Group Decision and Negotiation **12**: 361–395, 2003

Nash Equilibrium and Decentralized Negotiation in Auctioning Divisible Resources

RAJIV T. MAHESWARAN AND TAMER BAŞAR
Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL, USA

Abstract

We consider the problem of software agents being used as proxies for the procurement of computational and network resources. Mechanisms such as single-good auctions and combinatorial auctions are not applicable for the management of these services, as assigning an entire resource to a single agent is often undesirable and appropriate bundle sizes are difficult to determine. We investigate a divisible auction that is proportionally fair. By introducing the notion of price and demand functions that characterize optimal response functions of the bidders, we are able to prove that this mechanism has a unique Nash equilibrium for an arbitrary number of agents with heterogeneous quasilinear utilities. We also describe decentralized negotiation strategies which, with appropriate relaxation, converge locally to the equilibrium point. Given an agent with a sequence of jobs, we show how our analysis holds for a wide variety of objectives.

1. Introduction

We consider the *information economy* which is described by Kephart et al. (2000), as the merging of traditional markets, the Internet and autonomous agents to form a new marketplace where agents serve as proxies for buyers, sellers and intermediaries. Evidence of this can already be seen on the Internet. Prominent web sites such as Yahoo!, Amazon.com and Ebay are hosts to auctions where participants have the ability to give simple agents information about their valuations and have them bid incrementally for items of interest. Software agents facilitate the gathering of information in a low-cost and timely fashion in addition to assisting with other aspects of markets such as negotiation and payment. Kasbah (Chavez and Maes 1996) is an agent-mediated marketplace where the process of buying and selling is automated by the creation of autonomous software agents. The WALRAS algorithm is another development that calculates competitive equilibrium for agents who submit demand functions for single goods (Cheng and Wellman 1998).

Markets are important not only because they are the mechanisms for the exchange of many traditional goods but they have also emerged as a new paradigm for managing and allocating resources in complex systems. Among the many features that make them attractive is the establishment of currency, which allows for a common valuation of heterogeneous resources. This gives managers or agents the ability to specify preferences or establish priority. Markets are appropriate for decentralized systems because once a currency exchange protocol is established, negotiations can occur simultaneously at various nodes without the necessity of a central authority. Scalability is another advantage as new resources and new resource users can be added simply by establishing the ability to receive or give currency. Also, prices serve as useful low-dimensional feedback for control. Market-based control has been applied to factory scheduling, manufacturing systems, energy distribution and pollution management (Clearwater 1996).

We focus on markets for network bandwidth and computational resources such as processor share allocation. Mackie-Mason and Varian have advocated the application of economic mechanisms for networks, proposing usage-based pricing of the Internet using "smart markets" (MacKie-Mason and Varian 1993). They examined pricing as a mechanism for managing congestion in resources such as routers, ftp servers and the Web (MacKie-Mason and Varian 1995). This has led to the idea of packet marking and charging small amounts for marked packets (Gibbens and Kelly 1999), which has motivated a large body of work in modeling packets streams in an economic context as agents with utility functions. This trend of treating networks as markets has led to alternate methods of provisioning bandwidth. Companies such as Arbinet, RateXchange and Band-X.com have introduced exchange markets and bandwidth trading via auctions. A current push is to move toward dynamic real-time bandwidth trading. Invisible Hand Networks has created a product that allows for distributed real-time auctioning of Internet bandwidth.

Similar methods have been suggested for allocation of computational resources. Gagliano et al. have simulated auction-based allocations for scheduling tasks requiring processing (1995). POPCORN (Regev and Nisan 1998) and Spawn (Waldspurger et al. 1992) are two often referenced systems that use markets for computational resource allocation. This was extended to lottery based scheduling where tickets that were purchased determined the proportional share of processor allocated to a particular job (Waldspurger and Weihl 1995).

If auctions are to be used to dynamically allocate network bandwidth and computational resources, we must decide which mechanisms are appropriate. To find a meaningful auction mechanism, we must first classify the good for sale. Clearly, mechanisms for the sale of a single good do not apply to computational resources and network bandwidth. It is rare that these resources are allocated totally to a single user. Even if an agent chooses to purchase an entire resource, we expect at some point that the resource will be partitioned, and it is that exchange that we are interested in analyzing. There has been a lot of research in analyzing combinatorial auctions for multi-unit goods (de Vries and Vohra, to be published; Sandholm 1999; Wellman et al. 2001). However, these methods are inappropriate for the goods we are considering because network and computational resources are rarely partitioned into well-defined bundles that can be bid for in discrete quantities. Thus, we consider divisible or share auctions as a market mechanism. Though network bandwidth and computational resources are not continuously divisible, they are usually available in such high quantities (Mb, GHz) that the approximation is valid.

First, we must decide on what fairness principle our allocation must satisfy. One measure of fairness that existed early in network literature is the notion of *max-min fairness* (Bertsekas and Gallager 1991), where an allocation is chosen so that no agent could improve without simultaneously reducing an agent allocated at a lower level. This leads to equal shares in a single resource, and it has been argued that it is not an appropriate measure for networks. Kelly has introduced a notion of (weighted) *proportional fairness* where an allocation is made such that the sum of (weighted) proportional gains cannot be increased

(Kelly 1997; Kelly, Mauloo, and Tan 1998). This notion has generated momentum and work has been done to show that proportional fairness can be achieved within current protocols with the use of both fixed window and dynamic window adjustment schemes (La and Anantharam 2000; Massoulie and Roberts 1999). In computational resources, this notion of proportional allocation also exists as the lottery scheduling mechanisms partition resources in direct proportion to the tickets owned, where the tickets are analogous to weights. Because of the ease of implementability, proportional share systems are often advocated for resource allocation (Maheshwari 1995; Stoica et al. 1996). Thus, we wish to incorporate proportional allocation in our mechanism.

Another issue we choose to address is the cost of computation. As an agent economy for our network and computational resources evolves, large numbers of agents will be migrating throughout our resources at a rapid pace, with allocations that may have to be recomputed on the order of microseconds. Thus, the signaling load and the computation required to perform the allocation will have a significant impact on the implementability of the allocation scheme. We seek to design an auction that minimizes these effects. We address this by choosing single dimensional bids, which minimizes the signaling load. Partitioning the resource proportionally with respect to bid requires a computational load on the order of the number of agents present.

Another concern that we wish to address is the possibility of the lying auctioneer (Sandholm 1996). In a Vickrey auction, the buyer must trust that the second price is reported accurately, unless preferences of other agents are made public. The revelations of all bids might not be desired by the agents and can also be infeasible for rapid auctions with large numbers of participants. We desire a mechanism where an agent can verify an accurate allocation for its bid without additional signaling or unnecessary violation of private information. We show that for the proportionally fair auction, the network feedback is verifiable.

Inherent in the settings we are considering is the competition among agents attempting to gain access to limited computational and network resources. With the use of auction mechanisms, the performance of each agent is affected by the actions of all other agents. The autonomy of agents creates an environment where each agent is acting to better its own utility. The nature of this negotiation and the attempt to find an operating point calls for game theory (Başar and Olsder 1999; Owen 1995; Petrosjan and Zenkevich 1996). The rationality that game theory assumes of all its players, which may not always hold true with humans, is particularly fitting in the realm of electronic technology as software agents or network protocols do not deviate from the functionality they are given. To reach an equilibrium, agents must exchange information about their preferences with the resource. Requiring agents to transmit entire preference functions would yield an information revelation cost that has an unreasonably high signaling load. Typically, neither the resource nor agents have access to other participants' private information. Therefore, iterative algorithms based on network feedback that converge to stable operating points are often needed, which is what we pursue here. Our main contributions are a game-theoretic analysis of the proportionally fair auction for which we obtain a unique Nash equilibrium and decentralized update schemes that converge to the equilibrium point. We also show that this analysis holds for a wide variety of agent objectives.

The paper is organized as follows. In Section 2, we describe the allocation mechanism, model agent utilities and obtain optimal responses. In Section 3, we introduce the notion of a price function as a useful characterization of agent responses. In Section 4, we prove there is a unique Nash equilibrium for the mechanism. In Section 5, we introduce decentralized negotiation protocols and derive the conditions under which these update schemes converge locally. In Section 6, we show how one can obtain quasilinear utilities for agents whose utilities are alternatively defined. In Section 7, we introduce a scenario where each agent has a sequence of tasks, and we analyze it when agents want to minimize both the cost and time to complete the itinerary. In Section 8, we investigate the scenario where the itinerary has to be completed as fast as possible with a finite total budget. In Section 9, we investigate the case where agents have a time deadline and want to meet it with the lowest cost. Finally, in Section 10, we present some concluding thoughts and discuss some areas for further investigation.

2. Allocation mechanism and agent utility

We begin with N agents competing for a resource with fixed finite capacity. The resource is allocated using a market mechanism, where the partitions depend on the relative signals or bids sent by the agents. We assume that each agent submits a signal s_i to the resource. Then, $s = [s_1 \ldots s_N]$ represents all bids of competing agents. If $s_i \in \mathbf{R}$, then s is a vector of N elements. If s_i is of higher dimension, then s is an $M \times N$ matrix where M is the dimension of the column vector s_i. Because minimizing cost of communication is of interest to us, we restrict our analysis to one-dimensional signals. A divisible auction consists of two mappings. The first is from the bids, s, to a partition, $x(s)$, where $x_i(s) \in [0, 1]$ is the resource share allocated to the i-th bidding agent. The second is from the bids, s, to a cost vector, $c(s,x)$ where $c_i(s,x)$ is the cost associated with the i-th agent obtaining $x_i(s)$. The choices of x and c define the auction mechanism.

We want our allocations to be *proportionally fair* by weight. This holds if the allocation x^* satisfies:

$$\sum_{i=1}^{N} s_i \frac{x_i - x_i^*}{x^*_i} \leq 0$$

for any x where $\sum_{i=1}^{N} x_i = 1$ where s_i's denote the weights. This can be achieved with the following allocation rule:

$$x_i(s) = \frac{s_i}{\sum_j s_j}. \tag{1}$$

We note that this not only satisfies the proportionally fair criterion for allocation in networks, but it also matches the proportional share allocations in many of the systems proposed for computational resources.

In terms of cost of computation, we note that it takes $O(N)$ operations to perform the allocation presented in (Equation 1), which is the minimal cost for making variable allocations to N agents. The cost for each agent is

$$c_i(s, x) = s_i.$$

In this auction, if the feedback from the resource is the sum of all bids, an agent can immediately verify if it has been given an accurate allocation. If an agent knows the received allocation x_i and its own bid s_i, any bid total suggested by the auctioneer other than s_i/x_i can be immediately identified as a signal of an inaccurate allocation or a lying auctioneer. Furthermore, under this cost structure, each agent pays the same price per unit resource received.

We assume that each agent has a valuation $v_i(x_i)$ for receiving an allocation x_i. This valuation may be a characterization of the estimated performance as a function of a given share of the resource. For example, it could be the time to complete the processing of a job in a computational market, or the time to transmit data given a particular share of network bandwidth obtained. Each performance measure is translated into an equivalent value that can be compared with cost. Another derivation of the valuation could come from the estimated value of the sales that could be generated by obtaining a given share of the resource. This could be the case where agents act as brokers for computational resource and network bandwidth. We make the following assumptions about agents' valuations:

Assumption 1. For all $i \in \{1, \ldots, N\}$

– $v_i(x_i)$ is continuously differentiable
– $v_i'(x_i) > 0 \ \forall\, x_i \in (0, 1)$
– $v''_i(x_i) \le 0 \ \forall i\, x_i \in (0, 1)$.

