Index Policies for Resource Allocation in Wireless Networks

Nomesh Bolia

nomesh@unc.edu

Vidyadhar Kulkarni

vkulkarn@email.unc.edu

Department of Statistics and OR, UNC Chapel Hill

Abstract *We consider the problem of resource allocation for data transfer between the base station and the users within a cell of a wireless telecommunication network with infinite data queues for each user. The aim is to study the tradeoff between the conflicting objectives of maximizing the system throughput and the quality of service to an individual user. Using a policy improvement approach based on Markov Decision Processes, we develop an intuitive and easy to implement index policy. We also demonstrate its superior performance over the existing Proportional Fair Metric Algorithm through simulation experiments.*

1 Introduction

With the coming of third-generation wireless systems mobile internet users can obtain high data rates in cellular systems. To effectively utilize the wireless capacity we require efficient methods for assignment of wireless bandwidth to the various users using a common base station.

We start by considering a fixed set of N mobile data users in a wireless cell served by a single base station. We focus on the downlink (base station to mobile) direction, since in many applications such as web browsing, most of the data flow occurs in that direction. The base station maintains a separate queue of data for each user. Time is slotted and in each time slot the base station can transmit data to exactly one user. Let R_u^n be the channel rate of user u during time slot n, i.e., the amount of data that can be transmitted to user u during time slot n by the base station. How this information is obtained by the base station is irrelevant for our purposes. We assume that the base station knows at all time slots n a vector $R^n = (R_1^n, R_2^n, \ldots, R_N^n)$. $\{R^n, n \geq 0\}$ is a stochastic process that accounts for the random variation in data rates due to user mobility and other factors such as propagation. Georgiadis, Neely and Tassiulas (2006) present a good framework for resource allocation and related issues in this (and more general) setting.

An example of a resource allocation system widely known and used in practice is the CDMA2000 1xEV-DO system (Bender et al, 2000). There are two objectives to be fulfilled while scheduling the data transfer. The first is to obtain a high data transfer rate. This can be achieved by serving a user u in slot n whose channel rate R_u^n is the highest, i.e. following a myopic policy. However if we follow the myopic policy, we run the risk of severely starving users whose channel rate is low for a long time. The second objective then is to ensure that none of the users is severely *starved*. Thus these are conflicting objectives and any good algorithm tries to achieve a "good" balance between the two. Liu, Chong and Shroff (2001) present another algorithm that seeks to allocate resources to maximize system throughput under some Quality of Service (QoS) constraints that ensure some amount of fairness to each user. We shall comment on this algorithm later in section 5.

The outline of the rest of the paper is as follows. In section 2 we describe a popular algorithm called Proportional Fair Algorithm (PFA) currently in use in the 1xEV-DO system. In section 3 we formulate the bandwidth allocation problem as a Markov Decision Process under the assumption of a fixed number of users in the cell. We develop the index policy as an approximation to the optimal policy using one step of policy improvement algorithm. In section 4 we study the properties of the Linear Index Policy (LIP) which is a special case of the index policies developed in section 3. We also extend it to the case of variable numbers of users (arriving in the cell according to a Poisson process and staying for a random amount of time) in the cell. Finally, in section 5.2, we compare the LIP with PFA by using simulation and show that LIP is superior to PFA.

2 Proportional Fair Algorithm

The Proportional Fair Algorithm (PFA) is currently used in the 1xEV-DO system. In this paper we consider the case where every user always has data to transmit. The PFA aims to optimize a given function of the throughput achieved by all the users. A commonly used objective function is the Proportional Fair metric $\sum_u \log Q_u$, where Q_u is a given measure of the long term throughput achieved by user u. A useful characteristic of this metric is that although it is strictly increasing in the throughput of each user, it prevents any user from being starved since $\log Q_u \to -\infty$ as $Q_u \to 0$. The PFA is characterized by a single constant $\tau \in (0,1)$ as explained below. Assume that there are a fixed number N of users in the cell. Let $v(n)$ be the user served in slot n. For each user u, define $Q_u(0) = 1$, and compute $Q_u(n)$ recursively as follows:

$$Q_u(n+1) = \begin{cases} (1-\tau)Q_u(n) + \tau R_u^n & \text{if } u = v(n) \\ (1-\tau)Q_u(n) & \text{if } u \neq v(n), \end{cases} \tag{1}$$

see Andrews (1999). The constant $t_c = \frac{1}{\tau}$ is a measure of the time a user can remain unserved (Jalali et al, 2000). Clearly $Q_u(n)$ represents an exponentially filtered average service rate. The PFA algorithm chooses to serve user $v(n)$ in slot n where

$$v(n) = \arg\max_u \frac{R_u^n}{Q_u(n)}. \tag{2}$$

It can be proved that this algorithm maximizes $\sum_u \log Q_u(n+1)$ - $\sum_u \log Q_u(n)$ for each n (Andrews, 1999). In the next section we describe our algorithm based on a Markov Decision Process (MDP) approach.

3 Index Policy for Scheduling Data Transfer

3.1 Introduction and Formulation as MDP

In this section we start with a stochastic model for $\{R^n,\ n \geq 0\}$ and formulate the scheduling problem as an MDP. Let X_u^n be the state of user u at time n. This represents all the various factors such as the position of the user in the cell, the propagation conditions etc that determine the data rate received by u in time slot n. We assume that $\{X_u^n,\ n \geq 0\}$ is an irreducible Discrete Time Markov Chain (DTMC) on state space $\Omega = \{1, 2, \ldots, M\}$ with Probability Transition Matrix (PTM) $P^u = [p^u_{i_u, j_u}]$. Let R_k be the fixed data rate (or channel rate) associated with state $k \in \Omega$ of the DTMC. When $X_u^n = k$, the user u receives data from the base station at rate $R_u^n = R_k$. Let N be the total number of users in the cell (assumed constant in this section) and $X^n = [X_1^n, \ldots, X_N^n]$ be the state vector of all the users. Clearly $\{X^n, n \geq 0\}$ is a DTMC on Ω^N. We assume the users behave independently of each other and that each user has ample data to transmit.

