© 2017 IJSRST | Volume 3 | Issue 3 | Print ISSN: 2395-6011 | Online ISSN: 2395-602X
Themed Section: Science and Technology

Enhancing Resource Reservation by Applying Defer Time in FCFS Scheme for Grid Computing Environments

S. Nirmala Devi*[1], Dr. A. Pethalakshmi[2]

*[1]Department of Computer Science, Government Arts and Science College, Kangeyam, Tamilnadu, India

[2] Department of Computer Science, MVM Government Arts College, Dindigul, TamilNadu, India

ABSTRACT

Grid Computing provides a promising environment where the resources are shared for the earlier execution of the program. The resource management techniques include resource discovery, resource allocation and resource reservation. This paper analyzes the FCFS based advance resource reservation. It has been found that the resource denial is more in case of FCFS. Hence the defer time is introduced. Defer time refers to the time until which the reservation can be postponed. On analysis, it is found that the resource denial and resource idle time are reduced.
Keywords: Grid Computing, Resource Management, Resource Reservation, FCFS, Defer Time.

I. INTRODUCTION

The advent of networking in the computer field shows a vibrant improvement in communication and computation. This paved path for the advanced computing technologies, to provide earlier execution of the programs. The prominent computing technologies as distributed computing and parallel computing evolved during late 1970's. In distributed computing, the loosely-coupled systems co-operate and work to accomplish a particular task. In parallel computing, the tightly coupled systems co-operate to accomplish a particular task. In parallel computing the processor share the common memory, whereas in distributed computing the processor has separate memory. By exploiting these features, the grid computing has started its evolution from the mid of 1990's. The grid computing preserves the transparency and loosely coupled features of distributed computing. It inherits the parallel processing from parallel computing.

The grid provides a heterogeneous environment, where the nodes co-operate and work to accomplish a particular task. Each node possesses some resources. These resources in the grid environment are shared among the participating nodes in the virtual organization. The resources are left underutilized if kept with a particular node. Sharing enables the efficient utilization of resources. The grid environment is highly unpredictable as nodes and the associated resources may enter or leave the environment at any time. This feature makes the necessity for more improved methods towards resource management. The resource management includes the resource discovery, resource allocation and resource reservation. There were various tools available for grid computing as Globus ToolKit, Nimrod – G, Condor – G, Legion etc. The MDS handles the monitoring and discovery of services in the grid environment.

Grid Computing

Grid computing may be defined as loosely-coupled environment where the resources are shared to accomplish a particular task. The sharing can range from local network to the environment which sprawls across the world through internet. The sharing is made possible by means of forming virtual organizations. The resources are scattered across the heterogeneous environment and of dynamic nature. The resources in grid is said to be dynamic in nature as it can enter or leave the environment at any time. Since the grid computing environment is evolved from distributed computing, the environment is visualized as a large super computer, when a user submits a job.

Buyya and Venugopal[1] define grid as "a type of parallel and distributed system that enables the sharing, selection, and aggregation of geographically distributed autonomous resources dynamically at runtime depending on their availability, capability, performance, cost, and users' quality-of-service requirements". The grid environment strives for earlier execution of the program with optimal usage of resources. The major task of grid environment is resource management. The following figure 1, shows the grid computing environment.

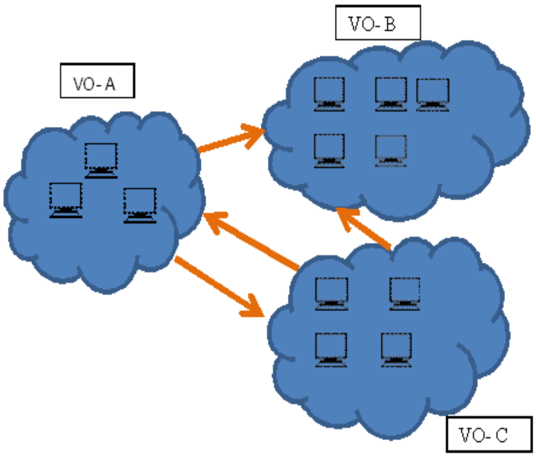

Figure 1. Grid Computing Environment

The sample grid environment depicted in figure 1, comprises of three virtual organizations (VOs). Each participating VOs consists of the nodes. The nodes are interconnected in the VO. The VOs are connected together in the internet. Thus forms the grid environment. The nodes can be the computational system, storage system or any other device. The following architecture diagram in figure 2 shows the resources and the LRM (Local Resource Manager) in the grid. The resources are associated with each computational node. The lower layer is the grid environment. Upon which the computational nodes built. There can be any number of nodes.

