A Discriminatory Pay-as-Bid Mechanism for Efficient Scheduling in the Sun N1 Grid Engine

Jochen Stößer[†], Philipp Bodenbenner[†], Simon See[‡], Dirk Neumann[†]

[†]Institute of Information Systems and Management
Universität Karlsruhe (TH)
Englerstr. 14, 76131 Karlsruhe, Germany
lastname@iism.uni-karlsruhe.de

[‡]Asia Pacific Sciene & Technology Center
Sun Microsystems, Inc.
50 Nanyang Avenue, Singapore 639798
simon@apstc.sun.com.sg

Abstract

Grid computing is a promising concept to increase the efficiency of existing computing systems and to cut down on IT expenses by allowing the dynamic access to computer resources across geographical and organizational boundaries. These inter-organizational settings require a scheduling strategy for flexibly and efficiently matching resource requests to idle resources. Market-based mechanisms promise a good fit to grids' strategic and dynamic nature by allowing resource requesters to express valuations in addition to technical metrics. The contribution of this paper is twofold: We present a discriminatory pay-as-bid market mechanism by Sanghavi and Hajek [14] and analytically show that it outperforms market-based proportional share – the currently most prominent grid market mechanism – with respect to both provider's surplus and allocative efficiency. We further illustrate that this mechanism is not a purely theoretical construct but that it can be integrated into the Sun N1 Grid Engine, a state-of-the-art grid scheduler.

1 Introduction

Grid computing denotes a computing model that distributes processing across an administratively and locally dispersed infrastructure. By connecting many heterogeneous computing resources, virtual computer architectures are created, increasing the utilization of otherwise idle resources [7]. A recent report by The Insight Research Corporation projects an increase in worldwide grid spending from $1.84 billion in 2006 to $24.52 billion in 2011 [1].

The Enabling Grids for E-sciencE (EGEE) project is an intriguing example for the value of grid technology in science. EGEE aims at developing a grid infrastructure for more than 240 scientific institutions in 45 countries. The EGEE grid currently consists of more than 36,000 CPUs and 5 Petabytes of storage [6].

The business case for grids is illustrated by the example of Synopsys [19]. Synposys is a world leader in integrated circuit (IC) design and requires massive computer resources for its computer-aided design processes, such as regression testing. In the past, each division within Synopsys maintained its own, separate computing cluster. The full computing power of these systems, however, was only needed in rare occasions to accommodate peak loads on the system. Obviously, this mode of operation thus led to tremendous inefficiencies. Synopsys leveraged Sun Microsystem's grid technology to connect computing resources across divisional boundaries, thus creating a virtual pool of computing resources which can be dynamically accessed on demand. In consequence, the runtime of regression tests could be reduced from about 12 hours to 2 hours.

While the grid scenario is closely related to older allocation schemes for computer resources, such as mainframe allocation, it is somewhat more general because the grid resources might be owned by different organizations. In such inter-organizational settings, scheduling of resource requests becomes a key challenge. What is needed is a set of mechanisms that enable users to discover, negotiate, and pay for the use of grid resources. Classic technical scheduling mechanisms such as first-come-first-serve or fair share are solely built on system-centric measures. Consequently, they do not take into account the strategic and dynamic situation in grids:

- *Scarce resources*: By nature, the concept of grids is all about sharing scarce resources. Excess demand has to be distributed to these resources so as to maximize the value provided by the grid system to its users.

- *Decentralized control*: The scarce resources are spread across organizational boundaries. There is no central-

1530-1605/08 $25.00 © 2008 IEEE

ized and complete knowledge about the state and the availability of these resources but the system depends on these organizations to report and act in a good-natured manner so as to be able to realize this value.

- *Self-interested agents*: The resource requesters and the organizations contributing their idle resources to a grid will generally try to selfishly maximize their individual benefit from participating in the system.

In the light of these characteristics, market-based mechanisms are deemed promising to provide a better fit to grids' strategic and dynamic nature by allowing resource requesters to express valuations in addition to technical metrics. Ultimately, prices are formed which help to balance the dynamic demand and supply in grids. The system can thus induce resource requesters to report truthfully and to distribute excess demand over time. To this end, the contribution of this paper is twofold:

- *Mechanism design:* We present a discriminatory pay-as-bid market mechanism by Sanghavi and Hajek [14] and analytically derive conditions under which it outperforms market-based proportional share – the currently most prominent grid market mechanism – with respect to both provider's surplus (Proposition 1) and allocative efficiency (Proposition 2).

- *Integration into Sun N1GE:* We show that this mechanism is not a purely theoretical construct but that it can be integrated into state-of-the-art grid schedulers to economically enrich the current allocation logics. We illustrate the basic design considerations for the case of the N1 Grid Engine (N1GE), the scheduler of Sun Microsystem's grid platform.

This paper is structured as follows. In Section 2, we discuss related work in the field of grid scheduling, before we introduce the pay-as-bid mechanism in Section 3. We provide an in-depth analysis of this mechanism and compare it to market-based proportional share. We present two design options for how to integrate this mechanism in the Sun N1 Grid Engine in Section 4. We subsequently propose extensions to the basic mechanism and discuss problems which may emerge in this context. Section 5 concludes the paper and points to future work.

2 Related Work

Current resource allocation schemes in grids can be distinguished into technical and market-based schedulers. Technical schedulers are based solely on system-centric measures and aim at maximizing resource utilization and/or balancing the system load. In contrast, market-based schedulers introduce economic principles to grids in order to maximize the economic value provided by such systems; auctions or negotiations explicitly involve the users in the allocation process. Such market mechanisms must be carefully tailored towards the pecularities of the application environment and the trading object. Consequently, a suite of mechanisms has been proposed for a range of grid application scenarios (see [13] and [20] for surveys).

In [15], [3] and [2], the scheduling problem in grids is formalized as an NP-hard periodic combinatorial allocation problem. In [3] and [17], heuristics are developed to migitigate this computational complexity. While these mechanisms account for dependencies between multiple grid resources (e.g. CPU, memory and bandwidth), they rely on strong technical information assumptions, such as knowledge about the time constraints and the resource requirements of applications.

