Ordinal Optimization for a Class of Deterministic and Stochastic Discrete Resource Allocation Problems

Christos G. Cassandras, *Fellow, IEEE*, Liyi Dai, *Member, IEEE*, and Christos G. Panayiotou

***Abstract*— The authors consider a class of discrete resource allocation problems which are hard due to the combinatorial explosion of the feasible allocation search space. In addition, if no closed-form expressions are available for the cost function of interest, one needs to evaluate or (for stochastic environments) estimate the cost function through direct online observation or through simulation. For the deterministic version of this class of problems, the authors derive necessary and sufficient conditions for a globally optimal solution and present an online algorithm which they show to yield a global optimum. For the stochastic version, they show that an appropriately modified algorithm, analyzed as a Markov process, converges in probability to the global optimum. An important feature of this algorithm is that it is driven by ordinal estimates of a cost function, i.e., simple comparisons of estimates, rather than their cardinal values. They can therefore exploit the fast convergence properties of ordinal comparisons, as well as eliminate the need for "step size" parameters whose selection is always difficult in optimization schemes. An application to a stochastic discrete resource allocation problem is included, illustrating the main features of their approach.**

***Index Terms*—Discrete-event systems, resource allocation, stochastic optimization.**

I. Introduction

DISCRETE optimization problems often arise in the context of resource allocation. A classic example is the buffer (or kanban) allocation problem in queueing models of manufacturing systems [10], [20], where a fixed number of buffers (or kanban) must be allocated over a fixed number of servers to optimize some performance metric. Another example is the transmission scheduling problem in radio networks [5], [18], where a fixed number of time slots forming a "frame" must be allocated over several nodes. In the basic model we will consider in this paper, there are K identical *resources* to be allocated over N *user classes* so as to optimize some system performance measure (objective function). Let the resources be sequentially indexed so that the "state" or "allocation" is represented by the K-dimensional vector $\boldsymbol{s} = [s_1, \cdots, s_K]^T$, where $s_j \in \{1, \cdots, N\}$ is the user class index assigned to resource j. Let $\mathcal{S}$ be the finite set of feasible resource allocations

$$\mathcal{S} = \{[s_1, \cdots, s_K]\colon\ s_j \in \{1, \cdots, N\}\}$$

where "feasible" means that the allocation may have to be chosen to satisfy some basic requirements such as stability or fairness. Let $L_i(\boldsymbol{s})$ be the class i cost associated with the allocation vector $\boldsymbol{s}$. The class of *resource allocation* problems we consider is formulated as

$$(\boldsymbol{RA}\mathbf{1}) \quad \min_{\boldsymbol{s}\in\mathcal{S}} \sum_{i=1}^{N} \beta_i L_i(\boldsymbol{s})$$

where β_i is a weight associated with user class i. $(\boldsymbol{RA}\mathbf{1})$ is a special case of a nonlinear integer programming problem (see [14], [16], and references therein) and is in general NP-hard [14]. However, in some cases, depending upon the form of the objective function (e.g., separability, convexity), efficient algorithms based on finite-stage dynamic programming or generalized Lagrange relaxation methods are known (see [14] for a comprehensive discussion on aspects of deterministic resource allocation algorithms). Alternatively, if no *a priori* information is known about the structure of the problem, then some form of a search algorithm is employed (e.g., simulated annealing [1], genetic algorithms [13]).

In general, the system we consider operates in a stochastic environment; for example, users may request resources at random time instants or hold a particular resource for a random period of time. In this case, $L_i(\boldsymbol{s})$ in $(\boldsymbol{RA}\mathbf{1})$ becomes a random variable, and it is usually replaced by $E[L_i(\boldsymbol{s})]$. Moreover, we wish to concentrate on complex systems for which no closed-form expressions for $L_i(\boldsymbol{s})$ or $E[L_i(\boldsymbol{s})]$ are available. Thus, $E[L_i(\boldsymbol{s})]$ must be estimated through Monte Carlo simulation or by direct measurements made on the actual system. Problem $(\boldsymbol{RA}\mathbf{1})$ then becomes a stochastic optimization problem over a discrete state space.

While the area of stochastic optimization over *continuous* decision spaces is rich and usually involves gradient-based techniques as in several well-known stochastic approximation algorithms [15], [17], the literature in the area of *discrete* stochastic optimization is relatively limited. Most known approaches are based on some form of random search, with the added difficulty of having to *estimate* the cost function at every step. Such algorithms have been recently proposed by Yan and Mukai [19] and Gong *et al*. [9]. Another recent contribution to this area involves the *ordinal* optimization approach presented in [11]. Among other features, this approach is intended

Manuscript received March 28, 1997; revised November 4, 1997. Recommended by Associate Editor, E. K. P. Chong. This work was supported in part by the National Science Foundation under Grants EEC-95-27422 and ECS-9624279, by AFOSR under Contract F49620-95-1-0131, and by the Air Force Rome Laboratory under Contract F30602-95-C-0242.

C. G. Cassandras is with the Department of Manufacturing Engineering, Boston University, Boston, MA 02215 USA.

L. Dai is with the Department of Systems Science and Mathematics, Washington University, St. Louis, MO 63130 USA.

C. G. Panayiotou is with the Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, MA 01003 USA.

Publisher Item Identifier S 0018-9286(98)04895-8.

to exploit the fact that ordinal estimates are particularly robust with respect to estimation noise compared to cardinal estimates (see also [7]); that is, estimating the correct order of two costs based on noisy measurements is much easier than estimating the actual values of these costs. The implication is that convergence of such algorithms is substantially faster. These recent contributions are intended to tackle stochastic optimization problems of arbitrary complexity, which is one reason that part of the ordinal optimization approach in [11] includes a feature referred to as "goal softening." On the other hand, exploiting the structure of some resource allocation problems can yield simpler optimization schemes which need not sacrifice full optimality. For example, in [3] an approach is proposed whereby introducing auxiliary control variables, the original discrete optimization problem $(\boldsymbol{RA1})$ is transformed into a continuous optimization problem. The latter may then be solved through a variant of a stochastic approximation algorithm.

In this paper, we first consider the deterministic version of problem $(\boldsymbol{RA1})$ for a class of systems that satisfy the separability and convexity assumptions, A1) and A2), respectively, defined in Section II. Subsequently, we provide a necessary and sufficient condition for global optimality, based on which we develop an optimization algorithm. We analyze the properties of this algorithm and show that it yields a globally optimal allocation in a finite number of steps. We point out that, unlike resource allocation algorithms presented in [14], an important feature of the proposed algorithm is that every allocation in the optimization process remains feasible so that our scheme can be used *online* to adjust allocations as operating conditions (e.g., system parameters) change over time. Next, we address the stochastic version of the resource allocation problem. By appropriately modifying the deterministic algorithm, we obtain a stochastic optimization scheme. We analyze its properties treating it as a Markov process and prove that it converges in probability to a globally optimal allocation under mild conditions.

As will be further discussed in the sequel, two features of the resource allocation scheme we analyze are worth noting because of their practical implications. All iterative reallocation steps are driven by ordinal comparisons, which, as mentioned earlier, are particularly robust with respect to noise in the estimation process. Consequently: 1) as in other ordinal optimization schemes (e.g., [11] and [12]), convergence is fast because short estimation intervals are adequate to guide allocations toward the optimal and 2) there is no need for "step size" or "scaling" parameters which arise in algorithms driven by cardinal estimates of derivatives or finite differences; instead, based on the result of comparisons of various quantities, allocations are updated by reassigning one resource with respect to the current allocation. This avoids the difficult practical problem of selecting appropriate values for these parameters, which are often crucial to the convergence properties of the algorithm.

The rest of the paper is organized as follows. In Section II, we use Assumption A1) to transform $(\boldsymbol{RA1})$ to an equivalent problem $(\boldsymbol{RA2})$. Subsequently, we consider the deterministic version of problem $(\boldsymbol{RA2})$ and present a characterization of the optimal allocation under certain conditions. In Section III, we propose an iterative descent algorithm and show convergence to a globally optimal allocation in a finite number of steps. In Section IV, we treat the stochastic version of the problem and develop an algorithm for solving it. We analyze the algorithm as a Markov process and show that it converges in probability to a globally optimal allocation. In Section V, we present an application to a stochastic resource allocation problem and illustrate the properties of our approach through several numerical results. We conclude with Section VI, where we summarize the work done and identify further research directions in this area.

II. Characterization of Optimal Allocations

In order to specify the class of discrete resource allocation problems we shall study in this paper, we define

$$n_i = \sum_{j=1}^{K} \mathbf{1}[s_j = i] \qquad i = 1, \cdots, N \tag{1}$$

where $\mathbf{1}[\cdot]$ is the standard indicator function and n_i is simply the number of resources allocated to user class i under some allocation $\boldsymbol{s}$. We shall now make the following assumption

A1) $L_i(\boldsymbol{s})$ depends only on the number of resources assigned to class i, i.e., $L_i(\boldsymbol{s}) = L_i(n_i)$.

This assumption asserts that resources are indistinguishable, as opposed to cases where the identity of a resource assigned to user i affects that user's cost function. Even though A1) limits the applicability of the approach to a class of resource allocation problems, it is also true that this class includes a number of interesting problems. Examples include: 1) buffer allocation in *parallel* queueing systems where the blocking probability is a function of the number of buffer slots assigned to each server (for details, see Section V); note, however, that A1) does not hold in the case of a *tandem* queueing system; 2) cellular systems where the call loss probability of each cell depends only on the number of channels assigned to each cell; and 3) scheduling packet transmissions in a mobile radio network, where the resources are the *time slots* in a transmission frame (see [5] and [18]).

Under A1), we can see that an allocation written as the K-dimensional vector $\boldsymbol{s} = [s_1, \cdots, s_K]$, can be replaced by the N-dimensional vector $\boldsymbol{s} = [n_1, \cdots, n_N]$. In this case, the resource allocation problem $(\boldsymbol{RA1})$ is reformulated as

$$(\boldsymbol{RA2}) \quad \min_{\boldsymbol{s} \in \mathcal{S}} \sum_{i=1}^{N} \beta_i L_i(n_i) \quad \text{s.t.} \sum_{i=i}^{N} n_i = K.$$

The cardinality of the state space involved in $(\boldsymbol{RA2})$ is given by $|\mathcal{S}| = ((K + N - 1)!/K!(N - 1)!)$, so that an exhaustive search is generally not feasible. Deterministic resource allocation problems with a separable cost function have been studied in the literature (for a thorough treatment see [14]). Several algorithms based on the theory of generalized Lagrange multipliers are presented in [14, Ch. 4] where the optimal solution can be determined in polynomial time. These algorithms are based on relaxing the resource constraint so that the determination of an optimal solution is based on examining several infeasible allocations. Moreover, all relevant information (in the form of an individual user cost vector $[\Delta L_1(1), \cdots, \Delta L_1(K), \cdots, \Delta L_N(1), \cdots, \Delta L_N(K)]$) is obtained prior to the optimization procedure. In other words, these are intended to be *offline* algorithms. As mentioned in the previous section, however, our ultimate goal is to solve stochastic resource allocation problems where the cost function is not available in closed-form. This requires that: 1) we resort to estimates of $L_i(n_i)$ and $\Delta L_i(n_i)$ for all $i = 1, \cdots, N$ over some observation period and 2) we iterate after every such observation period by adjusting the allocation which, therefore, *must remain feasible* at every step of this process. It is for this reason that we wish to derive *online* discrete optimization algorithms. We shall first deal with issue 2) above in Sections II and III. We will then address issue 1) in Section IV.

In addition to A1), we will make the following assumption regarding the cost functions of interest:

A2) For all $i = 1, \cdots, N, L_i(n_i)$ is such that $\Delta L_i(n_i + 1) > \Delta L_i(n_i)$

where

$$\Delta L_i(n_i) = L_i(n_i) - L_i(n_i - 1), \qquad n_i = 1, \cdots, K \quad (2)$$

with boundary values

$$\Delta L_i(0) \equiv -\infty \quad \text{and} \quad \Delta L_i(N+1) \equiv \infty.$$

This assumption is the analog of the usual convexity/concavity requirement for the vast majority of gradient-driven optimization over continuous search spaces. It is the assumption that typically allows an extremum to be a global optimum. The alternative is to settle for local optima. From a practical standpoint, most common performance criteria in systems where resource allocation arises are quantities such as throughput, mean delay, and blocking probability which generally satisfy such properties.

In what follows, there are two key results we will present. Our first result is a necessary and sufficient condition for global optimality in ($\boldsymbol{RA2}$). The second result is a necessary condition for global optimality in ($\boldsymbol{RA2}$) which requires an additional technical assumption in order to also become sufficient. Let $\boldsymbol{s}^*$ denote a solution of the optimization problem ($\boldsymbol{RA2}$), i.e., $\boldsymbol{s}^*$ is such that

$$L(\boldsymbol{s}^*) \le L(\boldsymbol{s}) \quad \text{for all } \boldsymbol{s} = [n_1, \cdots, n_N] \in \mathcal{S} \text{ s.t. } \sum_{i=i}^{N} n_i = K. \quad (3)$$

For simplicity, let $\beta_i = 1$ in ($\boldsymbol{RA2}$) for all $i = 1, \cdots, N$ for the remainder of the paper. (If it is required to have $\beta_i \neq 1$, then one can include that in the definition of L_i.)

Theorem 1: Under Assumptions A1) and A2), an allocation $\bar{\boldsymbol{s}} = [\overline{n}_1, \cdots, \overline{n}_N]$ is a global optimum [i.e., a solution of ($\boldsymbol{RA2}$)] if and only if

$$\Delta L_i(\overline{n}_i + 1) \ge \Delta L_j(\overline{n}_j) \quad \text{for any } i, j = 1, \cdots, N. \quad (4)$$

Proof: First, define the set

$$B(\boldsymbol{s}) = \{\boldsymbol{s}\text{: } \boldsymbol{s} = [n_1, \cdots, n_i + 1, \cdots, n_j - 1, \cdots, n_N] \text{ for some } i \neq j\}$$

which includes all feasible neighboring points to $\boldsymbol{s} = [n_1, \cdots, n_N]$, i.e., vectors which differ from $\boldsymbol{s}$ by $+1$ and -1 in two distinct components (recall that $n_1 + \cdots + n_N = K$). To prove that (4) is a necessary condition, assume that $\bar{\boldsymbol{s}}$ is a global optimum. Then, from (3), it is clear that $L(\bar{\boldsymbol{s}}) \le L(\boldsymbol{s}')$ for all $\boldsymbol{s}' \in B(\bar{\boldsymbol{s}})$. From this we can write

$$\sum_{i=1}^{N} L_i(\overline{n}_i) \le L_1(\overline{n}_1) + \cdots + L_i(\overline{n}_i + 1) + \cdots + L_j(\overline{n}_j - 1) + \cdots + L_N(\overline{n}_N)$$

or

$$L_i(\overline{n}_i) + L_j(\overline{n}_j) \le L_i(\overline{n}_i + 1) + L_j(\overline{n}_j - 1)$$

and, therefore

$$\Delta L_i(\overline{n}_i + 1) \ge \Delta L_j(\overline{n}_j) \quad \text{for any } i, j. \quad (5)$$

To prove the sufficiency of (4), let $\bar{\boldsymbol{s}} = [\overline{n}_1, \cdots, \overline{n}_N]$ be an allocation that satisfies (4), and let $\boldsymbol{s}^* = [n_1^*, \cdots, n_N^*]$ be a global optimum. Therefore, $[n_1^*, \cdots, n_N^*]$ satisfies (5), i.e.,

$$\Delta L_i(n_i^* + 1) \ge \Delta L_j(n_j^*) \quad \text{for any } i, j. \quad (6)$$

Let $n_i^* = \overline{n}_i + d_i$ for all $i = 1, \cdots, N$, where $d_i \in \{-K, \cdots, -1, 0, 1, \cdots, K\}$ and subject to the constraint

$$\sum_{j=1}^{N} d_j = 0$$

which follows from the constraint $n_1 + \cdots + n_N = K$. Then, define the set $\mathcal{A} = \{i\text{: } d_i = 0\}$. There are now two cases depending on the cardinality $|\mathcal{A}|$ of this set.

Case 1: $|\mathcal{A}| = N$. In this case we have $\overline{n}_i = n_i^*$ for all i, so that, trivially, $\bar{\boldsymbol{s}} \equiv \boldsymbol{s}^*$.

Case 2: $|\mathcal{A}| \neq N$. This implies that there exist indexes i, j such that $d_i > 0$ and $d_j < 0$. Therefore, we can write the following ordering:

$$\Delta L_j(\overline{n}_j + d_j + 1) \ge \Delta L_i(\overline{n}_i + d_i) \ge \Delta L_i(\overline{n}_i + 1) \ge \Delta L_j(\overline{n}_j) \quad (7)$$

where the first inequality is due to (6), the second is due to A2), and the third is due to our assumption that $\bar{\boldsymbol{s}}$ satisfies (4). However, for $d_j \le -2$, using A2), we have $\Delta L_j(\overline{n}_j) > \Delta L_j(\overline{n}_j + d_j + 1)$ which contradicts (7). It follows that for an allocation to satisfy (4) only $d_j = -1$ is possible, which in turn implies that (7) holds in equality, i.e.,

$$\Delta L_j(\overline{n}_j) = \Delta L_i(\overline{n}_i + d_i) = \Delta L_i(\overline{n}_i + 1). \quad (8)$$

Using A2), this implies that $d_i = 1$.

This argument holds for any (i, j) pair; therefore, we conclude that the only possible candidate allocations $\bar{\boldsymbol{s}}$ satisfying (4) are such that

$$\Delta L_i(\overline{n}_i + 1) = \Delta L_j(\overline{n}_j) \quad \text{for all } i, j \notin \mathcal{A}, d_i = 1, \; d_j = -1. \quad (9)$$

Let the difference in cost corresponding to $\bar{\boldsymbol{s}}$ and $\boldsymbol{s}^*$ be $\Delta(\bar{\boldsymbol{s}}, \boldsymbol{s}^*)$. This is given by

$$\begin{aligned}\Delta(\bar{\boldsymbol{s}}, \boldsymbol{s}^*) &= \sum_{i=1}^{N}[L_i(\overline{n}_i) - L_i(n_i^*)] \\ &= \sum_{\substack{i=1\\ i\not\in\mathcal{A}}}^{N}[L_i(\overline{n}_i) - L_i(\overline{n}_i + d_i)] \\ &= \sum_{\substack{i=1\\ i\not\in\mathcal{A}\\ d_i=-1}}^{N} \Delta L_i(\overline{n}_i) - \sum_{\substack{i=1\\ i\not\in\mathcal{A}\\ d_i=1}}^{N} \Delta L_i(\overline{n}_i + 1) = 0\end{aligned}$$

where in the last step we use (9) and $\Sigma_{j=1}^{N} d_j = 0$. This establishes that if $\bar{\boldsymbol{s}} = [\overline{n}_1, \cdots, \overline{n}_N]$ satisfies (4), then either $\bar{\boldsymbol{s}} \equiv \boldsymbol{s}^*$ as in Case 1 or it belongs to a set of equivalent optimal allocations that satisfy the equality in (3). ■

Note that Theorem 1 gives a necessary and sufficient condition that the optimal allocation must satisfy in terms of the cost differences $\Delta L_i(\cdot)$ for $i = 1, \cdots, N$ in only a small set of feasible allocations, namely the neighborhood of the optimal allocation $B(\boldsymbol{s}^*)$.

Next, we will derive a different necessary and sufficient condition for global optimality in solving (**RA2**), expressed in terms of $\max_{i=1,\cdots,N}\{\Delta L_i(n_i)\}$. As we will see in the proof of Theorem 2, necessity still relies on Assumptions A1) and A2) alone, but sufficiency requires an additional technical condition.

A3) Let $[\overline{n}_1, \cdots, \overline{n}_N]$ be an allocation such that

$$\max_{i=1,\cdots,N}\{\Delta L_i(\overline{n}_i)\} \leq \max_{i=1,\cdots,N}\{\Delta L_i(n_i)\}$$

for all $\boldsymbol{s} = [n_1, \cdots, n_N] \in \mathcal{S}$. If $i^* = \arg\max_{i=1,\cdots,N}\{\Delta L_i(\overline{n}_i)\}$, then

$$\Delta L_{i^*}(\overline{n}_{i^*}) > \Delta L_j(\overline{n}_j)$$

for all $j = 1, \cdots, N$, $j \neq i^*$.

