# Incentive Mechanisms for Smoothing Out A Focused Demand for Network Resources

Kevin Leyton-Brown, Ryan Porter, Balaji Prabhakar, Yoav Shoham, Shobha Venkataraman

*Computer Science Department, Stanford University, Stanford CA 94305*
*{kevinlb;rwporter;balaji;shoham;shobha}@cs.stanford.edu*

**Abstract**

We explore the problem of sharing network resources when users' preferences lead to temporally concentrated loads, resulting in an inefficient use of the network. In such cases external incentives can be supplied to smooth out demand, obviating the need for expensive technological mechanisms. Taking a game-theoretic approach, we consider a setting in which bandwidth or access to service is available during different time slots at a fixed cost, but all agents have a natural preference for choosing the same time slot. We present four mechanisms that motivate users to distribute the load by probabilistically waiving the cost for each time slot, and analyze the equilibria that arise under these mechanisms. [1]

## 1 Introduction

Competition for network resources is intrinsic to a network's operation and leads to congestion. Since users access resources in a distributed and uncoordinated fashion, it is common for a network to experience congestion even when the average demand for a resource is much less than its capacity. Some of these congestion epochs are simply a product of the statistical nature of user access patterns and traffic types, and are thus unpredictable. To cope with this lack of coordination among users and the unpredictability of congestion epochs, networks send "congestion signals" to users to help them share its resources in a fair and satisfactory fashion. For example, packets at a congested router may be either dropped or marked [5].

[1] This work was supported by DARPA grant F30602-00-2-0598 and by a Stanford Graduate Fellowship.

A great deal of network congestion is not only caused by a lack of coordination, but also by users who aim to selfishly maximize the bandwidth available to them (see Shenker [18]). There exists a substantial body of work on the fair management of this sort of congestion in networks. In particular, the problem of designing congestion control and pricing mechanisms to provide differentiated qualities-of-service (QoS) in the Internet has received a lot of attention recently. The first common type of solution to this problem is technological: the network can erect “bandwidth firewalls” between packet flows using scheduling algorithms like Weighted Fair Queuing [3]. Such scheduling algorithms decrease or eliminate the dependence of one flow’s QoS from the QoS of other flows. They can be difficult to implement in high-capacity routers, however, as they require the maintenance of per-flow state to distinguish, buffer and schedule the packets of individual flows. This has led researchers to explore trading off performance for simplicity of implementation, yielding router mechanisms that provide approximate fairness [6,17,16].

An alternate line of research takes an economic approach to congestion management. Following this approach the network attempts to induce users to condition their flows; this avoids the implementation complexity inherent in erecting explicit bandwidth firewalls. Using ideas from economics, MacKie-Mason and Varian [12] argued that this incentive can be provided by charging agents for the damage caused to others by their ill-conditioned flows. This work proposes a “smart market” that uses bids to set a price for network usage at each of several time slots. Gibbens and Kelly [8] suggest charging a user for the role its packets play in causing congestion; see also [9] and [10]. Odlyzko [14] proposes “Paris Metro Pricing”: partitions of the network that behave identically but charge different prices, inviting users to choose the partition they believe will offer the best balance of cost and congestion.

In some situations, times of high demand are regular and predictable. Such *focused loading* can occur because many users’ utility functions are maximized by using the network at some specific time. For example, early studies of long-distance telephone networks show a spike in usage when rates drop [13]. Figure 1 is a representative graph adapted from p. 450 of this paper, which shows telephone network traffic versus time of day. Note that usage falls off before the 1 PM rate drop, spikes afterwards and then falls off again. A recent study [1] considers dial-up data traffic in Ireland and the UK— where ISPs provide free Internet access but users pay for the duration of their phone connections— where a focused load on the telephone network occurs from an increase in data connections when phone charges drop. Web servers also experience focused loading just before deadlines, or just after new content or services are made available. While these times are known well in advance, users have no incentive to avoid accessing the web site close to the deadline and thus can cause server overloads or crashes, to which system managers typically respond by buying more resources.

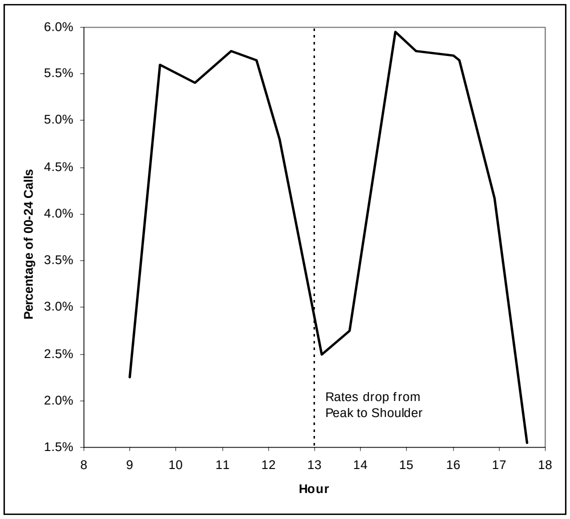


Fig. 1. Quarterly Trunk Calls on Weekdays in the United Kingdom, December 1975

What approaches would more directly address the source of the problem? It is instructive to examine a particularly elegant solution employed by radio broadcasters. To boost audience levels, radio shows routinely offer prizes to listeners such as concert tickets, vacations and money. Listeners tune in, wait for a signal such as a particular song and then call in hoping to win the prize. Of course, this invites an episode of severe focused loading at the switch board of the radio station as many listeners simultaneously call. The brilliantly simple way out is to announce that "caller number 9" will be the winner. This provides an incentive for listeners to randomize their call-in times—calling in too early or too late will not work—and the focused load is thereby diffused.

Of course, many of the general-purpose congestion management techniques surveyed above may also be applied to the special case of focused loading. We believe, however, that separate consideration of this special case is worthwhile, for two main reasons. First, the fact that focused loading occurs at very predictable times means that it is possible to know in advance the cases for which a specialized solution should be used. Second, the generality of the above congestion management techniques prevents them from explicitly taking into account information about agent valuation functions. Focused loading occurs because many agents prefer to use the network at the same time. This additional knowledge makes it possible to design mechanisms that collect more revenue and make fewer (e.g., computational) demands on the network.

In this paper [2] we propose a game theoretic model of the problem of defocusing predictable and time-dependent focused loads. We attempt to explain why techniques such as the radio show announcement can be effective, while also

[2] A preliminary version of this paper was presented at the ACM Conference on Electronic Commerce, 2001 [11].

contributing a formal model that permits analysis. While we do not rely on any particularly advanced results from game theory or mechanism design, we do assume that the reader is familiar with such concepts as individual rationality, risk attitudes (e.g,. risk neutrality, risk aversion) and dominant strategies. Also in the game theoretic tradition, we refer to users as agents. Good introductions to the concepts listed above are provided in [7,15].

In Section 2 we give a formal model of the temporal resource contention problem, define metrics for evaluating agent distributions and related notions of optimality, and specify agent utility functions. In Section 3 we propose a simple mechanism under which load balancing is a weak equilibrium for agents who value slots identically. We strengthen this to a strict equilibrium in Section 4 and also prove that this mechanism is arbitrarily close to optimal. In Sections 5 and 6 we relax the assumption that all agents have identical utility functions and present two mechanisms that balance load when only bounds on agent valuations are known. Since these mechanisms cannot take into account exactly how much each agent would be willing to pay to use the network, these mechanisms are not optimal; however, we prove a bound on their optimality which depends on the tightness of the bound on agent utility functions. If these mechanisms were used in the original case where agents value slots identically, then they too would be arbitrarily close to optimal. Finally, in Section 7 we summarize and compare the four mechanisms presented in this paper.

## 2 Problem Definition

In order to motivate the notation that we will use throughout the paper, it is helpful to begin with an example. Consider a network resource with a fixed number of identical time slots, where usage cost does not depend on the time slot. For example, consider a usage-based web service such as a pay-per-view streaming video service in which usage is divided into half-hour blocks from 7 PM to midnight. We assume that each agent wants to use the network during only one time slot, that each agent knows his own valuation for each slot, and that all agents' utilities are maximized by using the network during the same slot. For example, all agents might prefer to use the network from 7:00 to 7:30, having strictly monotonically-decreasing valuations for later slots as compared to earlier slots. Since time slots are priced identically, rational agents would all choose to use the network from 7:00 to 7:30, leading to a focused load. We further assume that although the capacity of the network resource is unlimited (e.g., hosted on an ASP) the operator of the resource has an exogenous desire for users to de-focus their demands (e.g., the ASP charges the operator for peak bandwidth used [3] ).

### 2.1 Mechanism Characteristics

In order to spread out the focused load, the network will provide agents with an incentive to choose slots other than $\overline{s}$. In this paper we will consider mechanisms in which agents are probabilistically spared the usage cost for the slot they choose. The cost of using the slot is waived according to a probability which depends on the slot chosen, and which is independent of the probabilities corresponding to other slots.

