Research on the Resource Sharing Among Hierarchically Distributed Virtual Enterprises Based on Grid System

Hongbo Wang*

School of Traffic and Logistics, Central South University of Forestry and Technology, Changsha, Hunan, 410004 China

Abstract: In the virtual manufacturing process based on grid technology, the user and resource provider share resource and service connected through different virtual enterprises. In order to realize the target, virtual enterprise manages the enterprises at its lower level, and then the hierarchically distributed environment is formed. The resource provider and virtual enterprise should achieve the agreement on the strategy of resource sharing, such as the quantity, type and proportion of resource sharing. Therefore, the users in the virtual enterprises at lower level could get access to the resource of the upper level to achieve the common aim. The problem of allocating resources among hierarchically distributed virtual enterprises is the focus, and at the same time, suitable proportion of sharing resource is also discussed. In addition, under specific proportion condition, a new resource sharing scheme is put forward. Compared with some other resource sharing methods, the effectiveness of sharing resource is verified by using simulation software. The most effective and suitable resource sharing method among hierarchically distributed enterprises is explored.

Keywords: Grid system, Hierarchical distribution, Resource allocation and sharing, Virtual enterprise.

1. INTRODUCTION

With the increasing number of virtual enterprises in the grid, the effective management oriented to virtual enterprises is greatly needed [1]. According to the organization style of virtual enterprises, there are four models which are monadic, hierarchy, alliance and hybrid. Hierarchical virtual enterprise model is commonly adopted in enterprise operation. Under this model, virtual enterprises realize their aims through managing the sub-level virtual enterprises, especially the resource sharing problem. The resource provider establishes a series of service level agreements to make the resource portion allocate to different virtual enterprise specifics. Under this resource allocation strategy, effective and fair resource utilization is realized through priority ordering among virtual enterprises. The following three aspects are the focus of the paper. The first is to construct hierarchical grid environment oriented to virtual enterprises based on resource sharing strategy. The second provides a solution to solve the resource sharing problem under the environment of hierarchically distributed virtual enterprises [2]. The last compares and discusses the different resource allocating schemes and then a suggestion is put forward to help managers make their decision more appropriate and useful. In the research work, some important points are involved, for example, the related works of resource allocation, the system model of hierarchically distributed virtual enterprises, the algorithm oriented to the key point of resource sharing, the suggestion on establishing the framework of resource allocation, and the method and process of simulation.

In this article, the research focus is on the problem of resource allocation in the hierarchically distributed virtual enterprise grid. Through a series of service agreements provided by the resource owner, resource shares allocated to different virtual enterprises are determined. Under the resource allocation strategy, an inner sharing strategy is provided to make every virtual enterprise utilize the resource effectively and fairly by priority ordering. Emphasis is placed on the following three points: 1. Based on resource sharing strategy, the hierarchical virtual enterprise network is constructed. 2. A solution is put forward to solve the resource sharing problem under hierarchically distributed virtual enterprise environment. 3. Survey about the scheme of resource allocation is conducted and suggestions are provided.

In the following part, the related research work about resource allocation under grid environment is provided, the systematical model of hierarchically distributed virtual enterprise is determined, an algorithm to solve the problem of resource sharing is made clear, the suggestion on constructing the resource allocation framework is put forward including resource allocation scheme in resource allocation middleware, and at last, authentic results are provided through software simulation.

2. RELATED RESEARCH STATUS

The updated research on grid technology is focused on the resource scheduling and protection among virtual enterprises under a grid environment [3]. Many experts implant a new framework into the resource scheduling based on sharing strategy as part of SPHINX scheduling system. In the framework, the resource is utilized and managed effectively through adjusting the resource usage or priority of resource requirement [4]. Other experts have proposed suggestions on

*Address correspondence to this author at the School of Traffic and Logistics, Central South University of Forestry and Technology, Changsha, Hunan, China; 410004; E-mail: wanghongbocs@163.com

