# A Global Benefit Maximization Task-Bundle Allocation

Meiguang Zheng, Zhigang Hu, Peng Xiao, Kai Zhang

# A Global Benefit Maximization Task-Bundle Allocation

Meiguang Zheng[1], Zhigang Hu[1], Peng Xiao[1], Meixia Zheng[2], Kai Zhang[3]

[1]School of Information Science and Engineering, Central South University, Changsha, China
[2]School of Foreign Language, Xinyu College, Xinyu, China
[3]Central Research and Design Institute, ZTE Nanjing, Nanjing, China
zhengJo@gmail.com
zghu@csu.edu.cn

**Abstract.** Obtaining maximal benefit is usually the most important goal pursued by Grid resource/service provider. As providers and users being non-cooperative inherently, it is a fundamental challenge to design a resource allocation strategy which seems to be fair. In order to adapt to large-scale Grid environment, we adopted a hierarchical grid structure with bundle tasks to describe the Grid system. A model called Intra-Site Cooperative-game of Task-bundle (ISCT) was proposed, in which all subordinate resources participated in making profits. We calculated task market price based on the theoretical proof that the system would gain maximal global benefit if and only if it was in a balanced state. Then we determined the task allocation solution with solving the task assignment amount vector. An Intra-Site Global Benefit Maximization Allocation for Task-bundle (ISGBMAT) was presented, which converted the Grid task-bundle allocation problem into an iteration process involving retail price, market price and assignment amount of tasks. Extensive simulation experiments with real workload traces were conducted to verify our algorithm. The experimental results indicated that ISGBMAT could provide an effective solution with global benefit and completion time optimization and also adapt to dynamic Grid market.

**Keywords:** cooperative game, pricing, global benefit maximization, intra-site allocation, grid computing.

## 1 Introduction

As one of the fundamental challenges in enabling computational grid [1] systems, task scheduling has been widely studied in the last decade. Large-scale scientific or engineering applications are usually mapped onto multiple distributed resources, so how to provide an efficient strategy without complicated coordination is a hard problem. Considering grid resource owners being similar to rational market participants, economic model becomes a topic of great interest in grid task scheduling strategies which are designed similarly to market supply-demand mechanisms[2][3]. Although the grid resource provider, also being the scheduling decision maker, usually attempts to obtain maximum commercial profits, it has to design a seemingly

fair allocation strategy since an obvious unreasonable allocation will arouse users' dissatisfaction. Current market-oriented paradigms have two following limitations: (1) most models have a selecting tendency in the bargain process. For example, commodity market model is biased towards grid user while auction mechanism favors resource owner; (2) the difficulty of resource pricing obscures evaluating allocation performance.

For above two reasons, a hierarchical grid structure with bundles of individual tasks in a Bags-of-Tasks (BoT) fashion [4] is used in this paper to characterize the grid system of large-scale environment. We propose a novel model called Intra-Site Cooperative-game of Task-bundle (ISCT), in which each subordinate resource participates in making profits. We provide a pricing scheme including market price and retail price of tasks, based on the theoretical proof that the system will obtain maximal global benefit if and only if it is in a balanced state. We determine the task allocation via solving the task assignment amount vector. Then the resource allocation problem is converted into an iterative computing process involving retail price, market price and assignment amount of tasks. Thus an Intra-Site Global Benefit Maximization Allocation for Task-bundle (ISGBMAT) is presented. Our contributions are theoretical and experimental: (1) We propose a task scheduling mechanism ISGBMAT which realizes global benefit maximization of the system; (2) We analytically and experimentally demonstrate important features of ISGBMAT, including efficient outcome with performance optimization and adaptive property of market self-selection.

The remainder of this paper is organized as follows. Section II reviews related work and compares them with the work proposed in this paper. Section III presents the grid system model, formulates specific problem and proposes our pricing scheme. Section IV derives the ISGBMAT algorithm and analyzes its key properties. Section V presents experimental details and simulation results. The final Section VI concludes the paper and highlights future directions.

## 2 Related Work

The work in this paper will focus on task scheduling algorithm. The problem of resource allocation and grid scheduling has been extensively investigated. Up to now, scheduling of tasks on the grid remains to be a complex optimization problem in which different objectives of grid system and users are need to be considered.

Kim [5] conducted a survey to provide detailed comparisons among lots of traditional resource allocation algorithms, which argued that policy performance is much affected by different user QoS demand. To this end, with the progressing of grid technologies towards a service-oriented paradigm and the developing of users' more sophisticated requirements, researchers have provided many economy-based approaches for more efficient management of grid resources.

Buyya [6] proposed a famous distributed computational economy-based framework called the Grid Architecture for Computational Economy (GRACE) [7], and developed a grid resource broker called Nimrod-G [8] to support economic scheduling algorithms for scheduling parameter sweep applications on the grid. Later,

according to different metrics of either system or users in specified application scenarios, researchers have designed the improved commodity-market-based algorithms. Current market-oriented researches mainly fall into auction-based mechanism [9-13] and game-based mechanism [14-16][19][20] two categories.

