A BALANCE PRINCIPLE FOR OPTIMAL ACCESS CONTROL[1]

Richard S. Brooks

COINS Technical Report 80-20

revised, December 1981[2]

Computer and Information Science
University of Massachusetts
Amherst, MA 01003

Abstract

A problem of optimal, distributed multi-access control of shared resources in a network of processing elements is addressed in this paper. Pareto optimality is used as a notion of optimal, decentralized resource sharing, and necessary conditions for optimal access control policies are derived. A simple yet powerful balance principle results from an interpretation of the optimality conditions. The balance principle is quite general and covers a number of known rules for multi-access control of a shared communication channel in a packet radio network. An adaptive algorithm for achieving and maintaining balance is also developed, and simulation results are presented.

---

1. This research was supported in part by National Science Foundatión Grant MCS78-04212 to the University of Massachusetts.

2. The original technical report of October 1980 has been substantially revised to include more discussion of the balance principle, figures on the theoretical performance of two access control schemes that use the balance principle, and details (including simulation results) of an adaptive balancing algorithm.

Table of Contents

Acknowledgments

I would like to thank Yechiam Yemini for many fruitful discussions concerning my extension of his balance principle, and Andy Barto for listening to my ideas and providing excellent feedback. I would also like to thank Michael Arbib, Scott Reed, Jasmina Pavlin, and Victor Lesser. for their helpful comments on various drafts of this paper.

1. Introduction

In this paper a problem of optimal, distributed multi-access control of shared resources in a network of processing elements (PEs) is addressed. A necessary condition for (Pareto) optimality is applied to the access control problem and an interpretation results in a general balance principle for optimal access control. The balance principle is quite general and is shown to hold for a wide range of utilization processes. Direct use of the balance principle is discussed, and theoretical performance figures are presented, comparing two schemes for which the balance principle was used to obtain Pareto-optimal policies and two standard schemes that do not use the balance principle. An adaptive "balancing algorithm" for achieving and maintaining optimal control in a quasi-static environment is also developed and simulation results are presented.

It is assumed that each PE has access to a subset of all resources, and demand for resources arises in a random manner.[3] Use of resources is assumed to be time-slotted (i.e., use is sunchronized to slots in time) and a PE requires a single time-slot of all accessible resources when it has demand.[4] When two or more PEs attempt to access a shared

---

3. In an environment where demand for resources is regular, a pre-determined access control scheme often works well; when demand is random, however, a scheme which is demand adaptive is often necessary to avoid a waste of resources.

4. This "greedy" assumption simplifies the analysis, but a probability distribution may be used to specify probabilistically which resources accessible to a PE will be needed when demand arises.

resource simultaneously, a collision occurs, and for most applications the shared resource cannot be properly utilized. Because of this, an access control scheme that decides which PEs with need of resources should have the right to access the resources is desirable. The objective of an access control scheme is to maximize the expected utilization of resources. Note that the utilization of resources when a collision occurs is taken to be zero for many applications, but this need not be the case.

The research presented in this paper is based on recent contributions to the field of Communication and Networking by Yechiam Yemini and Leonard Kleinrock [5],[10],[11] concerning a balance principle and adaptive algorithms for multi-access control of a shared communication channel in a packet radio network. This research extends their work to cover a broad range of objective functions which, in addition to specifying the utilization of a success (exclusive access to shared resources), may specify a penalty for collisions and/or sacrifices (decisions to not attempt access though in need of resources). A more general balance principle results because an alternative interpretation of the necessary conditions is found to be appropriate. A new adaptive balancing algorithm is developed and tested for the new balance principle.[5]

---

5. The new balance principle and adaptive balancing algorithm were developed to control simultaneous updating of control variables in a distributed iterative refinement algorithm. This application is developed in a forthcoming thesis [2].

2. Objective Functions

The major difficulty in determining an optimal control scheme in a distributed manner is the lack of a global view of the network. For this reason, the optimization will be based on a simple relationship among local, decentralized objective functions rather than on a global objective function. A network utilization operator mathematically expresses the expected utilization of resources by each PE; it is actually a vector of local utilization operators, one for each PE. These local utilization operators are used as local objective functions.

A number of utilization operators are possible depending, for instance, on which accessible resources will be needed when demand arises and how possible outcomes (success, collision, sacrifice, etc.) are combined. Different applications will call for different operators. While the balance principle will be developed below using only one utilization operator, it will later be shown to apply to a broad class of utilization operators.

Formally, let $\underline{\tilde{d}}^t = (\tilde{d}_1^t, \tilde{d}_2^t, \ldots, \tilde{d}_N^t)$ designate the demand process for time-slot t.[6] That is,

$$\tilde{d}_i^t = \begin{cases} 1 \text{ if } PE_i \text{ requires resources at time-slot } t, \\ 0 \text{ otherwise.} \end{cases}$$

There is no distinction between resources in this model because it is

---

6. A random variable and its mean are represented by $\tilde{x}$ and $\bar{x}$, respectively. Vectors are underlined and sets are capitalized.

assumed that when a processing element attempts to access resources, it attempts to access all resources to which it has access. Let $\pi^t(\underline{d})$ designate the distribution of $\underline{\tilde{d}}^t$. This distribution is not known, and is assumed to change slowly with time.

Define the resource utilization process of PE_i during time-slot t as

$$\tilde{u}_i^t = \begin{cases} 1 & \text{if } PE_i \text{ has demand and gains exclusive access to all shared resources,} \\ -a & \text{if } PE_i \text{ has demand and one or more PEs interfere with } PE_i, \\ 0 & \text{if } PE_i \text{ has demand but does not attempt to access shared resources, or if } PE_i \text{ has no demand.} \end{cases}$$

Thus, a PE is awarded one unit of utilization for a success and is penalized a units for a collision; a sacrifice or being idle (having no demand) results in no utilization.[7] For the packet broadcast problem examined by Yemini and others [1],[4] a penalty was not used (i.e., a=0) since a packet whose transmission is interfered with is not lost because a copy is always held for possible retransmission.

Consider a <u>probabilistic access control scheme</u>[8] where a utilization policy, $\underline{p} = (p_1, p_2, \ldots, p_N)$, is a vector of probabilities, such that for each time-slot PE_i will attempt to access shared resources with probability p_i if it has demand. Assuming that a PE attempts to access

---

7. A single penalty for a collision with any number of PEs and no penalty for a sacrifice is used here, but many other utilization processes are possible.

8. Other control schemes may be used provided the probability a PE will attempt utilization can be expressed as a function of the policy; another control scheme, the urn scheme, is described in Section 4.

all resources to which it has access when demand arises the mean utilization of PE_i when policy $\underline{p}$ is used, conditioned upon $\underline{d}_i^t = \underline{d}$ is

$$\bar{u}_i^t(\underline{d},\underline{p}) = \begin{cases} p_i \prod_{j \in I_i(\underline{d})} (1-p_j) - ap_i\left(1 - \prod_{j \in I_i(\underline{d})} (1-p_j)\right) & \text{if } d_i=1 \\ 0 & \text{if } d_i=0, \end{cases}$$

where $I_i(\underline{d}) = \{ j \mid (j \neq i) \wedge (d_j=1) \wedge (j \in I(i)) \}$,

and $I(i) = \{ j \mid PE_j$ potentially interferes with $PE_i \}$.

Clearly, PE_i's expected utilization given demand $\underline{d}$ and policy $\underline{p}$ is zero if $d_i=0$ (i.e., PE_i has no demand). If $d_i=1$, however, PE_i's expected utilization is 1 times the probability that it attempts and no PEs interfere minus a times the probability that it attempts and some PE interferes.

Assuming a uniform distribution of demand, the expected utilization of PE_i during time-slot t with policy $\underline{p}$ is given by the local utilization operator,

$$\bar{u}_i^t(\underline{p}) = \sum_{\underline{d} \in \{0,1\}^N} \pi^t(\underline{d})\ \bar{u}_i^t(\underline{d},\underline{p}).$$

To simplify notation, the time index, t, will be eliminated in expressions where there should be no confusion. The network utilization operator can now be described by

$$\underline{U}(\underline{p}) = \begin{bmatrix} \bar{u}_1(\underline{p}) \\ \bar{u}_2(\underline{p}) \\ \cdot \\ \cdot \\ \cdot \\ \bar{u}_N(\underline{p}) \end{bmatrix} .$$

Given a policy, $\underline{p}$, the network utilization operator yields the expected utilization of PEs.

3. Pareto Optimality and a Necessary Condition

Pareto optimality, a concept of mathematical economics and game theory [6],[7],[8], is an attractive choice of a decentralized optimality criterion because it is a weak form of optimality that requires minimal coordination of members of a decentralized community. At a Pareto-optimum it is impossible to increase the utilization of any PE by changing PEs' policies without decreasing the utilization of some other PE(s); thus, PEs are not selfish. A Pareto-optimal policy yields a Pareto-optimal utilization. Formally, a utilization vector, $\underline{u}$, is Pareto-optimal if and only if

1) it is attainable, i.e., $\underline{u}^* = \underline{U}(\underline{p}^*)$ for feasible policy $\underline{p}^*$, and
2) it is not dominated by another attainable utilization vector (i.e., $\neg\exists \underline{u}' \mid \forall i \; u_i' \geq u_i^*$, with at least one strict inequality).

The concept of Pareto optimality has a simple geometric interpretation. To see this, consider the case of two PEs that both have demand for a single, shared resource. If the penalty for a collision is one (i.e., a=1), the utilization operator is

$$\underline{U}(\underline{p}) = \begin{cases} \bar{u}_1(\underline{p}) = p_1(1-p_2) - p_1p_2 \\ \bar{u}_2(\underline{p}) = p_2(1-p_1) - p_2p_1 \end{cases}$$

and Figure 1 shows how the space of feasible policies is mapped onto attainable utilizations.