Computational Node 1	Computational Node 2	…	Computational Node N
LRM 1	LRM 2	…	LRM N

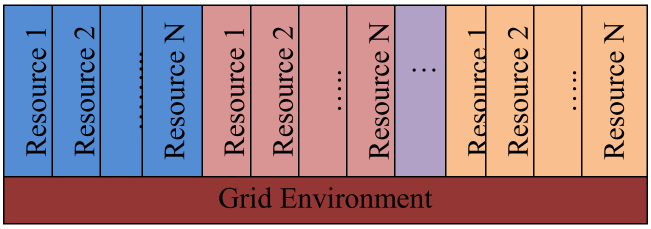

Figure 3. Architecture Diagram showing the resources and LRM

Resources

A resource may be a logical entity, such as a distributed file system, computer cluster, or distributed computer pool [2]. The physical entities such as printer, scanner also forms the resource. As a whole the resources includes computing resources, data resources, storage resources, service resources, network resources and web resources. All these resources are heterogeneous and dynamic in nature. And efficient utilization of resources is required for the problem solving in grid environment [3]. The grid resources can be categorized as computing resource, data resource, storage resource, service resource, network resource, web services. These resources are of dynamic in nature as it can enter or leave the environment at any time. Similarly, the resources are heterogeneous in nature as it belongs to any environment or platform.

Resource Management

There is various resource management activities involved in grid computing. The Local Resource Manager (LRM) maintains the details about the resources in that particular node. In addition to this, the details of the resources in the grid can be maintained centrally, in distributed fashion or by using any of the hybrid approach [4].

The resources management activities in the grid environment include resource discovery, resource allocation and resource reservation. The following figure 3 specifies the resource management activities.

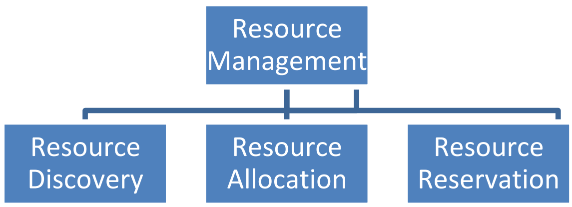

Figure 3. Resource Management Activities

Resource Reservation

The LRM (Local Resource Manager) maintains the details of the resources associated with the participating node in the grid environment. Irrespective of the reservation schemes available, the resource reservation is supposed to maintain the reservation list or queue. The reservation list maintains the job which has reserved the resource and also the start time and finish time of the resource request. The details on free slots are also maintained. The following figure 4 depicts the sample reservation queue.

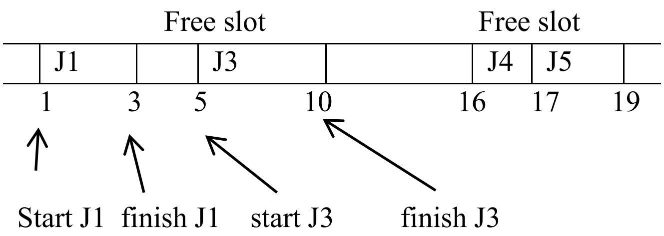

Figure 4. Reservation Queue

II. LITERATURE SURVEY

There are also various advanced reservation schemes available as First Come First Serve (FCFS) based reservation, priority based reservation, negotiation based reservation etc.,

The FCFS (First Come First Serve) is considered as the base algorithm [5]. This algorithm considers the start time and finish time of the resource requirement as input. The reservation is done on FCFS model. When the required time is not available then the resource denial occurs. Inspite of resource being idle, the resource denial occurs as the entire time duration could not be provided.