A fundamentally different approach are mechanisms which almost continuously assign resource *shares* to applications based on one-dimensional input only, e.g. single values which represent the users' valuations. Market-based proportional share is the currently most prominent proxy of such mechanisms [5, 9, 16]. With an allocation rule purely based on economic reasoning, e.g. the prominent Vickrey mechanism, all available resources would be given to the resource request with the highest valuation. From a technical viewpoint, however, avoiding *starvation*[1] may be an important consideration. Combining the economic and the technical viewpoint, it may be desirable to give "better" service to high-value processes but to also give at least "some" service to low-value processes in order to avoid starvation.

In the remainder of this section, the Sun N1 Grid Engine (N1GE), a state-of-the-art technical scheduler, and market-based proportional share are presented in more detail.

2.1 Sun N1 Grid Engine

The N1GE is a distributed resource management and scheduling system developed by Sun Microsystems [18]. Being an extension of the Solaris operating system, it administers and dynamically allocates the shared pool of heterogeneous resources such as computing power, memory and licensed software within an organization. The usage of these resources is managed so as to best achieve the goals of the organization, such as productivity, timeliness and level of service. The N1GE has been employed for setting up grids comprising a size of around 500-2,000 CPUs.

The N1GE scheduler consists of a waiting queue with pending jobs and a technical scheduler which subsequently assigns waiting jobs to idle resources (cf. Figure 1). The user submits a job together with a specification of the tech-

[1] In scheduling theory, starvation denotes the fact that low-priority processes are prevented from doing any progress because all resources are assigned to other higher-value processes.

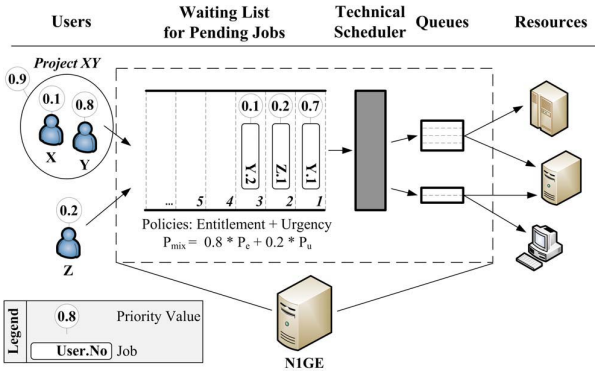

Figure 1: Scheduling process in the N1GE

nical requirements of the job. After receiving the job requests, the scheduler places the jobs in the waiting list of pending jobs. The position of a job in the waiting list is determined by the job's priority. This priority value is calculated by the scheduler using a pre-defined and static mix of different policies. A sample policy mix may comprise manually (by the administrator) set shares for individual users, user groups, a department or a project (also called *Entitlement policy*), an increase of priority for jobs which will reach their deadline soon or that have been waiting for a long time (*Urgency policy*). Additionally, users may be able to sort their own jobs according (*Custom policy*, POSIX) [4].

An example policy mix can look like this

$$P_{mix} = W_e * P_e + W_u * P_u + W_c * P_c$$

where P_{mix} is the dispatch priority, P_e is the normalized entitlement priority (on an interval between 0 and 1) and W_e is the entitlement weighting factor. P_u, W_u, P_c and W_c are defined accordingly for the urgency and custom priorities.

The key drawback of technical schedulers is that static priorities are manually set up and thus do not reflect the fluctuating demand in the system, thus leading to inefficient allocations from an economic viewpoint. To alleviate this problem and to increase efficiency, proportional share mechanisms have been introduced to grid systems.

2.2 Proportional Share

Proportional share allows for resource distribution with shares of unequal size for different users accounting for varying importance among them. Whereas scheduling according to pre-set, fixed shares for different users remains technical, market-based proportional share mechanisms dynamically base the resource share on the users' reported valuations, their "bids". The total amount of available resources is distributed among the requesters according to their reported valuations' fraction of the overall reported valuation across all resource requesters: User i with reported valuation w_i will receive a fraction of $\frac{w_i}{\sum_{j=1}^{n} w_j}$ of the available resource when a group of n users is competing for resource access. Systems using proportional share as allocation scheme have been proposed in [5], [9] and [16]. The problem with market-based proportional share is that it remains allocatively inefficient which we will illustrate in Section 3.3 by means of a numerical example.

In the following, an alternative mechanism is explored – a so-called "discriminatory pay-as-bid mechanism" [14] – that may improve on these present mechanisms.

3 A Discriminatory Pay-as-Bid Mechanism

Let vector $w = (w_1, \ldots, w_n)$ represent the positive bids of the users. The users receive shares $x = (x_1, \ldots, x_n)$, $x_j \in \mathbb{R}^0_+$, $\sum_{j=1}^{n} x_j = 1$, of the perfectly divisible good. These shares are calculated according to the pre-specified allocation mechanism τ. Thus, $x_i = \tau_i(w)$ is the quantity user i is allocated for a given bid vector w.

As is common in mechanism design, the users are assumed to have quasi-linear utility functions: $U_i(x) = v_i x_i - c(x_i)$, with $v_i \in \mathbb{R}_+$ and linear price function $c(x_i) = p_i x_i$ where p_i is user i's unit price, i.e. the price user i would have to pay if she got the whole resource unit ($x_i = 1$). Let $U_P(x)$ be the provider's utility function.

For evaluating and comparing market mechanisms, we first need to define the user behavior, i.e. the users' reporting of w, and a metric. For the former, we choose the widely used solution concept of Nash equilibria. In a Nash equilibrium w^{NE}, no user i can benefit by unilaterally deviating from w_i^{NE}. We interpret Nash equilibria as the final outcome of an iterative process. After each stage, the requesters can adjust their bids based on feedback about the other requesters' bids. Ultimately, we assume that every user knows the vector $v = (v_1, \ldots, v_n)$ consisting of all users' valuations.

A common metric for a mechanism's performance is its *performance ratio* in its Nash equilibrium. The performance ratio of mechanism τ is defined as

$$\frac{U(\tau(w^{NE}))}{U^*} = \frac{\sum_i U_i(\tau(w^{NE})) + U_P(\tau(w^{NE}))}{U^*},$$

i.e. the worst-case ratio of the social welfare generated by the specific mechanism if all bidders play their Nash strategy w_i^{NE} divided by the theoretical optimum U^*.

