A Hybrid Approach for Scheduling and Replication based on Multi-criteria Decision Method in Grid Computing

Nadia Hadi[1], Ghalem Belalem[2] , Nawel Doudou[3] and Amina Benzouak[4]

[1] Dept. Department of Computer Science, Faculty of Sciences, University of Oran – Es Senia, Algeria
BP 1524, EL M'Naouer, Oran, Algeria
Fax: (213) 41 51 47 69

[2] Dept. Department of Computer Science, Faculty of Sciences, University of Oran – Es Senia, Algeria
BP 1524, EL M'Naouer, Oran, Algeria
Fax: (213) 41 51 47 69

[3] Dept. Department of Computer Science, Faculty of Sciences, University of Oran – Es Senia, Algeria
BP 1524, EL M'Naouer, Oran, Algeria
Fax: (213) 41 51 47 69

[4] Dept. Department of Computer Science, Faculty of Sciences, University of Oran – Es Senia, Algeria
BP 1524, EL M'Naouer, Oran, Algeria
Fax: (213) 41 51 47 69

Abstract

Grid computing environments have emerged following the demand of scientists to have a very high computing power and storage capacity. One among the challenges imposed in the use of these environments is the performance problem. To improve performance, scheduling and replicating techniques are used. In this paper we propose an approach to task scheduling combined with data replication decision based on multi criteria principle. This is to improve performance by reducing the response time of tasks and the load of system. This hybrid approach is based on a non-hierarchical model that allows scalability.

Keywords: *Data Grids, Tasks Scheduling, Data Replication, Multi-criteria decision, Placement replicas.*

1. Introduction

Advances in technology, in this decade, have been particularly manifested by the significant improvement of computational power. However, this power is not sufficient for the new applications that are very demanding of computational resources. These new applications use large sets of information and generate large amounts of data. Data cannot be stored centrally but in a distributed manner and world users can send tasks that access to data. These advances have allowed access to computing resources at lower cost and them naturally leads to new concepts, such as grid computing.

To achieve these objectives, it is necessary to schedule the various tasks (client requests) and place the data so that the execution takes place under optimum conditions, by taking into account the hardware and the grid state.

In this context, we proposed a scheduling and multi-criteria replication approach in order to improve the system performance [2][18].

We aim to improve response time to client requests and reduce the system load and storage by using the replication only when necessary [4]. Our proposal is based on the decision to schedule the task [8] or replicate the required data using the methods of multi-criteria decision aid.

This paper is organized as follows: In the next section, we discuss scheduling techniques used in the grids, and some related works. In Section 3 we present the principles of the methods of multi-criteria decision aid and a description of our scheduling approach, using some *UML* diagrams and some necessary algorithms to show the various steps and features of the system. In Section 4 we present a comparative study of our work with previous work presented in Section 2. We reserve Section 5 to the experiment part launched by the simulator that we designed to meet the original objectives; we show some results and their interpretation. Finally, a conclusion is presented to synthesize this work and a set of perspectives are proposed for future extensions of the proposed approach.

2. Previous Work

To improve access time and data processing, several solutions have been proposed, among which we can find techniques for scheduling tasks and replicating data. These techniques are used to handle data access efficiently.
A good scheduling strategy will allow rapid access to desired data and a reduction in the system loads. Among the current works on the problem of scheduling tasks and replicate data, we can mention:

- K. Ranganathan and I. Foster realized the importance of data locality in scheduling problem. The latency of a task depends on the computing resource and the location of the required data [15], [16]. To reduce this latency, it is necessary to reduce the latency related to the calculation and waiting time.
- A. Chakrabarti and his colleagues [7] have improved Ranganathan's previous work: they proposed the approach IRS (Integrated Replication and Scheduling) to improve the scheduling performance. Performance is improved by refining successive data placement and scheduling of computational tasks.
- In [10] an approach of scheduling and data placement in the grid is proposed and explained: The authors present an algorithm that combines data management and scheduling. The tasks are sequenced directly to data location. They proposed a new semi-static algorithm that allows an optimization of the data mapping and request scheduling at run-time. They tested a set of heuristics designed to adapt scheduling and data placement to change in data usage patterns.
- In [12] and [9], a dynamic replication model is proposed that is based on an exponential rate. The files that are being requested in the present, are apt to be requested in the near future. Therefore, the popular file is detected by analysing the number of access to the data files by the users. And, the popular file can be deduced via its access rate from the clients. In other words, the popular data files are identified by analysing the access histories.
- In the work reported in [1], [11], the grid used is modeled using a tree level hierarchical structure where dynamic replication techniques are used. If there is no space for the replica, only the files with a low cost of transfer will be deleted i.e. considering the Bandwidth between source and destination. Bandwidth is an important factor for deleting and this leads to a better performance.