This assumption guarantees a unique solution to (**RA2**) and, as mentioned above, it is only used to prove sufficiency of Theorem 2. If the condition is violated, i.e., there is a set of optimal allocations, then, in the deterministic case, the algorithm will converge to one member of the set dependent on the initial allocation. In the stochastic case, the algorithm will oscillate between the members of the set as mentioned in the remark at the end of Section IV.

Theorem 2: Under assumptions A1) and A2), if an allocation $\bar{\boldsymbol{s}} = [\overline{n}_1, \cdots, \overline{n}_N]$ is a global optimum [i.e., a solution of (**RA2**)] then

$$\max_{i=1,\cdots,N}\{\Delta L_i(\overline{n}_i)\} \leq \max_{i=1,\cdots,N}\{\Delta L_i(n_i)\} \tag{10}$$

for all $\boldsymbol{s} = [n_1, \cdots, n_N] \in \mathcal{S}$. If in addition A3) holds, then (10) also implies that $\bar{\boldsymbol{s}}$ is a solution of (**RA2**).

Proof: Suppose that $\bar{\boldsymbol{s}}$ is a global optimum, and consider an allocation $\boldsymbol{s} = [n_1, \cdots, n_N]$ such that $\boldsymbol{s} \neq \bar{\boldsymbol{s}}$. We can then express n_i $(0 \leq n_i \leq K)$ as

$$n_i = \overline{n}_i + d_i \quad \text{for all } i = 1, \cdots, N$$

where $d_i \in \{-K, \cdots, -1, 0, 1, \cdots, K\}$ and subject to

$$\sum_{j=1}^{N} d_j = 0 \tag{11}$$

which follows from the fact that $[\overline{n}_1, \cdots, \overline{n}_N]$ is a feasible allocation. Let $i^* = \arg\max_{i=1,\cdots,N}\{\Delta L_i(\overline{n}_i)\}$. If $\boldsymbol{s} \neq \bar{\boldsymbol{s}}$, it follows from (11) that there exists some j such that $d_j > 0$ and two cases arise.

Case 1: If $j = i^*$, then

$$\begin{aligned}&\max_{i=1,\cdots,N}\{\Delta L_i(n_i)\} \\ &\quad \geq \Delta L_j(n_j) = \Delta L_{i^*}(\overline{n}_{i^*} + d_{i^*}) > \Delta L_{i^*}(\overline{n}_{i^*})\end{aligned}$$

where the last step is due to A2) since $d_{i^*} > 0$.

Case 2: If $j \neq i^*$, then first apply Theorem 1 to the optimal allocation $\bar{\boldsymbol{s}}$ to get

$$\Delta L_j(\overline{n}_j + 1) \geq \Delta L_{i^*}(\overline{n}_{i^*}). \tag{12}$$

Then, we can write the following:

$$\begin{aligned}\max_{i=1,\cdots,N}\{\Delta L_i(n_i)\} &\geq \Delta L_j(n_j) = \Delta L_j(\overline{n}_j + d_j) \\ &\geq \Delta L_j(\overline{n}_j + 1) \geq \Delta L_{i^*}(\overline{n}_{i^*})\end{aligned}$$

where the second inequality is due to A2) and the fact that $d_j \geq 1$, and the last inequality is due to (12). Hence, (10) is established.

Next, we show that if an allocation $\bar{\boldsymbol{s}}$ satisfies (10) and A3) holds, it also satisfies (4), from which, by Theorem 1, we conclude that the allocation is a global optimum. Let $i^* = \arg\max_{i=1,\cdots,N}\{\Delta L_i(\overline{n}_i)\}$ and suppose that (4) does not hold. Then, there exists a $j \neq i^*$ such that

$$\Delta L_j(\overline{n}_j + 1) < \Delta L_{i^*}(\overline{n}_{i^*}) = \max_{i=1,\cdots,N}\{\Delta L_i(\overline{n}_i)\}. \tag{13}$$

Note that if no such j were to be found, we would have $\Delta L_j(\overline{n}_j + 1) \geq \Delta L_{i^*}(\overline{n}_{i^*}) > \Delta L_k(\overline{n}_k)$ for all j, k [because of A3)] and we would not be able to violate (4) as assumed above.

Now, without loss of generality, let $i^* = 1$ and $j = N$ [j satisfying (13)]. Then, using A2), A3), and (13), the feasible allocation $[\overline{n}_1 - 1, \overline{n}_2, \cdots, \overline{n}_{N-1}, \overline{n}_N + 1]$ is such that

$$\begin{aligned}&\Delta L_1(\overline{n}_1) \\ &\quad = \max\{\Delta L_1(\overline{n}_1), \cdots, \Delta L_N(\overline{n}_N)\} \\ &\quad > \max\{\Delta L_1(\overline{n}_1 - 1), \Delta L_2(\overline{n}_2), \cdots, \Delta L_N(\overline{n}_N + 1)\}\end{aligned}$$

which contradicts (10) for the feasible allocation $[\overline{n}_1 - 1, \overline{n}_2, \cdots, \overline{n}_{N-1}, \overline{n}_N + 1]$ and the theorem is proved. ■

As already pointed out, A3) is not required in proving the necessity part of the theorem, but only the sufficiency. Also, note that Theorem 2 provides a characterization of an optimal allocation in terms of only the largest $\Delta L_i(\cdot)$ element in the allocation. What is interesting about (10) is that it can be interpreted as the discrete analog to continuous variable optimization problems. In such problems with equality constraints, it is well known that an optimal solution is characterized in terms of the partial derivatives of the cost function with respect to control variables (e.g., allocations expressed as nonnegative real numbers); specifically, at the optimal point all partial derivatives must be equal (e.g., see [8]). In order to derive a similar result for a discrete optimization problem, one must replace derivatives by finite cost differences, such as the quantities $\Delta L_i(\cdot), i = 1, \cdots, N$, defined in (2). The next "best thing" to equality in dealing with cost differences taken from a discrete set is to keep these differences as close as possible. This is expressed in terms of the maximum value of such finite differences at the optimal point in condition (10).

Having established some necessary and sufficient conditions that characterize the optimal allocation, namely Theorems 1 and 2, our next task is to develop an algorithm that iteratively adjusts allocations on line. These conditions then serve to determine a stopping condition for such an algorithm, guaranteeing that an optimal allocation has been found. In the next section, we propose such an algorithm taking advantage of (4) in Theorem 1.

III. Online Deterministic Optimization Algorithm

In this section, we present an iterative process for determining a globally optimal allocation and study its properties, which include a proof of convergence to such an allocation. In particular, we generate sequences $\{n_{i,k}\}, k = 0, 1, \cdots$ for each $i = 1, \cdots, N$ as follows. We define a set $\mathcal{C}_0 = \{1, \cdots, N\}$ and initialize all sequences so that an allocation $\boldsymbol{s}_0 = [n_{1,0}, \cdots, n_{N,0}]$ is feasible. Then, let

$$n_{i,k+1} = \begin{cases} n_{i,k} - 1, & \text{if } i = i_k^* \text{ and } \delta_k > 0 \\ n_{i,k} + 1, & \text{if } i = j_k^* \text{ and } \delta_k > 0 \\ n_{i,k}, & \text{otherwise} \end{cases} \tag{14}$$

where i_k^*, j_k^*, δ_k, and $\mathcal{C}_k$ are defined as follows:

$$i_k^* = \arg\max_{i \in \mathcal{C}_k} \{\Delta L_i(n_{i,k})\} \tag{15}$$

$$j_k^* = \arg\min_{i \in \mathcal{C}_k} \{\Delta L_i(n_{i,k})\} \tag{16}$$

$$\delta_k = \Delta L_{i_k^*}(n_{i_k^*,k}) - \Delta L_{j_k^*}(n_{j_k^*,k} + 1) \tag{17}$$

$$\mathcal{C}_{k+1} = \begin{cases} \mathcal{C}_k - \{j^*\}, & \text{if } \delta_k \le 0 \\ \mathcal{C}_k, & \text{otherwise.} \end{cases} \tag{18}$$

To complete the specification of this process, we need to: 1) ensure that the constraint $0 \le n_{i,k} \le K$ is never violated in (14) and 2) resolve the possibility that either i_k^* in (15) or j_k^* in (16) is not uniquely defined.

Regarding 1) above, the constraint may be violated in one of two ways. First, it may be violated if $n_{i,k} = 0$ for some i and $i = i_k^*, \delta_k > 0$ in (14). Observe, however, that $\Delta L_i(0)$ is undefined in (2), which in turn would make the definitions of i_k^*, j_k^* in (15) and (16), respectively, undefined, unless all i such that $n_{i,k} = 0$ are excluded from the set $\mathcal{C}_k$. Alternatively, we will henceforth set

$$\Delta L_i(0) \equiv -\infty$$

for all $i = 1, \cdots, N$, which clearly ensures that (15) may not yield i_k^* such that $n_{i_k^*,k} = 0$, unless $n_{i,k} = 0$ for all $i \in \mathcal{C}_k$; in this case, however, (17) gives $\delta_k \le 0$, therefore $n_{i,k+1} = n_{i,k}$.

Second, it is possible to violate $n_{i,k} \le K$ if $n_{i,k} = K$ for some i at step k. In this case, however, since $n_1 + \cdots + n_n = K$, we must have $\Delta L_j(0) = -\infty$ for all $j \ne i$ and, therefore, $j_k^* \ne i$.

Regarding 2) above, in (15) and (16) ties (i.e., if there is more than one index that qualifies as either i_k^* or j_k^*) are assumed to be arbitrarily broken. Moreover, in the case when $\max\{\Delta L_i(\cdot)\} = \min\{\Delta L_i(\cdot)\}$ and $|\mathcal{C}_k| > 1$, the choice of i_k^* and j_k^* is also arbitrary, provided $i_k^* \ne j_k^*$. Finally, for simplicity, we will adopt the following convention:

$$\text{If } i_k^* = p \text{ and } \delta_k \le 0, \text{ then } i_{k+1}^* = p. \tag{19}$$

This statement is trivial if the maximization in (15) gives a unique value. If, however, this is not the case and i_k^* is determined by arbitrarily breaking a tie, then we simply leave this index unchanged as long as $\delta_l \le 0$ for $l > k$, which implies that all $\Delta L_i(n_{i,k})$ values remain unchanged.

Interpretation of (14)–(18): Before proceeding with a detailed analysis of the processes $\{n_{i,k}\}, k = 0, 1, \cdots$, for each $i = 1, \cdots, N$, let us provide an informal description and interpretation of the full dynamic allocation scheme (14)–(18). Looking at (15), i_k^* identifies the user "most sensitive" to the removal of a resource among those users in the set $\mathcal{C}_k$, while in (16), j_k^* identifies the user who is "least sensitive." Then, (14) forces a natural exchange of resources from the least to the most sensitive user at the kth step of this process, provided the quantity δ_k is strictly positive (an interpretation of δ_k is provided below). Otherwise, the allocation is unaffected, but the user with index j_k^* is removed from the set $\mathcal{C}_k$ through (18). Thus, as the process evolves, users are gradually removed from this set. As we will show in the next section, the process terminates in a finite number of steps when this set contains a single element (user index), and the corresponding allocation is a globally optimal one.

As defined in (17), δ_k represents the "potential improvement" (cost reduction) incurred by a transition from allocation $\boldsymbol{s}_k$ to $\boldsymbol{s}_{k+1}$. That is

$$\delta_k = L(\boldsymbol{s}_k) - L(\boldsymbol{s}_{k+1}) \tag{20}$$

which is seen as follows:

$$\begin{aligned} L(\boldsymbol{s}_k) &- L(\boldsymbol{s}_{k+1}) \\ &= \sum_{i=1}^{N} L_i(n_{i,k}) - \sum_{i=1}^{N} L_i(n_{i,k+1}) \\ &= L_{i_k^*}(n_{i_k^*,k}) + L_{j_k^*}(n_{j_k^*,k}) - L_{i_k^*}(n_{i_k^*,k} - 1) \\ &\quad - L_{j_k^*}(n_{j_k^*,k} + 1) \\ &= \Delta L_{i_k^*}(n_{i_k^*,k}) - \Delta L_{j_k^*}(n_{j_k^*,k} + 1) = \delta_k. \end{aligned}$$

Note that if $\delta_k > 0$, which implies that the cost will be reduced by allocation $\boldsymbol{s}_{k+1}$, then the reallocation is implemented in (14). If, on the other hand, $\delta_k \le 0$, this implies no cost reduction under the candidate allocation $\boldsymbol{s}_{k+1}$, and $\boldsymbol{s}_k$ remains unchanged as seen in (14).

A. Properties of the Resource Allocation Process

We begin by establishing in Lemma 3.1 below a number of properties that the sequences $\{n_{i,k}\}$ and $\{\mathcal{C}_k\}$ in (14) and (18), respectively, satisfy. Based on these properties, we will show that $\{\boldsymbol{s}_k\}$, where $\boldsymbol{s}_k = [n_{1,k}, \cdots, n_{N,k}]$, converges to a globally optimal allocation. We will also use them to determine an upper bound for the number of steps required to reach this global optimum.

Lemma 3.1: The process defined by (14)–(18) is characterized by the following properties.

P1: $\Delta L_{i_k^*}(\cdot)$ is nonincreasing in $k = 0, 1, \cdots$, that is

$$\Delta L_{i_{k+1}^*}(n_{i_{k+1}^*,k+1}) \leq \Delta L_{i_k^*}(n_{i_k^*,k}) \quad \text{for all } k = 0, 1, \cdots. \tag{21}$$

P2: $\Delta L_{j_k^*}(\cdot)$ is *almost always* nondecreasing in $k = 0, 1, \cdots$, that is

$$\Delta L_{j_{k+b}^*}(n_{j_{k+b}^*,k+b}) \geq \Delta L_{j_k^*}(n_{j_k^*,k}) \quad \text{for all } k = 0, 1, \cdots \text{ and } b \in \{1, 2\}. \tag{22}$$

P3: Let $p = i_k^*$, and suppose there exists some $m > k$ such that $j_m^* = p$ and $p \neq i_l^*$ for all $k < l < m$. Then

$$\mathcal{C}_{m+1} = \mathcal{C}_m - \{p\}. \tag{23}$$

P4: Let $p = j_k^*$ and suppose there exists some $m > k$ such that $i_m^* = p$ and $p \neq j_l^*$ for all $k < l < m$. Then, there exists some $q, 1 \leq q \leq N-1$, such that

$$\mathcal{C}_{m+q+1} = \begin{cases} \mathcal{C}_{m+q} - \{p\}, & \text{if } |\mathcal{C}_{m+q+1}| > 1 \\ \{p\}, & \text{if } |\mathcal{C}_{m+q+1}| = 1. \end{cases} \tag{24}$$

P5: Let $i_k^* = p$. Then

$$n_{p,m} \leq n_{p,k} \quad \text{for any } k = 0, 1, \cdots, \text{ and for all } m > k. \tag{25}$$

P6: Let $j_k^* = p$. Then

$$n_{p,m} \geq n_{p,k} \quad \text{for any } k = 0, 1, \cdots, \text{ and for all } m > k. \tag{26}$$

Proof: The proof of this lemma is included in Appendix I.

Properties P3 and P4 are particularly important in characterizing the behavior of the resource allocation process in (14)–(18) and in establishing the main results of this section. In particular, P3 states that if any user p is identified as i_k^* at any step k of the process and as $j_m^*, m > k$, then this user is immediately removed from the $\mathcal{C}_m$ set. This also implies that $n_{p,m}$ is the number of resources finally allocated to p. Property P4 is a dual statement with a different implication. Once a user p is identified as j_k^* at some step k and as $i_m^*, m > k$, then there are two possibilities: either p will be the only user left in $\mathcal{C}_l, l > m$ and, therefore, the allocation process will terminate, or p will be removed from $\mathcal{C}_l$ for some $m < l < m + N - 1$.

This discussion also serves to point out an important difference between P5 and P6, which, at first sight, seem exact duals of each other. In P5, a user $p = i_k^*$ for some k will never in the future take any resources from other users. On the other hand, in P6 it is not true that a user $p = j_k^*$ will never in the future give away any resources to other users; rather, user p may give away *at most one resource* to other users. To clarify this we consider the following scenario. If $\delta_k > 0$, then from (14) $n_{p,k+1} = n_{p,k} + 1$. Now, if there exists $m > k$ such that $i_m^* = p$ and $\delta_m > 0$, then $n_{p,m+1} = n_{p,m} - 1 \geq n_{p,k}$ since $n_{p,m} \geq n_{p,k+1} = n_{p,k} + 1$. Thus, at step $m > k$ user p gives away a resource.

The next result establishes an upper bound in the number of steps required for the process defined by (14)–(18) to converge to a *final allocation*. A final allocation $\boldsymbol{s}_L$ is defined to be one at step L with $|\mathcal{C}_L| = 1$. At this point, we may allow one additional step to empty $\mathcal{C}_L$, but this is unnecessary since no further change in $\boldsymbol{s}_L$ is possible.

Lemma 3.2: The process defined by (14)–(18) reaches a final state $(\bar{\boldsymbol{s}}, \overline{\mathcal{C}})$ in L steps such that $|\overline{\mathcal{C}}| = 1$ and $L \leq K + 2(N-1)$.

Proof: See Appendix I.

Theorem 3: Let $\bar{\boldsymbol{s}} = [\overline{n}_1, \cdots, \overline{n}_N]$ be the final allocation of the process (14)–(18). Then, $\bar{\boldsymbol{s}}$ is a global optimum [i.e., a solution of $(\boldsymbol{RA2})$].

Proof: First, by Lemma 3.2, a final allocation exists. We will next show that this allocation satisfies

$$\Delta L_i(\overline{n}_i + 1) \geq \Delta L_j(\overline{n}_j) \quad \text{for any } i, j. \tag{27}$$

We establish this by contradiction. Suppose there exist $p \neq q$ such that (27) is violated, i.e., $\Delta L_p(\overline{n}_p + 1) < \Delta L_q(\overline{n}_q)$, and suppose that p, q were removed from $\mathcal{C}_k$ and $\mathcal{C}_l$, respectively (i.e., at steps k, l, respectively). Then, two cases are possible.

Case 1: $k < l$. For p to be removed from $\mathcal{C}_k$ in (18), the following should be true: $j_k^* = p$ and $\delta_k \leq 0$. However

$$\Delta L_p(n_{p,k} + 1) \geq \Delta L_{i_k^*}(n_{i_k^*,k}) \geq \Delta L_{i^*}(n_{i^*,l}) \geq \Delta L_q(n_{q,l})$$

where the first inequality is due to $\delta_k \leq 0$, the second is due to property P1 in (21), and the last is due to the definition of i_k^*. Therefore, our assumption is contradicted.

Case 2: $k > l$. Now q is removed from $\mathcal{C}_l$ first, therefore

$$\begin{aligned} \Delta L_q(n_{q,l}) &= \Delta L_{j_l^*}(n_{j_l^*,l}) \leq \Delta L_{j_k^*}(n_{j_k^*,k}) \\ &= \Delta L_p(n_{p,k}) < \Delta L_p(n_{p,k} + 1) \end{aligned}$$

where the two equalities are due to (18) and the fact that q and p were removed from $\mathcal{C}_l$ and $\mathcal{C}_k$, respectively. In addition, the first inequality is due to P2 in (22), and the last inequality is due to A2). Again, our assumption is contradicted.

Therefore, (27) holds. We can now invoke Theorem 1, from which it follows that (27) implies global optimality. ■

Corollary 3.1: The process (14)–(18) defines a *descent* algorithm, i.e.,

$$L(\boldsymbol{s}_k) \geq L(\boldsymbol{s}_l) \quad \text{for any } l > k.$$

Proof: This follows immediately from (14) and (18) and the fact that $\delta_k = L(\boldsymbol{s}_k) - L(\boldsymbol{s}_{k+1})$ in (20). ■

We conclude this section by presenting below a complete resource allocation algorithm which implements (14)–(18) and terminates by identifying a globally optimal allocation.

