

OPEN-LOOP ROUTING OF N ARRIVALS TO M PARALLEL QUEUES

Rodolfo A. Milito

AT&T Bell Laboratories
Crawfords Corner Rd., Holmdel, NJ 07733
and Faculty of Management,
Rutgers University
rodolfo@buckaroo.att.com

Emmanuel Fernández-Gaucherand †

Systems and Industrial
Engineering Department
University of Arizona
Tucson, AZ 85721
emmanuel@sie.arizona.edu

Keywords: *routing, call allocation, Intelligent Networks, Pontryagin's maximum principle*

Abstract

Distribution of arrivals to queues in parallel is a basic problem with a number of applications in computer communication networks. In this paper, we consider the problem of routing N arrivals to M queues in parallel when no information other than the prior statistics are available for the decision process.

The work we present here differs from results in the literature in two significant ways: a) We address the non-stationary problem of open-loop routing for $N < \infty$ arrivals; b) we do not decompose the problem into an allocation phase, and a routing phase, but look instead for an overall optimal policy.

Exhaustive enumeration and dynamic programming turn out to be computationally unfeasible approaches for realistic values of the parameters N and M . Examination of the necessary conditions for optimality as given by Pontryagin's Maximum Principle leads to the formulation of a Policy Iteration Algorithm. The algorithm exploits the local nature of the Maximum Principle for computational advantage. Convergence of the algorithm in a finite number of iterations, as well as monotonicity results, are established. While only convergence to a local minimum is guaranteed, extensive computational experimentation points to its near-optimality.

An unabridged version of the paper in postscript form is available from rodolfo@buckaroo.att.com

1. Motivation

There is a rich literature on the problem of distributing arrivals to queues in parallel. Routing to parallel queues is one of the most basic, non-trivial paradigms in normative queueing problems. The study of these paradigms is often pursued in the hope that they enhance our understanding of more complex, frequently intractable, queueing networks.

† Research partially supported by The Engineering Foundation under grant RI-A-93-10; in part by the National Science Foundation under grant NSF-INT 9201430; and in part by a grant from The AT&T Foundation.

As it turns out, the problem of routing to parallel queues is not only a nice paradigm to foster understanding, but it is a good model of many practical problems in its own right. Foremost among these problems, is the distribution of calls in an Intelligent Network to multiple customer sites^{[1], [2], [3], [4]}.

Most papers in the literature assume that perfect state information is available to the decision maker. This is a restrictive assumption, often violated in practice. Our ultimate goal is to give a rigorous treatment to the problem of routing arrivals to parallel queues when measurements are taken only every N arrivals. As a fundamental step in that direction, we address the problem of open-loop routing N arrivals to M queues with exponential servers.

So far, the literature regarding open-loop routing to parallel queues has considered the system in steady state, and has decomposed the problem into two phases: a) Find the set $\{p_i, i=1,2,\dots,M\}$ of routing probabilities to the different queues $\{Q_i, i=1,2,\dots,M\}$, that would minimize the cost if randomized routing were used; b) find the best routing sequence that in the long run assigns fraction p_i of the arrivals to queue Q_i . While intuitively and practically appealing, there is no guarantee that this decomposition yields optimal results. For problem b) in this decomposition framework, Hajek^[5] found the *most regular*, optimal sequence for the case $M=2$, and a lower bound on the cost for $M>2$ ^[6]. Rosberg^[7] proposed a *golden ratio policy*, for general $M \geq 2$, that achieves a cost close to Hajek's lower bound. Arian and Levy^[8] developed generalized round robin algorithms that are provably optimal for $M=2$, and come very close to Hajek's lower bound for $M>2$. Section 3 of Levy et. al.^[2] proposes yet another extension of Hajek's algorithm, to allow for fractions $\{p_i\}$ that change over time.

The work we present here differs from the existing approaches in two significant ways: a) We address the non-stationary problem of open-loop routing for $N < \infty$ arrivals; b) we do not decompose the problem into an allocation phase, and a routing phase, but instead look for an overall optimal policy.