The first of these captures the fact that an agent's performance or marginal valuation of performance should not change dramatically given a miniscule change in allocation. The second one is intuitive as an agent's valuation should increase with allocation, and the third one captures the effect of diminishing returns. Each agent's utility is the difference between the valuation of the allocation it received and cost of its bid:

$$U_i(s) = v_i(x_i(s)) - c_i(s, x).$$

Substituting from Equations (1) and (2), we have:

$$U_i(s) = U_i(s_i;s_{-i}) = v_i\left(\frac{s_i}{s_i + s_{-i} + \in}\right) - s_i, \qquad (3)$$

where $s_{-i} = \sum_{j=1}^{N-1} s_j - s_i$ is the sum of the bids of all agents excluding the i-th agent and $s_N = \in$ is a bid made by an agent representing the resource. By bidding $\in$, the resource has a way of declaring a reservation value for its resource and prevents the possibility of agents colluding to purchase the resource for an arbitrarily small amount of money. Each agent wishes to choose the bid that maximizes its utility. Given that all other agents bid s_{-i}, we want to find the best response $s_i = f_i(s_{-i})$. The first-order necessary condition for a maximizing interior solution is:

$$U_i'(s_i;s_{-i}) = v_i'(x_i(s))x_i'(s) - 1 = v_i'\left(\frac{s_i}{s_i + s_{-i} + \epsilon}\right)\frac{s_{-i} + \epsilon}{(s_i + s_{-i} + \epsilon)^2} - 1 = 0.$$

This can be rewritten as follows:

$$v_i'\left(\frac{s_i}{s_i + s_{-i} + \epsilon}\right)\frac{(s_i + s_{-i} + \epsilon)^2}{s_{-i} + \epsilon} = 0.$$

The LHS of the above equation is a decreasing function of s_i as $v'_i(\cdot)$ is decreasing in its argument, $s_i/(s_i + s_{-i} + \epsilon)$ is an increasing function of s_i and the second term has s_i only in the numerator. Thus, an interior solution exists if and only if the LHS is positive when $s_i = 0$. An agent will participate in the auction (submit a nonzero bid), if and only if:

$$v_i'(0) > s_{-i} + \epsilon. \tag{4}$$

Looking at the second-order condition, we have:

$$U_i''(s_i; s_{-i}) = v_i''\left(\frac{s_i}{s_i + s_{-i} + \epsilon}\right)\frac{(s_{-i} + \epsilon)^2}{(s_i + s_{-i} + \epsilon)^4} + v_i'\left(\frac{s_i}{s_i + s_{-i} + \epsilon}\right)\frac{-2(s_{-i} + \epsilon)}{(s_i + s_{-i} + \epsilon)^3} \tag{5}$$

which is negative due to Assumption 1, the nonnegativity of the bids, and the assumption that $\epsilon > 0$. If (Equation 4) is satisfied, any s_i that solves (Equation 4) uniquely maximizes the agent's utility and is the agent's unique response when the total of all other agents' bids is s_{-i} and the resource bids ϵ. If (Equation 4) is not satisfied for a particular s_{-i}, the i-th agent will bid zero. Thus, any agent with quasilinear utilities will have a unique optimal response to a fixed bid total from all the other agents. In the remainder of this chapter, for notational simplicity, we will assume that the resource's bid is captured in the term s_{-i}.

3. Price functions

Even though the resource allocation is accomplished via an auction mechanism, we note that ultimately each agent pays the same price per unit resource obtained. The auction can then be interpreted as a resource sold at a uniform price where the price is determined by the agents. The price per unit of the resource is $\Sigma_i s_i$, and each agent receives an allocation that is the ratio of its bid to that price. We can then define the *price* of a resource as the sum of all the bids, $p := \Sigma_i s_i$, including the resource's bid, for that resource.

Next we define a *price function*, $p_i(x_i)$: $\mathbf{R} \to \mathbf{R}$ as the price at which the agent would choose an allocation of x_i. The price function represents the set of cost-allocation pairs which are the unique optimal responses of a given agent over a range of bids of other agents, i.e., $s_i = p_i(x_i)\, x_i$ is the unique optimal response to $s_{-i} = p_i(x_i)(1 - x_i)$. The inverse of the price function is the *demand function*, $d_i(p)$: $\mathbf{R} \to \mathbf{R}$, which is defined as the quantity of resource that the agent would desire if the price was p. This is again generated by an agent's unique optimal response in a way such that $s_i = d_i(p)p$ is the agent's reaction to $s_{-i} = (1 - d_i(p))p$. The price and demand functions are expected to be differentiable decreasing functions of their argument and the existence of one implies the existence of a well-defined inverse. One way to obtain these functions is to take the optimal response $s_i = f_i(s_{-i})$, substitute $s_{-i} = p - s_i$, and solve the fixed-point equation $s_i = f_i(p - s_i)$. If a solution exists, one has s_i as a function of p. Then, making the substitution $s_i = p\, x_i$, one can obtain an equation in terms of x_i and p from which the price and demand functions can be obtained. However, due to the nature of our auction, we can obtain the price function directly from an agent's valuation.

Proposition 1. Given a valuation $v_i(x_i)$ that satisfies Assumption 1, there exists a corresponding differentiable decreasing price function characterized by $p_i(x_i) = v'_i(x_i)(1 - x_i)$.

Proof. Let $f_i(s_i)$ be the i-th agent's unique optimal response. By the first-order necessary condition, we have:

$$f_i(s_{-i}) + s_{-i} = v'_i\left(\frac{f_i(s_i)}{f_i(s_{-i}) + s_{-i}}\right) \frac{s_{-i}}{f_i(s-_i) + s_{-i}}$$

$$= v'_i\left(\frac{f_i(s_i)}{f_i(s_{-i}) + s_{-i}}\right)\left(1 - \frac{s_i}{(f_i(s_{-i}) + s_{-i}}\right).$$

By the definition of price, $p_i = f_i(s_{-i}) + s_{-i}$, and the allocation rule states $x_i = s_i/(f_i(s_{-i}) + s_{-i})$. Substituting this above, we have:

$$p_i(x_i) = v'_i(x_i)(1 - x_i).$$

We see that p_i is differentiable, with derivative:

$$p'_i(x_i) = v''_i(x_i)(1 - x_i) - v'_i(x_i)$$

which is strictly negative given Assumption 1. Thus, the price function is decreasing. ■

This property of the auction lets us go directly from knowing an agent's valuation to the price function which is a transformation of its optimal response. We can obtain the optimal bid from the price function as follows:

$$s_i = f_i(s_{-i}) = f_i(s_{-i}) \frac{f_i(s_{-i}) + s_{-i}}{f_i(s_{-i}) + s_{-i}} = x_i p_i(x_i) = v_i'(x_i)\,(1 - x_i)x_i\,.$$

We see that $p_i(0) = v'_i(0)$ which states that if the price is greater than its largest marginal valuation, the agent will choose not to participate. This reflects the condition stated in Equation 4 derived from the first-order necessary conditions. We also see that $p_i(1) = 0$, which states that the agent will purchase the entire resource if only the price is zero. This is equivalent to saying that the agent will demand the entire resource if the price was zero. This is a result of the structure of the auction where the only way an agent can obtain the entire resource is to be the only bidder, in which case the agent would make an arbitrarily small bid. This can never happen with the resource itself bidding $\in$. If the allocation was at equilibrium, the price per unit resource that the agent would be paying is less than its marginal utility by a factor of $(1 - x_i)$. This is the benefit gained by the agent for knowing its own effect on the price of the resource. Agents with larger allocations at equilibrium are able to scale their costs away from their marginal utility to a larger degree. In the case where there are many agents and each agent receives a small portion of the resource, i.e, $x_i << 1$, the prices being paid will be very close to the marginal valuations. The form of the price function also reflects the fact that shifting the valuation function by a constant will not change the optimal response, as the price function (and hence the agent's reaction function) depends only on the marginal valuation and not on the absolute valuation.

Example 1. We derive the price and demand functions for an agent with quasilinear utility and $v_i(x_i) = 2\,x_i$.

The utility function for this agent is $U_i(s_i;s_{-i}) = v_i(x_i(s)) - s_i = 2s_i/(s_i + s_{-i}) - s_i$. The marginal utility is $U'_i(s_i;s_{-i}) = 2s_{-i}/(s_i + s_{-i})^2 - 1$. To have a non-zero bid, we need $U'_i(0;s_{-i}) > 0$, which is equivalent to $v_i(0) = 2 > s_{-i}$. If this condition is met, we can solve $U'_i(s_i;s_{-i}) = 0$ to obtain the optimal response $s_i = f_i(s_{-i}) = \sqrt{2s_{-i}} - s_{-i}$. This means that if all other agents bid a total of 1, the i-th agent would bid $\sqrt{2} - 1 \approx 0.4142$. Letting p be the sum of the bids or "price", we can equivalently state that the price-allocation pair $p = 1.4142$, $x_i = 0.4142/1.4142 = 0.2929$ is an optimal state for the i-th agent. For every optimal bid pair $(s_{-i}, f_i(s_{-i}))$ there is an equivalent price-allocation pair and vice-versa. The first-order condition $U'_i(s_i;s_{-i}) = 0 \Rightarrow 2s_{-i} = (s_i + s_{-i})^2$. By making the substitutions $p = s_i + s_{-i}$ and $s_{-i} = p(1 - x_i)$, we obtain the price function $p = 2(1 - x_i) = v'_i(x_i)\,(1 - x_i)$. The demand function is $x_i = 1 - p/2$. Both these functions capture all the price-allocation pairs that are optimal for the i-th agent. □

Other sample valuation functions that satisfy Assumption 1 along with price and demand functions that display the optimal price-allocation pairs are given in Table 1. The valuation functions are plotted in Figure 1, the price functions are shown in Figure 2, and the demand functions are shown in Figure 3.

Table 1. Valuation, Price and Demand Function Equivalence

Agent	$v_i(x_i)$	$p_i(x_i)$	$d_i(p)$
1	$\log(x_i)/10$	$(1 - x_i)/(10\, x_i)$	$1/(1 + 10p)$
2	$\log(1 + x_i)$	$(1 - x_i)/(1 + x_i)$	$(1 - p)/(1 + p)$
3	x_i	$1 - x_i$	$1 - p$
4	$-1/(1 + x_i)$	$(1 - x_i)/(1 + x_i)^2$	$(-2p - 1 + \sqrt{1+8p})/(2p)$
5	$-1/(10\, x_i)$	$(1 - x_i)/(10x_i^2)$	$(-1 + \sqrt{1 +40\, p})/(20p)$

4. Nash equilibrium

An immediate question is whether there is an allocation of the resource at a price where all agents participating in the auction are satisfied. In the language of game theory, we ask whether there is a set of bids $\{s_i^*\}_{i=1}^N$, where N is the number of agents competing for the desired resource, such that no single agent wishes to deviate from its bid given that the other agents remain the same. This state, a Nash equilibrium, occurs if no agent can improve its quality by changing its bid under current market conditions, i.e.,

$$s_i^* = \arg\max_{s_i} U_i\,(s_i;\, s_{-i}^*)\ \forall\, i \in \{1, \ldots, N\},$$

where s_{-i}^* implies $s_j = s_j^*\ \forall\, j \neq i$. Because every agent's optimal response is captured in its price and demand functions, we can use these as tools to evaluate the existence of a Nash equilibrium.

We find it useful to work in the space of demand functions. Due to the structure of our auction, the total of the allocated resources will always be one. Given a particular price p, if the sum of all agent demands at that price, $\Sigma_i\, d_i(p)$, is not equal to one, then there is no

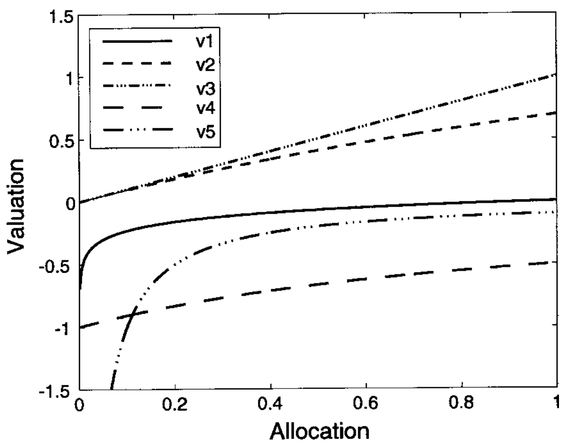

Figure 1. Sample Valuation Functions.