ALGORITHM (S1):

1.0 Initialize: $\boldsymbol{s}^{(0)} = [n_1^{(0)}, \cdots, n_N^{(0)}]$;
$\mathcal{C}^{(0)} = \{1, \cdots, N\}; k = 0$
Evaluate $\boldsymbol{D}(n_1^{(k)}, \cdots, n_N^{(k)}) \equiv [\Delta L_1(n_1^{(k)}), \cdots, \Delta L_N(n_N^{(k)})]$
If $|\mathcal{C}^{(k)}| = 1$ Goto **4.0**; Else Goto **2.1**

2.1 Set $i^* = \arg\max_{i \in \mathcal{C}^{(k)}} [\boldsymbol{D}(n_1^{(k)}, \cdots, n_N^{(k)})]$

2.2 Set $j^* = \arg\min_{i \in \mathcal{C}^{(k)}} [\boldsymbol{D}(n_1^{(k)}, \cdots, n_N^{(k)})]$

2.3 Evaluate $\boldsymbol{D}(n_1^{(k)}, \cdots, n_{i^*}^{(k)} - 1, \cdots, n_{j^*}^{(k)} + 1, \cdots, n_N^{(k)})$

2.4 If $\delta^{(k)} = \Delta L_{j^*}(n_{j^*}^{(k)} + 1) - \Delta L_{i^*}(n_{i^*}^{(k)}) < 0$ Goto **3.1**
Else Goto **3.2**

3.1 Update allocation: $n_{i^*}^{(k+1)} = n_{i^*}^{(k)} - 1$;
$n_{j^*}^{(k+1)} = n_{j^*}^{(k)} + 1$;
$n_m^{(k+1)} = n_m^{(k)}$ for all $m \in \mathcal{C}^{(k)}$ and $m \neq i^*, j^*$
Set $k \leftarrow k + 1$ and Goto **2.1**.

3.2 Replace $\mathcal{C}^{(k)}$ by $\mathcal{C}^{(k)} - \{j^*\}$;
If $|\mathcal{C}^{(k)}| = 1$ Goto **4.0**. Else Goto **2.2**

4.0 $\boldsymbol{s}^* = [n_1^{(k)}, \cdots, n_N^{(k)}]$. $STOP$.

IV. Online Stochastic Optimization Algorithm

In this section, we turn our attention to discrete resource allocation performed in a *stochastic* setting. When this is the case, the cost function $L(\boldsymbol{s})$ is usually an expectation whose exact value is difficult to obtain (except for very simple models). We therefore resort to estimates of $L(\boldsymbol{s})$ which may be obtained through simulation or through direct online observation of a system. In either case, we denote by $\tilde{L}^t(\boldsymbol{s})$ an estimate of $L(\boldsymbol{s})$ based on observing a sample path for a time period of length t. We are now faced with a problem of finding the optimal allocation using the noisy information $\tilde{L}^t(\boldsymbol{s})$.

It should be clear that the algorithm described by (14)–(18) does not work in a stochastic environment if we simply replace $L(\boldsymbol{s})$ by its estimate $\tilde{L}^t(\boldsymbol{s})$. For instance, suppose that $\delta_k > 0$; however, due to noise, we may obtain an estimate of δ_k, denoted by $\tilde{\delta}_k$, such that $\tilde{\delta}_k \leq 0$. In this case, rather than reallocating resources, we would remove a user from the $\mathcal{C}$ set *permanently*. This implies that this user can never receive any more resources, hence the optimal allocation will never be reached.

Therefore, certain modifications are necessary. In particular, we need to modify the process (14)–(18) in two ways. First, we will provide a mechanism through which users can re-enter the $\mathcal{C}$ set to compensate for the case where a user is erroneously removed because of noise. Second, we will progressively improve the estimates of the cost differences $\Delta L(\boldsymbol{s})$ so as to eliminate the effect of estimation noise; this can often be achieved by increasing the observed sample path length over which an estimate is taken. We will henceforth denote the length of such a sample path at the kth iteration of our process by $f(k)$.

The following is the modified process in a stochastic environment, denoted by $\{(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k)\}$, with $\tilde{\boldsymbol{s}}_k = [\tilde{n}_{1,k}, \tilde{n}_{2,k}, \cdots, \tilde{n}_{N,k}]$. After proper initialization, at the kth iteration we set

$$\tilde{n}_{i,k+1} = \begin{cases} \tilde{n}_{i,k} - 1, & \text{if } i = \tilde{i}_k^* \text{ and } \tilde{\delta}_k(\tilde{i}_k^*, \tilde{j}_k^*) > 0 \\ \tilde{n}_{i,k} + 1, & \text{if } i = \tilde{j}_k^* \text{ and } \tilde{\delta}_k(\tilde{i}_k^*, \tilde{j}_k^*) > 0 \\ \tilde{n}_{i,k}, & \text{otherwise} \end{cases} \tag{28}$$

where

$$\tilde{i}_k^* = \arg\max_{i \in \tilde{\mathcal{C}}_k} \{\Delta \tilde{L}_i^{f(k)}(\tilde{n}_{i,k})\} \tag{29}$$

$$\tilde{j}_k^* = \arg\min_{i \in \tilde{\mathcal{C}}_k} \{\Delta \tilde{L}_i^{f(k)}(\tilde{n}_{i,k})\} \tag{30}$$

$$\tilde{\delta}_k(\tilde{i}_k^*, \tilde{j}_k^*) = \Delta \tilde{L}_{\tilde{i}_k^*}^{f(k)}(\tilde{n}_{\tilde{i}_k^*,k}) - \Delta \tilde{L}_{\tilde{j}_k^*}^{f(k)}(\tilde{n}_{\tilde{j}_k^*,k} + 1) \tag{31}$$

$$\tilde{\mathcal{C}}_{k+1} = \begin{cases} \tilde{\mathcal{C}}_k - \{\tilde{j}_k^*\}, & \text{if } \tilde{\delta}_k(\tilde{i}_k^*, \tilde{j}_k^*) \leq 0 \\ \mathcal{C}_0, & \text{if } |\tilde{\mathcal{C}}_k| = 1 \\ \tilde{\mathcal{C}}_k, & \text{otherwise.} \end{cases} \tag{32}$$

It is clear that (28)–(32) define a Markov process $\{(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k)\}$, whose state transition probability matrix is determined by $\tilde{i}_k^*, \tilde{j}_k^*$, and $\tilde{\delta}_k(\tilde{i}_k^*, \tilde{j}_k^*)$. Before proceeding, let us point out that the only structural difference in (28)–(32) compared to the deterministic case of the previous section occurs in (32), where we reset the $\tilde{\mathcal{C}}_k$ set every time that it contains only one element. By doing so, we allow users that have been removed from the $\tilde{\mathcal{C}}_k$ set due to noise to re-enter the user set at the next step. Of course, the actual values of all $\Delta L_i(\cdot)$ are now replaced by their estimates, $\Delta \tilde{L}_i^{f(k)}(\cdot)$.

An obvious question that arises from a practical standpoint is that of obtaining the crucial cost difference estimates $\Delta \tilde{L}_i^{f(k)}(n_i)$. At first sight, to estimate these quantities two sample paths are required, one for $\tilde{L}_i^{f(k)}(n_i)$ and another for $\tilde{L}_i^{f(k)}(n_i - 1)$. However, for a large class of applications, one can exploit a variety of techniques based on perturbation analysis (PA) and concurrent estimation (CE) for discrete-event systems (e.g., see [2] and [6]) to obtain $\Delta \tilde{L}_i^{f(k)}(n_i)$ from a *single* sample path under an allocation n_i. Thus, the convergence of the process above can be substantially accelerated in many cases.

The following result simply establishes the fact that the modification in (32) does not alter the properties of the deterministic resource allocation process.

Theorem 4: The process described by (28)–(32), if driven by deterministic quantities such that $\Delta \tilde{L}_i^{f(k)}(\cdot) = \Delta L_i(\cdot)$, will yield the optimum $\boldsymbol{s}^*$.

Proof: If $\Delta \tilde{L}_i^{f(k)}(\cdot) = \Delta L_i(\cdot)$ for all i, the stochastic process (28)–(32) is the same as its deterministic version before $|\tilde{\mathcal{C}}_k| = 1$ in a finite number of steps. However, according to Theorem 3, $|\tilde{\mathcal{C}}_k| = 1$ means that the process has reached the optimum and will not change thereafter. ■

As stated earlier, the second modification we impose is to eliminate the effect of estimation noise by increasing the observed sample path length as the number of iterations increases. For this purpose, we make the following assumptions.

A4) For every i and every n_i, the estimate $\tilde{L}_i^t(n_i)$ is ergodic as the sample path length increases in the sense that

$$\lim_{t\to\infty} \tilde{L}_i^t(n_i) = L_i(n_i), \quad \text{a.s.}$$

A5) Let $\delta_k(i,j) = \Delta L_i(\tilde{n}_{i,k}) - \Delta L_j(\tilde{n}_{j,k}+1)$. For every $\delta_k(\tilde{i}_k^*, \tilde{j}_k^*) = 0$, there is a constant p_0 such that

$$\text{Prob}\,[\tilde{\delta}_k(\tilde{i}_k^*, \tilde{j}_k^*) \le 0 | \delta_k(\tilde{i}_k^*, \tilde{j}_k^*) = 0, (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \ge p_0 > 0$$

for any k and any pair $(\boldsymbol{s}, \mathcal{C})$.

Assumption A4) ensures that the effect of noise can be decreased by increasing the estimation interval over $k = 1, 2, \cdots$. This assumption is stronger than actually needed in Lemma 4.1 below, but it is mild and is satisfied by most systems of interest. Assumption A5) guarantees that an estimate does not always give one-side-biased incorrect information; it will only be needed once in the proof of Lemma 4.4. Both assumptions are mild in the context of discrete-event dynamic systems where such resource allocation problems frequently arise.

In the remainder of this section, we will study the process $\{(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k)\}$ so as to establish its convergence properties. Our main result is Theorem 5, where we show that this process converges in probability to the optimal allocation $\boldsymbol{s}^*$.

A. Properties of the Stochastic Resource Allocation Process

We begin with an auxiliary result that will prove very helpful in all subsequent analysis.

Lemma 4.1: Suppose that Assumption A4) holds and that $\lim_{k\to\infty} f(k) = \infty$. Then, for any pair $n_i, n_j, i \ne j$

$$\lim_{k\to\infty} \text{Prob}\,[\Delta\tilde{L}_i^{f(k)}(n_i) \ge \Delta\tilde{L}_j^{f(k)}(n_j)] = 0$$

and

$$\lim_{k\to\infty} \text{Prob}\,[\Delta\tilde{L}_i^{f(k)}(n_i) < \Delta\tilde{L}_j^{f(k)}(n_j)] = 1$$

provided that $\Delta L_i(n_i) < \Delta L_j(n_j)$.

Proof: The proof of this lemma is included in Appendix II.

Next, we introduce some useful properties of the process $\{\tilde{\boldsymbol{s}}_k\}$ in the following few lemmas. These properties pertain to the asymptotic behavior of probabilities of certain events crucial in the behavior of $\{(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k)\}$. We begin by defining these events and corresponding probabilities.

First, let

$$d_k(\boldsymbol{s}, \mathcal{C}) = 1 - \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{k+1}) \le L(\tilde{\boldsymbol{s}}_k) | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \tag{33}$$

so that $[1 - d_k(\boldsymbol{s}, \mathcal{C})]$ is the probability that either some cost reduction or no change in cost results from the kth transition in our process (i.e., the new allocation has at most the same cost). We will show in Lemma 4.2 that the probability of this event is asymptotically one, i.e., our process corresponds to an asymptotic descent resource allocation algorithm.

Next, given any state $(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k)$ reached by our process (28)–(32), define

$$A_k^{\max} = \{j | \Delta L_j(\tilde{n}_{j,k}) = \max_i \{\Delta L_i(\tilde{n}_{i,k})\}\} \tag{34}$$

$$A_k^{\min} = \{j | \Delta L_j(\tilde{n}_{j,k}) = \min_i \{\Delta L_i(\tilde{n}_{i,k})\}\}. \tag{35}$$

Observe that $A_k^{\max}$ and $A_k^{\min}$ are, respectively, the sets of indexes i_k^* and j_k^* defined in (15) and (16) of the deterministic optimization process (with exact measurement). Recall that i_k^*, j_k^* *need not be unique* at each step k, hence the need for these sets. We then define

$$a_k(\boldsymbol{s}, \mathcal{C}) = 1 - \text{Prob}\,[\tilde{i}_k^* \in A_k^{\max} | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \tag{36}$$

$$b_k(\boldsymbol{s}, \mathcal{C}) = 1 - \text{Prob}\,[\tilde{j}_k^* \in A_k^{\min} | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})]. \tag{37}$$

Here, $[1 - a_k(\boldsymbol{s}, \mathcal{C})]$ is the probability that our stochastic resource allocation process at step k correctly identifies an index $\tilde{i}_k^*$ as belonging to the set $A_k^{\max}$ (similarly for $[1 - b_k(\boldsymbol{s}, \mathcal{C})]$). We will show in Lemma 4.3 that these probabilities are asymptotically one.

Lemma 4.2: For any $\boldsymbol{s} = [n_1, n_2, \cdots, n_N] \in \mathcal{S}$ and any $\mathcal{C}$

$$\lim_{k\to\infty} d_k(\boldsymbol{s}, \mathcal{C}) = 0. \tag{38}$$

Moreover, define

$$d_k = \sup_{i\ge k} \max_{(\boldsymbol{s}, \mathcal{C})} d_i(\boldsymbol{s}, \mathcal{C}). \tag{39}$$

Then $d_k \ge d_k(\boldsymbol{s}, \mathcal{C}), d_k$ is monotone decreasing, and

$$\lim_{k\to\infty} d_k = 0. \tag{40}$$

Proof: See Appendix II.

Lemma 4.3: Suppose that Assumption A4) holds. Then, for every pair $(\boldsymbol{s}, \mathcal{C})$, we have

$$\lim_{k\to\infty} a_k(\boldsymbol{s}, \mathcal{C}) = 0, \qquad \lim_{k\to\infty} b_k(\boldsymbol{s}, \mathcal{C}) = 0. \tag{41}$$

Moreover, define

$$a_k = \sup_{i\ge k} \max_{(\boldsymbol{s}, \mathcal{C})} a_k(\boldsymbol{s}, \mathcal{C}), \qquad b_k = \sup_{i\ge k} \max_{(\boldsymbol{s}, \mathcal{C})} b_k(\boldsymbol{s}, \mathcal{C}). \tag{42}$$

Then $a_k \ge a_k(\boldsymbol{s}, \mathcal{C}), b_k \ge b_k(\boldsymbol{s}, \mathcal{C})$, both a_k and b_k are monotone decreasing, and

$$\lim_{k\to\infty} a_k = 0, \qquad \lim_{k\to\infty} b_k = 0. \tag{43}$$

Proof: Given the definition of the sets $A_k^{\max}$ and $A_k^{\min}$, the proof of the first part follows immediately from Lemma 4.1. The second part then follows from the fact that, by their definitions, a_k and b_k are monotone decreasing. ■

The last asymptotic property we need establishes the fact that there will be an improvement (i.e., strictly lower cost) to an allocation at step k if that allocation is not optimal. However, this improvement may not occur within a single step; rather, we show in Lemma 4.5 that such an improvement may require a number of steps α_k beyond the kth step, where α_k satisfies certain requirements. A related property needed to establish Lemma 4.5 is shown in Lemma 4.4; in particular, if an allocation is not optimal at step k, then the probability that this allocation remains unchanged over α_k steps is asymptotically zero.

To formulate the property above in a precise manner, we begin by choosing a sequence of integers $\{\alpha_k\}$ satisfying

$$\lim_{k\to\infty}\alpha_k=\infty, \qquad \lim_{k\to\infty}\alpha_k(a_k+b_k)=0$$
$$\lim_{k\to\infty}(1-d_{\lfloor k/2\rfloor})^{\alpha_k}=1 \tag{44}$$

where, for any x, $\lfloor x\rfloor=\{n|n\le x, n \text{ is integer}\}$ is the greatest integer smaller than x.

Such a sequence $\{\alpha_k\}$ exists. For example, any $\alpha_k\le\lfloor(\max\{d_{\lfloor k/2\rfloor}, a_k+b_k\})^{-1/2}\rfloor, \lim_{k\to\infty}\alpha_k=\infty$ satisfies (44) (without loss of generality, we assume that $d_k a_k b_k\neq 0$, otherwise α_k can take any arbitrary value). The choice of $\{\alpha_k\}$ is rather technical. Its necessity will be clear from the proof of the following Lemmas 4.4 and 4.5.

Observe that if $\{\alpha_k\}$ satisfies (44), we also have

$$\lim_{k\to\infty}(1-d_k)^{\alpha_k}=1 \tag{45}$$

since $d_k\le d_{\lfloor k/2\rfloor}$.

With this definition of α_k, we now set

$$e_k(\boldsymbol{s},\mathcal{C})=1-\text{Prob}\,[L(\tilde{\boldsymbol{s}}_{k+\alpha_k})<L(\tilde{\boldsymbol{s}}_k)|(\tilde{\boldsymbol{s}}_k,\tilde{\mathcal{C}}_k)=(\boldsymbol{s},\mathcal{C})] \tag{46}$$

and observe that $[1-e_k(\boldsymbol{s},\mathcal{C})]$ is the probability that strict improvement (i.e., strictly lower cost) results when transitioning from a state such that the allocation is not optimal to a future state α_k steps later. We will establish in Lemma 4.5 that this probability is asymptotically one. To do so, we first need the following result, asserting that if an allocation is not optimal, then the cost remains unchanged for α_k steps with asymptotic probability zero.

Lemma 4.4: Suppose that A4) and A5) hold and let $\{\alpha_k\}$ satisfy (44). Consider an allocation $\boldsymbol{s}=[n_1,\cdots,n_N]\neq\boldsymbol{s}^*$ and any set $\mathcal{C}$. Then

$$\lim_{k\to\infty}\text{Prob}\,[L(\tilde{\boldsymbol{s}}_{h+1})=L(\tilde{\boldsymbol{s}}_h), h=k,\cdots,k+\alpha_k-1$$
$$|(\tilde{\boldsymbol{s}}_k,\tilde{\mathcal{C}}_k)=(\boldsymbol{s},\mathcal{C})]=0. \tag{47}$$

Proof: See Appendix II.

With the help of the lemma above, we obtain the following.

Lemma 4.5: Suppose that A4) and A5) hold. For any allocation $\boldsymbol{s}=[n_1,\cdots,n_N]\neq\boldsymbol{s}^*$ and any set $\mathcal{C}$

$$\lim_{k\to\infty}e_k(\boldsymbol{s},\mathcal{C})=0. \tag{48}$$

Moreover, define

$$e_k=\sup_{i\ge k}\max_{\boldsymbol{s}\in(\mathcal{S},\mathcal{C})}e_i(\boldsymbol{s},\mathcal{C}). \tag{49}$$

Then $e_k\ge e_k(\boldsymbol{s},\mathcal{C}), e_k$ is monotone decreasing and

$$\lim_{k\to\infty}e_k=0. \tag{50}$$

Proof: See Appendix II.

B. Convergence of Stochastic Resource Allocation Process

With the help of the properties established in the previous section, we can prove the following theorem on the convergence of the process $\{(\tilde{\boldsymbol{s}}_k,\tilde{\mathcal{C}}_k)\}$ defined through (28)–(32).

Theorem 5: Suppose that Assumptions A4) and A5) hold and that the optimum $\boldsymbol{s}^*$ is unique. Then the process described by (28)–(32) converges in probability to the optimal allocation $\boldsymbol{s}^*$.

Proof: We begin by defining three auxiliary quantities we shall use in the proof.