A key issue when considering the use and (probably more importantly) the usability of markets is the question of how users come up with their valuation functions and interact with the market, i.e. express their valuations in order to eventually arrive at some sort of equilibrium, such as a Nash equilibrium. Human users may be released from the

burden of having to issue requests and offers manually. Instead, software agents may serve so as to hide the grids' complexity from human users by automatically trading resources based on the current resource consumption of applications and configurable bidding rules which automatically derive corresponding valuations [10, 12].

From a mechanism design perspective, we are looking for a mechanism with maximum performance ratio, i.e. the mechanism that maximizes the market's (and thus the grid system's) value across all users. Sanghavi and Hajek [14] propose an allocation mechanism τ^{sh} and show that it generates the optimal performance ratio. While this mechanism has been proposed for the allocation of network bandwidth, the setting essentially generalizes to the allocation of any divisible resource. In the remainder of this section, we will introduce this mechanism and compare it to the market-based proportional share mechanism with respect to provider's revenue and overall welfare. In the following section, we will then discuss possibilities of how such mechanisms can be integrated into state-of-the-art grid schedulers.

3.1 Two Users

For a scenario with two bidders l (low bidder) and h (high bidder), τ^{sh} allocates shares of the perfectly divisible resource as follows:

$$\tau_l^{sh}(w_l, w_h) = \frac{w_l}{2w_h} \text{ and } \tau_h^{sh}(w_l, w_h) = 1 - \frac{w_l}{2w_h}$$

Allocation scheme τ^{sh} is complemented by a so called *pay-as-bid pricing scheme* such that $c(x_i) = p_i x_i = w_i,\ i = l, h$. For two buyers, the worst case performance ratio of τ^{sh} adds up to 87.5% when both buyers have linear valuation functions [14]. In comparison the worst case efficiency of the proportional share mechanism is 82.84% [8].

Assume the low bidding user l has a quasi-linear valuation function $U_l(x_l) = v_l x_l - w_l$ with $v_l \in \mathbb{R}_+$. Further assume the high bidding user h to be characterized by utility function $U_h(x_h) = v_h x_h - w_h$ with $v_h \in \mathbb{R}_+,\ v_h \geq v_l$.

Lemma 1. *In the Nash equilibrium w^{sh} of the pay-as-bid mechanism τ^{sh}, user l bids $w_l^{sh} = \frac{v_l^2}{2v_h}$ and receives a share of $\tau_l^{sh}(w^{sh}) = \frac{v_l}{2v_h}$, whereas user h bids $w_h^{sh} = \frac{v_l}{2}$, thus receiving $\tau_h^{sh}(w^{sh}) = 1 - \frac{v_l}{2v_h}$.*

Proof. In the Nash equilibrium with bid vector w^{sh}, $\frac{\partial U_i(\tau^{sh}(w^{sh}))}{\partial w_i^{sh}} = 0,\ i = l, h$. Consequently, $\frac{\partial U_l(\tau^{sh}(w^{sh}))}{\partial w_l^{sh}} = \frac{v_l}{2w_h^{sh}} - 1 = 0 \Leftrightarrow w_h^{sh} = \frac{v_l}{2}$ and $\frac{\partial U_h(\tau^{sh}(w^{sh}))}{\partial w_h^{sh}} = \frac{v_h w_l^{sh}}{2(w_h^{sh})^2} - 1 = 0 \Leftrightarrow w_l^{sh} = \frac{2}{v_h}\left(\frac{v_l}{2}\right)^2 \Leftrightarrow w_l^{sh} = \frac{v_l^2}{2v_h}$. Inserting w^{sh} in τ^{sh} directly yields $\tau_l^{sh}(w^{sh}) = \frac{v_l}{2v_h}$ and $\tau_h^{sh}(w^{sh}) = 1 - \frac{v_l}{2v_h}$. □

The mechanism τ^{sh} allocates the resource shares in such a way that, in the Nash equilibrium w^{sh}, the low-bidding user l is pushed to zero utility:

$$U_l(\tau^{sh}(w^{sh})) = v_l \tau_l^{sh}(w^{sh}) - w_l^{sh} = v_l \frac{v_l}{2v_h} - \frac{v_l^2}{2v_h} = 0.$$

The high-bidding user h obtains utility of

$$U_h(\tau^{sh}(w^{sh})) = v_h \tau_h^{sh}(w^{sh}) - w_h^{sh} = v_h - v_l,$$

while revenue amounts to

$$r^{sh} = w_l^{sh} + w_h^{sh} = \frac{v_l^2 + v_l v_h}{2v_h}.$$

This constitutes the provider's utility $U_P(.)$ assuming a quasi-linear provider valuation function and zero reservation prices.

The central result of our analysis is that, from a resource provider's point of view, the pay-as-bid mechanism dominates market-based proportional share under certain conditions both with respect to provider's surplus and welfare. To be able to compare the results of both mechanisms, we first derive the provider's surplus generated by market-based proportional share in the Nash equilibrium. In doing so, we assume that, as the mechanism by Sanghavi and Hajek, the proportional share allocation rule is also complemented by the pay-as-bid pricing rule, as proposed in [5].

Lemma 2. *If combined with a linear uniform pricing scheme $\frac{\partial c_i(w)}{\partial w_i} = 1\ \forall i$ (e.g. pay-as-bid), in the Nash equilibrium w^{ps} of the proportional share mechanism τ^{ps}, user l bids $w_l^{ps} = \frac{v_l v_h}{v_l + v_h} - v_l\left(\frac{v_h}{v_l+v_h}\right)^2$ and receives a share of $\tau_l^{ps}(w^{ps}) = \frac{v_l}{v_l+v_h}$, whereas user h bids $w_h^{ps} = v_l\left(\frac{v_h}{v_l+v_h}\right)^2$, thus receiving $\tau_h^{ps}(w^{ps}) = \frac{v_h}{v_l+v_h}$.*

Proof. With proportional share, $\tau_l^{ps}(w) = \frac{w_l}{w_l+w_h}$ and $\tau_h^{ps}(w) = \frac{w_h}{w_l+w_h}$.