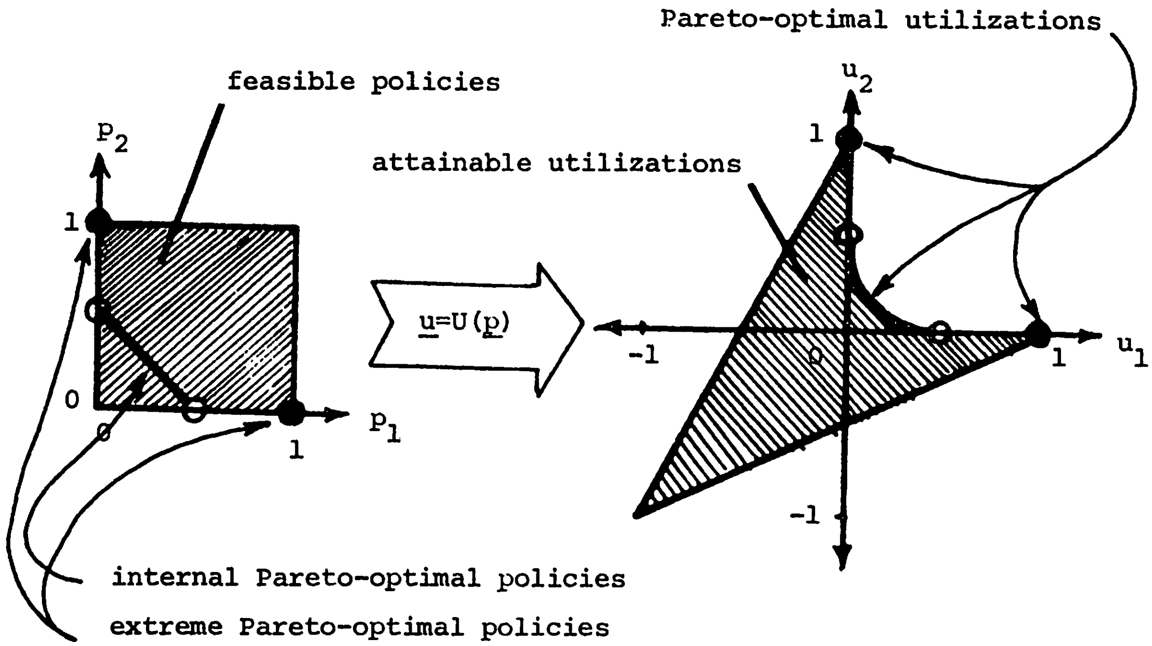


Figure 1: Utilization Operator

In this example, feasible policies are probability vectors for deciding whether to access shared resources when there is demand, so all components must lie between 0.0 and 1.0, inclusive. Each feasible policy vector has a corresponding utilization vector determined by the network utilization operator; these utilizations constitute the attainable utilizations. Pareto optimal utilizations are all utilizations on the upper-right boundary of the region of attainable utilizations because such utilizations are not dominated by any other attainable utilizations (i.e., there is no other utilization that gives more to one PE without taking utilization away from another PE).

The set of feasible policies includes internal policies and extreme policies. A policy, $\underline{p}'$, is an extreme policy if $p_i'=0$ or $p_i'=1$ for some i. The necessary condition for Pareto-optimality to be derived characterizes internal Pareto-optimal policies and does not necessarily hold for extreme policies. This is so because the extremality of $\underline{U}(\underline{p}')$ may be directly caused by the extremality of $\underline{p}'$. Extreme Pareto optimal policies must be found by other means.

Consider an internal Pareto-optimal policy, $\underline{p}^*$, and let $\underline{u}^*=\underline{U}(\underline{p}^*)$ be the resulting Pareto-optimal utilization. Provided $\underline{U}(\underline{p})$ is continuous, a small perturbation of $\underline{p}^*$ leads to a small perturbation of $\underline{u}^*$. The utilization of these perturbed policies are related to $\underline{u}^*$ by the following linear approximation:

$$\underline{U}(\underline{p}^* + \Delta\underline{p}) = \underline{u}^* + \Delta\underline{p}\,\frac{\partial\underline{U}(\underline{p})}{\partial\underline{p}}\bigg|_{\underline{p}=\underline{p}^*}$$

where $\frac{\partial\underline{U}(\underline{p})}{\partial\underline{p}}\Big|_{\underline{p}=\underline{p}^*}$ is the Jacobian matrix of $\underline{U}(\underline{p})$ at $\underline{p}^*$.

Because $\underline{p}^*$ is an internal point of the set of feasible policies, it admits perturbations in all directions. The extremality of $\underline{u}^*$ implies that the attainable perturbations of $\underline{u}^*$ must not admit perturbations in all directions. This condition occurs if the Jacobian matrix at $\underline{p}^*$ is singular, because (according to the linear approximation) when this is the case there is no perturbation of $\underline{p}^*$ that perturbs $\underline{u}^*$ in the direction perpendicular to the boundary surface of attainable utilizations. Note that this condition is necessary but not sufficient for determining these extrema.

The singularity of the Jacobian at $\underline{p}^*$ implies that there is a non-zero linear combination of rows of the Jacobian at $\underline{p}^*$ which yields a zero vector. If $\underline{c} = (c_1, c_2, \ldots, c_N)$ are the non-zero coefficients of such a linear combination, the necessary condition for Pareto optimality of an internal policy can be stated as

$$\forall i \quad \sum_{j=1\ldots N} c_j \frac{\partial \bar{u}_j(\underline{p})}{\partial p_i} = 0 \; .$$

4. The Balance Principle

For the utilization operator defined in Section 1, the elements of the Jacobian at $\underline{p}$ are

$$\frac{\partial \bar{u}_i(\underline{p})}{\partial p_j} = \begin{cases} \phi_i(\underline{p}) & \text{if } i=j \\ \psi_{ij}(\underline{p}) & \text{if } i\neq j, \end{cases}$$

where
$$\phi_i(\underline{p}) = \sum_{\underline{d}\in\{0,1\}^N} \pi(\underline{d}) \frac{\partial \bar{u}_i(\underline{d},\underline{p})}{\partial p_i},$$

$$\psi_{ij}(\underline{p}) = \sum_{\underline{d}\in\{0,1\}^N} \pi(\underline{d}) \frac{\partial \bar{u}_i(\underline{d},\underline{p})}{\partial p_j},$$

$$\frac{\partial \bar{u}_i(\underline{d},\underline{p})}{\partial p_i} = \begin{cases} \prod_{k\in I_i(\underline{d})} (1-p_k) - a\big(1-\prod_{k\in I_i(\underline{d})} (1-p_k)\big) & \text{if } d_i=1 \\ 0 & \text{if } d_i=0, \end{cases}$$

$$\frac{\partial \bar{u}_i(\underline{d},\underline{p})}{\partial p_j} = \begin{cases} -p_i \prod_{k\in I_i(\underline{d})\wedge k\neq j} (1-p_k) - ap_i \prod_{k\in I_i(\underline{d})\wedge k\neq j} (1-p_k) & \text{if } d_i=1\wedge j\in I_i(\underline{d}) \\ 0 & \text{otherwise.} \end{cases}$$

Obviously these expressions are the marginal expected utilizations for PE_i given an incremental change in p_i or p_j; but, these expressions also have another meaning. To see the alternative interpretations for $\phi_i(\underline{p})$ and $\psi_{ij}(\underline{p})$ it is necessary to look at the expressions by themselves (i.e., not as derivatives).

If PE_i has demand, $\phi_i(\underline{p})$ is 1 times the probability that no PEs with demand that may interfere with PE_i attempt to access shared resources minus a times the probability that some PE with demand that may interfere with PE_i does attempt (and hence collides with PE_i). If PE_i does not have demand, $\phi_i(\underline{p})$ is 0. Thus, $\phi_i(\underline{p})$ can be interpreted as PE_i's expected utilization with policy $\underline{p}$ given that it attempts to access shared resources if it has demand. Note that if a=0, $\phi_i(\underline{p})$ may alternatively be interpreted (as Yemini did) as the probability that shared resources to which PE_i has access will not be utilized given that PE_i does not attempt to access them if it has demand.

To interpret $\psi_{ij}(\underline{p})$, note that

$$\psi_{ij}(\underline{p}) = \begin{cases} -X_{ij}(\underline{p}) + Z_{ij}(\underline{p}) & \text{if } j \in I(i) \\ 0 & \text{otherwise,} \end{cases}$$

$$X_{ij}(\underline{p}) = \sum_{\underline{d} \in \{0,1\}^N} \pi(\underline{d})\, \bar{x}_{ij}(\underline{d},\underline{p}),$$

$$\text{where } \; Z_{ij}(\underline{p}) = \sum_{\underline{d} \in \{0,1\}^N} \pi(\underline{d})\, \bar{z}_{ij}(\underline{d},\underline{p}),$$

$$\bar{x}_{ij}(\underline{d},\underline{p}) = \begin{cases} p_i \prod_{j \in I_i(\underline{d}) \wedge k \neq j} (1-p_k) - ap_i(1 - \prod_{j \in I_i(\underline{d}) \wedge k \neq j} (1-p_k)) & \text{if } d_i = d_j = 1 \\ p_i \prod_{k \in I_i(\underline{d})} (1-p_k) - ap_i(1 - \prod_{k \in I_i(\underline{d})} (1-p_k)) & \text{if } d_i = 1 \wedge d_j = 0 \\ 0 & \text{if } d_i = 0. \end{cases}$$

$$\bar{z}_{ij}(\underline{d},\underline{p}) = \begin{cases} -ap_i \prod_{j \in I_i(\underline{d}) \wedge k \neq j} (1-p_k) - ap_i(1 - \prod_{j \in I_i(\underline{d}) \wedge k \neq j} (1-p_k)) & \text{if } d_i = d_j = 1 \\ p_i \prod_{k \in I_i(\underline{d})} (1-p_k) - ap_i(1 - \prod_{k \in I_i(\underline{d})} (1-p_k)) & \text{if } d_i = 1 \wedge d_j = 0 \\ 0 & \text{if } d_i = 0, \end{cases}$$

through the addition of some dummy terms. For $j \in I(i)$, $X_{ij}(\underline{p})$ can be interpreted as PE_i's expected utilization with policy $\underline{p}$ given PE_j does not attempt to access shared resources if it has demand. And, for $j \in I(i)$, $Z_{ij}(\underline{p})$ can be interpreted as PE_i's expected utilization with policy $\underline{p}$ given PE_j does attempt to access shared resources if it has demand. Note that $\bar{z}_{ij}(\underline{d},\underline{p}) = -ap_i$ after a little algebraic manipulation for $d_i = d_j = 1$ as it should since PE_i attempts to access shared resources with probability p_i and will suffer a collision for sure if it attempts (because it is given that PE_j will attempt to access shared resources and PE_j interferes with PE_i).