First, let us choose some $\epsilon>0$ such that

$$\epsilon<\min_{\{\boldsymbol{s},\boldsymbol{s}'\}}\{L(\boldsymbol{s})-L(\boldsymbol{s}')|L(\boldsymbol{s})-L(\boldsymbol{s}')>0\}. \tag{51}$$

Note that such $\epsilon>0$ exists because of the discrete nature of the cost function and the finiteness of the number of feasible allocations. Observe that ϵ is a real number *strictly* lower than the smallest cost difference in the allocation process.

Second, for any $\boldsymbol{s}$, set

$$q_{\boldsymbol{s}}=\lfloor(L(\boldsymbol{s})-L(\boldsymbol{s}^*))/\epsilon\rfloor.$$

Then

$$L(\boldsymbol{s})-L(\boldsymbol{s}^*)\ge q_{\boldsymbol{s}}\epsilon, \qquad L(\boldsymbol{s})-L(\boldsymbol{s}^*)<(q_{\boldsymbol{s}}+1)\epsilon. \tag{52}$$

Finally, we shall define a convenient sequence $\{\alpha_k\}$ that satisfies (44). To do so, let $q=\max_{\boldsymbol{s}\in\mathcal{S}}q_{\boldsymbol{s}}$, and, for any k, choose

$$\alpha_k=\left\lfloor\frac{1}{2q}\min\{k,(\max\{d_{\lfloor k/2\rfloor},a_k+b_k\})^{-1/2}\}\right\rfloor$$
$$\le\frac{1}{2q}\min\{k,(\max\{d_{\lfloor k/2\rfloor},a_k+b_k\})^{-1/2}\}.$$

Since the sequences $\{d_k\}$ and $\{a_k+b_k\}$ are monotone decreasing by their definitions, the sequence $\{\alpha_k\}$ is monotone increasing and it is easy to verify that it satisfies (44).

The next step in the proof is to define a particular subsequence of $\{(\tilde{\boldsymbol{s}}_i,\tilde{\mathcal{C}}_i)\}$ as follows. First, set $x=k-q\alpha_k$ and observe that

$$x=k-q\alpha_k\ge k/2. \tag{53}$$

Then, define a sequence of indexes $\{y_i\}$ with $i=0,\cdots,q_{\boldsymbol{s}}$ through

$$y_0=x, \qquad y_i=y_{i-1}+\alpha_{y_{i-1}}, \qquad i=1,2,\cdots,q_{\boldsymbol{s}}.$$

For sufficiently large k such that $\alpha_x\ge 1$, it is easy to verify by induction that

$$x=y_0<y_1<\cdots<y_{q_{\boldsymbol{s}}}\le k. \tag{54}$$

Now, for any k and x defined above, consider a subsequence of $\{(\tilde{\boldsymbol{s}}_i,\tilde{\mathcal{C}}_i), i=x,\cdots,k\}$, denoted by $\psi=\{(\tilde{\boldsymbol{s}}_{y_i},\tilde{\mathcal{C}}_{y_i}), i=0,1,\cdots,q_{\boldsymbol{s}}\}$, starting at $\tilde{\boldsymbol{s}}_{y_0}=\tilde{\boldsymbol{s}}_x=\boldsymbol{s}$, and such that either there is an $i, 0\le i\le q_{\boldsymbol{s}}-1$ such that

$$L(\tilde{\boldsymbol{s}}_{y_{j+1}})-L(\tilde{\boldsymbol{s}}_{y_j})\le-\epsilon$$
$$\text{for all } j=0,\cdots,i-1 \text{ and } \tilde{\boldsymbol{s}}_{y_j}=\boldsymbol{s}^*,$$
$$\text{for all } j=i,\cdots,q_{\boldsymbol{s}}$$

or

$$L(\tilde{\boldsymbol{s}}_{y_{i+1}}) - L(\tilde{\boldsymbol{s}}_{y_i}) \le -\epsilon \quad \text{and } \tilde{\boldsymbol{s}}_{y_i} \neq \boldsymbol{s}^* \quad \text{for all } i = 0, \cdots, q_{\boldsymbol{s}} - 1.$$

In other words, any such subsequence is "embedded" into the original process $\{(\tilde{\boldsymbol{s}}_i, \tilde{\mathcal{C}}_i)\}$ so as to give strictly decreasing costs, and if it reaches the optimum it stays there afterwards.

The subsequence defined above has the additional property that

$$\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}}} = \boldsymbol{s}^*. \tag{55}$$

This is obvious in the case where $\tilde{\boldsymbol{s}}_{y_j} = \boldsymbol{s}^*$ for some $j = 0, 1, \cdots, q_{\boldsymbol{s}}$. On the other hand, if $\tilde{\boldsymbol{s}}_{y_i} \neq \boldsymbol{s}^*$ for all $i = 0, 1, \cdots, q_{\boldsymbol{s}} - 1$, we must have

$$L(\tilde{\boldsymbol{s}}_{y_i}) - L(\tilde{\boldsymbol{s}}_{y_{i-1}}) \le -\epsilon, \qquad i = 1, \cdots, q_{\boldsymbol{s}}. \tag{56}$$

Adding the $q_{\boldsymbol{s}}$ inequalities above yields

$$L(\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}}}) - L(\tilde{\boldsymbol{s}}_x) \le -q_{\boldsymbol{s}}\epsilon$$

or, since $\tilde{\boldsymbol{s}}_x = \boldsymbol{s}$

$$L(\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}}}) - L(\boldsymbol{s}^*) \le L(\boldsymbol{s}) - L(\boldsymbol{s}^*) - q_{\boldsymbol{s}}\epsilon.$$

This inequality, together with (52), implies that

$$L(\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}}}) - L(\boldsymbol{s}^*) \le \epsilon.$$

Since ϵ satisfies (51), we must have $L(\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}}}) = L(\boldsymbol{s}^*)$ for all paths satisfying (56), which in turn implies $\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}}} = \boldsymbol{s}^*$ since the optimum $\boldsymbol{s}^*$ is assumed unique. Therefore, for every subsequence $\psi = \{(\tilde{\boldsymbol{s}}_{y_i}, \tilde{\mathcal{C}}_{y_i}), i = 0, 1, \cdots, q_{\boldsymbol{s}}\}$ considered, (55) holds.

Before proceeding with the main part of the proof, let us also define, for notational convenience, a set Ψ to contain all subsequences of the form ψ as specified above, or any part of any such subsequence, i.e., any $\{(\tilde{\boldsymbol{s}}_{y_n}, \tilde{\mathcal{C}}_{y_n}), \cdots, (\tilde{\boldsymbol{s}}_{y_m}, \tilde{\mathcal{C}}_{y_m})\}$ with $n \le m$ and $n, m \in \{0, 1, \cdots, q_{\boldsymbol{s}}\}$.

Then, for any $\boldsymbol{s} \neq \boldsymbol{s}^*$ and any $\mathcal{C}$, all sample paths restricted to include some $\psi \in \Psi$ form a subset of all sample paths that lead to a state such that $\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}}} = \boldsymbol{s}^*$, i.e.,

$$\begin{aligned} &\text{Prob}\,[\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}}} = \boldsymbol{s}^* | (\tilde{\boldsymbol{s}}_x, \tilde{\mathcal{C}}_x) = (\boldsymbol{s}, \mathcal{C})] \\ &\quad \ge \sum_{\{(\boldsymbol{s}_i, \mathcal{C}_i), i=1,\cdots,q_{\boldsymbol{s}}\} \in \Psi} \text{Prob}\,[(\tilde{\boldsymbol{s}}_{y_i}, \tilde{\mathcal{C}}_{y_i}) = (\boldsymbol{s}_i, \mathcal{C}_i), \\ &\qquad\qquad i = 1, \cdots, q_{\boldsymbol{s}} | (\tilde{\boldsymbol{s}}_x, \tilde{\mathcal{C}}_x) = (\boldsymbol{s}, \mathcal{C})]. \end{aligned} \tag{57}$$

Because $\{(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k)\}$ is a Markov process, setting $(\boldsymbol{s}_0, \mathcal{C}_0) = (\boldsymbol{s}, \mathcal{C})$, the previous inequality can be rewritten as

$$\begin{aligned} &\text{Prob}\,[\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}}} = \boldsymbol{s}^* | (\tilde{\boldsymbol{s}}_x, \tilde{\mathcal{C}}_x) = (\boldsymbol{s}, \mathcal{C})] \\ &\quad \ge \sum_{\{(\boldsymbol{s}_i, \mathcal{C}_i), i=1,\cdots,q_{\boldsymbol{s}}\} \in \Psi} \prod_{i=1}^{q_{\boldsymbol{s}}} \\ &\qquad \text{Prob}\,[(\tilde{\boldsymbol{s}}_{y_i}, \tilde{\mathcal{C}}_{y_i}) = (\boldsymbol{s}_i, \mathcal{C}_i) | (\tilde{\boldsymbol{s}}_{y_{i-1}}, \tilde{\mathcal{C}}_{y_{i-1}}) = (\boldsymbol{s}_{i-1}, \mathcal{C}_{i-1})]. \end{aligned}$$

In addition, let us decompose any subsequence ψ into its first $(q_{\boldsymbol{s}} - 1)$ elements and the remaining element $(\boldsymbol{s}_{q_{\boldsymbol{s}}}, \mathcal{C}_{q_{\boldsymbol{s}}})$. Thus, for any subsequence whose $(q_{\boldsymbol{s}} - 1)$th element is $(\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}-1}}, \tilde{\mathcal{C}}_{y_{q_{\boldsymbol{s}}-1}})$, there is a set of final states such that $(\boldsymbol{s}_{q_{\boldsymbol{s}}}, \mathcal{C}_{q_{\boldsymbol{s}}}) \in \Psi$, so that we may write (58), as shown at the bottom of the page.

Let us now consider two possible cases regarding the value of $\boldsymbol{s}_{q_{\boldsymbol{s}}-1}$.

Case 1: If $\boldsymbol{s}_{q_{\boldsymbol{s}}-1} = \boldsymbol{s}^*$, then, aggregating over all $\mathcal{C}_{q_{\boldsymbol{s}}}$ and recalling (55), we can write, for any $\mathcal{C}_{q_{\boldsymbol{s}}-1}$ in some subsequence of Ψ

$$\begin{aligned} &\sum_{(\boldsymbol{s}_{q_{\boldsymbol{s}}}, \mathcal{C}_{q_{\boldsymbol{s}}}) \in \Psi} \text{Prob}\,[(\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}}}, \tilde{\mathcal{C}}_{y_{q_{\boldsymbol{s}}}}) = (\boldsymbol{s}_{q_{\boldsymbol{s}}}, \mathcal{C}_{q_{\boldsymbol{s}}}) | (\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}-1}}, \tilde{\mathcal{C}}_{y_{q_{\boldsymbol{s}}-1}}) \\ &\qquad = (\boldsymbol{s}^*, \mathcal{C}_{q_{\boldsymbol{s}}-1})] \\ &\quad = \text{Prob}\,[\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}}} = \boldsymbol{s}^* | (\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}-1}}, \tilde{\mathcal{C}}_{y_{q_{\boldsymbol{s}}-1}}) = (\boldsymbol{s}^*, \mathcal{C}_{q_{\boldsymbol{s}}-1})]. \end{aligned}$$

Now let us consider a subsequence $\{(\tilde{\boldsymbol{s}}_i, \tilde{\mathcal{C}}_i)\}$ with $i = y_{q_{\boldsymbol{s}}-1}, \cdots, y_{q_{\boldsymbol{s}}}$ and $\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}-1}} = \tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}}} = \boldsymbol{s}^*$. Observing that all subsequences $\{(\tilde{\boldsymbol{s}}_i, \tilde{\mathcal{C}}_i)\}$ restricted to $\tilde{\boldsymbol{s}}_i = \boldsymbol{s}^*$ for all $i = y_{q_{\boldsymbol{s}}-1}, \cdots, y_{q_{\boldsymbol{s}}}$ form a subset of all the subsequences above, and exploiting once again the Markov property of the process $\{(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k)\}$, we can clearly write

$$\begin{aligned} &\text{Prob}\,[\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}}} = \boldsymbol{s}^* | (\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}-1}}, \tilde{\mathcal{C}}_{y_{q_{\boldsymbol{s}}-1}}) = (\boldsymbol{s}^*, \mathcal{C}_{q_{\boldsymbol{s}}-1})] \\ &\quad \ge \sum_{\{\mathcal{C}'_i, i=y_{q_s-1},\cdots,y_{q_{\boldsymbol{s}}}\}} \prod_{i=y_{q_s-1}+1}^{y_{q_s}} \\ &\qquad \cdot \text{Prob}\,[(\tilde{\boldsymbol{s}}_i, \tilde{\mathcal{C}}_i) = (\boldsymbol{s}^*, \mathcal{C}'_i) | (\tilde{\boldsymbol{s}}_{i-1}, \tilde{\mathcal{C}}_{i-1}) = (\boldsymbol{s}^*, \mathcal{C}'_{i-1})] \end{aligned}$$

where $\mathcal{C}'_{y_{q_{\boldsymbol{s}}-1}} = \mathcal{C}_{q_{\boldsymbol{s}}-1}$. Using the definition of $d_k(\boldsymbol{s}, \mathcal{C})$ in (33) and noticing that, given $\tilde{\boldsymbol{s}}_k = \boldsymbol{s}^*, L(\tilde{\boldsymbol{s}}_{k+1}) \le L(\tilde{\boldsymbol{s}}_k)$ is equivalent to $\tilde{\boldsymbol{s}}_{k+1} = \boldsymbol{s}^*$ when the optimum is unique, each term in the product above can be replaced by $[1 -$

$$\begin{aligned} &\sum_{\{(\boldsymbol{s}_i, \mathcal{C}_i), i=1,\cdots,q_{\boldsymbol{s}}\} \in \Psi} \prod_{i=1}^{q_{\boldsymbol{s}}} \text{Prob}\,[(\tilde{\boldsymbol{s}}_{y_i}, \tilde{\mathcal{C}}_{y_i}) = (\boldsymbol{s}_i, \mathcal{C}_i) | (\tilde{\boldsymbol{s}}_{y_{i-1}}, \tilde{\mathcal{C}}_{y_{i-1}}) = (\boldsymbol{s}_{i-1}, \mathcal{C}_{i-1})] \\ &\quad = \sum_{\{(\boldsymbol{s}_i, \mathcal{C}_i), i=1,\cdots,q_{\boldsymbol{s}}-1\} \in \Psi} \left(\prod_{i=1}^{q_{\boldsymbol{s}}-1} \text{Prob}\,[(\tilde{\boldsymbol{s}}_{y_i}, \tilde{\mathcal{C}}_{y_i}) = (\boldsymbol{s}_i, \mathcal{C}_i) | (\tilde{\boldsymbol{s}}_{y_{i-1}}, \tilde{\mathcal{C}}_{y_{i-1}}) = (\boldsymbol{s}_{i-1}, \mathcal{C}_{i-1})] \right. \\ &\qquad \left. \times \sum_{(\boldsymbol{s}_{q_{\boldsymbol{s}}}, \mathcal{C}_{q_{\boldsymbol{s}}}) \in \Psi} \text{Prob}\,[(\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}}}, \tilde{\mathcal{C}}_{y_{q_{\boldsymbol{s}}}}) = (\boldsymbol{s}_{q_{\boldsymbol{s}}}, \mathcal{C}_{q_{\boldsymbol{s}}}) | (\tilde{\boldsymbol{s}}_{y_{q_{\boldsymbol{s}}-1}}, \tilde{\mathcal{C}}_{y_{q_{\boldsymbol{s}}-1}}) = (\boldsymbol{s}_{q_{\boldsymbol{s}}-1}, \mathcal{C}_{q_{\boldsymbol{s}}-1})] \right) \end{aligned} \tag{58}$$

$d_{i-1}(\tilde{\boldsymbol{s}}_{i-1}, \tilde{\mathcal{C}}_{i-1})], i = y_{q_s-1}+1, \cdots, y_{q_s}$. In addition, from Lemma 4.2, we have $d_k \geq d_k(\boldsymbol{s}, \mathcal{C})$. Therefore

$$\begin{aligned}
&\text{Prob}\,[\tilde{\boldsymbol{s}}_{y_{q_s}} = \boldsymbol{s}^* | (\tilde{\boldsymbol{s}}_{y_{q_s-1}}, \tilde{\mathcal{C}}_{y_{q_s-1}}) = (\boldsymbol{s}^*, \mathcal{C}_{q_s-1})] \\
&\quad \geq \prod_{i=y_{q_s-1}+1}^{y_{q_s}} (1-d_i) \geq (1-d_{y_{q_s-1}})^{y_{q_s}-y_{q_s-1}} \\
&\quad \geq (1-d_x)^{y_{q_s}-y_{q_s-1}} \qquad (59)
\end{aligned}$$

where the last two inequalities follow from the fact that d_k is monotone decreasing and the fact that $y_i \geq x$.

Case 2: If $\boldsymbol{s}_{q_s-1} \neq \boldsymbol{s}^*$, then by the definition of any subsequence $\psi \in \Psi$, we must have a strict cost decrease, i.e., $L(\tilde{\boldsymbol{s}}_{y_{q_s}}) - L(\tilde{\boldsymbol{s}}_{y_{q_s-1}}) \leq -\epsilon$. Therefore, for any $\mathcal{C}_{q_s-1}$ in some subsequence of Ψ, we can now write

$$\begin{aligned}
&\sum_{(\boldsymbol{s}_{q_s}, \mathcal{C}_{q_s}) \in \Psi} \text{Prob}\,[(\tilde{\boldsymbol{s}}_{y_{q_s}}, \tilde{\mathcal{C}}_{y_{q_s}}) = (\boldsymbol{s}_{q_s}, \mathcal{C}_{q_s}) \\
&\qquad\qquad |(\tilde{\boldsymbol{s}}_{y_{q_s-1}}, \tilde{\mathcal{C}}_{y_{q_s-1}}) = (\boldsymbol{s}_{q_s-1}, \mathcal{C}_{q_s-1})]) \\
&\quad = \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{y_{q_s}}) - L(\tilde{\boldsymbol{s}}_{y_{q_s-1}}) \leq -\epsilon \\
&\qquad\qquad |(\tilde{\boldsymbol{s}}_{y_{q_s-1}}, \tilde{\mathcal{C}}_{y_{q_s-1}}) = (\boldsymbol{s}_{q_s-1}, \mathcal{C}_{q_s-1})] \\
&\quad = \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{y_{q_s-1}+\alpha_{y_{q_s-1}}}) < L(\tilde{\boldsymbol{s}}_{y_{q_s-1}}) \\
&\qquad\qquad |(\tilde{\boldsymbol{s}}_{y_{q_s-1}}, \tilde{\mathcal{C}}_{y_{q_s-1}}) = (\boldsymbol{s}_{q_s-1}, \mathcal{C}_{q_s-1})] \qquad (60)
\end{aligned}$$

recalling the choice of ϵ in (51).