Thus, in the Nash equilibrium w^{ps}, $\frac{\partial U_l(\tau^{ps}(w^{ps}))}{\partial w_l^{ps}} = v_l \frac{w_h^{ps}}{\left(w_h^{ps}+w_l^{ps}\right)^2} - 1 = 0 \Leftrightarrow \sqrt{v_l w_h^{ps}} = w_l^{ps} + w_h^{ps} \Leftrightarrow w_l^{ps} = \sqrt{v_l w_h^{ps}} - w_h^{ps}$. Analogously, $\frac{\partial U_h(\tau^{ps}(w^{ps}))}{\partial w_h^{ps}} = v_h \frac{w_l^{ps}}{\left(w_h^{ps}+w_l^{ps}\right)^2} - 1 = 0 \Leftrightarrow v_h w_l^{ps} = \left(w_l^{ps} + w_h^{ps}\right)^2 \Leftrightarrow v_h\left(\sqrt{v_l w_h^{ps}} - w_h^{ps}\right) = v_l w_h^{ps} \Leftrightarrow w_h^{ps} = v_l\left(\frac{v_h}{v_l+v_h}\right)^2$. Inserting w_h^{ps} above directly yields, $w_l^{ps} = \frac{v_l v_h}{v_l+v_h} - v_l\left(\frac{v_h}{v_l+v_h}\right)^2$.

Inserting w_h^{ps} and w_l^{ps} into $\tau^{ps}(w^{ps})$ returns $\tau_l^{ps}(w^{ps}) = \frac{\sqrt{v_l w_h} - w_h}{\sqrt{v_l w_h}} = 1 - \sqrt{\frac{w_h}{v_l}} = 1 - \frac{v_h}{v_l+v_h} = \frac{v_l}{v_l+v_h}$ and $\tau_h^{ps}(w^{ps}) = 1 - \tau_l^{ps}(w^{ps}) = \frac{v_h}{v_l+v_h}$. □

Consequently, market-based proportional share generates revenue of $r^{ps} = w_l^{ps} + w_h^{ps} = \sqrt{v_l w_h} = \frac{v_l v_h}{v_l + v_h}$.

Based on Lemma 1 and Lemma 2, we can now state our first central result:

Proposition 1. *For two users with linear valuation functions with slopes v_l and v_h ($v_l, v_h \in \mathbb{R}_+$ and $v_l \leq v_h$), in the unique Nash equilibria the discriminatory pay-as-bid mechanism generates a larger provider's revenue than proportional share ($r^{sh} \geq r^{ps}$) iff*

$$(\sqrt{2} - 1)v_h \leq v_l \leq v_h.$$

Proof.

$$\begin{aligned} r^{sh} - r^{ps} = \frac{v_l^2 + v_l v_h}{2v_h} - \frac{v_l v_h}{v_l + v_h} &\geq 0 \\ \Leftrightarrow \frac{(v_l + v_h)(v_l^2 + v_l v_h) - 2v_l v_h^2}{2v_h(v_l + v_h)} &\geq 0 \\ \Leftrightarrow (v_l + v_h)^2 &\geq 2v_h^2 \\ \Leftrightarrow v_l &\geq (\sqrt{2} - 1)v_h \end{aligned}$$

Furthermore, as defined earlier, $v_h \geq v_l$ and thus $r^{sh} \geq r^{ps} \Leftrightarrow (\sqrt{2} - 1)v_h \leq v_l \leq v_h$. □

This result is of significant importance in the context of enterprise/campus grid environments, which are the main application domains of N1GE. Especially in those scenarios, we hypothesize that the users can be assumed to have rather similar valuations for grid resources. Finally, we can use these results to also assess the allocative efficiency that is generated in the Nash equilibrium for two jobs, and in line with the results of Sanghavi and Hajek [14] we state the following proposition:

Proposition 2. *For two users with linear valuation functions with slopes v_l and v_h ($v_l, v_h \in \mathbb{R}_+$ and $v_l \leq v_h$) the discriminatory pay-as-bid mechanism generates an equal or larger total welfare in the Nash equilibrium compared to the proportional share mechanism with pay-as-bid pricing, that is $U(\tau^{sh}(w^{sh})) \geq U(\tau^{ps}(w^{ps}))$ for all combinations (v_l, v_h).*

Proof. From our previous results, $U(\tau^{sh}(w^{sh})) = U_l(.) + U_h(.) + U_P(.) = 0 + v_h - v_l + \frac{v_l^2 + v_l v_h}{2v_h} = \frac{v_l^2 + 2v_h^2 - v_l v_h}{2v_h}$ and $U(\tau^{ps}(w^{ps})) = U_l(.) + U_h(.) + U_P(.) = (v_l \tau_l^{ps}(w^{ps}) - w_l^{ps}) + (v_h \tau_h^{ps}(w^{ps}) - w_h^{ps}) + w_l^{ps} + w_h^{ps} = v_l \tau_l^{ps}(w^{ps}) + v_h \tau_h^{ps}(w^{ps}) = \frac{v_l^2 + v_h^2}{v_l + v_h}$.

Thus, $U(\tau^{sh}(w^{sh})) - U(\tau^{ps}(w^{ps})) \geq 0 \Leftrightarrow \frac{v_l^2 + 2v_h^2 - v_l v_h}{2v_h} - \frac{v_l^2 + v_h^2}{v_l + v_h} \geq 0 \Leftrightarrow v_l^2 + v_h^2 \geq 2v_l v_h \Leftrightarrow (v_l - v_h)^2 \geq 0$. □

Consequently, the discriminatory pay-as-bid mechanism of Sanghavi and Hajek [14] not only provides us with a better performance ratio (i.e. worst-case bound) than proportional share, but it outperforms the latter independently of the choice of v_l and v_h.

If considered separately from the allocation scheme, pay-as-bid pricing is a uniform pricing rule as defined in Lemma 2. However, if combined with allocation scheme τ^{sh}, the resulting mechanism as a whole produces discriminatory unit prices; the buyer with a lower bid pays a higher unit price than the high bidder. This volume discount encourages high bidders to bid higher, and thus closer to their true valuation, compared to a scenario with uniform prices where users can potentially benefit from shading their bids downwards.The discriminatory pay-as-bid mechanism is not a "fair" allocation mechanism in that it does not allocate the resource in proportion to the submitted bids but subsidizes the high bidding users. However, this is justified by the increase in overall efficiency.