The paper is organized as follows. In Section 2 we formulate the problem, and in Section 3 we mention two common approaches that suffer from a dimensionality problem. This

discussion motivates Section 4, which constitutes the core of the paper. Section 4 starts with the application of the maximum principle to the problem formulated in Section 2. While there is an extensive bibliography related to the application of Markovian decision process to the control of queues (see, e.g., Bertsekas^[9], Kumar and Varaiya^[10], Walrand^[11], Tijms^[12] and references therein), applications of Pontryagin's maximum principle to queueing problems have been more limited, and have focused on controlling continuous variables (service rates^[13] or arrival rates^[14]). There are compelling reasons for selecting Markovian decision processes as the tool of choice when full state information is available, which is certainly not our case. The dimensionality problems associated with the methods examined in Section 3 prompt us to look into Pontryagin's maximum principle. The maximum principle brings to the surface a rich structure, which leads to the derivation of an implementable policy iteration algorithm, whose convergence is established. Results of computational experiments are discussed in Section 5, and future work is outlined in Section 6.

2. Problem formulation

Consider the problem of routing N arrivals to M parallel queues ($Q_i, i = 1, 2, \dots, M$), with the objective of minimizing a certain performance measure. The information available to the decision maker is limited to the system dynamics (service time and interarrival time distributions), and distribution of the initial number of customers in each queue. Under conditions on the interarrival and service times to be delineated in the sequel, the problem is cast as a *completely unobservable* controlled Markov chain. Therefore, an equivalent formulation can be given in terms of an *information state* (equivalently, sufficient statistics); see Bertsekas^[9], Kumar and Varaiya^[10], Arapostathis et. al^[15]. Hence, the decision process is open-loop, and the system dynamics are

$$\pi_{n+1}^{(i)} = \pi_n^{(i)} P^{(i)}(a_n), \quad (2.1a)$$

where $\pi_n^{(i)}$ denotes the distribution of the number in queue i just before the n^{th} arrival. Note that the transition probability matrix $P^{(i)}(\cdot)$ depends on the action a_n , where $a_n = i$ means that the n^{th} arrival is routed to Q_i . The finite action space is $A = \{1, 2, \dots, M\}^1$. Given the finite horizon, the search for an optimal routing sequence can be restricted to the set of *pure* strategies without loss of generality. For notational convenience, we concatenate the vectors $\pi_n^{(i)}$ together, and call the resulting row vector π_n . The evolution of the marginals of the M queues is given by

$$\pi_{n+1} = \pi_n P(a_n), \quad (2.1b)$$

where the system matrix $P(a_n)$ is a block diagonal aggregate of the individual matrices $P^{(i)}(a_n)$.

We assume that $Q_i, (i \in \{1, 2, \dots, M\})$ is served according to the first-in-first-out (FIFO) discipline by an exponential server that

operates at rate $\mu^{(i)}$,² and that the interarrival times are i.i.d. random variables of known distribution $F_A(x)$. The distribution at the n^{th} arrival epoch is denoted $\pi_n = (\pi_n^{(1)}, \dots, \pi_n^{(M)})$. Suppose that the underlying cost structure for queue Q_i is given by an arbitrary function $(i, a) \rightarrow c^{(i)}(j, a)$, where j is the number in Q_i , and a is the routing decision for the current arrival. Since j is not known to the decision maker, an equivalent expected cost is formulated in terms of $\pi_n^{(i)}$, as the inner product

$$g^{(i)}(\pi_n^{(i)}, a_n) = \pi_n^{(i)} c_{a_n}^{(i)}, \quad (2.2)$$

where

$$c_{a_n}^{(i)} = [c^{(i)}(0, a_n), c^{(i)}(1, a_n), \dots].$$

The cost $g(\pi_n, a_n)$ considered in our optimization problem is an additive function of the cost incurred by each individual queue, i.e.,

$$g(\pi_n, a_n) = \sum_{i=1}^M g^{(i)}(\pi_n^{(i)}, a_n). \quad (2.3)$$

Let us illustrate the cost structure with two performance measures of interest: the sojourn time of the incoming arrival, and the probability that the number in Q_i exceeds a given threshold t_i . For the sojourn time cost, we have

$$c_{a_n}^{(i)} = \begin{cases} [1, 2, 3, \dots]^T / \mu^{(i)} & \text{if } a_n = i \\ [0, 0, 0, \dots]^T & \text{otherwise} \end{cases} \quad (2.4)$$

If the interest is to penalize the probability that the number in Q_i reaches or exceeds threshold t_i , one sets the first $t_i - 1$ components of the vector $c_{a_n}^{(i)}$ to 0, and the rest to 1, if $a_n \neq i$. If $a_n = i$, the t_i^{th} component is also set to 1.