We can now make use of the definition of $e_k(\boldsymbol{s}, \mathcal{C})$ in (46) and write

$$\begin{aligned}
&1 - e_{y_{q_s-1}} \\
&\quad = \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{y_{q_s-1}+\alpha_{y_{q_s-1}}}) < L(\tilde{\boldsymbol{s}}_{y_{q_s-1}}) \\
&\qquad\qquad |(\tilde{\boldsymbol{s}}_{y_{q_s-1}}, \tilde{\mathcal{C}}_{y_{q_s-1}}) = (\boldsymbol{s}_{q_s-1}, \mathcal{C}_{q_s-1})].
\end{aligned}$$

Then, making use of the monotonicity of $\{e_k\}$ established in Lemma 4.5 and the fact that $y_i \geq x$ for all $i = 1, \cdots, q_s$, we get

$$\begin{aligned}
&\sum_{(\boldsymbol{s}_{q_s}, \mathcal{C}_{q_s}) \in \Psi} \text{Prob}\,[(\tilde{\boldsymbol{s}}_{y_{q_s}}, \tilde{\mathcal{C}}_{y_{q_s}}) = (\boldsymbol{s}_{q_s}, \mathcal{C}_{q_s}) \\
&\qquad\qquad |(\tilde{\boldsymbol{s}}_{y_{q_s-1}}, \tilde{\mathcal{C}}_{y_{q_s-1}}) = (\boldsymbol{s}_{q_s-1}, \mathcal{C}_{q_s-1})] \\
&\quad \geq 1 - e_{y_{q_s-1}} \geq 1 - e_x. \qquad (61)
\end{aligned}$$

Therefore, combining both cases, i.e., inequalities (59) and (61), we obtain the inequality

$$\begin{aligned}
&\sum_{(\boldsymbol{s}_{q_s}, \mathcal{C}_{q_s}) \in \Psi} \text{Prob}\,[(\tilde{\boldsymbol{s}}_{y_{q_s}}, \tilde{\mathcal{C}}_{y_{q_s}}) = (\boldsymbol{s}_{q_s}, \mathcal{C}_{q_s}) \\
&\qquad\qquad |(\tilde{\boldsymbol{s}}_{y_{q_s-1}}, \tilde{\mathcal{C}}_{y_{q_s-1}}) = (\boldsymbol{s}_{q_s-1}, \mathcal{C}_{q_s-1})] \\
&\quad \geq \min\{(1-d_x)^{y_{q_s}-y_{q_s-1}}, 1-e_x\} \\
&\quad \geq (1-d_x)^{y_{q_s}-y_{q_s-1}}(1-e_x).
\end{aligned}$$

Returning to (57) and using the inequality above, we obtain

$$\begin{aligned}
&\text{Prob}\,[\tilde{\boldsymbol{s}}_{y_{q_s}} = \boldsymbol{s}^* | (\tilde{\boldsymbol{s}}_x, \tilde{\mathcal{C}}_x) = (\boldsymbol{s}, \mathcal{C})] \\
&\quad \geq (1-d_x)^{y_{q_s}-y_{q_s-1}}(1-e_x) \\
&\quad\quad \times \sum_{\{(\boldsymbol{s}_i, \mathcal{C}_i), i=1,\cdots,q_s-1\} \in \Psi} \prod_{i=1}^{q_s-1} \text{Prob}\,[(\tilde{\boldsymbol{s}}_{y_i}, \tilde{\mathcal{C}}_{y_i}) \\
&\quad = (\boldsymbol{s}_i, \mathcal{C}_i) | (\tilde{\boldsymbol{s}}_{y_{i-1}}, \tilde{\mathcal{C}}_{y_{i-1}}) = (\boldsymbol{s}_{i-1}, \mathcal{C}_{i-1})].
\end{aligned}$$

This procedure can now be repeated by decomposing a subsequence ψ with $(q_s - 1)$ elements into its first $(q_s - 2)$ elements and the remaining element $(\boldsymbol{s}_{q_s-1}, \mathcal{C}_{q_s-1})$ and so on. Note that in this case the value of the last state at each step of this procedure, $\boldsymbol{s}_{q_s-i}, i = 1, \cdots, q_s$, is not necessarily s^*. However, if $\boldsymbol{s}_{q_s-i-1} = \boldsymbol{s}_{q_s-i} = s^*$, then Case 1 considered earlier applies; if $\boldsymbol{s}_{q_s-i-1} \neq s^*$, then Case 2 applies.

Thus, after q_s such steps, we arrive at

$$\begin{aligned}
&\text{Prob}\,[\tilde{\boldsymbol{s}}_{y_{q_s}} = \boldsymbol{s}^* | (\tilde{\boldsymbol{s}}_x, \tilde{\mathcal{C}}_x) = (\boldsymbol{s}, \mathcal{C})] \\
&\quad \geq (1-d_x)^{y_{q_s}-y_0}(1-e_x)^{q_s}.
\end{aligned}$$

Since $x \geq k/2 \geq \lfloor k/2 \rfloor$ according to (53) and since d_k and e_k are monotone decreasing according to (39) and (49), respectively, we have

$$d_x \leq d_{\lfloor k/2 \rfloor} \quad \text{and} \quad e_x \leq e_{\lfloor k/2 \rfloor}.$$

Thus

$$\begin{aligned}
&\text{Prob}\,[\tilde{\boldsymbol{s}}_{y_{q_s}} = \boldsymbol{s}^* | \tilde{\boldsymbol{s}}_x, \tilde{\mathcal{C}}_x] \\
&\quad \geq (1-d_{\lfloor k/2 \rfloor})^{y_{q_s}-y_0}(1-e_{\lfloor k/2 \rfloor})^q. \qquad (62)
\end{aligned}$$

On the other hand, noting that $y_{q_s} \leq k$ according to (54), consider $\{(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k)\}$ starting from $(\tilde{\boldsymbol{s}}_x, \tilde{\mathcal{C}}_x)$. Then

$$\begin{aligned}
&\text{Prob}\,[\tilde{\boldsymbol{s}}_k = \boldsymbol{s}^* | (\tilde{\boldsymbol{s}}_x, \tilde{\mathcal{C}}_x) = (\boldsymbol{s}, \mathcal{C})] \\
&\geq \sum_{\{\mathcal{C}_i, i=y_{q_s},\cdots,k\}} \text{Prob}\,[(\tilde{\boldsymbol{s}}_{y_{q_s}}, \tilde{\mathcal{C}}_{y_{q_s}}) = (\boldsymbol{s}^*, \mathcal{C}_{y_{q_s}}), (\tilde{\boldsymbol{s}}_i, \tilde{\mathcal{C}}_i) \\
&\qquad\qquad = (\boldsymbol{s}^*, \mathcal{C}_i), i = y_{q_s}+1, \cdots, k \\
&\qquad\qquad |(\tilde{\boldsymbol{s}}_x, \tilde{\mathcal{C}}_x) = (\boldsymbol{s}, \mathcal{C})]
\end{aligned}$$

where we have used the fact that $\tilde{\boldsymbol{s}}_{y_{q_s}} = \boldsymbol{s}^*$. Using, once again, the Markov property and the same argument as in Case 1 earlier to introduce d_k, we get

$$\begin{aligned}
&\sum_{\{\mathcal{C}_i, i=y_{q_s},\cdots,k\}} \text{Prob}\,[(\tilde{\boldsymbol{s}}_{y_{q_s}}, \tilde{\mathcal{C}}_{y_{q_s}}) = (\boldsymbol{s}^*, \mathcal{C}_{y_{q_s}}), (\tilde{\boldsymbol{s}}_i, \tilde{\mathcal{C}}_i) \\
&\qquad\qquad = (\boldsymbol{s}^*, \mathcal{C}_i), i = y_{q_{s_x}}+1, \cdots, k \\
&\qquad\qquad |(\tilde{\boldsymbol{s}}_x, \tilde{\mathcal{C}}_x) = (\boldsymbol{s}, \mathcal{C})] \\
&\quad = \sum_{\{\mathcal{C}_i, i=y_{q_s},\cdots,k\}} \text{Prob}\,[(\tilde{\boldsymbol{s}}_{y_{q_s}}, \tilde{\mathcal{C}}_{y_{q_s}}) = (\boldsymbol{s}^*, \mathcal{C}_{y_{q_s}})| \\
&\qquad\qquad \cdot (\tilde{\boldsymbol{s}}_x, \tilde{\mathcal{C}}_x) = (\boldsymbol{s}, \mathcal{C})] \\
&\quad\quad \times \prod_{h=y_{q_s}}^{k-1} \text{Prob}\,[(\tilde{\boldsymbol{s}}_{h+1}, \tilde{\mathcal{C}}_{h+1}) \\
&\qquad\qquad = (\boldsymbol{s}^*, \mathcal{C}_{h+1}) | (\tilde{\boldsymbol{s}}_h, \tilde{\mathcal{C}}_h) = (\boldsymbol{s}^*, \mathcal{C}_h)] \\
&\quad \geq (1-d_{\lfloor k/2 \rfloor})^{y_{q_s}-y_0}(1-e_{\lfloor k/2 \rfloor})^q \prod_{h=y_{q_s}}^{k-1}(1-d_h) \\
&\quad \geq (1-e_{\lfloor k/2 \rfloor})^q (1-d_{\lfloor k/2 \rfloor})^{k-x} \\
&\quad = (1-e_{\lfloor k/2 \rfloor})^q (1-d_{\lfloor k/2 \rfloor})^{q\alpha_k}.
\end{aligned}$$

Consequently

$$\begin{aligned}
\text{Prob}\,[\tilde{\boldsymbol{s}}_k = \boldsymbol{s}^*] &= E[\text{Prob}\,[\tilde{\boldsymbol{s}}_k = \boldsymbol{s}^* | (\tilde{\boldsymbol{s}}_x, \tilde{\mathcal{C}}_x) = (\boldsymbol{s}, \mathcal{C})]] \\
&\geq (1-e_{\lfloor k/2 \rfloor})^q (1-d_{\lfloor k/2 \rfloor})^{q\alpha_k} \\
&= (1-e_{\lfloor k/2 \rfloor})^q [(1-d_{\lfloor k/2 \rfloor})^{\alpha_k}]^q \to 1, \\
&\qquad \text{as } k \to \infty \qquad (63)
\end{aligned}$$

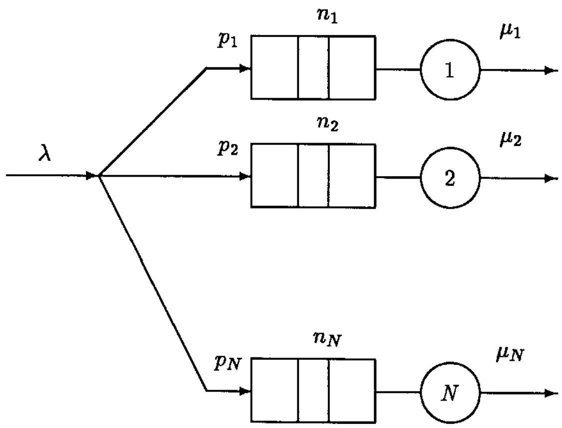


Fig. 1. Queueing system with N parallel servers.

where the limit follows from (50) in Lemma 4.5 and the choice of α_k satisfying (45). This proves that $\{\tilde{\boldsymbol{s}}_k\}$ converges to $\boldsymbol{s}^*$ in probability. ■

Remark: If the optimal allocation $\boldsymbol{s}^*$ is not unique, the analysis above can be extended to show that convergence is to a set of "equivalent" allocations as long as each optimum is neighboring at least one other optimum. When this arises in practice, what we often observe is oscillations between allocations that all yield optimal performance.

V. Applications and Simulation Results

The deterministic algorithm of Section III has been applied to several resource allocation problems for which one can readily verify that it converges to an optimal allocation and that it does so much faster than comparable algorithms such as exhaustive search, random search, and various forms of hill climbing (see [4]). In what follows, we will concentrate on the algorithm applied to *stochastic* discrete resource allocation problems.

As an application, we consider a buffer allocation problem for a queueing system as shown in Fig. 1, where each server represents a user and each buffer slot represents a resource that is to be allocated to a user. Jobs arrive at this system at a rate λ and are routed to one of the N users with some probability $p_i, i = 1 \cdots N$. Each user is processing jobs at a rate $\mu_i, i = 1, \cdots N$, and if a job is routed to a user with a full queue, the job is lost. Let $L_i(n_i)$ be the individual job loss probability of the ith server (n_i is the number of buffers allocated to the ith server). Our goal is to allocate all K available buffer slots to the users in order to minimize the objective function

$$\sum_{i=1}^{N} L_i(n_i) \quad \text{s.t.} \quad \sum_{i=1}^{N} n_i = K.$$

Remark: Here we have assumed that the coefficients of the individual cost functions are $\beta_i = 1$ for all $i = 1, \cdots, N$ just for testing purposes. Note, however, that one could use any coefficients to introduce job classes or a simple form of prioritization into the model.

Clearly, the structure of the objective function satisfies the separability assumption A1). In general, Assumption A3) is

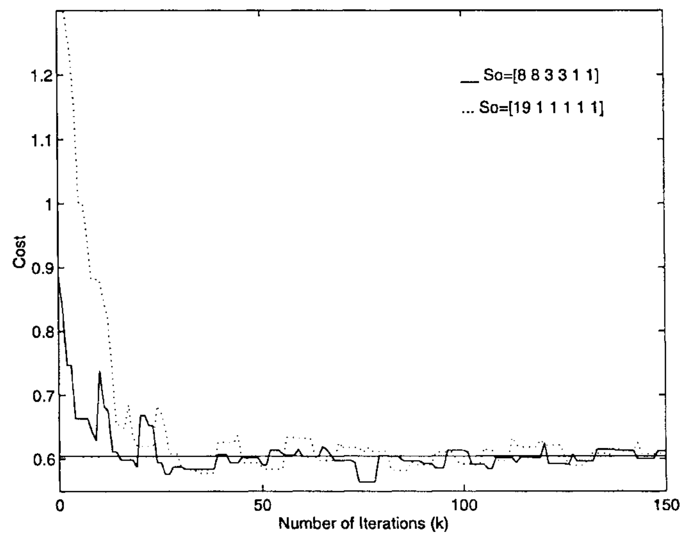


Fig. 2. Typical trace for different initial allocations.

not satisfied since there might be several permutations of an allocation that yield the same performance. However, this does not affect the convergence properties of our approach, as discussed in Section II. Assumptions A2) and A4) are both common for the problem considered and were verified through simulation. Finally, using unbiased performance estimators, one can also guarantee that A5) is satisfied and verify it through simulation.

For this problem, one can directly apply the algorithm corresponding to the process (28)–(32). For simplicity, we have assumed that the arrival process is Poisson with rate $\lambda = 5.0$, and all service times are exponential with rates $\mu_i = 1.0, i = 1, \cdots, 6$. Furthermore, all routing probabilities were chosen to be equal, i.e., $p_i = 1/6, i = 1, \cdots, 6$. Fig. 2 shows typical traces of the evolution of the algorithm for this system when $N = 6$ users and $K = 24$ buffer slots (resources). In this problem, it is obvious that the optimal allocation is [4, 4, 4, 4, 4, 4] due to symmetry.

The two traces of Fig. 2 correspond to two different initial allocations. In this case, the simulation length is increased linearly in steps of 3000 events per iteration, and one can easily see that even if we start at one of the worst possible allocations (e.g., [19, 1, 1, 1, 1, 1]), the cost quickly converges to the neighborhood of the optimal point.

The first issue that arises in the implementation of the algorithm is obtaining the finite differences $\boldsymbol{D} \equiv [\Delta\tilde{L}_1(n_1), \cdots, \Delta\tilde{L}_N(n_N)]$. Note that this is an online algorithm, so we can observe $[\tilde{L}_1(\overline{n}_1), \cdots, \tilde{L}_N(\overline{n}_N)]$, where $[\overline{n}_1, \cdots, \overline{n}_N]$ is the nominal allocation, i.e., the allocation that the actual system is presently using. In order to obtain the vector $\boldsymbol{D}$, we also need $\tilde{L}(\overline{n}_i - 1), i = 1, \cdots, N$, and for this purpose we assume that the reader is familiar with schemes that can extract such information from a single sample path such as finite perturbation analysis (FPA) and CE [2], [6].

In addition, every iteration requires an estimate $\Delta\tilde{L}_{j^*}(n_{j^*})$, which in turn requires $\tilde{L}_{j^*}(\overline{n}_{j^*}+1)$. However, we do not know in advance which user is going to be selected as j^*. If it turns out that some FPA or CE technique can be easily implemented

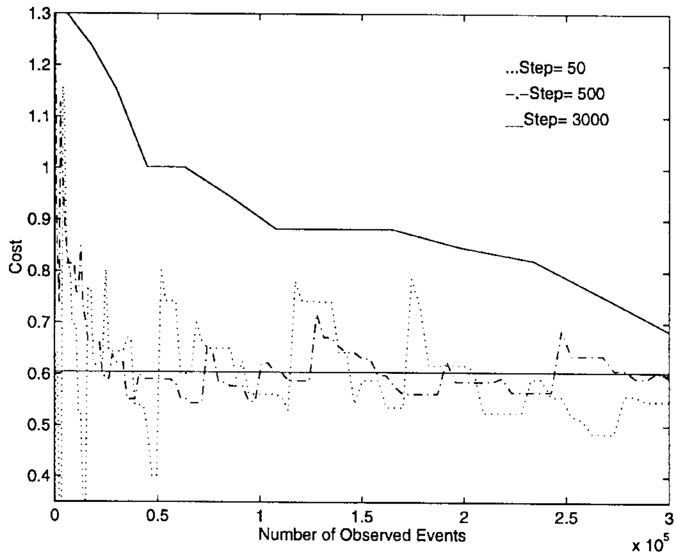


Fig. 3. Effect of $f(k)$ on the convergence of the algorithm.

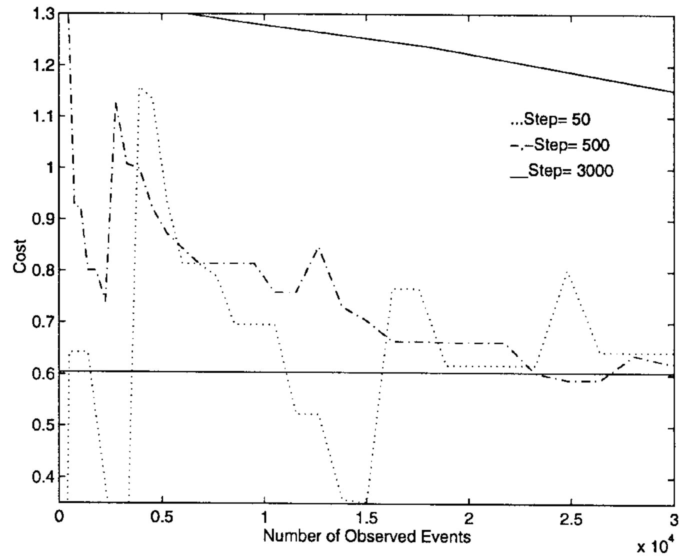


Fig. 4. Effect of $f(k)$ on the convergence of the algorithm.

with little additional computational cost, then one can use such a technique to evaluate $\tilde{L}_j(\overline{n}_j+1)$ for *all* $j=1,\cdots,N$ and then just use only the value corresponding to j^* when this user index becomes available. If this is not the case and this approach is computationally infeasible or too wasteful, an alternative is the following. An FPA or CE technique may be used to estimate the indexes i^*, j^* and then change the nominal allocation for some time interval in order to get an estimate of $\tilde{L}_{j^*}(\overline{n}_{j^*}+1)$. If it turns out that this change was appropriate, i.e., $\tilde{\delta}_k(\tilde{i}_k^*,\tilde{j}_k^*)>0$, then this new allocation is maintained; otherwise, we simply revert to the previous allocation and continue with the algorithm execution.

The next issue concerns the way in which we increase the simulation length $f(k)$. Here, we are faced with the usual tradeoff encountered in stochastic optimization algorithms: when k is small, using large values of $f(k)$ produces good estimates for the next iteration and hence the optimization process is more likely to make a step toward the optimal, but it forces the system to operate under a potentially high-cost allocation for a long period of time. Because of the *ordinal* nature of our algorithm, however, we take advantage of the fast convergence rate of ordinal estimates (see [7]) and rapidly reach a neighborhood of the optimal allocation even though estimates of the corresponding cost may not be very accurate. The effect of the way in which $f(k)$ is increased is seen in Fig. 3. When we increase $f(k)$ using large steps (3000 events), then we see that the algorithm converges to the optimal allocation monotonically, but slowly. When we increase $f(k)$ with smaller steps (500 events), then we see that we converge to the neighborhood of the optimal much faster at the expense of some oscillatory behavior. A magnified version of Fig. 3 is shown in Fig. 4, where it can be seen that the algorithm reaches the neighborhood of the optimal allocation in about 17 000 events. Finally, from Figs. 3 and 4 it can be easily seen that reducing $f(k)$ further (to 50 events) causes the system to change too fast resulting in slower convergence to the optimal allocation. Overall, however, it is worth pointing out the efficiency of this algorithm, since it is able to converge to the optimum by visiting only a small number of allocations out of a large search space; in the example considered here, the search space consists of $((K+N-1)!/K!(N-1)!)=118\,755$ allocations.