TABLE I. A SCENARIO CONSIDERED

JID	ST	FT
J1	3	7
J2	9	12
J3	6	8
J4	15	19
J5	22	24
J6	27	30
J7	32	33
J8	20	26
J9	37	40
J10	42	44

The table I depicts a scenario where ten jobs from J1 to J2 are considered with start time (ST) and finish time (FT). when FCFS approach is applied the reservation list would have allocation as defined in table II.

TABLE II. FCFS BASED ALLOCATION

JID	ST	FT	UT
J1	3	7	4
J2	9	12	3
J4	15	19	4
J5	22	24	2
J6	27	30	3
J7	32	33	1
J9	37	40	3
J10	42	44	2

It is evident from table II that the reservation for J3 and J8 are denied. The resources are also underutilized as the J3 is not provided even when is idle from 7 to 8 unit. This entire time units from 7 to 9 are the resource idle time. Thus the FCFS scheme suffers of drawbacks as more resource denial and increased resource idle time.

Proposed work:

The proposed algorithm tries to eliminate the drawbacks of the FCFS scheme in resource reservation. This section deals with the defer time (DT) and the resource reservation algorithm using FCFS with DT.

Defer Time

When a user requests for a resource it has to submit the start time and finish time until which it requires the resource. But there may be some situation where the reservation can be postponed to some period of time. Thus defer time refers to the time until which the reservation can be postponed.

Free Queue

The free queue is maintained in the LRM. This specifies the free time units in between the reservations from the reservation list. The queue consists of the start time and finish time of the free slot. From the example scenario as depicted in table III, the reservation for J1 and J2 are made. Hence the reservation list would have the entries for J1 and J2. When J3 enters, it is required to find the free slots. The free slot s1 with start time of 7 and finish time of 9 is identified. Because of this the defer time of J3 becomes 9. After negotiating with the process the resource reservation can be continued or denied.

On encountering J3 & J8, the free queues may have the following free slots as said in table table III and IV.

TABLE III. FREE QUEUE ON ENCOUNTERING J3 FROM THE SCENARIO

QID	QST	QFT
1	7	9

The free slots available are selected for allocation. When DT is not applied then the resource may be kept idle, in spite of resource request.

TABLE IV. FREE QUEUE ON ENCOUNTERING J8 FROM THE SCENARIO.

QID	QST	QFT
1	20	22
2	24	27
3	30	31

Though there were three free slots available those cannot be used, as the required six time units are not available with any of the free slots. These are the resource idle time. The resource idle time is reduced because of reserving for J3. Hence 4 time units are reduced. Hence this proposed FCFS with DT reduces the resource idle time also.

A Scenario with DT

For the same scenario as depicted in table I, the defer time is included and presented in table III.

TABLE V. A SCENARIO CONSIDERED WITH DT

JID	ST	FT	DT
J1	3	7	-
J2	9	12	-
J3	6	8	9
J4	15	19	-
J5	22	24	-
J6	27	30	-
J7	32	33	-
J8	20	26	31
J9	37	40	-
J10	42	44	-

In table III, the job id J3 is having the start time 6 and finish time 8. But one more value 9 is computed based on the reservation. After negotiating with the process, it is required to decide whether the job j3 and j8 can be reserved with the defer time.

FCFS with DT algorithm

The algorithm for using DT in addition to ST and FT is given below.

```
Algorithm FCFS_DT(job_id, ST, FT, DT)
if reservation_list is empty then
 no conflict found hence reserve
else
  for i = 1 to list_size
      if Start_new ≤ start_i && finish_new ≥ finish_i
          || Start_new ≤ finish_i || finish_new ≤ finish_i
           Put i into temp_list
      end if
   end for
  if temp_list is empty then
        No conflict found, hence reserve
 else
      for i = 1 to list_size
         free_reserve = start_i+1 – finish_i
         if free_reserve  && finish_i <= DT then
           insert(free_queue(QID, QST, QFT))
        end if
      end for
      for i = 1 to size (free_queue)
         free_size = QST – QFT
         required_time = finish_new – start_new
         if (start_new >= QST and
              required_time >= free_size) then
            reserve(QST, QFT) and
            insert in reservation list
         end if
      end for
  end if
end if
```

The free slots until DT are inserted into the free_queue with the queue id (QID), queue start time (QST) and queue finish time (QFT). The required slot is reserved, if it is available.