Popcorn [11] is a Web-based computing system for parallel application scheduling, in which jobs are auctioned in different mechanisms including Vickrey and first-price and k-price double auction. Garg [12] designed a meta-scheduler which used Continuous Double Auction (CDA) to map user applications onto resources. Within this CDA-based scheduler, job valuation was considered as a bid while resource valuation was considered as an ask. Zhao [13] proposed the BarSAA (barging based self-adaptive auction) grid task-bundle allocation algorithm. Our work has some similarities with BarSAA in the sense that we both handle allocation of bundle tasks, and view task as a commodity sold in the commercial market. However, BarSAA fixed market clearing price via searching tentative price vector, which could not always guarantee algorithm convergence.

In grid computing, game theory is extremely helpful in modeling behaviors of benefit-driven agents. The typical approaches define the objective utility functions and converge at the system equilibrium state on the basis of benefit maximization. Li [14] proposed a resource allocation strategy which used sequential game method to predict resource load for time optimization. Kwok [15] investigated the impact of selfish behaviors of individual machine by taking account of non-cooperativeness of machines. Considering that the gird scheduler has full control of processor's schedules, Rzadca [16] tried to produce solutions which are fair to all the participants by modeling scheduling as an extension of the well-known Prisoner's Dilemma (PD) game. However, due to the extreme complexity of the game process, the time complexity of his algorithm is usually very high. Even if there are only two organizations, the number of Pareto-optimal grid schedules is still very large.

Khan [17] classified game-based co-allocation models into three types: cooperative, semi-cooperative and non-cooperative. By extensive simulations, they concluded that the cooperative method leads to better task rejection, utilization and turnaround time as well as comparable load fairness and makespan. Inspired by this, we use cooperative game to model resource behaviors. With reference to the above mentioned related work, our models and formulations are novel in that we consider the cooperativeness of subordinate resources and the inherent non-cooperativeness between providers and users. Our work is also the first of its kind in providing a proven global benefit maximization allocation.

# 3 Problem Formulation

## 3.1 System Model

Kwok argued that a hierarchical structure is more feasible for the large-scale open grid computing system [15]. We follow the approach and our two-tier system model is shown in Fig. 1.

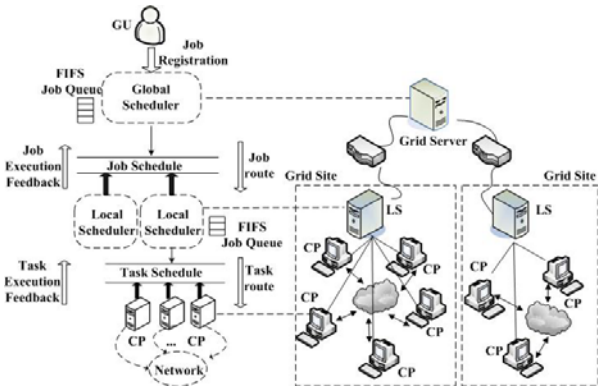


**Fig. 1.** Structure of hierarchical grid system

The up-tier consists of grid users, grid sites and a global scheduler. The global scheduler is responsible for handling users' applications submitted to the grid. Applications are inserted into a global FIFS (first in first service) queue. Global scheduler could use our former inter-site gaming strategy [20] to dispatch applications onto grid sites.

The down-tier contains computing peers (CPs) participating in making up of grid site and a local scheduler (LS). In each grid site there is a LS to allocate local resources, such as Portable Batch Scheduler (PBS) and Load Sharing Facility (LSF) used in GRAM. LS is critical in our model which behaves in dual mode. As a site agent, LS contends for its own site profits. As a local resource manager, LS breaks applications into tasks and generates the current available list of computing peers, further specifies the intra-site mapping rule. CPs contend tasks for their own utility.

In this paper we focus on the down-tier in which core problem is intra-site scheduling. Our grid system is limited in a single site domain.

Based on our former inter-site scheduling strategies [18-20], within a site we assume that tasks are colleted in mode of task-bundle [13], denoting a large set of identical independent tasks with no communications involved. The BoT size is much larger than Medium class [4] such as of size 1000 and over. We assume that after some time there are m tasks arrived. These tasks have identical attributes including computational size ($S_{cp}$) and communication size ($S_{cm}$). Current available CP list contains n computing peers. Since our system model is built upon a virtual grid market, the following definitions are critical and will be throughout this paper.

**Definition 1.** Task Retail Price.
Let $v_i (i = 1,2,..., n)$ denote the monetary value of one task setup by every CP. Every task should pay $v_i$ for successful execution on $CP_i$.

**Definition 2.** Task Market Price.
Let p* denote the monetary value of one task setup by LS. Every CP should pay LS p* for getting a task.

**Definition 3.** Task assignment amount vector $\boldsymbol{A} = (A_1,...,A_i,...,A_n)$.

Let $A_i (i = 1,2,...,n)$ denote the task assignment amount which will maximize $CP_i$'s utility at current task market price. Obviously, $A_i$ satisfies $\forall i, A_i \in N$ and $A_i \le m$.