More formally, a mechanism $\Phi$ is defined by a tuple $\langle t, m, N, f(.)\rangle$. The network operates over $t$ time slots, where each slot has a fixed usage cost of $m$, and where the set $N$ of $n$ agents, $a_1 \ldots a_n$, intend to use the network. Each agent $a_i$ takes an action $A_i$ of using a slot. The function $f : A_1 \times \cdots \times A_n \rightarrow [0, 1]^n$ maps the actions taken by all agents into individual probabilities $P_i$ that the cost of the slot chosen by $a_i$ will be waived. Though $f$ is specified by the mechanism, the network must draw from each $P_i$ to determine whether the usage cost will actually be waived for each agent. Note that the $P_i$'s are independent. By $q$ we denote the expected number of slots that will be offered to at least one agent for free. The distribution of agents is denoted $d$, and so $d(s)$ is the number of agents who chose slot $s$.

### 2.2 Agent Characteristics

We assume that all agents are risk neutral. Agent $a_i$'s valuation for slot $s$ is given by an arbitrary non-negative function $v_i(s)$. Let $\overline{s}_i = \arg\max_s v_i(s)$ and $\underline{s}_i = \arg\min_s v_i(s)$. Because we are concerned with cases in which focused loading occurs we will assume that all agents have identical and unique most- and least-preferred slots, although this assumption is not required for any of our results. (If agents find several slots to be the most preferable, some amount of load balancing is likely to occur without any intervention by the network, as agents will distribute themselves across these slots.) Therefore, we define constants $\overline{s}$ and $\underline{s}$ such that for all $i$, $\overline{s}_i = \overline{s}$ and $\underline{s}_i = \underline{s}$. In sections 3 and 4 we will make the assumption that all agents' valuation functions are identical

[3] A number of proposals for usage-based pricing of bandwidth suggest charging according to the "effective" bandwidth consumed by an operator. Roughly, the effective bandwidth of a connection is a value between the mean and peak bandwidths, capturing the trade-off between the long-term average amount of bandwidth used by the connection and the instantaneous peak bandwidth consumption. See, for example [4], and the references therein.

(in these sections we will use the notation $v$ rather than $v_i$ to describe agents' valuations). Of course this assumption is not realistic; we relax it in sections 5 and 6. Let $v^l$ and $v^u$ be lower and upper bounds on all agents' valuations, respectively: i.e., $\forall i, s\; v^l(s) \leq v_i(s) \leq v^u(s)$. It is important to note that these bounds apply to all agents: in our model no agent has a valuation for slot $s$ lower than $v^l(s)$ or higher than $v^u(s)$.[4] Using this notation, the restriction on agents' valuations in sections 3 and 4 can be understood as the case where $\forall s\, v^l(s) = v^u(s)$. Finally, each agent $a_i$ may also receive a signal from the network, denoted $\sigma(a_i)$.

In our model, the decision faced by agents is simply to choose a slot $s$. The space of agent strategies $\mathcal{S}$ is the space of all functions mapping from the information available to a probability distribution over slot choices. We denote an element of $\mathcal{S}$ as $S = \Pi(s)$: a distribution over slot choices. Agents are aware of the mechanism and consider it when determining their strategies. Let $\varphi \in \mathcal{S}^n$ denote a set of agent strategies, which we formally call a *strategy profile*. Let $\varphi(i)$ denote $a_i$'s strategy under strategy profile $\varphi$, and let $\{\varphi \setminus i, S\}$ denote the strategy profile where all agents $j \neq i$ choose the strategy $\varphi(j)$ and agent $a_i$ chooses the strategy $S$. We can write agent $a_i$'s expected utility under strategy profile $\varphi$ (recall that $\varphi(i)$ is a distribution over slot choices for agent $a_i$, and hence $\varphi(i)(s)$ is the probability that agent $a_i$ will choose slot $s$ under strategy profile $\varphi$):

$$u_i(\varphi) = \sum_{s_1=1}^{t} \cdots \sum_{s_i=1}^{t} \cdots \sum_{s_n=1}^{t} \Big[ \Big(\varphi(1)(s_1) \cdot \ldots \cdot \varphi(n)(s_n)\Big) \cdot \Big(v_i(s_i) - \Big(1 - f(s_1, \ldots, s_n)_i\Big) m\Big)\Big] \quad (1)$$

We can now give a key definition:

**Definition 1** *$\varphi$ is a* Nash equilibrium *of $\Phi$ if $\forall i, \forall S, u_i(\varphi) \geq u_i(\{\varphi \setminus i, S\})$.*

Intuitively, no agent can gain by unilaterally deviating from a Nash equilibrium. This type of equilibrium is also referred to as a weak Nash equilibrium since it is possible that the agent receives equal utility from alternative strategies. When no such alternative exists, we have a *strict* Nash equilibrium:

[4] While these bounds strengthen our results, the assumption that they exist is not unrealistic. The upper bound is easily justified by the fact that no agent is willing to pay an arbitrarily large amount. The lower bound is trickier, since agent $a_i$ might simply not be interested in using some slot $s$ (i.e. $v_i(s) = 0$). However, since we're interested in defocusing the load, in practice we will be considering time slots that agents want to use. Therefore, it is not unrealistic to assume that every agent has a non-zero valuation for every slot.

**Definition 2** *$\varphi$ is a* strict Nash equilibrium *of $\Phi$ if $\forall i, \forall S \neq \varphi(i), u_i(\varphi) > u_i(\{\varphi \setminus i, S\})$.*

Equation (1) is complicated because it accounts for the calculation of the probability that slot $s_i$ is free, starting from a strategy profile. Although this definition of utility is necessary for discussing Nash equilibria, in other parts of the paper we will find it more convenient to take as given the same distribution $p$ for all agents, indicating the probability of each slot being free. We can then specify an expression for $a_i$'s expected utility for choosing slot $s$:

$$u_i(s) = v_i(s) - \Big(1 - p(s)\Big)m \qquad (2)$$

### *2.3 Restrictions on the Class of Mechanisms*

We now consider restrictions on the class of mechanisms that could be used to solve the focused loading problem, not to make the problem easier to solve, but in order to identify solutions with desirable characteristics. First, we introduce a restriction concerned with agents' incentives to participate (as discussed below, this condition is stronger than the standard mechanism design requirement of individual rationality). Next, we discuss restrictions that could arise from implementation considerations and the case of continuous pricing.

**Definition 3** *A mechanism $\Phi$ is* participation-safe *if and only if $m \leq v^l(\overline{s})$.*

We will consider only participation-safe mechanisms in this paper; that is, we require that the fixed usage cost for the network resource must never exceed the lower bound on any agent's valuation for his most-preferred slot. Intuitively, this means that every agent will always be able to choose at least one slot in which his payment will never exceed his valuation, and hence that it will be rational for him to participate regardless of how the mechanism assigns free slots. Observe that participation-safety implies individual rationality, because, regardless of $P_i$, agent $a_i$ can choose slot $\overline{s}$ and achieve a non-negative utility. Individual rationality does depend on $P_i$, and thus is a weaker condition.

We do not restrict the class of mechanisms in order to simplify analysis. As it turns out, it is very easy to design and analyze mechanisms that have a fixed cost exceeding all agents' valuations, and then reward agents only when they behave as desired. Such mechanisms can have good theoretical characteristics (such as optimality, defined below) and can remain consistent with individual rationality by assuring agents non-negative expected gain. Indeed, it turns out that in what follows, everywhere we prove $\varepsilon$-optimality or $(c+\varepsilon)$-optimality, we could prove optimality or $c$-optimality, respectively, if we were not restricted to participation-safe mechanisms. However, we believe that such mechanisms would be considered unreasonable to deploy in practice despite their theoretical benefits, because they address the problem of focused loading by threatening agents with unviable alternatives—slots whose expected costs exceed agents' valuations—rather than giving agents positive incentives to behave as desired.

Because of the difficulty of implementing complex protocols on a highly-loaded network resource, it is worthwhile to consider various other restrictions on the class of mechanisms. For example, it may or may not be possible to reimburse agents after all agents have chosen a slot, as opposed to doing so after each agent chooses. Also, it may or may not be permissible for $f$ to depend on what slots agents chose, as this would require that information be stored for each agent, and again that billing be deferred until after all agents have selected slots. In some settings it might not be reasonable for the network to give signals to agents; in other cases, it would be possible to give signals but not to record which signals were given to which agents. The significance of the time, space and communication complexity of the mechanism may also vary depending on the setting. We discuss these and other trade-offs in section 7.

Also, it might appear that more powerful mechanisms could be designed if prices could be varied arbitrarily, as opposed to our model in which slots must be priced at either $m$ or 0. In fact, since we assume that agents are risk-neutral, agents will be indifferent between any slot priced on the range $[0, m]$ and the same slot made free with an appropriate probability. Furthermore, $m$ can be increased arbitrarily. In the case of risk-averse agents, such 'continuous pricing' *would* be useful: our results throughout this paper hold for risk-averse agents if and only if this sort of continuous pricing scheme is used. We have chosen not to emphasize continuous pricing because it would be likely to make greater computational and communication demands on the network; however, all our results are compatible with such a scheme, and furthermore our bounds on $q$ and $m$ (see, e.g, equations (7), (8), and (9)) may be dropped in this case.

