ARE UTILITY, PRICE, AND SATISFACTION RESOURCE ALLOCATION MODELS SUITABLE FOR LARGE-SCALE DISTRIBUTED SYSTEMS?

XIN BAI, LADISLAU BÖLÖNI, AND DAN C. MARINESCU

School of Electrical Engineering and Computer Science
University of Central Florida
Orlando, FL 32816-2362, USA
Email: (xbai, lboloni, dcm)@cs.ucf.edu

HOWARD JAY SIEGEL

Department of Electrical and Computer Engineering
and Department of Computer Science
Colorado State University
Fort Collins, CO 80523-1373, USA
Email: HJ@ColoState.edu

ROSE A. DALEY AND I-JENG WANG

Applied Physics Laboratory
Johns Hopkins University
11100 Johns Hopkins Road Laurel, MD 20723-6099, USA
Email: (Rose.Daley, I-Jeng.Wang)@jhuapl.edu

Computational, data, and service grids, peer to peer systems, and wireless communication systems are examples of open systems where the distinction between providers and consumers of resources is blurred. Individual members of the community contribute computing cycles, storage, services, and communication bandwidth to the pool of resources available to the entire community. While the popularity of such systems increases, their resource management models seldom take into account the utility for the consumers of the resources, and the incentives to provide resources. In this paper, we discuss a resource allocation model that takes into account the utility of the resources for the consumers and the pricing structure imposed by the providers. We show how a satisfaction function can express the preferences of the consumer both regarding the utility and the price of the resources. In our model, the brokers are mediating among the selfish interests of the consumers and the providers, and societal interests, such as efficient resource utilization in the system. We report on a simulation experiment to study the behavior of the system in steady state and in transient state.

1. Introduction and Motivation

In many social and man-made systems, scarce resources have to be shared among a large population of consumers. To study possible resource management policies, we have to develop resource consumption models that take into account different, possibly contradictory, views of the benefits associated with resource consumption as well as the rewards for providing resources to the consumer population. Such models tend to be very complex and only seldom amenable to analytical solutions.

Computational, service, and data grids, peer-to-peer systems, and ad-hoc wireless networks are examples of open systems where the distinction between resource providers and consumers is blurred. Individual members of the community contribute computing cycles, storage, services, and communication bandwidth to the pool of resources available to the entire community. The same individual could be both a provider and consumer of resources: a provider in some instances and/or for some types of resources, or a consumer in other instances and/or for different types of resources. An efficient and fair utilization of the resources can be obtained only through a scheme that gives incentives to the providers to share their resources and that encourages the consumers to maximize the utility of the received resources. A well-tested model for such a scheme is based on an economic model, in which the resources need to be paid for in a real or virtual currency. This model has the advantage of being provably scalable, and we can successfully reuse or adapt the models that govern the economy in our society.

Economic models are attractive for resource providers, beneficial for the consumers of resources, and have societal benefits. Indeed, providers benefit from contributing their resources and are encouraged to re-invest some of their profits into additional resources. Consumers enjoy fair treatment as the resource allocation is governed by rules that do not depend on the individual consumer. Moreover, providers and consumers have a say in the market and can make their own decisions to maximize their utility and/or profits. When system-centric scheduling policies are replaced by consumer-centric policies the system becomes more responsive to consumer needs and important problems are solved with higher priority. Economic models allow resource allocation and management to be more efficient, the demand and supply is regulated through economic activities and fewer resources are wasted, and excess capacity and overloading are averaged over a very large number of providers and consumers. Resources, e.g., CPU

cycles, main memory, secondary storage, and network bandwidth/latency, are treated uniformly and this can facilitate the design of large-scale distributed systems, such as computational grids. The system is more scalable and decision-making is distributed.

In an economic model, all the participants are considered self-interested. The resource providers are trying to maximize their revenues. The consumers want to obtain the maximum possible resources for the minimum possible price. The large number of participants makes one-to-one negotiations expensive and unproductive.

It is desirable to have a middleman, a broker, mediate access to system resources and consider multiple objectives. The role of a broker is to reconcile the selfish objectives of individual resource providers and consumers with some global, societal objectives, e.g., to maximize the resource utilization of the system.

Several projects proposed approaches based on economic models. Radio bandwidth management for wireless and mobile systems takes advantage of utility and price concepts [2]. The models used to study the benefits and drawbacks of different bandwidth allocation schemes take into account the individual consumer utility as well as pricing structures.

Several systems, including Enhanced MOSIX [1], Nimrod/G [5], Rexec/Anemone [7], Condor [8], Mungi [9], Mariposa [10], Mojo Nation [11], Popcorn [12], SETI@home [13], and Spawn [16], use market-based models for trading computational resources [6]. The efficiency of resource allocation under two different market schemes, commodities markets and auctions are discussed in [17] and [18]. These papers define concepts such as price stability, market equilibrium, consumer efficiency, and producer efficiency and show that the commodities markets are better choices for controlling grid resources than auction strategies. In this paper, we discuss a resource allocation model that takes into account the utility of the resources for the consumers and the pricing structure imposed by the providers. We show how a satisfaction function can express the preferences of the consumer both regarding the utility and the price of the resources. In our model, the brokers are mediating among the selfish interests of the consumers and the providers, and societal interests, such as efficient resource utilization in the system. We report on a simulation experiment to study the behavior of the system in steady state and in transient state. In [3] we use a simpler model based upon a synthetic quantity to represent resource vectors and in [4] we expand the model for a network of resource managers with a tree topology.

This paper is organized as follows. In Section 2, we introduce utility-

price based models for resource allocation in large-scale distributed systems, and in Section 3 present a simulation study for evaluation of such models.

2. Utility, Price, and Satisfaction Functions

We propose to use: (i) a *utility function*, $0 \leq u(r) \leq 1$, to represent the utility provided to an individual consumer, where r represents the amount of allocated resources; (ii) a *price function*, $p(r)$, imposed by a resource provider, and (iii) a *satisfaction function*, $s(u(r), p(r)), 0 \leq s \leq 1$, to quantify the level of satisfaction; the consumer satisfaction depends on both the provided utility and the paid price.