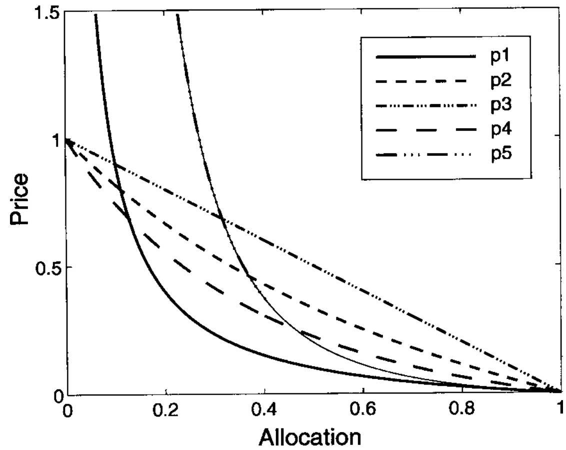

Figure 2. Sample Price Functions.

way to partition the resource without giving at least one agent an allocation such that $x_i \neq d_i(p)$. This implies that $s_i \neq \arg\max_t U_i(t;s_{-i})$ and the agent would gain by unilaterally changing its bid. Thus, to find a Nash equilibrium, it is equivalent to ask whether there is a price (or bid total) where the total demand of all the agents at that price is equal to one. Valid demand functions for elastic agents are assumed to be decreasing functions of price that go to zero as the price tends to infinity. We know that this holds for quasilinear utility functions with valuations that satisfy Assumption 1.

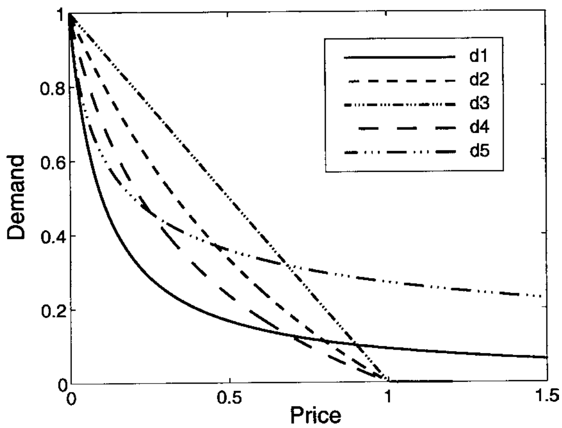

Figure 3. Sample Demand Functions

Proposition 2. Given any set of demand functions $\{d_i(\theta)\}_{i=1}^N$, where $\Sigma_{i=1}^N d_i(0) > 1$, $\lim_{\theta\to\infty} d_i(\theta) = 0, \forall i$, and $d_i(\theta_1) > d_i(\theta_2)\ \forall\ \theta_1, \theta_2$ such that $\theta_1 < \theta_2$ for $i = 1, \ldots, N$, there exists a unique value θ^* such that $\Sigma_{i=1}^N d_i(\theta^*) = 1$.

Proof. Let $\bar{d}(\theta) = \Sigma_{i=1}^N d_i(\theta)$. Then $\bar{d}(\theta)$ is a continuously decreasing function, whose maximum is $\bar{d}(\theta) > 1$. We also have $\lim_{\theta\to\infty} \bar{d}(\theta) = 0$, which implies that for some $\bar{d}\,\theta$ sufficiently large, $\bar{d}(\bar{\theta}) < 1$. Applying the Intermediate Value Theorem for $\bar{d}(\theta)$ on $[0,\bar{\theta}]$, we know that there exists at least one θ^* such that $\bar{d}(\theta^*) = \Sigma_{i=1}^N d_i(\theta^*) = 1$. Let us assume that there are at least two values of θ where $\bar{d}(\theta) = 1$. Let us choose two of these values as θ_1^* and θ_2^*, where $\theta_1^* < \theta_2^*$. Then, we have $d_i(\theta_1^*) > d_i(\theta_2^*)\ \forall\ i = 1, \ldots, N$, which implies that $\bar{d}(\theta_1^*) > \bar{d}(\theta_2^*)$. But we have $\bar{d}(\theta_1^*) = \bar{d}(\theta_2^*) = 1$, which is a contradiction and thus we can have only one θ where $\bar{d}(\theta) = \Sigma_{i=1}^K d_i(\theta) = 1$. ■

The individual demand functions for agents described in Table 1, their total demand and the resulting equilibrium price are shown in Figure 4. By working in the space of demand functions, we can use the property that the demands are decreasing to easily see that there is a unique Nash equilibrium. Uniqueness of the Nash equilibrium is significant as we have a single desired operating point. Thus, given any set of agents there is a unique set of bids that yield an allocation where each agent is satisfied. This set of bids can be characterized in terms of the demand functions and a Nash equilibrium price, θ^*, as follows:

$$\{s_i : s_i = d_i(\theta^*)\ \theta^*\}_{i=1}^N.$$

The condition $\Sigma_{i=1}^N d_i(0) > 1$ is satisfied for almost all agents as (for price function p), $p(1) = 0 \Rightarrow d(0) = 1$ unless the marginal valuation at one is infinite which will not occur for any reasonable valuation. This also requires that $N > 2$, and this is always satisfied as we have

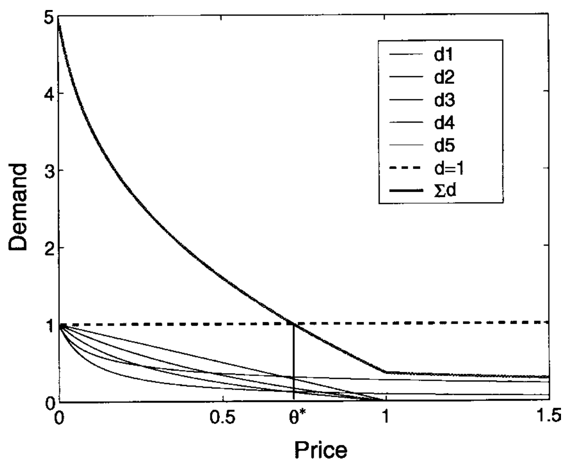

Figure 4. Nash Equilibrium for Agents described in Table 1.

the bids of the resource and at least one agent requesting service. Clearly, θ* determines which agents receive service as any agent with $d(\theta^*) = 0$ will have a zero bid as its optimal response.

5. Decentralized bidding algorithm

Knowing that there is a unique Nash equilibrium, the natural question that follows is how to arrive at that allocation. If the demand functions of all the agents, $d_i(\theta)_{i=1}^N$, were communicated to the resource, it could calculate the equilibrium allocation by a binary search over θ and enforce it immediately. However, this would add a significant signaling load which we want to avoid. Also, if the resource is operating as a profit maker as opposed to a mediator, agents would not want to reveal their private information. Thus, it would be desirable if the agents could reach the Nash equilibrium allocation in a decentralized manner.

For agents to make decentralized updates, they require some information. We assume that each agent is aware of the share of the resource that it currently receives. Also, the resource can provide the current price (or equivalently, the total of all bids) for the resource. If, at time slot n, the i-th agent bids s_i^n, then the price would be $\Sigma_{i=1}^N s_i^n$ and the i-th agent would receive $s_i^n / \Sigma_{i=1}^N s_i^n$ of the resource. This feedback from the resource prevents the possibility of the lying auctioneer that exists in second price auctions. Any agent can verify the price being announced by the resource as being valid by comparing it to the ratio of its bid to its allocation, which are both known to the agent. To obtain a viable decentralized algorithm, we seek a set of update policies $\{f_i\}_{i=1}^N$ such that if $s_i^{n+1} = f_i(s^n)$, where $s^n = [s_1^n\, s_2^n \cdots s_N^n]$, then $\lim_{n\to\infty} s_i^n = s_i^* = d_i(\theta^*)\,\theta^*$, $i = 1, \ldots, N$, where θ^* is a Nash equilibrium price. After bids are made by all the agents, the i-th agent will receive a feedback pair

$$(\theta, x_j) = \left(\sum_{j=1}^{N} s_j, \frac{s_j}{\Sigma_{j=1}^K s_j} \right)$$

which denotes the congestion for that current time slot and the service rate received. The agent knows that if this pair does not lie on the curve $(\theta, d_i(\theta))$ or equivalently $(p_i(x_i), x_i)$, the current price-allocation pair is not optimal. Thus, any viable update algorithm must project the feedback pair to a point on the demand or price curve. If a user receives allocation x_i for some bid, it would project to the point $(p_i(x_i), x_i)$ on the price curve and the corresponding bid would be $s = p_i(x_i)\, x_i$.

We propose an update algorithm where each agent projects the feedback point vertically onto the price function, as shown in Figure 5. This method has the advantage that the agent does not even require the feedback of the current price of the resource. Each agent projects its allocation to the price it would desire for the current allocation and makes the appropriate bid. This further reduces the signaling load required by the auction. It also eliminates the need to worry about the truthfulness of the auctioneer. This leads to the following decentralized update scheme:

$$s_i^{n+1} = (s_i^n / \bar{s}^n)\, p_i\, (s_i^n / \bar{s}^n) \tag{6}$$

where n and $n + 1$ denote the two consecutive iteration stages and $\bar{s}^n := \Sigma_{i=1}^K s_i^n$. Simulations have shown that the above update scheme does not converge for all price functions. An alternative then is the following relaxed version of the update scheme:

$$s_i^{n+1} = \alpha_i\, (s_i^n / \bar{s}^n)\, p_i (s_i^n / \bar{s}^n) + (1 - \alpha_i)\, s_i^n \tag{7}$$

where $\alpha_i \in (0, 1]$, which also covers the unrelaxed case ($\alpha_i = 1$). A graphical interpretation of the relaxed update scheme can be seen in Figure 6. This update scheme depends on knowing only the received allocation, $s_i^n / \bar{s}_i^n$, and the previous bid s_i^n. The relaxed version of the update scheme requires no additional signaling and only requires that each agent store its last bid in memory. As α_i approaches zero, the time to convergence will delay as bids change more slowly. Thus, we desire to find the largest α_i that the i-th agent should use that will make the algorithm stable.