The Jacobian of the utilization operator at $\underline{p}$ can now be stated as

$$\frac{\partial \underline{U}(\underline{p})}{\partial \underline{p}} = \begin{bmatrix} \phi_1(\underline{p}) & \psi_{12}(\underline{p}) & \psi_{13}(\underline{p}) \cdots & \psi_{1N}(\underline{p}) \\ \psi_{21}(\underline{p}) & \phi_2(\underline{p}) & \psi_{23}(\underline{p}) \cdots & \psi_{2N}(\underline{p}) \\ \psi_{31}(\underline{p}) & \psi_{32}(\underline{p}) & \phi_3(\underline{p}) \cdots & \psi_{3N}(\underline{p}) \\ \vdots & \vdots & \vdots & \vdots \\ \psi_{N1}(\underline{p}) & \psi_{N2}(\underline{p}) & \psi_{N3}(\underline{p}) \cdots & \phi_N(\underline{p}) \end{bmatrix} .$$

And, the necessary condition for Pareto optimality of an internal policy (that there exist a non-zero linear combination of rows of the Jacobian at $\underline{p}$ that yields a zero vector) can be stated as

$$\exists \underline{c} \mid \underline{c} \neq (0,0,\ldots 0) \wedge \forall i=1,2,\ldots,N$$

$$c_i \phi_i(\underline{p}) + \sum_{j \in I(i)} c_j Z_{ji}(\underline{p}) = \sum_{j \in I(i)} c_j X_{ji}(\underline{p}) .$$

For convenience, a PE which may interfere with PE_i will be called a <u>neighbor</u> of PE_i, and a PE and its neighbors form a <u>neighborhood</u>. Noticing that $\underline{c}$ can be interpreted as a vector of pay-off coefficients similar to a vector of Lagrangian multipliers, and discounting the case where PE_i has no demand (since then $\phi_i(\underline{p})=0$ and $Z_{ji}(\underline{p})=X_{ji}(\underline{p})$), the following interpretation, the <u>balance principle</u>, is possible:

> At a Pareto-optimal policy there exists a non-zero vector of payoff coefficients such that for each PE, expected neighborhood utilization payoff given the PE has demand and attempts to access shared resources equals expected neighborhood utilization payoff given the PE has demand but does not attempt to access shared resources.

This balance principle differs from Yemini's which, for each PE, equates expected silence (empty slots) given the PE has demand with expected utilization by the PE's neighbors given the PE has demand. In the section following the next section on direct use of the balance principle, the balance principle developed above is shown to hold for a wide range of utilization operators which may include a penalty for collisions and/or sacrifices. Yemini's balance principle is less general as it does not hold for such a wide range of utilization operators.

5. Direct Use of the Balance Principle

A direct way of using the balance principle just derived involves manipulating the balance principle equations into expressions for optimal policies given the demand, payoff coefficients, and penalty (if applicable). These expressions are then used with estimates of the current demand, current payoff coefficients, and estimate of the penalty (if applicable) to solve for optimal policies.

To understand how this works, consider the case of two PEs that share a single resource with penalty α for collisions. Assuming that the probability that each PE has demand for the shared resource is estimated to be $\underline{d} = (d_1, d_2)$, the balance principle states that

$$\text{for } PE_1: \quad c_1((1-d_2p_2)-\alpha d_2p_2) + c_2(-\alpha d_2p_2) = c_2d_2p_2,$$

$$\text{and for } PE_2: \quad c_2((1-d_1p_1)-\alpha d_1p_1) + c_1(-\alpha d_1p_1) = c_1d_1p_1.$$

Here $c_1((1-d_2p_2)-\alpha d_2p_2) + c_2(-\alpha d_2p_2)$ is PE_1's expected neighborhood utilization payoff given it has demand and attempts to access the resource because $c_1((1-d_2p_2)-\alpha d_2p_2)$ is PE_1's expected utilization payoff given that it has demand and attempts to access the resources and $c_2(-\alpha d_2p_2)$ is PE_2's expected utilization payoff given that PE_1 has demand and attempts to access the resource. Also, $c_2d_2p_2$ is PE_1's expected neighborhood utilization payoff given it has demand yet does not attempt to access the resource because PE_1's expected utilization payoff given that it has demand yet does not access the resource is zero and PE_2's expected utilization payoff given that PE_1 has demand yet does not access the resource is $c_2d_2p_2$. The situation for PE_2 is similar.

After some manipulation, the expressions for optimal policies are

$$p_1 = c_1/(c_1+c_2)d_1(1+\alpha)$$

$$\text{and } p_2 = c_2/(c_1+c_2)d_2(1+\alpha).$$

Thus, if $d_1=d_2=1$ (i.e., PE_1 and PE_2 are both known or believed to be busy) and $\alpha=1$ (i.e., the penalty for collisions is known or believed to be one) then $c_1/c_2=1$ selects the Pareto-optimal policy $p_1=p_2=1/4$. One can, in fact, prove that when $d_1=d_2=1$ and $\alpha=1$ (as is the case shown in Figure 1) all internal Pareto-optimal policies lie on the line $p_2 = -p_1 + 1/2$. To obtain the expected utilization payoffs for PEs one merely plugs the optimal policies into the network utilization operator.

It should be noted that the vector of payoff coefficients, $\underline{c}$, which may be used to select a particular Pareto-optimal solution does not directly set the relative value of utilization for PEs. For example, if $c_1/c_2=2$ for the example just described and depicted in Figure 1, the policy $p_1=1/6, p_2=1/3$ is selected, for which $u_1/u_2=1/4$. At an internal Pareto-optimal policy the ratio of the coefficients that satisfies the balance principle defines the orientation of the utilization boundary surface, not the ratio of expected utilizations. By fixing the coeffients and then solving for a Pareto-optimal policy, a particular Pareto-optimal policy can be chosen; however, care must be taken to choose a vector of coefficients for which a solution exists.

6. Proof of Generality

It will now be shown that the general balance principle developed in the previous section applies for a broad range of utilization operators. Let $E[\tilde{u}_j|\underline{p}\wedge\text{attempt}(i)]$ be PE_j's expected utilization with policy $\underline{p}$ given that PE_i attempts to access shared resources. Similarly, let $E[\tilde{u}_j|\underline{p}\wedge\text{sacrifice}(i)]$ and $E[\tilde{u}_j|\underline{p}\wedge\text{idle}(i)]$ be PE_j's expected utilization with policy $\underline{p}$ given that PE_i has demand but does not attempt to access shared resources and PE_i has no demand, respectively.

From Section 2, the local utilization operator for PE_j is

$$\bar{u}_j(\underline{p}) = \sum_{\underline{d}\in\{0,1\}^N} \pi(\underline{d})\ \bar{u}_j(\underline{d},\underline{p}).$$

Now, though,

$$\bar{u}_j(\underline{d},\underline{p}) = \begin{cases} p_i E[\tilde{u}_j|\underline{p}\wedge\text{attempt}(i)] + (1-p_i)E[\tilde{u}_j|\underline{p}\wedge\text{sacrifice}(i)] & \text{if } d_i=1 \\ E[\tilde{u}_j|\underline{p}\wedge\text{idle}(i)] & \text{if } d_i=0, \end{cases}$$

and, provided $E[\tilde{u}_j|\underline{p}\wedge\text{attempt}(i)]$, $E[\tilde{u}_j|\underline{p}\wedge\text{sacrifice}(i)]$, and $E[\tilde{u}_j|\underline{p}\wedge\text{idle}(i)]$ are independent of p_i for all j (including j=i)

$$\frac{\partial\bar{u}_j(\underline{p})}{\partial p_i} = \sum_{\underline{d}\in\{0,1\}^N} \pi(\underline{d})\ \frac{\partial\bar{u}_j(\underline{p},\underline{d})}{\partial p_i},$$

where

$$\frac{\partial\bar{u}_j(\underline{d},\underline{p})}{\partial p_i} = \begin{cases} E[\tilde{u}_j|\underline{p}\wedge\text{attempt}(i)] - E[\tilde{u}_j|\underline{p}\wedge\text{sacrifice}(i)] & \text{if } d_i=1 \\ 0 & \text{if } d_i=0. \end{cases}$$

Using the necessary conditions for Pareto optimality and discounting the case where $d_i=0$ (since 0=0), the general balance principle results:

$$\exists \underline{c} \mid \forall i=1,2,\ldots N$$

$$c_i E[\tilde{u}_i \mid \underline{p}\text{^attempt}(i)] + \sum_{j \in I(i)} c_j E[\tilde{u}_j \mid \underline{p}\text{^attempt}(i)] =$$

$$c_i E[\tilde{u}_i \mid \underline{p}\text{^sacrifice}(i)] + \sum_{j \in I(i)} c_j E[\tilde{u}_j \mid \underline{p}\text{^sacrifice}(i)].$$

This balance principle is valid for a wide range of utilization processes and operators; however, for some applications a special case of the balance principle is appropriate. This case arises when a PE's utilization of one shared resource is not dependent on the PE gaining exclusive access to other shared resources. When this condition holds, a PE's neighbors' utilization of resources not shared with the PE is independent of the PE's actions, and neighbors' conditional expected utilization payoff <u>of resources shared with the PE</u> may be used in place of neighbors' conditional expected utilization payoff of all shared resources.

7. The Urn Scheme and Some Performance Measures

An interesting alternative to the probabilistic decision mechanism described above is the urn scheme [5],[10],[12]. Under the urn scheme, each PE_i draws k_i numbers from identical pseudo-random number generators using a common seed to determine which PEs may attempt utilization of shared resources. The probability of drawing a particular number from a pseudo-random number generator in a sample of k numbers is similar to the probability of drawing a ball of a particular color from an urn containing various colored balls in a sample of k balls; hence, the name "urn" scheme. The use of a common seed and identical pseudo-random number generators coordinates the decisions of PEs without communication.

Actually, a variety of urn schemes are possible, depending upon how numbers (colors) are assigned to PEs and what numbers can be drawn from a pseudo-random number generator (how many balls of each color are in an urn). Perhaps the simplest urn scheme assigns a unique number to each PE, and has PEs draw numbers from identical, randomly ordered lists of the assigned numbers. A better urn scheme, though, assigns the fewest numbers to PEs such that each PE is assigned a number yet in each neighborhood no PE has the same number, and has PEs draw numbers from identical, randomly ordered lists of the (fewer) assigned numbers. Note that when each PE may interfere with all PEs, the schemes are equivalent. The use of multiple urns has been proposed [12] which produces a scheme similar to the tree scheme [3].

The techniques used to derive the balance principle for the probabilistic scheme may also be used to determine an optimal k for an urn scheme. The urn scheme, as Kleinrock and Yemini showed in [5] and will be shown below, achieves a utilization similar to the probabilistic scheme when demand is light, and significantly better utilization when demand is heavy.