The utility function should be an non-decreasing function of r, i.e., we assume that the more resources are allocated to the consumer, the higher the consumer utility is. However, when enough resources have been allocated to the consumer, i.e., some threshold is reached, an increase of allocated resources would bring no improvement on the utility. For example, if a parallel application could use at most 100 nodes of a cluster, its utility reflected by a utility function does not increase if its allocation increases from 100 to 110 nodes. The above requirements are reflected by the following equations:

$$\frac{du(r)}{dr} \geq 0, \qquad \lim_{r \to \infty} \frac{du(r)}{dr} = 0 \tag{1}$$

Sigmoid functions follow Equation (1) and are often used to model utility. A sigmoid is a tilted S-shaped curve that could be used to represent the life-cycles of living, as well as man-made, social, or economical systems. It has three distinct phases: an incipient or starting phase, a maturing phase, and a declining or aging phase, as shown in Figure 1.

In the context of resource allocation, a sigmoid quantifies the utility provided to an individual when the amount of resources allocated to the consumer increases. We expect the utility to be a concave function and reaches saturation as the consumer gets all the resources it can use effectively. The utility function could be a sigmoid

$$u(r) = \frac{(r/\omega)^{\zeta}}{1 + (r/\omega)^{\zeta}}$$

where ζ and ω are constants provided by the consumer, $\zeta \geq 2$, and $\omega > 0$. Clearly, $0 \leq u(r) < 1$ and $u(\omega) = 1/2$.

The price could be a linear function of the amount of resources:

$$p(r) = \xi \cdot r$$

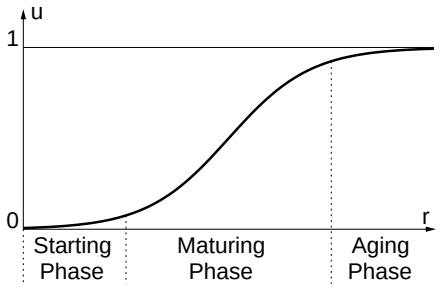

Figure 1. A sigmoid includes three phases: the starting phase, the maturing phase, and the aging phase. Normally consumers do not want the amount of allocated resource to be at the starting phase because the utility is too low; they also do not want the amount of allocated resource to be at the aging phase because they can get a little lower utility while saving a large amount of resources.

where ξ is the *unit price*. The unit price of the resources can be set by convention to a constant, or it can vary based on supply and demand. The variable unit price ξ might be (a) subject to a peer-to-peer negotiation between the consumer and the provider, (b) set in a centralized way similar to a commodity exchange, requiring global information about the supply and demand, or (c) determined by local estimate of the supply and demand. For example, based on the ratio of the allocated resources to the total resources of the provider, a function could give a lower price for the low ratio and a higher price for the high ratio.

A consumer satisfaction function takes into account both the utility provided and the price paid. For a given utility, the satisfaction function should increase when the price decreases and, for a given price, the satisfaction function should increase when the utility u increases. These requirements are reflected in Equation (2).

$$\frac{\partial s}{\partial p} \leq 0, \quad \frac{\partial s}{\partial u} \geq 0 \tag{2}$$

Furthermore, a normalized satisfaction function should satisfy the following conditions:

- the degree of satisfaction, $s(u(r), p(r))$, for a given price $p(r)$, approaches the minimum, 0, when the utility, $u(r)$, approaches 0;
- the degree of satisfaction, $s(u(r), p(r))$, for a given price $p(r)$, approaches the maximum, 1, when the utility, $u(r)$, approaches infinity;

- the degree of satisfaction, $s(u(r), p(r))$, for a given utility $u(r)$, approaches the maximum, 1, when the price, $p(r)$, approaches 0; and
- the degree of satisfaction, $s(u(r), p(r))$, for a given utility $u(r)$, approaches the minimum, 0, when the price, $p(r)$, approaches infinity.

These requirements are reflected by Equation (3) and (4).

$$\forall p > 0, \lim_{u \to 0} s(u, p) = 0, \qquad \lim_{u \to \infty} s(u, p) = 1 \tag{3}$$

$$\forall u > 0, \lim_{p \to 0} s(u, p) = 1, \qquad \lim_{p \to \infty} s(u, p) = 0 \tag{4}$$

A candidate satisfaction function that conforms to the Equation (2), (3), and (4) could be:

$$s(u, p) = 1 - e^{-\kappa \cdot u^{\mu} \cdot p^{-\epsilon}} \tag{5}$$

where μ and ϵ control the sensitivity of s to utility and price, and $\kappa = -\log \alpha$, with α a reference value for the satisfaction function. Similar functions are widely used in the field of microeconomics [15]. A typical shape of the satisfaction function for a sigmoid utility function and a linear price function is shown in Figure 2. Satisfaction decreases after a peak value because continuing to pay more as resources increase after that point does not increase utilization.

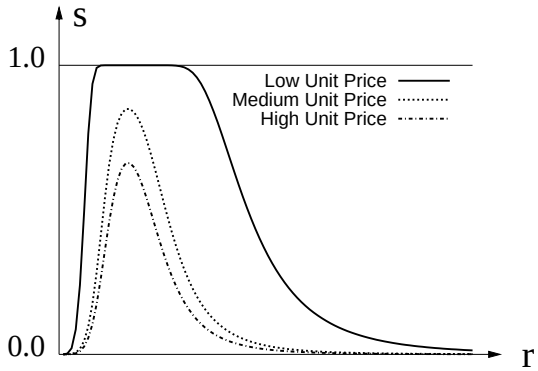

Figure 2. The satisfaction function of the amount of resources r for a sigmoid utility function and linear price functions, $0 \leq s \leq 1$. For the same amount of resources, the higher is the price, the lower is the satisfaction.