Let Y_u^n be the "starvation age" (or simply "age") of the user u at time n, defined as the time elapsed (in number of slots) since the user u was served most recently. Thus, the age of the user is zero at time $n+1$ if it is served in the n^{th} time slot. Furthermore, for $m \geq 1$, if the user was served in time slot n and it is not served for the next m time slots, its age at time $n+m$ is $m-1$. Let $Y^n = [Y_1^n, \ldots, Y_N^n]$ be the age vector at time n. The base station (BS) serves exactly one user in each time slot. Let $v(n)$ be the user served in the n^{th} time slot. The age variables change according to

$$Y_u^{n+1} = \begin{cases} Y_u^n + 1 & \text{if } u \neq v(n) \\ 0 & \text{if } u = v(n) \end{cases} \tag{3}$$

The "state of the system" at time n is given by $(X^n, Y^n) \in \Omega^N X Z^N$, where $Z = \{0, 1, 2, \ldots\}$. The "state" is thus a vector of $2N$ components and we assume that it is known at the base station in each time slot. After observing (X^n, Y^n) the BS decides to serve one of the N users in the time slot n. We need a reward structure in order to make this decision optimally. We describe such a structure below. If we serve user u in the n^{th} time slot, we earn a reward of $R_u^n = R_{X_u^n}$ for this user and none for the others. In addition, there is a cost of $D_l(y)$ if user l of age y is not served in slot n. Clearly, we can assume $D_l(0) = 0$ since there is no starvation at age zero. Thus the net reward of serving user u at time n is

$$R_{X_u^n} - \sum_{l \neq u} D_l(Y_l^n). \tag{4}$$

For convenience we use the notation

$$W_u^n = \sum_{l \neq u} D_l(Y_l^n).$$

The problem of scheduling a user in a given time slot can now be formulated as a Markov Decision Process (MDP). The decision epochs are $\{1, 2, \dots\}$. The state at time n is (X^n, Y^n), where both X^n and Y^n are vectors of N components, whose evolution has been described above. The action space in every state is $\{1, 2, \dots, N\}$ where action u corresponds to serving the user u. The reward in state (X^n, Y^n) corresponding to action u is $R_u^n - W_u^n$. Let the transition probability under action u from (i, s) to (j, t) ($i, j \in \Omega^N$ and $s, t \in Z^N$) be denoted by $p((j, t)|(i, s), u)$. It is given by

$$p((j,t)|(i,s),u) = \begin{cases} p^1_{i_1,j_1} p^2_{i_2,j_2} \cdots p^N_{i_N,j_N} = p_{ij} & \text{if } t_u = 0 \text{ and } t_l = s_l + 1 \text{ for } l \neq u \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

Let $V_T(i, t)$ be the optimal reward starting from state $(X^0, Y^0) = (i, t)$ at time 0 over time periods $0, 1, 2, \dots, T$. If user u is served at time 0, the age vector in the next time slot is $t^u = (t_1 + 1, \dots, t_{u-1} + 1, 0, t_{u+1} + 1, \dots, t_N + 1)$. We also define

$$W_u(t) = \sum_{l \neq u} D_l(t_l).$$

A standard Dynamic Programming (DP) argument then yields the following Bellman equation

$$V_T(i,t) = \max_{u=1,2,\ldots,N} \left[R_{i_u} - W_u(t) + \sum_j p_{ij} V_{T-1}(j, t^u) \right], \tag{6}$$

We wish to determine the action $u = u(i,t)$ that maximizes $\lim_{T\to\infty} V_T(i,t)/T$, i.e. the long run average reward. It is well known (Puterman, 1994) that such a policy $\{u(i,t) : (i,t) \in \Omega^N X Z^N\}$ exists if there is a constant g and a bias function $w(i,t)$ satisfying

$$g + w(i,t) = \max_u \{R_{i_u} - W_u(t) + \sum_j p_{ij} w(j, t^u)\}. \tag{7}$$

Furthermore, any u that maximizes $R_{i_u} - W_u(t) + \sum_j p_{ij} w(j, t^u)$ over all $u \in \{1, \ldots, N\}$ is an optimal action $u(i,t)$ in state (i,t).

3.2 Policy Improvement Approach

In this section we use a policy-improvement approach to develop a heuristic scheduling policy. We first describe the standard policy improvement algorithm (Puterman, 1994).

1. Let π^0 be an arbitrary policy that chooses action $\pi^0(i,t)$ in state (i,t). Set $n = 0$.

2. Policy Evaluation Step: Compute the solution to

$$g_n + w_n(i,t) = R_{i_u} - W_u(t) + \sum_j p_{ij} w_n(j, t^u), \ (i,t) \in \Omega^N X Z^N$$

   where $u = \pi^n(i,t)$.

3. Policy Improvement Step: Let

$$\pi^{n+1}(i,t) = \arg\max_u \{R_{i_u} - W_u(t) + \sum_j p_{ij} w_n(j, t^u)\}. \tag{8}$$

   If $\pi^n(i,t)$ maximizes the RHS, choose $\pi^{n+1}(i,t) = \pi^n(i,t)$.

4. If $\pi^{n+1} \neq \pi^n$, set $n = n+1$ and go to step 2. Else, STOP. π^{n+1} is the optimal policy.

Under certain conditions (section 8.5.2, Puterman, 1994), one can show that this algorithm terminates in a finite number of steps.

Next we describe how a heuristic policy can be developed using just one policy improvement step. In each time slot, given the state (i, t) of the process, the aim is to compute for each user u an index (i.e. a real number depending on (i_u, t_u)) that depends solely on the current state (i_u, t_u) of that user. The heuristic scheduling policy then serves the user with the maximum index in each time slot. Such policies are referred to as Index Policies (Weber, 1992, Glazebrook et al, 2002 and Opp et al, 2004) and, as we shall see in the context of this problem, perform very well. A major contribution of this paper is the derivation of a closed form expression of the index for each user. Further, although this index will be developed under our current assumptions of Markovian evolution of the system and a constant number of users, we will see that our Index policy does not use the parameters of the Markovian structure. Hence we believe that the policy will work well even when this assumption is relaxed. The method of developing such an index policy involves choosing an "appropriate" initial policy and modifying it by a single step of policy improvement algorithm of the MDP. We discuss an appropriate initial policy in the next subsection.

3.3 Initial Policy

Consider a stationary state-independent policy that serves user u with probability p_u in any time slot. Here $p_1, \ldots, p_N$ are fixed numbers such that $p_u > 0$, $\sum_u p_u = 1$. Let

$$p = [p_1, p_2, \ldots, p_N],$$

g_p be the long run reward and

$$w_p = \{w(i, t), (i, t) \in \Omega^N X Z^N\}$$

be the bias vector for this policy satisfying the equation

$$g_p + w_p(i,t) = \sum_u p_u \left\{ R_{i_u} - W_u(t) + \sum_j p_{ij} w_p(j, t^u) \right\}. \tag{9}$$

The computation of g_p is fairly straightforward and we give the main result in theorem 3.1. This result will be used in subsection 3.5 for choosing the optimal p_u, $u = 1, \ldots, N$. Computing $w_p(i,t)$ is not simple. However, as we shall see in the next subsection, we do not need to compute $w_p(i,t)$ per se, but only the difference $w_p(j,t^u) - w_p(j,t+e)$, where $e = (1,\ldots,1) \in Z^N$.

Let $\pi^u = [\pi_1^u, \ldots, \pi_M^u]$ be the steady state distribution of the Markov Chain $\{X_u^n : n \geq 0\}$, $u = 1, 2, \ldots, N$. Then, since M is finite, it is well known (Kulkarni, 1996) that π^u exists and is the unique solution to

$$\begin{aligned} \pi^u &= \pi^u P^u \\ \sum_{m=1}^{M} \pi_m^u &= 1. \end{aligned}$$

We define

$$\phi_u(p_u) = \sum_{k=1}^{\infty} D_u(k)(1-p_u)^k p_u, \tag{10}$$

and

$$A_u = \sum_m \pi_m^u R_m. \tag{11}$$

Then, the following theorem gives an expression for g_p.