In the paper [13], a dynamic hierarchical replication algorithm is proposed, it places replicas in appropriate sites, ie best site that has the highest number of access for that particular replica. The algorithm minimizes access latency by selecting the best replica when various sites hold replicas. The proposed replica selection strategy selects the best replica location for the users running jobs by considering the replica requests that waiting in the queue and data transfer time. It stores the replica in the best site where the file has been accessed most, instead of storing files in many sites.

3. Proposed Approach

In this section, we present a scheduling and replication approach that is based on the multicreteria decision of transfer task or replication the data to reduce response time and system load by ensuring adequate availability of data. Our approach should be able to choose between scheduling a task to a node that has the required data or replication of required data to the node containing the task taking into account various criteria simultaneously. This will allow us improve the system performance by reducing the response time and number of generated replicas.
In our approach of task scheduling tasks are managed from their arrival until their execution to minimize performance metrics, such as response time and the number of created replicas. It allows measuring the influence of our approach on data replication.

3.1 Objectives

The originality of the proposed approach is the use of decision-making methods based on multi-criteria decision. It is to choose between scheduling the request sent to the node that contains the required data or replicate the data required to the node that submit the query. This choice is conditioned by a number of criteria related to the Grid state, such as the size of the data to be replicated, overloading the node hosting the required data, insufficient capacity Storage of node of the issued task, ...
The approach proposed in this paper has several objectives:

- It allows the formation of groups of nodes within sites; On the other hand, the fact of using the clustering process (concept of groups) allows better management of scheduling and replication that resulted in a considerable gain in time and a reduction in the number of replicas;
- It makes a decision to either replicate data or scheduling requests by using the multi-criteria methods.
- It can be applied to any topology: hierarchical, flat or whatever.
- It considered the two types of tasks (reading and writing) with the management of data consistency during the update.

3.2 Proposed Model

The proposed model of grid consists of several sites. Each site includes a number of nodes (see Figure 1). A node consists of a computing element and storage element. In a site, nodes are interconnected by intra-site bandwidth (LAN). These links can form a certain topology, such as the ring topology, the star topology or a random topology. A site consists of a set of groups (clusters) and sites are interconnected by inter-site bandwidth (WAN).

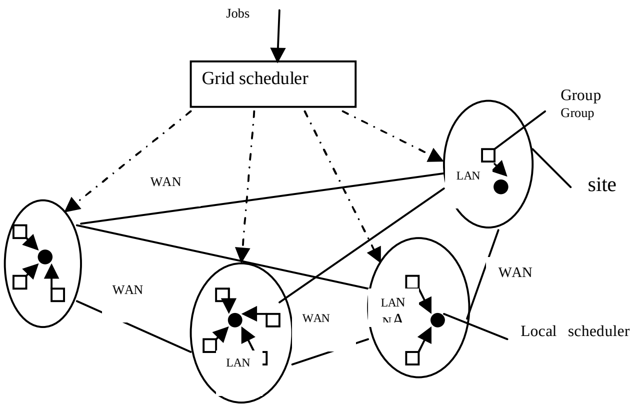

Fig. 1: Proposed Model

3.3 The Process of Aid Decision

The use of methods for decision aid based on a single criterion decision quickly showed its limitations in solving problems of multiple choices. Indeed, the optimization of economic function is difficult when the problem under consideration involves factors not easily quantifiable.
The use of methods of multi-criteria decision aid was therefore widely used in recent years [17], [6]. They allow to takes into account various viewpoints in the decision process which are expressed through the importance given by each actor (makers) to judgment criteria considered. This allows managing conflicts between several actors.
Finally, the methods of multi-criteria decision aid are used to guide discussion towards a set of possible solutions or options.