Consider a system with n providers offering computing resources and m consumers. For the sake of simplifying the model, we assume that the two sets are disjoint. Call $\mathcal{U}$ the set of consumers and $\mathcal{R}$ the set of providers. The n providers are labeled 1 to n and the m consumers are labeled 1 to m. Consider provider $R_j,\ 1 \leq j \leq n$, and consumer $U_i,\ 1 \leq i \leq m$, that could potentially use resources of that provider.

Let r_{ij} denote the resource of R_j allocated to consumer U_i and let u_{ij} denote its utility for consumer U_i. Let p_{ij} denote the price paid by U_i to provider R_j. Let t_{ij} denote the time U_i uses the resource provided by R_j. Let c_j denote the resource capacity of R_j.

The term "resource" here means a vector with components indicating the actual amount of each type of resource:

$$r_{ij} = (r_{ij}^1\ r_{ij}^2 \ldots r_{ij}^l)$$

where l is a positive integer and r_{ij}^k corresponds to the amount of resource of the k-th type. The structure of r_{ij} may reflect the rate of CPU cycles, the physical memory required by the application, the secondary storage, the number of nodes and the interconnection bandwidth (for a multiprocessor system or a cluster), the network bandwidth (required to transfer data to/from the site), the graphics capabilities, and so on.

The utility of resource of the k-th type provided by R_j for consumer U_i is a sigmoid:

$$u_{ij}^k = u(r_{ij}^k) = \frac{(r_{ij}^k/\omega_i^k)^{\zeta_i^k}}{1 + (r_{ij}^k/\omega_i^k)^{\zeta_i^k}}$$

where ζ_i^k and ω_i^k are constants provided by consumer U_i, $\zeta_i^k \geq 2$, and $\omega_i^k > 0$. Clearly, $0 < u(r_{ij}^k) < 1$ and $u(\omega_i^k) = 1/2$.

The overall utility of resources provided by R_j to U_i could be defined as:

- the product over the set of resources provided by R_j, i.e., $u_{ij} = \prod_{k=1}^{l} u_{ij}^k$, or
- the weighted average over the set of resources provided by R_j, i.e., $u_{ij} = \frac{1}{l}\sum_{k=1}^{l} a_{ij}^k u_{ij}^k$, where a_{ij}^k values are provided by consumer U_i.

We consider a linear pricing scheme $p_{ij}^k = \xi_j^k \cdot r_{ij}^k$, though more sophisticated pricing structures are possible. Here ξ_j^k represents the unit price for resource of type k provided by provider R_j. The amount consumer U_i

pays to provider R_j for a resource of type k is $p_{ij}^k \times t_{ij}$. The total cost for consumer U_i for resources provided by provider R_j is

$$p_{ij} = \sum_{k=1}^{l} p_{ij}^k \times t_{ij}$$

Based on Equation 5, we define the degree of satisfaction of U_i for a resource of the k-th type provided by provider R_j as

$$s_{ij}^k(u_{ij}^k, p_{ij}^k) = 1 - e^{-\kappa_i^k {u_{ij}^k}^{\mu_i^k} (p_{ij}^k/\phi_i^k)^{-\epsilon_i^k}}, \qquad \kappa_i^k, \phi_i^k, \mu_i^k, \epsilon_i^k > 0$$

where μ_i^k and ϵ_i^k control the sensitivity of s_{ij}^k to utility and price; ϕ_i^k and κ_i^k are normalization constants; ϕ_i^k is a reference price; and $\kappa_i^k = -\log \alpha$, with α a reference value for the satisfaction function. Additional details regarding these parameters can be found in Section 3.

The overall satisfaction of consumer U_i for resources provided by R_j could be defined as:

- the product over the set of resources provided by R_j, i.e., $s_{ij} = \prod_{k=1}^{l} s_{ij}^k$, or
- the weighted average over the set of resources provided by R_j, i.e., $s_{ij} = \frac{1}{l}\sum_{k=1}^{l} b_{ij}^k s_{ij}^k$, where b_{ij}^k values are provided by consumer U_i.

The role of a broker is to mitigate access to resources. In this paper, we consider a *provider-broker-consumer model* that involves the set of resource providers $\mathcal{R}$, the set of consumers $\mathcal{U}$, and broker B. This model assumes that a consumer must get all of its resources from a single provider. Brokers have "societal goals" and attempt to maximize the average utility and revenue, as opposed to providers and consumers that have individualistic goals; each provider wishes to maximize its revenue, while each consumer wishes to maximize its utility and do so for as little cost as possible. To reconcile the requirements of a consumer and the candidate providers, a broker chooses a subset of providers such that the satisfaction is above a high water mark and all providers in the subset have equal chances to be chosen by the consumer. We call the size of this subset *satisficing size*, where the word "satisfice" was coined by Nobel Prize winner Herbert Simon in 1957 to describe a behavior of attempting to achieve at least some minimum level of a particular variable instead of striving to achieve its maximum possible value [14]. We denote the satisficing size as σ.

The resource negotiation protocol consists of the following steps (Figure 3):

(1) All providers reveal their capacity and pricing parameters to the broker: $\forall R_j \in \mathcal{R}$ send vectors c_j and ξ_j where each element corresponds to one type of resource.
(2) A consumer U_i sends to the broker:
    (a) the parameters of its utility function: vectors ζ_i and ω_i where each element corresponds to one type of resource,
    (b) the parameters of its satisfaction function: vectors μ_i, ϵ_i, κ_i and ϕ_i where each element corresponds to one type of resource, and
    (c) the number of candidate resource providers to be returned.
(3) The broker performs a matchmaking algorithm and returns a list of candidate resource providers $\mathcal{R}^i$ to consumer U_i.
(4) Consumer U_i selects the first provider from $\mathcal{R}^i$ and verifies if the provider can allocate the required resources. If it can not, the consumer moves to the next provider from the list until the resources are allocated by a provider R_j.
(5) R_j notifies the broker about the resource allocation to U_i.