Theorem 3.1. *We have*

$$g_p = \sum_{u=1}^{N} [-\phi_u(p_u) + A_u p_u]. \tag{12}$$

Proof. Since all the users are independent, the total average net reward g_p is just the sum of the average net reward accrued to each user. We first derive the total average reward earned and then the total average cost incurred. Consider user u. In each time slot user u is served with probability

p_u. Clearly, the total average reward earned per unit time in steady state is

$$\sum_{u=1}^{N} A_u p_u. \tag{13}$$

Note that the age process $\{Y_u^n : n \geq 0\}$ is a regenerative process with $Y_u^n = 0$ as the regeneration point. Let S_u denote the length of a regenerative cycle and C_u the cost accrued in one regenerative cycle. Then, average cost is given by (Kulkarni, 1996)

$$\frac{\mathrm{E}[C_u]}{\mathrm{E}[S_u]}. \tag{14}$$

Now the randomized policy implies that

$$\Pr[S_u = k] = (1 - p_u)^{k-1} p_u,$$

for $k = 1, 2, \ldots$ and hence

$$\mathrm{E}[S_u] = \frac{1}{p_u}. \tag{15}$$

Recall that $D_u(0) = 0$. Using $C_u = \sum_{l=0}^{S_u-1} D_u(l)$, we get

$$\mathrm{E}[C_u] = \sum_{k=2}^{\infty} [D_u(1) + \ldots + D_u(k-1)](1 - p_u)^{k-1} p_u$$

which can be simplified to

$$\mathrm{E}[C_u] = \sum_{k=1}^{\infty} D_u(k)(1 - p_u)^k. \tag{16}$$

Using (10), (14), (15) and (16), the total expected starvation cost per unit time in steady state is

$$\sum_{u=1}^{N} \phi_u(p_u) \tag{17}$$

yielding (12) as the total net reward per unit time in steady state. □

Corollary 3.2. *Suppose $D_u(n) = K_u n$ for $n \geq 0$. Then $\phi_u(p_u)$ is given by*

$$\phi_u(p_u) = K_u \left(\frac{1 - p_u}{p_u} \right). \tag{18}$$

Proof. Putting $D_u(t_u) = K_u t_u$ in (10), we get

$$\phi_u(p_u) = K_u \sum_{k=1}^{\infty} k(1-p_u)^k p_u,$$

which reduces to (18), as required. □

We thus have an initial randomized scheduling policy based on the probability vector p and an expression for the long run reward per time slot for such a policy. In the next section we apply the policy improvement step (step 3 in the policy improvement algorithm described in section 3.2) once to obtain a better policy. We see later for the examples considered in section 5 that even this one-step improved policy, for appropriate values of the parameters K_u, gives us throughput that is close to the maximum possible throughput.

3.4 Policy Improvement Step

For a given (i,t) the policy improvement step seeks to maximize

$$R_{i_u} - W_u(t) + \sum_j p_{ij} w_p(j, t^u)$$

over all $u \in \{1, 2, \ldots, N\}$. This is equivalent to maximizing

$$I_u(i,t) = R_{i_u} - W_u(t) + \sum_l D_l(t_l) + \sum_j p_{ij}\left[w_p(j,t^u) - w_p(j,t+e)\right], \tag{19}$$

over all $u \in \{1, 2, \ldots, N\}$ since for a given (i,t), the additional term $\sum_l D_l(t_l) - \sum_j p_{ij} w_p(j, t+e)$ does not depend on u. The improved policy then serves the user with the highest index $I_u(i,t)$. To compute $I_u(i,t)$, we need an expression for

$$w_p(j, t^u) - w_p(j, t+e), \tag{20}$$

which we derive in the next theorem.

Theorem 3.3. *Given the initial policy p and the reward structure in section 3.1, $I_u(i,t)$ is given* *by*

$$I_u(i,t) = R_{i_u} + D_u(t_u) + \sum_{k=1}^{\infty}(1-p_u)^{k-1}[D_u(t_u+k+1) - D_u(k)]. \tag{21}$$

Proof. We only need an expression for the difference in biases given in (20). Suppose we follow the randomized policy and consider two sample paths of the $\{(X^n, Y^n), n \geq 0\}$ process under this policy denoted by $\{(X^{n,m}, Y^{n,m}), n \geq 0\}$ for $m = 1, 2$. We assume that $Y^{0,1} = t^u$, $Y^{0,2} = t + e$ and

$$X^{n,1} = X^{n,2} \text{ for } n \geq 0. \tag{22}$$

Let $v^m(n)$ be the user served in slot n sample path m. We assume that

$$v^1(n) = v^2(n) \text{ for } n \geq 0. \tag{23}$$

Equations (22) and (23) state the manner in which the two sample paths are coupled. Now fix a $u \in \{1, 2, \ldots, N\}$ and assume the user u has been served at time 0. Let S denote the time when we next serve the user u, i.e. $S = \min\{n > 0 : v^1(n) = u\}$. Then recall that $Y_v^{n,m}$ denotes the age of user v in time slot n along the sample path $m \in \{1, 2\}$ and we have for $n \leq S$

$$\begin{aligned} Y_v^{n,m} &= t_v + 1 + n \text{ for } v \neq u, m = 1, 2, \\ Y_u^{n,1} &= n, \\ Y_u^{n,2} &= t_u + 1 + n \end{aligned} \tag{24}$$

and for $n > S$

$$Y^{n,1} = Y^{n,2}. \tag{25}$$

From (22), there is no difference in the rewards accrued by the two sample paths. Let C_n^m, $m = 1, 2$ be the cost incurred by the path m in time slot n, and $C_n = C_n^1 - C_n^2$ be their difference. Then (24), (25) and

$$C_n = \sum_{v=1}^{N} \left[D_v(Y_v^{n,1}) - D_v(Y_v^{n,2})\right], \tag{26}$$

implies that

$$C_n = \begin{cases} D_u(t_u + n + 1) - D_u(n) & \text{for } n \leq S \\ \\ 0 & \text{for } n > S. \end{cases} \tag{27}$$

Hence, given $S = k$,

$$w_p(j, t^u) - w_p(j, t + e) = \sum_{n=1}^{k} C_n$$

Under the policy p we have,

$$Pr(S = k) = p_u(1 - p_u)^{k-1}, \text{ for } k \geq 1. \tag{28}$$

Then, we have

$$w_p(j, t^u) - w_p(j, t + e) =$$
$$\sum_{k=1}^{\infty} p_u(1 - p_u)^{k-1} \left[\{D_u(t_u + 2) + \ldots + D_u(t_u + k + 1)\} - \{D_u(1) + \ldots + D_u(k)\}\right].$$

Rearranging and simplifying, we get

$$w_p(j, t^u) - w_p(j, t + e) = \sum_{k=1}^{\infty} (1 - p_u)^{k-1} [D_u(t_u + k + 1) - D_u(k)]. \tag{29}$$

Further, using $W_u(t) = \sum_{l \neq u} D_l(t_l) = \sum_l D_l(t_l) - D_u(t_u)$, and (29), $I_u(i, t)$ is given by

$$R_{i_u} + D_u(t_u) + \sum_{k=1}^{\infty} (1 - p_u)^{k-1} [D_u(t_u + k + 1) - D_u(k)],$$

as required. □

Starting with the randomized policy the policy improvement step then yields the policy that chooses action u in state (i, t) that maximizes $\{I_u(i, t), u = 1, 2, \ldots, N\}$. Thus the improved policy is an index policy. Specifically, we shall study the index policy with linear incremental cost, i.e. $D_u(t_u) = K_u t_u$. The index for this policy is particularly simple and is given by the next corollary.