Our goal is to minimize the sum of the costs seen by the first N arrivals given an initial distribution π_0 at the epoch of the first arrival, i.e.,

$$G^*(\pi_0) := \min_{(a_0, \dots, a_{N-1})} G(\pi_0, a_0, \dots, a_{N-1}) \quad (2.5)$$

where

$$G(\pi_0, a_0, \dots, a_{N-1}) = \sum_{n=0}^{N-1} g(\pi_n, a_n), \quad (2.6)$$

subject to the dynamics (2.1).

2.1 Determination of the transition probability matrices

Let $q_n^{(i)}$ denote the number of customers in Q_i , including the one in service, just before the n^{th} arrival, and define

X_{n+1} = *potential number of departures from Q_i between arrivals*,

that is, the number of departures from Q_i during the interarrival time if the queue never emptied. The time between arrival n and arrival $n+1$ is denoted A_{n+1} . The state evolution of Q_i is given³ by a recursion of Lindley's type (see Milito^[16]) as:

2. State-dependent service rates allows modeling queues with multiple servers.

3. The random variables $X_n^{(i)}, X_{n+1}^{(i)}$ are not independent, because both of them depend on A_n . However, for any given routing sequence $\{a_n\}$, the marginal distributions of $\{q_n^{(i)}\} i=1, \dots, M$, can be computed based on the prior statistics.

1. A could also include 0, namely, the decision to block an arrival, when such a decision is admissible.

$$q_{n+1}^{(i)} = [q_n^{(i)} + \mathbf{1}(a_n=i) - X_{n+1}]^+, \quad (2.7)$$

where $\mathbf{1}(A)$ is the indicator function of the event A . We are interested in the characterization of the transition probability matrix $P^{(i)}(a_n)$. Towards this goal, define

$$v_j^{(i)} := P\{X_{n+1}^{(i)} = j\} = \int_0^{\infty} \frac{(\mu^{(i)} t)^j}{j!} e^{-\mu^{(i)} t} dF_A(t). \quad (2.8)$$

The dependence of the transition probability matrix $P^{(i)}(a_n)$ on action a_n is binary: either arrival n is routed to Q_i , or it is routed elsewhere. To simplify the notation, denote

$$P^{(i)}(a_n) = \begin{cases} \{p_{kl}^{(i)}(1)\} & \text{if } a_n = i \\ \{p_{kl}^{(i)}(0)\} & \text{otherwise,} \end{cases} \quad (2.9)$$

where

$$p_{kl}^{(i)}(1) = \begin{cases} 0 & \text{if } l > k+1 \\ v_{k+1-l}^{(i)} & \text{if } 0 < l \leq k+1. \\ 1 - \sum_{j=0}^k v_j^{(i)} & l=0 \end{cases} \quad (2.10)$$

and

$$p_{kl}^{(i)}(0) = \begin{cases} 0 & \text{if } l = k > 0 \\ v_{k-l}^{(i)} & \text{if } 0 < l \leq k. \\ 1 - \sum_{j=0}^{k-1} v_j^{(i)} & \text{if } l=0 \end{cases} \quad (2.11)$$

Simple expressions of $\{v_j^{(i)}\}$ can be obtained for a number of important distributions of the interarrival times, including deterministic, Erlangian, and uniform distributions.

The discussion throughout the main body of the paper is limited to queues with single, exponential servers. An adequate redefinition of the elements of the transition probability matrices extends the treatment to exponential servers with state-dependent rates.

3. Exhaustive enumeration, and dynamic programming approaches

Neither exhaustive enumeration nor dynamic programming are feasible methodologies for moderate sizes of M and N .

