Advances in Stochastic Modelling, edt. J. Artalejo, A. Krishnamoorty, Notable Publications, 2002 (55-72).

ON PERFORMANCE EVALUATION AND OPTIMIZATION PROBLEMS IN QUEUES WITH RESEQUENCING *

B. Dimitrov, D. Green Jr., V. Rykov, and P. Stanchev
Kettering University, Flint, Michigan, USA,

Key Words: Performance evaluation, Resequencing Models, Optimal policies for heterogeneous services, Matrix-analytic approaches

Abstract

Studies of queuing models with resequencing of jobs to keep the order at the output in the order of arrival are reviewed. A focus on recent authors results regarding optimal jobs assignment to heterogeneous servers is presented.

1 Introduction and Notations

The resequencing problems arise in numerous practical situations, where multiple parallel service applies. They appear in communication system networks, distributed database systems, production flow-networks, in information networks and other multi-server queuing models. The reason: simultaneously served jobs use different physical servers when passing through a given node or link of the network. Different service rates, unequal job lengths, possible errors, varying processing times, etc. may cause variable delays into the node, and mixed completion times. However, if these jobs are completed in an order different from the order of arrival at the node, it may destroy the processing order at the next nodes. In order to ensure the right processing through the network, the original order needs to be restored. Thus, an additional queue of already served jobs, called the *recequencing buffer* (RSB) may be formed before departure from the node where late arrived and early served jobs are kept. The RSB is emptied when all earlier arrived jobs complete their services. The queue in front of the servers (if any) is named *primary buffer* (PB). The waiting times and the queue lengths in the PB and RSB are denoted by W_{PB}, W_{RSB}, $L_{PB}, and\ L_{RSB}$. The input flow of jobs, if Poissonian, has intensity λ with possible subscripts whenever different classes of jobs are considered. The service rate on server i is μ_i. The numeration of servers usually is in the order which makes sure that $\mu_1 \geq \mu_2 \geq \ldots \geq \mu_n$ is satisfied.

Various performance characteristics of the queuing models with (and without) resequencing depend on rules for assigning jobs from PB to servers. Rules depend also on the availability of information about the situation in the systems, and on the control objectives. There are several possibilities to control assignment policy depending on observability of the system states and dispatcher position. It is possible to dispatch jobs to separate queues in front of a PB (dispatcher position D_1) without the opportunity to change the server or directly assign servers after the common PB (dispatcher position D_2). A principal scheme of a queuing system with resequencing, with different positions of dispatcher (only one of D_1 or D_2 is allowed), and a following next service station, is shown in Fig. 1.

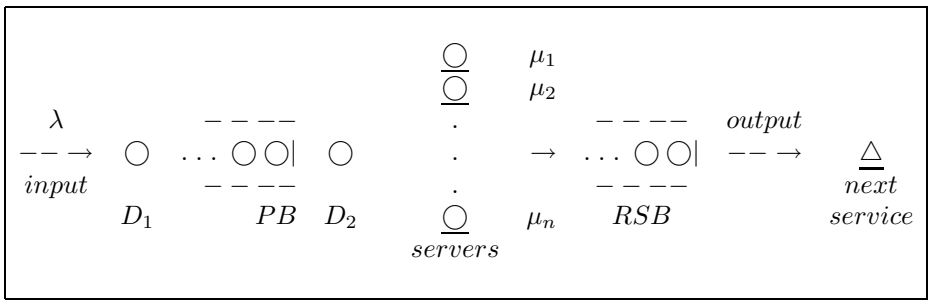

Fig. 1 Resequenced queuing system - a fragment.

*This work is partially supported by RFFI Grants No. 99-01-00034 and No. 01-07-90259, and Drant No. 160062 from Kettering University

It was B. Krishnamoorthi (1963) who first noticed heuristically that the disorder in a heterogeneous M/M/2/∞ queue would be minimal if the slower server is used only when the (PB) queue exceeds $m+1$, where m is the integer part of the number μ_1/μ_2. By disorder he means the relative number of jobs served (and therefore, departure) before other jobs which arrived earlier. In this connection B. Krishnamoorthi proposed the slower server to pick up for service only jobs waiting at the position $m+1$, if any, or otherwise to stay unused. The argument in favor of this rule is simple: to minimize the waiting time (and disordered served jobs on the output) it is preferable to wait in the queue for the fast server if the remaining waiting time to the end of this service $m\mu_1^{-1}$ is less than the mean service time μ_2^{-1} on the slower server. Hence, solution of a resequencing problem is equivalent to finding an optimal direction of the waiting jobs to the available servers.

About 11 years later Washburn (1974) analyzed the total delay in a $M/M/n/\infty$ queue under a "no passing rule", i.e. when departures must respect the order of arrivals. Also "zero-service times" were allowed with probability p. The expected waiting time was studied. He mentioned that the mean waiting penalty time (in the RSB) implied by a no passing rule increases with the traffic level even when the number of servers is infinite.

Researchers showed low interest to resequencing problems in the 60's and 70's. Two related papers of Krishnamoorthi (1963), and Washburn (1974) went virtually unnoticed. Publicity of practical realizations of resequenced models came before most theoretical studies which triggered an intensive research effort. The IBM system network architecture (Gray and McNeil, 1979), and the French public network TRANSPAC (Danet et al. 1976) were designed with multi-parallel communication links named Transmission Groups (briefly, TG). First-in-first-out order is required by the TG protocol. A transmitted package through a link of a TG cannot be scheduled for further transmission until all packages that started their transmission earlier within the same TG arrive at the receiving node.

In this paper we give a short review of some general models on resequencing within one service node. We specially focus on models and results of Kettering University researchers. Some optimal assignment problems which include certain aspects of the resequencing are presented. Further analysis of optimal assignment policies and the MAP models of resequencing problems are also considered in brief.