Multi-criteria decision making is often based on relationships between actions, and criteria that express the preferences of decision makers in order to compare the actions between them. These actions are the possible solutions considered in the multi-criteria decision process.
The multi-criteria decision process uses a double entry table, knows as matrix performance where each row is an action and each column is a criterion. The intersection between row and column represents the judgment or performance.
The decision-making process which is used is built on two multi-criteria methods:
1-The method *Electre-I* [20]: this is to choose between two actions "replicate data" or "task schedule", taking into account the following criteria:
C1: the size of requested data.
C2: the Number of replicas of requested data in the task group.
C3: the size of the queue of node having task.
C4: The number of nodes in the group having a capacity of storage greater than the size of the data.
2-The Method *Electre-III* [20]: In case the result of *Electre-I* is "replicate" then it should determine the location or the node where we have to replicate the data.
More information can be used in the grid, we can mention:

- A task represents the requirement requested by a client, it may be a request to read or write on a single data;
- Data is original and has no replicas at the system initialization; data are stored in node in different sites.
- A node can submit a task and may have several data but can have only one copy of each data at every moment during the course of the system.
- The node has the following characteristic: An identifier, compute element, storage element, a queue of tasks, reliability and constancy.

3.4 Scheduling Process

The proposed scheduling process is essentially composed of several schedulers (see Figure 2).

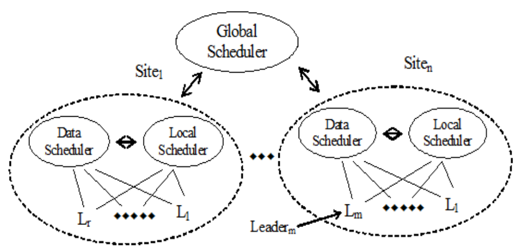

Fig. 2: Interaction between schedulers

These schedulers are set up to manage and facilitate the scheduling mechanism:

- Global scheduler for the whole grid: It receives tasks from the user and sends them to the best node according to proposed algorithm.
- Local scheduler at each site: It has replica algorithm and control data replication according to proposed algorithm (Algorithms 3 and 4), and sends tasks to nodes in different groups of his site.
- Data scheduler for each site: It controls data transferring in each node and provides a mechanism for accessing the data directory.

A. Preparation phase

It is a phase that develops and prepares the needed information to apply the proposed scheduling approach. It starts with the formation of groups and then designs a set of directories.

- The formation of node groups:

Several techniques and algorithms are proposed in the literature for the formation of groups. Our proposal is summarized by the following algorithm (Algorithm 1).

Algorithm 1 : Formation of groups

```
Input ( site_i : site)
Output (groups of site : groups)
If (site_i.Number of nodes=1) Then
group ← site_i;
Else
For all (site.i.List_nodes) do
Compute (node.number_link);
End for;
J← 0;
For all (site_i .list_nodes) do Max_nblink(list_nodes,
node);
If (node_taken=0) then j ←j+1;
Group_j ← make_group(node, nodes_related);
End if;
End for;
End if;
End.
```

Each group is managed by a particular node which is selected from the group; we call this node the leader of the group. The leader knows the history of all the nodes of his group. The choice of leader is made by a voting procedure, in our work, the node with the highest stability is elected, else the most reliable, else the one with the highest speed.

-The Design of the required directories of information:

The proposed process of scheduling uses a set of information organized into directories. These directories contain all the information in different stages of operation of our proposed scheduling approach.

We used several directories:

i. A directory of the global scheduler for the whole grid: it contains information on all sites of the grid.
ii. A directory of the local scheduler of each site, the local scheduler has information on all groups of its site.
iii. A directory of the scheduler data at each site: the scheduler data collects information on all data of its site and puts it in its directory of scheduling data.
iv. A directory for the group's history: the leader has the history of all nodes in the group.

B. Analysis and Treatment Phase

This phase consists of two steps:

a. Identification of the type of task (writing or reading)
b. Task Treatment, two types of treatments are possible: the task is of reading and access to data or task is of writing and modifying the data.

The following activity diagram determines the functionality and the type of treatment in the case of a reading task (See Figure3).