4. Approach based on the maximum principle

Recall that $G(\pi_0, a_0, \dots, a_{N-1})$ denotes the cost for an initial distribution π_0 and given sequence $(a_0, a_1, \dots, a_{N-1})$ (see equation 2.6). We seek a routing sequence that minimizes $G(\pi_0, \cdot)$ subject to the system dynamics determined by equations (2.1). Pontryagin's maximum principle (see, for instance, Bryson and Ho^[17] or Whittle^[18]) views such an optimization as a joint choice of $\{(\pi_n, a_{n-1}), n=0, 1, \dots, N-1\}$ subject to the constraints. To account for the constraints consider Lagrange multipliers $\xi_n, n=1, \dots, N$, and define

$$\begin{aligned} \tilde{G}(\pi_0, a_0, \dots, a_{N-1}) \\ = \sum_{n=0}^{N-1} [g(\pi_n, a_n) + (\pi_n P(a_n) - \pi_{n+1}) \xi_{n+1}]. \end{aligned} \quad (4.1)$$

We can now introduce the Hamiltonian function

$$H(\pi_n, \xi_{n+1}, a_n) = g(\pi_n, a_n) + \pi_n P(a_n) \xi_{n+1}. \quad (4.2)$$

Suppose that for an appropriate adjoint sequence $\{\bar{\xi}_n\}$ there exists an orbit $\{(\bar{\pi}_n, \bar{a}_{n-1})\}$ that minimizes (4.1). Pontryagin's maximum principle establishes that $\bar{\xi}_n, \bar{\pi}_n, \bar{a}_{n-1}$ necessarily satisfy

$$\bar{\pi}_{n+1} = \frac{\partial H(\bar{\pi}_n, \bar{\xi}_{n+1}, \bar{a}_n)}{\partial \xi_{n+1}} \Big|_{\bar{\xi}_{n+1}}, \quad (4.3a)$$

$$\bar{\xi}_n = \frac{\partial H(\bar{\pi}_n, \bar{\xi}_{n+1}, \bar{a}_n)}{\partial \pi_n} \Big|_{\bar{\pi}_n}, \quad (4.3b)$$

$$H(\bar{\pi}_n, \bar{\xi}_{n+1}, \bar{a}_n) = \min_{a_n \in \mathbf{A}} H(\bar{\pi}_n, \bar{\xi}_{n+1}, a_n). \quad (4.3c)$$

Recall that π_n is a vector obtained by concatenating the M vectors $\pi_n^{(i)}$, where $\pi_n^{(i)}$ is the probability distribution of the number in Q_i . Likewise, ξ_n decomposes in M vectors, each one associated with a particular queue:

$$\xi_n^T = \left[[\xi_n^{(1)}]^T \cdots [\xi_n^{(M)}]^T \right]. \quad (4.4)$$

We also note that as a result of the linear nature of the cost structure (2.3)

$$\frac{\partial g(\pi_n, a_n)}{\partial \pi_n^{(i)}} = c_{a_n}^{(i)}. \quad (4.5)$$

Therefore, the above necessary conditions specialize to

$$\bar{\pi}_{n+1}^{(i)} = \bar{\pi}_n^{(i)} P^{(i)}(\bar{a}_n) \quad (4.6a)$$

$$\bar{\xi}_n^{(i)} = c_{\bar{a}_n}^{(i)} + P^{(i)}(\bar{a}_n) \bar{\xi}_{n+1}^{(i)}, \quad n=0, \dots, N-1. \quad (4.6b1)$$

$$\bar{\xi}_N^{(i)} = 0 \quad (4.6b2)$$

$$\bar{a}_n = \arg \min_{a_n \in \mathbf{A}} \left\{ \sum_{i=1}^M \bar{\pi}_n^{(i)} c_{a_n}^{(i)} + \sum_{i=1}^M \bar{\pi}_n^{(i)} P^{(i)}(a_n) \bar{\xi}_{n+1}^{(i)} \right\}. \quad (4.6c)$$

Regarding (4.6b2) we observe that in general, $\xi_N^{(i)} = \frac{\partial K(\pi_N)}{\partial \pi_N}$, where $K(\pi_N)$ is the terminal cost. In our formulation $K(\cdot) \equiv 0$.

For future reference we define the myopic policy $\{\hat{a}_0, \hat{a}_1, \dots, \hat{a}_{N-1}\}$ as the sequence generated by the one step-ahead minimization

$$\hat{a}_n = \arg \min_{a_n \in \mathbf{A}} \left\{ \sum_i \pi_n^{(i)} c_{a_n}^{(i)} \right\} \quad n=0, 1, \dots, N-1. \quad (4.7)$$

If the stage cost is the sojourn time of the arrival (equation 2.4), the myopic policy coincides with the individually optimal policy^{[19], [11]}.