Grid system performs as follow: LS sells tasks to available CPs at a uniform price called task market price. Every task will pay CP's retail price for the successful execution. Different CP will demand its own task retail price according to its resource performance. All CPs' profits constitute grid system's benefit and CPs cooperate with LS aiming at maximizing global resource benefit. Based on this critical property, we name our model Intra-Site Cooperative-game of Task-bundle (ISCT) in the following text. Note that tasks and CPs are inherently non-cooperative. Tasks hope to minimize their cost, which would inevitably lower down the profits of CPs.

The utility function of LS is defined as

$$U^L = p^* \cdot \sum_{i=1}^{n} A_i, \quad \sum_{i=1}^{n} A_i = m \tag{1}$$

The utility function of $CP_i$ is defined as

$$U_i^C = A_i \cdot v_i - A_i \cdot p^* \tag{2}$$

The global benefit is the sum of all the CPs' profits and the LS's revenue, which equals to the sum cost of n tasks:

$$U^G = U^L + \sum_{i=1}^{n} U_i^C = \sum_{i=1}^{n} (A_i \cdot v_i) \tag{3}$$

As the task provider, LS should determine reasonable p* and $\boldsymbol{A}$. The solution of cooperative model can be viewed as a distribution of benefit and will be characterized by the 2-tuple $< p^*, \boldsymbol{A} >$. Indeed, solution space of $\boldsymbol{A}$ is the set of scheduling schemes.

### 3.2 Pricing

We use $t_i$ to represent estimated completion time (ECT) of $CP_i$ finishing one task. Let $Cm_i$ denote $CP_i$'s communication capacity and $Cp_i$ denote $CP_i$'s computation capacity. ECT contains setup time $t^{setup}$, transmission time $t^{trans}$ and execution time $t^{exec}$ three parts in which $t^{setup}$ is fixed. $t_i$ is calculated as follow:

$$t_i = t^{setup} + t^{trans} + t^{exec} = t^{setup} + \frac{S_{cm}}{Cm_i} + \frac{S_{cp}}{Cp_i} \tag{4}$$

Generally users would like pay more money for faster service. Here we map the ECT to task retail price by $v_i = 1/t_i$. We use a discount rate considering different deal price at different quantities. Let $k (k \in N, k \le m)$ represent different quantities of tasks and $v_i(k)$ represent mean deal price for k tasks on $CP_i$, then $v_i(k)$ satisfies

$$v(0) = 0, \quad \forall i, k_1 < k_2 \Rightarrow v_i(k_1) > v_i(k_2) \tag{5}$$

Let $V_i^k$ be the sum deal price for k tasks on $CP_i$, $\alpha (0 < \alpha < 1)$ be the discount rate and $q = 1 - \alpha \ (0 < q < 1)$, we can get

$$V_i^1 = v_i(1) = v_i = \frac{1}{t_i},\ V_i^k = \frac{1}{t_i} \cdot \sum_{j=1}^{k} (1-\alpha)^{j-1} = \frac{1}{t_i} \cdot \frac{1-q^k}{\alpha} \quad (6)$$

We assume that different CPs have the identical discount rate. $v_i$ can be regarded as the base price from which we can evaluate performance of $CP_i$. As $V_i^1$ has the same value with $v_i$, $V_i^1$ is denoted as $V_i^b$. We can get

$$V_i^k = V_i^b \cdot \frac{(1-q^k)}{\alpha},\ v_i(k) = \frac{V_i^b}{k} \cdot \frac{(1-q^k)}{\alpha} \quad (7)$$

# 4 Design and Analysis of ISGBMAT Algorithm

For the convenience of representation, some definitions are given as follows.

**Definition 4.** $CPs^+(p^*)$, $CPs^-(p^*)$ and $CPs^0(p^*)$.
At $p^*$, $CPs^+(p^*) = \{CP_i \mid U_i^C > 0, i \in [1,...,n]\}$ is the set of CPs with positive profits; $CPs^-(p^*) = \{CP_i \mid U_i^C < 0, i \in [1,...,n]\}$ and $CPs^0(p^*) = \{CP_i \mid U_i^C = 0, i \in [1,...,n]\}$ respectively are the set of CPs with negative and zero profits.

**Definition 5.** Balanced State.
At $< p^*, A_i >$, $CP_i$ will be in a balanced state if $CP_i \in CPs^0(p^*)$; at $< p^*, \boldsymbol{A} >$, the ISCT system will be in a balanced state if $\forall CP_i$ satisfies $CP_i \in CPs^0(p^*)$.

## 4.1 Calculation Method for ISCT

For each $CP_i$ in the available CP list, we assume $A_i$ is positive. So $\forall i,\ 1 \le i \le n$ satisfies $A_i \in N^+$.

**Theorem 1.** If ISCT system is in a balanced state at $< p^*, (A_1,...,A_n) >$, adjusting $< p^*, (A_1,...,A_n) >$ will not affect the global benefit $U^G$.
**Proof** Let $(U_1^C, U_2^C,..., U_n^C)$ represent the profits of CPs at $< p^*, (A_1,...,A_n) >$ and $(U_1^{C'}, U_2^{C'},..., U_n^{C'})$ represent the profits of CPs at $< p^* + \Delta p^*, (A_1 + \Delta A_1, \ldots, A_n + \Delta A_n) >$.