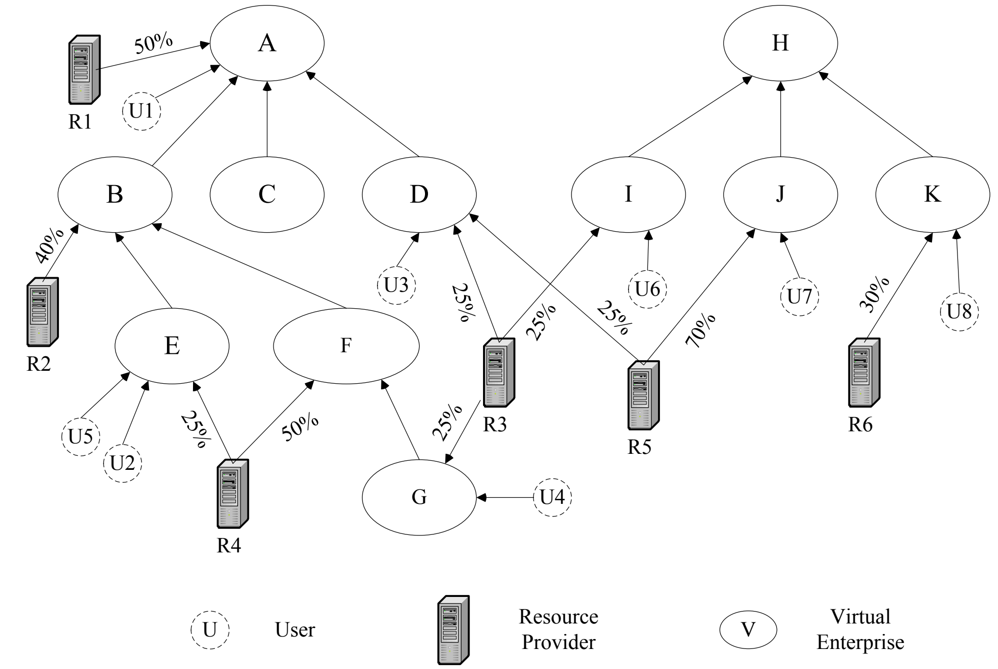

Fig. (1). Hierarchical virtual enterprise instance.

resource scheduling among virtual enterprises and have evaluated the aggregated resource utilization.

Another hot point in grid technology research is the resource allocation problem among many resource sources. Some experts are engaged in research on resource allocation in multicluster system through analysis and simulation aiming at the influence of the change of some parameters such as the structure of resource and batch size. In addition, there are some research works focused on the strategy of resource scheduling and selection of many important points such as the protection mechanism of grid resource allocation [5].

3. HIERARCHICAL VIRTUAL ENTERPRISE MODEL

3.1. Hierarchical Virtual Enterprise Environment

As many virtual enterprises are distributed in the grid environment, the sharing service among these virtual enterprises is absolutely necessary. One virtual enterprise can be divided into many sub-enterprises to manage the resource effectively. Therefore, virtual enterprise can be regarded as a group of users, resource providers and sub-virtual-enterprises.

The structure of virtual enterprises is shown in Fig. (**1**). Point A includes B, C and D. Point B also includes E and F. The structure can be viewed as an entity structure or logical structure.

There are user U1, resource R1 and three sub-virtual-enterprises B, C, and D included in A. At the same time, the resource owner can provide resources to many different virtual enterprises. For example, R3 provides resources to D, I and G.

3.2. System Model

1. Virtual enterprise model: there are user, resource owner and virtual enterprise in the grid system [6]. The user is a terminal entity, which submits mission to grid system and begins to operate by the resources in virtual enterprises. The resource owner allocates different resource shares to the user in virtual enterprise, which is a virtual organization integrating users, resource owners and sub-virtual-enterprises shares together to realize the target of the organization. Therefore, the grid can be defined as $G(U, R, V)$. U stands for the user group, R refers to the resource owner group and V means virtual enterprise group in the grid. The component of virtual enterprise A presented in Fig. (**1**) is shown in Table **1**.

The user group, resource provider group and sub-virtual enterprise group involved in virtual enterprise v can be represented as U_v, R_v, V_v. Virtual enterprise V (VE_v) can be defined as (U_v, R_v, V_v). Under the hierarchical virtual enterprise environment, the following terminologies are defined.

Parent virtual enterprise: if a virtual enterprise V is a child node of V', then V' is the parent virtual enterprise of V, represented as parent (v).

Ancestor virtual enterprise: for a specific virtual V, in the distribution tree of virtual enterprise, all the virtual enterprises from node V to the root node are called as an ancestor virtual enterprise of V and represented as ancs(v).

Descendant virtual enterprise: in the distribution tree of virtual enterprise, all the virtual enterprises from node V to the leaf virtual enterprise are the descendant virtual enterprises of V and represented as desc (v).

Root virtual enterprise: if a virtual enterprise V has no parent virtual enterprise, then it is a root virtual enterprise.

Leaf virtual enterprise: if a virtual enterprise V has no child virtual enterprise, then it is a leaf virtual enterprise.

Table 1. The components of virtual enterprises A in Fig. (1).