4.1 A Policy Iteration Algorithm

The necessary conditions for optimality dictated by the maximum principle suggest the following algorithm:

Algorithm 1:

Initialization: Set $a_n^{(0)} = \hat{a}_n, \pi_n^{(0)} = \hat{\pi}_n, n=0, 1, \dots, N-1$, where $\{\hat{a}_n\}$ is an arbitrary action sequence, and $\{\hat{\pi}_n\}$ is the pdf induced by this sequence with initial condition π_0 . The myopic policy (4.7) is a reasonable choice for this initial guess.

Step 1 (Evaluation): For the current action sequence $\{a_n^{(k)}\}$ find the adjoint sequence $\{\xi_n^{(k)}\}$ using equations (4.6b).

Step 2 (Policy Improvement): For every n , $0 \leq n < N$, find the minimizing $a_n^{(k+1)}$ (equation 4.6c) and the corresponding state $\pi_{n+1}^{(k+1)}$ (equation 4.6a). Resolve ties in favor of the old decision.

Step 3 (Test): Stop if $\{a_n^{(k+1)}\} = \{a_n^{(k)}\}$. Otherwise, go back to step 1, now based on the new action sequence $\{a_n^{(k+1)}\}$.

4.2 Analysis of convergence of the Policy Iteration Algorithm

Recall (2.6), and for a given action sequence $\{a_n\}$, recursively define the cost-to-go $G_{N-l}(\cdot)$:

$$G_N(\pi_0, a_0, \dots, a_{N-1}) = G(\pi_0, a_0, \dots, a_{N-1}) \quad (4.14a)$$

$$G_{N-l}(\pi_l, a_l, \dots, a_{N-1}) = g(\pi_l, a_l) \quad (4.14b)$$

$$+ G_{N-l-1}(\pi_l, P(a_l), a_{l+1}, \dots, a_{N-1}), \quad l=0, \dots, N-1,$$

$$\text{where} \quad G_{N-l}(\pi_l, a_l, \dots, a_{N-1}) = \sum_{n=l}^{N-1} g(\pi_n, a_n). \quad (4.15)$$

The cost of the $(N-l)$ horizon problem can be written in terms of its associated Hamiltonian:

$$G_{N-l}(\pi_l, a_l, \dots, a_{N-1}) = H_{N-l}(\pi_l, \xi_{l+1}, a_l) \quad (4.16)$$

The following Lemma will be used in the sequel.

Lemma 1: For any fixed π_l , action sequence $\{a_l^{(k)}, \dots, a_{N-1}^{(k)}\}$ and corresponding adjoint sequence $\{\xi_{l+1}^{(k)}, \dots, \xi_{N-1}^{(k)}\}$, consider the local minimization

$$\bar{a}_l = \arg \min_{i \in \mathcal{A}} \left(\sum_{i=1}^M \pi_l^{(i)} c_{a_l}^{(i)} + \sum_{j=1}^M \pi_l^{(j)} P^{(j)}(i) \xi_{l+1}^{(j)} \right), \quad (4.17)$$

and impose the condition that ties are resolved in favor of $a_l^{(k)}$. Then,

$$G_{N-l}(\pi_l, \bar{a}_l, a_{l+1}^{(k)}, \dots, a_{N-1}^{(k)}) \leq G_{N-l}(\pi_l, a_l^{(k)}, \dots, a_{N-1}^{(k)}), \quad (4.18)$$

where equality holds if and only if $\bar{a}_l = a_l^{(k)}$.

Lemma 1 follows easily from the fact that

$$G_{N-l}(\pi_l, \bar{a}_l, a_{l+1}^{(k)}, \dots, a_{N-1}^{(k)}) \quad (4.19)$$

$$= \min_{a_l} H(\pi_l, \xi_{l+1}^{(k)}, (a_l^{(k)}, \dots, a_{N-1}^{(k)}), a_l),$$

and neither π_l nor ξ_{l+1} depend on a_l .