At $< p^*, (A_1,...,A_n) >$, according to definition 4 and definition 5, it can be known that $\forall CP_i$ satisfies $U_i^C = A_i \cdot (v_i - p^*) = 0$, then

$$U^G = U^L + \sum_{i=1}^{n} U_i^C = U^L + \sum_{i=1}^{n} 0 = U^L.$$

Assumed after adjusting, ISCT system is being at $< p^{*}+\Delta p^{*}, (A_1 + \Delta A_1, \ldots, A_n + \Delta A_n) >$. According to Formula (2), we can get $U_i^{C'} = A_i \cdot (v_i - p^{*}) - \Delta p^{*} \cdot (A_i + \Delta A_i) + \Delta A_i \cdot (v_i - p^{*}) = -\Delta p^{*} \cdot (A_i + \Delta A_i)$

Provided that the total task amount m is fixed, then $\sum_{i=1}^{n}(A_i + \Delta A_i) = \sum_{i=1}^{n} A_i$. It can be known that at $< p^{*}+\Delta p^{*}, (A_1 + \Delta A_1, \ldots, A_n + \Delta A_n) >$, the global benefit $U^{G'}$ is

$$U^{G'} = U^{L'} + \sum_{i=1}^{n} U_i^{C'} = (p^{*}+\Delta p^{*}) \cdot (\sum_{i=1}^{n}(A_i + \Delta A_i)) - \Delta p^{*} \cdot \sum_{i=1}^{n}(A_i + \Delta A_i)$$
$$= p^{*} \cdot \sum_{i=1}^{n}(A_i + \Delta A_i) = p^{*} \cdot \sum_{i=1}^{n} A_i = U^{L} = U^{G}$$

**Theorem 2.** If ISCT system is in an unbalanced state at $< p^{*}, (A_1, ..., A_n) >$, the global benefit $U^G$ can always be increased by adjusting $< p^{*}, (A_1, ..., A_n) >$.

**Proof** At $< p^{*}, (A_1, ..., A_n) >$, if system is in an unbalanced state, according to Formula (3) we can get $U^G = \sum_{i=1}^{n}(A_i \cdot v_i)$ and $U^G$ is independent of $p^{*}$. If $p^{*}$ is adjusted to $p_0^{*}$ and $p_0^{*} = v_k, 1 \le k \le n$, the system will satisfy the following conclusions at $< p_0^{*}, (A_1, ..., A_n) >$.

(a)The system is still in unbalanced state or else it will contradict with the precondition of the theorem.

(b) According to formula (3), the global benefit remains the same.

(c) Provided $p_0^{*} = v_k$, the $CP_k$ is in a balance state, that is $CP_k \in CPs^0(p_0^{*})$.

According to (a), it can be known that $\exists CP_i$ satisfies $CP_i \in CPs^{+}(p_0^{*})$ or $CP_i \in CPs^{-}(p_0^{*})$, where $i \neq k$.

If $CP_i \in CPs^{+}(p_0^{*})$, task assignment amount of $CP_k$ and $CP_i$ can be adjusted to $A_k^{'} = A_k - \Delta\delta$ and $A_i^{'} = A_i + \Delta\delta$ respectively, where $\Delta\delta > 0$.

For $CP_k \in CPs^0(p_0^{*})$, adjustment of $CP_k$ will not affect the global benefit. For $CP_i \in CPs^{+}(p_1^{*})$, increasing $CP_i$'s task assignment amount will increase the global benefit. Thus the global benefit at $< p_0^{*}, (A_1, ...A_k^{'}, ...A_i^{'}, ..., A_n) >$ could be larger than that at $< p^{*}, (A_1, ..., A_n) >$.

If $CP_i \in CPs^{-}(p_0^{*})$, the analysis and conclusion are similar to the above, and the only difference is $\Delta\delta < 0$.

**Theorem 3.** ISCT system will obtain maximal global benefit if and only if it is in a balanced state.

Combining Theorem 1 and Theorem 2, it is easy to get Theorem 3. So, the system can improve its global benefit by repeatedly adjusting $< p^{*}, (A_1, ..., A_n) >$ to make the system in or near to a balanced state.