2 General Model and Some Results

Most of the explicit analytic results are obtained for queuing resequencing models with Poisson arrivals and exponential service times. However, the work of Baccelli et al. (1984) gives a general and most abstract statement of the resequencing problems as follows:

Consider a sequence of jobs $\{\phi_n\}_{n=0}^{\infty}$, entering a service system at (an increasing sequence of) corresponding instances $\{a_n\}_{n=0}^{\infty}$. Each job ϕ_n is delayed within the system by some time B_n, $n = 0, 1, \ldots$. At the output of the system the objects appear at instance $t_n = a_n + B_n$ which are not necessarily in chronological order any more (i.e. it is possible that $t_n > t_k$ for $k > n$). The objects are then processed by a resequencing algorithm (RA); ϕ_n will receive some service of duration S_n and will depart at time d_n. However, this service can only be given in the same order as the original arrival instance; that is, ϕ_n must be served after ϕ_{n-1} and before ϕ_{n+1}. The RA delays an object ϕ_n until all of the objects ϕ_k with $k < n$ have been released by the RA. (One could recognize here the $G/G/\infty$ system with resequencing and a next service to be incorporated, as shown in Fig. 1. Service times will be denoted by B_n. The second service must accept the jobs in their original arriving order as in the first node). The case of $S_n = 0$ for all $n \geq 1$ is named a "pure resequencing problem". The non-zero service times are assumed to consider the "effect of an output service time on resequencing delays". The questions are: What all this would cost, and in particular, what is the total effective delay $W_n = d_n - a_n$?

Let $A_n = a_n - a_{n-1}$ be the inter-arrival times. The following *Total Delay Equation for the resequencing Algorithm* is obtained:

$$W_n = \begin{cases} B_n + S_n, & \text{if } B_n > W_{n-1} - A_n, \\ B_n + S_n + [W_{n-1} - A_n - B_n], & \text{otherwise.} \end{cases}$$

In the case of $\{A_n\}$, $\{B_n\}$, and $\{S_n\}$ are independent sequences of i.i.d. random variables it is shown that under the condition $\mathbf{E}[S_{n-1} - A_n] < 0$, the sequence of delays

$$Y_1 = D_1 \;\; Y_n = \max\,[B_n,\; Y_{n-1} + S_{n+1} - A_n],\; n \geq 2$$

converges in distribution to a proper random variable Y. Its conditional limiting probability distribution

$$U(x) = \lim_{n\to\infty} \; P\{Y_n \leq x \mid D_n \leq x\}$$

is obtained in terms of Laplace-Stieltjes Transform (LST) in cases of Poisson arrivals and hyper-exponential delays. The Winner-Hopf factorization technique is applied. In particular, for one server system, and $S_n = 0$

the results of Kamoun et al. (1981) concerning $M/M/\infty$ resequenced queues are obtained. Areas of application, as in Packet-Switching Networks, Distributed Databases, and Pure Mathematical Problems are outlined.

Kamoun et al. (1981) introduced the concept of *Star Task* (briefly, ST). This is the case where each job does not wait in RSB. A ST will eventually release the RSB from all jobs started later and served earlier. The embedded L_{RSB} is the bunch of jobs departing with the ST. For the exponential case the authors found the inter-departure distribution between two consecutive STs as well as the waiting time distribution of jobs which are not STs. Harrus and Plateau (1982) extended these results for $M/G/\infty$ queues with resequencing, i.e. for general distribution of the service time. Also, they found a number of stationary characteristics: the mean number of jobs in the RSB, the p.d.f. of the service time of a ST, as well as the joint distribution of inter-departure time between two consecutive STs, and the distribution of the number of jobs in RSB just before departure of a ST. Finite second moment of the service times is required in order to ensure finite expectations of the listed characteristics. Explicit particular results are obtained for the case of exponential service time and the early known results for $M/M/\infty$ queues with the same kind of resequencing were confirmed. Graphical illustrations show the dependence of the mean stationary characteristics on the variance of the service time.

3 Optimal Assignment Problems

A special direction of resequencing problems is the optimal assignment of the jobs to heterogeneous servers. In such a way, the resequencing time, or another performance measures can be minimized. It is possible to construct some specific policy for directing the waiting jobs to the available servers of lower rate, in order to reduce the total waiting time, or the proportion of jobs that will experience resequencing (as in Krishnamoorthi, 1963), or optimize other important performance measure. A policy, which produces optimal performance measure is named *optimal policy* (OP).

The solution of an optimal assignment problem depends on the observation possibilities (information available) for the system states. There are two principal cases: no information about system states, and observable system states. The second case may have two sub-cases. First, dispatcher D_1 is located before the PB and sends new arriving jobs to separate queues at each server. Second, dispatcher D_2 is located between PB and servers, and sends jobs from from the head of the queue to servers at times of arrivals or ends of services. We discuss each of these schemes further. We begin with the assignment problem from common PB.

3.1 Common PB. Observable system states

This model considers dispatcher D_2 between the PB and servers, where PB represents the common queue for all servers. Dispatcher makes decisions at the arriving or service completion of jobs, according to information about total queue length q, the availability of servers $d = (d_1, \ldots, d_n)$, and the rates of service.

The hypothetical result of Krishnamoorthi (1963) was proved in the work of Lin and Kumar (1984). They proved that in case of resequenced queue with two heterogeneous servers and common PB the OP (which minimizes the expected total delay of a job in the system) is of the following threshold type:

Theorem 1. *The faster server will serve the first job from the head of the PB whenever it becomes available for service. The slower server should be utilized if and only if the queue length exceeds a readily computed threshold value m^*.*

Lin and Kumar (1984) also calculated the mean total delay for any given threshold m in the $M/M/2$ resequencing queue. The optimal m^* is then determined by the use of an iterative algorithm. This result was generalized in Rykov (2001). The system with n servers represented by a controllable Markov process $\{Z(t)\} = \{(X(t), U(t))\}$ with observable component $\{X(t)\} = \{(Q(t), D(t))\}$, where $Q(t)$ is the queue length and $D(t) = \{(D_1(t), \ldots, D_n(t))\}$ describes the states of servers: $D_i(t) = 0$ or 1, if server i is free or busy. The phase space of this component is $E = \mathbf{N} \times \{0, 1\}^n$.