Figures 2 and 3 show the theoretical performance of the probabilistic and urn schemes, and compare their performance with a serial scheme (which gives access rights to one PE each iteration in a fixed order) and a perfect scheme (which gives access rights to one PE with demand each iteration if there is such a PE). These figures depict the case where a single resource is shared by a number of PEs, and each PE has demand each time-slot, independent of other PEs and other time-slots. Expected global utilization vs. demand for two, four, and eight identical PEs at a Pareto optimum assuming no penalty for collisions is shown in Figure 2. Figure 3 shows expected global utilization vs. demand for two identical PEs at a Pareto optimum assuming a range of penalties for collisions.

The effectiveness of the probabilistic scheme is clearly limited with moderate and high demand. The urn scheme performs better than the probabilistic scheme, but requires a common seed for the identical pseudo-random number generators. Note that the urn scheme performs like the probabilistic scheme for low demand, and like the serial scheme for high demand.

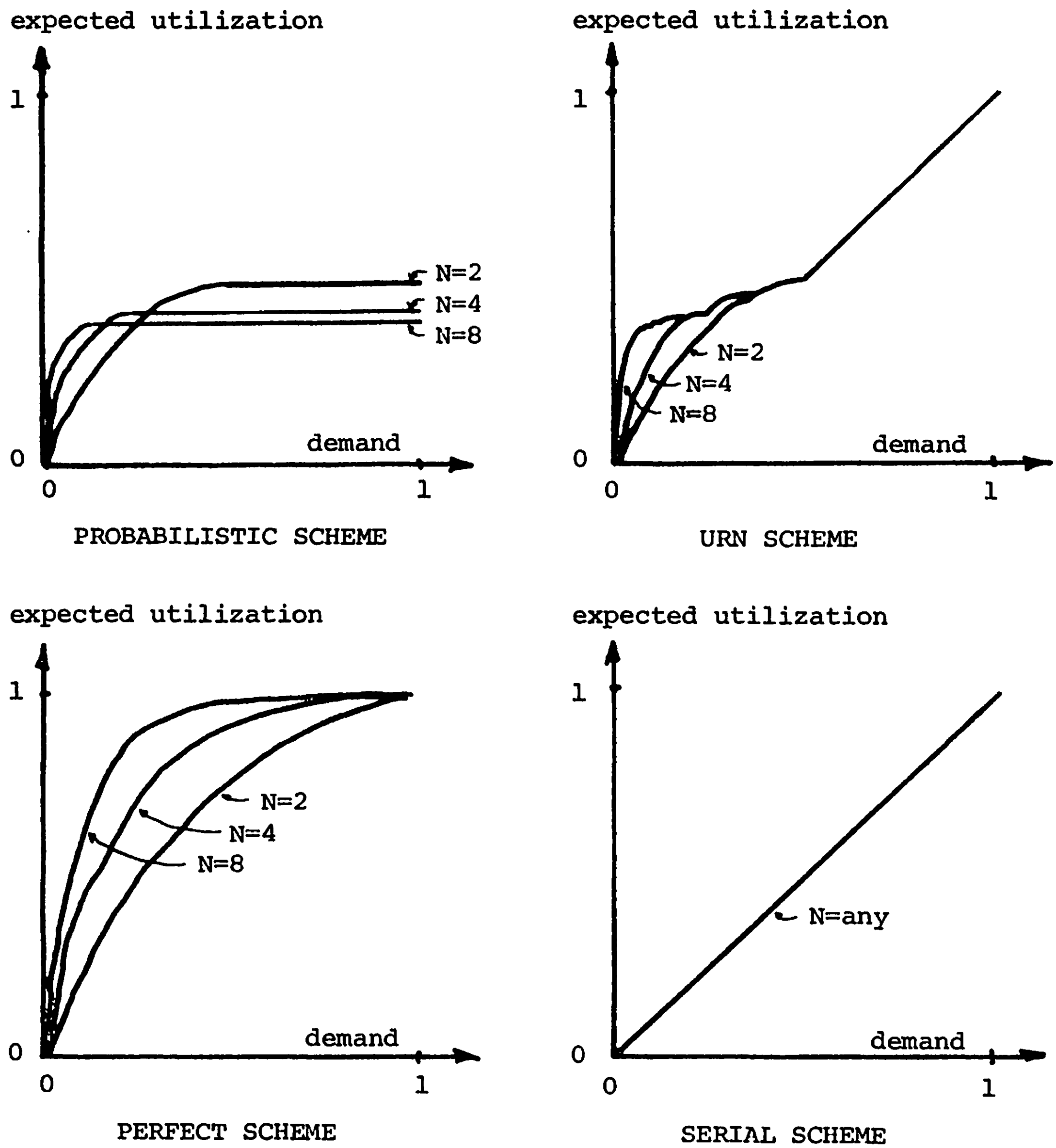


Figure 2: Expected Utilization vs. Demand with N PEs and no Penalty

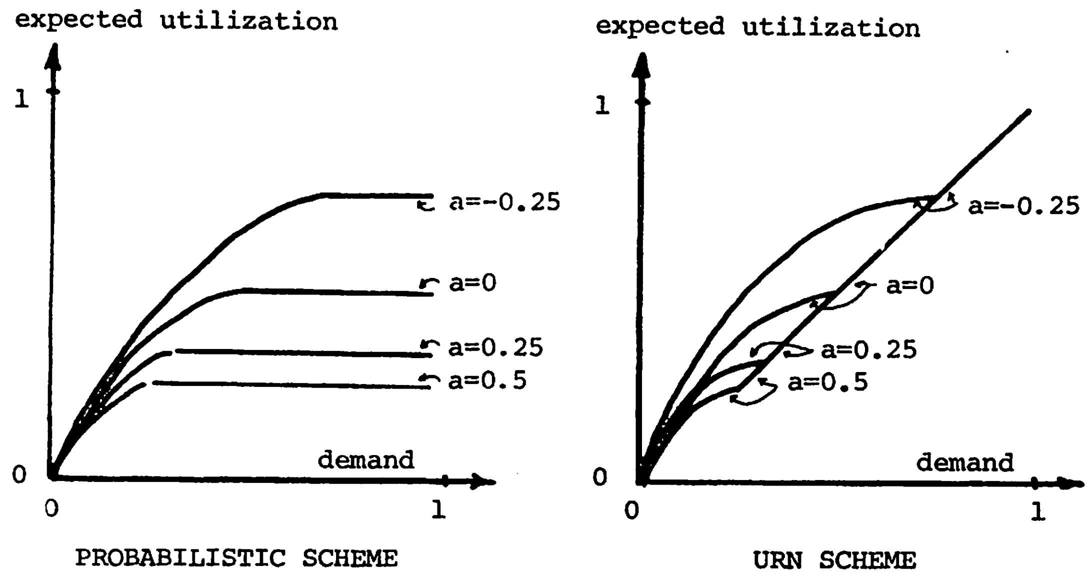


Figure 3: Expected Utilization vs. Demand with 2 PEs and Penalt a

A perfect scheme (implemented with a central processor or global negotiation) is, of course, more effective; but, the communication costs of the perfect scheme can be prohibitive. The benefits of a perfect scheme are greatest for large N and low demand, but communication costs are high for large N since communication costs are geometrically related to N.

Figure 3 shows that a penalty for collisions forces the probabilistic and urn schemes to adopt more conservative policies; thus, they become less effective as the penalty increases. A penalty of a=1/2, for example, cuts utilization in half. A negative penalty (actually a reward) causes the policies to be more optimistic.

8. An Adaptive Balancing Algorithm

It is possible for the balance principle developed in the previous section to serve as the basis for an adaptive, distributed access control algorithm that is quasi-static (i.e., appropriate for slowly changing environments) and entirely decentralized. The algorithm implements a decentralized, iterative process that estimates and balances PEs' conditional expected neighborhood utilization payoffs (expected neighborhood utilization payoff given the PE has demand and attempts to access shared resources, and expected neighborhood utilization payoff given the PE has demand but sacrifices). While a detailed analysis of the algorithm is beyond the scope of this paper, a description and some simulation results are included below.

Conventional adaptive access control algorithms rely on an estimate of the number of PEs with demand and an optimality condition similar to the balance principle to compute an optimal policy [5],[9]. In much the same way, an estimate of the number of PEs with demand can be used in conjunction with the balance principle derived in Section 4 to compute optimal policies. Estimates of the number of PEs with demand can be maintained by heuristic means [5] or by recursive estimation [9]. Unfortunately, these schemes are limited to the case where there is a single resource shared by all PEs; this is not so for the scheme proposed below.

The alternative adaptive access control scheme proposed below incrementally adjusts policies based on estimated conditional expected utilization payoffs. The estimates of conditional expected utilization payoffs are maintained through observations and/or communicated information concerning PEs' actual utilizations, and may be supplemented by inferences of PEs' conditional expected utilizations. Note that estimates of PEs' demand need not be maintained, and a precise model of the utilization process need not be known if actual utilizations are observed or communicated. Furtheremore, the algorithm is appropriate for arbitrary network topologies.

For reasons soon to be explained, it is actually best that each PE maintain separate estimates of conditional expected utilization payoffs for itself and each of its neighbors. This usually involves obtaining neighbors' utilizations via communication each iteration a PE has demand. With utilization processes for which the special case of the balance principle discussed briefly at the end of Section 5 holds, only a PE's neighbors' utilization of resources shared with the PE are needed to update the PE's estimates; this can be obtained without communication if utilization of resources to which a PE has access is observable by the PE.

The payoff coefficients, $\underline{c}$, determine the particular Pareto-optimal policy and utilization that is obtained when balance is achieved. Thus, the adaptive balancing algorithm does not strive to find some random Pareto-optimum, it strives to find the particular Pareto-optimum selected by the coefficients, $\underline{c}$. These coefficients may be preset or manipulated, via a central processor, a hierarchy, or a second level of adaptation, to distribute utilization among PEs of the network. Unfortunately, for irregular networks $\underline{c}=(1,1,\ldots 1)$ does not give equal utilization to all PEs. Although a PE need not keep separate estimates because of this, a PE must distinguish between neighbors when observing utilizations if $\underline{c}=(1,1,\ldots 1)$ so that the utilization payoffs can be multiplied by the appropriate payoff coefficients.