Given the system is in an unbalanced state at $< p^*, (A_1, ... A_n) >$ and would be adjusted to $< p_0^*, (A_1', ..., A_n') >$. If $p_0^*$ satisfies $\min \sum_{i=1}^{n} (v_i - p_0^*)^2$, system is the nearest to a balanced state. It is clear that the solution is

$$p_0^* = \frac{1}{n} \cdot \sum_{i=1}^{n} v_i \tag{8}$$

### 4.2 Solution of Task Assignment Amount Vector

LS will assign tasks to the CPs according to their task retail price $v_i$. Generally, tasks tend to be executed on the CP with faster resource for better time metric or with lower retail price to meet cost metric. This may arise two extremities: the most expensive or the cheapest CP will occupy too much tasks. As mentioned in section III our pricing scheme use discount, and mean deal price $v_i(k)$ is a decreasing function of task quantity k. Similar with the case in a commodity market, a CP can get more tasks assigned by lowering its mean deal price. However, the mean deal price can not be too low or CP will benefit zero or negative considering that CP still should pay LS p*. This constraint avoids aforementioned extremities. According to Formula (7), as mean deal price of $CP_i$ being a function of its task assignment amount $A_i$, utility function defined in Formula (2) can be rewritten as $U_i^C = A_i \cdot v_i(A_i) - p^* \cdot A_i$. Since $v_i(A_i) = \frac{V_i^b}{A_i} \cdot \frac{(1 - q^{A_i})}{\alpha}$, we have $U_i^C = V_i^b \cdot \frac{(1 - q^{A_i})}{\alpha} - p^* \cdot A_i$. To solve $\frac{d(U_i^C)}{d(A_i)} = 0$ we can get the equation $\frac{V_i^b}{\alpha} \cdot q^{Ai} \cdot \ln q + p^* = 0$, so the solution of $A_i$ is

$$A_i = \log_q \left( \frac{-\alpha \cdot p^*}{\ln q \cdot V_i^b} \right) \tag{9}$$

In Formula (9) $A_i$ may be negative, which means it lose money in business for $CP_i$ to execute tasks. According to our pricing scheme, utility of $CP_i$ executing kth task can be defined as

$$U_i^C(k) = V_i^b \cdot q^{k-1} - p^*, \ 0<q<1. \tag{10}$$

Provided $U_i^C(k)$ is a decreasing function of k, it is easy to know that $CP_i$'s utility $U_i^C$ is maximal as long as the lowest utility for single task $U_i^C(k)$ is not negative, which is represented as follow:

$$V_i^b \cdot q^{A_i - 1} - p^* \geq 0, \ 0<q<1 \tag{11}$$

So we can get $A_i \leq 1 + \log_q (p^* / V_i^b)$ and task assignment amount can be modified as

$$A_i = \left\lfloor 1 + \log_q (p^* / V_i^b) \right\rfloor \tag{12-1}$$

In some extreme cases, some CPs with super capability may get most of tasks. In order to alleviate this monopoly phenomenon, here we introduce $N_{max}$ to limit the

maximum quantity of tasks that a CP of super capability can get. Hence Formula (12-1) can be modified as follow:

$$A_i = \begin{cases} A_i, & \text{if } A_i < N_{max} \\ N_{max}, & \text{if } A_i \geq N_{max} \end{cases} \quad (12\text{-}2)$$

**INPUT**: $m, n, S_{cp}, S_{cm}, \alpha, N_{max}$ .

**OUTPUT**: p*, $\boldsymbol{A}$ .

```
1.   Begin
2.     Initialization flagCon=1, A = LF = F = 0 , UB = N_max ;
3.     for i=1 to n do
4.          calculate V_i^b ;  v_i = newV_i^b = V_i^b ;
5.     end for
6.     calculate p*;
7.     while ∑_{i=1}^n A_i + ∑_{i=1}^n F_i ≠ m  do
8.           for i=1 to n do
9.              LF_i = F_i ; calculate F_i ;
10.          end for
11.          flagCon=any(LF~=F);
12.          if ~flagCon
13.             for i=1 to n do
14.                A_i = A_i + F_i ; newV_i^b = V_i^b · q^{A_i} ;
15.                UB_i = UB_i − F_i ; F_i = LF_i = 0 ;
16.             end for
17.          end if
18.          for i=1 to n do
19.             v_i = newV_i^b * (1 − q^{F_i}) / α / F_i ;
20.          end for
21.          update p*;
22.    end while
23.    for i=1 to n do
24.      Calculate CP_i's income from tasks, the payment from
25.      CP_i's to LS and CP_i's profits
26.    end for
27.  output p*, A
28.end.
```

**Fig. 2.** Intra-Site Global Benefit Maximization Allocation for Task-bundle (ISGBMAT)

### 4.3 ISGBMAT Algorithm

The ISGBMAT algorithm is described in Fig.2.

Based on our previous analysis result, ISGBMAT uses an iteration adjustment in the sequence of retail price, market price and assignment amount of tasks. We save the latest task assignment amount in $F_i$ and the task assignment amount of last round in $LF_i$. In step4, for each $CP_i$, $V_i^b$ is initialized according to Formula (6). In step 6, p* is calculated with Formula (8).

A flag 'flagCon' is used to partition the iteration into inner and outer. The transition of flagCon from 1 to 0 represents that a round assignment is over. In a round, task assignment amount of each CP adjusts according to Formula (12) and $F_i$ should satisfy $0 \le LF_i \le F_i \le UB_i$ (Step 9). Accordingly, the mean retail price of CP changes by Formula (7) (Step 19) and the latest p* is calculated with Formula (8) (Step 21). After getting the result of $F_i$, it need compare $F_i$ with $LF_i$ to update flagCon (Step 11). The equality denotes that the amount can not be updated anymore and the inner iteration ends. Then it should set flagCon zero and update retail price of CPs (Step 14). Here $newV_i^b$ is used to memorize $V_i^b$ of every round.