Virtual Enterprise (V)	User (U_v)	Resource (R_v)	Sub-virtual Enterprise (V_v)
A	{U1}	{R1(50%)}	{B,C,D}
B	Φ	{R2(40%)}	{ E, F }
C	Φ	Φ	Φ
D	{U3}	{R3(25%), R5(25%)}	Φ
E	{U2,U5}	{R4(25%)}	Φ
F	Φ	{R4(50%)}	{G}
G	{U4}	{R3(25%)}	Φ

Middle virtual enterprise: if a virtual enterprise is not a root virtual enterprise, nor leaf virtual enterprise, then it is a middle virtual enterprise.

2. Resource sharing strategy model: in the article, from resource sharing perspective, the operation strategy between the resource provider and its virtual enterprise is considered. The resource sharing strategy of resource provider r is reflected by the maximal resource share it provides to virtual enterprise V. The shared resource is represented as $share(r,v)$. The resource sharing strategy is a service level agreement established between the resource provider and the virtual enterprise. In Fig. (**1**), R3 provides 25% resource to D, 25% resource to G and 25% resource to I.

The share of resource is reflected by the percentage of all the resources of resource owner. For the scheduling policy of space sharing, the share reflects the number of processors distributed to virtual enterprise. For the scheduling policy of time sharing, it reflects the percentage of the total processing capacity distributed to virtual enterprise.

3. Working model: the job herein refers to a batch of mission, composed by many independent missions. In order to achieve the results, missions must be completed before the deadline.

4. THE RESOURCE SHARING IN HIERARCHICAL VIRTUAL ENTERPRISE

Virtual enterprise can allocate the operation work to child virtual enterprise to achieve its purpose. The resource provider of virtual enterprise allows the users in descendant virtual enterprise to use its resource, as long as the sharing strategy is not destroyed [7]. Because users have access to the resource in the virtual enterprises itself and the ancestor virtual enterprises, the allocation of resource in resource middleware has become a problem.

Suppose the resource capacity in upper virtual enterprise is better than that in the lower one and the users in lower layer are willing to utilize the resource in upper layer, this can reduce the QOS of the users of virtual enterprise in upper layer and cause invalid resource use. Therefore, the solution to the problem of resource allocation in hierarchical virtual enterprises is very important for protecting the fairness and effectiveness of resource allocation [8].

In order to solve the problem, the emphasis should be placed on the following two points, which are internal resource sharing strategy and resource allocation strategy. The internal resource sharing strategy shows the number of resources preferentially allocated to specific virtual enterprise users. So in resource allocating scheme, the first thing is to select priority resources. A heuristic algorithm is presented to solve the resource sharing problem with a new resource allocation framework and plan.

4.1. A Fair Resource Allocation Problem

A job is produced from the users and sent to virtual enterprise. Suppose user provides job with poisson process, with arrival rate as $\lambda^{(i)}$. The average processing rate of every resource provider is $u^{(j)}$. The following Fig. (**2**) describes the structure of virtual enterprise A in Fig. (**1**) with the above mentioned elements.

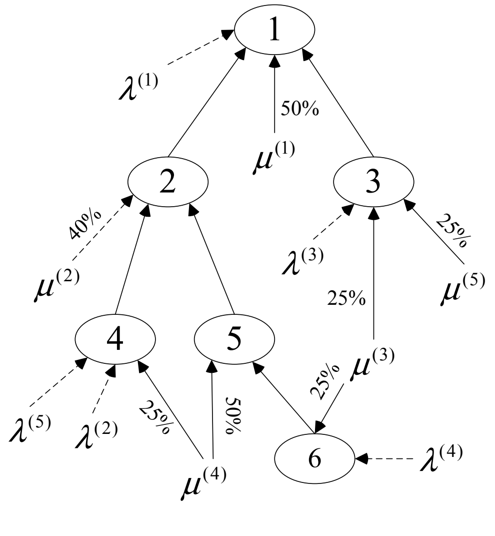

Fig. (2). The structure of virtual enterprise A in Fig. (**1**).

The job arrival rate of virtual enterprise can be calculated from the job arrival rate of users involved. If two independent processes appear with rates of λ^{1} and λ^{2}, respectively, then the total rate is $\lambda^{1} + \lambda^{2}$. Therefore, the job arrival rate of virtual enterprise V can be expressed as $\lambda_V = \sum_{\mathrm{u}\in U_V} \lambda^{(\mathrm{u})}$.

Similarly, the resource processing rate of virtual enterprise V can be expressed as $U_V = \sum_{\mathrm{r}\in R_{\mathrm{V}}} \mathrm{share}(r,v)\times U^{(r)}$.

