tag{A.13}$$

$$v^n_{(i+e_1,s,x)} \overset{(3.6)}{\geq} v^n_{(i,s,x)} \tag{A.14}$$

Multiply (A.10) and (A.11) with p, sum (A.11) from $j=2$ to m; maximize the left side of (A.12); multiply (A.14) (with y instead of x) with $p_{xy}(1-q_{xy})$ and sum over y; multiply (A.14) with $1 - m\,p - \sum_y p_{xy}$ and sum all to get $v^{n+1}_{(i+e_1,s,x)} \geq v^{n+1}_{(i,s,x)}$.

We proof (3.7) with $i_1 < i_2$.

$$\delta_{i_1}(\mathbb{1}_{\{s=1\}} + v^n_{(i-e_1,s-1,x)}) + (1-\delta_{i_1})v^n_{(i,s,x)} \overset{(3.7)}{=} \delta_{i_2^*}(\mathbb{1}_{\{s=1\}} + v^n_{(i^*-e_2,s-1,x)}) + (1-\delta_{i_2^*})v^n_{(i^*,s,x)} \tag{A.15}$$

Analogously with the roles of 1 and 2 interchanged.

$$j > 1: \quad \delta_{i_j}(\mathbb{1}_{\{s=1\}} + v^n_{(i-e_j,s-1,x)}) + (1-\delta_{i_j})v^n_{(i,s,x)} \overset{(3.7)}{=} \delta_{i_j^*}(\mathbb{1}_{\{s=1\}} + v^n_{(i^*-e_j,s-1,x)}) + (1-\delta_{i_j^*})v^n_{(i^*,s,x)} \tag{A.16}$$

Let a^* (resp. a') be the maximizing action in (i,s,x) (resp. (i^*,s,x)). Then because of the optimality of the GSQP for n and $i_1 < i_2$ we have: $a^* \neq 2$; if $a^* = 1$ then $a' = 2$ and if $a^* > 2$ then $a' = a^*$. Now we have

$$\max_a \left\{ \sum_y p_{xy}\, q_{xy}\, v^n_{(i+e_a,s,y)} \right\} = \max_a \left\{ \sum_y p_{xy}\, q_{xy}\, v^n_{(i^*+e_a,s,y)} \right\} \tag{A.17}$$

$$v^n_{(i,s,x)} \overset{(3.7)}{=} v^n_{(i^*,s,x)} \tag{A.18}$$

Multiply (A.15) and (A.16) with p, sum (A.16) from $j = 3$ to m; add (A.17); multiply (A.18) (with y instead of x) with $p_{xy}(1 - q_{xy})$ and sum over y; multiply (A.18) with $1 - m\,p - \sum_y p_{xy}$ and sum all to get (3.7).

Proof of (3.9).

Take $j_1 = 1$, $j_2 = 2$ and $i_1 < i_2$. Then

$$\mathbb{1}_{\{s=1\}} + \sum_k \beta_k\, v^n_{(i+ke_1-e_1,s-1,x)} \overset{(3.9)+(3.10)}{\geq} \delta_{i_1}\left(\mathbb{1}_{\{s=1\}} + \sum_k \beta_k\, v^n_{(i+ke_2-e_1,s-1,x)}\right) + (1-\delta_{i_1})\sum_k \beta_k\, v^n_{(i+ke_2,s,x)} \tag{A.19}$$

$$\mathbb{1}_{\{s=1\}} + \sum_k \beta_k\, v^n_{(i+ke_1-e_2,s-1,x)} \overset{(3.9)}{\geq} \mathbb{1}_{\{s=1\}} + \sum_k \beta_k\, v^n_{(i+ke_2-e_2,s-1,x)} \tag{A.20}$$

$$j > 2: \quad \delta_{i_j}\left(\mathbb{1}_{\{s=1\}} + \sum_k \beta_k\, v^n_{(i+ke_1-e_j,s-1,x)}\right) + (1-\delta_{i_j})\sum_k \beta_k\, v^n_{(i+ke_1,s,x)} \overset{(3.9)}{\geq} \delta_{i_j}\left(\mathbb{1}_{\{s=1\}} + \sum_k \beta_k\, v^n_{(i+ke_2-e_j,s-1,x)}\right) + (1-\delta_{i_j})\sum_k \beta_k\, v^n_{(i+ke_2,s,x)} \tag{A.21}$$

Take $a^* = \arg\min_j \{i_j\} = \arg\min_j \{i_j | j \neq 2\}$.

$$a^* = 1: \quad \sum_l \beta_l \max_a \left\{ \sum_y p_{xy}\, q_{xy} \sum_k \beta_k\, v^n_{(i+le_1+ke_a,s,y)} \right\} \geq$$

$$\sum_l \beta_l \sum_y p_{xy}\, q_{xy} \sum_k \beta_k\, v^n_{(i+le_1+ke_2,s,x)} = \qquad \text{(A.22)}$$

$$\sum_l \beta_l \max_a \left\{ \sum_y p_{xy}\, q_{xy} \sum_k \beta_k\, v^n_{(i+le_2+ke_a,s,x)} \right\}$$

$$a^* \geq 3: \quad \sum_l \beta_l \max_a \left\{ \sum_y p_{xy}\, q_{xy} \sum_k \beta_k\, v^n_{(i+le_1+ke_a,s,x)} \right\} =$$

$$\sum_l \beta_l \sum_y p_{xy}\, q_{xy} \sum_k \beta_k\, v^n_{(i+le_1+ke_{a^*},s,x)} \overset{(3.9)}{\geq}$$

$$\sum_l \beta_l \sum_y p_{xy}\, q_{xy} \sum_k \beta_k\, v^n_{(i+le_2+ke_{a^*},s,x)} = \qquad \text{(A.23)}$$

$$\sum_l \beta_l \max_a \left\{ \sum_y p_{xy}\, q_{xy} \sum_k \beta_k\, v^n_{(i+le_2+ke_a,s,x)} \right\}$$

$$\sum_k \beta_k\, v^n_{(i+ke_1,s,x)} \overset{(3.9)}{\geq} \sum_k \beta_k\, v^n_{(i+ke_2,s,x)} \qquad \text{(A.24)}$$

Multiply (A.19), (A.20) and (A.21) with p, sum (A.21) from $j = 3$ to m; add (A.22) or (A.23); multiply (A.24) (with y instead of x) with $p_{xy}(1-q_{xy})$ and sum over y; multiply (A.24) with $1 - m\,p - \sum_y p_{xy}$ and sum all to get (3.9).