Lemma 2: Let the sequence $\{a_n^{(k)}, k=0, 1, \dots\}$ be generated according to the prescription of Algorithm 1. Then,

$$G(\pi_0, a_0^{(k+1)}, \dots, a_{N-1}^{(k+1)}) \leq G(\pi_0, a_0^{(k)}, \dots, a_{N-1}^{(k)}), \quad (4.20)$$

where equality holds if and only if $\{a_n^{(k)}\} \equiv \{a_n^{(k+1)}\}$, i.e., if the algorithm has converged.

Proof: The proof is by induction. According to recursion (4.14) and Lemma 1

$$G_N(\pi_0, a_0^{(k+1)}, a_1^{(k)}, \dots, a_{N-1}^{(k)}) \quad (4.21)$$

$$= g(\pi_0, a_0) + G_{N-1}(\pi_0, P(a_0^{(k+1)}), a_1^{(k)}, \dots, a_{N-1}^{(k)})$$

$$\leq G_N(\pi_0, a_0^{(k)}, \dots, a_{N-1}^{(k)})$$

Apply now Lemma 1 iteratively to $G_{N-l}(\cdot)$, $l=1, \dots, N-1$, and the proposition follows.

Theorem 1: Algorithm 1 converges in a finite number of steps to a control sequence that is not inferior to the initial sequence. Furthermore, the cost monotonically decreases with each iteration.

Proof: The monotonicity of the cost has been proved in Lemma 2. The convergence follows from the monotonicity, and the fact that the policy space is finite.

5. Numerical results

To ameliorate the problem of local minima, we considered a variant of the basic Algorithm 1. This variant, which looks 2 steps ahead but accepts only the first decision, is called Algorithm 1a.

Our extensive experiments suggest that the algorithms converge to local minima that is not far from the global minimum. A plausible explanation for the mechanism at work is the following. Call allocation can be thought as consisting of two processes: the allocation of workload to each queue (i.e., determination of the number of arrivals to be sent to each queue), and sequencing (i.e., determination of the order in which the arrivals are to be routed). Gross deviations of the workload allocation from the optimal one should translate into considerably higher costs. Therefore, it is reasonable to expect that all stationary points of the algorithms yield allocations that are close to the optimal. Figure 1 provides experimental evidence to support this conjecture. The following parameters were chosen: arrival rate $\lambda=3$, service rates $\mu^{(1)} = \mu^{(2)} = 1$, $\mu^{(3)} = 2$, initial state $(5, 0, 5)$. The two sets of three plots on the upper side of the figure display the histograms of the cost for the initial policy, and the policies corresponding to Algorithm 1 and Algorithm 1a. Both sets display the same data, but on different scales. Algorithm 1 does well, but it is outperformed by Algorithm 1a. The distribution of the allocation for the randomly chosen initial policy is displayed in the three subsequent plots. The next three triplets display the histograms of the allocation of arrivals to Q_1 , Q_2 , and Q_3 for the initial policy, the policy generated by Algorithm 1, and the policy generated by Algorithm 1a. The distribution of the allocation is striking: in all cases, for both policies, Q_3 was sent 11 arrivals, in sharp contrast with the wide distribution exhibited by the initial (random) sequence. Of the remaining 10 arrivals, Q_1 received either 2 or 3 (and consequently, Q_2 got either 8 or 7). Therefore, the total workload to be processed by Q_1 , Q_2 , and Q_3 is either $(5+2)/1$, $(0+8)/1$, $(5+11)/2$, or $(5+3)/1$, $(0+7)/1$, $(5+11)/2$, respectively. (We do not suggest that equating the workload is always optimal; only that it works for this example.) The implication is that, at least in this example, both algorithms do a remarkable job in finding the right allocation. The fact that Algorithm 1a outperforms Algorithm 1 has to be attributed to a better choice of the sequence that implements the allocation. The random choice of the initial sequence was used to test the algorithms. For this example, a far better initial choice is the individually optimal sequence. For the chosen parameters, the individually optimal sequence yields a cost of 48.2324, with an allocation of $(2, 7, 12)$. Algorithm 1 so initialized converges to a sequence with allocation $(2, 8, 11)$ and cost 48.0658. The sequence generated by Algorithm 1a has the same allocation $(2, 8, 11)$, but does a better job at sequencing, yielding a lower cost of 47.9043. This tends to reinforce the conjecture that the set of policies yielded by the algorithms are nearly optimal from the point of view of workload allocation.

