J. Basic. Appl. Sci. Res., 3(1s)70-72, 2013
© 2013, TextRoad Publication

ISSN 2090-4304
Journal of Basic and Applied Scientific Research
www.textroad.com

Design and Implementation of Resource Allocation Algorithm Based on Parallel Intelligent Factors in Grid Networks

Zohreh Royaee

Department of computer engineering, Genaveh branch, Islamic Azad University, Genaveh, Iran

Received: June 10 2013
Accepted: July 9 2013

ABSTRACT

Network computation is a calculation method using computation power make several computer works a network and considering them as a unique virtual computer structure might perform great computation on them. One of the most important features of this network is computers arrival and exit to this network whole day. Its difficult to use this network because of above mentioned cases, thus, its required to introduce a flexible method due to these modifications in order to rehabilitate information and resources in grid networks. In grid network, various units included with different functions. One of these units is interfaces, their main function is to receive requests and allocate the most appropriate available resources in order to perform a new request. In this research, it was tried to introduce parallel method using intelligent factors in order to schedule and allocate resources. Results showed a speed rate four times higher than series systems.

KEYWORDS: grid networks, simple parallel, alternation, expert system, resource allocation

INTRODUCTION

Network computation enable virtual organizations to share distributed resources geographically in order to reach common goals with each other. Its possible with lack of a place and central control and reliable relationships. In order to solve problems in grid it was required that the best resources (computation power, memory, and data base and software) made possible as soon as possible and also it was applied in order to solve part of the problem. Yet, above mentioned method of data rehabilitation each tried to answer to the requests in the least time, but none of them are capable to come up with grid modifications and react flexibly to these changes.

We aimed to insert a intelligent program in units of grid network which act as interfaces, so that, by using network through time, these interfaces were enabled to understand network and available resources appropriately and minimize answer times. In fact, it might said that we convert each interfaces to intelligent factor those capable of learning and whenever required, it was used several interfaces in order to decrease allocation time, and schedule and each interface searched a distinct part.In following, this research introduce series method, and it was represent given method, performance of new method and comparisons and performed experiments in order to study this idea performance with previous methods.

2. Series methods algorithms

In order to execute requests in a grid, the request was forward to interfaces unit in network. This unit connects with other units then, detect required resources for specified works and select the best and most appropriate resources and schedule work for them. In series method, there are two stations, one act as cache memory of computer and maintains high valuable resources and the other one maintains other available resources of networks.

We defined an interface which initially search for all three requests(memory, processor, and graphic card) in cache and if it found a resource with all requested features of graphic card, processor, and memory increases the value one step higher and refer user program to that resource, otherwise, returns to main table and performed all steps again. Finally, it was updated both cache and main tables as a collection user request, was given to found resource in order to execute.

3.parallrl algorithm

It was used two stations, one act as cache memory of computer and maintains high valuable resources and the other one maintains other available resources of networks. It was considered three interfaces, each of them search for resources in a different specific distance: the first interface searches distances lower than 500km, the second interface searches between 500 to 1000km due to user system that requested a program for execution. In each interface, 3 factors, graphic card study, memory study and processor study search parallel. First factor in first interface search systems in lower than 500km than user system and only searched time and volume of requested processor. Second factor in first interface parallel with the first factor search systems for user requested graphic card in the same distance and finally, the third factor in first interface search systems for requested memory at the same distance.

Totally, interface search the cache station initially and if found resource which contains all 3 user requested factors, allocate the program to the resource and improve value of the resource one step higher, otherwise, return to base station and perform all steps again. Then cache table and main table updated completely. If a complete resource

*Corresponding Author: Zohreh Royaee, Department of computer engineering, Genaveh branch, Islamic Azad University, Genaveh, Iran zroyaee@yahoo.com

contains all three request factors not found, automatically, the system selects some similar resources as user requested, features of new resource is that each one contains at least one factor of user requested factors then executed the program on those resources parallel.

Three interfaces search parallel (parallel) called 1,2, and 3 interfaces.

Each interval searched predetermined resource in a restricted geographical district.

Three other interfaces namely; graphic card interface, memory interface, and processor interface shown in figure executed in second layer of alternation, all search for one related resource among resources

Finally combination unit combined found resource and select most appropriate resources for proper execution of request. 2 and 3 interfaces shown briefly in figure and main figure of each of them are similar to first interface.

4.proposed algorithm implementation

Now, after determination of details and performance of proposed algorithm, we implement this designed expert system by C#(C SHARP) programming language in both parallel and series modes, finally both implementation modes compared in time and function concepts.

First, we have a form contains resource register and intelligent search.

Press resource register bottom, programmer might add new resources to system those used in the searches.Intelligent search bottom designed to receive user requested information according to user program.Press intelligent search bottom, user enter into search form.

System introduced most appropriate resources to user based on given algorithm and after user confirmation, program execute on search systems in series mode.

5. time comparison of intelligent expert system parallel and series execution

In this step, system execute implemented expert in both serial and parallel modes on one processor(one core), two processor (two cores), and four processors(4 core) and obtain each execution time and finally compare obtained results.