Corollary 3.4. *Suppose $D_u(n) = K_u n$ for $n \geq 0$. Then the index is given by*

$$I_u(i, t) = R_{i_u} + K_u t_u (1 + \frac{1}{p_u}) + \frac{K_u}{p_u}. \tag{30}$$

Proof. Putting $D_u(n) = K_u n$ in (21), we get

$$I_u(i,t) = R_{i_u} + K_u t_u + K_u(t_u+1)\sum_{k=1}^{\infty}(1-p_u)^{k-1},$$

which simplifies to (30), as required. □

An index policy based on the index (30) is called a Linear Index Policy (LIP). Clearly, the initial policy p is arbitrary. This immediately begs the question: what p should one use? There are two possible answers:

1. Choose $p_u = \frac{1}{N}$ for all u. In this case,

$$I_u(i,t) = R_{i_u} + K_u t_u(N+1) + K_u N. \tag{31}$$

   We call this policy Uniform Linear Index Policy (ULIP).

2. Choose the p that maximizes the long run average reward g_p of the policy. We discuss the methods of doing that in the next section and call the resulting policy Optimal Linear Index Policy (OLIP).

3.5 Optimizing the Average Reward

In this section we see how an optimal value of p that maximizes g_p can be obtained. We describe a Lagrangian based algorithm to solve the following problem:

$$\text{maximize } g_p$$

$$\text{subject to } \sum_{u=1}^{N} p_u = 1.$$

Define

$$L_P = \sum_{u=1}^{N} -\phi_u(p_u) + A_u p_u + \theta(1 - \sum p_u),$$

where θ is the Lagrangian multiplier. Differentiate w.r.t. p_u $(u = 1, 2, \dots, N)$ to get

$$-\phi_u^{'}(p_u) + A_u = \theta. \tag{32}$$

It is easy to see that $\phi_u(\cdot)$ is decreasing and $\phi_u^{'}(\cdot)$ is an increasing function of p_u and that

$$\phi_u^{'}(p_u) = -\sum_{k=0}^{\infty}(k+1)(1-p_u)^k\{D_u(k+1) - D_u(k)\}.$$

In our specific case of linear incremental penalty, i.e., $D_u(n) = K_u n$, $\phi_u(p_u) = K_u(1-p_u)/p_u$ from (18), we get

$$\phi_u^{'}(p_u) = \frac{K_u}{p_u^2}.$$

Using this, (32) yields

$$p_u(\theta) = \sqrt{\frac{K_u}{\theta - A_u}} \tag{33}$$

where p_u is written as $p_u(\theta)$ to emphasize its dependence on θ. But $\sum_{u=1}^{N} p_u(\theta) = 1$. So the optimal $p_u(\theta)$ subject to the given constraints can be obtained by solving for the optimal θ (call it θ^*) from

$$F(\theta) = \sum_{u=1}^{N}\sqrt{\frac{K_u}{\theta - A_u}} = 1 \tag{34}$$

and putting $\theta = \theta^*$ in (33). It is clear that $F(\theta)$ is a decreasing function of θ and we can solve (34) by bisection methods (Burden and Faires, 2000) if we can identify a θ_l and θ_h such that $F(\theta_l) > 1$ and $F(\theta_h) < 1$. The following theorem gives such θ_l and θ_h.

Theorem 3.5. *Let*

$$\theta_l = max_u(K_u + A_u) \tag{35}$$

and

$$\theta_h = max_u(K_u N^2 + A_u). \tag{36}$$

Then, assuming there are atleast two users u_1, u_2 such that $K_{u_1} > 0, K_{u_2} > 0$, $F(\theta_l) > 1$ and $F(\theta_h) < 1$.

Proof. Let

$$v = \text{argmax}(K_u + A_u). \tag{37}$$

Then $p_v(\theta_l) = 1$, $p_u(\theta) > 0$ for u satisfying $K_u{>}0$ and therefore $F(\theta_l){>}1$ since $K_u{>}0$ for at least one $u \neq v$. Now from (36), for each u we have

$$\sqrt{\frac{K_u}{\theta_h - A_u}} < \frac{1}{N} \tag{38}$$

yielding $F(\theta_h) < 1$, as required. □

In the next section, we describe the performance analysis methodology used in the paper.

4 Performance Analysis of LIP and PFA

As mentioned in the introduction the allocation algorithm strives to find the balance between high throughput and low starvation age. Thus we can evaluate the performance of an allocation algorithm by studying the following performance measures:

$$\begin{aligned} B &= \text{long run expected throughput per time slot,} \\ \zeta &= \text{long run expected starvation age of a user,} \\ \rho_d &= \text{long run probability that a user is starved for longer than } d \text{ time slots.} \end{aligned}$$

Having high throughput is important for the service provider to ensure full utilization of the existing infrastructure and maximize profits. It is easy to see that both the long run expected starvation age ζ and the long run probability of a user starving more than a predetermined number of slots ρ_d are measures of the consistency of service, or fairness to users, and are directly related to customer satisfaction. Therefore both ζ and ρ_d are measures of Quality of Service (QoS) to individual users. In this and the next section "QoS level" refers to the (actual or estimated) value of either ζ or ρ_d. We compare the LIP and PFA by plotting throughput against these two QoS levels for each of the two algorithms. Using the probability of delay of packets, that we are estimating by ρ_d, as a QoS has been widely prevalent in the literature (Shakkottai (2003) and Eryilmaz (2005)).

In this section we study these three performance measures for the LIP and PFA algorithms. The mean throughput B is clearly controlled using the parameter $\tau \in [0, 1]$ in the case of PFA and $K_u \in [0, \infty)$ in the case of LIP. As an illustration of the effect of τ and K_u on the throughput, we present three theorems corresponding to the extreme values of these parameters. For this section and the next, we assume that all the users have the same PTM P with limiting distribution $\pi = [\pi_1, \dots \pi_M]$, and that $\{X^n : n \geq 0\}$ is aperiodic. Furthermore, we assume that $K_u = K$ for all $u \in 1, 2, \dots, N$ yielding $p_u = \frac{1}{N}$ for OLIP, since in this setting, using (33), $p_u = \frac{1}{N}$ optimizes the average reward of the randomized policy. Thus under our assumptions of the user processes and user parameters LIP is the same as ULIP and OLIP.