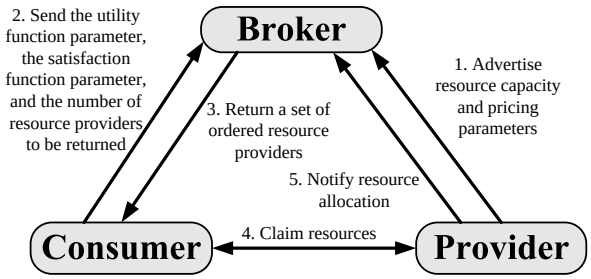

Figure 3. The brokering process: 1) Providers send to the broker their resource capacity and pricing parameters; 2) A consumer sends to the broker a request; 3) The broker executes a brokering algorithm and returns to the consumer a list of resource providers; 4) The consumer selects the first provider from the returned list and confirms that it can satisfy the resource requirements; if it can not, chooses the next provider, until one of the providers allocates the needed resources. 5) If a resource provider allocates resources to the consumer, it notifies the broker about this allocation.

The algorithm performed by the broker is summarized in Figure 4. The amount of resources to be allocated is determined during the algorithm according to a *broker strategy*. Simple strategies would be to allocate the same amount of resources to every consumer, or to allocate to every consumer

a random amount of resources. A better strategy, used by our system, is to allocate an amount of resources such that the utility of each type of resource to the consumer reaches a certain *target utility* τ. To determine the amount of resources allocated to the consumer, the broker uses Equation 6(a) derived from the definition of $u(r)$, Equation 6(b):

$$r = e^{\frac{ln(\frac{\tau}{1-\tau})}{\zeta} + ln(\omega)} \quad (a) \qquad u(r) = \frac{(r/\omega)^{\zeta}}{1 + (r/\omega)^{\zeta}} \quad (b) \qquad (6)$$

BROKERING ALGORITHM
INPUT request *req*, τ, σ, a finite set of resource providers *ps*
OUTPUT a finite set of suggested resource providers *ss*
BEGIN
    determine *amount* so that *req.u*(*amount*) = τ
    FOR each resource provider *rp* in *ps*
        r = *min* (*amount,* available resources of *rp*)
        satisfaction = *req.s* (*req.u*(*r*), *rp.p*(*r*))
    END FOR
    sort elements in *ps* according to their *satisfaction*s
    randomize the sequences of the first σ items in *ps*
    keep the elements in *ps* that have the highest *req.cardinality* satisfaction degrees and remove the rest
    ss = *ps*
END

Figure 4. The brokering algorithm performed by the broker. req contains a utility function u, a satisfaction function s, and a $cardinality$ that specifies the number of resource providers to be returned by the broker. τ is the target utility. σ is the satisficing size.

Several quantities are used in the next section to characterize the resource management policy for broker B and its associated providers and consumers:

(a) The average hourly revenue for providers. The revenue is the sum of revenues for all of its resource types. This average is over the set of all providers connected to broker B.

(b) The consumer admission ratio. This ratio is the number of admitted consumers over the number of all consumers connected to B. A consumer is admitted into the system when there is a provider able to allocate resources, otherwise the consumer is dropped.

(c) The average consumer overall utility. This average is over the set of all admitted consumers connected to broker B.

(d) The average consumer overall satisfaction. This average is over the set of all admitted consumers connected to broker B.

3. A Simulation Study

To evaluate the model presented in Section 2, we present the results of a simulation study of the provider-broker-consumer model, conducted in the YAES simulation environment [19]. The model used for simulation is relatively complex and requires a fair number of choices for the distribution and the range of several random variables.

The behavior of the model is determined by two parameters, τ and σ, chosen by the controlling authority, in our case the broker. $\sigma = 1$ means that the consumer accepts only the best match; when $\sigma > 1$, all providers in the subset chosen by the broker based on a high water mark have an equal chance to be selected.

We compare the system performance of our scheme for several σ values with a *random strategy* where we randomly choose a provider from the set of all providers, without considering the satisfaction function.

Recall the resource types allocated by provider R_j to consumer U_i are denoted as a resource vector $r_{ij} = (r_{ij}^1\ r_{ij}^2 \dots r_{ij}^l)$. For example, if the k-th component is secondary storage, then $r_{ij}^k = 20GB$ is the amount of secondary storage provided by R_j to consumer U_i. The associated utility vector is $u_{ij} = (u_{ij}^1\ u_{ij}^2\ \dots u_{ij}^l)$ and the associated satisfaction vector $s_{ij} = (s_{ij}^1\ s_{ij}^2 \dots\ s_{ij}^l)$.

The *demand-capacity ratio* for a resource type k is the ratio of the amount of resources requested by the consumers to the total capacity of resource providers for resource type k, $\sum_j c_j^k$. The level of demand is practically limited by the sigmoid shape of the utility curve and the finite financial resources of the consumers. In our model, the consumers do not provide the precise amount of resources needed, they only specify their utility function. In the computation of the demand-capacity ratio, for each consumer and each resource, it is assumed that for the requested r_{ij} value the corresponding u_{ij} value is 0.9. The *demand-capacity ratio* vector for all resource types is $\eta_j = (\eta_j^1\ \eta_j^2 \dots \eta_j^l)$. To simplify the interpretation of the results of our simulation we only consider the case when $\eta_j^1 = \eta_j^2 = \dots = \eta_j^l = \eta$.

We simulate a system of 100 clusters and one broker. The number of nodes of these clusters is a random variable normally distributed with the mean of 50 and the standard deviation of 30. Each node is characterized by a resource vector containing the CPU rate, the amount of main memory,

and the disk capacity. For example, the resource vector for a node with one 2 GHz CPU, 1 GB of memory, and a 40 GB disk is $(2GHz, 1GB, 40GB)$. Initially, there is no consumer in the system. Consumers arrive with an inter-arrival time exponentially distributed with the mean of 2 seconds. The service time t_{ij} is exponentially distributed with the mean of λ seconds. By varying the λ value we modify demand-capacity ratio so that we can study the behavior of the system under different loads.

The parameters of the utility function of consumers, i.e., u_{ij}^k, are uniformly distributed in the intervals shown in Table 1. The CPU rate, memory space, and disk space of a request, r_{ij}^k, are exponentially distributed with the mean of $2GHz$, $4GB$, and $80GB$, and in the range of [$0.1GHz$, $100GHz$], [$0.1GB$, $200GB$], and [$0.1GB$, $1000GB$], respectively.