Here UB is used to save the realtime remaining maximum assignment amount, which is uniformly initialized with $N_{max}$ (Step 2) and will be updated according to the latest assignment mount (Step 15). Set task assignment amount to be UB if it figures larger than UB (Step 9).

The time complexity of the algorithm is $O(n \cdot S)$ where n is the number of computing peers and S is the iteration times of the outside loop (Step7-Step22). It is easy to know that the maximum value of S is equal to m. So the complexity of ISGBMAT is lower than $O(n \cdot m)$.

## 5 Performance Evaluation

### 5.1 Experimental Setup

The performance of ISGBMAT proposed in this paper is evaluated based on a prevalent grid simulator GridSim 5.0 [21]. A multi cluster computational grid model is constructed, consisting of 12 clusters which referenced to the related data in the American large-scale grid application TeraGrid [22]. Fig. 3 illustrates system snapshots of available computing and network resources that each CP is willing to contribute. The computing speed is represented in MIPS and network bandwidth is represented in bps. In Fig. 3, system's $H_D = 11.0468$ (see Definition 6).

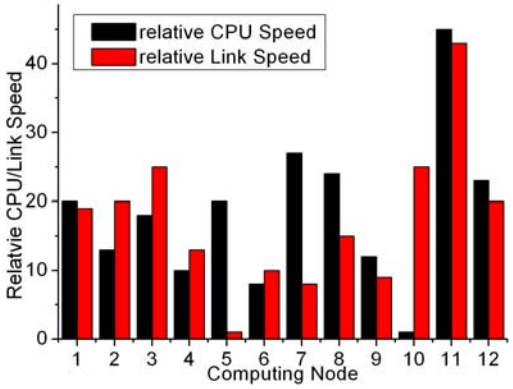


**Fig. 3.** System snapshot of CPU/Link

Our experiment uses real workload traces gathered from existing supercomputers and collected in the Parallel Workload Archive [23]. The basic workload consists of 1000 tasks. Based on Formula (4), we classify tasks to three categories according to compute communication ratio, which are (1) computation-intensive: $S_{cp}/S_{cm} = 1000:1$ ; (2) neutrality (Cp size and Cm size are equal ): $S_{cp}/S_{cm} = 500:500$ ; (3) communication-intensive: $S_{cp}/S_{cm} = 1:1000$ . Setup time $t^{setup}$ is fixed to 5ms. The following two metrics are important in our experiment.

**Definition 6.** Heterogeneous degree $H_D$ .

$H_D \geq 0$ , which indicates the resource performance difference between CPs, is defined as:

$$H_D = \frac{1}{2} \cdot \left( \sqrt{\frac{1}{n} \cdot \sum_{i=1}^{n} (Cp_i - \overline{Cp})^2} + \sqrt{\frac{1}{n} \cdot \sum_{i=1}^{n} (Cm_i - \overline{Cm})^2} \right) \quad (13)$$

Where $\overline{Cp}$ and $\overline{Cm}$ represent the mean computing and network ability respectively.

**Definition 7.** Workload Fairness FI .

$FI \in (0,1]$ indicates workload balance situation concerning ECT in every CP. We quantity the workload fairness by using the Jain's fairness index [24]:

$$FI = \frac{(\sum_{i=1}^{n} T_i)^2}{n \cdot \sum_{i=1}^{n} T_i^2} \quad (14)$$

$T_i$ is the ECT for $CP_i$ to finish all the assigned tasks. The strategy is perfectly fair if the value of FI is 1.

In the simulation experiments, ISGBMAT is compared with three other economy models: Commodity Market (CM) Model (Flat) [6], Proportional Share [6] and Double Auction [12], in terms of global benefit, completion time and total time. Completion time is the longest completion time on single CP while total time is the sum of completion time on all CPs. The benefit and payments in ISGBMAT are expressed

with virtual grid dollars (G$). For each scenario, we did the experiments 50 times independently and took the average value for different metrics.

### 5.2 Experimental Results and Analysis

*1) Performance comparison*

**Table 1.** Scheduling Result Comparison with three other algorithms

| Algorithm | Performance metrics | | |
|---|---|---|---|
| | ***Global benefit (G$)*** | ***Completion time (h:m:s)*** | ***Total time (h:m:s)*** |
| CM(flat) | 60424 | 2:00:37 | 18:21:03 |
| Double Auction | 61976 | 1:49:51 | 17:43:27 |
| Proportional Share | 60790 | 1:23:46 | 16:56:31 |
| ISGBMAT | 62765 | 1:37:50 | 16:48:25 |

As shown in Table 1, we observe that our algorithm outperforms others in terms of global benefit and can obtain fairly good time metrics. CM(flat) has performed the worst in every case. The reason for this behavior is that CM(flat) with static price could not adapt to dynamic market. Proportional Share fairly allocates tasks among various CPs on the basis of CP prices, which leads to good time metric but poor global benefit. In Double Auction, it is prone to allocate tasks to expensive (high performance) CPs which results in high global benefit at the expenses of time. In ISGBMAT, task market price adjusts iteratively until the equilibrium is reached. The Global benefit is increased by adjusting task assignment amount of CPs and retail price of CP is updated accordingly. We have proved global benefit maximization property of ISCT. Contrasting with conventional economy model in commodity market, we introduce dynamic task market price as leverage to the proposed ISGBMAT, which leads to workload balance and reduction of the completion time.