Theorem 4.1. *Let $\tau \to 1$ in PFA and $K \to \infty$ in LIP. Then, in the limit, both PFA and LIP converge to the round-robin service rule, i.e., every user is served once every N slots and the limiting throughput is*

$$B = \sum_{k=1}^{M} \pi_k R_k. \tag{39}$$

Furthermore, in this limiting regime,

$$\begin{aligned} \zeta &= \frac{N-1}{2}, \\ \rho_d &= \begin{cases} 0 & \text{if} \quad d > N-1 \\ \frac{N-1-d}{N} & \text{if} \quad 0 \leq d \leq N-1. \end{cases} \end{aligned} \tag{40}$$

Proof. First, consider LIP with $K \to \infty$. Then, from (31), $I_u(i, t) = R_{i_u} + K t_u (1+N)$. As $K \to \infty$, the R_{i_u} term becomes insignificant. For sufficiently large K, an equivalent index is $K(1 + N)t_u$, i.e. the LIP serves the user with the lowest age. This leads to the round-robin service rule.

Next, setting $\tau = 1 - h$ and letting $h \to 0$ in (1) and (2) yields

$$\frac{R_u^n}{Q_u(n)} = \begin{cases} \frac{R_u^n}{h^{(n-Y_u^n)} R_u^{(n-Y_u^n)}} & \text{if } u \text{ not served at } n-1 \\ \frac{R_u^n}{R_u^{n-1}} & \text{if } u \text{ served at } n-1. \end{cases} \tag{41}$$

Clearly the quantity $R_u^n/Q_u(n)$ for the user served in slot $n-1$ remains finite in slot n, but that for other users (not served in slot $n-1$) can become arbitrarily large (as $h \to 0$) in slot n. Thus in the limit as $h \to 0$, the PFA doesn't serve the same user in two consecutive slots. Furthermore, for sufficiently small h, the quantities $\frac{R_u^n}{Q_u(n)}$ in (41) are in the same order as the ages Y_u^n for $u \in \{1, 2, \ldots, N\}$. Thus, PFA too serves the user with the lowest age yielding the round-robin service rule.

Now, since all users have the same PTM P, $R_{v(n)}^n = R_k$ with probability π_k for $k \in \{1, 2, \ldots N\}$ as $n \to \infty$. Further, since the state space of X^n is finite, $\{R_{v(n)}^n, n \geq 0\}$ is ergodic, we get (39) (Norris, 1997), as desired.

Finally, as a consequence of the round-robin rule, in any given slot n, the age vector for all the N users is either $(0, 1, \ldots, N-1)$ or a permutation of this sequence. Therefore,

$$\zeta = \frac{1}{N}[1 + 2 + \ldots + N - 1] = \frac{N-1}{2}.$$

Similarly in any time slot, there is no user with age greater than d for $d > N-1$ and the number of users with age greater than d for $0 \leq d \leq N-1$ is equal to $N-1-d$, yielding (40), as required. □

The maximum possible throughput is achieved by using what we call the "myopic policy" that serves user $v(n) = \text{argmax}_u R_u^n$ in time slot n. The next theorem gives an expression for the throughput achieved by the myopic policy. For the sake of notational convenience, let

$$\begin{aligned} \alpha_k &= \pi_1 + \pi_2 + \ldots + \pi_k, \text{ and} \\ q_k &= (\alpha_k)^N - (\alpha_{k-1})^N \end{aligned}$$

Theorem 4.2. *Let $\tau \to 0$ in PFA and $K \to 0$ in LIP. Then both PFA and LIP converge to the myopic policy and the throughput of the myopic policy is given by*

$$B = \sum_{k=1}^{M} q_k R_k. \tag{42}$$

Proof. $K \to 0$ in (31) immediately shows that the LIP reduces to the myopic policy. Similarly, letting $\tau \to 0$ in (2) shows that the PFA approaches the myopic policy.

We know that for $n \to \infty$, $Pr[X_u^n = k] = \pi_k$, $1 \le k \le M$, for $u \in \{1, \ldots, N\}$. Let

$$U^n = \max\{X_1^n, X_2^n, \ldots, X_N^n\}.$$

Then,

$$Pr[U^n \le k] = Pr[X_1^n \le k] Pr[X_2^n \le k] \ldots Pr[X_N^n \le k] = (\alpha_k)^N.$$

Therefore,

$$q_k = Pr[U^n = k] = Pr[U^n \le k] - Pr[U^n \le k-1] = (\alpha_k)^N - (\alpha_{k-1})^N.$$

Clearly,

$$B = \sum_{k=1}^{M} \lim_{n\to\infty} Pr[U^n = k] R_k,$$

yielding

$$B = \sum_{k=1}^{M} q_k R_k, \tag{43}$$

as required. □

In section 5.2, we also consider the *dynamic population* case where the users arrive in the cell according to a Poisson process with rate λ and remain in the cell for a generally distributed time with mean a. Let $N(t)$ be the number of users in a dynamic cell at time t. Then $\{N(t), t \ge 0\}$ is the number of customers in an $M/G/\infty$ queue and the LIP index of (31) is modified for this case as follows:

$$I(u,t) = R_{i_u} + Kt_u(N(t)+1). \tag{44}$$

Using the results for the $M/G/\infty$ queue (Kulkarni, 1996) we see that the number of users in steady state is a Poisson random variable with parameter λa. The following theorem gives an expression for the myopic policy throughput in this setting. We need the following notation:

$$s_k = e^{-\lambda a(1-\alpha_k)} - e^{-\lambda a(1-\alpha_{k-1})}.$$

Theorem 4.3. *The throughput of the myopic policy in the dynamic population case is given by*

$$B = \sum_{k=1}^{M} s_k R_k. \tag{45}$$

Proof. Let N represent the number of users in the cell in steady state. It is known that N is a Poisson random variable with parameter λa. The myopic policy chooses the user to be served in a given time slot independent of the starvation age. Therefore, from (43),

$$\mathrm{E}[\text{Throughput}|N=n] = \sum_{k=1}^{M} \left[(\alpha_k)^n - (\alpha_{k-1})^n\right] R_k.$$

Since $B = \mathrm{E}[\mathrm{E}[\text{Throughput}|N]]$, we have

$$B = \sum_{k=1}^{M} \left(\mathrm{E}\left[(\alpha_k)^N\right] - \mathrm{E}\left[(\alpha_{k-1})^N\right]\right) R_k,$$

which yields (45) using $\mathrm{E}\left[z^N\right] = e^{-\lambda a(1-z)}$ from standard probability theory (Gross and Harris, 1985). □

The performance of LIP can be compared with that of PFA using plots of B vs ζ and B vs ρ_d for both the policies as τ and K vary. Since analytical results are not available except for the extreme values of these parameters we resort to simulation to estimate these quantities. This is done in the next section.

5 Simulation Results

We now use simulation to estimate B, ζ and ρ_d for the LIP and PFA for a given K and τ. We begin by formulating the estimators for these parameters below.