TABLE I
AVERAGE NUMBER OF ITERATIONS FOR DIFFERENT SYSTEM UTILIZATION

λ	Ave. k
5.0	182.0
3.0	75.2
1.0	110.3

A related issue arises in situations where the algorithm is used to track changes in the operating environment of the system (e.g., changes in the arrival or processing rates in the example of Fig. 1). In this case, if we allow $f(k)$ to become infinitely large, then we lose the ability to adapt the allocation to the new conditions. One may therefore be willing to sacrifice optimality for this adaptivity property.

Lastly, we have investigated the performance of the algorithm as the system utilization changes. Table I shows the average number of iterations required (over ten different initial allocations) before the system described above remains at the optimal allocation for $M=50$ consecutive iterations for different arrival rates when $f(0)=10\,000$ events and $f(k)$ is increased by 10 000 events at every iteration. As shown in Table I, when the system utilization is high (i.e., $\lambda=5.0$) then in order to get good estimates of the loss probability through simulation we need to run long simulations, and this is reflected in the high number of iterations required before we settle to the optimal allocation. When the utilization is reduced ($\lambda=3.0$), then convergence to the true loss probability is much faster, and therefore the algorithm settles at the optimal in fewer iterations. Finally, when we decrease the utilization even more ($\lambda=1.0$), the simulation estimates converge even faster; however, the difference between the objective functions at neighboring allocations becomes very small and the system ends up oscillating between near-optimal allocations, which in turn increases the number of iterations that are required before the system settles at the optimal allocation. Note that the average number of iterations may seem high; however, most of the time the system is already in the neighborhood of the optimal allocation.

VI. Conclusions and Future Work

We have considered a class of resource allocation problems which can be formulated as discrete optimization problems. We have derived necessary and sufficient conditions for a globally optimal solution of the deterministic version of the problem and proposed an explicit algorithm which we have shown to yield a globally optimal allocation. We have subsequently studied the stochastic version of the problem, in which costs can only be estimated, and proved that an appropriately modified version of this algorithm converges in probability to the optimal allocation, assuming this allocation is unique.

A crucial feature of our stochastic resource allocation algorithm is the fact that it is driven by *ordinal* estimates; that is, each allocation step is the result of simple comparisons of estimated quantities rather than their *cardinal* values. Based on recent results demonstrating the fast convergence of such estimates (often exponential, as in [7]), it is not surprising that the numerical results for the example presented in Section V suggest a very efficient resource allocation scheme for this type of problem. It remains, however, to further analyze the convergence rate of the process considered in Section IV, which is the subject of ongoing research. We also note that an attractive feature of the discrete resource allocation algorithm is the absence of a "step size" parameter, which is necessary in gradient-based schemes and often crucial in guaranteeing convergence.

As mentioned in the introduction, an alternative approach for solving such discrete resource allocation problems involves transforming them into continuous optimization problems which may be solved using standard iterative gradient-based schemes (see [3]). Although this approach was shown to be promising, a full analysis of its properties and comparison with the algorithm in this paper have yet to be carried out. Finally, we should point out that the resource allocation problems studied in this paper are based on certain assumptions regarding the cost structure of the problem. For example, we have considered cost functions in problem (***RA2***) which are separable in the sense that the ith user's cost function depends only on the number of resources this user is allocated. The relaxation of these assumptions and its implication to our approach is the subject of ongoing research.

Appendix I
Proofs of Lemmas Found in Section III

Proof of Lemma 3.1: To prove P1, first note that if $\delta_k \leq 0$, then, from (14), $\Delta L_{i^*_{k+1}}(n_{i^*_{k+1},k+1}) = \Delta L_{i^*_k}(n_{i^*_k,k})$. On the other hand, if $\delta_k > 0$, then there are two cases.

Case 1: If $i^*_k = i^*_{k+1}$ then, from A2)

$$\Delta L_{i^*_k}(n_{i^*_k,k}) > \Delta L_{i^*_k}(n_{i^*_k,k} - 1) = \Delta L_{i^*_{k+1}}(n_{i^*_{k+1},k+1}).$$

Case 2: If $i^*_k \neq i^*_{k+1} = p$, then there are two possible subcases.

Case 2.1: If $p = j^*_k$, since $\delta_k > 0$, we have

$$\Delta L_{i^*_k}(n_{i^*_k,k}) > \Delta L_p(n_{p,k} + 1) = \Delta L_{i^*_{k+1}}(n_{i^*_{k+1},k+1}).$$

Case 2.2: If $p \neq j^*_k$, then by the definition of i^*_k and the fact that $n_{p,k} = n_{p,k+1}$

$$\Delta L_{i^*_k}(n_{i^*_k,k}) \geq \Delta L_p(n_{p,k}) = \Delta L_{i^*_{k+1}}(n_{i^*_{k+1},k+1}).$$

■

The proof of P2 follows the lines of P1. More specifically, note that if $\delta_k \leq 0$, then, from (14) and the definition of j^*_k [i.e. (16)] $\Delta L_{j^*_{k+1}}(n_{j^*_{k+1},k+1}) \geq \Delta L_{j^*_k}(n_{j^*_k,k})$, so in this case $b = 1$. On the other hand, if $\delta_k > 0$, then there are two cases.

Case 1: If $j^*_k = j^*_{k+1}$ then, from A2)

$$\Delta L_{j^*_k}(n_{j^*_k,k}) < \Delta L_{j^*_k}(n_{j^*_k,k} + 1) = \Delta L_{j^*_{k+1}}(n_{j^*_{k+1},k+1})$$

and so again $b = 1$.

Case 2: If $j^*_k \neq j^*_{k+1} = p$, then there are two possible subcases.

Case 2.1: If $p = i^*_k$, since $\delta_k > 0$, we have

$$\Delta L_{j^*_{k+1}}(n_{j^*_{k+1},k+1}) = \Delta L_p(n_{p,k} - 1).$$

If now $\Delta L_p(n_{p,k} - 1) \geq \Delta L_{j^*_k}(n_{j^*_k,k})$, then $b = 1$. On the other hand, it is possible that $\Delta L_p(n_{p,k} - 1) < \Delta L_{j^*_k}(n_{j^*_k,k})$. In this case, we know that $j^*_{k+1} = p$. Using P1 and the fact that

$$\begin{aligned}\Delta L_{j^*_{k+1}}(n_{j^*_{k+1},k+1} + 1) &= \Delta L_p(n_{p,k+1} + 1) = \Delta L_p(n_{p,k}) \\ &= \Delta L_{i^*_k}(n_{i^*_k,k})\end{aligned}$$

it is clear that $\delta_{k+1} \leq 0$ which implies that $\mathcal{C}_{k+1} = \mathcal{C}_k - \{p\}$. This fact and the definition of j^*_k imply that $\Delta L_{j^*_{k+2}}(\cdot) \geq \Delta L_{j^*_k}(\cdot)$, so in this case $b = 2$.

Case 2.2: If $p \neq i^*_k$, then by the definition of j^*_k and the fact that $n_{p,k} = n_{p,k+1}$

$$\Delta L_{j^*_k}(n_{j^*_k,k}) \leq \Delta L_p(n_{p,k}) = \Delta L_{j^*_{k+1}}(n_{j^*_{k+1},k+1}).$$

■

Next, we prove property P3. First, note that when $p = i^*_k$ we must have $\delta_k > 0$. Otherwise, from (14), we get $n_{i,k+1} = n_{i,k}$ for all $i = 1, \cdots, N$. From (19), this implies that $i^*_{k+1} = i^*_k = p$, which violates our assumption that $p \neq i^*_l$ for $k < l < m$. Therefore, with $\delta_k > 0$, (14) implies that $n_{p,k+1} = n_{p,k} - 1$. In addition, $n_{p,m} = n_{p,k+1} = n_{p,k} - 1$, since $p \neq i^*_l$ for all l such that $k < l < m$ and $p \in \mathcal{C}_m$. We then have

$$\begin{aligned}\delta_m &= \Delta L_{i^*_m}(n_{i^*_m,m}) - \Delta L_p(n_{p,m} + 1) \\ &= \Delta L_{i^*_m}(n_{i^*_m,m}) - \Delta L_p(n_{p,k}) \\ &= \Delta L_{i^*_m}(n_{i^*_m,m}) - \Delta L_{i^*_k}(n_{i^*_k,k}) \leq 0\end{aligned}$$

where the last inequality is due to P1. Therefore, (23) immediately follows from (18). ■

To prove P4, first note that when $p = j^*_k$ we must have $\delta_k > 0$. If $\delta_k \leq 0$, then from (18), p is removed from $\mathcal{C}_k$, in which case $p \notin \mathcal{C}_m$ for any $m > k$ and it is not possible to have $p = i^*_m$ as assumed. Therefore, with $\delta_k > 0$, we get $n_{p,k+1} = n_{p,k} + 1$ from (14). Moreover, $n_{p,m} = n_{p,k+1} = n_{p,k} + 1$, since $p \neq j^*_l$ for all l such that $k < l < m$, and $p \in \mathcal{C}_m$. We now consider two possible cases.

Case 1: If $\delta_m > 0$, then $n_{p,m+1} = n_{p,m} - 1 = n_{p,k}$. The following subcases are now possible.

Case 1.1: If there is at least one $j \in \mathcal{C}_{m+1}$ such that $\Delta L_j(n_{j,m+1}) > \Delta L_{p,m+1}(n_{p,m+1})$, then we are assured that $i^*_{m+1} \neq p$. If $j^*_{m+1} = \arg\min_{i \in \mathcal{C}_{m+1}} \{\Delta L_i(n_{i,m+1})\}$ is unique, then, since $j^*_k = p$ and $n_{p,m+1} = n_{p,k}$, it follows from P2 that $j^*_{m+1} = p$. Now consider δ_{m+1} and observe that

$$\begin{aligned}\delta_{m+1} &= \Delta L_{i^*_{m+1}}(n_{i^*_{m+1},m+1}) - \Delta L_p(n_{p,m+1} + 1)\\ &= \Delta L_{i^*_{m+1}}(n_{i^*_{m+1},m+1}) - \Delta L_p(n_{p,m})\\ &= \Delta L_{i^*_{m+1}}(n_{i^*_{m+1},m+1}) - \Delta L_{i^*_m}(n_{i^*_m,m}) \leq 0\end{aligned}$$

where the last inequality is due to P1. Therefore, from (18), $\mathcal{C}_{m+2} = \mathcal{C}_{m+1} - \{p\}$ and (24) holds for $q = 1$.

If, on the other hand, j^*_{m+1} is not unique, then it is possible that $j^*_{m+1} \neq p$ since we have assumed that ties are arbitrarily broken. In this case, there are at most $q \leq N - 1$ steps before $j^*_{m+q} = p$. This is because at step $m + 1$ either $\delta_{m+1} \leq 0$ and j^*_{m+1} is removed from $\mathcal{C}_{m+1}$, or $\delta_{m+1} > 0$ and, from (14), $n_{j^*_{m+2},m+2} = n_{j^*_{m+1},m+1} + 1$, in which case $\Delta L_{j^*_{m+2}}(n_{j^*_{m+2},m+2}) > \Delta L_{j^*_{m+1}}(n_{j^*_{m+1},m+1})$ from A2). The same is true for any of the q steps after m. Then at step $m + q + 1$, we get $\delta_{m+q+1} \leq 0$ by arguing exactly as in the case where j^*_{m+1} is unique, with $m+1$ replaced by $m+q+1$, and again (24) holds.

Case 1.2: If $\Delta L_j(n_{j,m+1})$ is the same for all $j \in \mathcal{C}_{m+1}$, then it is possible that $i^*_{m+1} = p$. In this case, $\delta_l < 0$ for all $l > m + 1$ due to A2). Therefore, j^*_{m+1} will be removed from $\mathcal{C}_{m+1}$ through (18). Moreover, since $i^*_{m+1} = p$ by (19), this process repeats itself for at most $q \leq N - 1$ steps resulting in $\mathcal{C}_{m+q+1} = \{p\}$.

Case 2: If $\delta_m \leq 0$ and $j^*_m = r$ where $r \neq p$, then $\mathcal{C}_{m+1} = \mathcal{C}_m - \{r\}$. In this case, note that $i^*_{m+1} = i^*_m = p$, and depending on the sign of δ_{m+1}, we either go to Case 1 or we repeat the process of removing one additional user index from the $\mathcal{C}_{m+1}$ set. In the event that $\delta_l \leq 0$ for all $l > m$, all j^*_l will be removed from the $\mathcal{C}_l$ set. The only remaining element in this set is p, which reduces to Case 1.2 above. ■

Property P5 follows from P3 by observing in (14) that the only way to get $n_{p,m} > n_{p,k}$ is if $j^*_l = p$ and $\delta_l > 0$ for some $k < l < m$. However, P3 asserts that this is not possible, since p would be removed from $\mathcal{C}_l$. ■

Property P6 follows from P4 by a similar argument. The only way to get $n_{p,m} < n_{p,k}$ is if $i^*_l = p$ and $\delta_l > 0$ for some $k < l < m$. However, it is clear from the proof of P4 that p would either be removed from $\mathcal{C}_l$, possibly after a finite number of steps, or simply remain in this set until it is the last element in it. ■

Proof of Lemma 3.2: We begin by first establishing the fact that the process terminates in a finite number of steps bounded by $K(N + 1)$. This is easily seen as follows. At any step k, the process determines some i^*_k (say p) with two possibilities: 1) either user p gives one resource to some other user through (14) or 2) one user index is removed from $\mathcal{C}_k$ through (18), in which case $i^*_{k+1} = p$, and we have the exact same situation as in step k [if Case 2) persists, clearly $|\mathcal{C}_l| = 1$ for some $l \leq k + N - 1$]. Under Case 1), because of property P5, p cannot receive any resources from other users, therefore in the worst case p will give away all of its initial resources to other users and will subsequently not be able to either give or receive resources from other users. Since $n_{p,k} \leq K$ for any k, it follows that p can be involved in a number of steps that is bounded by $K + 1$, where one is the extra step when p is removed from $\mathcal{C}_k$ at some k. Finally, since there are N users undergoing this series of steps, in the worst case the process terminates in $N(K + 1)$ steps.

This simple upper bound serves to establish the fact that the process always terminates in a finite number of steps. We will use this fact together with some of the properties in Lemma 3.1 to find a tighter upper bound. Let the initial allocation be $\boldsymbol{s}_0$. Since the process always terminates in a finite number of steps, there exists some final allocation $\bar{\boldsymbol{s}} = [\overline{n}_1, \cdots, \overline{n}_N]$ which, given $\boldsymbol{s}_0$, is unique since the algorithm is deterministic. An allocation $n_{i,k}$ at the kth step can be written as follows:

$$n_{i,k} = \overline{n}_i + d_{i,k}$$

where $d_{i,k} = \{-K, \cdots, -1, 0, 1, \cdots, K\}$ and $\Sigma_{i=1}^N d_{i,k} = 0$ for all $k = 0, 1, \cdots$, since all allocations are feasible. Now define the following three sets:

$$A_k = \{i\colon\ d_{i,k} > 0\}, \qquad B_k = \{i\colon\ d_{i,k} = 0\}$$
$$C_k = \{i\colon\ d_{i,k} < 0\}$$

and note that at the final state $d_i = 0$ for all $i = 1, \cdots, N$. Due to P3, at every step we have $i^*_k \in A_k$ (recall that once a user is selected as i^*_k it can only give away resources to other users). Similarly, due to P4, $j^*_k \in B_k \cup C_k$.

At every step of the process, there are only two possibilities.

1) If $\delta_k > 0$, let $p = i^*_k \in A_k$ and $q = j^*_k \in B_k \cup C_k$. Then, at the next step, (14) implies that $d_{p,k+1} = d_{p,k} - 1$ and $d_{q,k+1} = d_{q,k} + 1$.
2) If $\delta_k \leq 0$, then a user index from B_k is removed from the set $\mathcal{C}_k$.

Moreover, from the definitions of the three sets above, we have

$$\sum_{i=1}^N d_{i,k} = 0 = \sum_{\substack{i=1\\ i\in A_k}}^N d_{i,k} + \sum_{\substack{i=1\\ i\in C_k}}^N d_{i,k}$$

and, therefore, we can write

$$\mathcal{P}_k = \sum_{\substack{i=1\\ i\in A_k}}^N d_{i,k} = -\sum_{\substack{i=1\\ i\in C_k}}^N d_{i,k}$$

where $0 \leq \mathcal{P}_k \leq K$ for all $k = 0, 1, \cdots$ since $0 \leq n_{i,k} \leq K$.

Now let $\mathcal{P}_0$ be the initial value of $\mathcal{P}_k$ and let $|A_0|$ be the initial cardinality of the set A_k. We separate the number of steps required to reach the final allocation into three categories.

1) Clearly, $\mathcal{P}_0$ steps (not necessarily contiguous) are required to make $\mathcal{P}_k = 0$ for some $k \geq 0$ by removing

one resource at each such step form users $i \in A_l, l \leq k$. During any such step, we have $\delta_l > 0$ as in Case 1 above.

2) These $\mathcal{P}_0$ steps would suffice to empty the set A_k if it were impossible for user indexes to be added to it from the set $B_l, l \leq k$. However, from property P4 it is possible for a user j such that $j \in B_k$ and $j \notin A_l$ for all $l < k$ to receive at most one resource, in which case we have $j \in A_k$. There are at most $N - |A_0|$ users with such an opportunity, and hence $N - |A_0|$ additional steps are possible. During any such step, as in 1), we have $\delta_l > 0$ as in Case 1 above.

3) Finally, we consider steps covered by Case 2 above. Clearly, $N - 1$ steps are required to reach $|\mathcal{C}_k| = 1$ for some k.

Therefore, the number of steps L required to reach the final allocation is such that $L \leq \mathcal{P}_0 + N - |A_0| + N - 1$. Observing that $\mathcal{P}_0 \leq K$ and $|A_0| \geq 1$, we get $L \leq K + 2(N-1)$. Note that $|A_0| = 0$ implies that the $\boldsymbol{s}_0 = \bar{\boldsymbol{s}}$, and in this case only $N-1$ steps [see 3) above] are required to reach the final state. Thus, $N - 1$ is the lower bound on the required number of steps. ■

Appendix II
Proofs of Lemmas Found in Section IV

Proof of Lemma 4.1: Let

$$\tilde{x}_k = \Delta \tilde{L}_i^{f(k)}(n_i) - \Delta \tilde{L}_j^{f(k)}(n_j)$$
$$x = \Delta L_i(n_i) - \Delta L_j(n_j) < 0.$$

Then, Assumption A4) and $\lim_{k\to\infty} f(k) = \infty$ guarantee that

$$\lim_{k\to\infty} \tilde{x}_k = x, \quad \text{a.s.}$$

Since a.s. convergence implies convergence in probability, we know that for every $\epsilon > 0$

$$\lim_{k\to\infty} \text{Prob}\,[|\tilde{x}_k - x| \geq \epsilon] = 0.$$

Setting $\epsilon = -x > 0$, we obtain

$$\lim_{k\to\infty} \text{Prob}\,[|\tilde{x}_k - x| \geq -x] = 0. \tag{64}$$

Finally, since $\text{Prob}\,[\tilde{x}_k \geq 0] \leq \text{Prob}\,[|\tilde{x}_k - x| \geq -x]$, it immediately follows from (64) that

$$\lim_{k\to\infty} \text{Prob}\,[\tilde{x}_k \geq 0] = 0$$

which is the statement of the lemma. ■

Proof of Lemma 4.2: Let $\delta_k(i,j)$ be defined as in Assumption A5). Given $\tilde{\boldsymbol{s}}_k = \boldsymbol{s}, \tilde{\mathcal{C}}_k = \mathcal{C}$, consider the event $L(\tilde{\boldsymbol{s}}_{k+1}) > L(\tilde{\boldsymbol{s}}_k)$. According to the process (28)–(32) and (20):

$$\text{If } \tilde{\delta}_k(\tilde{i}_k^*, \tilde{j}_k^*) > 0, \text{ then } L(\tilde{\boldsymbol{s}}_{k+1}) - L(\tilde{\boldsymbol{s}}_k) = -\delta_k(\tilde{i}_k^*, \tilde{j}_k^*)$$
$$\text{and if } \tilde{\delta}_k(\tilde{i}_k^*, \tilde{j}_k^*) \leq 0, \text{ then } L(\tilde{\boldsymbol{s}}_{k+1}) = L(\tilde{\boldsymbol{s}}_k).$$

Therefore, $L(\tilde{\boldsymbol{s}}_{k+1}) > L(\tilde{\boldsymbol{s}}_k)$ occurs if and only if

$$\tilde{\delta}_k(\tilde{i}_k^*, \tilde{j}_k^*) > 0 \quad \text{and} \quad \delta_k(\tilde{i}_k^*, \tilde{j}_k^*) < 0.$$

Then

$$\begin{aligned}
&\text{Prob}\,[L(\tilde{\boldsymbol{s}}_{k+1}) > L(\tilde{\boldsymbol{s}}_k) | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\
&\quad = \text{Prob}\,[\tilde{\delta}_k(\tilde{i}_k^*, \tilde{j}_k^*) > 0 \quad \text{and} \quad \delta_k(\tilde{i}_k^*, \tilde{j}_k^*) < 0] \\
&\quad = \sum_{\{(i,j)|\delta_k(i,j)<0\}} \text{Prob}\,[\Delta \tilde{L}_i^{f(k)}(n_i) \\
&\qquad\qquad - \Delta \tilde{L}_j^{f(k)}(n_j + 1) > 0 \\
&\qquad\qquad \text{and} \quad (\tilde{i}_k^*, \tilde{j}_k^*) = (i,j)] \\
&\quad \leq \sum_{\{(i,j)|\delta_k(i,j)<0\}} \text{Prob}\,[\Delta \tilde{L}_i^{f(k)}(n_i) \\
&\qquad\qquad - \Delta \tilde{L}_j^{f(k)}(n_j + 1) > 0].
\end{aligned} \tag{65}$$

For each pair of (i,j) satisfying $\delta_k(i,j) < 0$, we know from Lemma 4.1 that

$$\lim_{k\to\infty} \text{Prob}\,[\Delta \tilde{L}_i^{f(k)}(n_i) - \Delta \tilde{L}_j^{f(k)}(n_j + 1) > 0] = 0.$$

Taking this limit in (65), and also noticing the finiteness of the set $\{(i,j)|\delta_k(i,j) < 0\}$ for any pair $(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})$, we obtain

$$\begin{aligned}
&\lim_{k\to\infty} d_k(\boldsymbol{s}, \mathcal{C}) \\
&\quad = \lim_{k\to\infty} \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{k+1}) > L(\tilde{\boldsymbol{s}}_k) | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] = 0
\end{aligned}$$

and the proof of (38) is complete.