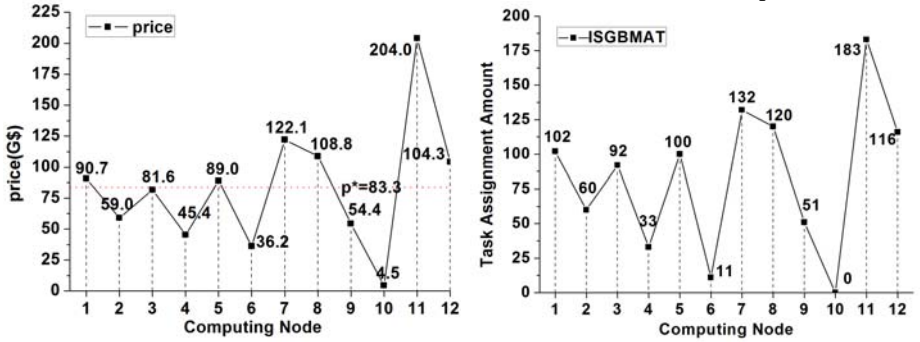


**Fig. 4.** (a)Retail price of CPs (b) Allocation result of ISGBMAT

*2) Self-selection property of ISGBMAT*

ISGBMAT preserves self-selection property which is inherited from market model. For more deep explanation, details of each CP's retail price and allocation result are shown in Fig.4(a) and Fig.4(b) respectively. In this set of experiments, CPs are configured just as they are in Fig.3 and tasks attribute is set to be computation-intensive. As shown in Fig.4(b), there is no task assigned to CP10 which means CP10 has been knocked out. We observe that CP10's relative CPU Speed is so low (Fig.3) that it will be priced (in Fig.4(a), PriceCP10=4.5) far behind market price (p*=83.3) when executing computation-intensive tasks.

*3）Interplay of* $N_{max}$, FI *and* $H_D$

In the last set of experiments we demonstrate how $N_{max}$, FI and $H_D$ interplay. In Fig. 5, for simplicity, we define mean assignment amount $A_{mean} = \lfloor m/n \rfloor$. Here $N_{max}$ will set to be $\lambda$ times of $A_{mean}$. There are three curves which represent different groups of CPs with resource capability, referencing to different $H_D$. Results show that higher degree of heterogeneity will decrease FI of allocation, which is coincident with our common cognition that 'diversity calls forth unfairness'.

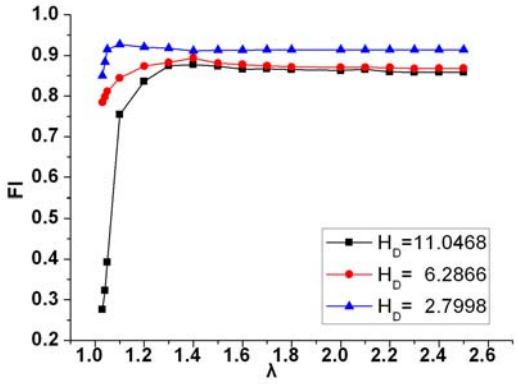


**Fig. 5.** Interplay of $N_{max}$, FI and $H_{Degree}$

We observe that these three curves have similar sketch. In the condition $\lambda = 1$, workload is enforced to be distributed balancedly, which will lead to poor FI. As $\lambda$ going up, $N_{max}$ becomes bigger and bigger, and the system will have better FI accordingly. FI will converge with $\lambda$ getting to a certain maximum constraint. Then task assignment amount has no limit so that $N_{max}$ is useless. This analysis result gives advice for more effective solution with a feasible $N_{max}$.

## 6 Conclusion

In this paper we propose an efficient Intra-Site Global Benefit Maximization Allocation for Task-bundle (ISGBMAT) based on our model called Intra-Site

Cooperative-game of Task-bundle (ISCT) in hierarchical computational grids. There are two major contributions of our design. First of all, our solution of ISCT will realize maximal global benefit of system which has been validated by mathematical analysis. In addition, based on ISCT, ISGBMAT is designed by formulating resource allocation as an iteration process involving retail price, market price and allocation amount of tasks. It provides a novel economy-based scheme for resource allocation. Extensive simulations verify the efficiency of ISGBMAT. In our ongoing work, we will explore different discount rate $\alpha$ setting to different CPs for more balanced workload and further adapt our strategy to other workload mode and pricing schemes.