The focus of resource sharing is the allocation of the processing rate of a virtual enterprise to its descendent virtual enterprise to reduce the total waiting time [9]. If the service delivered from virtual enterprise i to its descendent virtual enterprise j is defined as $P_{i,j}$, then the service processing rate of virtual enterprise j will increase by $P_{i,j}\cdot\mu_j$. The actual service processing rate of virtual enterprise i is $\mu_i^{\mathrm{a}} = \sum_{j\in\{i\}\cup ancs(i)} P_{j,i}\cdot\mu_j$.

Suppose the actual service processing rate and job arrival rate of virtual enterprise i are μ_i^{a} and λ_i, respectively, then for the users of i, the waiting time is $\frac{1}{\mu_i^{\mathrm{a}}-\lambda_i}$. The problem of sharing resource among virtual enterprises is to make $\sum_{i\in \mathrm{V}} \frac{1}{\mu_i^{\mathrm{a}}-\lambda_i}$ minimal. Similarly, the following constraint conditions must be satisfied, *i.e.* $\sum_{\mathrm{j}\in\{i\}\cup \mathrm{desc}(i)} P_{i,j}=1$ $P_{i,j}\geq 0$ μ_i^a £?λ_i $(i\in v)$.

4.2. Resource Sharing Decision

The problem of virtual enterprise resource sharing is a nonlinear optimization problem [10]. A heuristic algorithm is proposed to solve the problem in the article. The first step is the local optimization problem of the allocation of the resource of virtual enterprise i to its descendent virtual enterprise. Suppose the actual service rate of virtual enterprise j is μ_i^a. Actually, the problem is to decide $P_{i,i}$ and every $P_{i,j}$ $(j\in desc(i))$ to reduce the total waiting time $(\mu_{\mathrm{i}}^a=0)$. So to virtual enterprise i, the resource sharing problem represented as $SHARE_i$ is to make $\sum_{\mathrm{j}\in i\cup desc(i)} \frac{1}{P_{i,j}\cdot\mu_i+\mu_j^a-\lambda_j}$ minimal and satisfy the constraint condition $\sum_{\mathrm{j}\in i\cup desc(i)} P_{i,j}=1; P_{i,j}\geq 0$.

The program of the heuristic algorithm is shown in Table **2**. The output of the algorithm is $P_{i,j}$ of every virtual enterprise, the initial value of which is 0. The share allocated is represented as δ. Under the circumstance of specific δ, by the algorithm, the best virtual enterprise is found to reduce the waiting time further.

Table 2. Local optimization algorithm.

Local-Fair-share(i) Algorithm
$w_{i,j}(p)=\frac{1}{p\cdot u_i+\mu_j^a-\lambda_j}$: waiting time function
$p_{i,i}\leftarrow\frac{\lambda_i}{\mu_i}+\delta$;
For all j in desc(i) do $p_{i,j}\leftarrow 0$;
$T\leftarrow\{i\}\cup desc(i)$;
For all j in T do
$\Delta w_{i,j}\leftarrow w_{i,j}(p_{i,j}+\delta)-w_{i,j}(p_{i,j})$;
$p\leftarrow p_{i,i}$;
While p<1.0 do
$j\leftarrow\arg\max_{j\in T}\Delta w_{i,j}$
$p_{i,j}\leftarrow p_{i,j}+\delta$;
$\Delta w_{i,j}\leftarrow w_{i,j}(p_{i,j}+\delta)-w_{i,j}(p_{i,j})$;
$p\leftarrow p+\delta$;
Endwhile
For all j in T do
$\mu_j^a\leftarrow\mu_j^a+p_{i,j}\cdot\mu_i$;

The initial value of $P_{i,i}$ is defined as $\frac{\lambda_i}{\omega_i}+\delta$. By allocating sharing resource δ, the waiting time decrement of virtual enterprise j is represented as $\mathrm{i}\div\omega_{i,j}$. The algorithm is used to select the virtual enterprise with maximal $\mathrm{i}\div\omega_{i,j}$, giving the value of $P_{i,j}$ and updating μ_j^a.

The algorithm of resource sharing is shown in the following Table **3**. The algorithm Fair-share(v) decides the percentage of shared resource. At first, the sharing problem oriented to child virtual enterprise is determined. If a virtual enterprise is leaf node in the hierarchical structure, it allocates all resources to itself. Otherwise, through the algorithm Local-fair-share (), it allocates the resource to its descendant virtual enterprise [11].

Table 3. Resource sharing algorithm of virtual enterprise.