5.1 The Estimators

Let L be the number of independent sample paths simulated and T be the number of slots in each path. Let $R_{v(n)}^{n,l}$ be the throughput in the n^{th} slot of the l^{th} sample path, $l = 1, 2, \ldots, L$ and $n = 1, 2, \ldots, T$. Then the estimator $\hat{B}$ of B is given by

$$\hat{B} = \frac{1}{L} \sum_{l=1}^{L} \left(\frac{1}{T} \sum_{n=1}^{T} R_{v(n)}^{n,l} \right). \tag{46}$$

Let $Y_u^{n,l}$ be the age of the user u in the n^{th} slot in sample path l, $l = 1, 2, \ldots, L; n = 1, 2, \ldots, T$. Then we define the estimator $\hat{\zeta}$ of ζ as

$$\hat{\zeta} = \frac{1}{L} \sum_{l=1}^{L} \left[\frac{1}{T} \sum_{n=1}^{T} \left(\frac{1}{N} \sum_{u=1}^{N} Y_u^{n,l} \right) \right]. \tag{47}$$

Similarly the estimator $\hat{\rho_d}$ of ρ_d is given by

$$\hat{\rho_d} = \frac{1}{L} \sum_{l=1}^{L} \left[\frac{1}{T} \sum_{n=1}^{T} \left(\frac{1}{N} \sum_{u=1}^{N} \mathbf{1}\{Y_u^{n,l} > d\} \right) \right], \tag{48}$$

where

$$\mathbf{1}\{Y_u^{n,l} > d\} = \begin{cases} 1 & \text{if } Y_u^{n,l} > d \\ 0 & \text{otherwise.} \end{cases} \tag{49}$$

In the *dynamic population* case estimators are obtained in a similar manner. The estimator $\bar{B}$ for B is still defined by the right hand side of equation (46). The estimators $\bar{\zeta}$ of ζ and $\bar{\rho_d}$ of ρ_d however need to be modified to replace N with $N(t)$ $(t = 1, 2, \ldots, T)$ and at the same time exclude contribution from the time slots for which $N(t) = 0$. Thus the estimates in the *dynamic population* case are given by

$$
\begin{aligned}
\bar{B} &= \frac{1}{L}\sum_{l=1}^{L}\left(\frac{1}{T}\sum_{n=1}^{T} R_{v(n)}^{n,l}\right), \\
\bar{\zeta} &= \frac{1}{L}\sum_{l=1}^{L}\left[\frac{1}{T}\sum_{t=1,N(t)>0}^{T}\left(\frac{1}{N(t)}\sum_{u=1}^{N(t)} Y_u^{t,l}\right)\right], \\
\bar{\rho_d} &= \frac{1}{L}\sum_{l=1}^{L}\left[\frac{1}{T}\sum_{t=1,N(t)>0}^{T}\left(\frac{1}{N(t)}\sum_{u=1}^{N(t)} \mathbf{1}\{Y_u^{t,l} > d\}\right)\right].
\end{aligned}
$$

It is worth noting that all these estimators are for long-run performance measures, and so we need to collect samples only from the stationary region of the Markov Chain $\{X^n, n \geq 0\}$ (and of $\{N(t), t \geq 0\}$ in the dynamic population case). To ensure this, both in the constant and dynamic population case, we start the simulation in the stationary distribution of the $\{X^n, n \geq 0\}$ and the $\{N(t), t \geq 0\}$ process. We plot estimate of B vs estimates of ζ and ρ_d for both the policies, and see that for any given QoS level, the LIP produces higher throughput.

5.2 Simulation Parameters

We use the following set of available data rates (kbps) (Bender et al, [7]): R ={38.4, 76.8, 102.6, 153.6, 204.8, 307.2, 614.4, 921.6, 1228.8, 1843.2, 2457.6}. Thus, each Markov Chain $\{X_u^n, n \geq 0\}$ has $M = 11$ states. We use $d = 100$, $T = 10^5$ and

$$
P = \alpha I + \frac{1-\alpha}{M-1}(D - I),
$$

where I is an MXM identity matrix and D is an MXM matrix with all entries equal to 1. This implies that the Markov Chain stays in a given state for *Geometric*$(1-\alpha)$ number of time slots and then moves to one of the remaining $M-1$ states with equal probability. Note that P is doubly stochastic, and hence $\pi_k = 1/M = 1/11$. Further, the length of a time slot is $1.67 * 10^{-3}$ seconds (Jalali, 2002). We choose $\alpha = 0.9999$ implying that on the average the Markov Chain stays in one state for 10^4 time slots, i.e. 16.7 seconds, before changing states. To get the required plots we vary K for LIP and τ for the PFA. The actual ranges are mentioned in the plots.

5.3 Constant Number of Users

We take samples from $L = 100$ sample paths of the Markov Chain. We conduct the simulation for various values of N and display the plots of $\hat{B}$ vs $\hat{\zeta}$ in figures 1(a) and 1(b). We obtain the plots by running the LIP and PFA algorithms for a range of values of K (between 0.01 and 50) and τ (between 0.5 and 1) respectively. Each point on the LIP and PFA plot corresponds to a unique value of K and τ respectively. Both K and τ decrease as we move from left to right on the curves. As expected, $\hat{\zeta}$ and $\hat{B}$ decrease with τ and K. The points with the minimum $\hat{\zeta}$ correspond to $K = 50$ and $\tau = 0.5$, with both the curves approaching the throughput for the round-robin policy (calculated to be 722.62 using (39)) as proved in theorem 4.1. Similarly, the points with the maximum $\hat{\zeta}$ correspond to $K = 0.01$ and $\tau = 0.0001$ with both LIP and PFA converging to the throughput for the myopic policy (using (42), calculated to be 2121.31 for $N = 10$ and 2452.34 for $N = 50$) as proved in theorem 4.2. As is clear from figure 1, LIP outperforms PFA significantly over the entire range of $\hat{\zeta}$. Thus no matter what QoS level we choose for the users, using LIP always results in better system throughput. Also PFA throughput converges to the myopic policy throughput (this is the maximum throughput achievable) at a much slower rate than the LIP throughput.

Similarly figure 2 shows the plots of $\hat{B}$ vs $\hat{\rho}_d$ for different values of N. As in the plots of figure 1, each point on the LIP and PFA curves of the plots of figure 2 represents a unique value of K and τ respectively. Both K and τ increase as we move from left to right on the curves. We see in both figures 2(a) and 2(b) that for any given value of $\hat{\rho}_d$, the LIP throughput is significantly higher than that of PFA. Thus LIP demonstrates better performance at all QoS levels. Further, even at very high levels of $\hat{\rho}_d$, the PFA throughput does not approach the myopic policy throughput (which, as noted above, is the maximal throughput). On the other hand, using LIP gives us an option of achieving a throughput close to the myopic policy throughput for a high QoS level.

The substantial domination of the LIP plot for both the OoS measures clearly demonstrates the superiority of LIP over PFA. Next, we consider the case when users can enter or leave the cell.

5.4 Poisson Arrival of Users

In this section, we assume users arrive according to a poisson process with rate λ. Once in the cell, sojourn time of a user is exponentially distributed with mean a. Thus, in steady state the number of users is a Poisson random variable with mean $N_{\text{avg}} = \lambda a$. The LIP index is given by (44). We then serve the user with the maximum value of index $I(u,t)$. For PFA, we initialize Q_u to 1 for a newly arriving user and $R_u^n/Q_u(t)$ is maximized among all the $N(t)$ users present in the cell at the beginning of the time slot t. Again, we consider $L = 100$ sample paths of the process $\{(X^t, Y^t), t \geq 0\}$ with T time slots in each sample path. We observe that in this setting, since users also leave the system, the exponentially filtered data transfer rate $Q_u(\cdot)$ in (2) does not always smoothen out before their departure. As a result, the performance (neither $\bar{B}$ nor $\bar{\zeta}$, $\bar{\rho}_d$) of PFA is not monotonic in τ.