Given a particular vector of payoff coefficients, $\underline{c}$, let PE_i's estimate of its expected neighborhood utilization payoff (nup) given policy $\underline{p}$ and that PE_i attempts to access shared resources be

$$E[nup_i | \underline{p} \wedge attempt(i)] =$$

$$c_i E[\tilde{u}_j | \underline{p} \wedge attempt(i)] + \sum_{j \in I(i)} c_j E[\tilde{u}_j | \underline{p} \wedge attempt(i)]$$

and, let PE_i's estimate of its expected neighborhood utilization payoff given PE_i sacrifices be

$$E[nup_i | \underline{p} \wedge sacrifice(i)] =$$

$$c_i E[\tilde{u}_j | \underline{p} \wedge sacrifice(i)] + \sum_{j \in I(i)} c_j E[\tilde{u}_j | \underline{p} \wedge sacrifice(i)].$$

Furthermore, assuming a probabilistic scheme, let PE_i's estimate of a neighbor's expected utilization payoff (given PE_i has demand) be

$$E[up_j | \underline{p} \wedge demand(i)] =$$

$$p_i c_j E[\tilde{u}_j | \underline{p} \wedge attempt(i)] + (1-p_i) c_j E[\tilde{u}_j | \underline{p} \wedge sacrifice(i)].$$

Note that $E[up_j | \underline{p}\text{^}demand(i)]$ is a good estimate of a neighbor's expected utilization pay-off.

Now, if $E[nup_i | \underline{p}\text{^}attempt(i)] < E[nup_i | \underline{p}\text{^}sacrifice(i)]$ then PE_i knows that its neighbors are over-utilizing shared resources and are interfering with its attempts to access shared resources. For this reason, PE_i should notify its neighbors to decrease their attempts to access shared resources. It is important to note that changes in PE_i's policy will not directly affect the imbalance seen by PE_i, but changes to neighbors' policies will. For this reason, changes to a PE's policy are, for the most part, "neighbor-directed" rather than self-directed.

If $E[nup_i | \underline{p}\text{^}attempt(i)] > E[nup_i | \underline{p}\text{^}sacrifice(i)]$ then PE_i knows that its neighbors are under-utilizing shared resources and are wasting PE_i's sacrifices. It is tempting to say that PE_i should notify neighbors with low $E[up_j | \underline{p}\text{^}demand(i)]$ to increase their attempts to access shared resources; however, under-utilization of shared resources by neighbors may be due to neighbors' attempting too much (and colliding) as well as attempting too little.

Two approaches to resolving this uncertainty are for a PE to try to determine if neighbors are under-utilizing because they are attempting too much or too little, or for a PE to let the neighbors determine that. For a PE to determine why neighbors are under-utilizing shared resources it might look at its own expected utilization payoff given it attempts to access shared resources; if this is low then neighbors are attempting too much, and if high then neighbors are attempting too little. For a neighbor to determine if it is attempting too much or too little, the neighbor need only look at its policy or note whether or not it just collided or sacrificed; if its policy is high or it just collided then it is attempting too much, and if its policy is low or it just sacrificed then it is attempting too little.

To distribute utilization quickly in the manner specified by the payoff coefficients it is important that PE_i direct stronger notifications to increase/decrease attempts to access shared resources to neighbors with low/high $E[up_j | p\text{-}demand(i)]$. This is why separate estimates of conditional expected utilization payoffs for PE_i and its neighbors should be kept. Note that if $c_i \neq c_j$ for some neighbor, PE_j, then PE_i must distinguish between different neighbors' utilization of resources anyway.

In general a PE should readjust its likelihood of attempting to access given demand in accordance with notifications received from neighbors, by changing its policy. Because of the problem mentioned in the previous paragraph, though, a PE might be required to decrease its attempts to access if it has just collided or its policy is high, and otherwise readjust its attempts to access according to notifications received from neighbors. PEs' policies are bounded, so a PE cannot increase its policy beyond 1.0 or decrease it beyond 0.0 for the probabilistic scheme; the upper and lowwer limits for the urn scheme are N and 0.0, respectively.

Results of a simple adaptive balancing algorithm are included in the next section. This algorithm can be described by the following rules:

> 1) If a PE's expected neighborhood utilization payoff given it attempts to access resources is greater than its expected neighborhood utilization payoff given it sacrifices, then it should notify neighbors with minimal estimated utilization payoff to increase their policies by two step-sizes, and should notify other neighbors to increase their policies by one step-size;
>
> 2) If a PE's expected neighborhood utilization payoff given it attempts to access resources is less than its expected neighborhood utilization payoff given it sacrifices, then it should notify neighbors with maximal estimated utilization payoff to increase their policies by two step-sizes, and should notify other neighbors to increase their policies by one step-size;
>
> 3) Each PE lowers its policy by one step-size if it suffers a collision; otherwise, it increases its policy by an amount determined by summing notifications received from neighbors and normalizing by dividing by the number of neighbors. Policies are only permitted to take on values between 0.0 and 1.0 (inclusive) for the probabilistic scheme, or between 0.0 and N (inclusive) where N is the number of PEs for the urn scheme.

A step-size specifies the basic magnitude of changes to policies, thus controlling the reactiveness of the algorithm.

Remember that the balance principle represents a necessary but not sufficient condition for Pareto-optimality of an internal policy. For example, the rude policy, $\underline{p}=(1,1,...1)$, satisfies the balance principle if a=0 and demand is high, yet is not Pareto-optimal. This does not, however, appear to present a problem for the adaptive balancing algorithm just described, which will not remain at the rude policy if there are any collisions. Other non-Pareto optimal equilibrium have not been encountered.

9. Simulation Results

Simulation results of the adaptive balancing algorithm using the probabilistic and urn schemes to control access to a single shared resource are presented in this section. Demand is generated randomly and independently for each PE and iteration. The probability that each PE has demand can be specified, and initial policies and estimates can be provided so that transient behavior can be observed.

A simple estimation scheme can be used to maintain estimates of conditional expected utilizations, from which conditional expected neighborhood utilization payoffs are calculated. A "window-size" is specified, and when PE_i has demand and attempts to access shared resources (say after time-slot t) it updates $E[\tilde{u}_j | \underline{p}\text{^attempt}(i)]$ using the rule,

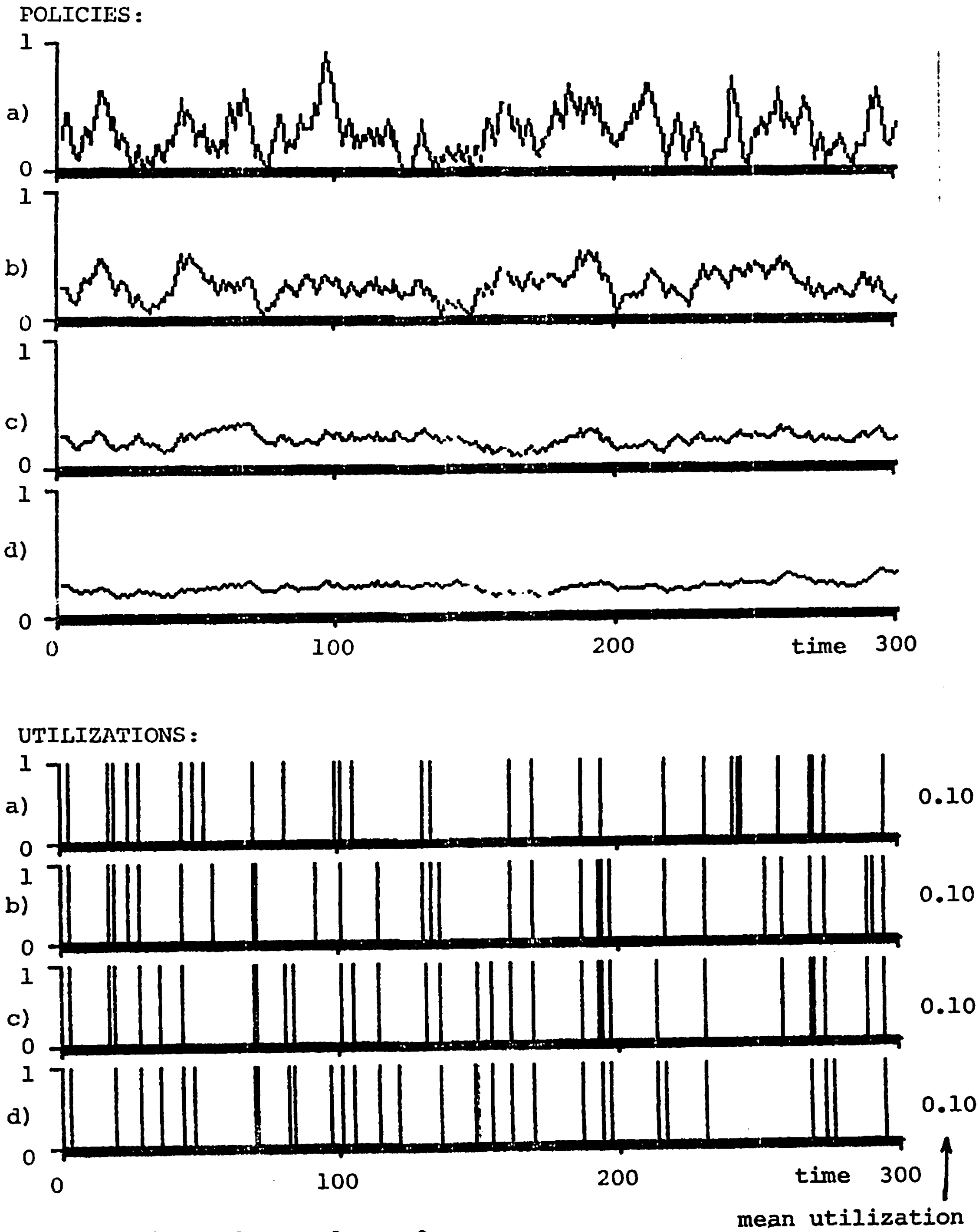
$$E[\tilde{u}_j | \underline{p}\text{^attempt}(i)] :=$$
$$((\text{window\_size}-1)E[\tilde{u}_j | \underline{p}\text{^attempt}(i)] + \tilde{u}_j)/\text{window\_size}$$

for each neighbor. Similar rules are used to update $E[\tilde{u}_j | \underline{p}\text{^attempt}(i)]$ and, when PE_i has demand and sacrifices, $E[\tilde{u}_j | \underline{p}\text{^sacrifice}(i)]$ for each neighbor and $E[\tilde{u}_j | \underline{p}\text{^sacrifice}(i)]$. By varying window-size, the amount of history retained by PEs can be controled. Although the simple estimation scheme works it can be many iterations between attempts/sacrifices if demand is high/low and estimates may get old.

To obtain the simulation results shown below a more complex estimation scheme was used with the adaptive balancing algorithm described in the previous section. This estimation scheme took advantage of the network topology (a single resource shared by all PEs) by inferring conditional expected utilization payoffs of PEs from knowledge of which PEs attempt to access resources. This permitted the updating of all estimates of conditional expected utilization payoffs each iteration reguardless of the actions of PEs, and resulted in a more responsive algorithm.