### *2.4 Evaluating Outcomes*

The network has two aims: to balance the load caused by the agents' selection of slots and to collect as much revenue as possible. We denote the network's expected revenue given a mechanism $\Phi$ and equilibrium $\varphi$ as $E[R|\Phi, \varphi]$. The network collects a payment of $m$ from each participating agent except for those who receive free slots. Expected revenue is given by:

$$E[R|\Phi,\varphi] = \sum_{i=1}^{n}\sum_{s_1=1}^{t}\cdots\sum_{s_n=1}^{t}\Big[\Big(\varphi(1)(s_1)\cdot\ldots\cdot\varphi(n)(s_n)\Big)\cdot \Big(1-f(s_1,\ldots,s_n)_i\Big)m\Big] \quad (3)$$

We define $g$ as the monetary value to the network of the variance of load across the set of time slots. Lower variance corresponds to a more even load and thus to a higher dollar value; thus $g$ must decrease strictly as variance increases. We will say that load is balanced when $g$ is maximized, which corresponds to minimal variance. We define the *superlinear summation* class of functions to be the set of functions in which $g(d) = -\kappa\sum_i h(d(i))$, where $h$ is superlinear in $d(i)$ and $\kappa$ is a constant that is used to indicate the relative importance of load balancing to the network. Note that this measure is only reasonable if we assume that each agent consumes about the same amount of load. The expected value of load balancing is given by:

$$E[g|\varphi] = \sum_d g(d)Prob(d|\varphi) \quad (4)$$

Maximizing revenue and maximizing $g$ are conflicting goals, as it costs the network more to induce an agent to choose slot $\underline{s}$ than to choose slot $\overline{s}$. Indeed, note that revenue is maximized in the original focused loading equilibrium when all agents choose $\overline{s}$ and $\forall i\, P_i = 0$. According to our problem definition, agents are *willing* to distribute themselves this way, and thus this equilibrium can be achieved without waiving any agents' usage fees. In some systems this could be a desirable outcome; however, we have assumed that the mechanism designer would prefer at least some balancing of the load. The network must therefore trade off quality of load balancing against expected revenue; the degree of trade-off desired may be specified through the choice of $\kappa$. Given definitions of the expected values $R$ and $g$, we can define $z$, the network operator's evaluation of equilibrium $\varphi$ of mechanism $\Phi$:

$$z(\Phi,\varphi) = E[R|\Phi,\varphi] + E[g|\varphi] \quad (5)$$

It will be useful to define the best possible distribution of agents given a free slot distribution that applies to all agents. Imagine a mechanism $\Phi_{all}$ in which all strategy profiles are in equilibrium, and $P_i = p(A_i)$. Intuitively, this is the best distribution of agents for the mechanism, given the constraint that the free slot distribution must be the same for all agents.

**Definition 4** *A distribution $d$ is ideal for $p(s)$ if and only if an equilibrium*

*$\varphi$ which deterministically results in distribution $d$ maximizes $z(\Phi_{all}, \varphi)$.*

Note that this expression may not have a unique maximum. We will denote an ideal distribution $d$ as $d^*$.

Next, we define the optimality of an equilibrium under a mechanism. Essentially, an equilibrium of a given mechanism is optimal if there does not exist another equilibrium of any other mechanism that yields a higher expected value of $z$.

**Definition 5** *A mechanism-equilibrium pair $(\Phi, \varphi)$ is optimal if and only if for all other pairs $(\Phi', \varphi')$, $z(\Phi, \varphi) \geq z(\Phi', \varphi')$, where $n$ is held constant.*

This definition of optimality is problematic when agents have different valuation functions that are not known by the network—the case we take up in sections 5 and 6. An optimal mechanism for this case would have to set each agent's expected payment to exactly his valuation for any slot chosen, by constructing a different $P_i$ for each agent. For every set of agents there does exist a set of such mechanisms. However, it is impossible to select such a mechanism based on the information available; furthermore such a mechanism will violate our restriction that it be participation-safe, because an agent $a_i$ who chooses slot $\overline{s}$ is charged $v_i(\overline{s})$, which can be exceed $v^l(\overline{s})$. To overcome this difficulty we provide an alternate notion of optimality that bounds the average loss per agent as compared to an optimal mechanism:

**Definition 6** *A mechanism-equilibrium pair $(\Phi, \varphi)$ is c-optimal if and only if for all other pairs $(\Phi', \varphi')$, $z(\Phi, \varphi) + cn \geq z(\Phi', \varphi')$, where $n$ is held constant and $c > 0$.*

For convenience, we will also make use of the term *[c-]optimal* to refer to equilibria alone, in cases where the mechanism giving rise to the equilibrium is unambiguous.

**Definition 7** *An equilibrium $\varphi$ is [c-]optimal if $\varphi$ is an equilibrium of mechanism $\Phi$, and $(\Phi, \varphi)$ is [c-]optimal.*

We call $\varphi'$ where all agents choose the same slot a *focused-loading equilibrium.* We assume that $g$ and $v$ do not take values that would cause $\varphi'$ to be optimal. This assumption is only required for our proof of Theorem 2, but it is a reasonable one for us to make since if $\varphi'$ *were* optimal, we would have no problem to solve in the first place.

## 3 Preselection Mechanism

In this section we consider a simple mechanism, designed to make agents indifferent between all time slots despite their initial preferences. This mechanism will be formally referred to as $\Phi_1$, and informally called 'preselection', since it decides which slots will be free before observing the actions of the agents. This mechanism is unrealistic in several ways, and we do not discuss it here in order to propose that it should be used in practice. Indeed, such a mechanism is an obvious first approach to the problem of focused loading, and so it is important to demonstrate its insufficiency. Furthermore, the exposition of this mechanism will prove useful as a starting point for the discussion of more sophisticated mechanisms.

$\Phi_1$works as follows:

(1) The network determines free slots by drawing from $p$. (Thus, $P_i = p(A_i)$.)
(2) Agents choose a slot.

### *3.1 Equilibria*

We know from the definition of the problem that when there is no chance that they will win a free slot agents prefer slot $\overline{s}$ to slot $\underline{s}$. We can overcome this preference by biasing $p(s)$. An agent's expected utility is given by $u_i(s) = v(s) - (1 - p(s))m$. Recall that we assume $v^l = v^u$ until section 5; here we use (unsubscripted) $v$ to denote the valuation function that all agents share. We can make agents indifferent between slots by requiring that all time slots will have the same expected utility for agents: that is, that the expected utility derived from each time slot is equal to the average expected utility over all time slots. This is expressed by the equation $v(s) - \Big(1 - p(s)\Big)m = \frac{1}{t}\sum_i \Bigg(v(i) - \Big(1 - p(i)\Big)m\Bigg)$. Algebraic manipulation and $q = \Sigma_s p(s)$ give us:

$$p^*(s) = \frac{\frac{1}{t}\Big(qm + \sum_i v(i)\Big) - v(s)}{m} \tag{6}$$

Observe that since free slots are free for all agents, $q$ represents the expected number of free slots. Because we will find this probability distribution useful throughout the paper, we have given it a name: $p^*$.

If free slots are awarded according to $p^*$, it is a weak Nash equilibrium for all agents to select a slot uniformly at random. We will call this equilibrium $\varphi_1$. Consider the case where all other agents play according to $\varphi_1$, and one remaining agent $a_i$ must decide his strategy. Since the choice of any slot entails equal utility on expectation, $a_i$ can do no better than to randomly pick a slot. Again, $\varphi_1$ is only a weak equilibrium: indeed, there is no strategy $a_i$ could follow that would make him worse off.

We now make several remarks about the preselection mechanism. First, note that the above analysis assumes that $a_i$ is risk-neutral. If $a_i$ is risk-averse, he will prefer slot $\overline{s}$, since it gives the largest fixed payment, $v(\overline{s})$. Second, this mechanism is not susceptible to collusion, because each agent is indifferent between all pure strategies regardless of the actions of other agents. Finally, since *all* strategy profiles are weak equilibria under the preselection mechanism, it would be reasonable to ask why we pay special attention to $\varphi_1$. It may be argued that randomization is a "natural" response to indifference, and so we will consider this as a primary case in the next subsection; however, none of our results depend on the assumption that agents will choose this strategy.