Considering the mean age measure of QoS first, we show in figure 3 plots of $\bar{B}$ vs $\bar{\zeta}$. As in subsection 5.3, the bottom left point corresponds to $\tau = 1$ and $K = 50$ approaching the round-robin policy (theorem 4.1), and the top right point corresponds to $\tau = 0.0001$ and $K = 0.01$ approaching the myopic policy (theorem 4.2). The theoretical value computed for the round-robin policy remains the same at 722.62. The values of the myopic policy throughput, however, are now given by theorem 4.3, and we compute them to be 2078.31 for $N_{\text{avg}} = 10$ and 2451.01 for $N_{\text{avg}} = 50$. For any given level of the QoS (any value of $\bar{\zeta}$), the LIP throughput is considerably more than the

PFA throughput. It is also worth noting that with PFA we can never achieve QoS better than a certain level, while LIP gives us the option to serve at all QoS levels.

Next we consider ρ_d as the QoS measure and plot $\bar{B}$ vs $\bar{\rho_d}$ for various combinations of N_{avg} in figure 4. On both the LIP and PFA curves, K and τ vary as in the plots of figure 3. Across the entire achievable range of QoS, LIP throughput is much better than the PFA throughput.

The graphs corresponding to both the QoS measures demonstrate the significantly better performance of the LIP algorithm. Another algorithm for resource scheduling in the setting of this paper is suggested in Liu, Chong and Shroff (2001). However for the identical users case considered in this section their algorithm reduces to the myopic policy. Hence we do not need to study it separately.

6 Conclusion

In this paper we have used the MDP formulation and the policy improvement approach to develop an Index policy for the data transfer problem in wireless telecommunication networks. The index is a simple, intuitive, closed form expression and we have demonstrated its superior performance over the existing PFA over a wide parameter space. Future extensions of this work can include developing effective policies under the setting of a finite queue of data for each user and a non-Markovian evolution of the state of the users.

Acknowledgements

The authors would like to thank Professor Kevin Glazebrook of Lancaster University Management School and Dr. Siamak Sorooshyari of Alcatel-Lucent for discussions and valuable feedback that helped improve the quality of this paper.

REFERENCES

Andrews, M. A survey of scheduling theory in wireless data networks. 1999. *IMA Volumes in Mathematics and its applications* **143**, 1-18.

Bender, P., P. Black, M. Grob, R. Padovani, N. Sindhushayana, and A. Viterbi. 2000. CDMA/HDR: A bandwidth efficient high speed data service for nomadic users. *IEEE Communications Magazine*, July 2000.

Burden, R., and Faires, D. 2000. *Numerical Analysis.* Brooks/Cole.

Eryilmaz, A. 2005. Efficient and Fair Scheduling for Wireless Networks. *Ph.D. Thesis.*

Georgiadis, L., Neely, M., and Tassiulas, L. 2006. Resource allocation and cross-layer Control in wireless networks. *Foundations and Trends in Networking* **1**.

Glazebrook, K., J. Nino-Mora, and P.S. Ansell. 2002. Index policies for a class of discounted restless bandits. *Advances in Applied Probability* **34**, 754-774.

Gross, D., and C. Harris. 1985. *Fundamentals of Queueing Theory.* John Wiley & Sons, Inc.

Jalali, A., R. Padovani, and R. Pankaj. 2000. Data Throughput of CDMA-HDR a high efficiency-high data rate personal communication wireless system. *Proceedings of the IEEE Semiannual Vehicular Technology Conference*, 1854-1858.

Kulkarni, V. 1996. *Modeling and Analysis of Stochastic Systems.* Chapman & Hall.

Liu, X., Chong, E., and Shroff, N. 2001. Opportunistic Transmission Scheduling With Resource-Sharing Constraints in Wireless Networks. *IEEE Journal on Selected Areas in Communications* **19**, 2053-2064.

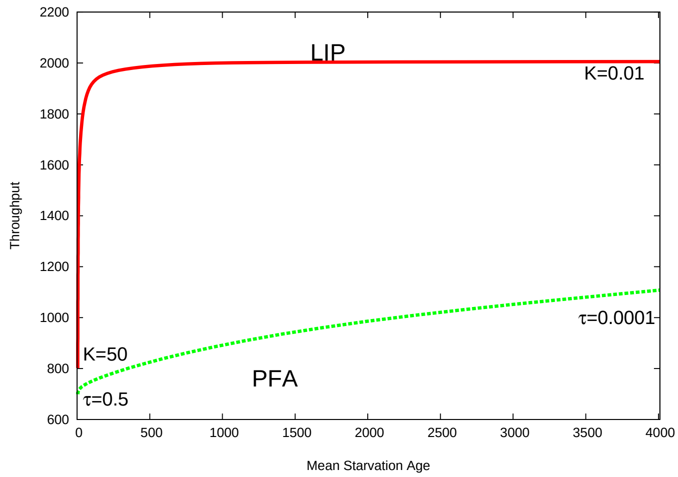
Norris, J. 1997. *Markov Chains.* Cambridge University Press.

Puterman, M. 1994. *Markov Decision Processes - Discrete Stochastic Dynamic Programming.* John Wiley & Sons, Inc. 446-447.

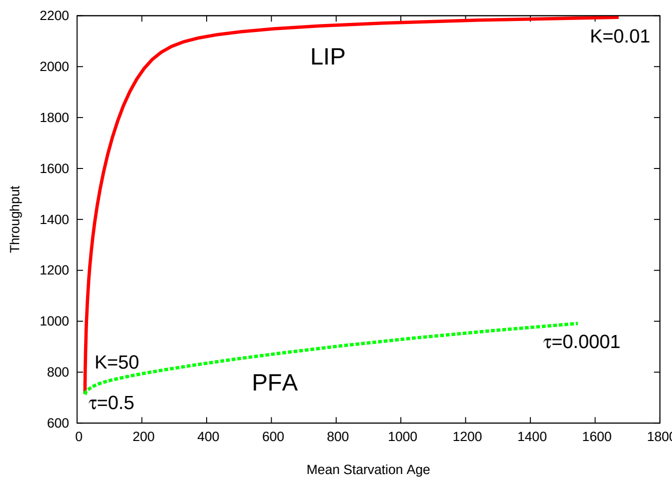
Opp, M., K. Glazebrook, and V. Kulkarni. 2004. Outsourcing Warranty Repairs: Dynamic Allocation. *NRLQ* **52**, 381-398.

Shakkottai, S. 2003. Modes of overflow, effective capacity and QoS for wireless scheduling. *ISIT*, p. 334.

Weber, R. On the Gittins Index for Multiarmed Bandits. 1992. *The Annals of Applied Probability* **2**, 1024-1033.