3.2 n Users

An extension of the above mechanism from two to n buyers was developed in [14]. This mechanism still has the property of a "volume discount", i.e. higher bidders pay lower prices.

For n buyers and a given payment vector $w = (w_1, \ldots, w_n)$, the following allocation rule is proposed:

$$\tau_i^{sh}(w) = \frac{w_i}{w_{max}} \int_0^1 \prod_{j \neq i} \left(1 - s \frac{w_j}{w_{max}}\right) \mathrm{ds}$$

with at least two $w_i \geq 0$ and w_{max} being the maximum bid. This allocation rule simplifies to the optimal mechanism for two buyers given above for $n = 2$.

In contrast to the case for two buyers, it is hard to determine an exact value for the worst case efficiency for an unlimited number of buyers. Instead, [14] calculate the interval [0.8703, 0.875] as bounds for the worst case efficiency.

The proposed mechanism is still close to the theoretical maximum worst case efficiency, i.e. 87.5%. But a guarantee that mechanism τ^* is the optimal one can no longer be given.

Even if machines are assumed to be obedient, a "lying auctioneer" could be a problem in this mechanism (cf. [11]). The allocation rule is based on the the highest bid, w_{max}, which is not publicly known and could thus be manipulated by the provider to change the allocation in his favor. Therefore, in practice it might be necessary to somehow publish and verify w_{max}.

3.3 Numerical Example

In this section, proportional share and the discriminatory pay-as-bid mechanism are compared concerning their allocation and the resulting efficiencies by means of a simple numerical example. Table 1 provides a brief overview of the results of this example.

There are two divisions within a company – divisions L and H – which execute computational jobs on a shared pool of computing resources.

Division H temporarily demands more resources than division L, which is reflected in a higher valuation for the computing resources: $U_L(x_L) = 2.1x_L - w_L$ and $U_H(x_H) = 5x_H - w_H$. The provider's valuation function is given by $U_P(x) = w_L + w_H$. Each division sends one resource request to the central market-based scheduler. Attached to both jobs are the corresponding valuations. Now we determine the allocation of resource shares to the jobs for both market-based mechanisms.

For the two jobs and the given valuation functions, the proportional share mechanism arrives at the Nash equilibrium with bid vector $w^{ps} = (w_L^{ps}, w_H^{ps}) \approx (0.437, 1.041)$. In this equilibrium point, $x_L^{ps} = 0.296$ is allocated to division L and $x_H^{ps} = 0.704$ is allocated to division H. None of the divisions has an incentive to unilaterally deviate from these bids. The unit prices are $p_L^{ps} = p_H^{ps} = 1.478$. Hence, the proportional share mechanism generates an overall social welfare of $U(x^{ps}) = 4.142$. This corresponds to a performance ratio of 82.84% compared to the optimal allocation. In this optimal allocation, the high bidding user is given a resource share of 1.0 whereas the low bidder receives nothing. This allocation would create the maximum social welfare of 5.

The discriminatory pay-as-bid mechanism reaches the Nash equilibrium for a bid vector $w^{sh} = (w_L^{sh}, w_H^{sh}) = (0.441, 1.05)$. With these bids a resource share of $x_H^{sh} = 0.79$ is assigned to division H and the remaining $x_L^{ps} = 0.21$ are allocated to L. This allocation generates a total social welfare $U(x^{sh}) = 4.391$, which stands for a performance ratio of 87.82%. Applying the pay-as-bid-rule to this allocation exemplifies the "volume discount" for the high bidding division H. It pays a unit price of $p_H^{ps} = 1.329$ whereas division L has to pay a notably higher unit price of $p_L^{sh} = 2.1$.

Whereas in this example the provider's revenue created by the discriminatory pay-as-bid mechanism ($r^{sh} = 1.491$) is only slightly higher than with the proportional share mechanism ($r^{ps} = 1.478$), overall welfare increases by almost 5% from $U(x^{ps}) = 4.142$ to $U(x^{sh}) = 4.391$. While here both the high bidder and the provider benefit from switching to the pay-as-bid mechanism, as showed above, this gain mainly comes at the expense of the low bidder, whose utility becomes zero and who is thus indifferent to not participating at all.

	Prop. Share	**Pay-as-Bid**
Optimal Bids	$w_L^{ps} = 0.437$ $w_H^{ps} = 1.041$	$w_L^{sh} = 0.441$ $w_H^{sh} = 1.05$
Allocation	$x_L^{ps} = 0.296$ $x_H^{ps} = 0.704$	$x_L^{sh} = 0.21$ $x_H^{sh} = 0.79$
Unit prices	$p_L^{ps} = p_H^{ps} = 1.478$	$p_L^{sh} = 2.1$ $p_H^{sh} = 1.329$
Utilities Users	$U_L(x_L^{ps}) = 0.185$ $U_H(x_H^{ps}) = 2.479$	$U_L(x_L^{sh}) = 0$ $U_H(x_H^{sh}) = 2.9$
Provider's Revenue	$r^{ps} = 1.478$	$r^{sh} = 1.491$
Social Welfare	$U(x^{ps}) = 4.142$	$U(x^{sh}) = 4.391$
Performance ratio	82.84%	87.82%

Table 1: Numerical example

In certain cases, the provider might be willing to actually sacrifice revenue in return for a higher social welfare. E.g. in the case of Synopsys, attributing the resources to the division with the higher valuation may be more important than creating revenue from internal sources. Especially in cases when provider revenue increases, one option might be to "subsidize" the lower bidder in order to convince this bidder to participate in the pay-as-bid mechanism. It will be an interesting avenue for future research to explore how such an incentive may be implemented in the pay-as-bid mechanism and how it changes the users' strategic considerations.

4 Integration into the Sun N1 Grid Engine

The following section is dedicated to the integration of the market-based mechanism into the technical scheduling environment of N1GE. Two possible approaches are evaluated: 1) a modular extension of N1GE, with an additional market-based policy and 2) the displacement of the current technical scheduler with the discriminatory pay-as-bid mechanism. The analysis is concluded with a comparison of both approaches in Section 4.3.