### *3.2 Bounds on q and m*

It appears that deviation from $\varphi_1$ will never be profitable for agents, since we have guaranteed that all slots provide the same expected utility. Consider the most profitable deviation, from $\underline{s}$ to $\overline{s}$. We have claimed that the utility of both slots is the same: $v(\overline{s})-\Big(1-p(\overline{s})\Big)m = v(\underline{s})-\Big(1-p(\underline{s})\Big)m$. However if $qm$ is too small or too large, $p(\underline{s}) - p(\overline{s}) > 1$ will hold. Since we want to interpret $p(\underline{s})$ and $p(\overline{s})$ as probability measures, we must add the constraints $p(\overline{s}) \geq 0$ and $p(\underline{s}) \leq 1$. Without these constraints, the equation for $p^*$ still makes sense if we consider continuous pricing rather than our default model of free/non-free slots; $p > 1$ corresponds to an expected slot cost of less than zero (paying agents to choose a slot) while $p < 0$ corresponds to an expected slot cost of more than $m$. Substituting $p(\overline{s}) \geq 0$ into equation (6) and rearranging, we get:

$$q \geq \frac{tv(\overline{s}) - \Sigma_i v(i)}{m} \tag{7}$$

For the second condition, we require that $p(\underline{s}) \leq 1$, which gives us:

$$q \leq \frac{t\Big(v(\underline{s}) + m\Big) - \Sigma_i v(i)}{m} \tag{8}$$

We must also ensure that a value of $q$ exists for a given $m$ and $v$. Intersecting the two bounds and simplifying, we get:

$$m \geq v(\overline{s}) - v(\underline{s}) \tag{9}$$

Indeed, if $m < v(\overline{s}) - v(\underline{s})$ then if an agent were certain to win a free slot in $\underline{s}$ and guaranteed never to win a free slot in $\overline{s}$, he would still prefer $\overline{s}$ to $\underline{s}$.

### *3.3 Maximizing Revenue*

Equation (3) gave a general expression for $E[R|\Phi, d]$. However, under equilibrium $\varphi_1$ all agents randomly select a slot, which allows us to write an expression for $E[R|\Phi_1, \varphi_1]$ that does not include a summation. In $\varphi_1$ expected revenue is given by the percentage of non-free slots times cost per slot times number of agents:

$$E[R|\Phi_1, \varphi_1] = \left(1 - \frac{q}{t}\right) mn \tag{10}$$

Increasing $m$ will increase expected revenue; however, recall that we require that the mechanism be participation-safe, and hence that $m \leq v^l(\overline{s})$. Regardless of the particular value of $m$, reducing $q$ (the expected number of free slots) will increase expected revenue.

We will now show how the network can maximize revenue. We define $v_{avg}$ as $\frac{1}{t}\sum_s v(s)$. The requirement that an agent's utility for slot $s$ must be greater than or equal to zero—i.e., that $v(s) - \Big(1 - p(s)\Big)m \geq 0$—can be rewritten, substituting in $p^*$, as $v_{avg} - (1 - \frac{q}{t})m \geq 0$. The seller's revenue will be maximized when all agents get zero utility. Thus we must have:

$$\left(1 - \frac{q}{t}\right) m = v_{avg} \tag{11}$$

We substitute in the lower bound for $q$ from equation (7): i.e., $q = \frac{1}{m}\Big(tv(\overline{s}) - \Sigma_i v(i)\Big)$. Rearranging for $m$, we get $m = v(\overline{s})$. This satisfies equation (9) and ensures that the mechanism is participation-safe, so we are done.

This is intuitive because when we minimize $q$ we set $p(\overline{s}) = 0$. We know that agents are indifferent between all slots, and so agents will be willing to choose any slot when the cost of $\overline{s}$ does not exceed their valuation. We thus set $m = v(\overline{s})$ and (plugging $m$ into the lower bound on $q$) $q = t\Big(1 - v_{avg}/v(\overline{s})\Big)$.

We have shown that each agent can be made to pay an expected amount exactly equal to his utility for any slot he chooses. However, $\varphi_1$ is not guaranteed to achieve an ideal distribution of agents, and therefore $\varphi_1$ is not optimal. The easiest way to show this is to present another equilibrium of the preselection mechanism that *is* optimal.

### *3.4 Optimal Equilibria*

Consider an equilibrium in which each of the agents deterministically chooses one slot. (Recall that *any* strategy is rational under $\Phi_1$, and thus that any set of strategies is a weak equilibrium.) In one such equilibrium, agents deterministically choose slots so that the distribution of all agents is ideal; we will call this equilibrium $\varphi_1^*$. Unsurprisingly, we can show:

**Theorem 1** $(\Phi_1, \varphi_1^*)$ *is optimal.*

**PROOF.** Please see the appendix.

**REMARK.** Recall that a mechanism-equilibrium pair is optimal when there does not exist another mechanism that has an equilibrium giving rise to a distribution that yields a higher value according to the evaluation function $z$.

The equilibrium $\varphi_1^*$ is optimal, but it is extremely unlikely that it would arise through the choices of real agents. As mentioned above, the fact that agents are indifferent between all slots means that *every* combination of agent strategies is a weak equilibrium. In fact, the preselection mechanism gives rise to many equilibria that minimize $g(d)$. For example, the case in which all agents choose slot $\overline{s}$ is a weak equilibrium. Since discouraging focused loading is the purpose of the preselection mechanism, it is undesirable to find that such behavior remains an equilibrium! However, this drawback is inherent to the setting as we have modeled it so far; a preselection mechanism can only yield weak equilibria or focused-loading equilibria.

**Theorem 2** *When agents have identical utility functions and no signals are given to agents and the network preselects* $p$ *before agents move, all equilibria are either weak or focused-loading.*

**PROOF.** Please see the appendix.

**REMARK.** Intuitively, this proof shows that under the conditions of the preselection mechanism any incentive given to one agent is given to all the agents, and that the mechanism designer must therefore choose between encouraging all agents to choose the same slot and making all agents indifferent between a set of slots.

In fact, we can show another negative result: there does not exist an optimal mechanism that is participation-safe and that gives rise to a strict equilibrium.

**Theorem 3** *There does not exist an optimal* $(\Phi, \varphi)$ *for which* $\varphi$ *is a strict equilibrium and* $m \leq v(\overline{s})$*.*

**PROOF.** Please see the appendix.

Theorem 3 shows that strict, optimal equilibria do not exist for participation-safe mechanisms. However, if we allow networks with different characteristics than those we allowed in this section, we can see that it is possible to get close to a strict, optimal equilibrium when agents have identical utility functions and no signals are given to agents, and $p$ depends on the agents' actions. Intuitively, consider a mechanism that sets $p = (1 + \varepsilon)p^*$ if agents achieve an ideal distribution, and $p = 0$ otherwise. Further, consider a set of (pure) agent strategies where agents *happen* to distribute themselves according to $d^*$ for $p^*(s)$. This is an equilibrium because agents are penalized for deviating. Intuitively, it is nearly optimal because agents achieve an ideal distribution with respect to the mechanism, and the probability of awarding free slots is arbitrarily close to the probability from the optimal mechanism-equilibrium pair described in theorem 1. However, it would be extremely difficult for agents to coordinate to this equilibrium in real play. In the next section we will show how the use of a non-binding coordination phase before the selection of slots can help agents to reach strict, nearly-optimal equilibria.

## 4 Bulletin Board System Mechanism

In this section we assume that agents are given a *bulletin board system*: a forum in which all communications are visible to all agents and the identity of agents is associated with their transmissions. For simplicity, we allow a very limited form of communication: agents indicate the slot that they intend to choose. We assume that agents do not all indicate slots at the same time; rather, they indicate sequentially during the first phase. Let $d_j(s)$ denote the number of agents who have indicated that they will choose slot $s$ after a total of $j$ agents have posted to the bulletin board. $d^*$ will again be the ideal distribution for $p * (s)$. Agents' communications through the bulletin board are *cheap talk*: a technical term that indicates that these communications are not binding in any way. Even so, the bulletin board can help agents to coordinate on desirable equilibria. Mechanism $\Phi_2$ follows:

(1) The network picks "potentially free"[5] slots according to $(1 + \varepsilon)p^*$.

[5] We redefine $q$ as the expected number of "potentially free" slots; the same redef-

(2) Agents communicate through the bulletin board.
(3) Agents choose time slots.
(4) If $d = d^*$, then "potentially free" slots are made to be free. That is, $P_i = p^*(A_i)$. Otherwise, all agents are made to pay for their slots ($P_i = 0$).

### *4.1 Equilibria*

A strict equilibrium in $\Phi_2$, which we call $\varphi_2$, is for the $i^{\text{th}}$ agent to indicate on the bulletin board a slot $s$ such that $d_{i-1}(s) < d_i^*(s)$, and ultimately to choose that slot $s$. Consider the case where all other agents follow $\varphi_2$ and agent $a_i$ must decide his strategy. If $a_i$ cooperates and chooses slot $s$ then the distribution of agents will be $d^*$ and so $a_i$ will receive an expected utility of $v(s) - \Big(1 - (1+\varepsilon)p^*(s)\Big)m$. If $a_i$ defects to slot $s'$, one of two cases will result. In the first case, agents indicating their choices after $a_i$ will compensate for his deviation by choosing different slots; thus $a_i$ will receive the same expected utility as he would have received if he had not deviated. In the second case, $a_i$ will be late enough in the sequence of agents indicating their choices that the agents who indicate after him will be too few to bring the distribution back to $d^*$. In this case $a_i$ will receive an expected utility of $v(s') - m$. The key point is that $a_i$ does not know the total number of agents, and so he must assign non-zero probability to the second case, regardless of the number of agents who have already indicated. Furthermore, we must show that $a_i$ will choose the slot he indicated on the bulletin board even though his selection was not binding. If all other agents follow $\varphi_2$ then there is clearly no incentive for $a_i$ to choose a different slot than he indicated, because that would certainly prevent $d = d^*$ and reduce his payoff. Therefore $\varphi_2$ is strict as long as $v(s) + (1+\varepsilon)p^*(s)m > v(s')$ for all $s, s'$ such that $1 \le s, s' \le t$. Simplifying, we derive the conditions similar to those described in section 3.

$$\frac{tv(\overline{s}) - \Sigma_i v(i)}{m} \le q \le \frac{t\Big(v(\underline{s}) + \frac{m}{1+\varepsilon}\Big) - \Sigma_i v(i)}{m} \tag{12}$$

Again, we must intersect the two bounds to get a bound on $m$, which we combine with the constraint on participation-safe mechanisms:

$$(1+\varepsilon)\Big(v(\overline{s}) - v(\underline{s})\Big) \le m \le v(\overline{s}) \tag{13}$$

This equilibrium relies on the fact that each agent can choose a slot as if he were the last agent and achieve the distribution $d^*$, even if all agents before

---

inition is required for section 6.

him chose slots in this same way. We prove that this greedy approach works in section 4.2.