Definition (39) immediately implies that d_k is monotone decreasing and that $d_k \geq d_k(\boldsymbol{s}, \mathcal{C})$. The limit (40) then follows from (38). ■

Proof of Lemma 4.4: Given $\tilde{\boldsymbol{s}}_k = \boldsymbol{s}, \tilde{\mathcal{C}}_k = \mathcal{C}$, consider the event $L(\tilde{\boldsymbol{s}}_{k+1}) = L(\tilde{\boldsymbol{s}}_k)$. According to the process (28)–(32) and (20)

$$\text{If } \tilde{\delta}_k(\tilde{i}_k^*, \tilde{j}_k^*) > 0, \text{ then } L(\tilde{\boldsymbol{s}}_{k+1}) - L(\tilde{\boldsymbol{s}}_k) = -\delta_k(\tilde{i}_k^*, \tilde{j}_k^*)$$
$$\text{and if } \tilde{\delta}_k(\tilde{i}_k^*, \tilde{j}_k^*) \leq 0, \text{ then } L(\tilde{\boldsymbol{s}}_{k+1}) = L(\tilde{\boldsymbol{s}}_k).$$

Therefore, $L(\tilde{\boldsymbol{s}}_{k+1}) = L(\tilde{\boldsymbol{s}}_k)$ occurs if and only if

$$\begin{aligned}
&\text{either } \{\tilde{\delta}_k(\tilde{i}_k^*, \tilde{j}_k^*) > 0, \quad \delta_k(\tilde{i}_k^*, \tilde{j}_k^*) = 0\} \\
&\text{or } \{\tilde{\delta}_k(\tilde{i}_k^*, \tilde{j}_k^*) \leq 0\}.
\end{aligned} \tag{66}$$

for every k.

For notational convenience, consider, for any k, the events

$$A_k^+ = \{\tilde{\delta}_k(\tilde{i}_k^*, \tilde{j}_k^*) > 0, \quad \delta_k(\tilde{i}_k^*, \tilde{j}_k^*) = 0\}$$

and

$$A_k^- = \{\tilde{\delta}_k(\tilde{i}_k^*, \tilde{j}_k^*) \leq 0\}.$$

Next, for any $i \geq 1$, define the following subset of $\{k, \cdots, k+i-1\}$:

$$\tilde{R}(i) = \{h\colon \tilde{\delta}_h(\tilde{i}_h^*, \tilde{j}_h^*) \leq 0, h \in \{k, \cdots, k+i-1\}\}$$

and let $\tilde{I}_k$ be the cardinality of the set $\tilde{R}(\alpha_k)$. In addition, for any given integer I, let $\tilde{R}_I(\alpha_k)$ denote such a set with exactly I elements. Then, define the set

$$\tilde{Q}_I(\alpha_k) = \{k, \cdots, k+\alpha_k-1\} - \tilde{R}_I(\alpha_k)$$

containing all indexes $h \in \{k, \cdots, k+\alpha_k-1\}$ which do not satisfy $\tilde{\delta}_h(\tilde{i}_h^*, \tilde{j}_h^*) \le 0$.

Finally, define

$$\mathcal{A}_I^+(\alpha_k) = \{A_h^+, \text{for all } h \in \tilde{Q}_I(\alpha_k)\}$$

and

$$\mathcal{A}_I^-(\alpha_k) = \{A_h^-, h \in \tilde{R}_I(\alpha_k)\}.$$

Depending on the value of $\tilde{I}_k$ defined above, we can write

$$\begin{aligned}\text{Prob}\,&[L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), h = k, \cdots, k+\alpha_k-1 \\ &|(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &= \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), h = k, \cdots, k+\alpha_k-1, \\ &\qquad \tilde{I}_k \le |\mathcal{C}| + N | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\quad + \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), h = k, \cdots, k+\alpha_k-1, \\ &\qquad \tilde{I}_k > |\mathcal{C}| + N | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})]. \end{aligned} \tag{67}$$

We will now consider each of the two terms in (67) separately.

The first term in (67) can be rewritten as

$$\begin{aligned}\text{Prob}\,&[L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), h = k, \cdots, k+\alpha_k-1, \\ &\tilde{I}_k \le |\mathcal{C}| + N | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &= \sum_{I=0}^{|\mathcal{C}|+N} \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), h = k, \cdots, k+\alpha_k-1, \\ &\qquad \tilde{I}_k = I | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})]. \end{aligned} \tag{68}$$

Using the notation we have introduced, observe that

$$\begin{aligned}\text{Prob}\,&[L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), h = k, \cdots, k+\alpha_k-1, \\ &\tilde{I}_k = I | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &= \sum_{\tilde{R}_I(\alpha_k)} \text{Prob}\,[\mathcal{A}_I^-(\alpha_k), \mathcal{A}_I^+(\alpha_k) | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &= \sum_{\tilde{R}_I(\alpha_k)} \text{Prob}\,[\mathcal{A}_I^+(\alpha_k) | \mathcal{A}_I^-(\alpha_k), (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\qquad \cdot \text{Prob}\,[\mathcal{A}_I^-(\alpha_k) | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})]. \end{aligned} \tag{69}$$

Set $h' = k + \alpha_k - 1$ and, without loss of generality, assume that $h' \notin \tilde{R}_I(\alpha_k)$ (otherwise, if $h' \in \tilde{R}_I(\alpha_k)$ there must exist some M such that $k + \alpha_k - M \notin \tilde{R}_I(\alpha_k)$, and the same argument may be used with $H' = k + \alpha_k - M$). Then

$$\begin{aligned}\text{Prob}\,&[\mathcal{A}_I^+(\alpha_k) | \mathcal{A}_I^-(\alpha_k), (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &= \text{Prob}\,[A_{h'}^+, \mathcal{A}_I^+(\alpha_k - 1) | \mathcal{A}_I^-(\alpha_k), (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &= \sum_{(\boldsymbol{s}', \mathcal{C}')} \text{Prob}\,[A_{h'}^+, (\tilde{\boldsymbol{s}}_{h'}, \tilde{\mathcal{C}}_{h'}) = (\boldsymbol{s}', \mathcal{C}'), \\ &\qquad \mathcal{A}_I^+(\alpha_k - 1) | \mathcal{A}_I^-(\alpha_k), (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})]. \end{aligned} \tag{70}$$

Recalling the definition of $A_{h'}^+$, we can write

$$\begin{aligned}\text{Prob}\,&[\mathcal{A}_{h'}^+, (\tilde{\boldsymbol{s}}_{h'}, \tilde{\mathcal{C}}_{h'}) = (\boldsymbol{s}', \mathcal{C}'), \mathcal{A}_I^+(\alpha_k - 1) | \mathcal{A}_I^-(\alpha_k), \\ &(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &= \text{Prob}\,[\tilde{\delta}_{h'}(\tilde{i}_{h'}^*, \tilde{j}_{h'}^*) > 0 | \delta_{h'}(\tilde{i}_{h'}^*, \tilde{j}_{h'}^*) = 0, \\ &\qquad (\tilde{\boldsymbol{s}}_{h'}, \tilde{\mathcal{C}}_{h'}) = (\boldsymbol{s}', \mathcal{C}'), \mathcal{A}_I^+(\alpha_k - 1), \mathcal{A}_I^-(\alpha_k), \\ &\qquad (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\quad \times \text{Prob}\,[\delta_{h'}(\tilde{i}_{h'}^*, \tilde{j}_{h'}^*) = 0, (\tilde{\boldsymbol{s}}_{h'}, \tilde{\mathcal{C}}_{h'}) = (\boldsymbol{s}', \mathcal{C}'), \\ &\qquad \mathcal{A}_I^+(\alpha_k - 1) | \mathcal{A}_I^-(\alpha_k), (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})]. \end{aligned}$$

Then, the Markov property of the process (28)–(32) implies that

$$\begin{aligned}\text{Prob}\,&[\mathcal{A}_{h'}^+, (\tilde{\boldsymbol{s}}_{h'}, \tilde{\mathcal{C}}_{h'}) = (\boldsymbol{s}', \mathcal{C}'), \mathcal{A}_I^+(\alpha_k - 1) | \mathcal{A}_I^-(\alpha_k), \\ &(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &= \text{Prob}\,[\tilde{\delta}_{h'}(\tilde{i}_{h'}^*, \tilde{j}_{h'}^*) > 0 | \delta_{h'}(\tilde{i}_{h'}^*, \tilde{j}_{h'}^*) = 0, \\ &\qquad (\tilde{\boldsymbol{s}}_{h'}, \tilde{\mathcal{C}}_{h'}) = (\boldsymbol{s}', \mathcal{C}')] \\ &\quad \times \text{Prob}\,[\delta_{h'}(\tilde{i}_{h'}^*, \tilde{j}_{h'}^*) = 0, (\tilde{\boldsymbol{s}}_{h'}, \tilde{\mathcal{C}}_{h'}) = (\boldsymbol{s}', \mathcal{C}'), \\ &\qquad \mathcal{A}_I^+(\alpha_k - 1) | \mathcal{A}_I^-(\alpha_k), (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})]. \end{aligned} \tag{71}$$

However, by Assumption A5)

$$\begin{aligned}\text{Prob}\,&[\tilde{\delta}_{h'}(\tilde{i}_{h'}^*, \tilde{j}_{h'}^*) > 0 | \delta_{h'}(\tilde{i}_{h'}^*, \tilde{j}_{h'}^*) = 0, (\tilde{\boldsymbol{s}}_{h'}, \tilde{\mathcal{C}}_{h'}) = (\boldsymbol{s}', \mathcal{C}')] \\ &\le 1 - p_0. \end{aligned}$$

Thus, (71) becomes

$$\begin{aligned}\text{Prob}\,&[\mathcal{A}_{h'}^+, (\tilde{\boldsymbol{s}}_{h'}, \tilde{\mathcal{C}}_{h'}) \\ &= (\boldsymbol{s}', \mathcal{C}'), \mathcal{A}_I^+(\alpha_k - 1) | \mathcal{A}_I^-(\alpha_k), (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\le (1-p_0)\,\text{Prob}\,[\delta_{h'}(\tilde{i}_{h'}^*, \tilde{j}_{h'}^*) = 0, (\tilde{\boldsymbol{s}}_{h'}, \tilde{\mathcal{C}}_{h'}) = (\boldsymbol{s}', \mathcal{C}'), \\ &\qquad \mathcal{A}_I^+(\alpha_k - 1) | \mathcal{A}_I^-(\alpha_k), (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\le (1-p_0)\,\text{Prob}\,[(\tilde{\boldsymbol{s}}_{h'}, \tilde{\mathcal{C}}_{h'}) = (\boldsymbol{s}', \mathcal{C}'), \\ &\qquad \mathcal{A}_I^+(\alpha_k - 1) | \mathcal{A}_I^-(\alpha_k), (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})]. \end{aligned}$$

Using this inequality in (70), we obtain

$$\begin{aligned}\text{Prob}\,&[\mathcal{A}_I^+(\alpha_k) | \mathcal{A}_I^-(\alpha_k), (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\le (1-p_0)\,\text{Prob}\,[\mathcal{A}_I^+(\alpha_k - 1) | \mathcal{A}_I^-(\alpha_k), \\ &\qquad (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})]. \end{aligned}$$

Continuing this recursive procedure, we finally arrive at

$$\text{Prob}\,[\mathcal{A}_I^+(\alpha_k) | \mathcal{A}_I^-(\alpha_k), (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \le (1-p_0)^{\alpha_k - I}$$

which allows us to obtain the following inequality from (68) and (69):

$$\begin{aligned}\text{Prob}\,&[L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), h = k, \cdots, k+\alpha_k-1, \\ &\tilde{I}_k \le |\mathcal{C}| + N | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\le \sum_{I=0}^{|\mathcal{C}|+N} \sum_{\tilde{R}_I(\alpha_k)} (1-p_0)^{\alpha_k - I} \\ &\qquad \cdot \text{Prob}\,[\mathcal{A}_I^-(\alpha_k) | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\le \sum_{I=0}^{|\mathcal{C}|+N} (1-p_0)^{\alpha_k - I} \le p_0^{-1}(1-p_0)^{\alpha_k - (|\mathcal{C}|+N)}. \end{aligned} \tag{72}$$

Since $0 \leq 1 - p_0 < 1$ by Assumption A5), and since $\lim_{k\to\infty} \alpha_k = \infty$ according to (44), the preceding inequality implies that the first term in (67) is such that

$$\lim_{k\to\infty} \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), h = k, \cdots, k + \alpha_k - 1, \tilde{I}_k < |\mathcal{C}| + N | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] = 0. \tag{73}$$

Next we consider the second term in (67). Let $\tilde{J}_k$ be the first step after k that we have either $\tilde{i}_h^* \notin A_h^{\max}$ or $\tilde{j}_h^* \notin A_h^{\min}, h = k, k+1, \cdots, k+\alpha_k - 1$. Clearly, $k \leq \tilde{J}_k \leq k + \alpha_k$. We also use (without confusion) $\tilde{J}_k = k + \alpha_k$ to mean that $\tilde{i}_h^* \in A_h^{\max}$ and $\tilde{j}_h^* \in A_h^{\min}$ for all $h = k, k+1, \cdots, k+\alpha_k - 1$. Then the second term in (67) can be written as

$$\begin{aligned} &\text{Prob}\,[L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), h = k, \cdots, k+\alpha_k - 1, \tilde{I}_k > |\mathcal{C}| + N | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &= \sum_{J=k}^{k+\alpha_k-1} \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), h = k, \cdots, k + \alpha_k - 1, \tilde{I}_k > |\mathcal{C}| + N, \tilde{J}_k = J | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\quad + \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), \tilde{i}_h^* \in A_h^{\max}, \tilde{j}_h^* \in A_h^{\min}, h = k, \cdots, k+\alpha_k - 1, \tilde{I}_k > |\mathcal{C}| + N (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})]. \end{aligned} \tag{74}$$

We shall now consider each of the two terms in (74) separately. In the first term, for any $J, k \leq J < k + \alpha_k$, we have

$$\begin{aligned} &\text{Prob}\,[L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), h = k, \cdots, k + \alpha_k - 1, \tilde{I}_k > |\mathcal{C}| + N, \tilde{J}_k = J | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\leq \text{Prob}\,[\tilde{J}_k = J | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &= \sum_{\{(\boldsymbol{s}', \mathcal{C}')\}} \text{Prob}\,[\tilde{J}_k = J | (\tilde{\boldsymbol{s}}_J, \tilde{\mathcal{C}}_J) = (\boldsymbol{s}', \mathcal{C}')] \\ &\quad \cdot \text{Prob}\,[(\tilde{\boldsymbol{s}}_J, \tilde{\mathcal{C}}_J) = (\boldsymbol{s}', \mathcal{C}') | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \end{aligned} \tag{75}$$

where the second step above follows from the Markov property of (28)–(32). Moreover

$$\begin{aligned} &\text{Prob}\,[\tilde{J}_k = J | (\tilde{\boldsymbol{s}}_J, \tilde{\mathcal{C}}_J) = (\boldsymbol{s}', \mathcal{C}')] \\ &\leq \text{Prob}\,[\{\tilde{i}_J^* \notin A_J^{\max}\} \cup \{\tilde{j}_J^* \notin A_J^{\min}\} | (\tilde{\boldsymbol{s}}_J, \tilde{\mathcal{C}}_J) = (\boldsymbol{s}', \mathcal{C}')] \\ &\leq \text{Prob}\,[\tilde{i}_J^* \notin A_J^{\max} | (\tilde{\boldsymbol{s}}_J, \tilde{\mathcal{C}}_J) = (\boldsymbol{s}', \mathcal{C}')] \\ &\quad + \text{Prob}\,[\tilde{j}_J^* \notin A_J^{\min} | (\tilde{\boldsymbol{s}}_J, \tilde{\mathcal{C}}_J) = (\boldsymbol{s}', \mathcal{C}')] \\ &\leq a_J + b_J \leq a_k + b_k \end{aligned}$$

where we have used (36), (37), (42), and the monotonicity of a_k and b_k. This inequality, together with (75), implies that

$$\sum_{J=k}^{k+\alpha_k-1} \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), h = k, \cdots, k + \alpha_k - 1, \tilde{I}_k > |\mathcal{C}| + N, \tilde{J}_k = J | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \leq (\alpha_k - 1)(a_k + b_k).$$

By Lemma 4.3 and (44) it follows that

$$\lim_{k\to\infty} \sum_{J=k}^{k+\alpha_k-1} \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), h = k, \cdots, k + \alpha_k - 1, \tilde{I}_k > |\mathcal{C}| + N, \tilde{J}_k = J | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] = 0. \tag{76}$$

As for the second term in (74), we note the following facts.

1) Given that $\tilde{\boldsymbol{s}}_k \neq \boldsymbol{s}^*$, then either there is a $j \in \tilde{\mathcal{C}}_k$ such that $\delta_k(i_k^*, j) > 0$ for any $i_k^* \in A_k^{\max}$, or the set $\tilde{\mathcal{C}}_h$ first decreases to $|\tilde{\mathcal{C}}_h| = 1$ according to (32) and then is reset to $\mathcal{C}_0$, in which case there is a $j \in \mathcal{C}_0$ such that $\delta_k(i_k^*, j) > 0$ (otherwise $\tilde{\boldsymbol{s}}_k$ would be the optimum according to Theorem 1). Therefore, without loss of generality, we assume that there is a $j \in \tilde{\mathcal{C}}_k$ such that $\delta_k(i_k^*, j) > 0$ for any $i_k^* \in A_k^{\max}$.
2) As long as (66) holds and $\tilde{i}_h^* \in A_h^{\max}, \tilde{j}_h^* \in A_h^{\min}$

$$\max_{j\in\tilde{\mathcal{C}}_h} \{\Delta L_i(\tilde{n}_{i,j})\} = \max_{j\in\tilde{\mathcal{C}}_k} \{\Delta L_i(\tilde{n}_{i,j})\}, \quad h = k, k+1, \cdots, k+\alpha_k - 1.$$

3) One user is deleted from the set $\tilde{\mathcal{C}}_h$ every time $\tilde{\delta}_k(\tilde{i}_h^*, \tilde{j}_h^*) \leq 0$.