Allocation based on FCFS with DT

By applying this algorithm, the resource for J3 is reserved. The resource for J8 could not be provided as 6 time units are not available until the defer time of 38. So, the reservation for J8 is denied and is represented in table VI.

TABLE VI. RESERVATION BASED ON FCFS WITH DT

JID	ST	FT	UT
J1	3	7	4
J3	7	9	2
J2	9	12	3
J4	15	19	4
J5	22	24	2
J6	27	30	3
J7	32	33	1
J9	37	40	3
J10	42	44	2

III. RESULTS AND DISCUSSION

Performance Analysis

The proposed FCFS algorithm with defer time considers two parameters for consideration.

Resource Denial

The major drawback in FCFS algorithm is the resource denial. The resource may be available after a period of time but it is not considered in FCFS. In table I, the J3 requires the resource from 6 to 8 time units. Since J1, reserves until 7 time unit. J3 is denied. Similarly J8 is also denied.

Defer time is included in table III, because of this the resource denied for J3 is provided in the proposed approach. But J8 is not provided with the resource. Hence the resource denial is reduced in the proposed algorithm.

This is computed by hit ratio.

Hit Ratio (HR) = number of allotted number of requested reservations

The hit ratio while applying FCFS is 8:10. It is reduced in FCFS with DT to 9:10. This is represented in the below figure.

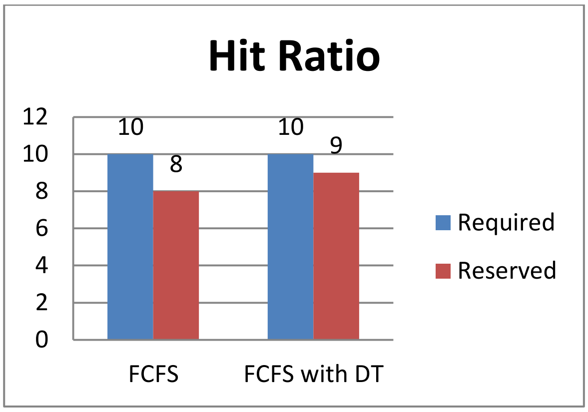

Figure 5. Hit Ratio by Applying FCFS and FCFS with DT

Resource Idle Time(RIT)

It has been identified there is resource idle time in FCFS. Form table I, it is learnt that the resource is idle for 2 time units from 7 to 9. But the J3 is not reserved.

In FCFS with DT, the resource idle time is utilized and J3 is reserved. By this the RIT is decreased in FCFS with DT.

IV. CONCLUSION

Thus resource reservation aims at guarantying the availability of resources when required. The FCFS scheme has drawbacks as more resource denial and more resource idle time is available. These are reduced in FCFS with DT. The work can be improved to cloud computing.

IV. REFERENCES

[1]. Buyya, R.; Venugopal, S. "A Gentle Introduction to Grid Computing and technologies". CSI Communications 2005, 29(1), 9–19. Google Scholar]

[2]. Foster, Ian, Carl Kesselman, and Steven Tuecke. "The anatomy of the grid: Enabling scalable virtual organizations." International journal of high performance computing applications 15, no. 3 (2001): 200-222.

[3]. Devi, S. Nirmala, and A. Pethalakshmi. "Resource Discovery for Grid Computing Environment Using Ant Colony Optimization by Applying Routing Information and LRU Policy." In Global Trends in Computing and Communication Systems, pp. 124-133. Springer Berlin Heidelberg, 2012.

[4]. Buyya, Rajkumar, David Abramson, Jonathan Giddy, and Heinz Stockinger. "Economic models for resource management and scheduling in grid computing." Concurrency and computation: practice and experience 14, no. 13-15 (2002): 1507-1542.

[5]. Anthony Sulistio and Rajkumar Buyya, “A Grid Simulation Infrastructure supporting Advance Reservation”, proceedings of the 16th Intenational Conference on parallel and Distributed Computing and Systems, Nov 9 – 11 2004, MIT, Cambridge, USA, pp. 1-7.