All simulations involved the case of 4 PEs that share a single resource with payoff coefficient vector, $\underline{c}$=(1,1,1,1). The next 7 figures show the behavior of the adaptive balancing algorithm. For each figure, PEs' policies are shown in the top half of the figure, and utilizations are shown in the bottom half. Because utilizations are discrete events, the average utilization is shown to the right of the graphs.

Figures 4 and 5 depict the probabilistic scheme starting at a Pareto-optimum. Figure 4 shows the policies and utilizations of a typical PE with various step-sizes and Figure 5 shows the policies and utilizations of a typical PE with various window-sizes. In both cases, all PEs have demand each iteration with probability 1.0 (i.e., $\underline{d}$=(1.0,1.0,1.0,1.0) for all iterations) and there is no penalty for collisions, so the Pareto-optimal policy is $\underline{p}$=(0.25,0.25,0.25,0.25); each PE's expected utilization payoff should be approximately 0.11. Note that the algorithm functions well even when the step-size is quite



window-size = 2, penalty = 0,

step-size = { a) 0.10, b) 0.05, c) 0.02, d) 0.01 }

Figure 4: Probabilistic Scheme at Equilibrium with various step-sizes

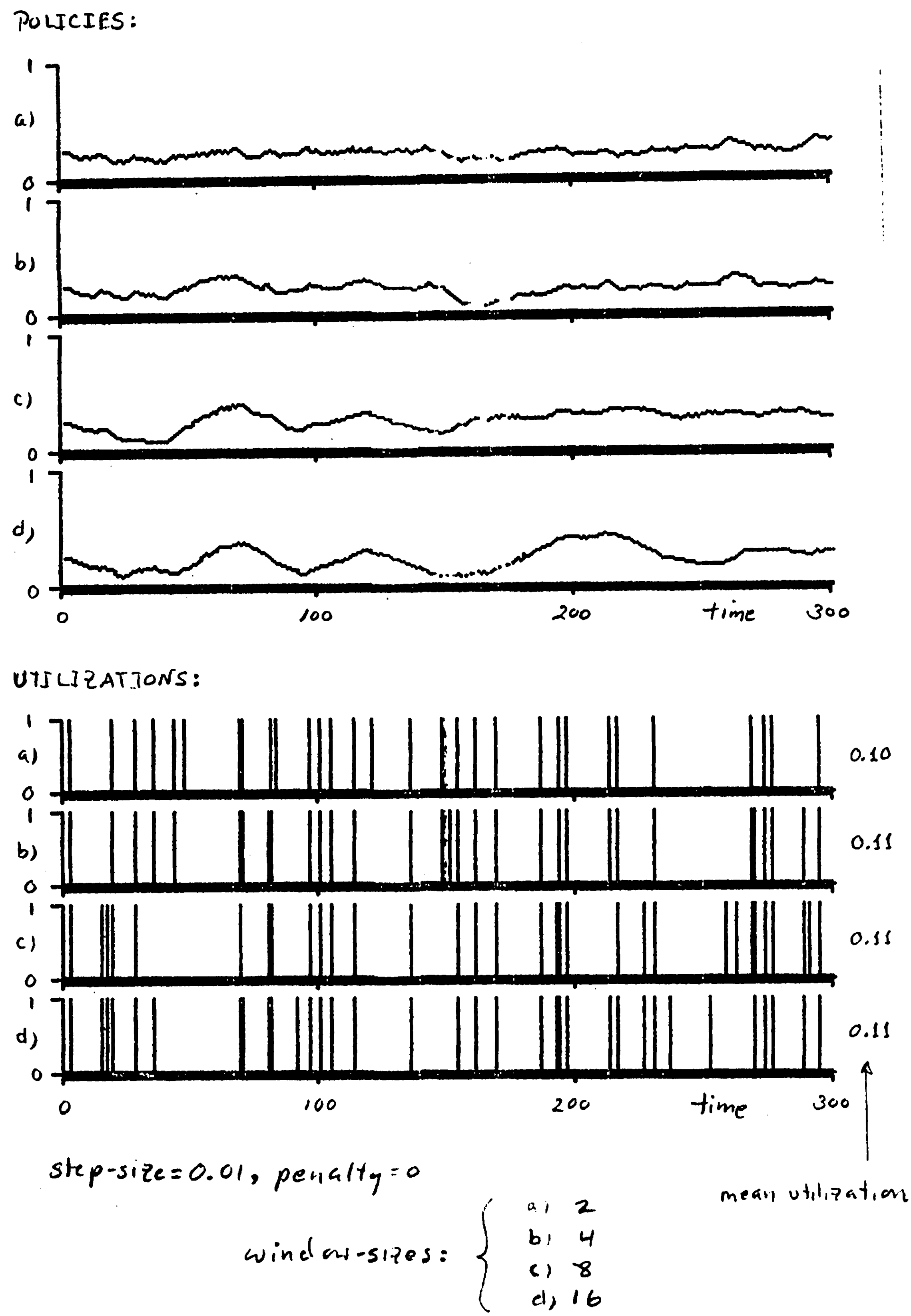


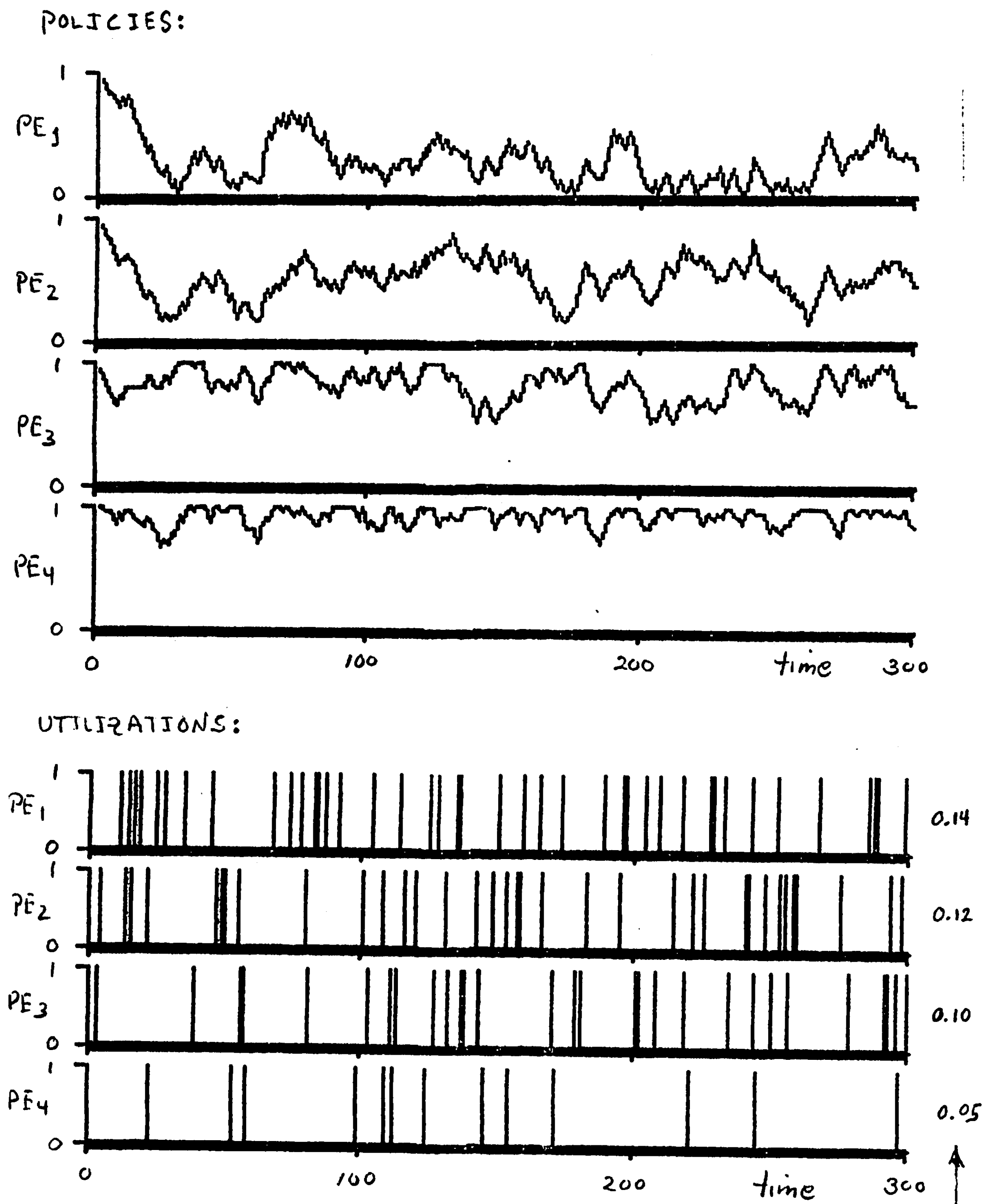
Figure 5: Probabilistic Scheme at Equilibrium with various window-sizes

high.

Figures 6 and 7 depict the probabilistic scheme coping with a dramatic change in load. The PEs begin with policies and estimates appropriate for $\underline{d}$=(0.25,0.25,0.25,0.25), but are faced with $\underline{d}$=(1.0,0.5,0.25,0.1) for all iterations. A window-size of 2 and step-size of 0.1 are assumed for Figure 6; a window-size of 2 and step-size of 0.01 are assumed for Figure 7. It took approximately 20 iterations to adapt with a step-size of 0.1, and approximately 120 iterations to adapt with a step-size of 0.01.