An analysis of the possibility of collusion in the bulletin board mechanism is not appropriate, because agents are already encouraged to coordinate with each other. Any agent or cartel of agents who deviated would hurt themselves along with all other agents.

It is well known that any game having an equilibrium arising from cheap talk coordination has other equilibria in which agents ignore the cheap talk [2]. The bulletin board mechanism is no exception. All agents choosing $\overline{s}$ (focused loading) is an equilibrium when the resulting $d$ could not be transformed into $d^*$ by one agent choosing a different slot. Note, however, that $\varphi_2$ Pareto-dominates all equilibria where the cheap talk is ignored and a different distribution results.

### *4.2 Greedy Assignment of Slots*

In $\varphi_2$ each agent chooses a slot that would result in an optimal distribution if he were the last agent to post to the bulletin board. For this reason it is important to show that we can assign slots to agents greedily, with the guarantee of achieving the ideal distribution for whatever number of agents eventually participate.

We must introduce new notation to describe changes as each agent chooses a slot in turn. (Readers who do not intend to read the proof for lemma 1 can safely skip to section 4.3.) First, we will subscript $d$ to indicate the total number of agents in the distribution, so that we can describe the distributions that result after only a subset of agents have chosen slots. By $d_i^*$ we denote the optimal distribution of $i$ agents. Second, we define $\Delta(d_i, s)$ to be the increase in $z$ if one agent is added to slot $s$, relative to $d_i$. Define the decomposition $\Delta(d_i, s) = \Delta_E(d_i, s) + \Delta_g(d_i, s)$, where $\Delta_E(d_i, s)$ is the increase in $E[R|\Phi, d_i]$, and $\Delta_g(d_i, s)$ is the increase in $g(d_i)$. In equilibrium $\Delta_E(d_i, s)$ does not depend on $d_i$, but only on $p(s)$ and $m$. (We assume here that $p$ does not depend on $d_i^*$.) Two properties follow from the fact that $g$ is *superlinear summation*:

(1) $\Delta_g(d_i, s)$ is strictly monotonically decreasing in $d_i(s)$
(2) $\Delta_g(d_i, s) = \Delta_g(d'_j, s)$ for all distributions $d'_j$ where $d'_j(s) = d_i(s)$

Since $\Delta_E$ does not depend on $d_i$, $\Delta$ also has these properties.

We now describe a function $\gamma$: let $\gamma(i)$ represent the slot number that will be assigned to $a_i$, where $a_i$ is the $i^{\text{th}}$ agent to register. Let $d_i^{\gamma}(s)$ be the number of times $s$ occurs in $\{\gamma(1), \ldots, \gamma(i)\}$. We note that $\forall s \Delta(d_0^{\gamma}(s)) = 0$. We can now inductively define $\gamma$: $\gamma(i) = \arg\max_s \Delta(d^{\gamma}_{i-1}(s))$.

**Lemma 1** *$\forall i$ $d^{\gamma}_i$ is ideal under $\Phi_2$.*

**PROOF.** Please see the appendix.

**REMARK.** This lemma demonstrates that greedy assignment of slots to agents leads to an ideal distribution when we assign slots according to $\gamma$ as defined above.

### *4.3 $\varepsilon$-Optimality*

Although theorem 3 showed that the bulletin board mechanism cannot be optimal, it turns out that it can be made arbitrarily close to optimal. We now show that there exists no other equilibrium of any other mechanism which will yield a value of $z$ larger than $z(\Phi_2, \varphi_2) + \varepsilon$ for arbitrarily small $\varepsilon$.

**Theorem 4** *$(\Phi_2, \varphi_2)$ is $\varepsilon$-optimal.*

**PROOF.** Please see the appendix.

**REMARK.** This is a key result, because it shows that we can get arbitrarily close to an optimal equilibrium with a mechanism that could actually be used in practice. Furthermore, the fact that the equilibrium is strict is encouraging, because it means that an agent could not reduce $z$ by deviating from $\varphi_2$ without also reducing his own utility.

### *4.4 Implementation Considerations*

We point out that $\varepsilon$-optimality means that the mechanism can lose $\varepsilon$ per agent; in practice, $\varepsilon$ would have to be large enough to overcome agents' indifference between nearly-identical payoffs and encourage them to coordinate.

Although we speak about agent strategies throughout this paper, it is worthwhile to note that in a real system these strategies would probably be implemented in software that most users would not be able to change easily. Of course, this is not an argument against equilibrium analysis or the careful design of economic mechanisms. If agents could gain by deviating, there would be an incentive for users to change their software, and once software has been modified it is easily redistributed. However, the fact that the mechanism designer could in many cases distribute client software is significant because it can act as a coordination device: agents' common knowledge of using the same software could help them to coordinate to an equilibrium the mechanism designer has preselected. Although the bulletin board mechanism gives rise to non-$\varepsilon$-optimal equilibria, these might be avoided if client software helped agents to coordinate to $\varphi_2$.

## 5 Collective Reward Mechanism

We now consider the more general and realistic case where each agent may have a different $v_i$, bounded by $v^l$ and $v^u$, as described in section 2. Recall that since the network does not know each agent's $v$, we can no longer tune $m$, $q$, and $p$ to extract the maximum amount of revenue from each agent.

In this section we also allow the network to give signals to agents, to allow the agents to coordinate to a desirable equilibrium; we also show how collective reward may be used to prevent agents from deviating. We define mechanism $\Phi_3$ as follows:

(1) Each agent indicates that he will participate.
(2) The network gives a signal to each agent from $\{1, \ldots, t\}$.
(3) Agents choose time slots.
(4) The network determines whether each slot will retroactively be made free.

In this mechanism, the chance that slot $s$ will be free, $p(s)$, depends on the number of agents who chose slot $s$, $d(s)$. Let $count(s)$ be the number of agents who were given the signal $s$. Define $d^+(s) = d(s) - count(s)$. For the collective reward mechanism $\Phi_3$:

$$p(s) = \begin{cases} p^b(s) & \text{if } d^+(s) \leq 0 \\ 0 & \text{if } d^+(s) > 0 \end{cases} \tag{14}$$

Thus $P_i = p^b(A_i)$ if $d^+(s) \leq 0$ and $P_i = 0$ otherwise, where $p^b(\cdot)$ is defined below.

We will assign signals to agents so that $count(s) = d^*(s)$, where $d^*$ is now ideal for $p^b(s)$. The idea of this mechanism is that agents who choose the slot $s$ to which they are assigned will get that slot free with probability $p^b(s)$, and agents who deviate to another slot will pay $m$. The $p(s)$ used for this mechanism will thus differ from $p(s)$ for the previous two mechanisms. The intuitive reason for the change is that in $\Phi_1$ and $\Phi_2$ we used $p$ to make agents indifferent between all slots. Now, however, we use $p$ so that agents will not deviate from an assignment to a particular slot. We will construct $p^b$ so that each agent $a_i$ will choose his assigned slot even when $a_i$ has the lowest possible valuation for the slot corresponding to his signal, and the highest possible valuation for $\overline{s}$, the most profitable slot to which he could deviate. When an agent is assigned a slot $s \neq \overline{s}$, this condition can be formalized as:

$$v^l(s) - \Big(1 - p^b(s)\Big)m = v^u(\overline{s}) - m + \varepsilon \tag{15}$$

Here as before $\varepsilon$ is a small, positive value used to make agents strictly prefer the slot to which they are assigned. It can be interpreted as an offset to $v^u$, giving us a strict upper bound on agents' utilities. If we make an agent with this impossibly high valuation for slot $\overline{s}$ indifferent between his assigned slot and $\overline{s}$, then any agent who actually plays the game must prefer his assigned slot. We can now derive $p^b$:

$$p^b(s) = \begin{cases} \frac{v^u(\overline{s}) - v^l(s) + \varepsilon}{m} & \text{if } s \neq \overline{s} \\ 0 & \text{if } s = \overline{s} \end{cases} \tag{16}$$

The case of $s = \overline{s}$ is considered separately because an agent assigned to this slot has no incentive to deviate. Note that if $v_i(s) = v_i(\overline{s})$ is possible for an $s \neq \overline{s}$, then we would have to change the definition of $p^b$ to maintain a strict equilibrium, giving $\varepsilon'$ probability of awarding $\overline{s}$ free.