To introduce the controlling process let us denote by $J_0(x)$ and $J_1(x)$ the subsets of idle and busy servers at state x respectively. The controlling process $\{U(t)\} = \{U_j(t) : \ j \in \{0\} \cup J_1(x)\}$ is also multidimensional with number of components $\#(J_1(x)) + 1$ dependent on state x of the observable process $\{X(t)\}$. The components indicate the label of server which has to be switched on in the case of arrival ($U_0(t)$) or service completion by j-th server ($U_j(t)$). The components take values in the control (decision) set $A = \{0, 1, dots, n\}$, and the decision $U_j(t) = 0$ denotes to do not use any server, while decision $U_j(t) = k$ means to switch on the server k. Thus the set of decisions (control space) is $A = \{0, 1, \ldots, n\}$.

Under the considered assumptions, the process $\{Z(t)\} = \{(X(t), U(t))\}$ is a Markov decision process with phase space $E = \mathbf{N} \times \{0, 1\}^n$ and control space $A = \{0, 1, \ldots, n\}$. The states are denoted by $x = (q, d_1, \ldots, d_n)$, and the control $a = 0$ means that no additional server must be switched on, and the control $a = k$ denotes that the k-th server must be switched on. Denote by e_k the $(n+1)$-dimensional vector $e_k = (\underbrace{0, \ldots, 0}_{k-1}, 1, \underbrace{0, \ldots, 0}_{n-k})$

and by $f = \{f(x) : \ x \in E\}$ some control policy. Then the transition intensities of the process $\{Z(t)\}$ under policy $f = \{f(x) : \ x \in E\}$ can be presented in the form

$$\lambda_{xy}(f(x)) = \begin{cases} \lambda, & \text{for } y = x + e_{f(x)}, \\ \mu_j, & \text{for } y = x - e_j + 1_{\{q>0\}}(e_{f(x-e_j-e_0)} - e_0), \quad j \in J_1(x), \\ 0, & \text{otherwise.} \end{cases}$$

The problem of minimization of disorder in the output process is reduced to the problem of minimizing the long run mean number of jobs in the system.

It is well known (see for example Kitaev and Rykov, 1995) that for a Markov decision problem with finite or denumerable phase and decision spaces and the long-run average criterion, the optimality principle is valid. The optimal policy is a solution of the optimality equations in policy space. It is shown in Rykov (2000) that the optimality equations for the considered problem can be represented in the following form

$$v(x) = l(x) + \lambda T_0 v(x) + \sum_{1 \le j \le K} \mu_j T_j v(x) - g = Bv(x), \quad x \in E. \tag{1}$$

Here $l(x) = q(x) + \sum_{1 \le j \le n} d_j(x)$ is the number of jobs in the system at state x, the *gain*, g, is the minimal long run mean number of jobs in the system, $v = \{v(x : \ x \in E)\}$ denotes the *value function* of the model, T_0 and T_j are operators defined by the formulas

$$\begin{aligned} T_0 v(x) &= \min[v(x + e_k) : \quad k \in A_0(x)], \\ T_j v(x) &= \begin{cases} T_0 v(x - e_j - e_0), & \text{for } j \in J_1(x), \quad q(x) > 0, \\ v(x - e_j), & \text{for } j \in J_1(x), \quad q(x) = 0, \\ v(x), & \text{for } j \in J_0(x), \end{cases} \end{aligned}$$

and $Bv(x)$ denotes the operator which transforms the functions $v(x)$ according to requirements (1). By analyzing these equations one can recognize that the function to be minimized, so called *Bellman function* of the model can be represented in the form

$$b(x; k) = v(x + e_k). \tag{2}$$

Now we deal with multidimensional state space; thus in order to get the threshold policy we need to define a certain order in the state and control spaces. First, the servers are labeled in order of their intensities decreasing, i.e. $\mu_1 \ge \mu_2 \ge \cdots \ge \mu_n$, and the same is true for the order of components of the vector $d = (d_1, \ldots, d_n)$. This arrangement determines the usual total order in the decision set A, where $0 < 1 < \ldots < n$. Define a partial order in $E = \mathbf{N} \times \{0, 1\}^n$ in the following way:

$$x + e_j \ge x, \qquad \text{for all } j \in \{0\} \cup J_0(x),$$

$$x + e_i \ge x + e_j, \quad \text{for all } i, j \in J_0(x) \quad \text{with } i \ge j \ \ (\mu_i \le \mu_j),$$

and

$$x + e_0 + e_i \ge x + e_j, \quad \text{for all } i, j \in J_0(x).$$

Note, that in this partial order the points $x + e_0$ and $x + e_j$, $(j \ne 0)$ are not comparable. By analyzing the properties of the value function of the model $v = \{v(x) : \ x \in E\}$ and taking into account the structure of the Bellman function (2) for model we get the following result. (Due to the definition of operators T_j, $j = 0, 1, \ldots, n$, optimal decisions are needed only at the times of arrival).

Theorem 2. *For the considered model any optimal policy is of a threshold type, i.e. for each state x, there exists some level $m^*(x)$ of the queue length depending on the subset of busy servers $J_1(x)$ such that it is recommended to switch on an additional server only if $q(x + e_0) > m^*(x)$; in this case one should switch on the fastest idle server. On the other hand, if in some state x, the optimal decision recommends not to use any idle servers, then this decision is also optimal for all y with the same subset of busy servers (i.e for which $J_1(y) = J_1(x)$), and lesser queue length, $q(y) \le q(x)$.*

This theorem gives just a qualitative description for any optimal policy. The determination of the optimal thresholds levels needs further numeric analysis. The numerical solution can be found by the use of the Howard's algorithm (Howard, 1960) for searching an optimal policy. Some numerical examples based on this approach were given in Dimitrov et al. (2001).

3.2 Observable separate queues

In this subsection we discuss a problem of packeting job assignments to heterogeneous servers subject to minimization of the mean processing cost. As a special case of the model, when holding cost satisfies $c_i = 1$ for all $i = 1, 2, \ldots, n$, the minimization of total sojourn time will be found (and it is equivalent to solving the resequencing problem). In this case the dispatcher D_1 stays in front of the PB which consists of n separate queues for each server. The problem with separate observable queues was considered in scheduling theory context in Rykov and Levner (1998) targeting minimization of the long run mean processing cost, and in Rykov (2000) subject to minimization of the delay probability.