To investigate local stability, we linearize the update algorithm around the equilibrium bids $\{s_i^*\}$ to get

$$s_i^{n+1} - s_i^* = \sum_{j=1}^{N} \left.\frac{\partial s_i^{n+1}}{\partial s_j^n}\right|_{s=s^*} (s_j^n - s_i^*)$$

where

$$\frac{\partial g_i}{\partial s_i} = \alpha_i \left[\frac{(\bar{s} - s_i)}{\bar{s}^2}\, p_i \left(\frac{s_i}{\bar{s}}\right) + \frac{s_i}{\bar{s}}\, p_i' \left(\frac{s_i}{\bar{s}}\right) \frac{(\bar{s} - s_i)}{\bar{s}^2} \right] + (1 - \alpha_i)$$

$$\left.\frac{\partial g_i}{\partial s_i}\right|_{s=s^*} = \alpha_i \left[(1 - x_i^*) \left(\frac{p_i(x_i^*) + x_i^* p_i'(x_i^*)}{\theta^*} \right) \right] + (1 - \alpha_i)$$

$$\frac{\partial g_i}{\partial s_i} = \alpha_i \left[\frac{(- s_i)}{\bar{s}^2}\, p_i \left(\frac{s_i}{\bar{s}}\right) + \frac{s_i}{\bar{s}}\, p_i' \left(\frac{s_i}{\bar{s}}\right) \frac{(- s_i)}{\bar{s}^2} \right]$$

$$\left.\frac{\partial g_i}{\partial s_j}\right|_{s=s^*} = \alpha_i \left[(- x_i^*) \left(\frac{p_i(x_i^*) + x_i^* p_i'(x_i^*)}{\theta^*} \right) \right]$$

Let us define $\in_i^n := s_i^n - s_i^*$, and

$$q_i := \frac{p_i(x_i^*) + x_i^* p_i'(x_i^*)}{\theta}$$

where x_i^* is the equilibrium allocation for the i-th agent, $p_i'(\cdot)$ is the derivative of the price function $p_i(\cdot)$, and θ^* is the equilibrium bid total. We have $p_i(x_i^*) \leq \theta^*$, as any agent that receives a positive equilibrium allocation will have a price function that yields the equilibrium price at its allocation and any agent that receives a zero allocation must have $p_i(0) \leq \theta^*$. Furthermore, we have $p_i'(x_i^*) < 0$, which implies $q_i \leq 1$. We assume that all agents are restricted to those whose price or demand functions ensure that the agents are neither infinitely sensitive nor completely insensitive to the price at equilibrium. This will be satisfied if $q_i \neq -\infty$ and $q_i \neq 1$. The linearized system is $\in^{n+1} = \tilde{J} \in^n$ where

$$\tilde{J} = AJ + (I - A), \quad J = (I - X)Q, \; X = x^T 1_N, \quad x = [x_1^* x_2^* \ldots x_N^*], \quad 1_N = [1\ 1 \ldots 1],$$

$$Q = \begin{bmatrix} q_1 & 0 & \cdots & 0 \\ 0 & q_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & q_N \end{bmatrix}, \qquad A = \begin{bmatrix} \alpha_1 & 0 & \cdots & 0 \\ 0 & \alpha_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \alpha_N \end{bmatrix}.$$

Proposition 3. If α_i is chosen such that $\alpha_i < 2/(1 - q_i)$, $\forall_i$, then all the eigenvalues of $\tilde{J}$ are in the unit circle and the update scheme described by Equation 7 is locally stable.

Proof. Let λ be an eigenvalue of $\tilde{J}$, and y a corresponding eigenvector. We have

$$\tilde{J}x = (A(I - X)Q + (I - A))y = \lambda y.$$

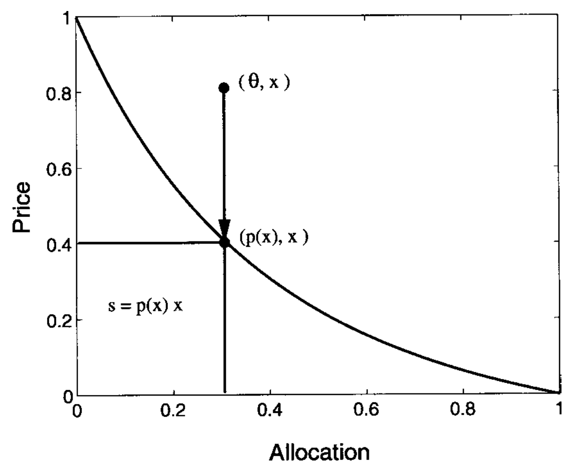

Figure 5. Projection of Allocation x on Price Function $p(x)$.

Multiplying from the left by Q, we have

$$(QA(I - X)Q + Q(I - A))y = \lambda \, Qy.$$

Since I,Q and A are diagonal matrices, we have the following relations

$$QA = AQ, \quad Q(I - A) = (I - A)Q$$

and thus,

$$[AQ(I - X) + (I - A) \, Qy = \lambda \, Qy \qquad [AQ - AQX + I - A] \, r = \lambda r$$

where $r = Qy$. Let $\bar{r} := 1_N r$. Then,
Thus, for every λ that is an eigenvalue of $\tilde{J}$, we have:

$$AQr = \begin{bmatrix} \alpha_1 q_1 r_1 \\ \vdots \\ \alpha_N q_N r_N \end{bmatrix}, \quad AQXr = \begin{bmatrix} \alpha_1 q_1 x_1^* 1_N \\ \vdots \\ \alpha_N q_N x_N^* 1_N \end{bmatrix} \quad r = \begin{bmatrix} \alpha_1 q_1 x_1^* \\ \vdots \\ \alpha_N q_N x_N^* \end{bmatrix} \bar{r}.$$

$$\alpha_i q_i r_i - \alpha_i q_i x_i^* \bar{r} + (1 - \alpha_i) \, r_i = \lambda r_i \; \forall_i$$

If we assume that the candidate λ has an eigenvector such that $\bar{r} = 0$, for every $r_i \neq 0$ we have

$$\lambda = \alpha_i q_i + (1 - \alpha_i) =: \tilde{q}_i \tag{8}$$

Thus, we will have a stable system if we choose $\alpha_i \in (0, 1]$ such that

$$\alpha_i < 2/(1 - q_i) \tag{9}$$

If, on the other hand, the candidate eigenvalue λ has an eigenvector such that $\bar{r} \neq 0$, we have

$$(\alpha_i q_i + (1 - \alpha_i) - \lambda) \, r_i = \alpha_i q_i x_i^* \bar{r}$$

$$\frac{\alpha_i q_i x^*_i}{\alpha_i q_i + (1 - \alpha_i) - \lambda} = \frac{r_i}{\bar{r}}$$

Summing over i, we have

$$\sum_{i=1}^{N} x_i^* \frac{\alpha_i q_i}{\alpha_i q_i + (1 - \alpha_i) - \lambda} = 1$$

$$\sum_{i=1}^{N} x_i^* \frac{\alpha_i q_i}{\tilde{q}_i - \lambda} = 1$$

$$\sum_{i=1}^{\mathrm{N}} x_i^* z_i = 1 \tag{10}$$

where $z_i = (\alpha_i q_i)/(\tilde{q}_i - \lambda) =: z_i^R + jz_i^I$ is complex.

If we can choose α_i such that $z_i^R < 1$ for all i, for some candidate eigenvalue λ, then we know that λ cannot be an eigenvalue for $\tilde{J}$, because $\Sigma_{i=1}^N x_i^* z_i^R < \Sigma_{i=1}^N x_i^* = 1$. We now investigate how to choose α_i in a way such that $z_i^R < 1$ for all candidate eigenvalues on or outside the unit circle. If we can do this, we know the resulting system is locally stable as the only valid eigenvalues for $\tilde{J}$ must lie inside the unit circle. Let $\lambda = \sigma + j\omega$. Then,

$$\begin{aligned} z_i &= \frac{\alpha_i q_i}{(\tilde{q}_i - \lambda)} \\ &= \frac{\alpha_i q_i}{(\tilde{q}_i - (\sigma + j\omega))} \\ &= \frac{\alpha_i q_i}{((\tilde{q}_i - \sigma) - j\omega)} \\ &= \frac{\alpha_i q_i((\tilde{q}_i - \sigma) + j\omega)}{((\tilde{q}_i - \sigma)^2 + \omega^2)} \end{aligned}$$

from which we get

$$z_i^R = \frac{\alpha_i q_i(\tilde{q}_i - \sigma)}{((\tilde{q}_i - \sigma)^2 + \omega^2)}$$

We assume that α_i has been chosen to satisfy (Equation 9), implying $-1 < \tilde{q}_i < 1$.

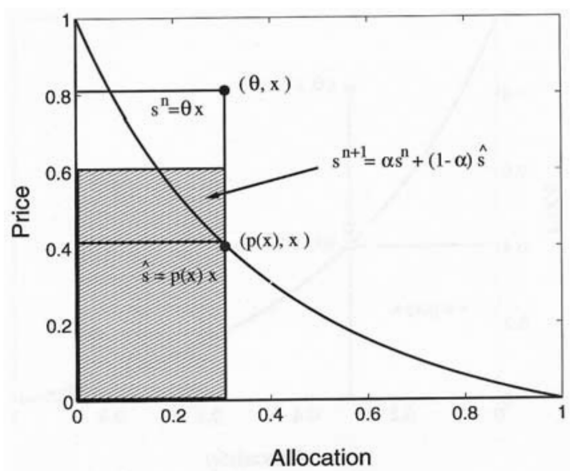

Figure 6. Graphical Interpretation of Relaxed Update Scheme.

Case 1. Let us assume that $q_i \geq 0$. Then, $\sigma \geq q_i \Rightarrow z_i^R \leq 0$ so we assume $\sigma < \tilde{q}_i$. If $|\sigma| \geq 1$, $q_i \geq 0$, then

$$
\begin{aligned}
z_i^R &= \frac{\alpha_i q_i(\tilde{q}_i - \sigma)}{(\tilde{q}_i - \sigma)^2} \\
&= \frac{\alpha_i q_i}{(\tilde{q}_i - \sigma)} \\
&\leq \frac{\alpha_i \tilde{q}_i}{\tilde{q}_i - \sigma)} \\
&\leq 1,
\end{aligned}
$$

where the final inequality is due to the fact that $\sigma < q_i < 1$ and $\sigma\,| \geq 1$ imply $\sigma < 0$. For any nonzero candidate λ, at least one of the first or last inequalities will be strict.

If $|\sigma| < 1$, $|\lambda| \geq 1 \Rightarrow \omega^2 \geq 1 - \sigma^2$. Then,

$$
\begin{aligned}
z_i^R &\leq \frac{\alpha_i q_i(\tilde{q}_i - \sigma)}{(\tilde{q}_i - \sigma)^2 + (1 - \sigma^2)} \\
&= \frac{\alpha_1 q_i(\tilde{q}_i - \sigma)}{\tilde{q}_i^2 - 2\sigma\tilde{q}_i + \sigma^2 + 1 - \sigma^2} \\
&= \frac{\alpha_1 q_i(\tilde{q}_i - \sigma)}{\tilde{q}_i(\tilde{q}_i - \sigma) + (1 - \sigma\tilde{q}_i)} \\
&\leq \frac{\alpha_1 \tilde{q}_i(\tilde{q}_i - \sigma)}{\tilde{q}_i(\tilde{q}_i - \sigma) + (1 - \sigma\tilde{q}_i)} \\
&< 1,
\end{aligned}
$$

where the final inequality is due to the fact that $|\sigma| < 1$ and $|\tilde{q}_i| < 1$ imply $1 - \sigma\,\tilde{q}_i > 0$.

Case 2. Let us now consider $q_i < 0$. If $\sigma \leq \tilde{q}_i$, then $z_i^R \leq 0$, so we assume $\sigma > \tilde{q}_i$. Consider $\sigma \geq 1 > \tilde{q}_i$. Then, we have

$$
\begin{aligned}
z_i^R &\leq \frac{\alpha_i \tilde{q}_i(\tilde{q}_i - \sigma)}{(\tilde{q}_i - \sigma)^2} \\
&= \frac{\alpha_i q_i}{(\tilde{q}_i - \sigma)} \\
&= \frac{\alpha_i q_i}{\alpha_i q_i + 1 - \sigma - \alpha_i} \\
&\leq 1.
\end{aligned}
$$

because $\sigma \geq 1$ implies $1 - \sigma - \alpha_i < 0$. For any nonzero candidate λ, at least one of the first of last inequalities will be strict.

If $\tilde{q}_i \sigma < 1$, $|\lambda| \geq 1 \Rightarrow \omega^2 \geq 1 - \sigma^2$. Then, we have

$$z_i^R \leq \alpha_i q_i \frac{\tilde{q} - \sigma}{(\tilde{q}_i - \sigma)^2 + 1 - \sigma^2)} =: \alpha_i q_i f(\sigma)$$

where

$$\frac{\partial f(\sigma)}{\partial \sigma} = \frac{\tilde{q}_i^2 - 1}{[(\tilde{q}_i - \sigma)^2 + (1 - \sigma)^2]^2} < 0$$

because $|\tilde{q}_i| < 1$. Thus, $f(\sigma)$ is minimized and the bound on z_i^R is maximized as $\sigma \to 1$. This implies

$$\begin{aligned} z_i^R &< \frac{\alpha_i q_i (\tilde{q}_i - 1)}{(\tilde{q}_i - 1)^2 + (1 - 1^2)} \\ &= \frac{\alpha_i q_i}{\tilde{q}_i - 1} \\ &= \frac{\alpha_i q_i}{\alpha_i q_i + 1 - \alpha_i - 1} \\ &= \frac{\alpha_i q_i}{\alpha_i (q_i - 1)} \\ &= \frac{q_i}{q_i - 1} \\ &< 1. \end{aligned}$$

Thus, if α_i is chosen to satisfy (Equation 9), any candidate λ outside the unit circle will not be a viable eigenvalue of $\tilde{J}$. ■

Corollary 1. If $|q_i| < 1, \forall_i$, then the unrelaxed algorithm described in (Equation 6) is locally stable.