We now need to define bounds on $m$. The condition that $p^b(s) \leq 1$ can be rewritten, combined with the requirement that the mechanism be participation-safe, as:

$$v^u(\overline{s}) - v^l(\underline{s}) + \varepsilon \leq m \leq v^l(\overline{s}) \tag{17}$$

For $\Phi_3$ $q$ is defined as:

$$q = \sum_{i \neq \overline{s}} \Big( \frac{v^u(\overline{s}) - v^l(i) + \varepsilon}{m} \Big) \tag{18}$$

To maximize expected revenue, the collective reward mechanism sets $m$ to its upper bound of $v^l(\overline{s})$.

An equilibrium $\varphi_3$ is for each agent $a_j$ to select the slot corresponding to his signal.[6] Consider the case where all other agents follow this strategy, and one remaining agent $a_i$ decides his strategy. If agent $a_i$ selects slot $s$ as above, then his expected utility is $u_i(s) = v_i(s) - \Big(1 - p^b(s)\Big)m$. Deviating to even the best slot only gives him $u_i(\overline{s}) = v_i(\overline{s}) - m$. We have defined $p^b$ so that in this case $a_i$ strictly prefers slot $s$.

There are no equilibria of the collective reward mechanism for which $d \neq d^*$. Consider any distribution of agents such that $d \neq d^*$. There must be some $s_1$ such that $d^+(s_1) < 0$, and some other $s_2$ such that $d^+(s_2) > 0$. An agent in $s_2$ thus has no chance of a free slot, and he receives utility of at most $v_i(\overline{s}) - m$. If he switches to $s_1$, then his probability of receiving a free slot becomes $p^b(s_1)$ because $d^+(s_1) \leq 0$. Since $p^b$ is constructed so that this agent receives more utility, on expectation, than $v_i(\overline{s}) - m$, he has incentive to move to slot $s_1$. However, there do exist equilibria in which agents do not select slots corresponding to the signals they receive. For example, consider the case where agent $a_i$ deterministically selects the slot $\sigma(n+1-i)$. (Note that this could occur even if agent $a_i$ did not *know* what signal agent $a_{n+1-i}$ receives.) In this case the distribution of agents is $d^*$, and so the analysis above demonstrates that all agents have a disincentive to deviate. Another example is where all agents select the slot corresponding to their signals except where agent $a_i$ chooses slot $\sigma(j)$ and agent $a_j$ chooses slot $\sigma(i)$.

Harmful collusion is not possible under the collective reward mechanism. A single agent who deviates from $\varphi_3$ can harm other agents by denying them a chance at a free slot. However, no set of agents is able to *improve* other agents' chance of getting a free slot, and so there is no way that a cartel of agents could benefit from colluding.

**Theorem 5** *$(\Phi_3, \varphi_3)$ is c-optimal for* $c = \max_s\Big(v^u(s) - v^l(s)\Big) + \varepsilon$.

**PROOF.** Please see the appendix.

**REMARK.** Because it depends on bounds rather than on agents' actual valuations, $\varphi_3$ is not optimal. However, this theorem shows that we can prove

[6] This note is intended for readers familiar with game theory. Consider the space of all functions $H : \mathbb{N} \to \{1, \ldots, t\}$ mapping from agent names to suggested slots. Let $Prob$ be a probability distribution over all functions $h \in H$ that give rise to the agent distribution $d^*$. If signals are assigned based on an $h$ drawn from $Prob$ then $\varphi_3$ can easily be formulated as a correlated equilibrium. However, for ease of exposition and to emphasize the sequential assignment of agent signals for implementation reasons, we do not make further use of this formulation.

a bound on the optimality of $\varphi_3$, showing that the network can lose no more than $\max_s\Big(v^u(s) - v^l(s)\Big)+\varepsilon$ in revenue from each agent.

It follows from this statement that if we revert back to the setting from sections 3 and 4 (where $v^u(s) = v^l(s)$), the network will lose only $\varepsilon$ in revenue from each agent. It is only the change to bounds on valuation functions that causes the weaker claims on optimality for this mechanism and the next.

**Corollary 1** $(\Phi_3, \varphi_3)$ *is $\varepsilon$-optimal for $v^l = v^u$.*

**PROOF.** This follows directly from the preceding theorem, because $v^l = v^u$ implies that $c = \varepsilon$.

### *5.2 Implementation Considerations*

We observe that it may involve less overhead to assign single, persistent signals to agents if the game will be repeated many times. In this case, the collective reward mechanism may be used as above but without the signalling phase, and with each agent $a_j$ who did not participate counted by $d^+$ as having participated in slot $\sigma(j)$. This allows $\varphi_3$ to hold in the case where signals are not assigned repeatedly with the penalty that $\varphi_3$ will only be $c$-optimal for $c = \max_s\Big(v^u(s) - v^l(s)\Big)+\varepsilon$ when all agents participate.

## 6 Discriminatory Mechanism

A disadvantage of the bulletin board mechanism is that it reimburses some agents at the end of the game rather than simply waiving their fees. This requires tracking individual agents' behavior and executing more financial transactions, both of which could be costly to the network. Also, the bulletin board mechanism has non-optimal equilibria. Finally, irrational agents can harm others in both the bulletin board and collective reward mechanisms. These problems are eliminated by the discriminatory mechanism, $\Phi_4$, which makes use of agent signals and also discriminates by offering different free slots to different agents (although, as we will see in section 6.2 it makes new demands of the network that will sometimes be undesirable):

(1) Each agent indicates that he will participate.
(2) The network assigns signals to agents from $\{1, \ldots, t\}$ according to the $d^*$ that is ideal for $p^b$.
(3) The network chooses "potentially free" slots according to $p^b$.
(4) Each agent indicates what slot he selects.

(5) The network checks only those agents in each slot $s_i$ that was picked to be "potentially free" (for all agents who chose other slots, $P_i = 0$) . If agent $a_j$ in slot $s_i$ has $\sigma(a_j) = s_i$ then $P_j = p^b(A_j)$; otherwise $P_j = 0$.

### *6.1 Equilibria*

Agent $a_i$'s dominant strategy is to choose the slot corresponding to his signal. The analysis exactly follows that for $\varphi_3$; we call this equilibrium $\varphi_4$. The only difference is that an agent's expected utility does not depend on other agents' strategies, and hence $\varphi_4$ is an equilibrium in dominant strategies. A consequence is that $\varphi_4$ is unique. By exactly the same argument that was given in the proof of theorem 5, $(\Phi_4, \varphi_4)$ is $c$-optimal for $c = \max_s\big(v^u(s) - v^l(s)\big) + \varepsilon$. The same corollary also holds, and so $(\Phi_4, \varphi_4)$ is $\varepsilon$-optimal for the special case where $v^u = v^l$.

It may seem disappointing from a game-theoretic point of view that neither strategy nor even payoffs under the discriminatory mechanism depend on the actions of other agents. However, this may be seen as an advantage of the discriminatory mechanism, since irrational agents are not able to harm others.

### *6.2 Implementation Considerations*

As compared to the collective reward mechanism, the discriminatory mechanism makes two additional demands of the network. First, the network must keep track of the signals that are given to agents in the second step, so that they can be verified in the fifth step. In collective reward the system does not need any sort of user accounts; rather, it greedily assigns signals to agents, recording only the *number* of agents who received each signal.

Second, the discriminatory mechanism requires the network to verify user identities. In contrast, the collective reward mechanism simply counts the number of agents who chose each slot. Under the discriminatory mechanism the network only has to check the identity of agents from $q$ slots on expectation, since agents who choose a slot that is not potentially free do not have to be checked. It would be possible for the network to assume that all agents in possibly free slots have played according to the dominant strategy and to randomly check only a subset of the agents in these slots, but this would reduce the penalty for defection and thus sacrifice $c$-optimality.

In order to permit this verification, the mechanism can assign signals to agents in two different ways. The obvious option is to assign signals to agents as described in theorem 1, to store the numbers in some sort of user account requiring login and then to verify that agents selected the appropriate slot by requiring them to log in again before using the network resource. This approach requires further data storage by the mechanism, but the resulting $d$ will be ideal and thus the mechanism will be $c$-optimal as argued above. If this data storage is not desirable, a deterministic function may be used to calculate the slot that may be offered free to a given agent, and the same function may be used to determine whether each agent has selected the appropriate slot. For example, a hash of the agent's IP address—or of any other identifying information from the packet header—could be used. This approach has the disadvantage that it sacrifices optimality and for steps 2 and 5 in the mechanism, but the advantage that no information about identifying individual agents must be stored by the mechanism.[7] Indeed, if the function itself is publicized then the first two steps may be omitted from the mechanism, requiring only one interaction between agents and the network.