Consider the problem of scheduling K homogeneous jobs with random processing times $B_j,\ j = 1, 2, \ldots, K$ of the mean workload b between n servers subject to minimization of the total processing cost. The holding cost for the i-th server is c_i (it can include both the waiting cost h and service cost u_i on the i-th server: $c_i = h + u_i$). The processing rate of ith server is μ_i, therefore the expected service time for any job on the i-th server is $b_i = b\mu_i^{-1},\ \ i = 1, \ldots, n$.

The decision rule for this problem can be presented as an assignment $\mathbf{k} = (k_1, k_2, \ldots, k_n)$ of $K = \sum_{1 \le i \le n} k_i$ jobs to n servers. The cost of processing k_i jobs on the i-th server is

$$C_i = c_i \sum_{1 \le j \le k_i} (k_i + 1 - j) B_j.$$

Its expected value is $\mathbf{E}[C_i] = c_i b_i (1 + 2 + \ldots + k_i) = c_i b \frac{k_i(k_i+1)}{2\mu_i}$. Thus, the processing cost $v(\mathbf{k})$ of assignment $\mathbf{k}$ can be represented by

$$v(\mathbf{k}) = \sum_{i=1}^{n} c_i b \frac{k_i(k_i+1)}{2\mu_i}. \tag{3}$$

Let us introduce parameters $\gamma_i = \mu_i^{-1} c_i$, which represent the single job "processing cost" by i-th server, and denote by $[x]$ the integer part of number x. The following theorem was proved in Rykov and Levner (1998).

Theorem 3. *The necessary and sufficient condition for an assignment* $\mathbf{k}^* = (k_1^*, k_2^*, \ldots, k_n^*)$ *to be optimal is the fulfillment of the following inequalities:*

$$k_i^* \le \min_{1 \le j \le n} \left[\frac{\gamma_j}{\gamma_i} (k_j^* + 1) \right] \tag{4}$$

for all $i = 1, \ldots, n$.

Remark 1. Assume that K jobs are already optimally assigned to n servers, and their allocation is $(k_1, k_2, ..., k_n)$, where $\sum_{1 \le i \le n} k_i = K$. Then the allocation of the next, $(K+1)$-st, job to any server with label $i^* = \operatorname{argmin}\{\gamma_i(k_i + 1) :\ 1 \le i \le n\}$ with minimal value of $\gamma_i(k_i + 1)$, is optimal.

Remark 2. Although the results here are obtained in the scheduling theory approach, it also remains valid for queuing models with non observed service times of the currently served jobs because in either arrival or service completion time the dispatcher is solving a new job scheduling problem.

Remark 3. The results, represented in this section, are valid for any (not necessary exponential) distribution of service times.

As a special case of this solution we get the following result.

Corollary 1. *An assignment, which minimizes the total sojourn time of all jobs in the system, consists in assigning each new job to the server with minimal value of* $(k_i + 1)\mu_i^{-1}$.

The Theorem above gives the possibility to obtain some estimations for the optimal assignment $\mathbf{k}$, and propose an approximate heuristic decision rule.

Corollary 2. A heuristic decision rule. *An approximate solution* $\hat{\mathbf{k}} = (\hat{k}_1, \hat{k}_2, \ldots, \hat{k}_n)$ *to the optimal assignment problem above is given by the formulas*

$$\hat{k}_1 = \left[\frac{n\mu_1}{M} \right], \qquad \hat{k}_i = \left[\frac{n\mu_i}{M} \right], \tag{5}$$

where the notation $M = \mu_1 + \ldots + \mu_n$ *is used. This assignment usually gives a lower bound of the optimal solution. Hence, allocating the last job (in accordance with Remark 1) to the server with minimal value of* $(\hat{k}_i + 1)\mu_i - 1$ *is recommended.*

These results show that, as in previous case, there is no simple rule for determination of threshold levels. These levels depend on the states of servers and queue lengths. Nevertheless, for any system state these rules are allowed to determine the label of server to which newly arriving job should be send.

Notice that as a special case for $n = 2$ of Corollary 1 the heuristic rule of Krishnamoorthi can be obtained: the optimal threshold level for assigning jobs to two heterogeneous servers equals $m^* = [\frac{\mu_1}{\mu_2}] + 1$. Let m^* be the first job to go to server 2. Then by Corollary 1 we get

$$\frac{m^*+1}{\mu_1} < \frac{1}{\mu_2} \le \frac{m^*+1}{\mu_1} \qquad \text{i.e.} \qquad m^* < \frac{\mu_1}{\mu_2} \le m^* + 1$$

Thus, $m^* = \left[\frac{\mu_1}{\mu_2}\right] + 1$, and this is the Krishnamoorthi's result.

As an Example of Theorem 3 consider its application on assignment of $K = 20$ jobs to $n = 3$ servers with equal processing costs and processing intensities $\mu_1 = 5,\ \mu_2 = 3,\ \mu_3 = 1$. The calculations according to the rules of Theorem 3 are shown in the following table

Table 1.

#of job	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
# of server	1	2	1	1	2	1	1	2	3	1	2	1	1	2	1	1	2	3	1	2

One can recognize here Krishnamoorthi's rule for the first two servers. The second server starts only when the queue before the first server attains $m_1^* = \left[\frac{\mu_1}{\mu_2}\right] + 1 = 2$ waiting jobs. This rule also takes place for the third server, if we consider the two first as a common server with intensity $\mu = \mu_1 + \mu_2 = 8$. The third server should be loaded only when the common queue at first two servers attains the level $m_1^* = \frac{\mu_1+\mu_2}{\mu_3} + 1 = 9$.

Other numerical examples represented in Rykov and Levner(1998).