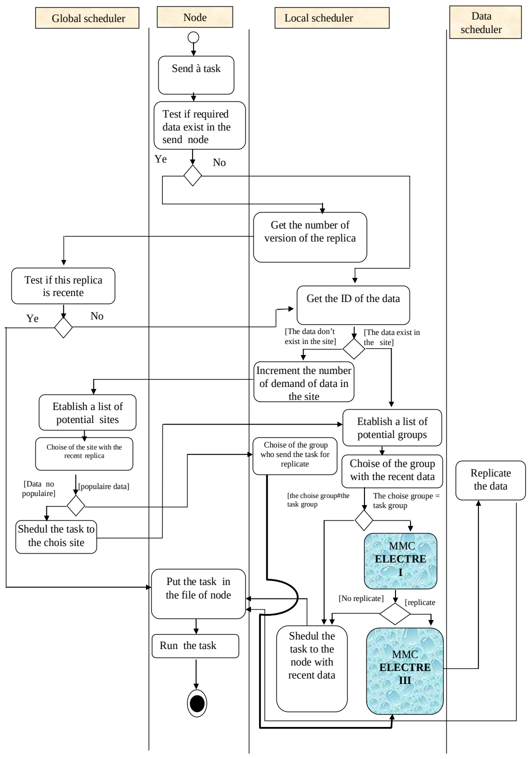

Fig. 3: Activity Diagram for the Reading Task

Several algorithms are used for the treatment of the reading task and are necessary to make the decision to transfer or replicate data.

1- The decision of replicate the data is described by Algorithm 2 (It returns true if we should replicate, false if not).

Algorithm 2 : Decision of replication

Input (the site , number of nodes , history)
s1 ← threshold(number of nodes)
/* Function "threshold" calculates the product of the number of nodes * *f*, Where *f* is a constant representing the desired availability of replicas */
s2 ← threshold2(history)
/* Function threshold2 calculates the product of the number of access (Population) * *f*1 , Where *f*1 is a constant representing the desired popularity of data */
If(site.numberOfReplicas<s1)**And**
(history.numberAccessofData>s2) **Then**
replication ←True
Else
replication ←False
End if

2-The decision of scheduling the task to the node that has the data or replication of the data is made by Algorithm 3.

Algorithm 3: Replicate or Scheduling

Input: the chosen group: SelectGroup;
Task: task; Data of task: $Data_i$
Begin
If (SelectGroup != TaskGroup) **Then**
Schedule ( );
/* Schedule the task to the node with the appropriate replica */
Else
/* the chosen group is the group of task */
/* apply the multi-criteria method ELECTRE I to make a decision (replicate or Schedule) */
Call –**ELECTRE I**( );
If (Call-**ELECTRE** I( ) = Replicate) **Then**
/* must determine the location of the replica (which node in the group), we used multi-criteria method ELECTRE III */
Call **ELECTRE III**( );
/* result is a node among the nodes of the selected group */
Replicate ($Datat_i$; $Node_j$) ;
/* $Node_j$: is the result **of ELECTRE III ( )***/
Else
/* Schedule the task to the node with the given task ($Data_i$) */
Schedule ();
End If;
End If;
End.

The writing tasks modify the content of replicas (see Figure 4). It creates the problem of incoherence between the replicas.

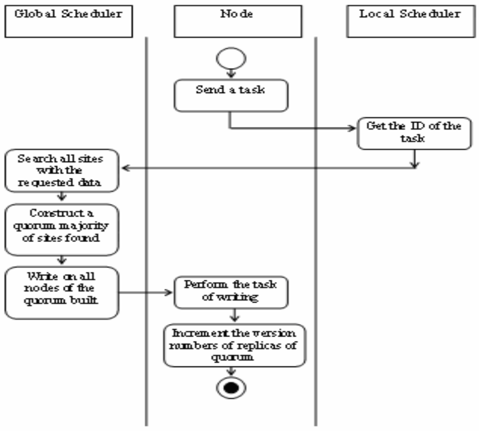

Fig. 4: Activity Diagram for Writing Task

To resolve this problem and ensure coherence, we used the technique *Quorum* [5]. Algorithm 4 calculates the Quorum of site.

Algorithm 4 : Calculation_Quorum

/* this algorithm calculates the Quorum of a site
to perform the writing task*/
Input (The group selected : Select group)
If (site.belong(DataOfTask)) **Then**
numberRep ← number_Of_replicats_Of_Data()
Quorum ← (nbrRep/2)+1 /*calculate Quorum majority*/
End if

IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 5, No 2, September 2012
ISSN (Online): 1694-0814
www.IJCSI.org

4. Comparative study

The Grid architecture for which the replication and scheduling technique is developed makes the strategies different from one another. Almost all the replication strategies are found to improve the availability of data and reduce the response time. Table 1 shows a summarized form of features of all strategies presented in related works above.