Proof. The proof of local stability depends on $|\tilde{q}_i| < 1$. If $\alpha_i = 1$ (the unrelaxed case), $\tilde{q}_i = q_i$ and thus we require $|q_i| < 1$ for local stability. ■

We can interpret q_i as an indicator of the price sensitivity of the i-th agent at equilibrium. Thus, the basic algorithm will converge unless an agent's sensitivity is higher than a particular threshold. The condition that $|q_i| < 1 \ \forall_i$ restricts the set of price functions that can be guaranteed to be locally stable as those agents with valuations that lead to greater price sensitivities risk not converging to an operating point. The relaxation changes the sensitivity parameter q_i to $\tilde{q}_i$, moving it inside the unit circle if q_i is too negative. By reducing the magnitude of the reactions of the agent, relaxation makes the agent less aggressive and effectively dampens the price sensitivity of the agent.

We note that q_i is determined from the allocation at equilibrium, which is not known *a priori* to the agent. Each agent must take into account all possible equilibrium allocations when choosing its relaxation parameter. If the agent minimizes the function

$$q_i(x_i) = (p_i(x_i) + x_i p'_i(x_i)) / p_i(x_i)$$

over the domain $x_i \in (0, d_i(\in))$, let $\hat{q}_i = \arg\min_{x_i \in (0, d_i(\in))} q_i(x_i)$. If all agents choose $\hat{\alpha}_i < 2/(1 - \hat{q}_i)$, then $\hat{\alpha}_i < 2/(1 - \hat{q}_i)\ \forall q_i$ and the update scheme will converge regardless of the equilibrium allocation. In effect, each agent is choosing a relaxation parameter that will be able to dampen the allocation at which the price sensitivity is highest.

We note that if $x_i^* > 0$ and $p'_i(x_i^*) \in (-\infty, 0)$, then $q_i \in (-\infty, 1)$ and we will be able to find an appropriate relaxation parameter. If $x_i^* = 0$, then $q_i = p_i(0)/\theta^*$. In this case we will have $q_i < 1$ unless $p_i(0) = \theta^*$ where $q_i = 1$. In this case, relaxation will not help and it will be possible to have an eigenvalue of one. This situation occurs only when an agent's maximum marginal valuation is exactly the price of the resource. If the valuations of agents are drawn from a set of functions where the maximum marginal distribution is a continuous random variable, then the situation where $p_i(0) = \theta^*$ is a zero probability event. Simulations have shown that if this case occurs, the algorithm converges to a point close to the optimal equilibrium allocation. Though we have only done a local analysis, simulations have shown that the relaxed update scheme converges globally for all cases where it converges locally. Proof of global convergence and and alternate update schemes are areas for further investigation. Simulations of the algorithm can be seen in Figures 7, 8 and 9.

6. Equivalent valuations

We modeled our agents with quasilinear utilities and valuation functions satisfying Assumption 1. As we will see in following sections, there are alternate ways to define agent utilities. To apply the decentralized bidding algorithms, all we require is the existence of price functions (or equivalently, demand functions) that characterize the agents' optimal responses. Given a price function with certain properties, we show that we can produce a quasilinear utility function that yields the same optimal response. In addition to justifying our model of quasilinear utilities, this enables us to gain insight into alternatively defined agent utilities. The quasilinear utility yields an instantaneous valuation even though the original agent utility might have had complementarities over time or cost. With equivalent instantaneous valuations, we have a common base for comparing agent utilities and understanding qualitative behavior in their responses.

Proposition 4. Let $p(x)$ be a positive continuous decreasing function of $x \in (0, 1)$ where $p(1) = 0$ and $p(0) = \lim_{x \searrow 0} p(x)$. If and only if $q(x) := p(x)/(1 - x)$ is differentiable, and $dq/dx \leq 0\ \forall\ x \in (0, 1)$, then the valuation function,

$$v(x) = -\int_x^1 \frac{p(y)}{1-y} dy \tag{11}$$

satisfies Assumption 1, and produces the same optimal response characterized by $p(x)$.

Proof. This is straightforward to see as $v(x)$ is differentiable with $v'(x) = q(x) = p(x)/(1-x)$ which is continuous as $p(x)$ is a continuous function. It is also straightforward to see that the price function generated by that valuation function is the same as the price function used in generating the valuation function, thus the optimal response that each characterizes will be the same. Because $p(x)$ is positive and $(1-x)$ is positive for $x \in (0, 1)$, we have $v'(x) > 0$ for $x \in (0, 1)$. We also have $v''(x) = dq/dx \leq 0$ for $x \in (0, 1)$; thus we have satisfied the conditions of Assumption 1. The "only if" part results from the fact that if either $q(x)$ is not differentiable or $dq/dx > 0$ for some $x \in (0, 1)$, then either $v''(x)$ will not exist or will be positive for some $x \in (0, 1)$, violating the assumed properties of $v(x)$. ■

We defined $p(0)$ as the limit of $p(x)$ as $x \searrow 0$ to allow for the possibility that $p(0)$ might be infinite. From Proposition 4, we note that the main factor in whether a price function has an equivalent utility is whether $p(x)/(1-x)$ is a decreasing function as all the other conditions of Assumption 1 are satisfied through construction or by natural properties of $p(x)$. Looking at this we can find more intuitive necessary and sufficient conditions for equivalence to hold.

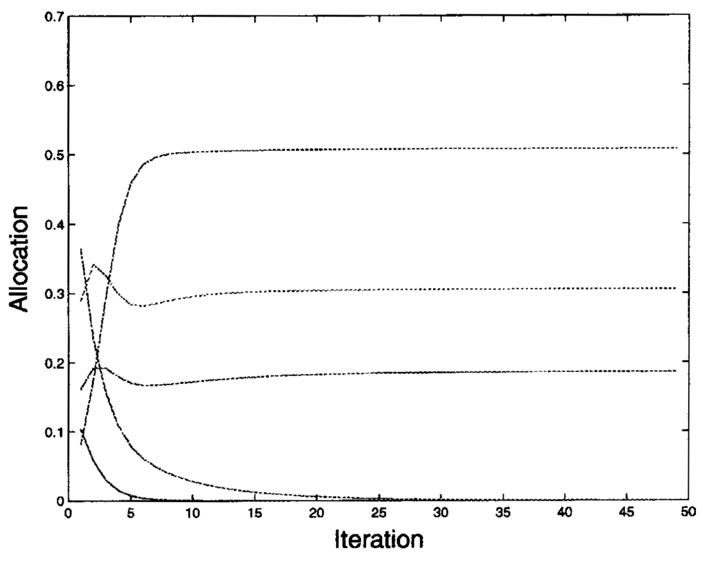

Figure 7. Iterations of 5 Agents Using Relaxed Update.

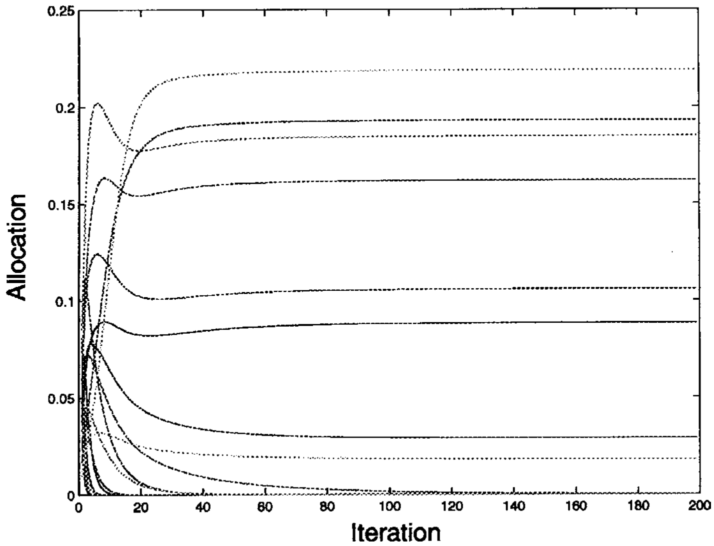

Figure 8. Iterations of 20 Agents Using Relaxed Update.

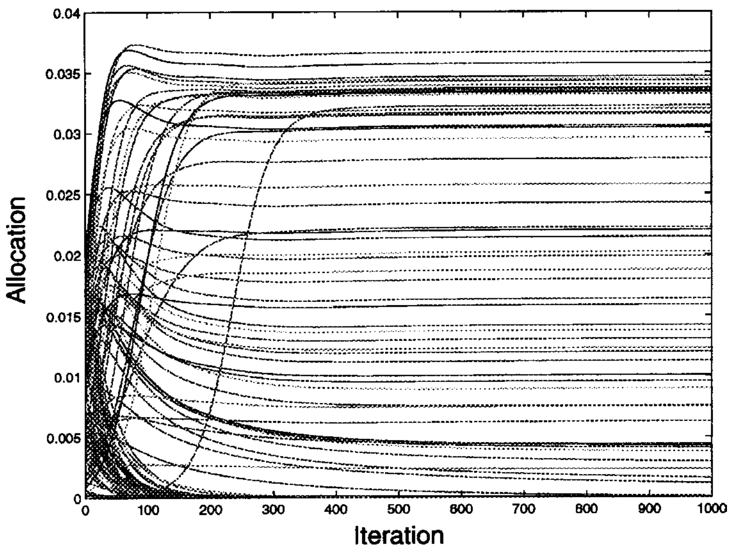

Figure 9. Iterations of 100 Agents Using Relaxed Update.

Corollary 2. For a given price function, $p(x)$, with $p(0) < \infty$, to have an equivalent instantaneous valuation function satisfying Assumption 1, it is necessary that $p(x) \leq p(0)\,(1 - x)$ $\forall\, x \in (0, 1)$.

Proof. Let $f(x) = p(x) - p(0)\,(1 - x)$. Then, we have

$$\frac{d}{dx}\,\frac{p(x)}{1-x} = \frac{d}{dx}\left(p(0) + \frac{f(x)}{1-x}\right) = \frac{f'(x)\,(1-x) + f(x)}{(1-x)^2} \leq 0\ \forall\, x \in (0, 1)$$

as a necessary condition from Proposition 4. This implies that it is necessary that $f'(x)(1 - x) + f(x) \leq 0$. Let us assume that $f(x_0) > 0$ for some $x_0 \in (0, 1)$. Let $x_1 = \max\,\{x : f(x) = 0, x < x_0\}$. We know that the set is non-empty since $f(0) = 0$. By the continuity of $p(x)$, and hence of $f(x)$, we know that $f(x) > 0\ \forall\, x \in (x_1, x_0)$. By the Mean Value Theorem, there exists $x_2 \in (x_1,x_0)$ such that $f'(x_2) = (f(x_0) - f(x_1))\,/\,(x_0 - x_1) > 0$. But since $f(x_2) > 0$, we have $f'(x_2)(1 - x_2) + f(x_2) > 0$, which violates the necessary condition. Thus, we must have $f(x) \leq 0 \Rightarrow p(x) \leq p(0)(1 - x)\ \forall\, x \in (0, 1)$. ■

Intuitively, this means that a bounded price function must lie below the line segment connecting $p(0)$ and $p(1)$ for it to have an equivalent instantaneous valuation. It can be further shown that any viable price curve cannot go above and come back below any line with negative slope that intersects $(x,p(x)) = (1, 0)$. This can be shown by using a modified version of the previous proof. If the price function is strictly convex, we have the following sufficiency condition.