4.1 The Pay-as-Bid Mechanism as Additional Policy

In the following section, the application of the discriminatory pay-as-bid mechanism as a new policy in N1GE is discussed. The objective is to use the market-based mechanism as an instrument for partioning of the priority value. Analogous to the current N1GE system the users submit their jobs along with a specification document to state the job's resource requirements. In addition the users send a

one-dimensional bid (a single real number) to signal their valuation of the submitted job. These bids are then used by the discriminatory pay-as-bid mechanism to allocate shares of the priority value, which then again are used to determine the order in the waiting list of pending jobs. After sorting the jobs according to their assigned priority value the technical scheduler traverses the waiting list and picks a job for execution. Since the jobs can have different resource requirements the first job that fits the specifications of a currently available resource is chosen. Thus it could happen that a job that is heading the waiting list is not chosen due to its resource requirements. But still, taking up one of the front slots in the waiting list increases the possibility of early execution. Thus it is an incentive for jobs, respectively their owners, to compete for a high share of the priority value. As long as the job is not released for execution by the technical scheduler, a user can always abort his own jobs.

In addition to this new policy, the currently available policies in N1GE will still be at the administrator's disposal. This can be rather easily be achieved by adding the new *discriminatory pay-as-bid policy* to the existing policy mix (see figure 2). The value which is determined by the market-based mechanism is weighted and added to the total priority value of a job:

$$\hat{P}_{mix} = P_{mix} + W_{sh} * N_{sh}$$

Thereby, $\hat{P}_{mix}$ is the new dispatch priority, P_{mix} the priority value generated with the current N1GE policy mix and N_{sh}, W_{sh} the discriminatory pay-as-bid priority (on an interval between 0 and 1) respectively the corresponding weighting factor for this new policy.

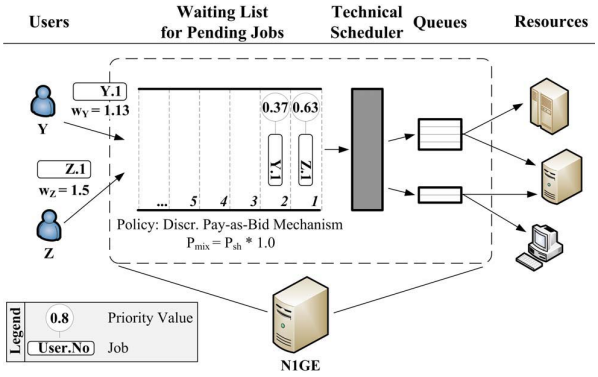

Figure 2: Modified scheduling process in N1GE with the discriminatory pay-as-bid mechanism as additional policy

The main objective of the integration process is to preserve the economic properties of the discriminatory pay-as-bid mechanism. Therefore, the approach is to incorporate the mechanism mostly unchanged into a N1GE environment that is tailored to the needs of the mechanism. To guarantee a sound concurrence of N1GE system and the discriminatory pay-as-bid mechanism as a new policy, a number of extensions and modifications have to be considered on both sides. Adaptation of the discriminatory pay-as-bid mechanism requires a number of considerations in different areas. The major challenges are evaluated next and suitable solutions are proposed. Issues concerning the payment and the enforcement thereof, are not in the scope of this work and thus it plays only a tangential role in the evaluation.

4.1.1 Re-evaluation of the Priority Value

Recalculation of the priority values of all pending jobs is necessary whenever a new job enters the waiting list. Every new bid changes the priority values of all waiting jobs. In addition, a re-evaluation falls due for each updated bid that is received in the system. A job that leaves the waiting list for any reason does not require an update of the waiting list. The re-evaluation is processed periodically according to a pre-defined interval. A continous evaluation of newly arrived jobs would put to much additional load on the provider. Jobs, arriving at the waiting list during such an interval, are gathered and inserted into the waiting list within the next interval. In the worst case, utilisation of this periodic allocation scheme delays all arriving jobs by the pre-defined interval. This is undesirable especially for time critical and urgent jobs. Currently, the bidding interval, or scheduler interval as it is called in N1GE, is set to 2-3 seconds. The testbed environments work with a 15 second interval on default. Consequently, these values seem to be reasonable references for the re-evaluation interval.

4.1.2 Bid Updating

Since re-evaluation of the priority value is indispensable in the N1GE scenario anyway, bid updating imposes no further computational effort on the scheduler. But it tremendously increases the communication effort. For each bid that is updated an additional message, carrying the single real value, has to be sent by the user and processed by the provider. However, allowing bid updates cannot interfere with the economic properties of the mechanism, since the Nash equilibrium is considered to be the final point of repeated plays. Thus, bid updates can even improve the speed of convergence since users can always adjust their bids towards the equilibrium. These adjustments are based on the behaviour of the other jobs in the queue. The bid updates can be interpreted as part of the 'myopic best response' bidding strategy which leads to the Nash equilibrium. A received bid, respectively bid update, for a particular job is valid until another modified bid for this job is submitted by the user. Another feature of bid updates is the prevention

of job starvation. Whenever job starvation impends the job, the user can intervene and update the bid.

4.1.3 Feedback

For a convergence to the Nash equilibrium point, the users need feedback on their bids to see what resource shares they received. Using this partial market information, they will adjust their bids in "myopic best responses" to finally reach the Nash equilibrium point. Consequently, an accurate feedback is essential to preserve this equilibrium concept. By now, N1GE scheduler does not support this kind of direct feedback. The reporting tool for the waiting list, named *qstat* in N1GE, shows the priority value for each job. This priority value is the accumulated value for all policies that are part of the policy mix which is employed in the specific scenario. Thus, the users are not able to determine the fraction of the priority value that was calculated based on their bid. The speed of convergence to the Nash equilibrium can be further improved by publishing the whole waiting list in an anonymized form. This enables the users to monitor both the priority value and the corresponding position in the waiting list.

4.2 The Pay-as-Bid Mechanism as Scheduler

In this scenario the discriminatory pay-as-bid mechanism is applied as a scheduler to directly allocate the resources to the bidding users. In contrast to the scenario assumed so far, the mechanism does not calculate the respective priority value for each bidding user, but it determines the actual share of resources a user gets.