VE-share(G) Algorithm
G=(U,R,V)
For each VE v in V do
$\lambda_v = \Sigma_{u \in U_v} \lambda^{(u)}$;
$\mu_v = \Sigma_{r \in R_v} share(r,v) \times \mu^{(r)}$
Endfor
For each root VE $r \in V$ do
Fair-share(r);
Fair-share(v) algorithm
For all child VE *i* in V_v do
Fair-share(i);
If V_v==Φ then
$p_{v,v} \leftarrow 1.0$;
$\mu_v^a \leftarrow \mu_v$;
Else
Local-fair-share(v);
Endif

The call and return function of the Fair_share function (1) is shown in Fig. (**3**). The percentage of resource sharing in virtual enterprise is decided after the value of descendant virtual enterprise is determined. In Table **4**, in the first column, all the function calls are listed in post order traversal way. In the second column, the allocation value $P_{i,j}$ of every function return point is listed.

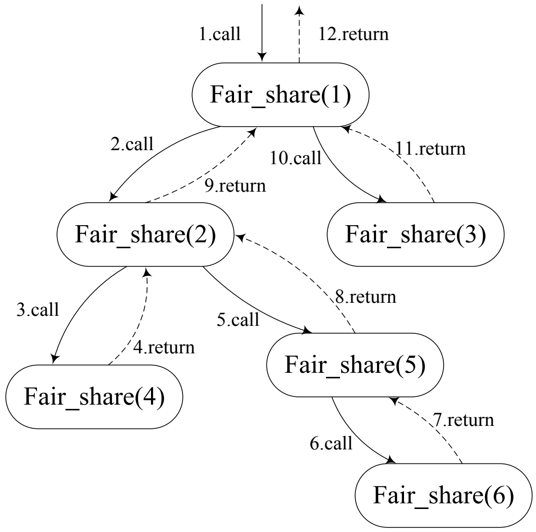

Fig. (3). The call and return function.

Table 4. The allocation value.

Return Sequence	Decision Value
4. Return Fair_share(4)	$P_{4,4}$
7. Return Fair_share(6)	$P_{6,6}$
8. Return Fair_share(5)	$P_{5,5}$ $P_{5,6}$
9. Return Fair_share(2)	$P_{2,2}$ $P_{2,4}$ $P_{2,5}$ $P_{2,6}$
11. Return Fair_share(3)	$P_{3,3}$
12. Return Fair_share(1)	$P_{1,1}$ $P_{1,2}$ $P_{1,3}$ $P_{1,4}$ $P_{1,5}$ $P_{1,6}$

5. RESOURCE ALLOCATIONS IN VIRTUAL ENTERPRISE

5.1. The Allocation Framework of Internal Resource in Virtual Enterprise

The allocation framework [12] of internal resource in virtual enterprise suggested in the article utilizes a collaborative middleware system [13] oriented to resource allocation of virtual enterprise. There is a resource middleware in every virtual enterprise, which masters all the strategies of virtual enterprise and allocates every mission delivered by virtual enterprise users. At the same time, it provides the strategy information for resource middleware of other virtual enterprises.

(1) Submit job. When a user submits a job, at first, the user should know the information about the job and the virtual enterprise and should submit the job in accordance with the strategy to the middleware. The validation of the job is verified by the strategy engineer of virtual enterprise in the middleware.

(2) Collect resource sharing information. In order to provide the best resource for user, the middleware collects resource sharing information in ancestor virtual enterprise from strategy tree of virtual enterprise. The user can get access to the resource in ancestor virtual enterprise, asperforming of work by the user is not only restricted to virtual enterprise itself but is for ancestor virtual enterprise as well.

(3) Allocate resource. Based on resource sharing information collected from other virtual enterprises, the resource middleware allocates the resource to a specific job. Depending on the resource provider's different capacities, the mission can be divided into many resource providers. Accepting it or not is dealt with the local scheduling of resource provider.

(4) Update sharing strategy. If a resource provider accepts a job from middleware, at first, it verifies the validation of the job. For example, virtual enterprise in use should be one of the corresponding virtual enterprises or its child virtual enterprise. Through the local scheduling algorithm, based on resource middleware, the resource provider updates the changing strategy.

Every resource provider has a local scheduling, which accepts submitted mission and allocates mission by scheduling strategy. Local scheduling only accepts the mission satisfying the requirements. The strategy controller communicates with resource middleware to acquire status information such as present system utilization rate and sharing strategy,.