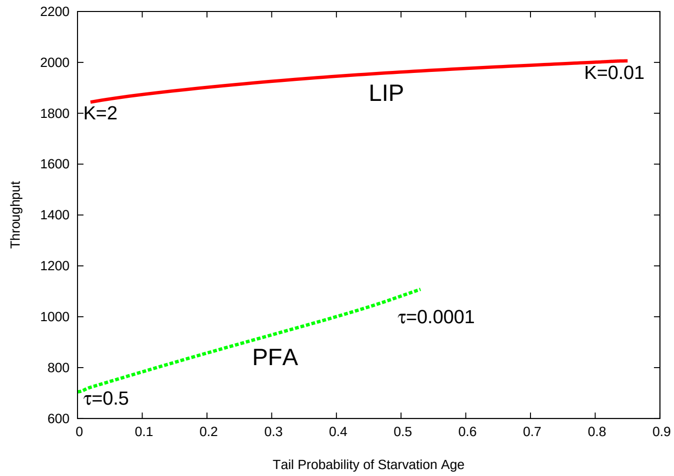


(a) N = 10

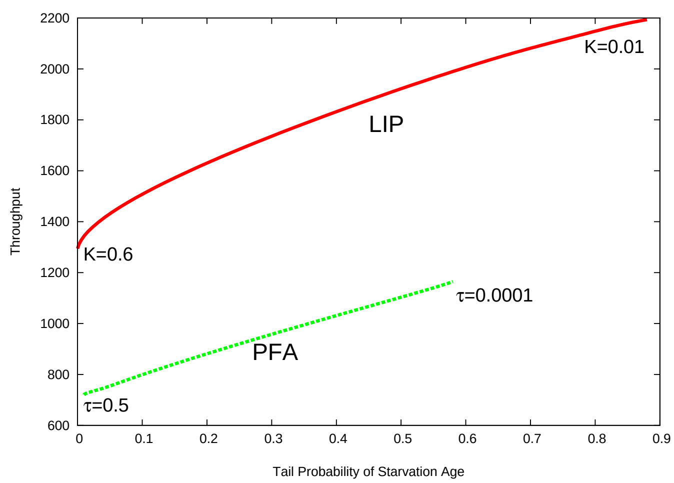


(b) N = 50

Figure 1: Throughput Vs Mean Starvation Age: Constant N. Each point on the LIP plot corresponds to a value of K and on the PFA plot to a value of τ. K varies from 0.01 to 50, with the throughput for $K = 0.01$ matching the myopic policy throughput closely. τ varies from 0.0001 to 0.5.

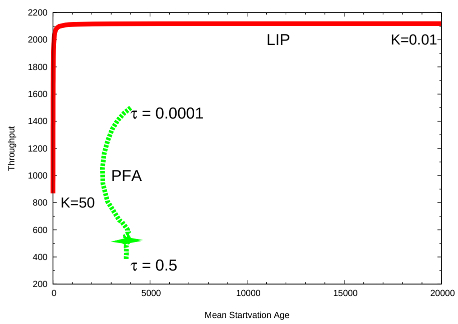


(a) N = 10

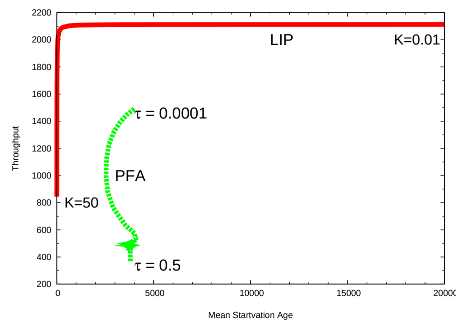


(b) N = 50

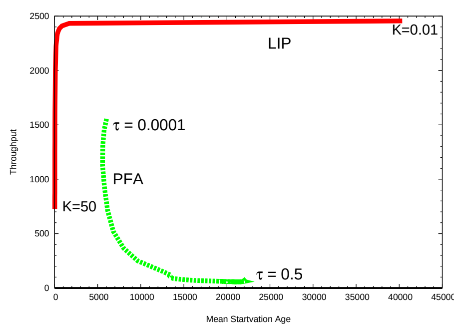
Figure 2: Throughput Vs Probability of starvation for greater than d slots for LIP and PFA: Constant N. Each point on the LIP plot corresponds to a value of K and on the PFA plot to a value of τ. K varies from 0.01 to 2 for $N = 10$, and from 0.01 to 0.6 for $N = 50$. For bigger values of K we have $\hat{\rho}_d = 0$, so we do not plot those points here. The throughput for $K = 0.01$ matches the myopic policy throughput closely in all cases. τ varies from 0.0001 to 0.01. For bigger values of τ we have $\hat{\rho}_d = 0$, so we do not plot the corresponding points here.



(a) $N_{\text{avg}} = 10$, $a = 10$

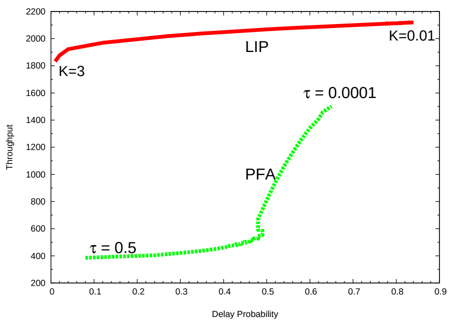


(b) $N_{\text{avg}} = 10$, $a = 15$

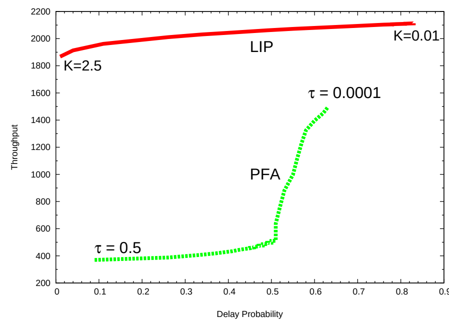


(c) $N_{\text{avg}} = 50$, $a = 10$

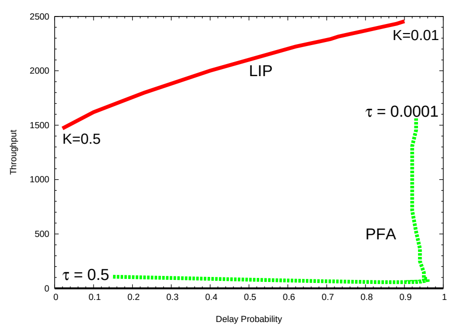
Figure 3: Throughput Vs Mean Starvation Age for varying N. Each point on the LIP plot corresponds to a value of K and on the PFA plot to a value of τ. K varies from 0.01 to 50, with the throughput for $K = 0.01$ matching the myopic policy throughput closely. τ varies from 0.0001 to 0.5.



(a) $N_{\rm avg} = 10,\ a = 10$



(b) $N_{\rm avg} = 10,\ a = 15$



(c) $N_{\rm avg} = 50,\ a = 10$

Figure 4: Throughput Vs $\bar{\rho}_d$ for varying N. τ varies from 0.0001 to 0.5. K varies from 0.01 to 3 in 4(a), from 0.01 to 2.5 in 4(b), and from 0.01 to 0.5 in 4(c). We have omitted all points corresponding to bigger values of K because $\bar{\rho}_d = 0$ for those cases.