Table 1. The parameters for the simulation are uniformly distributed in the intervals displayed in this table.

Parameter	CPU	Memory	Disk
ξ	[5, 10]	[5, 10]	[5, 10]
ω	[0.4, 0.9]	[0.5, 1.5]	[10, 30]
κ	[0.02, 0.04]	[0.02, 0.04]	[0.02, 0.04]
μ	[2, 4]	[2, 4]	[2, 4]
ϵ	[2, 4]	[2, 4]	[2, 4]
ϕ	[40, 60]	[80, 120]	[1800, 2200]

A consumer request specifies only the parameters of the utility function, ω and ζ, for each element of the resource vector (CPU, Memory, Disk). More precisely, for each element: (a) we generate the amount r_{ij}; (b) we choose a value for ω; (c) set $u = 0.9$ and compute the corresponding value of ζ.

We investigate the performance of the model under various scenarios of demand-capacity ratio, for different target utility and satisficing size. We study the evolution in time and assume that the system reaches steady-state after of a transient period of the first 10^4 seconds. When we study the effect of τ, we use $\sigma = 1$, and when we study the effect of σ, we use $\tau = 0.9$. In each case, we run the simulation 50 times and show the average value and a 95% confidence interval.

We report the average hourly revenue, the consumer admission ratio, the average consumer satisfaction, and the average consumer utility collected over the most recent one hour interval:

(a) As function of time for several levels of target utility, τ, Figures 5, 6, 7,

and 8. We choose λ so that the demand-capacity ratio is 1.0.

(b) As function of time for several levels of satisficing size, σ, Figure 9, 10, and 11. We choose λ so that the demand-capacity ratio is 0.5. The system is capable of handling all consumer requests for all values of σ and the consumer admission ratio is approximately 1.0 for all cases.

(c) As function of time for several levels of demand-capacity ratio, η, Figures 12 and 13.

For a multi-dimensional resource, we let the overall utility be the product of the utility of all types of resource, and we let the overall satisfaction be the product of the satisfaction of all types of resource.

The average hourly revenue increases rapidly during the transient period, slowly decreases due to the resource fragmentation[a] towards the end of the transient period, and reaches a stable value in steady state, as shown in Figure 5. The larger is τ, the more resources are allocated to consumers, and the higher is the average hourly revenue.

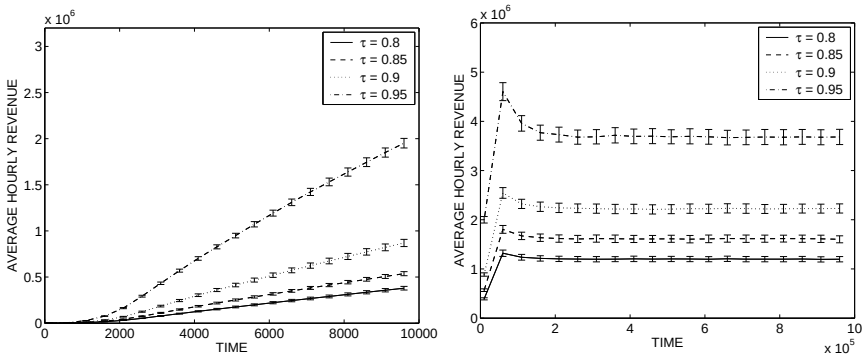

Figure 5. Average hourly revenue vs. time (in seconds) for $\sigma = 1$ and $\tau = 0.8, 0.85, 0.9$, and 0.95. Left: transient period. Right: steady state.

When $\tau = 0.8$, $\tau = 0.85$, and $\tau = 0.9$, the system is capable of handling all consumer requests, the consumer admission ratio is approximately 1.0, and the three plots overlap with each other, as shown in Figure 6. When

[a]Resource fragmentation is an undesirable phenomena; in our environment the amount of resources available cannot meet the target utility value for any request and an insufficient amount of resources is allocated.

$\tau = 0.95$, during the transient period some consumer requests are dropped. As time goes on, the consumer admission ratio slowly increases. More consumers can be admitted into the system due to the fragmentation of the resources because the system is no longer capable of ensuring the required levels for $\tau = 0.95$ and allocates lower amounts of resources. The resource fragmentation effect is noticeable in other graphs as well. In steady state the admission ratio is 1.

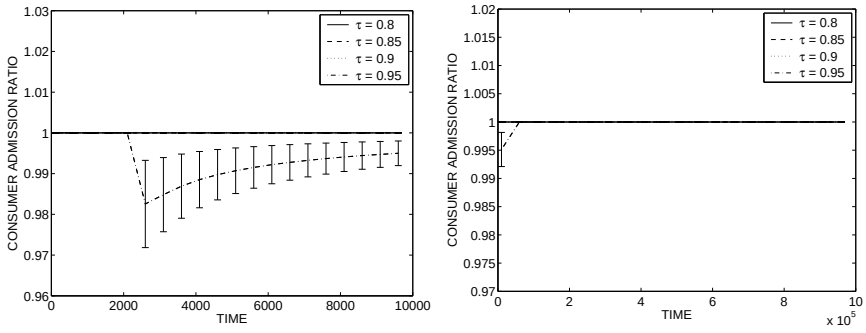

Figure 6. Consumer admission ratio vs. time (in seconds) for $\sigma = 1$ and $\tau = 0.8, 0.85, 0.9$, and 0.95. Left: transient period. Right: steady state.

The average consumer satisfaction decreases slowly during the transient period and then increases and reaches a stable value in steady state, as shown in Figure 7. The average consumer satisfaction is higher when τ is smaller; the smaller is τ, the more consumers can be admitted by resource providers with cheaper prices and these consumers experience higher satisfaction.

The average consumer utility decreases slowly during the transient period because of the resource fragmentation; some resources are allocated to consumers due to their cheaper price although they are not enough to allow the utility to reach the specified τ value, as shown in Figure 8. The average utility reaches a stable value during the steady state. The average consumer utility is lower when τ is smaller.