Work / **Characteristics**	Ranganathan [15], [16]	Chakrabarti [7]	Tang [19]	Desprez [12]	Dang [9]	Madi [12]	Horri [11]	Mansouri [13]	Our Approach
Year	2002/ 2003	2005	2006	2006/2007	2007	2008	2008	2012	2012
Topology	Multi-tier with several s ES	Multi-tier With centralized ES	Multi-tier with two level and centralized scheduler	Multi-tier with three levels	Multi-tier with centralized replicator and centralized resource Broker	Multi-tier with two levels and centralized scheduler	Multi-tier with three levels and centralised replica manager	Multi-tier with three levels	Multi-tier with two levels and centralized ES
Replication decision	Yes centralised	Yes decentralised	Yes two dynamic centralised and dynamic distributed	yes	Yes centralised	Yes Centralized	Yes centralised	Yes	Yes Decentralised and multicriteria
Reduce mean job time	yes	Yes	Yes	yes	yes	yes	yes	Yes	Yes
Scalability	No	No	yes	yes	No	?	?	?	yes
Reduce number of replicas	?	?	?	?	yes	?	?	?	yes
Type of task	Read	Read	Read	Read	Read	Read	Read	Read	Read /Write
Scheduling decision	yes	Yes Decentralised with specific scheduling algorithms	Yes Scheduling Heuristics	Yes site selection taking into account the time o f transfer	Algos of optorsim	Yes With scheduling heuristics	Yes Algorithms of resource Broker	Algos of optorsim	Yes decentralized and multicreteria Algorithm
Replica placement	yes	Yes		Yes A linear program solution	Yes Dynamic algorithm	Yes Distributed dynamic algo	yes	Yes	Yes Decentralized and Multicriteria Algorithm
Popular replica	yes	?	yes	yes	yes	?	?	Yes	yes
Queue size	no	yes	?	?	yes	?	yes	yes	yes
Simulator used	Chicsim	GridSim	XDrepsim	Grid sim	Optorsim	XDrepsim	Optorsim	Optorsim	

Table 1: Summarized Form of Features of all Strategies presented in Related Works

We have summarized in this table the different strategies combined replication and scheduling for a grid environment. It can be observed that there exists no standard architecture for a grid environment. Most of the work done follows a hierarchical architecture but actually a general graph is a more realistic architecture. Different modifications of the hierarchal architecture have been proposed to make it closer to the real grid environment.

Most of the techniques proposed have used simulation to evaluate and test the algorithms. It is also observed from this survey that most of the strategies compare the results with the very basic strategies like LFU and LRU and they do not compare the proposed ones with the existing strategies which are already far better than the basic LFU and LRU.

Another open research question is totally ignored by most of the work done: they consider that data in the grid environment is read only and hence replication is easy. In reality the data is not always read only; rather it is updatable. The replication strategies are unable to cope with data consistency when it is not read only, so in our work we considered that the task can be of two types, reading and writing. We note that in the table cells marked with '?' means of work have not touched this feature in the article.

5. Experimental Analysis

In order to analyze our approach and study its performance, we considered several performance metrics, such as Response Time and Number of replicas.

5-1 *Simulation Framework*

These metrics are calculated from the parameters shown in Table 2.

Parameter	Meaning
TF	The time of the end of task execution
TA	The arrival time of task
TR	The response time of a task
Ta	Waiting time of task
TM	Processing time of task
TTA	transfer time of the replica
NR_t	The number of tasks
TR_m	The average Response Time
NB_t	The number of tasks
NR_m	The average Number of replicas per site.
NR_d	The total number of replicas of the data *d*
NB_s	The number of sites in the grid

Table2: Description of Parameters.

To measure the performance of our model we chose the following metrics:

A. Response Time:

This performance metric measures the speed of processing a request from a client. We calculated the response time *TR* of a request *r*, by:

$$TR = TF - TA \quad (1)$$

Where TF= TTR+TM+Ta

The end time of execution of the task is the transmission time of the replica (TTR) plus the processing time TM plus the waiting time (Ta). If the data required by the task was not replicated when TTR is 0. And TTR= size/Bp (The size of the replica from the bandwidth).