3.3 Minimization of the delay probability

We consider the problem of jobs allocation between n heterogeneous servers in another scheduling theory approach. As in previous subsection, let K homogeneous jobs, each with random workload B_j and common cumulative distribution function (CDF) $B(t) = \mathbf{P}\{B_j \le t\}$ have to be allocated on the servers subject to minimization of the probability of getting high delay. If the i-th server processes a job with intensity μ_i, then the service time on this server would have a distribution

$$F_i(t) = B(\mu_i t).$$

Denote by W the common processing time and by W_i the sojourn time in i-th queue, so that $W = \max\{W_i : i = 1, \dots, n\}$. Then the problem consists in job allocation on the servers in such a way that the probability of the common delay, W, to exceed a given amount of time t_0 to be minimal. That is

$$\mathbf{P}\{W > t_0\} \Longrightarrow \min.$$

Instead of minimizing the delay probability, we will maximize the probability that the common processing time gets below t_0. Due to independence of the servers we have

$$\mathbf{P}\{W \le t_0\} = \mathbf{P}\{W_i \le t_0,\ i = 1, \dots, n\} = \prod_{i=1}^{n} \mathbf{P}\{W_i \le t_0\}. \tag{6}$$

Denote by $\mathbf{k} = \{k_i,\, i = 1, \dots, n\}$ the schedule with k_i jobs allocated to the i-th server and by $g_i(k_i)$ the probability that the processing time of the i-th server is less than t_0. We get

$$g_i(k_i) = g_i(k_i, t_0) = \mathbf{P}\{W_i \le t_0\} = F_i^{(*k_i)}(t_0), \qquad g_i(0) \equiv 1,$$

where superscript "*" denotes convolution. Then the problem minimization of delay probability (MDP problem) can be rewritten on the form

$$g(\mathbf{k}) = \prod_{i=1}^{n} g_i(k_i) \Longrightarrow \max. \tag{7}$$

The following theorem was proved in Rykov (2000).

Theorem 4. *A schedule* $\mathbf{k}^* = (k_1^*, k_2^*, \dots, k_n^*)$ *is optimal for MDP problem if and only if it satisfies the inequalities*

$$\frac{g_i(k_i^*)}{g_i(k_i^* - 1)} \ge \max_{1 \le j \le n} \frac{g_j(k_j^* + 1)}{g_j(k_j^*)} \tag{8}$$

for all $i = 1, \ldots, n$.

Remark 1. Assume that K jobs are already optimally allocated to n servers, and the allocation is $(k_1, k_2, ..., k_n)$, where $\sum_{1 \le i \le n} k_i = K$. Then the next $(K+1)$-st job should be directed to a server with label i^*, which satisfies the inequalities (12), in order to have the $K+1$ jobs optimally allocated.

The theorem allows us to propose the following combinatorial algorithm for calculation of optimal allocation.

Renumber the servers in increasing order of their "productivity" $g_i(1)$:

$$g_1(1) \ge g_2(1) \ge \ldots \ge g_n(1) \tag{9}$$

and set $g_0(k) \equiv g_{n+1}(k) \equiv \infty$ for any k.

The key idea of the algorithm is to load the servers according to their increasing "productivity", i.e. according to inequalities (9). At each iteration (corresponding to an arrival of a new job), the algorithm verifies where the inequalities (12) holds, including the new job in consideration.

Some numerical examples illustrate the work of this algorithm in Rykov (2000).

3.4 Optimal control of systems with non-observable states

If the information about system states is not available then it is still possible to find some stochastic mechanism to optimize the system control. The control rule can be realized in several ways: either directly by a random mechanism (by distributing the newly arriving jobs at random to the servers according to some probability distribution), or by some reasonable cyclic routine, realizing a suitable optimal distribution of the jobs on the servers. In each of these cases the control rule can be determined by a set of probabilities $\xi = (\xi_1, \ldots, \xi_n), \quad \xi_1 + \ldots + \xi_n = 1$, and the system is partitioned into n individual $\mathbf{M/M/1/\infty}$-type subsystems, each with Poisson input of intensity $\lambda_i = \lambda \xi_i$ and exponentially distributed service times. The objective function in this case has the form

$$g = g(\xi) = \lim_{t \to \infty} t^{-1} \mathbf{E}^{\xi} \sum_{1 \le i \le n} c_i \mathbf{E}^{\xi} L_i(t)\, dt = \sum_{1 \le i \le n} c_i \mathbf{E}^{\xi} l_i, \tag{10}$$

where $L_i(t) = Q_i(t) + D_i(t)$ is the number of jobs in i-th subsystem at time t, and l_i is its expected stationary value. If the condition

$$\lambda < \sum_{1 \le i \le n} \mu_i \tag{11}$$

is fulfilled, then the mean number $l_i = \mathbf{E}^{\xi} L_i(.)$ of jobs in each subsystem is given by (see Gnedenko and Koenig 1981, Ch. 3),

$$l_i = \mathbf{E}^{\xi} L_i = \mathbf{E}^{\xi} Q_i + \mathbf{E}^{\xi} D_i = \frac{\rho_i}{1 - \rho_i},$$

where $\rho_i = \lambda_i \, \mu_i^{-1}$. Using these remarks, for the functional (10) which has to be minimized we obtain the following expression

$$g(\xi) = g(\xi_1, \ldots, \xi_n) = \lambda \sum_{i=1}^{n} \frac{c_i \xi_i}{\mu_i - \lambda \xi_i}. \tag{12}$$

Therefore, the problem is reduced to the following: minimize the functional (12) with respect to the variables $\xi = (\xi_1, \ldots, \xi_n)$ under the restrictions

$$\sum_{i=1}^{n} \xi_k = 1, \qquad \xi_i \ge 0, \qquad \mu_i \ge \lambda \xi_i \quad (i = 1, 2, \ldots, n). \tag{13}$$

This is a non-linear optimization problem and its solution can be found using any mathematical programming. There are many papers dealing with this problem. A suitable bibliography with historical guidelines can be found in Dimitrov (1997). Some special approach based on the Lagrange multipliers is proposed in Rykov and Ermolaev (1999). It allows the exclusion of the first of the restrictions in (13), and by using an appropriate approximation method and algorithm, the solution can be completed. This approach is extended by Dimitrov et al.(2001), with more numerical studies and comparison of queues with heterogeneous servers under various restrictions on the decision process.