Figure 9 shows that the average hourly revenue increases rapidly during the transient period. It decreases slowly due to resource fragmentation after the transient period and leads to a stable value in steady state. A small value of σ limits the number of choices the broker has and this restriction leads to lower average hourly revenues. The larger is σ, the higher is the

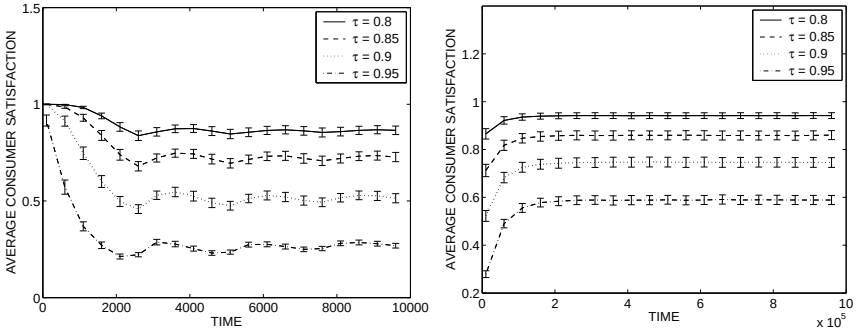

Figure 7. Average consumer satisfaction vs. time (in seconds) for $\sigma = 1$ and $\tau = 0.8, 0.85, 0.9$, and 0.95. Left: transient period. Right: steady state.

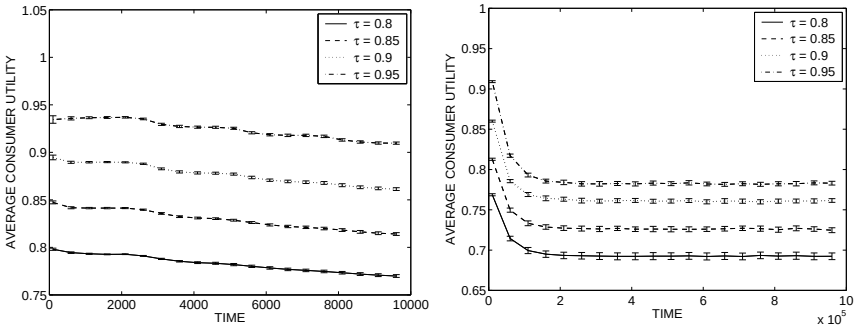

Figure 8. Average consumer utility vs. time (in seconds) for $\sigma = 1$ and $\tau = 0.8, 0.85, 0.9$, and 0.95. Left: transient period. Right: steady state.

average hourly provider revenue. The random strategy, which corresponds to the maximum value of $\sigma =\mid \mathcal{R} \mid$ has the highest average hourly provider revenue. As we shall see shortly the random strategy leads to the lowest consumer satisfaction.

We notice from Figure 10 that the average consumer satisfaction decreases slowly during the transient period and then increases and leads to a stable value in steady state. The average consumer satisfaction is higher when σ is smaller. Indeed, when $\sigma = 1$ we direct the consumer to that resource provider that best matches the request. When we select at random one provider from the set of all providers we observe the lowest average consumer satisfaction. Indeed, when we resort to a random strategy we have a high probability to select a less than optimal match for a given request.

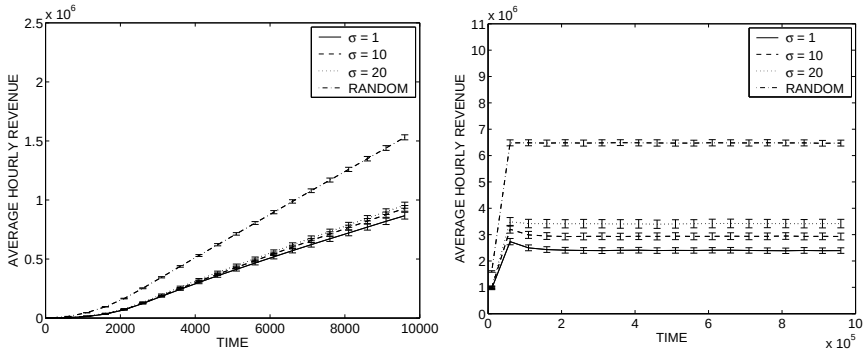

Figure 9. Average hourly revenue vs. time (in seconds) for $\tau = 0.9$ and $\sigma = 1, 10, 20,$ and 50. For the random strategy, $\sigma =\mid \mathcal{R} \mid = 50$. Left: transient period. Right: steady state.

The optimal match is the top ranked element of the candidate resource provider list.

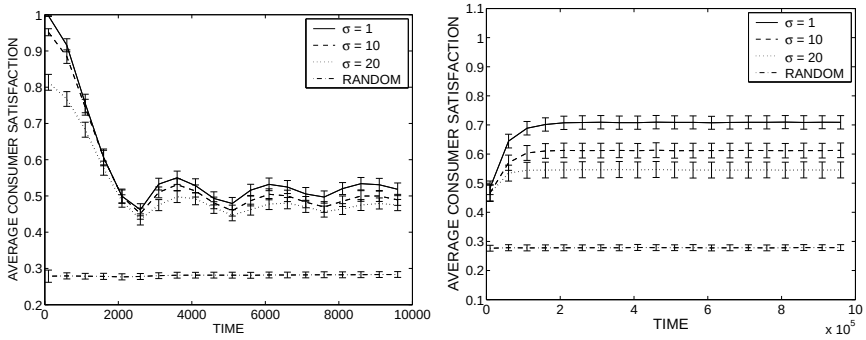

Figure 10. Average consumer satisfaction vs. time (in seconds) for $\tau = 0.9$ and $\sigma = 1, 10, 20,$ and 50. For the random strategy, $\sigma =\mid \mathcal{R} \mid = 50$. Left: transient period. Right: steady state.