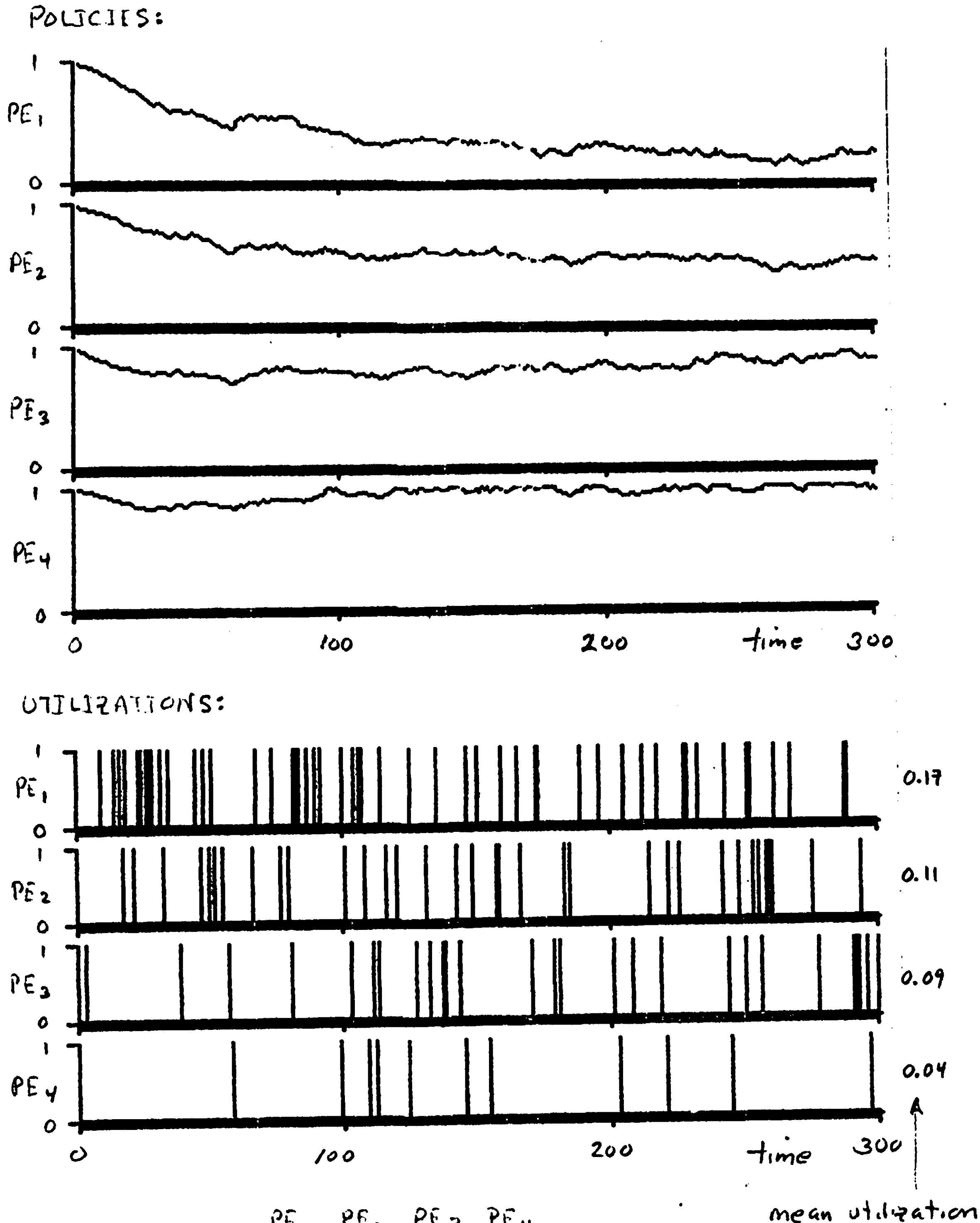
The Pareto-optimal policy sought by the network of PEs for runs depicted in Figures 6 and 7 is an extreme policy, $\underline{p}$=(0.25,0.5,1.0,1.0). Notice that for i=1, 2, and 3 $d_i p_i$=0.25, but $d_4 p_4$=0.1. In this situation the adaptive balancing algorithm is striving to find a $\underline{p}$ such that $d_i p_i$=0.25 for all PEs. However, balance cannot be achieved with $\underline{c}$=(1,1,1,1). Thus, after an initial transient period PE_1, PE_2, and PE_3 repeatedly send messages to PE_4 requesting that it increase its attempts to access shared resources; PE_4, though, cannot increase p_4 beyond 1.0. This behavior of the adaptive balancing algorithm at such extreme Pareto optima seems reasonable.

The probabilistic scheme at equilibrium, but with various penalties for collisions is depicted in Figure 8. The policies and utilizations are as expected. Note that the particular version of adaptive balancing algorithm that was simulated has each PE_i decrease p_i if it is involved in a collision; this may be inappropriate for negative penalties (i.e., when collisions are beneficial).



	PE_1	PE_2	PE_3	PE_4
demand	1.0	0.5	0.25	0.1
initial policies	1.0	1.0	1.0	1.0
initial estimates	0.4	0.4	0.4	0.4

Figure 6: Transient Behavior of Probabilistic Scheme with large step-size



	PE1	PE2	PE3	PE4
demand	1.0	0.5	0.25	0.1
initial policies	1.0	1.0	1.0	1.0
initial estimates	0.4	0.4	0.4	0.4

Figure 7: Transient Behavior of Probabilistic Scheme with small step-size

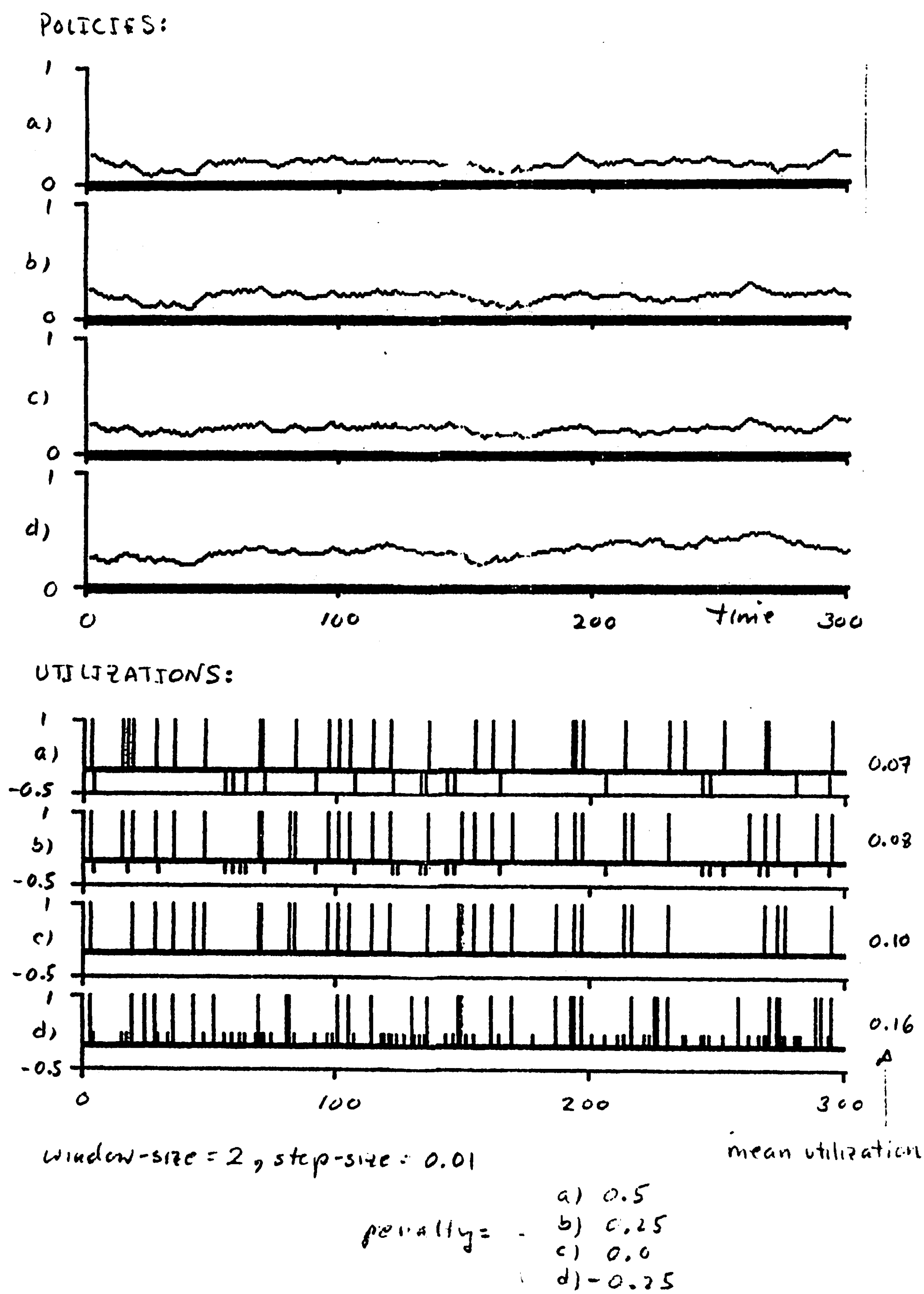
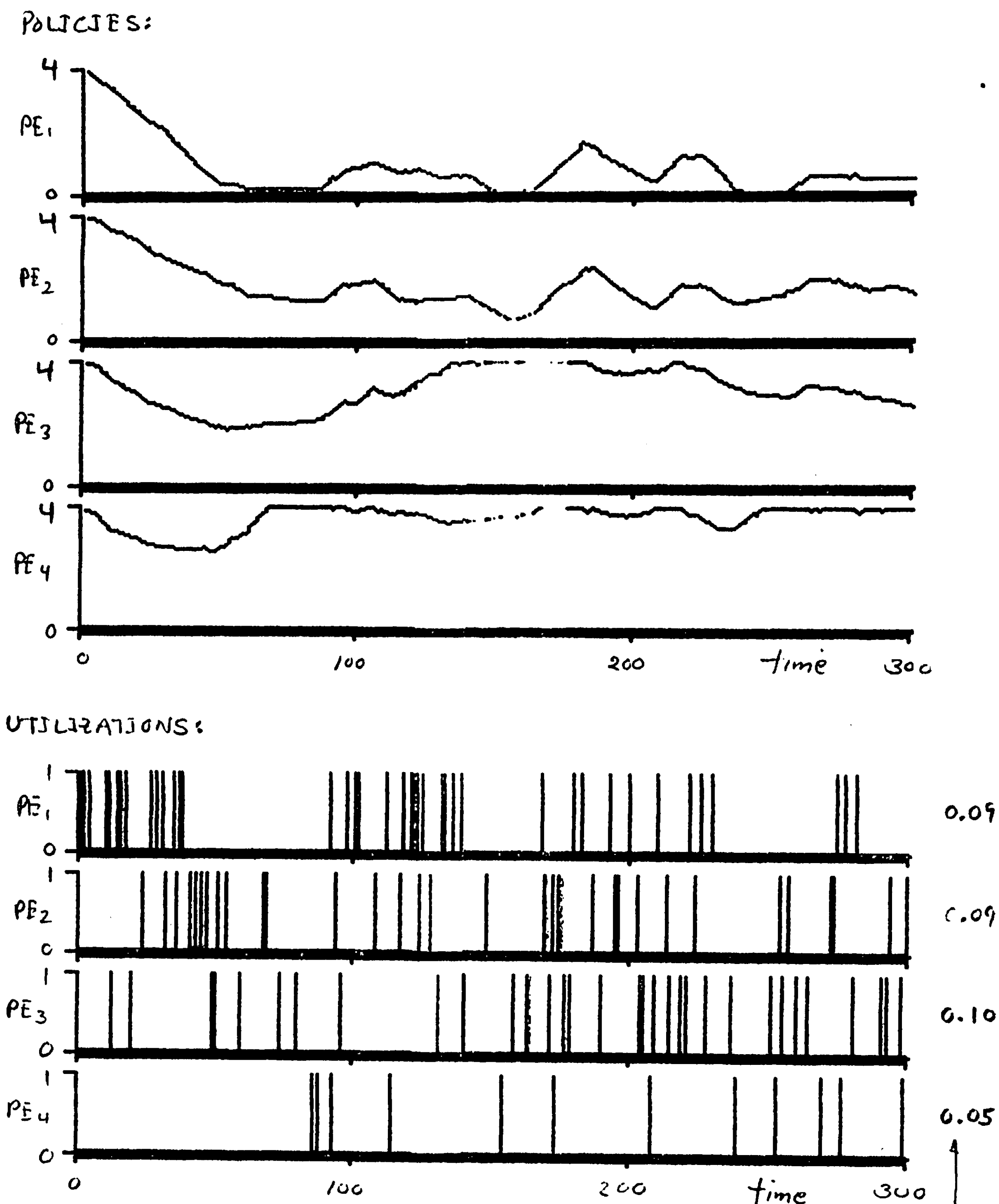