4 Further Analysis

Further discussion on threshold policies is given by Iliadis and Lien (1988), who surveyed the published articles on resequencing, concerning the $M/M/2$ queue. For any given m they introduce in consideration $m+1$ fixed position policies (FPP) $\pi_i(m),\ i = 1, ..., m+1$, which works in the following manner: If any job is in PB, the faster server must serve. If the waiting jobs in PB are more than m and the slower server becomes (or is) idle, it takes for service the job from position i. The authors derive explicit expressions for the steady state probability distribution of the triplet (L_{PB}, x_1, x_2), where x_i's are indicators (busy or not) of the two servers ($i = 1, 2$). Then for each of the two policies $\pi_1(m)$, and $\pi_{m+1}(m)$ they derive explicit expressions for the resequencing delay, queue lengths, and the proportion of jobs that will experience resequencing delays. It is proven that π_{m+1} always yields smaller proportion of resequenced jobs than π_1, provided that $\mu_1 > \mu_2$ is true. The polynomial decision function

$$f_m(z) = z^{m+2} + (m-1)z^2 - 2mz + m \tag{14}$$

determines the policy that yields minimum mean delay in steady state, for $z = \mu_1/(\mu_1 + \mu_2)$. The OP does not depend neither on the buffer size B (it needs to be at least $m+1$), nor on the load of the system, but on the relative service ratio z. Numerical examples show a comparison between performance measures of the two policies $\pi_1(m)$ and $\pi_{m+1}(m)$, and also how to use the decision rule above in determination of preference of one policy versus the other.

Iliadis and Lien (1993) continue the study of the $M/M/2/B$ systems under an arbitrary policy $\pi_i(m)$. Using an embedded Markov chain approach for an extended state process, they found that the steady state resequencing delay is determined by the survival function

$$P\{W_{RSB} > t\} = \frac{r}{\rho(1-p_B)}[z(z^i S - 1)e^{-\mu_2 t} + (1-z)\sum_{j=0}^{i-1}(1 - z^{i-j})\frac{(\mu_i t)^j}{j!}e^{-\mu_i t}].$$

Here $\rho = \lambda/(\mu_1 + \mu_2)$ is the traffic load, z is the relative service ratio, p_B is the loss probability, and r and S are explicit expressions given in terms of the steady state probabilities of the process. The OP $\pi_{m^*+1}(m^*)$ yields smallest fraction of resequenced jobs, if $\mu_1 > \mu_2$. Moreover, the following holds:

Let n^* be the integer part of the number $ln(1-z)/(ln\ z)$. Then for any given threshold $m < n^*$ the optimal FPP is $\pi_{m+1}(m)$, and it minimizes the mean resequencing delay; if $m \geq n^*$, then the optimal FPP is π_{n^*}. The OP depends on the PB's size B only when $B < n^*$. The impact of policies $\pi_i(m)$ on the resequencing delay of a job was numerically illustrated.

Ayoun and Rosberg (1991) also studied $M/M/2/B$ models with the question: are there other FPPs for which the policy $\pi_{n^*}(m)$ is still the optimal one? The authors conducted precise probability analysis to calculate performance characteristics under each of proposed new policies, which differ by the rule of dispatching a job from the queue to the slower server when it is getting free. Their logistic analysis and exact calculations revealed a large scale of feasible rerouting policies and discounted times (costs) of resequencing, for which, if the PB size $B \geq n^*$, then every threshold policy with a given m can be improved by rerouting jobs from position n^* to the slower server. This is an analytical work and no numerical examples were presented.

Sasase et al. (1992) analyzed the most general $\vec{M}/M/n/B$ model, where C classes of jobs form the input, and class-dependent thresholds policies are introduced. The fastest server 1 always serves the job from the head of the queue. Whenever server j becomes idle ($j = 2, ..., n$), and servers 1,2,...,$j-1$ are busy, server j starts processing a waiting job of class i taken from the head of the queue if and only if the total number of jobs in PB exceeds the given threshold number $m_{j-1,\,i}$; otherwise it stays idle. The order $m_{1,\,i} \leq m_{2,\,i} \leq ... \leq m_{n-1,\,i}$ is assumed. The resequenced jobs wait only for those of their own class that has been bypassed during its transmission. (Note, that in this model some class(es) of jobs may not be subject to resequencing.) The authors use steady state balance equations and the method of inclusion-exclusion to derive expressions for the mean queuing delay and the resequencing delay for each class and for the entire system. Also they mention a possibility two jobs of same class to start simultaneously being served on two different servers. This creates a new component in the study of resequencing delay, which has been completely analyzed. A numerical example illustrates the policy on the model $\vec{M}/M/2/B$, where jobs of class 2 are not resequenced. Compared to conventional $m-$threshold policy, it is shown that the class-dependent $(m_{1,1},\ m_{1,2})$ threshold policy is a way to reduce the resequencing delay within some specific classes of jobs.

Varma (1991a) demonstrates a new approach in the study of a $M/M/2/B$ resequenced system in order to obtain the probability distribution of L_{RSB}. By introducing a variable Z to show which of the two busy servers has started servicing earlier, he notices that the infinitesimal matrix of the extended process $(L_{PB}, x_1, x_2, L_{RSB}, Z)$, has a block diagonal structure. This yields a matrix-geometric solution for the steady state probabilities of the corresponding continuous time Markov chain. (This fact is suggested by Neuts(1981), and opened the horizon for more general studies.) Getting the marginal distributions of each component of the

extended process, and effectively calculating various performance measures is then a matter of suitable choice of algorithmic manipulations.

By making use of the matrix-geometric approach Varma suggests an effective way to determine the size of PB, or that of RSB, which warrants certain minimum percentage of losses, or to reduce over-waiting resequenced jobs.

Varma (1991b) also analyzed the $M/M/2/\infty$ model under threshold type policy that minimizes the end-to-end delay. He introduced the *cost per unit time* function $c(t) = L_{PB}(t) + x_1(t) + x_2(t) + L_{RSB}(t)$ under discounting rate α. A policy π^* is optimal if it minimizes the cost per policy $J(\pi) = \int_0^\infty e^{-\alpha t}\, c(t)dt$ in steady state conditions. The author uses specific results of stochastic dynamic programming (based on discrete structure of the components of $c(t)$), to derive that the OP does not depend on the number of jobs present in RSB. He found that keeping the faster server busy whenever there are jobs in PB is a step towards the OP. His conjecture is that the threshold FPP policy $\pi(n^*)$ is optimal for the new cost approach too.