Corollary 3. If a given price function, $p(x)$, is strictly convex, having $p''(x) > 0\ \forall\, x \in (0, 1)$, and $p(1) = 0$, then the price function has an equivalent instantaneous valuation.

Proof. To have an equivalent instantaneous valuation, from Proposition 4 it is sufficient if

$$\frac{d}{dx}\,\frac{p(x)}{1-x} = \frac{p'(x)\,(1-x) + p(x)}{(1-x)^2} \leq 0 \tag{12}$$

From Taylor's theorem, we have $p(1) = p(x) + p'(x)(1 - x) + p''(y)(1 - x)^2/2$ for some $y \in (x, 1)$. Rearranging the terms, we have $p'(x)(1 - x) + p(x) = p(1) - p''(y)(1 - x)^2/2$. Because $p(1) = 0$ and $p''(y) > 0$, the expression is negative and Eq. 12 is satisfied. Thus, $p(x)$ has an equivalent instantaneous valuation. ■

The previous two corollaries give us easy visual clues as to whether we can extract a concave valuation function from a price function. If the price curve is not sublinear, we know we cannot. If the price curve is convex, we know we can. We apply the previous results to specific agent tasks with various utilities in the following sections.

Jobs in series

We now consider a scenario where agents are generated at some subset of nodes in a network with a sequence of jobs to complete. The jobs require access to resources available at various nodes throughout the network. Based on some budget constraints, each agent attempts to purchase resources throughout the network to complete its set of jobs according to a given performance measure or utility function. For now, we will assume that the agents have perfect knowledge about the states of demand (or equivalently, the prices) of various resources in the network. In a network of resources being managed by electronic markets, it is reasonable to assume that there will be a mechanism to provide price information to agents. From this, an agent will choose a sequence of resources that it will attempt to purchase service from to complete its tasks. Let us assume that the *i*-th agent has a sequence of jobs with K_i tasks, where q_i^k is the size of the job of the *k*-th task. Let C_i^k be the capacity of the resource providing the service needed by the *k*-th task of the *i*-th agent. We assume that every resource allocates its services using the proportionally fair auction characterized by Eqs 1 and 2. In this context, the bid will constitute a payment that the agent is willing to make per unit of time that it uses the resource. Let s_i^k be the bid of the *i*-th agent for resource chosen for the *k*-th task on its itinerary and s_{-i}^k be the sum of the bids of other agents competing for that resource (which includes the bid $\in^k$ made by the *k*-th resource). Then, the rate of service obtained by the *i*-th agent for its *k*-th task is

$$x_i^k = C_i^k \left(\frac{s_i^k}{s_i^k + s_{-i}^k} \right) .$$

Then, the time taken to complete that job will be

$$t_i^k = \frac{q_i^k (s_i^k + s_{-i}^k)}{C_i^k s_i^k} .$$

The expense to the agent is the bid times the duration of service, which yields

$$e_i^k = s_i^k t_i^k = \frac{q_i^k (s_i^k + s_{-i}^k)}{C_i^k} .$$

The decision that faces the agent is how to balance its performance as measured by the time taken to complete its jobs and the cost of obtaining service. We first consider the following criterion:

$$\min \left(\sum_{k=1}^{K_i} e_i^k + \alpha_i^k \sum_{k=1}^{K_i} t_i^k \right)$$

where α_i^k represents the relative value to the i-th agent of the time taken to complete the k-th job relative to its cost. The agent is minimizing a weighted combination of the total cost and the total time taken to complete all its jobs. Substituting for t_i^k and e_i^k, and taking the derivative with respect to s_i^k, we have

$$\frac{q_i^k}{C_i^k} - \frac{\alpha_i^k q_i^k s_{-i}^k}{C_i^k (s_i^k)^2} = 0$$
$$(s_i^k)^2 = \alpha_i^k s_{-i}^k.$$

It can be seen that the minimization is a convex function of s_i^k and the preceding equation represents the optimal response in terms of the other agents' bid total. Let p_i^k be the price of the resource chosen by the i-th agent to complete its k-th task. The price is equivalent to the sum of all the bids made at that resource, so $p_i^k = s_i^k + s_{-i}^k$. Substituting this, we have

$$(s_i^k)^2 = \alpha_i^k (p_i^k - s_i^k)$$
$$(s_i^k)^2 + \alpha_i^k s_i^k - \alpha_i^k p_i^k = 0$$
$$s_i^k = \frac{-\alpha_i^k + \sqrt{(\alpha_i^k)^2 + 4\alpha_i^k p_i^k}}{2}$$

which is the optimal response in terms of the price of the resource. We choose the greater root of the quadratic as it is the only positive solution which is required of all bids. Dividing the optimal bid by p_i^k gives us the demand function associated with this optimal response as $d_i^k(p_i^k) = s_i^k(p_i^k) / p_i^k$. By substituting $s_i^k = p_i^k x_i^k$ into the quadratic above, we get

$$(p_i^k)^2 (x_i^k)^2 + \alpha_i^k p_i^k x_i^k - \alpha_i^k p_i^k = 0$$

which we can solve to obtain the following price function:

$$p_i^k = \frac{a_i^k (1 - x_i^k)}{(x_i^k)^2}.$$

We note that

$$\frac{d}{dx_i^k} \frac{p_i^k}{1 - x_i^k} = \frac{d}{dx_i^k} \frac{\alpha_i^k}{(x_i^k)^2} = \frac{-2\alpha_i^k}{(x_i^k)^3} < 0$$

so we can apply the results of Proposition 4 to find the equivalent valuation function:

$$v_i^k (x_i^k) = \frac{-\alpha_i^k}{x_i^k}.$$

We have thus shown that agents are able to extract a price function and a valuation function even though their objectives were not maximizations of quasilinear utilities. Here, the minimization problem can be decoupled with respect to each job. Then for each job, the problem can be interpreted as a minimization problem for a job broken into several pieces (of arbitrarily small size) to be completed in series with the same objective. In the limit, the problem tends to an instantaneous optimization and thus, an instantaneous valuation of service.

8. Finite budget

We now consider the situation where an agent is given an endowment, E_i, which it may not exceed as it attempts to minimize the total time taken to complete its jobs. There is no benefit for returning any of the endowment, so E_i can also be interpreted as a hard cap on spending. This can be expressed as the following optimization problem:

$$\min \sum_{k=1}^{K_i} t_i^k \quad \text{such that} \quad \sum_{k=1}^{K_i} e_i^k \leq E_i.$$

We solve this problem using Lagrangian methods. We first introduce the Lagrangian

$$\mathcal{L} = \sum_{k=1}^{K_i} t_i^k + \lambda \left(\sum_{k=1}^{K_i} e_i^k - E_i \right).$$

Substituting for t_i^k and e_i^k and taking partial derivatives with respect to s_i^k, we have

$$\frac{\partial \mathcal{L}}{\partial s_i^k} = \frac{-q_i^k s_{-i}^k}{C_i^k (s_i^k)^2} + \lambda \frac{q_i^k}{C_i^k} = 0 \qquad \Rightarrow \qquad \lambda = \frac{s_{-i}^k}{(s_i^k)^2}.$$

We know that $s_{-i}^k \geq \epsilon_i^k > 0$ where ϵ_i^k is the bid made by the resource providing the service needed by the k-th task on the i-th agent's itinerary, which implies $\lambda > 0$. Because λ is identical for all tasks, we have the following relationships between optimal bids:

$$s_i^k = s_i^k \sqrt{\frac{s_{-i}^k}{s_{-i}^j}} \tag{13}$$

The optimal assignments of bids is then proportional to the square root of the demand by other agents. Incorporating the constraint, we get

$$\lambda \frac{\partial \mathcal{L}}{\partial y} = \lambda \left(\sum_{k=1}^{K_i} \frac{q_i^k (s_i^k + s_{-i}^k)}{C_i^k} - E_i \right) = 0.$$

Since $\lambda > 0$, we know that the second factor must be zero. Substituting for $\{s_i^k\}_{k=2}^{K_i}$ in terms of s_i^1 using Eq. 13, we have

$$\frac{q_i^1}{C^1}(s_i^1 + s_{-i}^1) + \sum_{k\neq 1} \frac{q_i^k}{C_i^k}\sqrt{\frac{s_{-i}^k}{s_{-i}^1}}\, s_i^1 + \sum_{k\neq 1} \frac{q_i^k}{C_i^k}\, s_{-i}^k - E_i = 0.$$

Solving this for s_i^1, we get

$$s_i^1 = \frac{E_i - \sum_{k\neq 1} \frac{q_i^k}{c_i^k} s_{-i}^k \; \frac{q_i^1}{C^1} s_{-i}^1}{\frac{q_i^1}{C^1} + \sum_{k\neq 1} \frac{q_i^k}{C_i^k}\sqrt{\frac{s_{-i}^k}{s_{-i}^1}}}$$

which is the optimal bid for the first or current job of the *i*-th agent in terms of the demand of the resources for the jobs in its itinerary. For an agent to implement this strategy, it must have estimates of the demand at resources that it plans to visit in the future. This fits with the notion of a finite budget as one has to have an idea of how much money one will need in the future to know how much one can reasonably spend now. The reaction function is parameterized by the agent's beliefs about the future regardless of their accuracy. In this analysis, we will not address the effects of accuracy. Instead we focus on the competition among agents, and thus we can assume that the estimates are accurate. We can rewrite the optimal response as follows

$$s_i^1 = f_i(s_{-i}^1) := \frac{\alpha^i - \beta^i s_{-i}^1}{\beta^i + \frac{\gamma^i}{\sqrt{s_{-i}^1}}},$$

where

$$\alpha_i := E_i - \sum_{k\neq 1} \frac{q_i^k}{C_i^k} s_{-i}^k,$$

$$\beta_i := \frac{q_i^1}{C^1},$$

$$\gamma_i := \sum_{k\neq 1} \frac{q^k{}_i}{C_k{}^i} \sqrt{s^k{}_{-i}}.$$

Intuitively, α_i represents the estimate of the maximum money available for the current job. If that amount is less than zero, the agent cannot afford to purchase service under the current state of the network. This is reflected in the optimal response, as a negative α_i would yield a negative s_i^k since β_i and s_{-i}^1 are positive and agents are required to submit non-negative bids. In fact, we see that the optimal response will yield a negative bid whenever $s_{-i}^1 > \alpha_i / \beta_i$. There is intuition behind this as well. Because β_i represents the minimum time required to complete the current job, and α_i is the maximum money available for the current job, α_i / β_i is the largest amount of money per unit time that the i-th agent could spend or bid for this resource. If the other agents' total bids, s_{-i}^1, create a price that is greater than the i-th agent's spending limit, it will choose not to participate. The third parameter, γ_i, is a factor that when divided by $\sqrt{s_{-i}^1}$ gives an estimate of the excess time (time above the minimum time to complete a job) necessary to complete the remaining tasks. Thus, the optimal response is the ratio of the excess money for all the jobs to the estimated excess time for all the jobs. We note that when an agent is completing its last task, $\gamma_i = 0$. This yields an optimal response of $s_i^k = \alpha_i / \beta_i - s_{-i}^k$ This again matches intuition as an agent unconcerned with future tasks will want to bid at the highest rate possible (to minimize time) while spending every bit of money available to it. The budget constraint can be rewritten as $\beta_i s_i^k + \beta_i s_{-i}^k \leq \alpha_i$. When the i-th agent maximizes its bid under that constraint, we get the same equation as the optimal response. If the price is larger than α_i / β_i, the agent will not be able to participate and if the price is less than that quantity, the agent will bid until the price reaches that limit. We drop the superscript, with the knowledge that the bids are in reference to the current resource. Making the substitution, $s_{-i} = p_i - s_i$, yields

$$s_i = \frac{\alpha_i - \beta_i \ (p_i - s_i)}{\dfrac{\beta_i + \gamma_i}{p_i - \sqrt{s_i}}} ,$$

from which we obtain the following quadratic in s_i,

$$(\gamma_i)^2 \ (s_i)^2 + (\alpha_i - \beta_i \, p_i)^2 \ s_i - (\alpha_i - \beta_i \, p_i)^2 \ p_i = 0.$$