## 7 Comparison of Different Mechanisms

Table 1 summarizes and contrasts the mechanisms discussed in this paper. For convenience, we have divided the display into three parts: (i) a list of mechanism characteristics, (ii) a comparison of the outcomes of the mechanisms, and (iii) costs associated with executing the mechanisms.

## 8 Conclusion

Focused loading is a predictable network congestion problem. It is caused by a preference users have for transacting with a network resource at a specific time when the network charges transactions equally over a period of time. For example, focused loading frequently causes web servers to crash. In this paper we have taken an economic approach to de-focusing load by devising incentive schemes for encouraging users to desynchronize their transaction times. While general congestion-management techniques may be applicable to this problem, the use of a specialized solution is attractive because additional information about the problem can be used to increase revenue and reduce demands on the network.

[7] Another disadvantage is that an agent could register from one computer, receive a slot assignment, use the network from a second computer and be denied a chance for a free slot because the second computer's IP did not hash to the same signal. This could be addressed by requiring agents to use the network from the computer from which they registered, and permitting them to register again if they change their mind about which machine they want to use.

| | $\Phi_1$: **Preselection** | $\Phi_2$: **Bulletin Board** | $\Phi_3$: **Collective Reward** | $\Phi_4$: **Discriminatory** |
|---|---|---|---|---|
| **Earliest possible free slot selection** | Before any time slots | After all time slots | After each time slot | After each time slot |
| **Agent signals** | No | No | Yes | Yes |
| **The network must store agent signals** | No | No | No | Yes, or hash IP |
| **Agents may have different $v$ functions** | No | No | Yes | Yes |
| **Time required for coordination phase** | None | Substantial | Negligible | Negligible |
| **Type of equilibrium or strategy** | Weak equilibrium | Strict equilibrium | Strict equilibrium | Dominant strategy |
| **Non-optimal equilibria exist** | Yes | Yes | No | No |
| **Revenue increases if agents deviate** | No | Yes | Yes | Yes |
| **Harmful collusion** | No | No | No | No |
| **Irrational actions harm other agents** | No actions are irrational | Yes | Yes | No |
| **Time cost after coordination phase** | $O(n)$ | $O(n)$ | $O(n)$ | $O(n)$ |
| **Storage cost** | $O(q)$ (free slots) | $O(t)$ ($d$) | $O(n)$ (moves) | $O(n)$ (signals, identities) |
| **Communication cost** | $O(n)$ | $O(nt)$ | $O(n)$ | $O(n)$ |

Table 1
Comparison of $\Phi_1$, $\Phi_2$, $\Phi_3$, $\Phi_4$

We present a theoretical model of the problem, and discuss four mechanisms that induce selfish agents to smooth out their resource demands by probabilistically waiving the cost of resource usage. We show one very simple mechanism that achieves a weak load-balancing equilibrium, and three other, somewhat more complex mechanisms that balance load in strict equilibria or dominant strategies. Two of our mechanisms concern the case where all agents have the same valuations for different time slots, and two generalize to the case where the mechanism knows only bounds on agent valuations. We prove optimality and $\varepsilon$-optimality of the revenue/load balancing trade-off in the first case, and a bound on the optimality of this trade-off in the second case.

In future work, we plan to apply the methods proposed in this paper to defocus the load at web servers operating under transaction deadlines.

## References


[1] D. Blair. Impact of the Internet on core switching network. In *Proc. of ENPW'98*, Les Arcs, France, March 1998.

[2] V.P. Crawford and J. Sobel. Strategic information transmission. *Econometrica*, 50(6):1431–1451, November 1982.

[3] A. Demers, S. Keshav, and S. Shenker. Analysis and simulation of a fair queuing algorithm. *Journal of internetworking research and experience*, pages 3–26, October 1990.

[4] M.Stoer D.Songhurst, G.Stamoulis. Usage-based charging using effective bandwidths: studies and reality. In *Proceedings of the International Teletraffic Congress, ITC-16*, 1999.

[5] S. Floyd. Tcp and explicit congestion notification. *ACM Computer Communication Review*, 24(5):10–23, October 1994.

[6] S. Floyd and V. Jacobson. Random early detection gateways for congestion avoidance. *IEEE/ACM Transactions on Networking*, 1(4):397–413, August 1993.

[7] D. Fudenberg and J. Tirole. *Game Theory*. MIT Press, 1991.

[8] R. Gibbens and F. Kelly. Resource pricing and the evolution of congestion control. *Automatica*, 35:1969–1985, 1999.

[9] R.J. Gibbens and P.B. Key. The use of games to assess user strategies for differential quality of service in the internet. Technical report, Microsoft Research, Cambridge, November 1999.

[10] P. Key and D. McAuley. Differential QoS and pricing in networks: Where flow control meets game theory. In *IEE Proc Software 146.*, 1999.

[11] K. Leyton-Brown, R. Porter, S. Venkataraman, and B. Prabhakar. Smoothing out focused demand for network resources (short paper). In *ACM Conference on Electronic Commerce*, 2001.

[12] J.K. MacKie-Mason and H. Varian. Pricing the internet. In B. Kahin and J. Keller, editors, *Public Access to the Internet*. Prentice-Hall, 1994.

[13] B. Mitchell. Pricing policies in selected European telephone systems. In *Proceedings of 6th Conference on Telecommunications Policy Research*, pages 437–475, 1978.

[14] A. Odlyzko. A modest proposal for preventing internet congestion. Technical Report TR 97.35.1, AT&T Research, September 1997.

[15] M.J. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, 1994.

[16] R. Pan, L. Breslau, B. Prabhakar, and S. Shenker. Approximate fairness through differential dropping. In *Submitted*, 2001.

[17] R. Pan, B. Prabhakar, and K. Psounis. CHOKe: A stateless active queue management scheme for approximating fair bandwidth allocation. In *Proceedings of IEEE INFOCOM 2000*, pages 942–951, 2000.

[18] S. Shenker. Making greed work in networks: A game-theoretic analysis of switch service disciplines. *IEEE/ACM Transactions on Networking*, 3:819–831, 1995.

## A Proofs of Theorems

**Theorem 1** $(\Phi_1, \varphi_1^*)$ *is optimal.*

**PROOF.** Let $E[R_i|\Phi, \varphi]$ be the expected revenue extracted from agent $a_i$, given mechanism $\Phi$ and equilibrium $\varphi$. First, we prove by contradiction that $\Phi_1$ yields at least as large a $z$ as any other $\Phi$, both given the *same* equilibrium of the respective mechanisms. Assume that there exists a pair ($\Phi$, $\varphi$) such that $z(\Phi, \varphi) > z(\Phi_1, \varphi)$. Since the equilibrium is constant, we can expand $z$ on both sides and simplify to get $E[R|\Phi, \varphi] > E[R|\Phi_1, \varphi]$, which implies $E[R_i|\Phi, \varphi] > E[R_i|\Phi_1, \varphi]$ for at least one agent $a_i$. $\Phi_1$ sets values of $p$, $q$ and $m$ so that for all slots agent $a_i$'s expected utility is $v(s) - E[R_i|\Phi_1, \varphi] = 0$. Thus for $\Phi$ we have $\forall s\ u_i(s) < 0$, implying that $\Phi$ is non-participation-safe, a contradiction.

Second, we consider the case where $\Phi$ and $\Phi_1$ give rise to different equilibria. As described above, under $\varphi_1^*$ agents deterministically distribute themselves so as to give rise to the distribution $d^*$. Recall that $d^*$ is an ideal distribution: $\forall\varphi \left(z(\Phi_1, \varphi_1^*) \geq z(\Phi_1, \varphi)\right)$. Thus, $\forall\Phi, \varphi\ z(\Phi_1, \varphi_1^*) \geq z(\Phi_1, \varphi) \geq z(\Phi, \varphi)$. □

**Theorem 2** *When agents have identical utility functions and no signals are given to agents and the network preselects* $p$ *before agents move, all equilibria are either weak or focused-loading.*

**PROOF.** Consider two agents $a_i$ and $a_j$, without restriction. The network has only three choices with respect to $a_i$'s preferences:

(1) $a_i$ strictly prefers some slot $s_k$ to every other slot. However, every other agent $a_j$ has the same preference. Therefore, no agents will choose any other slot. This is a strict equilibrium, but it is also a focused-loading equilibrium.

(2) $a_i$ will (non-strictly) prefer some slot $s_k$ to all other slots: he will strictly prefer $s_k$ to $s_l$, and will be indifferent between $s_k$ and at least one other slot. Thus no agents will choose slot $s_l$, and $g$ will not be minimized. Any set of mixed strategies over slots between which agents are indifferent will constitute a weak equilibrium.

(3) $a_i$ is indifferent between all pairs of slots $s_k$ and $s_l$. In this case $a_i$ receives the same payment regardless of his action, so randomizing uniformly over all the slots is not a dominated strategy. Indeed, randomization is a weak, load-balancing equilibrium, as shown above.