5 The MAP/M/2 Models

Varma (1991a) possibly traced a new way in resequencing studies. It was seen that the *Matrix-analytic methods* can be helpful in effective computations and study of performance measures when explicit equations for the state probabilities are at hands. Almost all of the previous works with numerical components dealt with Poisson input and exponential service times, or were discrete time models. However, in communication engineering, packets may arrive in a very bursty and stochastic manner; the inter-arrival times may be dependent. This motivates the use of more general input processes, which can be modeled by Markov arrival process (MAP).

At Kettering University an intensive study of these models for resequenced systems is going. Bin and Chakravarthy (1997a) conducted a complete steady state analysis of $MAP/M/2$ systems with non-renewal input, two heterogeneous servers and threshold-type scheduling. Bin and Chakravarthy (1997b) continue their study to find the optimal allocation of messages for the same system in the sense of search for optimal policy, as it was stated previously by Iliadis and Lien (1988, 1993). It is assumed $\mu_1 > \mu_2$, one threshold number m, and policies $\pi_1(m)$, or $\pi_{m+1}(m)$ are used. By making use of the method of Matrix-Geometric solution, they involve a large scale state space $\vec{X} = (L_{PB}, X_1, X_2, \eta)$, where X_i indicates the state of i^{th} server; η indicates the phase of the MAP, $(1 \le \eta \le M)$. In this way the stationary distribution at arrival, and at arbitrary time is obtained as solution of a matrix equation. It allows one to calculate effectively the CDF of residual delay in RSB, the probability of a job to be delayed, to establish all necessary and sufficient conditions for minimizing the resequenced proportion, or expected system delay. A numerically stable algorithm for computing performance measures is developed. It is proven that policy $\pi_{m+1}(m)$ always yields a smaller fraction of resequenced jobs than $\pi_1(m)$. Also, there exists a *polynomial decision function* in $z = \mu_1/(\mu_1 + \mu_2)$ which determines the policy that yields the minimum delay. Its form is given by equation (1), found by Iliadis and Lien(1988) for the Markovian models.

Chakravarthy et al. (1995, 1998) gave another point of view in the analysis of queuing models with resequencing, namely by introducing randomized policy in directing the heading jobs to the idle servers. They consider the $MAP/M/2/B$ model with $\mu_1 > \mu_2$ and infinite capacity of RSB. The proposed randomized (probability) control policy $\pi(p)$ means that when a new arriving job finds both servers idle, with probability $p \in [0, 1]$ it is directed to server 1; otherwise it goes to server 2. The finite capacity of RSB may cause blocking of servers when RSB (if finite) is full. In addition another randomized policy $\pi(p, p_B)$ appears: When the blocked RSB is released, the waiting job from the head of the queue with probability $p_B \in [0, 1]$ is directed to server 1; otherwise it goes to server 2. Matrix-Geometric approach is applicable again. It gives the following results, the random vector $(L_{RSB}, x_1, x_2, L_{PB}, \varepsilon, \eta)$, where ε indicates which server serves the job entered earlier (and the order of this description is also important). The infinitesimal operator governing the process is obtained, and an explicit steady state analysis of the process is performed. Algorithmic procedures for calculation of the steady state probabilities of the vector $(L_{RSB}, x_1, x_2, L_{PB}, \varepsilon, \eta)$ are worked out. Stationary waiting time distribution of an admitted job is then obtained. The system performance measures $\mathbf{E}(L_{PB})$, $\mathbf{E}(L_{RSB})$, P_B, and the mean sojourn time in the system are effectively found and graphically illustrated. Numerical examples with correlated and uncorrelated inter-arrival times show some curious effects: There are values of admission probabilities p, and p_B which are neither 0, nor 1, which produce optimal performance of that particular model.

6 Conclusions

The resequencing problems in Queuing Theory have its specific scope of research. Mostly Markovian models have been studded. However, as it is shown in Section 3, some scheduling theory approaches indicate that

the solutions for Markovian case may be valid for systems with general service distributions which are not Markovian. The Matrix-Geometric approach, and MAP models offer new technical, theoretical and practical way in the study of resequencing. Threshold selection policies, preliminary partitioning for resequencing, group rerouting, windowing, multiple repeated services, non-traditional randomized control policies in resequenced system may reduce considerably the delay, and may improve the total performance of the system.

Questions on optimal control of multi-server queues with resequencing will keep researchers engaged in active and challenging problems in the light of booming global network service for many years to come.

References

[1] AYOUN S. & ROSBERG Z. (1991) Optimal routing to two parallel heterogeneous servers with resequencing. *IEEE Trans. on Automatic Control*, **36**, No. 12, 1436-1449.

[2] AYOUN S. (1989) *Optimal control of a queuing system with two heterogeneous servers with resequencing.* M.S. thesis, Dept. Electr. Eng., Technion, Haifa, Israel.

[3] BACCELLI F., GELENBE E., & PLATEAU B. (1984) An end-to-end approach to the resequencing problem. *Jour. of the Association for Computing Machinery*, **31**, No. 3, 474-485.

[4] BHARATH-KUMAR K. & KERMANI P. (1983) Analysis of a resequencing problem in communication network. In *Proc. IEEE INFOCOM*, San Diego, CA, Apr. 1983.

[5] BIN L. & CHAKRAVARTHY S. (1997a) A finite capacity queue with non-renewal and exponential dynamic group services. *ORSA Journal on Computing*, **9**, 276-287.

[6] BIN L., & CHAKRAVARTHY S. (1997b) Performance analysis of a resequencing model in telecommunications network. In *Fifth Internat. Conf. in Telecommun. Systems*, 633-652.

[7] BRONSTEIN O.I. & RYKOV V.V. (1965) On optimal priority disciplines in queueing systems. *Izv. AN USSR, Techn. Cyb.*, No. 6, 28-37 (in Russian).

[8] BUYUKKOC C., VARAIYA P. & WALRAND J. (1985) The $c\mu$ rule revised. *Advances in Applied Probability*, **17**, 237-238.