In addition to taking responsibility of allocating job, the resource middleware takes charge of supervising the arrival rate of user's job. With the change of job arrival rate, the percentage of resource utilization also changes. Under this condition, the resource middleware informs the root virtual enterprise of updating resource sharing. Then the resource middleware of root virtual enterprise refers to the fair-share algorithm shown in Table **3** and transfers the changed strategy to the resource middleware of all the descendant virtual enterprises. When a resource provider joins or leaves a virtual enterprise, the sharing strategy changes and the strategy updating process occurs.

5.2. Resource Allocation Scheme

Resource middleware is in charge of many strategies and data structures corresponding to resource allocation. Some strategies involved in resource middleware of virtual enterprise V are as follows:

$share_v^{\max}$: The maximum of sharing resource provided by resource provider of virtual enterprise, which is the sum of all the sharing resources provided by the resource provider.

$share_v^{curr}$: The number of resources that are used by virtual enterprise at present.

$P_{v,v'}$: The utilization percentage of resource that the virtual enterprise v allocates to its descendant virtual enterprise v'.

$u_{v,v'}$: The present resource usage of descendant virtual enterprise v'.

The resource allocation algorithm of resource middleware is shown in Table **5**. At first, the resource provider of virtual enterprise is chosen, and then the resources are distributed among other ancestor virtual enterprises.

The process of resource allocation consists of two phases: the allocation based on internal strategy and the one based on external strategy. In each while cycle, the resource usage strategy abides by internal sharing strategy ($P_{i,j}$), and every virtual enterprise has a percentage of resource. Therefore, the resource middleware at first allocates the resource according to the internal sharing strategy. After allocation based on internal strategy is carried out, the remaining submissions allocate resource with external strategy, just as the second while cycle shown in Table **5**.

If all the sub-missions are allocated successfully, the algorithm reaches the end and the job is accepted. If there is not enough resource for operation, the allocated submissions before are cancelled and the job is returned. The users can submit the returned job again, or the resource middleware can manage the waiting sequence of those returned jobs [14].

Table 5. Resource allocation scheme.

VE-wide-resource-allocation(J,v) Algorithm

```
/* J=(P, {$l_1$, l2… lp}, d) refers to a job;
v stands for virtual enterprise*/
Task_index← 1;
i←v;
While $i\neq\Phi$ do
 While $u_{i,v}<p_{i,v}$ do
     For each $r\in R_i$ do
        (alloc,load)←submit(J,task_index,r,i);
 task_index←task_index+alloc;
 $u_{i,v} \leftarrow u_{i,v}$ +load;
   If task_index>p then return accept;
     endfor
    Endwhile
   i←parent(i);
Endwhile
i←v;
While $i\neq\Phi$ do
 While $share_i^{curr}<share_i^{\max}$ do
      For each $r\in R_i$ do
        (alloc, load) ←submit(J,task_index,r,i);
      Task_index←task_index+alloc;
        $share_i^{curr} \leftarrow share_i^{curr}$ +load;
        If task_index>p then return accept;
      endfor
 endwhile
 i←parent(i);
endwhile
Cancel all allocated sub-tasks in the above.
Return reject;
```

6. SIMULATION RESULTS

Herein, the resource allocation scheme is simulated by the gridsim toolkit [15]. Under three different conditions, the simulation result in hierarchical virtual enterprise is shown in Fig. (**4**). Five kinds of resources are used as shown in Table **6**. In the first and second situation, every virtual enterprise has only one resource provider. Under the first circumstance, all resource providers have identical resource capacity. Under the second circumstance, the resource capacity of resource providers is different (suppose R1 >R2 >R3 =R4 =R5). Under the third circumstance, R1 and R3 provide all resources to VE_1 and VE_2 respectively, other resource providers allocate their resource to the corresponding virtual enterprise on average.

Table 6. The characteristics of resource.

Resource Type	Performance of Processor (MIPS)	Number of Processor
R1500	1500	20
R1250	1250	20
R1000	1000	20
R750	750	20
R500	500	20

The job of every user is generated in the style of poisson distribution, with the internal arrival time of five minutes. The number of task in every job is a random number among 2 to 32. The scope of every job is from 100,000MIPS to 1,000,000MIPS. The deadline is beyond the average execution time by 20% to 100%. The total number of jobs submitted by every user is 1000.

This scheme is here compared with other resource allocation schemes, including least load first, random method and round robin method. Least load first method selects the resource provider with least usage to access hierarchical virtual enterprise. Random and round robin methods select resource providers in random and round robin respectively.

If the job of user can satisfy the deadline, it will be accepted and allocated to the resource provider. After simulation, the average receptance of virtual enterprise user is as shown in Fig. (**5**). Under the first condition, every resource provider has the same resource content. The receptance of VE_1 and VE_2 is lower than that of other virtual enterprises, because their resources are shared by other virtual enterprise users. On the contrary, the receptance of the users of VE_3, VE_4 and VE_5 is relatively high because they can access the resource provider at a higher level. Compared with other allocation schemes, VE_1 and VE_2 in sharing scheme have higher receptance.