We can also compute the average response time (TR_m) of a set of requests:

$$TR_m = \sum_{i=1..NB_t} TR_i / NB_t \,. \quad (2)$$

B. Number of replicas:

This metric measures the influence of Multi-criteria scheduling and replication algorithm of our approach on the number of replicas created by comparing the number of replicas created with an approach that does not use Multi-criteria Methods

We calculated for each data, the total number of replicas in the whole grid (*NR*).

We also calculated the average number of replicas (NR_{m_d}) of a data (*d*), per site, by the following equation:

$$NR_{m\_d} = NR_d / NB_s \,. \quad (3)$$

5-2 Experimental Results

For the experiment we built our own simulator based on the grid model proposed, where we implemented the two following approaches:

1-Our approach which uses the methods of multi-criteria decision aid for the choice of scheduling the task issued by the query or replicate the required data to the node or group of the task issued.

2-The second approach uses a scheduling and replication algorithm without multi-criteria decision making.

And make a comparative study of the two approaches depending on the results obtained during the simulations in order to evaluate the performance and the gain.

The simulations have been classified into three categories according to experience the impact of variations in the characteristics of the grid on the metric used.

A. **Experiment 1 "Impact of number of nodes"**

For this experiment, the simulations were performed with the following parameters: 8 sites, 10 data, two replicates per given data, 50 tasks and we varied the number of nodes from 20 to 500 in the grid.

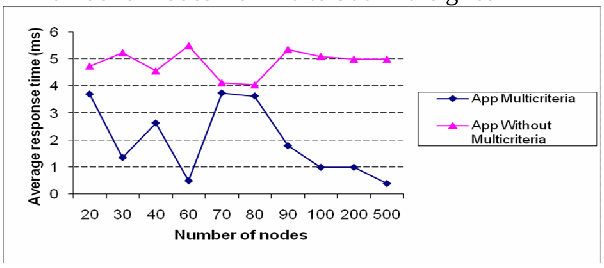

Fig. 5: Impact of number of nodes on the average response time.

The results show a gain result in the average response time for our approach (See Figure 5).

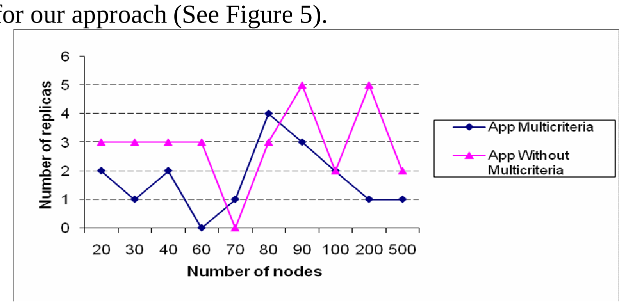

Fig. 6: Impact of number of nodes on the number of replicas created.

The results show a reduction in the number of replicas created when the number of nodes exceeds 70 nodes (see

Figure 6) and thus we deduce that our approach allows scalability.

B. Experiment 2: "Impact of the number of sites"

For this experiment, the simulations were performed with the following parameters: 40 nodes, 10 data, two replicates for each data, a wide bandwidth WAN 120 MB/S and LAN 200 MB/S, 50 tasks and we have varying the number of sites from 4 to 100.

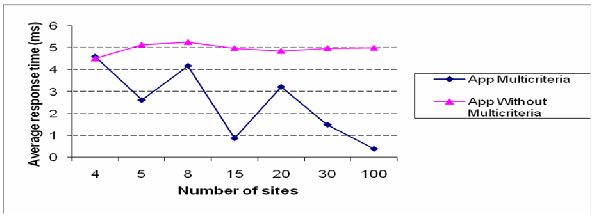

Fig. 7: Impact of number of sites on the average response time.

The results show a gain of 43% for our approach compared to the second approach (See Figure 7).

Fig. 8: Impact of number of sites on the number of replicas.

The results show that regardless of the number of sites, our approach reduces significantly the number of replicas created (see Figure 8).

C. Experiment 3: Impact of the Number of Tasks

For this experiment, the simulation parameters used are: 8 sites, 30 nodes, 10 data, two replicates per data, the wide bandwidth WAN 120 MB/S LAN and wide 200 MB/S, the number of tasks were ranged from 20 to 10,000 (50% reading and 50% writing).