[9] CHAKRAVARTHY S., CHUKOVA S., & DIMITROV B. (1995) A stochastic model with resequencing of messages. *IEMS´95 Proceedings: Annual International Conference on Industry, Engineering, and Management Systems*, pp. 601-606.

[10] CHAKRAVARTHY S., CHUKOVA S., & DIMITROV B. (1998) Analysis of $MAP/M/2/K$ queuing model with infinite resequencing buffer. In: *Performance Evaluation* **31**, 211-228.

[11] CONWAY R.W., MAXWELL W.L., MILLER L.W. (1967) *Theory of scheduling.* Adisson. Wissley Mass.

[12] DANET A., DESPRES R., LeREST A., PICHON G. & RITZENTHALER S. (1976) The French public packet switching service: The TRANSPACK network. In: *Proc. 3rd Int. Comp. Commun. Conf.*, Toronto, Canada, 251-260.

[13] DIMITROV B., GREEN D. Jr., RYKOV V. (2001) On the influence of observability and controllability to the optimal control quality. In: *Proceedings of The 5-th International Conference on Optimization: Techniques and Applications (ICOTA 2001).* Hong Kong, **2**, 669-687.

[14] DIMITROV B. (1997) Queues with Re-sequencing. A survey and recent results. *Nonlinear Analysis, Theory, Methods & Applications*, **30**, No. 8, 5447-5456.

[15] GNEDENKO B.V. & KOENIG D. (Eds.) (1981) *Handbuch der Bedienungstheorie.* Berlin, Akademie-Verlag, **1, 2**.

[16] GRAY J. P. & McNEIL T. B. (1979) SNA multiple system networking. *IBM Syst. J.*, **18**, 263-279.

[17] HARRUS G. & PLATEAU B. (1991) Queuing analysis of a reordering issue. *IEEE Trans. on Software Engineering*, **SE-4**, No. 2, 113-122.

[18] HOWARD R. A. (1960) *Dynamic Programming and Markov Processes.* Wiley, New York.

[19] ILIADIS I. (1988) *Resequencing control and analysis in computer networks*. Ph.D. thesis, Dept. Electr. Eng., Columbia Univ., NY.

[20] ILIADIS I. & LIEN L. Y.-C. (1993) Resequencing control for a queuing system with two heterogeneous servers. *IEEE Trans. on Communications*, **41**, No 6, 951-961.

[21] ILIADIS I. & LIEN L. Y.-C. (1988) Resequencing delay for a queuing system with two heterogeneous servers under a threshold-type scheduling. *IEEE Trans. on Communications*, **36**, No 6, 692-702.

[22] KAMOUN F., KLEINROCK L. & MUNTZ R. (1981) Queuing analysis of reordering issue in a distributed database concurrency control mechanism. In: *Proc. 2^{nd} Int. Conf. on Distributed Computing Systems*, IEEE Press, 13-23.

[23] KITAEV M. YU. & RYKOV, V.V. (1995) *Controlled queueing systems*. CRC Press, New York.

[24] KRISHNAMOORTHI B. (1963) On Poisson queue with two heterogeneous servers. *Operations Research*, **11**, 321-330.

[25] LIN W. & KUMAR P. R. (1984) Optimal control of a queuing system with two heterogeneous servers. *IEEE Trans. on Automatic Control*, **29**, No. 8, 696-703.

[26] NEUTS M. F. (1981) *Matrix-Geometric Solution in Stochastic Models: An Algorithmic Approach*, The Johns Hopkins University Press, Baltimore, MD.

[27] NEUTS M. F. (1989) *Structured Stochastic Matrices of M/G/1 type and Their Applications*, Marcel Dekker, NY.

[28] NEUTS M. F. (1981) Modeling based on the Markovian Arrival Process. *IEICE Trans. on Communications*, **E75-B**, No. 12, 1255-1265.

[29] RYKOV V. & ERMOLAEV A. (1999) Model of technical equipment choice for date transmission network. In: *Proceedings of the International Conference DCCN'99*. November 9-13, 1999, Tel-Aviv. M.: IITP RAS, 35-44.

[30] RYKOV V., LEVNER E. (1998) On optimal allocation of jobs between heterogeneous servers. In: *Distributed Computer Communication Networks. Proceedings of the International Workshop*, June 16-19 1998, Moscow. IITP RAS, Moscow, 20-28.

[31] RYKOV V. (2000) On jobs allocation to heterogeneous servers for the probability delay minimization. In: *Simulation in Industry (12th European Simulation Symposium)*. Sept. 28-30, 2000. University of Hamburg. Germany, pp. 561-565.

[32] RYKOV V. (2001) Monotone Control of Queueing System with heterogeneous Servers. *QUESTA*, **37**, 391-403.

[33] SASASE I., NISHIO Y. & NAKAMURA H. (1992) The effect of message-class dependent threshold-type scheduling on the delay for the M/M/N queue. *IEEE Publication, ICC'92*, 506-510.

[34] SHACHAM N. & TOWSLEY D. (1991) Resequencing delay and buffer occupancy in selective repeat *ARQ* with multiple receivers. *IEEE Trans. on Communications*, **39**, No 6, 928-936.

[35] VARMA S. (1987) *Some problems in queuing systems with resequencing*. M.S. thesis, Univ. Maryland, College Park, MD.

[36] VARMA S. (1991a) Optimal allocation of customers in a two server queue with resequencing. *IEEE Trans. on Automatic Control*, **36**, No. 11, 1288-1293.

[37] VARMA S. (1991b) A matrix geometric solution to a resequencing problem. *Performance Evaluation*, **12**, 103-114.

[38] VEKLEROV E.B. (1971) On optimal priority disciplines in queueing systems. *Autom. and Remote Control*, N0. 6, 149-153 (in Russian).

[39] WASHBURN A. (1974) A multi-server queue with no passing. *Operations Research*, **22**, 428-434.

[40] YUM T.-S. P. & NGAI T.-Y. (1986) Resequencing of messages in communication networks. *IEEE Trans. on Communications*, **34**, No. 2, 143-149.