Under the second condition, although the resource provider of VE_1 and VE_2 has good performance, the receptance of VE_1 and VE_2 in other schemes is still low. The virtual enterprise user in sharing scheme shows comparatively equal receptance. Under the third condition, more complex resource sharing strategies are simulated. Because there are many resource providers for VE_3, under other resource allocation scheme conditions except for sharing scheme, VE_3 shows higher receptance. The users in sharing scheme show similar receptance as the outcome on the far right as shown in Fig. (**5**).

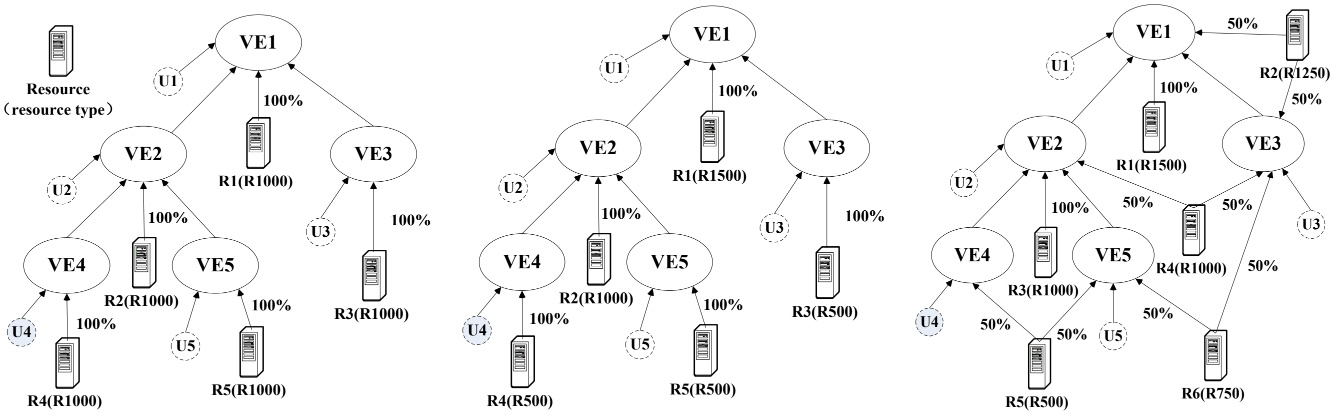

Fig. (4). The simulation environment of virtual enterprises.

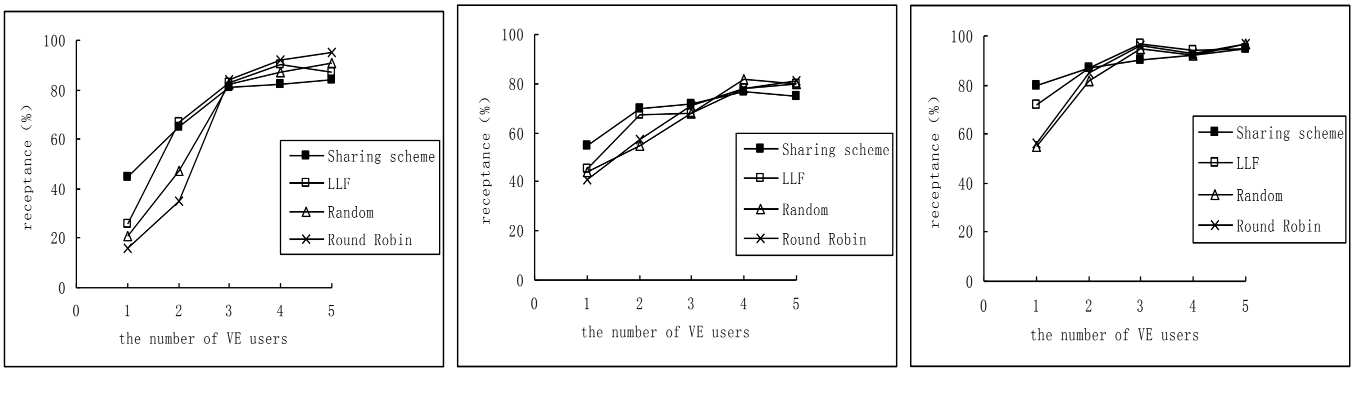

Fig. (5). The average receptance of virtual enterprise user under three different conditions.

Table 7. The receptance and standard deviation of different strategies under three conditions.