The major challenge, that has to be mastered in this scenario, is the architectural limitation of N1GE, which only allows a single task to be executed on each slot at a time. Therefore resources can no longer be assumed as being perfectly divisible. There are two solution concepts to handle this restriction of N1GE. Firstly, the N1GE's limitation can be attacked from the mathematical side. This requires a modification of the mechanism's allocation rule to allow discrete shares only. Likewise, the problem can be tackled from the technical side. Employing virtualization tools could restore the perfect divisibility of the resources. Both solution concepts are discussed in the following – showing advantages and disadvantages of both.

4.2.1 Discrete Resources

The main challenge is to modify the current mechanism in such a way that it can be used for discrete resources. The calculated shares may only equal a multiple of $\frac{1}{m}$, where m is the number of currently idle resources. By putting up a linear integer program a completely new mechanism is created. Since the discriminatory pay-as-bid mechanism is already the optimal mechanism (at least for two users) the linear integer program can not improve the current allocation and will most likely result in losses in the user's utilities and overall efficiency. In addition, further issues have to be resolved. Firstly, the impact on the computational tractability has to be analysed. This is mainly dependent on the complexity class of such a linear integer program. Secondly, the convergence to a (Nash) equilibrium needs a more detailed examination. Taking all the considerations into account, this approach of restricting the mechanism to discrete resources is not very promising. Solving the linear integer program does hardly scale and will get computational intractable for a large number of jobs.

4.2.2 Virtualization

This approach tries to solve the restriction from the technical perspective. Virtualization could be the solution to the "indivisibility constraint". It is a technique for hiding the physical characteristics of the underlying computing resources from the way in which other systems or end users interact with those resources. Not only homogeneous resources can be joined to form a virtual resource, with virtualization any type of resource can be combined. State-of-the-art virtualization tools, such as e.g. VMWare (http://www.vmware.com), can reach an almost perfect divisibility. Instead of directly addressing resources, the scheduler communicates with a virtual machines which again manages the resources as such. To the scheduler, the available resources appear to be one big resource. This would again fit the requirements of the discriminatory pay-as-bid mechanism, which is designed for allocation of a single unit of a perfectly divisble resource.

In the following, a 'virtual resource' is referred to as a single logical representation of multiple resources. Furthermore, it is assumed that virtualization is employed as a mean to enable the usage of the discriminatory pay-as-bid mechanism for scheduling in N1GE. The adaptation of the mechanism to discrete resources is discarded due to the drawbacks of such an approach discussed earlier. Figure 3 shows the modified N1GE scheduling process.

The challenges in this application scenario differ essentially from the ones identified in the first scenario. For application of the discriminatory pay-as-bid mechanism as a direct scheduler, N1GE's central architecture and workflow has to be radically changed. The market mechanism is not integrated as an modular extension but directly embedded into N1GE's core structure.

4.2.3 Mode of Allocation

The scheduler allocates the jobs for a single time slot only. Consequently, a new allocation is calculated for all cur-

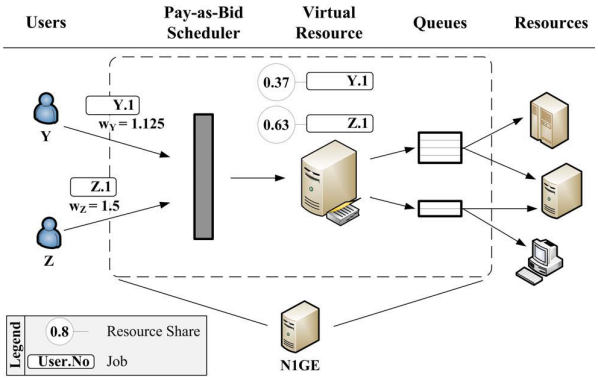

Figure 3: Modified scheduling process in N1GE with the discriminatory pay-as-bid mechanism as direct scheduler

rently running jobs after each time slot. This mode of allocation allows a flexible adjustment of short-time changes in the demand and supply situation. None of the resources is blocked for more than one time slot. The downside is an increased effort, both on the users' and provider's side. The users need to constantly monitor the allocation process for their running jobs and have to submit bids for every allocation interval. Otherwise, when no bid is received for a particular job, this job is suspended. Furthermore, the users are no longer able to determine the total execution time and corresponding payments for their jobs in advance. Moreover, the allocation for a single time slot burdens the provider with migration and re-prioritization of (nearly) all jobs after each allocation interval. However, this type of allocation does not affect the economic properties of the market mechanism. This can be easily proved since the method equals the initially proposed scenario by [14], in which the bandwidth of a network link was allocated for a single predefined time slot.

4.2.4 Job Length

A further factor that is not directly employed in the allocation scheme yet, is the job length. A major problem in this context is the determination of the job length in advance, i.e. before execution of the job. This is particularly a problem in an environment with heterogeneous resources. As already discussed in the previous section, the job length can indirectly be incorporated by allocating the resource for a single time slot. The total payment of the user is determined by summing up the fractional payments made for each allocation interval. Consequently, the job length does not need to be known in advance. Furthermore, it can be guaranteed that the user only pays for units of the resource that he really used, which adds an additional notion of fairness compared to a fixed price, per job and time slot, paid in advance.

4.2.5 Feedback

Finally, a comprehensive feedback is the base for sustaining the convergence to the Nash equilibrium. The user needs to be informed about the share that he receives. This enables the user to see the impact of his bid on the resource share that is finally allocated to him. Analogous to the "policy scenario" the provider has to put the level of allocated resource share at the user's disposal.

4.3 Implications

The two integration scenarios discussed in the previous subsections represent different approaches of extending the N1GE with economic features. Introducing the discriminatory pay-as-bid mechanism as an additional policy constitutes an modular extension of the N1GE. Most of the current architecture and structure retains unchanged. Solely the payment system requires major modifications. The policy mix still covers the existing policies besides the newly introduced market-based principle. In contrast, the direct scheduling scenario requires modifications and replacement of core components. Consequently, the implementation effort is much higher with this approach. In addition, the allocated resources are limited to a single type whereas the modular extension supports heterogeneous resources. On the other hand, giving the discriminatory pay-as-bid mechanism direct access to the resources entails a multitude of advantages. For instance, a waiting list is dispensable since all jobs get a share of the resource. Furthermore, the additional in-between scheduler can be left out which removes complexity from the scheduling process. Both scenarios offer promising enhancements of N1GE. The modular extension offers a quick solution for incorporation of the discriminatory pay-as-bid mechanism, whereas the higher flexibility of the second approach comes at the expense of higher implementation effort.