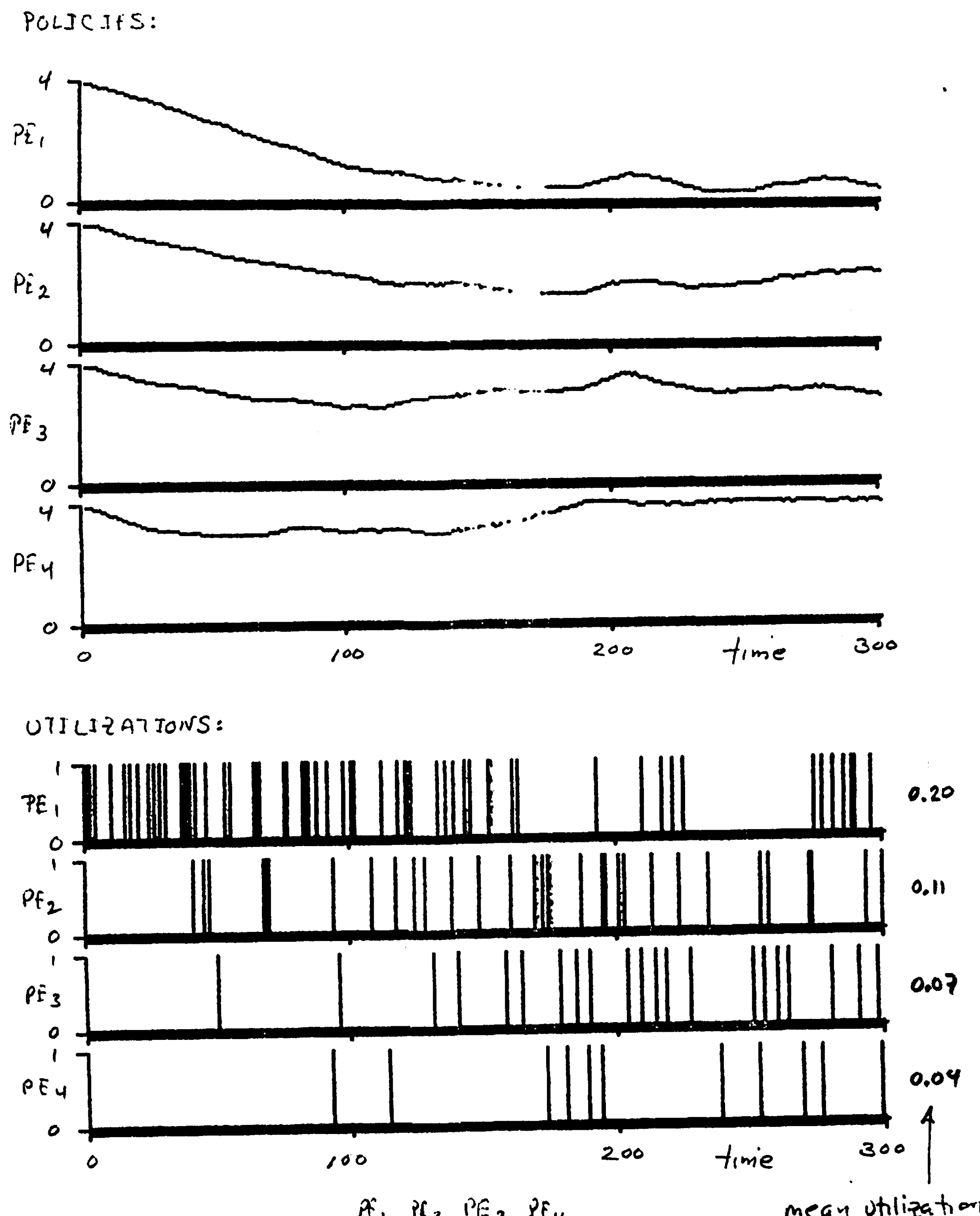
Figure 8: Probabilistic Scheme at Equilibrium with various penalties

Finally, Figures 9 and 10 depict the urn scheme coping with the same dramatic change in load as the probabilistic scheme faced in Figure 6 and Figure 7. With the urn scheme, policies range from 0.0 to N. Also, policies are rounded off to integers when used for the urn scheme because an integral number of numbers are to be drawn from pseudo-random number generators. Note that PE_1 would have a constant policy of 1.0 and utilization of 0.25 if the other PEs also had high demand, but other PEs' low demand forces PE_1 to sacrifice some of these slots.



	PE1	PE2	PE3	PE4
demand	1.0	0.5	0.25	0.1
initial policies	4.0	4.0	4.0	4.0
initial estimates	0.4	0.4	0.4	0.4

Figure 9: Transient Behavior of Urn Scheme with large step-size



	PE_1	PE_2	PE_3	PE_4
demand	1.0	0.5	0.25	0.1
initial policies	4.0	4.0	4.0	4.0
initial estimates	0.4	0.4	0.4	0.4

Figure 10: Transient Behavior of Urn Scheme with small step-size

10. Conclusions

A simple, yet powerful "balance principle" for optimal, distributed multi-access control has been derived using the concept of Pareto optimality. The balance principle covers a broad range of objective functions; the objective function may include penalties for collisions and other events. The principle states that when PEs are using a Pareto-optimal policy there exists a vector of payoff coefficients such that for all PEs expected neighborhood utilization payoff given the PE has demand and attempts to access shared resources equals expected neighborhood utilization payoff given the PE has demand but does not attempt to access shared resources.

Provided that the probability that a PE will attempt to access shared resources given the PE has demand can be expressed as a function of the access control policy, the balance principle is appropriate. Two important access control schemes of this type, the probabilistic and urn schemes, were examined. The urn scheme is better than the probabilistic scheme because it performs better at high demand. A high penalty for collisions tends to make the probabilistic and urn schemes less effective because PEs must be more cautious to avoid collisions.

The balance principle can be used in a number of ways. Given that demand is regular and statistics are known, the balance principle can be used to derive an optimal, fixed policy. When demand varies randomly over time, the balance principle can be used in a conventional manner to determine the optimal policy given an estimate of the demand and penalty statistics. Alternatively, the balance principle can be used in an adaptive balancing algorithm as described in Section 5. PEs' theoretical expected utilization with a Pareto-optimal policy was shown in Figures 2 and 3.

The communication requirements of the adaptive balancing algorithm depends on a number of factors. Most of the communication is involved in maintaining estimates of conditional expected utilization payoffs, unless utilization of resources by each PE and its neighbors can be observed by the PE. The messages transmitted between PEs to adjust policies are small, and PEs need not send messages to neighbors each iteration.

Results of simulations of the adaptive balancing algorithm were discussed in Section 6, and indicate that the adaptive balancing algorithm does achieve Pareto-optimal utilizations. The algorithm is effective with a large step-size which makes it reasonably quick to adapt to changes in demand. Further development and testing of the adaptive balancing algorithm in a more complex setting appears to be in order.

References

[1] N. Abramson, "Packet Switching with Satellites," AFIPS Conference Proceedings, Vol. 42, p. 695-702, June 1973.

[2] R.S. Brooks, Experiments in Distributed Problem Solving with Iterative Refinement, Ph.D. Thesis, Computer and Information Science, University of Massachusetts, Amherst, MA, in preparation.

[3] J. Capetanakis, "Tree Algorithms for Packet Broadcast Channels," IEEE Transactions on Information Theory, Vol. IT-25, No. 5, p. 505-515, September 1979.

[4] L. Kleinrock and S. Lam, "Packet-Switching in a Slotted Satellite Channel," AFIPS Conference Proceedings, Vol. 42, p. 703-710, June 1973.

[5] L. Kleinrock and Y. Yemini, "An Optimal Adaptive Scheme for Multiple-Access Broadcast Communication," Proceedings of ICC Conference, Toronto, Canada, June 1978.

[6] R.D. Luce and H. Raiffa, Games and Decisions, New York: John Wiley and Sons, Inc., 1957.

[7] A. Rapoport, Two-Person Game Theory: The Essential Ideas, Ann Arbor: University of Michigan Press, 1966.

[8] A. Rapoport, N-Person Game Theory: Concepts and Applications, Ann Arbor: University of Michigan Press, 1970.

[9] A. Segall, "Recursive Estimation from Discrete-Time Point Processes," IEEE Transations on Information Theory, Vol. IT-22, No. 4, p. 422-431, July 1976.

[10] Y. Yemini, On Channel Sharing in Discrete-Time, Packet-Switched, Multi-Access Broadcast Communication, Ph.D. Disertation, Computer Science Department, School of Engineering and Applied Science, UCLA, 1978.

[11] Y. Yemini and L. Kleinrock, "On a General Rule for Access Control or, Silence is Golden ...," Distributed Sensor Networks, ISI Working Paper 12, USC Information Sciences Institute, Marina del Rey, CA, April 1979.

[12] Y. Yemini and L. Kleinrock, "Broadcasting in an Urn: An Optimal Adaptive Multiaccess Scheme," in preparation, 1980.