	Resource Allocation Scheme	The First Condition	The Second Condition	The Third Condition
Average receptance(%)	Sharing scheme	71.4	69.8	88.8
	LLF	70.6	67.6	89
	Random	65.6	65.8	84.2
	Round Robin	64.4	65.6	85.4
Standard deviation	Sharing scheme	16.59	7.78	5.72
	LLF	23.67	12.44	10.22
	Random	27.23	14.57	17.31
	Round Robin	32.52	14.83	17.09

The average receptance and standard deviation in Fig. (**5**) are shown in Table **7**. Compared with other schemes, the sharing scheme has higher receptance. At the same time, the lower standard deviation also reflects that this scheme is comparatively fair.

CONCLUSION

In this paper, based on hierarchically distributed virtual enterprises, a resource allocation scheme is provided for realizing fair resource sharing among these virtual enterprises. Under specific resource sharing strategy conditions, the internal equal sharing strategy is put forward. The resource middleware in virtual enterprise manages the strategy and the position of virtual enterprises, so that they can collaborate with one another to induce effective resource allocation. From the results, it is shown that the sharing scheme is comparatively fair than other schemes and the practical effect is better.

CONFLICT OF INTEREST

The author confirms that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

Thanks for the support from Hunan province philosophy and social science fund project (13YBA358), high level talents introduction project of central south university of forestry and technology (0318).

REFERENCES

[1] J. Pu, and Y. Zhu, "Research on the use of grid technology in virtual enterprise," *Computer Integrated Manufacturing System*, vol. 11, no. 7, pp. 1019-1024, 2005.

[2] S. Venugopal, R. Buyya, and K. Ramamohanarao, "A taxonomy of data grids for distributed data sharing management and processing, *ACM Computing Surveys*, vol. 38, no. 1, pp. 1-53, 2007.

[3] Z. Lining, "The development and research situation of grid technology," *Modern Intelligence*, vol. 25, no. 10, pp. 69-72, 2005.

[4] F. Li, D. Shan, T. Ma, "Research on the collaborative production scheduling mode of virtual enterprise group based on multi-agent," *Computer Application Research*, vol. 30, no. 6, pp. 1624-1629, 2013.

[5] F. Wang, Y. Hou, J.Yang, "Research on spatial data sharing and interoperability technology under grid environment," *Computer Science*, vol. 36, no. 1, pp. 96-100, 2009.

[6] J. Zhang, D. Yang, "The architecture model of modern logistics information grid," *System Simulation Technology*, vol. 1, no. 3, pp. 123-130, 2005.

[7] A. Sulistio, G. Poduval, R. Buyya, and C. K. Tham, "On incorporating differentiated levels of network service into GridSim," *Future Generation Computer Systems (FGCS)*, vol. 23, no. 4, pp. 606-615, 2007.

[8] Q. Zhao, and R. Xiao, "Research on game analysis and mechanism design of virtual enterprise information sharing," *Computer Integrated Manufacturing System*, vol. 13, no. 8. pp. 566-1571, 2007.

[9] B. Huang, N. Xiao, and B. Liu, "The design and realization of data access under grid environment," *Computer Engineering and Science*, vol. 27, no. 3, pp. 13-15, 2005.

[10] J. Sun, and Q. Sun, "Discussion on the information organization and management under grid environment," *Modern Intelligence*, no. 7, pp. 51-54, 200.7

[11] J. Xiao, Y. Wang, and N. Zou, "Research on the grid information resource establishment and sharing," *Intelligence Magazine*, vol. 24, no. 10, pp. 12-14, 2005.

[12] J. P. Goux, S. Kulkarni, J. Linderoth, and M. Yoder, "An Enabling Framework for Master-Worker Applications on the Computational Grid," In: *Proc. 9th IEEE Symp. On High Performance Distributed Computing*, 2000, IEEE Press.

[13] X. Xie, G. He, and D. Zhou, "The design and realization of manufacturing grid data resource integrated system by using middleware technology," *Modern Manufacturing Engineering*, no. 4, pp. 24-28, 2011.

[14] X. Wang, "Research on the work flow modeling oriented to grid monitoring," *Computer Technology and Development*, vol. 23, no. 8, pp. 70-74, 2013.

[15] R. Buyya and M. Murshed, "GridSim: a toolkit for the modeling and simulation of distributed management and scheduling for Grid computing," *Concurrency and Computation: Practice and Experience*, vol. 14, no. 13-15, pp. 1175-1220, 2002.

Received: September 22, 2014 Revised: November 30, 2014 Accepted: December 02, 2014