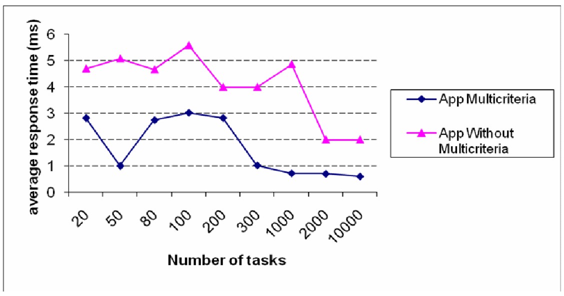

Fig. 9: Impact of Number of Tasks on average Response Time.

So, according to the results, whatever the number of tasks, the average response time for our approach is always lower than the other approach with a gain of 57% (see Figure 9).

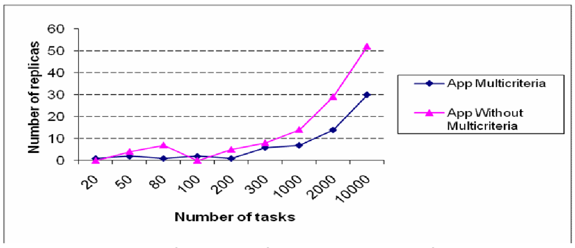

Fig. 10: Impact of number of tasks on number of replicas created.

For this experiment, the results show that from 10000 tasks, the number of replicas created is reduced by over 45% compared to the second approach (see Figure 10).

The Analysis of results allowed us to prove that the proposed approach in general improve the system performance and scalability and this is mainly due to the strategic placement of replicas created using methods for multi-criteria decision support for the placement of replicas and decision making between the replication of the data required and the scheduling of the query to the node with the required data, which meets the initial objectives.

5. Conclusion and Future Work

In the data grid environment, the primary goal of data replication is to shorten the data access time experienced by the job and consequently reduce the job turnaround time.

This is achieved with the hybrid approach of data replication and scheduling of tasks and especially by the introduction into the model proposed methods of multi-criteria decision support.

This approach relies on the decision of transferring the task to the node with the given data or replication of the required data to the node or the group of nodes that issued the request under a number of criteria under consideration in decision making.

The experiments have proved a winning result in the response time achieved (over 40%) and a significant reduction in the number of replicas created. This, whatever the characteristics of the data grid as the number of nodes, number of sites, number of tasks or the number of data. Always, the results in the different simulations are much lower compared to the approach without multi-criteria decision methods.

From the results obtained we can say that our approach has achieved its objectives, such as improved response time due to the scheduling tasks where there is required data and minimizing the number of replicas.

This confirms our theoretical goals that the performance of the task scheduling is positively influenced by the multi-criteria replication process which leads to equilibrate the system load.

Several perspectives can be the object of our present work, we can mention:

- Improving the location of replicas based on a market model of buyer-seller type using the auction [14];
- An interesting extension is to make our approach tolerate faults by using the prediction of replicas;
- Considering the weak consistency in many applications where strong consistency (accuracy) is not sought [5],[6];
- Eliminating the central entity that corresponds to the global scheduler and replace it by peer-to-peer architecture between local schedulers and collaborative negotiation [3];
- Expanding our proposal by a multi-agent system for making decisions.

References

[1] J. H. Abawajy, “Adaptive hierarchical scheduling policy for enterprise grid Computing systems”, Journal Network Computer and Application, Vol. 32, No. 3, 2009, pp.770-779.

[2] S. Bansal, B. Kothari, C. Hota, “Dynamic Task-Scheduling in Grid Computing using Prioritized Round Robin Algorithm”, IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 2, March 2011, pp. 472-477.

[3] M. Barhamgi, D. Benslimane, O. M. Aris, “PWSMS: A peer-to-peer web service Management system for data sharing in collaborative environments”, International Journal of Computer Systems and Engineering (IJCSSE), Vol.23, No. 2, 2008, pp. 89-106.

[4] G. Belalem, F. Bouharaoua, “Dynamic strategy of placement of the replicas in data Grids”, 9th International Conference, Pact’07, Pereslavl-Zalessky, Russia, Lecture Notes in Computer science (LNCS), Parallel Computing Technologies, Victor Malyshkin (ed), Vol. 4671, 2007, pp. 496-506.