The previous facts 1)–3) imply that, when

$$L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), \quad \tilde{i}_h^* \in A_h^{\max} \tilde{j}_h^* \in A_h^{\min}, \quad h = k, \cdots, k+\alpha_k - 1, \quad \tilde{I}_k > |\mathcal{C}| + N$$

with probability one there exists a $\tilde{M}_k, k \leq \tilde{M}_k \leq k + \alpha_k - 1$ such that

$$\tilde{\delta}_{\tilde{M}_k}(\tilde{i}_{\tilde{M}_k}^*, \tilde{j}_{\tilde{M}_k}^*) \leq 0, \qquad \delta_{\tilde{M}_k}(\tilde{i}_{\tilde{M}_k}^*, \tilde{j}_{\tilde{M}_k}^*) > 0.$$

Then, the second term in (74) becomes

$$\begin{aligned} &\text{Prob}\,[L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), \tilde{i}_h^* \in A_h^{\max}, \tilde{j}_h^* \in A_h^{\min}, h = k, \cdots, k + \alpha_k - 1, \tilde{I}_k > |\mathcal{C}| + N | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\leq \text{Prob}\,[\tilde{\delta}_{\tilde{M}_k}(\tilde{i}_{\tilde{M}_k}^*, \tilde{j}_{\tilde{M}_k}^*) \leq 0, \delta_{\tilde{M}_k}(\tilde{i}_{\tilde{M}_k}^*, \tilde{j}_{\tilde{M}_k}^*) > 0 | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &= \sum_{M=k}^{k+\alpha_k-1} \text{Prob}\,[\tilde{\delta}_M(\tilde{i}_M^*, \tilde{j}_M^*) \leq 0, \delta_M(\tilde{i}_M^*, \tilde{j}_M^*) > 0 | \tilde{M}_k = M, (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\quad \times \text{Prob}\,[\tilde{M}_k = M | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})]. \end{aligned} \tag{77}$$

Using Lemma 4.1, we know that

$$\begin{aligned} &\text{Prob}\,[\tilde{\delta}_M(\tilde{i}_M^*, \tilde{j}_M^*) \leq 0, \delta_M(\tilde{i}_M^*, \tilde{j}_M^*) > 0 | \tilde{M}_k = M, (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\leq \sum_{\{(i,j)\in\tilde{\mathcal{C}}_M, \delta_M(i,j)>0\}} \text{Prob}\,[\tilde{\delta}_M(i,j) \leq 0 | \tilde{M}_k = M, (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \to 0 \\ &\qquad \text{as } k \to \infty. \end{aligned}$$

Therefore, we get from (77)

$$\lim_{k\to\infty} \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{h+1}) = L(\tilde{\boldsymbol{s}}_h), \tilde{i}_h^* \in A_h^{\max}, \tilde{j}_h^* \in A_h^{\min}, h = k, \cdots, k+\alpha_k - 1, \tilde{I}_k > |\mathcal{C}| + N | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] = 0.$$

The combination of this fact with (76) and (73) yields the conclusion of the lemma. ■

Proof of Lemma 4.5: First, given $(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})$ and some α_k defined as in (44), consider sample paths such that $L(\tilde{\boldsymbol{s}}_{i+1}) \le L(\tilde{\boldsymbol{s}}_i)$ for all $i = k, k+1, \cdots, k+\alpha_k - 1$. Observe that any such sample path can be decomposed into a set such that $L(\tilde{\boldsymbol{s}}_{i+1}) < L(\tilde{\boldsymbol{s}}_i)$ for some $k \le h \le k+\alpha_k - 1$ and a set such that $L(\tilde{\boldsymbol{s}}_{i+1}) = L(\tilde{\boldsymbol{s}}_i)$ for all $i = k, k+1, \cdots, k+\alpha_k - 1$. Thus, we can write

$$\begin{aligned}\{L(\tilde{\boldsymbol{s}}_{i+1}) &\le L(\tilde{\boldsymbol{s}}_i), i = k, \cdots, k+\alpha_k - 1\} \\ &= \{\exists k \le h \le k+\alpha_k - 1 \text{ s.t. } L(\tilde{\boldsymbol{s}}_{h+1}) < L(\tilde{\boldsymbol{s}}_h), \text{ and} \\ &\qquad L(\tilde{\boldsymbol{s}}_{i+1}) \le L(\tilde{\boldsymbol{s}}_i), i = k, \cdots, k+\alpha_k - 1, i \ne h\} \\ &\quad \cup \{L(\tilde{\boldsymbol{s}}_{i+1}) = L(\tilde{\boldsymbol{s}}_i), i = k, \cdots, k+\alpha_k - 1\}. \qquad (78)\end{aligned}$$

Therefore

$$\begin{aligned}\text{Prob}\,&[L(\tilde{\boldsymbol{s}}_{i+1}) \le L(\tilde{\boldsymbol{s}}_i), i = k, \cdots, k+\alpha_k - 1 \\ &\quad |(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &= \text{Prob}\,[\exists k \le h < k+\alpha_k \text{ s.t. } L(\tilde{\boldsymbol{s}}_{h+1}) < L(\tilde{\boldsymbol{s}}_h), \text{ and} \\ &\qquad L(\tilde{\boldsymbol{s}}_{i+1}) \le L(\tilde{\boldsymbol{s}}_i), i = k, \cdots, k+\alpha_k - 1, \\ &\qquad i \ne h | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\quad + \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{i+1}) = L(\tilde{\boldsymbol{s}}_i), i = k, \cdots, k+\alpha_k - 1 \\ &\qquad |(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\le \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{k+\alpha_k}) < L(\tilde{\boldsymbol{s}}_k) | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\quad + \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{i+1}) = L(\tilde{\boldsymbol{s}}_i), i = k, \cdots, k+\alpha_k - 1 \\ &\qquad |(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})]. \qquad (79)\end{aligned}$$

Using Lemma 4.4, the second term on the right-hand side above vanishes as $k \to \infty$, and (79) yields

$$\begin{aligned}\lim_{k\to\infty} \text{Prob}\,&[L(\tilde{\boldsymbol{s}}_{i+1}) \le L(\tilde{\boldsymbol{s}}_i), i = k, \cdots, k+\alpha_k - 1 \\ &\quad |(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})]. \\ &\le \lim_{k\to\infty} \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{k+\alpha_k}) < L(\tilde{\boldsymbol{s}}_k) | (\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})]. \qquad (80)\end{aligned}$$

On the other hand, we can write

$$\begin{aligned}\text{Prob}\,&[L(\tilde{\boldsymbol{s}}_{i+1}) \le L(\tilde{\boldsymbol{s}}_i), i = k, \cdots, k+\alpha_k - 1 \\ &\quad |(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &= \sum_{\left\{ \substack{(\boldsymbol{s}_i, \mathcal{C}_i), i = k+1, \cdots k+\alpha_k, \\ L(\boldsymbol{s}_i) \le L(\boldsymbol{s}_{i-1}), i = k+1, \cdots, k+\alpha_k} \right\}} \\ &\quad \cdot \text{Prob}\,[(\tilde{\boldsymbol{s}}_i, \tilde{\mathcal{C}}_i) = (\boldsymbol{s}_i, \mathcal{C}_i), i = k+1, \cdots, k+\alpha_k \\ &\qquad |(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})].\end{aligned}$$

The Markov property of $\{(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k)\}$ implies that

$$\begin{aligned}\text{Prob}\,&[(\tilde{\boldsymbol{s}}_i, \tilde{\mathcal{C}}_i) = (\boldsymbol{s}_i, \mathcal{C}_i), i = k+1, \cdots, k+\alpha_k \\ &\quad |(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &= \prod_{i=k+1}^{k+\alpha_k} \text{Prob}\,[(\tilde{\boldsymbol{s}}_i, \tilde{\mathcal{C}}_i) = (\boldsymbol{s}_i, \mathcal{C}_i) \\ &\qquad |(\tilde{\boldsymbol{s}}_{i-1}, \tilde{\mathcal{C}}_{i-1}) = (\boldsymbol{s}_{i-1}, \mathcal{C}_{i-1})].\end{aligned}$$

Thus

$$\begin{aligned}\text{Prob}\,&[L(\tilde{\boldsymbol{s}}_{i+1}) \le L(\tilde{\boldsymbol{s}}_i), i = k, \cdots, k+\alpha_k - 1 \\ &\quad |(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &= \sum_{\left\{ \substack{(\boldsymbol{s}_i, \mathcal{C}_i), i = k+1, \cdots, k+\alpha_k, \\ L(\boldsymbol{s}_i) \le L(\boldsymbol{s}_{i-1}), i = k+1, \cdots, k+\alpha_k} \right\}} \prod_{i=k+1}^{k+\alpha_k} \\ &\quad \cdot \text{Prob}\,[(\tilde{\boldsymbol{s}}_i, \tilde{\mathcal{C}}_i) = (\boldsymbol{s}_i, \mathcal{C}_i) \\ &\qquad |(\tilde{\boldsymbol{s}}_{i-1}, \tilde{\mathcal{C}}_{i-1}) = (\boldsymbol{s}_{i-1}, \mathcal{C}_{i-1})] \\ &= \sum_{\left\{ \substack{(\boldsymbol{s}_i, \mathcal{C}_i), i = k+1, \cdots, k+\alpha_k - 1, \\ L(\boldsymbol{s}_i) \le L(\boldsymbol{s}_{i-1}), i = k+1, \cdots, k+\alpha_k - 1} \right\}} \prod_{i=k+1}^{k+\alpha_k - 1} \\ &\quad \cdot \Bigg(\text{Prob}\,[(\tilde{\boldsymbol{s}}_i, \tilde{\mathcal{C}}_i) = (\boldsymbol{s}_i, \mathcal{C}_i) | (\tilde{\boldsymbol{s}}_{i-1}, \tilde{\mathcal{C}}_{i-1}) \\ &\qquad = (\boldsymbol{s}_{i-1}, \mathcal{C}_{i-1})] \\ &\qquad \cdot \sum_{\{(\boldsymbol{s}_j, \mathcal{C}_j), L(\boldsymbol{s}_j) \le L(\boldsymbol{s}_{j-1}), j = k+\alpha_k\}} \text{Prob}\,[(\tilde{\boldsymbol{s}}_j, \tilde{\mathcal{C}}_j) \\ &\qquad = (\boldsymbol{s}_j, \mathcal{C}_j) | (\tilde{\boldsymbol{s}}_{j-1}, \tilde{\mathcal{C}}_{j-1}) = (\boldsymbol{s}_{j-1}, \mathcal{C}_{j-1})] \Bigg) \\ &= \sum_{\left\{ \substack{(\boldsymbol{s}_i, \mathcal{C}_i), i = k+1, \cdots, k+\alpha_k - 1, \\ L(\boldsymbol{s}_i) \le L(\boldsymbol{s}_{i-1}), i = k+1, \cdots, k+\alpha_k - 1} \right\}} \prod_{i=k+1}^{k+\alpha_k - 1} \\ &\quad \cdot (\text{Prob}\,[(\tilde{\boldsymbol{s}}_i, \tilde{\mathcal{C}}_i) = (\boldsymbol{s}_i, \mathcal{C}_i) | (\tilde{\boldsymbol{s}}_{i-1}, \tilde{\mathcal{C}}_{i-1}) = (\boldsymbol{s}_{i-1}, \mathcal{C}_{i-1})] \\ &\quad \times \text{Prob}\,[L(\tilde{\boldsymbol{s}}_{k+\alpha_k}) \le L(\boldsymbol{s}_{k+\alpha_k - 1}) | (\tilde{\boldsymbol{s}}_{k+\alpha_k - 1}, \tilde{\mathcal{C}}_{k+\alpha_k - 1}) \\ &\qquad = (\boldsymbol{s}_{k+\alpha_k - 1}, \mathcal{C}_{k+\alpha_k - 1})]).\end{aligned}$$

Now, recalling the definition of $d_k(\boldsymbol{s}, \mathcal{C})$ in (33), observe that the last term in the product above is precisely $[1 - d_{k+\alpha_k - 1}(\tilde{\boldsymbol{s}}_{k+\alpha_k - 1}, \tilde{\mathcal{C}}_{k+\alpha_k - 1})]$. Moreover, by Lemma 4.2 we have $d_{k+\alpha_k - 1} \ge d_{k+\alpha_k - 1}(\tilde{\boldsymbol{s}}_{k+\alpha_k - 1}, \tilde{\mathcal{C}}_{k+\alpha_k - 1})$. Therefore, we get

$$\begin{aligned}\text{Prob}\,&[L(\tilde{\boldsymbol{s}}_{i+1}) \le L(\tilde{\boldsymbol{s}}_i), i = k, \cdots, k+\alpha_k - 1 \\ &\quad |(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k) = (\boldsymbol{s}, \mathcal{C})] \\ &\ge (1 - d_{k+\alpha_k - 1}) \sum_{\left\{ \substack{(\boldsymbol{s}_i, \mathcal{C}_i), i = k+1, \cdots, k+\alpha_k - 1, \\ L(\boldsymbol{s}_i) \le L(\boldsymbol{s}_{i-1}), i = k+1, \cdots, k+\alpha_k - 1} \right\}} \\ &\qquad \cdot \prod_{i=k+1}^{k+\alpha_k - 1} \text{Prob}\,[(\tilde{\boldsymbol{s}}_i, \tilde{\mathcal{C}}_i) = (\boldsymbol{s}_i, \mathcal{C}_i) \\ &\qquad\qquad |(\tilde{\boldsymbol{s}}_{i-1}, \tilde{\mathcal{C}}_{i-1}) = (\tilde{\boldsymbol{s}}_{i-1}, \mathcal{C}_{i-1})] \\ &\ge \cdots \ge \prod_{i=k}^{k+\alpha_k - 1} (1 - d_i) \\ &\ge (1 - d_k)^{\alpha_k} \qquad (81)\end{aligned}$$

where the last inequality follows from Lemma 4.2, where it was shown that d_i is monotone decreasing in i. Hence, since α_k satisfies (45), we get

$$\begin{aligned}\lim_{k\to\infty} \text{Prob}\,&[L(\tilde{\boldsymbol{s}}_{i+1}) \le L(\tilde{\boldsymbol{s}}_i), i = k, \cdots, k+\alpha_k - 1 \\ &\quad |(\tilde{\boldsymbol{s}}_k, \tilde{\mathcal{C}}_k)] = 1.\end{aligned}$$

Finally, using this limit, (80), and recalling the definition of $e_k(\boldsymbol{s}, \mathcal{C})$ in (46), we readily conclude that (48) holds.

Moreover, (49) immediately implies that e_k is monotone decreasing and $e_k \geq e_k(\boldsymbol{s}, \mathcal{C})$. The limit (50) then follows from (48). ∎

REFERENCES

[1] E. Aarts and J. Korst, *Simulated Annealing and Boltzmann Machines*. New York: Wiley, 1989.

[2] C. G. Cassandras, *Discrete Event Systems, Modeling, and Performance Analysis.* Homewood, IL: Irwin, 1993.

[3] C. G. Cassandras and V. Julka, "A new approach for some combinatorially hard stochastic optimization problems," in *Proc. 31st Ann. Allerton Conf. Commun., Control, and Computing*, 1993.

[4] ——, "Descent algorithms for discrete resource allocation problems," in *Proc. 33rd Conf. Decision and Control*, 1994, pp. 2639–2644.

[5] ——, "Scheduling policies using marked/phantom slot algorithms," *Queueing Systems: Theory and Appl.*, vol. 20, pp. 207–254, 1995.

[6] C. G. Cassandras and C. G. Panayiotou, "Concurrent sample path analysis of discrete event systems," in *Proc. 35th IEEE Conf. Decision and Control*, 1996, pp. 3332–3337.

[7] L. Dai, "Convergence properties of ordinal comparison in the simulation of discrete event dynamic systems," *J. Optimization Theory and Appl.*, vol. 91, pp. 363–388, 1996.

[8] R. G. Gallager, "A minimum delay routing algorithm using distributed computation," *IEEE Trans. Commun.*, vol. COM-25, pp. 73–85, 1977.

[9] W. B. Gong, Y. C. Ho, and W. Zhai, "Stochastic comparison algorithm for discrete optimization with estimation," in *Proc. 31st IEEE Conf. Decision and Control*, 1992.

[10] Y. C. Ho, M. A. Eyler, and T. T. Chien, "A gradient technique for general buffer storage design in production line," *Int. J. Production Res.*, vol. 17, no. 6, pp. 557–580, 1979.

[11] Y. C. Ho, R. S. Sreenivas, and P. Vakili, "Ordinal optimization in DEDS," *J. Discrete Event Dynamic Systems: Theory and Appl.*, vol. 2, pp. 61–88, 1992.

[12] Y. C. Ho, "Heuristics, rule of thumb, and the 80/20 proposition," *IEEE Trans. Automat. Contr.*, vol. 39, pp. 1025–1027, 1994.

[13] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Univ. Michigan Press, 1975.

[14] T. Ibaraki and N. Katoh, *Resource Allocation Problems: Algorithmic Approaches*. Cambridge, MA: MIT Press, 1988.

[15] J. Kiefer and J. Wolfowitz, "Stochastic estimation of the maximum of a regression function," *Ann. Math. Statistics*, vol. 23, pp. 462–466, 1952.

[16] R. G. Parker and R. L. Rardin, *Discrete Optimization*. Boston: Academic, 1988.

[17] H. Robbins and S. Monro, "A stochastic approximation method," *Ann. Math. Statistics*, vol. 22, pp. 400–407, 1951.

[18] J. E. Wieselthier, C. M. Barnhart, and A. Ephremides, "Optimal admission control in circuit-switched multihop radio networks," in *Proc. 31st IEEE Conf. Decision and Control*, 1992, pp. 1011–1013.

[19] D. Yan and H. Mukai, "Stochastic discrete optimization," *SIAM J. Control Optim.*, vol. 30, 1992.

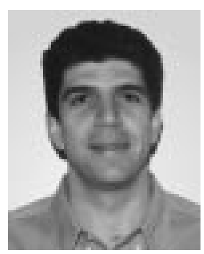
[20] H. Yan, X. Y. Zhou, and G. Yin, "Finding optimal number of Kanbans in a manufacturing system via stochastic approximation and perturbation analysis," in *Proc. 11th Int. Conf. Analysis and Optimization of Systems*, 1994, pp. 572–578.

Christos G. Cassandras (S'82–M'82–SM'91–F'96), for a photograph and biography, see p. 655 of the May 1998 issue of this TRANSACTIONS.

Liyi Dai (M'90) received the B.S. degree from Shandong University, Shandong, China, in 1983, the M.S. degree from the Institute of Systems Science, Academia Sinica, Beijing, China, in 1986, and the Ph.D. degree from Harvard University, Cambridge, MA, in 1993.

From 1986 to 1988, he was with the Institute of Systems Science, Academia Sinica. Since 1993, he has been an Assistant Professor in the Department of Systems Science and Mathematics at Washington University, St. Louis, MO. His research interests include discrete-event dynamic systems, simulation, stochastic optimization, communication systems, and singular systems. He has published over 35 papers in various journals and is the author of *Singular Control Systems* (Berlin: Springer-Verlag, 1989).

Dr. Dai is listed in *Who's Who Among Asian Americans* and is a recipient of the NSF CAREER award. He is an Associate Editor of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL.



Christos G. Panayiotou received the B.S. degree in electrical engineering from the University of Massachusetts, Amherst, in 1994. He is now a Ph.D. candidate in electrical engineering and is pursuing an MBA degree at the Isenberg School of Management, both at the aforementioned university.

His research interests include optimization and control of discrete-event systems, computer simulation, communication networks, and manufacturing systems.

Mr. Panayiotou is a member of Tau Beta Pi.