We can rewrite the quadratic as follows

$$(\gamma_i)^2 \left(\frac{s_i^{\,2}}{p_i}\right)(p_i)^2 + \alpha_i - \beta_i p_i)^2 \left(\frac{s_i}{p_i}\right) \ p_i - (\alpha_i - \beta_i p_i)^2 \ p_i = 0,$$

and by dividing by p_i which we know to be positive, and making the substitution, $x_i = s_i / p_i$, we have

$$(\gamma_i)^2 \ (x_i)^2 \ p_i + (\alpha_i - \beta_i \, p_i)^2 \ (x_i - 1) = 0,$$

which we can solve for x_i to obtain the demand function:

$$x_i = \frac{-(\alpha_i - \beta_i p_i)^2 + \sqrt{(\alpha_i - \beta_i p_i)^4 + 4 p_i \gamma_i^2 \ (\alpha_i - \beta_i p_i)^2}}{2 \gamma_i^2 p_i}$$

We choose the greater root because we need $x_i \geq 0$. Rearranging the quadratic with respect to p_i, we have

$$\beta_i^2 p_i^2 - \left(2\alpha_i\beta_i + \frac{\gamma_i^2 x_i^2}{1 - x_i}\right) p_i + \alpha_i^2 = 0$$

which we solve to obtain the following price function.

$$p_i = \frac{\alpha_i}{\beta_i} + \frac{1}{2\beta_i^2 (1 - x_i)} \left[\gamma_i^2 x_i^2 - \sqrt{\gamma_i^4 x_i^4 + 4\alpha_i \beta_i \gamma_i^2 x_i^2 (1 - x_i)}\right]$$

We choose the lesser root because the price function must reflect the fact that the agent cannot participate if the price is greater than α_i / β_i.

An economy of agents with budget limits has been simulated in Bredin et al. (2000). In the following figures, we can see how the price functions change as we vary parameters of the agent with jobs in series with a finite budget. We simplify the scenario and consider an agent with two jobs. Let E_i be the endowment, q_i^1 be the size of the first job, q_i^2 be the size of the second job, and s_{-i}^2 be the demand of other agents for the second job. For Figure 10, $q_i^1 = 0.1302$, $q_i^2 = 0.6638$, $s_{-i}^2 = 0.2544$. For Figure 11, $E_i = 0.6375$, $q_i^2 = 0.7927$, $s_{-i}^2 = 0.4787$. For Figure 12, $E_i = 0.8336$, $q_i^1 = 0.0579$, $s_{-i}^2 = 0.3529$. For Figure 13, $E_i = 1.1791$, $q_i^1 = 0.3764$, $q_i^2 = 0.5936$.

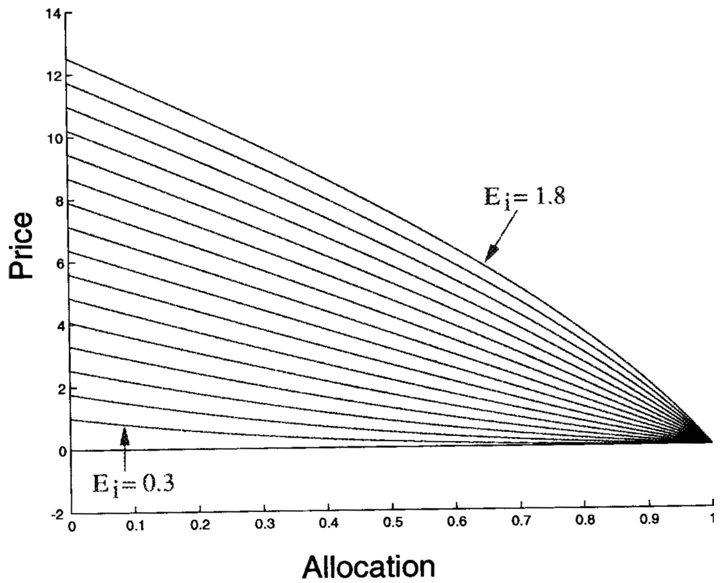

Figure 10. Price Functions for $E_i \in \{0.3, 0.4, \ldots, 1.8\}$.

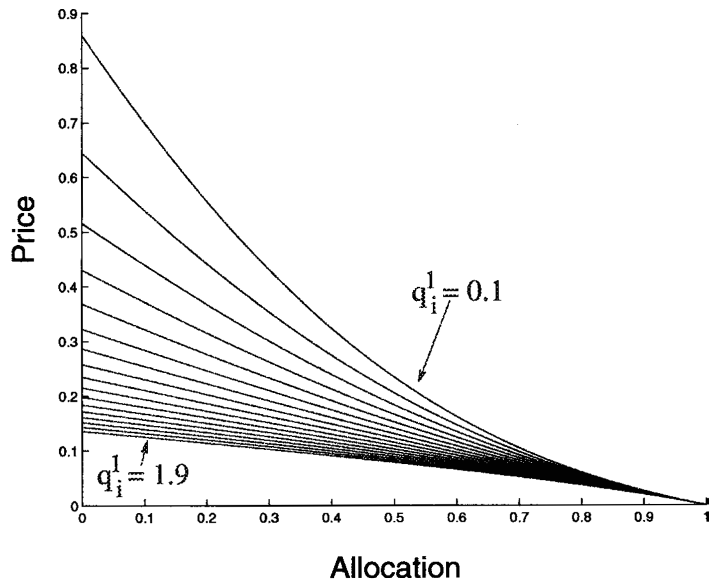

Figure 11. Price Functions for $q_i^1 \in \{0.1, 0.2, \ldots, 1.9\}$.

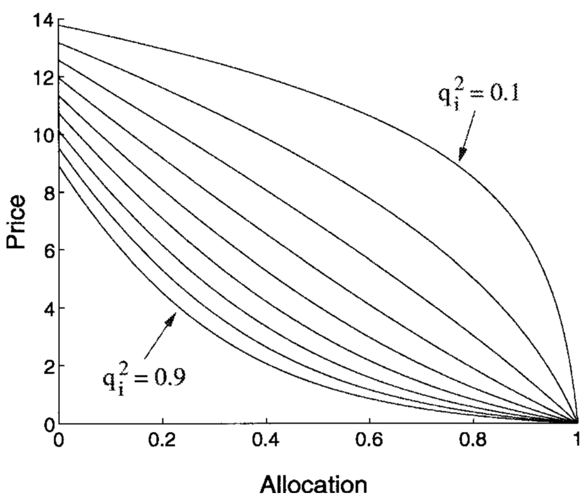

Figure 12. Price Functions for $q_i^2 \in \{0.1, 0.2, \ldots, 0.9\}$.

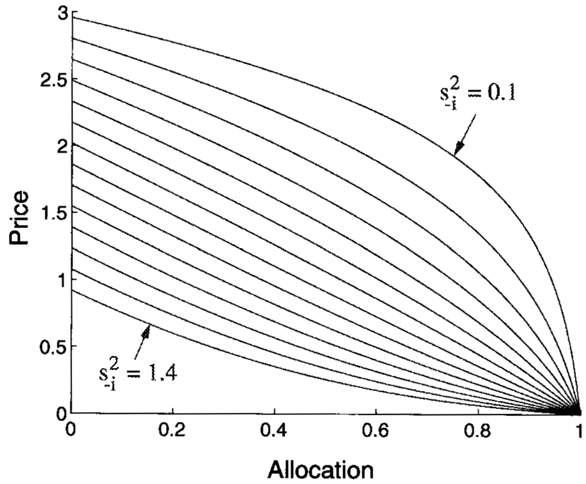

Figure 13. Price Functions for $s_{-i}^2 \in \{0.1, 0.2, \ldots, 1.4\}$.

We can see that as these parameters vary, the price curves uniformly increase or decrease and also change from convex (which by Corollary 3, we know has an equivalent instantaneous valuation) to concave (which by Corollary 2, we know does not have an equivalent instantaneous valuation that satisfies Assumption 1). To gain an intuitive understanding of these figures, we must understand what it means when a price curve does not have a valuation that satisfies Assumption 1.

Modifying, the second-order condition presented in Eq. 5, we see that if

$$v''(x)(1-x) < 2v'(x),$$

then x is a unique maximizing allocation if it meets the first-order necessary condition $p(x) = v'(x)(1-x)$. We note that for all decreasing price functions, we have

$$p'(x) = v''(x)(1-x) - v'(x) < 0,$$

which implies that the modified second-order condition is met. Furthermore, given a decreasing price function, if we substitute the valuation function generated by Eq. 11 into the modified second-order condition, we have

$$\frac{p'(x)\,(1-x) + p(x)}{(1-x)^2}\,(1-x) < \frac{2p(x)}{1-x}$$
$$\Rightarrow p'(x) < \frac{p(x)}{1-x}.$$

Since the LHS is negative and the RHS is positive, we know this is satisfied. The preceding thus shows that a valuation function does not necessarily have to be concave for a unique maximizing response to exist, though the convexity has limits. If a valuation is strictly convex, the effect is to push the agent into higher equilibrium allocations and higher equilibrium costs. This is counter to the concept of diminishing returns but the complementarities induced by the coupling of the jobs with the same finite budget forces the agent to deviate from a concave valuation. This gives us some intuition into the changes in the price functions as parameters vary.

As E_i increases, the agent has more money to spend and therefore can accept a higher price for each allocation. Thus, the price curves are uniformly higher as the endowment increases. Also, as an agent has more money to spend it is encouraged to purchase higher allocations, which would explain why the curves become more concave (implying a convex valuation) as the endowment increases. As the current job size, q_i^1, increases and the budget remains the same, the agent cannot afford to spend as much on the current job; thus the price curves become uniformly lower. However, as the current job size increases with respect to the future jobs, its effect on overall performance increases as well, and even though the agent cannot spend as much money, it is encouraged to seek a higher allocation, and thus the price curve becomes more concave as q_i^1 increases. Increasing future job sizes q_i^k and future demands s_{-i}^k for $k > 1$, both have the effect of increasing the importance of the future jobs and minimizing the importance of the first job. This is why we see that the price functions get uniformly lower and progressively convex as we want the agent to spend less money and settle for lower allocations as it is necessary to save more of the finite endowment for the future.

This scenario shows the robustness of looking at an optimal response in the form of a price function. It is important that the price functions are associated with valuations so that we can find a direct relation between motivation and action. By limiting ourselves to strictly concave valuations (and strictly convex valuations that satisfy the modified second-order condition), we assure ourselves of unique responses. Investigating more complex forms of valuations is an open area for further research.