[5] G. Belalem, Y. Slimani, “A Hybrid approach to replica management in data Grids”, International Journal of Web and Grid Services (IJWGS), Vol 3, No.1. 2007, pp. 2-18.

[6] I. Boudali, K. Ghedira, “A Distributed Multi-criteria approach for traffic regulation in public transportation system”. International journal of Applicable Artificiel Intelligence, Vol. 23, N°.7, 2009, pp. 559-632.

[7] A. Chakrabarti, R. A. Dheepak, S. Sengupta, “Integration of Scheduling and Replication in Data Grids”, HiPC, Vol. 3 N° 6, 2005, pp. 375-385.

[8] K. S. Chatrapati, J. U. Rekha, A. V. Babu, “Competitive Equilibrium Approach for Load Balancing a Data Grid”, IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 2, March 2011, pp. 427-437

[9] N. N. Dang, S. B. Lim, “Combination of replication and scheduling in data Grid”, Int. journal of computer science and network Security, Vol.7, No. 3, 2007, pp. 304-308.

[10] F. Desprez, A. Vernois, “Simultaneous Scheduling of Replication and Computation for Data Intensive applications on the Grid”, Journal of Grid Computing, Vol. 4, No. 1, 2006, pp. 19-31.

[11] A. Horri, R. Sepahvand, G. Dastghaibyfard, “A Hierarchical Scheduling and Replication Strategy”, International journal of computer science and network security, Vol. 8, No. 8, 2008, pp. 30-35.

[12] M. K. Madi, S. Hassan, “Dynamic Replication Algorithm in Data Grid: Survey”, International Conference on Network Applications: Protocols and Services, 2008.

[13] N. Mansouri, G-H. Dastghaibyfard, “A dynamic replica management strategies in data Grid”, Journal of network and computer applications, Vol.35, N°4, 2012, pp. 1297-1303.

[14] M. Mikolaj, “Cluster-based analysis and recommendation of sellers in online Auctions”, International Journal of Computer Systems and Engineering (IJCSSE), Vol. 22, No. 5, 2007, pp. 279-288.

[15] K. Ranganathan, I. Foster, “Decoupling Computation and Data Scheduling in Distributed Data-Intensive Applications”, International Symposium of High Performance distributed computing, Scotland, UK, 2002, pp.352-358.

[16] K. Ranganathan, I. Foster, “Simulation Studies of Computation and Data Scheduling Algorithms for Data Grids”, Journal of Grid Computing, Vol.1, No.2, 2003, pp. 53-62.

[17] O. Roux, D. Duvivier, V. Dhaevers, N. Meskens, A. Artiba., “A .Multi-criteria Approach to rank scheduling strategies”, International journal of production Economies, Vol.112, N°1, 2008, pp. 192-201.

[18] G. Sudha Sadasivam, V. Viji Rajendran, “An Efficient approach to Task Scheduling in Computational Grids”, International Journal of Computer Science and Applications, IJCSA, Vol.6, No. 1, 2009, pp. 53-69.

[19] M. Tang, B. S. Lee, X. Tang, C. K. Yeo, “The impact of data replication on job scheduling performance in the Data Grid”, Future Generation Computing System, Vol. 22, No. 3, 2006, pp. 254-268.

[20] J. Buchanan, D. Vanderpooten, “Ranking projects for an electricity utility using ELECTRE III”. International Transactions in Operational Research, Vol. 14, No. 4, 2007, pp. 309-323.

Nadia Hadi: is a doctoral student in computer science and her current research interests are Distributed system, Grid computing, Placement of Data, Data Replication and scheduling in Grid. She participated as a researcher in several projects since 2005 to date which the themes are “Distributed System” and “Resource Management in Grid Computing”.

Ghalem Belalem graduated from Department of Computer Science, Faculty of Sciences, University of Oran, Algeria, where he received PhD degree in computer science in 2007. He is now a research fellow of management of replicas in data replicas in data grid. His current research interests are distributed systems; grid computing, could computing and data grid placement of replicas and consistency in large scale systems and mobile environment.

Nawel Doudou A Master candidate in Department of computer science, Faculty of Sciences, University of Oran, Algeria. Her research interests are replication strategies and scheduling.

Amina Benzouak A Master candidate in Department of computer science, Faculty of Sciences, University of Oran, Algeria. Her research interests are replication strategies and scheduling.