Figure 11 shows that the average consumer utility drops slowly during the transient period because of system fragmentation; some resources are allocated to consumers due to their cheaper price, although they are not enough to allow the utility to reach the target value, τ. In steady state the average utility reaches a stable value. The average consumer utility is lower when σ is smaller. The random strategy has the highest average consumer

utility; when σ is large consumers have a better chance to get resources according to the τ values.

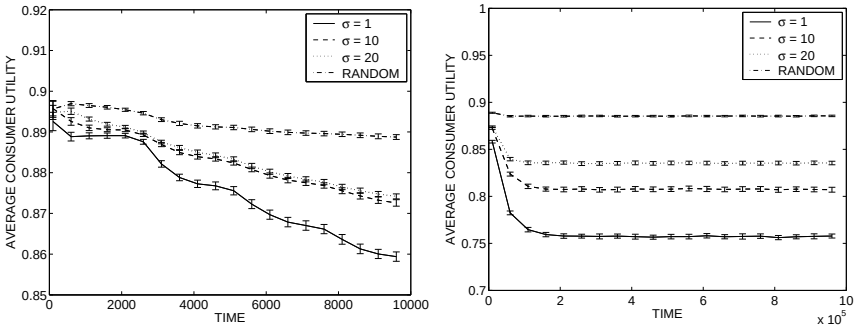

Figure 11. Average consumer utility vs. time (in seconds) for $\tau = 0.9$ and $\sigma = 1, 10, 20,$ and 50. For the random strategy, $\sigma =\mid \mathcal{R} \mid = 50$. Left: transient period. Right: steady state.

Figure 12 (Left) shows that the average hourly revenue at first increases rapidly during the transient period. It decreases slowly due to resource fragmentation after some 10^5 seconds, at the beginning of the steady-state period, and then it reaches a steady value. The larger is η, the higher is the average hourly revenue. Figure 12 (Right) shows that when η is set to 0.25, 0.50, or 0.75, the system is capable of handling all requests and the corresponding plots overlap with each other. When $\eta = 1.0$ some requests are dropped. As time goes on, the consumer admission ratio slowly increases due to resource fragmentation. During the steady state the consumer admission ratio is 1.

The average consumer satisfaction drops during the transient period, then increases, and converges to a steady value, as shown in Figure 13 (Left). The smaller is η, the earlier the system reaches the steady state and the higher is the average consumer satisfaction. The average consumer utility drops during the transient period and reaches a steady value, as shown in Figure 13 (Right). The smaller is η, the earlier the system reaches the steady state and the higher is the average consumer utility.

4. Conclusions and Future Work

Economic models have significant advantages over other models of resource sharing among a large user population. Scalability, fairness, distributed-

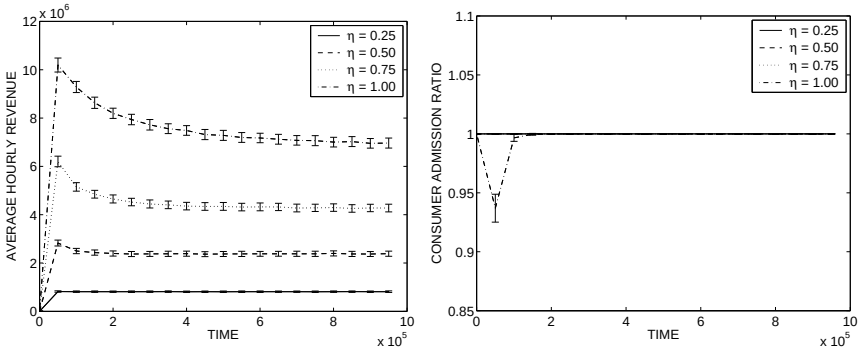

Figure 12. Left: average hourly revenue vs. time (in seconds) for $\tau = 0.9$ and $\sigma = 1$. Right: consumer admission ratio vs. time (in seconds) for $\tau = 0.9$ and $\sigma = 1$.

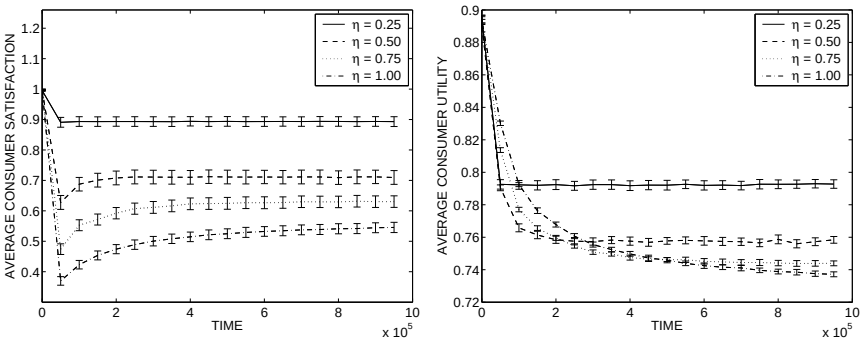

Figure 13. Left: average consumer satisfaction vs. time (in seconds) for $\tau = 0.9$ and $\sigma = 1$. Right: average consumer utility vs. time (in seconds) for $\tau = 0.9$ and $\sigma = 1$.

decision making, and the ability to automate the resource allocation, are only a few of the advantages of economic models. The interest of the distributed systems community in economic models for resource allocation is reflected by a fair number of studies [1,5,6,7,8,9,10,11,12,16,17,18] published in recent years.

The research reported in this paper is part of our effort to introduce resource allocation models based upon utility, price, and satisfaction function [3,4] for large-scale distributed systems. Such models have proved their potential in a different context, when the only resource is the radio bandwidth, the size of the population is limited, and each participant has a unique role, is a consumer [2]. The heterogeneity of a large-scale distributed system, the

large spectrum of resources and demands placed upon these resources, the scale of the system, the autonomy of individual resource providers, and the dual role of individual actors, consumer of some resources and provider for others, add complexity to the models we study. Due to space limitations we cannot fully analyze the properties of utility, price, and satisfaction functions. We only show that the satisfaction reaches an optimum for some level of resource allocation for linear price functions. In this paper and in [3] we consider a model based upon a three party system, provider-broker-consumer while in [4] we consider hierarchical models when the optimization criteria is the optimization of the satisfaction function.