Our final example aims to dispell the notion that the myopic policy is always near optimal, and consequently, the algorithms developed here are of little practical interest. The criterion considered in the experiment presented in Fig. 2, penalizes the probability that the delay experienced by an arrival exceeds a certain threshold. The rationale is that beyond certain threshold impatient customers may abandon. This may well model human behavior, e.g., in the case of 1-800 calls, or a protocol mechanism with time-out (see Milito^[16]).

The following parameters were used in the experiment: arrival rate $\lambda=6$, service times $\mu^{(1)}=1$, $\mu^{(2)}=2$, $\mu^{(3)}=3$, horizon= 21 arrivals, initial state (5, 10, 30). Algorithm 1, initialized with the myopic policy, achieves a dramatic reduction of the cost in only two iterations. The interpretation of these results is not difficult. The system is initially loaded. In fact, the initial expected delay faced by the first arrival is either 5 or 10, depending on the routing decision. Compare this to the cost, which is the probability that the waiting time exceeds a delay of 5 units. The fact that $\lambda = \mu_1 + \mu_2 + \mu_3$ implies that the congestion is unlikely to be cleared for the duration of the horizon. The individually optimal policy chooses the best decision for each arrival, regardless of its impact on subsequent arrivals. The policy chosen by Algorithm 1 *sacrifices* the first 8 arrivals, which are likely to suffer long delays, to clear the way for the remaining 13 arrivals. By doing so, it achieves a 75% reduction from the cost of the individually optimal policy. Should blocking had been included in the admissible set, and should the cost of blocking not been excessive, Algorithm 1 would have considered it. Also note that balancing the workload is not a good idea in this example. Indeed, the individually optimal policy, which performs poorly, yields a more balanced load than that generated by Algorithm 1.

6. Conclusions

We have presented an approach to the problem of routing N arrivals to M parallel queues based on Pontryagin's maximum principle. A policy iteration algorithm has been proposed, and its convergence established. Furthermore, the cost decreases monotonically with each iteration. Extensive numerical experiments point to the near optimality of the algorithm. Finally, we showed that the myopic policy is not necessarily a good policy.

Future lines of research extend in three directions:

- i) The optimal stationary open-loop policy for the infinite horizon problem;
- ii) the optimal stationary policy for the infinite horizon problem, when measurements are made available every N arrivals;
- iii) the multiclass environment, with class dependent service time requirements.

ACKNOWLEDGEMENTS

Discussions with S. Durinovic-Johri, Y. Kogan, and Y. Levy from AT&T Bell Laboratories, and Professor Marcel Neuts from University of Arizona are greatly appreciated.