9. Finite time

We consider here another agent task where a sequence of jobs needs to be completed in a specified amount of time, T_i, while minimizing the cost accrued. This can be expressed as the following optimization problem:

$$\min \sum_{k=1}^{K} e_i^k \text{ such that } \sum_{k=1}^{K} t_i^k \leq T_i.$$

We introduce the following Lagrangian:

$$\mathcal{L} = \sum_{k=1}^{K} e_i^k + \lambda \left(\sum_{k=1}^{K} t_i^k - T_i\right),$$

substitute for t_i^k and e_i^k, and take partial derivatives with respect to s_i^k to get:

$$\frac{\partial \mathcal{L}}{\partial s^k_i} \;\frac{q^k_i}{\overline{C}^k_i} + \lambda \left(\frac{-q^k_i \, s^k_{-i}}{C^k_i \, (s^k_i)^2} \right) = 0 \quad \Rightarrow \quad \lambda = \frac{(s^k_i)^2}{s^k_{-i}}$$

Because s^k_{-i} includes the resource's bid, and the agent's bid must be positive, we know that $\lambda > 0$. Also, because λ is identical for all the resources, we have the following relationship between the bids:

$$s^k_i = s^j_i \sqrt{\frac{s^k_{-i}}{s^j_{-i}}} \quad . \tag{14}$$

We also have the following equation incorporating the constraint:

$$\lambda \frac{\partial \mathcal{L}}{\partial \lambda} = \lambda \sum_{k=1}^{K} \frac{q^k_i}{C^k_i} + \frac{q^k_i \, s^k_{-i}}{C^k_i \, s^k_i} - T_i = 0.$$

Since $\lambda > 0$, the latter factor must be identically zero. Substituting Eq. 4}), we have:

$$\frac{q^1_i \, s^1_{-i}}{C^1 \, s^1_i} + \sum_{k \neq 1} \frac{q^k_i \sqrt{s^k_{-i}}}{C^k_i} \frac{\sqrt{s^1_{-i}}}{s^1_i} = T - \sum_{k=1}^{K} \frac{q^k_i}{C^k_i}. \tag{15}$$

Introducing the following variables,

$$\alpha_i := T_i - \sum_{k \neq 1}^{K} \frac{q^k_i}{C^k_i}$$

$$\beta_i := \frac{q^1_i}{C^1}$$

$$\gamma_i := \sum_{k=1} \frac{q^k_i}{C^k_i} \sqrt{s^k_{-i}}$$

we can rewrite Eq. 15 as:

$$\beta_i s^1_{-i} + \gamma_i \sqrt{} \; s^1_{-i} = \alpha_i s^1_i.$$

Dropping the superscript, and substituting $s_{-i} = p_i - s_i$, where p_i is the price of the current resource in the itinerary, we have:

$$\alpha_i \, s_i = \beta_i \, (p_i - s_i) + \gamma_i \sqrt{p_i - s_i}$$
$$(\alpha_i + \beta_i) \, s_i - \beta_i \, p_i = \gamma_i \sqrt{p_i - s_i}.$$

We note that since the RHS of the previous equation is always positive, we require $s_i > p_i$ $\beta / (\alpha + \beta)$, for a solution to exit. Squaring both sides, we have

$$(\alpha_i + \beta_i)^2 s_i^2 - 2\beta_i (\alpha_i + \beta_i) p_i s_i + \beta_i^2 p_i^2 = \gamma_i^2 (p_i - s_i)$$

$$(\alpha_i + \beta_i)^2 s_i^2 + (\gamma_i^2 - 2\beta_i (\alpha_i + \beta_i) p_i) s_i + \beta_i^2 p_i^2 - \gamma_i^2 p_i = 0.$$

Making the substitution $s_i = p_i x_i$, and then dividing by p_i which we know to be positive, we have the following equation which characterizes the optimal response:

$$(\alpha_i + \beta_i)^2 p_i x_i^2 + (\gamma_i^2 - 2\, \beta_i (\alpha_i + \alpha_i) p_i) x_i + (\beta_i^2 p_i - \gamma_i^2) = 0$$

We can solve this for x_i to obtain the following demand function

$$x_i = \frac{\beta i}{\alpha_i + \beta_i} + \frac{-\gamma_i^2 + \sqrt{\gamma_i^4 + 4\gamma_i^2 (\alpha_i + \beta_i) \alpha_i p_i}}{2(\alpha_i + \beta_i)^2 p_i} ,$$

and solve for p_i to obtain the following price function

$$p_i = \frac{\gamma^2{}_i (1 - x_i)}{[(\alpha_i + \beta_i) x_i - \beta)]2} = 0.$$

We choose the greater root for x_i since $s_i > p_i\, \beta / (\alpha + \beta) \Rightarrow x_i > \beta / (\alpha + \beta)$. Similarly, we realize that even though the price function is defined $\forall\, x \in (0, 1)$, it is only valid for $x_i > \beta / (\alpha + \beta)$. We note that $p_i(x_i)/(1 - x_i)$ is decreasing on $(\beta / (\alpha + \beta), 1)$. Thus, defining the valuation as in Eq. 11, we have

$$v_i(x_i) = \left(\frac{\gamma_i^2}{\alpha_i + \beta_i}\right) \left(\frac{1}{\alpha_i} - \frac{1}{(\alpha_i + \beta_i) x_i - \beta_i}\right) \quad \forall\, x \in \left(\frac{\beta_i}{\alpha_i + \beta_i}\right), 1.$$

Sample plots of valuation, demand and price functions can be seen in Figure 14. We see that the equivalent valuation is a concave increasing function on a subinterval of the allocation space and meets the conditions of Assumption 1 on this subinterval. If we set the valuation to be $-\propto$ on $(0, \beta / (\alpha + \beta))$, the resulting optimal response would match the optimal response of the optimization problem stated in the beginning of this section. We see that the demand function does not go to zero as the price increases and the price function increases to infinity above an allocation of zero. The reason for this is that the agent is effectively inelastic with respect to allocation close to the minimal allocation requirement, and even exorbitant prices will not deter the agent. This is due to the lack of a constraint on the expenditure accrued. Nevertheless, this case of an inelastic agent can also be modeled with a demand function, price function and an equivalent instantaneous valuation.

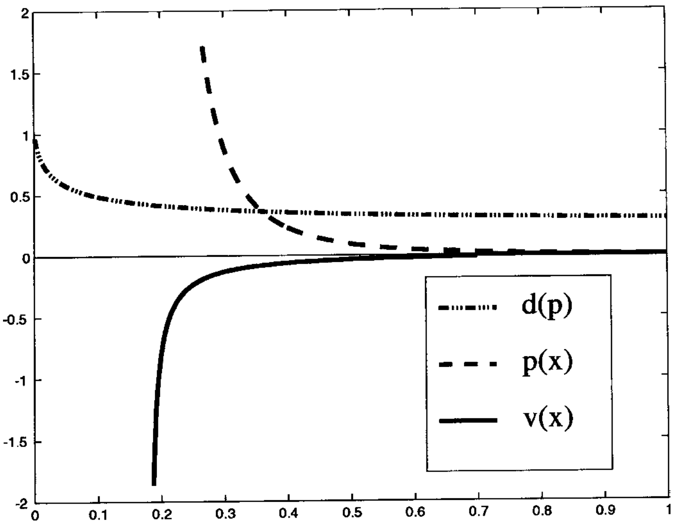

Figure 14. Demand, Price and Valuation Functions for Agent with Finite Time Constraint for $\alpha_i = 0.9641$, $\beta_i = 0.2077$, $\gamma_i = 0.1611$.

10. Conclusion

We have analyzed a proportionally fair divisible auction that is verifiable and has low signaling and computational loads. Redefining optimal responses as price functions allows us to show that the mechanism has a unique Nash equilibrium. We develop decentralized algorithms that converge to the equilibrium without needing resource feedback or sharing private information. We demonstrate the robustness of our characterization by investigating several utility models of an agent given a sequence of tasks.

In agent economies for computational and network resources, divisible auctions are appropriate mechanisms for allocation. There are many areas open for further investigation. We have studied the effects of coalition formation and extended the auction to a multiple resource setting. The seller revenue problem and generalization of divisible auction mechanisms are some of the issues to be addressed in the future.

References

Başar, T. and G. J. Olsder. 1999. "Dynamic Noncooperative Game Theory," *in Classics in Applied Mathematics*, 2nd Ed., Philadelphia: SIAM.

Bertsekas, D. and R. Gallagher. 1991. *Data Networks*, 2nd Ed., Englewood Cliffs, N.J.: Prentice-Hall.

Bredin, J., R.T. Maheswaran, O. O. Imer, T. Başar, D. Kotz, and D. Rus. 2000. "A Game-Theoretic Formulation of Multi-Agent Resource Allocation," *in Proceedings of the 4th International Conference on Autonomous Agents*. Barcelona, Spain, 349–356.

Chavez, A. and P. Maes. 1996. "Kasbah: An Agent Marketplace for Buying and Selling Goods," *in Proceedings*

of the 1st International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology. London, 75–90.
Cheng, J. Q. and M. P. Wellman. 1998. "The WALRAS Algorithm: A Convergent Distributed Implementation of General Equilibrium Outcomes," *Journal of Computational Economics* 12, 1–23.
Clearwater, S. H. (ed.), *Market-Based Control: A Paradigm for Distributed Resource Allocation*. Singapore: World Scientific.
De Vries, S. and R. Vohra. (to appear). "Combinatorial Auctions: A Survey," *INFORMS Journal of Computing*.
Gagliano, R. A., M. D. Fraser, and M. E. Schaefer. 1995. "Auction Allocation for Computing Resources," *Communications of the ACM* 38 (6), 88–102.
Gibbens, R. J. and F. P. Kelly. 1999. "Resource Pricing and the Evolution of Congestion Control," *Automatica* 35 (12), 1969–1985.
Kelly, F. 1997. "Charging and Rate Control for Elastic Traffic," *European Transactions on Telecommunications* 8 (1), 33–37.
Kelly, F., A. Maulloo, and D. Tan. 1998. "Rate Control for Communication Networks: Shadow Prices, Proportional Fairness and Stability," *Journal of the Operations Research Society* 49 (3) 237–252.
Kephart, J. O., J. E. Hanson, and A. R. Greenwald. 2000. "Dynamic Pricing by Software Agents," *Computer Networks* 32 (6), 731–752.
La, R. J. and V. Anantharam. 2000. "Charge-Sensitive TCP and Rate Control in the Internet," *in Proceedings of the IEEE INFOCOM*, 1166–1175.
MacKie-Mason, J. K. and H. R. Varian. 1993. "Pricing the Internet," *in Public Access to the Internet*. JFK School of Government.
Mackie-Mason, J. K. and H. R. Varian. 1995. "Pricing Congestible Network Resources," *IEEE Journal on Selected Areas in Communications* 13 (7), 1141–1148.
Maheshwari, U. 1995. "Charge-Based Proportional Scheduling," Technical Memorandum MIT/LCS/TM-529, MIT Laboratory for Computer Science.
Massoulie, L. and J. Roberts. 1999. "Bandwidth Sharing: Objectives and Algorithms," *in Proceedings of IEEE INFOCOM*, vol. 3. New York, 1395–1403.
Owen, G. 1995. *Game Theory*, 3rd Ed., New York: Academic Press.
Petrosjan, L. A. and N. A. Zenkevich. 1996. *Game Theory*. Singapore: World Scientific.
Regev, O. and N. Nisan. 1998. "The POPCORN Market: An Online Market for Computational Resources," *in Proceedings of the 1st International Conference on Information and Computational Economies*. Charleston, SC, 148–157.
Sandholm, T. 1999. "An Algorithm for Optimal Winner Determination in Combinatorial Auctions," *in Proceedings of the 16th International Joint Conference on Artificial Intelligence*. Stockholm, 542–547.
Sandholm, T. W. 1996. "Limitations of the Vickrey Auction in Computational Multiagent Systems," *in Proceedings of the 2nd International Conference on Multi-Agent Systems*. Kyoto, Japan, 299–306.
Stoica, I., H. Abdel-Wahab, K. Jeffay, S. K. Baruah, J. E. Gehrke, and C. G. Plaxton. 1996. *in Proceedings of the 17th IEEE Real-Time Systems Symposium*. Washington, D.C., 288–299.
Waldspurger, C. and W. E. Weihl. 1995. "Stride Scheduling: Deterministic Proportional-Share Resource Management," Technical Memorandum MIT/LCS/TM-528, MIT Laboratory for Computer Science.
Waldspurger, C. A., T. Hogg, B. A. Huberman, J. O. Kephart, and W. S. Stornetta. 1992. "Spawn: A Distributed Computational Economy," *IEEE Transactions on Software Engineering* 18 (2), 103–117.
Wellman, M., W. Walsh, P. Wurman, and J. Mackie-Mason. 2001. "Auction Protocols for Decentralized Scheduling," *Games and Economic Behavior* 35, 271–303.