Economic models are notoriously difficult to study. The complexity of the utility, price, and satisfaction-based models precludes analytical studies and in this paper we report on a simulation study. The goal of our simulation study is to validate our choice of utility, price, and satisfaction function, to study the effect of the many parameters which characterize our model, and to get some feeling regarding the transient and the steady-state behavior of our models. We are primarily interested in qualitative rather than quantitative results, we are interested in trends, rather than actual numbers. It is too early to compare our model with other economic models proposed for resource allocation in distributed systems, but we are confident that a model that formalizes the selfish goals of consumers and providers, as well as societal goals, has a significant potential. This is a preliminary study that cannot provide a definite answer to the question posed in the title of the paper. Our intention is to draw the attention of the community to the potential of utility, price, and satisfaction-based resource allocation models.

The function of a broker is to monitor the system and set τ and σ for optimal performance. For example, if the broker perceives that the average consumer utility is too low, it has two choices: increase τ or increase σ. At the same time, the system experiences an increase of average hourly avenue and a decrease of average consumer satisfaction. The fact that increasing utility could result in lower satisfaction seems counterintuitive, but reflects the consequences of allocating more resource; we increase the total cost possibly beyond the optimum predicated by the satisfaction function. The simulation results shown in this paper are consistent with those in [3] where we use a much simpler model based upon a synthetic quantity to represent vector resources.

Backward utility functions allow us to describe the behavior of systems when the amount of allocated resources is reduced. Models supporting

elastic resource allocation using backward utility functions could provide additional insights. In our study we ignored the fact that when the utility or the satisfaction are low, the consumer may reject the allocation. More refined models should take such rejection into account.

A fair number of questions require further investigations including: (a) Are there better alternatives to the utility, price and satisfaction functions we introduced? (b) Is the policy aiming to achieve maximum satisfaction sound, how should we take into account the societal importance of activities carried out by individual resource consumers? (c) How can we apply the models to more complex networks of resource managers? (d) What composition rules should be used to describe the utility and/or the satisfaction for a group of consumers? (e) How can we define more complex utility functions that take into account additional constraints related to quality of service, system reliability, and deadlines?

5. Acknowledgments

This research was supported in part by National Science Foundation grants MCB9527131, DBI0296035, ACI0296035, and EIA0296179, the Colorado State University George T. Abell Endowment, and the DARPA Information Exploitation Office under contract No. NBCHC030137.

References

1. Y. Amir, B. Awerbuch, A. Barak, R. S. Borgstrom, and A. Keren. An opportunity cost approach for job assignment in a scalable computing cluster. *IEEE Transactions on Parallel and Distributed Systems*, 11(7):760–768, 2000.
2. L. Badia and M. Zorzi. On utility-based radio resource management with and without service guarantees. In *Proc. ACM MSWiM 2004, Modelling, Analysis, and Simulation of Wireless and Mobile Systems*, pages 244–251. ACM Press, 2004.
3. X. Bai, L. Bölöni, D. C. Marinescu, H. J. Siegel, R. A. Daley, and I.-J. Wang. A brokering framework for large-scale heterogeneous systems. to appear in Proc. of the 20th IEEE Int. Parallel and Distributed Processing Symp., 2006.
4. X. Bai, L. Bölöni, D. C. Marinescu, H. J. Siegel, R. A. Daley, and I.-J. Wang. Utility, price, and satisfaction-based models for resource allocation in large-scale distributed systems. In preparation.
5. R. Buyya, D. Abramson, and J. Giddy. Nimrod/g: An architecture of a resource management and scheduling system in a global computational grid. In *Proc. of the 4 th Int. Conf. on High Performance Computing in the AsiaPacific Region*, volume 1, pages 283–289, 2001.
6. R. Buyya, H. Stockinger, J. Giddy, and D. Abramson. Economic models for management of resources in peer-to-peer and grid computing. In *Proc.*

of the SPIE Int. Conf. on Commercial Applications for High-Performance Computing, pages 13–25, Denver, USA, August 20-24 2001.
7. B. Chun and D. Culler. Market-based proportional resource sharing for clusters. Technical report, University of California, Berkeley, September 1999.
8. CONDOR. URL http://www.cs.wisc.edu/condor/.
9. G. Heiser, F. Lam, and S. Russel. Resource management in the mungi single-address-space operating system. In *Proc. of the 21 st Australasian Computer Science Conf.*, pages 417–428, February 1998.
10. Mariposa. URL http://mariposa.cs.berkeley.edu/.
11. Mojo Nation. URL http://www.mojonation.net/.
12. N. Nisan, S. London, O. Regev, and N. Camiel. Globally distributed computation over the internet - the popcorn project. In *ICDCS '98: Proc. of The 18th Int. Conf. on Distributed Computing Systems*, pages 592–601, Washington, DC, USA, 1998. IEEE Computer Society.
13. SETI@home. URL http://setiathome.ssl.berkeley.edu/.
14. H. A. Simon. *Models of Man*. Wiley, 1957.
15. H. R. Varian. *Intermediate Microeconomics: A Modern Approach*. Norton, New York, March 1999.
16. C. A. Waldspurger, T. Hogg, B. A. Huberman, J. O. Kephart, and W. S. Stornetta. Spawn: A distributed computational economy. *Software Engineering*, 18(2):103–117, 1992.
17. R. Wolski, J. S. Plank, J. Brevik, and T. Bryan. Analyzing market-based resource allocation strategies for the computational Grid. *The Int. Journal of High Performance Computing Applications*, 15(3):258–281, Fall 2001.
18. R. Wolski, J. S. Plank, J. Brevik, and T. Bryan. G-commerce: Market formulations controlling resource allocation on the computational grid. In *Proc. of the 15th Int. Parallel & Distributed Processing Symposium (IPDPS-01)*, pages 23–27, San Francisco, CA, April 2001.
19. YAES. URL http://netmoc.cpe.ucf.edu/.