REFERENCES

1. R.A. Milito, Y. Levy, and Y. Arian, "Dynamic Algorithms for Distributed Queues with Abandonments," *Teletraffic and Datatraffic in a Period of Change, Proc. of 13th ITC*, pp. 329-334. Copenhagen, 1991.
2. Y. Levy, S. Durinovic-Johri, and R.A. Milito, "Dynamic Network Call Distribution with Periodic Updates," *Proc. of 14th ITC*, Antibes, France, June 1994.
3. R.B. Robrock II, "The Intelligent Network: Changing the Face of Telecommunications," *Proceedings of the IEEE*, vol. 79, pp. 7-20, 1991.
4. R.B. Robrock II, "The Intelligent Network," *The Electrical Eng. Handbook*, CRC Press, pp. 1468-1476, 1993.
5. B. Hajek, "Extremal Splittings of Point Processes," *Math. Oper. Res.*, vol.10, pp. 543-556, 1985.
6. B. Hajek, "The Proof of a Folk Theorem on Queueing Delay with Applications to Routing in Networks," *JACM*, vol. 30, n0. 4, October 1983, pp. 834-851.
7. Z. Rosberg, "Deterministic Routing to Buffered Channels," *IEEE Trans. Comm.*, vol. COM-34, n. 5, May 1986, pp. 504-507.
8. Y. Arian and Y. Levy, "Algorithms for Generalized Round Robin Routing", *Operations Research Letters* 12, 1992, pp. 313-319.
9. D. Bertsekas, *Dynamic Programming and Stochastic Control*, Academic Press
10. P.R. Kumar and P. Varaiya, *Stochastic Systems: Estimation, Identification and Adaptive Control*, Prentice-Hall, 1986.
11. J. Walrand, *An Introduction to Queueing Networks*, Prentice Hall, 1988.
12. H.C. Tijms, *Stochastic Modeling and Analysis: A computational approach*, J. Wiley & Sons, 1986.
13. C.F. Klein and W.A. Gruver, "Optimal Control of Markovian Queueing Systems," *Optimal Control Applications and Methods*, vol. 2, 1981, pp. 23-24.
14. F.T. Man, "Optimal Control of Time-Varying Queueing Systems," *Management Science*, vol. 19, n. 11, July 1973, pp. 1249-1256.
15. A. Arapostathis, V.S. Borkar, E. Fernández-Gaucherdan, M.K. Ghosh, and S.I. Marcus, "Discrete-Time Controlled Markov Processes with Average Cost Criterion: A Survey," *SIAM Journal of Control and Optimization*, vol. 31, 1993, pp. 282-344.
16. R.A. Milito, "On Arrival Driven Queueing Models: Correlated Arrivals, Traffic Policing and Admission Control," *Proceedings of the 32nd Conference on Decision and Control*, San Antonio, TX, December 1993, pp. 2743-2749.
17. A.E. Bryson, Y.C. Ho, *Applied Optimal Control*, J. Wiley & Sons, 1975.
18. P. Whittle, *Optimization Over Time: Dynamic Programming and Stochastic Control*, vol. I, J. Wiley & Sons, 1982.
19. S. Stidham Jr., "Optimal control of admission to a queueing system," *IEEE Trans. Autom. Cntrl.*, AC-30, 1985, pp. 705-713.

FIG. 1: COST AND WORKLOAD ALLOCATION

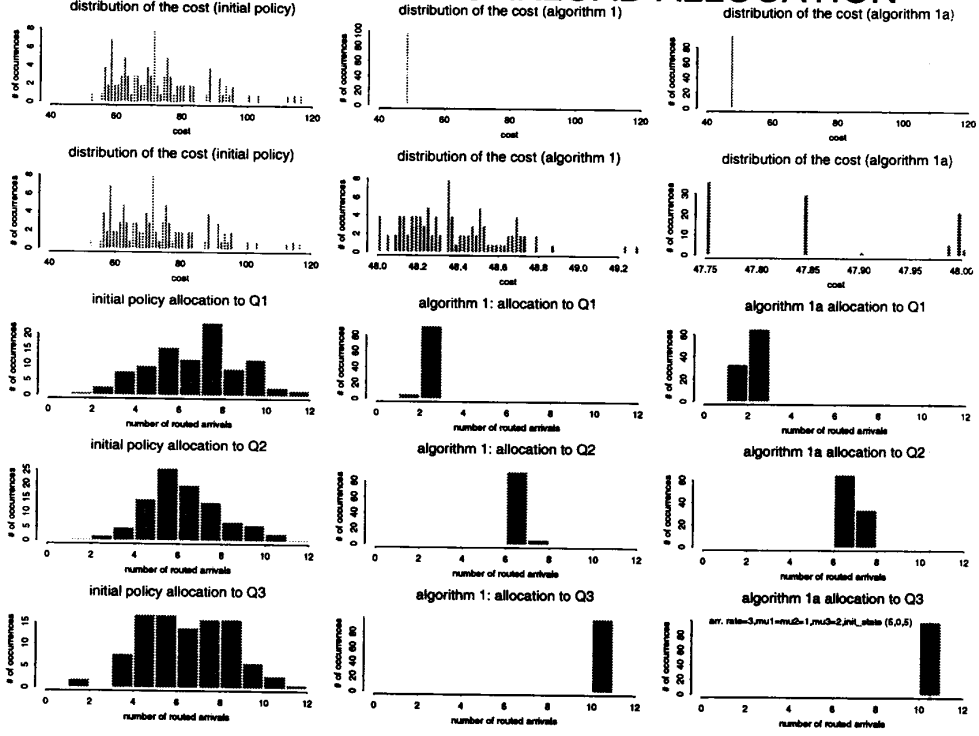


FIG. 2: Individually optimal policy not necessarily near-optimal