5 Conclusion

The extended model of the Sanghavi/Hajek pay-as-bid mechanism is a promising addition to the N1GE scheduler. Employing a market-based mechanism for resource allocation in grids offers new possibilities on both sides, for providers as well as for buyers. Current technical schedulers require an administrator to specify user weights based on these users relative importance, regardless of the dynamic demand and supply situation, leading to inefficiencies. To this end, the Sanghavi/Hajek pay-as-bid mechanism allows flexible reactions to changes in the demand and supply situation. Moreover, it offers an elaborated pricing

scheme where prices reflect the current market situation and induce users to report their true valuations to the system. The administrator no longer needs to adjust the weights manually but the users can directly express the urgency of their jobs. Furthermore, the market prices can be leveraged for usage-based accounting of shared computer resources.

In comparison to other market-based mechanisms, the discriminatory pay-as-bid mechanism scores with its ease-of-use. In addition, it has an increased worst case performance ratio as compared to market-based proportional share and is close to the theoretical maximum. Above all, the extended model imposes a very low communicational and computational burden on the scheduling process and allows for real-time allocations.

Further work has to be done on analyzing the extensions to the basic mechanism and their impact on the mechanism's economic properties. We explicitly mentioned the need to incentivize the low bidding user to participate in the grid. It would be very interesting to actually implement the mechanism within the N1GE scheduler. This would allow us to examine the mechanism's behavior and performance in practice. A decentralized version of the mechanism would be desirable to support decentralized waiting queues as well. This might be necessary to keep the N1GE scheduler applicable for very large clusters ($> 20,000$ cores), which will be demanded in the near future.

Acknowledgements

We would like to thank the anonymous reviewers for their valuable comments. This research was partially supported by the EU IST programme under grant #034286 SORMA – "Self-Organizing ICT Resource Management".

References

[1] Anonymous. Grid Computing: A Vertical Market Perspective 2006-2011. Technical report, The Insight Research Corportation, Boonton, NJ, USA, 2006.

[2] A. AuYoung, B. Chun, A. Snoeren, and A. Vahdat. Resource allocation in federated distributed computing infrastructures. *Proceedings of the 1st Workshop on Operating System and Architectural Support for the On-demand IT InfraStructure*, 2004.

[3] R. Bapna, S. Das, R. Garfinkel, and J. Stallaert. A Market Design for Grid Computing. Technical report, Technical report, Department of Operations and Information Management, University of Connecticut, 2005.

[4] C. Chaubal. Scheduler Policies for Job Prioritization in the Sun N1 Grid Engine 6 System. Technical report, Sun BluePrints Online, Sun Microsystems, Inc., Santa Clara, CA, USA. http://www.sun.com/blueprints/1005/819-4325.pdf, 2005.

[5] B. N. Chun and D. E. Culler. Market-based proportional resource sharing for clusters. Technical report, University of California at Berkeley, CA, USA, 2000.

[6] Enabling Grids for E-sciencE (EGEE) project. Website, 2007. Available online at http://www.eu-egee.org/; visited on August 29th 2007.

[7] I. Foster, S. Tuecke, and C. Kesselman. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International Journal of High Performance Computing Applications*, 15(3):200–222, 2001.

[8] R. Johari and J. N. Tsitsiklis. Efficiency Loss in a Network Resource Allocation Game. *Mathematics of Operations Research*, 29(3):407–435, 2004.

[9] K. Lai, B. Huberman, and L. Fine. Tycoon: A Distributed Market-Based Resource Allocation System. *Arxiv preprint cs.DC/0404013*, 2004.

[10] J. K. MacKie-Mason and M. P. Wellman. Automated markets and trading agents. In L. Tesfatsion and K. L. Judd, editors, *Handbook of Computational Economics*.

[11] R. T. Maheswaran and T. Basar. Coalition formation in proportional fair divisible auctions. In *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 25–32, 2003.

[12] D. Neumann, S. Lamparter, and B. Schnizler. Automated bidding for trading grid services. In *Proceedings of the European Conference on Information Systems (ECIS)*, 2006.

[13] D. Neumann, J. Stößer, and C. Weinhardt. Bridging the Adoption Gap – Developing a Roadmap for Trading Grids. *Electronic Markets*, 2007, forthcoming.

[14] S. Sanghavi and B. Hajek. Optimal Allocation of a Divisible Good to Strategic Buyers. *43rd IEEE Conference on Decision and Control*, 2004.

[15] B. Schnizler, D. Neumann, D. Veit, and C. Weinhardt. Trading Grid Services-A Multi-attribute Combinatorial Approach. *European Journal of Operational Research, forthcoming*, 2006, forthcoming.

[16] I. Stoica, H. Abdel-Wahad, K. Jeffay, S. Baruah, J. Gehrke, and C. Plaxton. A proportional share resource allocation algorithm for real-time, time-shared systems. *IEEE Real-Time Systems Symposium*, 1996.

[17] J. Stößer, D. Neumann, and A. Anandasivam. A truthful heuristic for efficient scheduling in network-centric grid os. In *Proceedings of the 15th European Conference on Information Systems (ECIS)*, 2007.

[18] Sun Microsystems, Inc. *Sun N1 Grid Engine 6 User's Guide*. Santa Clara, CA, USA. http://docs.sun.com/app/docs/doc/817-6117/, 2004.

[19] Sun Microsystems, Inc. Grid Computing Solutions. Website, 2007. Available online at http://www.sun.com/software/grid/; visited on August 29th 2007.

[20] R. Wolski, J. Plank, J. Brevik, and T. Bryan. Analyzing Market-Based Resource Allocation Strategies for the Computational Grid. *International Journal of High Performance Computing Applications*, 15:258–281, 2001.