**Acknowledgements.** This paper is supported by the NSFC project (Grant No. 60970038).

## References

1. Foster, I., Kesselman, C.: The Grid: Blueprint for a New Computing Infrastructure. Morgan Kauffman, San Francisco (2004)
2. Chun, B. N., Culler, D. E.: User-centric Performance Analysis of Market-based Cluster Batch Schedulers. In: 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid, pp. 30-38. IEEE Computer Society, Berlin (2002)
3. Yeo, C. S., Buyya, R.: Service Level Agreement based Allocation of Cluster Resources: Handling Penalty to Enhance Utility. In: 7th IEEE International Conference on Cluster Computing, pp. 1--10. IEEE Press, Boston (2005)
4. Losup, A., Sonmez, O., Anoep, S., Epema, D.: The Performance of Bags-of-tasks in Large-scale Distributed Systems. In: 17th International Symposium on High Performance Distributed Computing, pp. 97--108. ACM, Boston (2008)
5. Kim, J. K., Shivle, S., Siegel, H. J., Maciejewski, A.A., Braun, T.D., Schneider, M. et al: Dynamic Mapping in a Heterogeneous Environment with Tasks Having Priorities and Multiple Deadlines. In: 17th International Symposium on Parallel and Distributed Processing, pp. 98--112. IEEE Computer Society, Nice (2003)
6. Buyya, R.: Economic-based Distributed Resource Management and Scheduling for Grid Computing. Australia: Monash University (2002)
7. GRACCE: Grid Application Coordination, Collaboration and Execution. http://www.cs.uh.edu/~gracce
8. Buyya, R., Abramson, D., Giddy, J.: Nimrod/G: An Architecture for a Resource Management and Scheduling System in a Global Computational Grid. In: Proceedings of International Conference/Exhibition on High Performance Computing in Asia-Pacific Region, pp. 283--289. Beijing (2000)
9. Lai, K., Rasmusson, L., Adar, E., Zhang, L., Huberman, A.: Tycoon: An Implementation of a Distributed, Market-based Resource Allocation System. Multiagent and Grid System. vol. 1(3), pp. 169--182. (2005)
10. AuYoung, A., Chun, B., Snoeren, A., Vahdat, A.: Resource Allocation in Federated Distributed Computing Infrastructures. In: 1st Workshop on Operating System and Architectural Support for the On-Demand IT Infrastructure (2004)
11. Regev, O., Nisan, N.: The POPCORN Market Online Markets for Computational Resources. Decision Support System, vol. 28(1-2), pp. 177-189. (2000)

12. Garg, S. K., Venugopal, S., Buyya, R.: A Meta-scheduler with Auction based Resource Allocation for Global Grids. In: 14th IEEE International Conference on Parallel and Distributed Systems, pp.187--194. IEEE, Melbourne (2008)
13. Zhao, H., Li, X. L.: Efficient Grid Task-Bundle Allocation using Bargaining Based Self-adaptive Auction. In: 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, pp. 4--11. IEEE Computer Society, Shanghai (2009)
14. Li, Z. J., Cheng, C. T., Huang, F. X., Li, X.: A Sequential Game-based Resource Allocation Strategy in Grid Environment. Journal of Software. vol. 17(11), pp. 2373--2383. (2000)
15. Kwok, Y. K., Song, S., Hwang, K.: Selfish Grid Computing: Game Theoretic Modeling and NAS Performance Results. In: 5th IEEE/ACM International Symposium on Cluster Computing and the Grid, vol. 2, pp. 1143--1150. IEEE Computer Society, Cardiff (2005)
16. Rzadca, K., Trystram, D., Wierzbicki, A.: Fair Game-theoretic Resource Management in Dedicated Grids. In: 7th IEEE/ACM International Symposium on Cluster Computing and the Grid, pp.343-350. IEEE Computer Society, Rio de Janeiro (2007)
17. Khan, S. U., Ahmad, I.: Non-cooperative, Semi-cooperative, and Cooperative Games-based Grid Resource Allocation. In: 20th IEEE International Symposium on Parallel and Distributed Processing, pp. 10. IEEE Press, Rhodes Island (2006)
18. Xiao, P., Hu, Z. G.: A novel QoS-based Co-allocation Model in Computational Grid. In: IEEE Global Communications Conference 2008, pp. 1562--1566. IEEE Press, New Orleans (2008)
19. Hu, Z. J., Hu, Z. G., Ding, C. S.: Game-theory-based Robust-enhanced Model for Resource Allocation in Computational Grids. Systems Engineering-Theory&Practice. vol. 29(8), pp. 102--110 (2009)
20. Zheng, M. G., Hu, Z. G., Zhang, K.: Performance-efficiency Balanced Optimization based on Sequential Game in Grid Environments. Journal of South China University of Technology. vol. 38(1), pp. 92--96+107 (2010)
21. GridSim, http://www.cloudbus.org/gridsim
22. TeraGrid, http://www.teragrid.org
23. Parallel Workloads Archive, http://www.cs.huji.ac.il/labs/parallel/workload
24. Jain, R., Chiu, D., Hawe, W.: A quantitative measure of fairness and discrimination for resource allocation in shared computer systems. CoRR (1998)