The only strict equilibrium is a focused-loading equilibrium; all other equilibria are weak. □

**Theorem 3** *There does not exist an optimal* $(\Phi, \varphi)$ *for which* $\varphi$ *is a strict equilibrium and* $m \leq v(\overline{s})$*.*

**PROOF.** We will prove this statement by contradiction. Assume that there exists an optimal $(\Phi, \varphi)$ in which $\varphi$ is a strict equilibrium. Since $\varphi$ is a strict equilibrium, the difference (call it $x$) between expected utility from slot $s$ and the highest expected utility of any other slot must be positive. By the assumption that $m \leq v(\overline{s})$, deviation to $\overline{s}$ would result in no less than 0 utility. Thus by strictness of $\varphi$, agents in slots $s \neq \overline{s}$ have positive expected utility of $x$. If we create $\Phi'$ by altering $P_i$ so that the expected utility of $s$ is decreased by $x$, then the revenue is increased, and it is still an equilibrium (albeit weak) for $a_i$ to select slot $s$. The fact that revenue is higher in $(\Phi', \varphi)$ than $(\Phi, \varphi)$ but that both give rise to the same distribution contradicts the claim that $(\Phi, \varphi)$ is optimal. □

**Lemma 1** $\forall i$ $d_i^\gamma$ *is ideal under* $\Phi_2$*.*

**PROOF.** Define $d_i \geq d_j'$ as $\forall s\, d_i(s) \geq d_j'(s)$. We will prove the following statement that is stronger than the theorem: $\forall j, i \geq j$, there exists an ideal distribution $d_i^*$ such that $d_i^* \geq d_j^\gamma$.

We will first prove this statement by induction on $j$. The base case, where $j = 0$, trivially holds because $\forall s\, d_0^\gamma(s) = 0$.

For the inductive step, assume that there exists a $d_i^*$ for all $i \geq j$ such that $d_i^* \geq d_j^\gamma$, in order to prove that there exists a $d_i^*$ for all $i \geq j+1$ such that $d_i^* \geq d_{j+1}^\gamma$. From the inductive assumption we know that there exists a $d_i^* \geq d_j^\gamma$ for each $i \geq j+1$. Let $s_k = \gamma(j+1)$: hence $s_k = \arg\max_s \Delta(d_j^\gamma, s)$.

We now prove that there exists an ideal distribution $d_i'^*$ consistent with this greedy choice. If $d_i^*(s_k) \geq d_j^\gamma(s_k)+1$, then $d_i'^* = d_i^*$. Otherwise, $d_i^*(s_k) = d_j^\gamma(s_k)$. Consider a slot $s_l$ where $d_i^*(s_l) \geq d_j^\gamma(s_l)+1$. Let $\Upsilon(d, s, c)$ be distribution $d$ but with $c$ agents added to slot $s$. Let $d' = \Upsilon(d_i^*, s_l, -1)$, and let $d'' = \Upsilon(d', s_k, 1)$. We know from the first property of $\Delta$ that $\forall s\, (\Delta(d', s) \leq \Delta(d_j^\gamma, s))$, since $d' \geq d_j^\gamma$. Similarly, from the second property of $\Delta$ we know that $\Delta(d', s_k) = \Delta(d_j^\gamma, s_k)$, since $d'(s_k) = d_j^\gamma(s_k)$. Therefore, $s = s_k$ maximizes $\Delta(d', s)$. This implies that $z(\Phi, d'') \geq z(\Phi, \Upsilon(d', s_l, 1))$. Since $\Upsilon(d', s_l, 1) = d_i^*$ is ideal, $d''$ must also be ideal. Since $d'' \geq d_{j+1}^\gamma$, we have proven the inductive step. □

**Theorem 4** $(\Phi_2, \varphi_2)$ *is* $\varepsilon$*-optimal.*

**PROOF.** First, we prove by contradiction that $\Phi_2$ yields $z$ that is within $n\varepsilon$ of any other $\Phi$, both given the *same* equilibrium. Assume that there exists a pair $(\Phi, \varphi)$ such that $z(\Phi, \varphi) > z(\Phi_2, \varphi) + n\varepsilon$. Since the equilibrium is the same for both mechanisms, we can expand $z$ on both sides and simplify to get $E[R|\Phi, \varphi] > E[R|\Phi_2, \varphi] + n\varepsilon$, which implies $E[R_i|\Phi, \varphi] > E[R_i|\Phi_2, \varphi] + \varepsilon$ for at least one agent $a_i$. $\Phi_2$ sets values of $p$, $q$ and $m$ so that for all slots agent $a_i$'s expected utility is $v(s) - E[R_i|\Phi_2, \varphi] = \varepsilon$. Thus for $\Phi$ we have $\forall s\ u_i(s) < 0$, implying that $\Phi$ is non-participation-safe, a contradiction.

Second, we now consider the case where $\Phi$ and $\Phi_2$ have different equilibria. As shown above in lemma 1, the ideal distribution $d^*$ is achieved by $(\Phi_2, \varphi_2)$, hence $\forall \Phi, \varphi\ z(\Phi_2, \varphi_2) \geq z(\Phi_2, \varphi) \geq z(\Phi, \varphi) - n\varepsilon$. □

**Theorem 5** $(\Phi_3, \varphi_3)$ *is c-optimal for* $c = \max_s\Big(v^u(s) - v^l(s)\Big) + \varepsilon$.

**PROOF.** Define $v^{l+c-\varepsilon}(s) = v^l(s) + c - \varepsilon$: an upper bound on $v^u$ and thus on all possible $v$ functions for agents. We now define variants of $\Phi_3$ based on different agent $v$ functions: $\Phi_3^a$ when agents have different, arbitrary $v$ functions, and $\Phi_3^l$ and $\Phi_3^{l+c-\varepsilon}$ for the cases when all agents' functions are $v^l$ and $v^{l+c-\varepsilon}$, respectively. In each variant we assume that the network has full knowledge of agents' valuations and can set different $p$'s for each agent. Let $d^a$, $d^l$, and $d^{l+c-\varepsilon}$ be the corresponding ideal distributions arising from $\varphi_3$ in their respective mechanisms. The revenue extracted from each agent in equilibrium of $\Phi_3^a$, $\Phi_3^l$ or $\Phi_3^{l+c-\varepsilon}$ is: $(1 - p(s))m = (1 - \frac{v(\overline{s}) - v(s) + \varepsilon}{v(\overline{s})})v(\overline{s}) = v(s) - \varepsilon$. We also make the change that each of these variants of $\Phi_3$ sets $\varepsilon = 0$ when it determines $p^b$. This has the consequence that equilibrium $\varphi_3$ still holds but is no longer strict. Each variant will then extract the full $v(s)$ from each agent in $\varphi_3$. Each of these mechanism-equilibrium pairs is optimal, following an argument analogous to the one given in the proof of theorem 1 (not given here): the mechanism makes each agent pay exactly his valuation, and achieves an ideal distribution. Thus, for any set of arbitrary $v$ functions that $\Phi_3$ encounters, $z(\Phi_3^a, \varphi_3)$ represents the optimal evaluation. We now bound how far $\Phi_3$ can be from this amount.

By definition, $z(\Phi_3^l, \varphi_3) = g(d^l) + \sum_i v^l(s_i)$. We know that $d^{l+c-\varepsilon} = d^l$ because $v^{l+c-\varepsilon}$ differs only by a constant from $v^l$ at each slot. Thus, $z(\Phi_3^{l+c-\varepsilon}, \varphi_3) = z(\Phi_3^{l+c-\varepsilon}, \varphi_3) = g(d^l) + \sum_i v^{l+c-\varepsilon}(s_i) = g(d^l) + \sum_i v^l(s_i) + (c - \varepsilon)n$. This implies that $z(\Phi_3^l, \varphi_3) + (c - \varepsilon)n = z(\Phi_3^{l+c-\varepsilon}, \varphi_3)$; it remains to show that $z(\Phi_3^{l+c-\varepsilon}, \varphi_3) \geq z(\Phi_3^a, \varphi_3)$. Note that $z(\Phi_3^{l+c-\varepsilon}, \varphi_3) \geq z(\Phi_3^{l+c-\varepsilon}, \varphi_3)$ by definition of $d^l$. Also, $z(\Phi_3^{l+c-\varepsilon}, \varphi_3) \geq z(\Phi_3^a, \varphi_3)$ because $v^{l+c-\varepsilon}$ is an upper bound on each of the $v$'s in the case of $\Phi_3^a$ and $g(d^a)$ is common to both terms. Thus $z(\Phi_3^l, \varphi_3) + (c - \varepsilon)n = z(\Phi_3^{l+c-\varepsilon}, \varphi_3) \geq z(\Phi_3^a, \varphi_3)$. Now we return to the real $\Phi_3$. The optimal distribution is $d^l$, and in $\varphi_3$ the network extracts $\varepsilon$ less revenue from each agent than $\Phi_3^l$ did because it does not set $\varepsilon = 0$. Thus, $z(\Phi_3, \varphi_3) + n\varepsilon = z(\Phi_3^l, \varphi_3)$. Combining the last two equations, we can conclude: $z(\Phi_3, \varphi_3) + cn \geq z(\Phi_3^a, \varphi_3)$, and thus that $\varphi_3$ is $c$-optimal. □