5.1.intelligent expert system execution time diagrams in series and parallel modes

In serial execution mode, intelligent factors were executed in turn subsequently and due to program serial execution, execution time of every method (intelligent factors) computed separately and finally, final execution time obtained by total times of all intelligent factors.

In this project, due to application of one interface in series discussion that simultaneously search all features and factors are nor separate in the other hand it was applied 2 interfaces in parallel discussion that each of them separated in three line that each line search specific features. Thus, its impossible to compute separately time of each factor from above figure, because a factor determined before hand and lines inside parallel factor in series discussion simultaneously studied by interface, thus, results are complicated and they are not separated to compute time separately. Thus, we prefer to compare total series time with total parallel time.

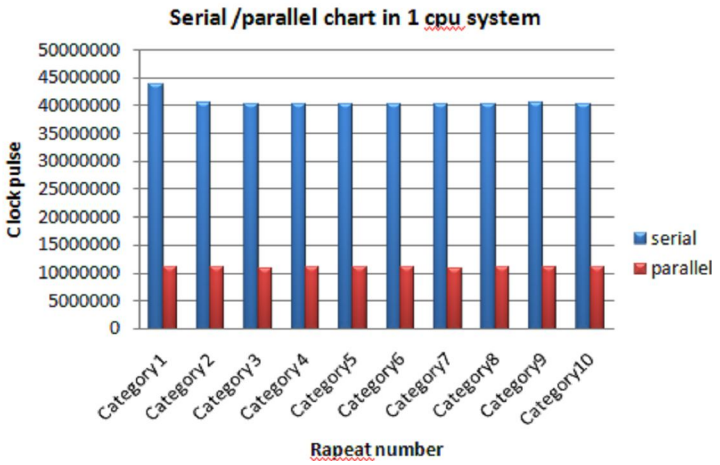

Diagram1*.comparison of series and parallel execution in single core system with ten iteration*

Parallel execution time optimized four times higher than series execution.

In ten cores, I experience ten iteration that I showed them in the diagram but in two cores and four cores we just mention the average in diagram.

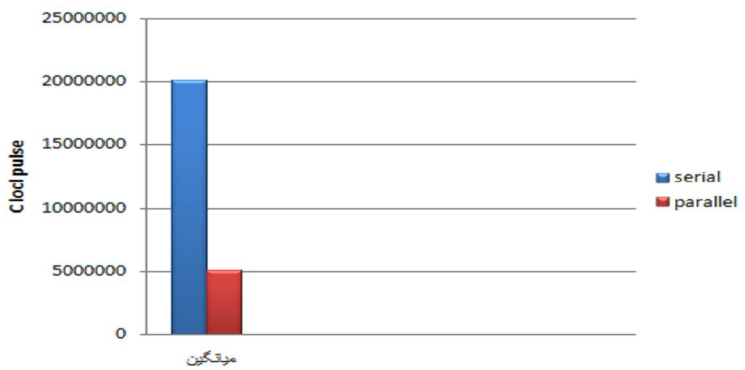

Diagram2*.comparison of series and parallel execution in two core system(average).*

As observed, execution time of two processors system is half of the single processor

5.2.Comparison table of given algorithm and series mode

In this step, results of intelligent expert system execution time on single core, double core and four cores systems obtained in series and parallel mode and compare them in following table:

System specifications:

Single core:2.4GHZ frequency
Double core: 2.6. GHZ frequency
Four core: 2.4 GHZ frequency

Table 1*. comparison of serial and parallel execution time in intelligent expert system.*

	Single core computer	Double core computer	Four cores computer
Serial execution time(clock pulse)	40003154	20000056	13360067
Parallel execution time(clock pulse)	12000312	5000215	3340005
	Single core computer	Double core computer	Four cores computer
Serial execution time(mili/second)	16.66	7.69	5.56
Parallel execution time(mili/second)	5.003	2.01	1.39

6. CONCLUSION

In this research, it was attempted to move forward even one step higher in resource allocation and study of their performance rate by representing a new idea.

Using intelligent factors combination and parallel method and also application of interface we are capable to reach 4 to 8 times higher speed in resource marketing in grid networks, thus, it gain more importance due to this issue that future grid network will own internet and computation and currently, resource allocation is the main challenges in the grid.

REFERENCES

[1] Buyya, R., Abramson, D., Giddy, J., Stockinger, H.(2010): Economic models for resource management and scheduling in gridcomputing , The journal of Concurrency and Computation .

[2]Buyya, R .(2010): Economic-based distributed resource management and scheduling for grid computing, Ph.D Thesis ,School of Computer Science and Software Engineering , Monash University , Melbourne , Australia .

[3]Reed, D. (2010): Grids: The teragrid and beyond, IEEE Computer, 36.

[4] Baker, M., Buyya, R.., Laforenza,D.(2009): Grids and Grid technologies for wide-area distributed computing.Softw. Pract. Exper. (in press) (DOI: 10.1002/spe.488).

[5] Mafi,r., Deldari,h..(2008): parallel programming based on portable factors on the grid. Iran journal of computer execution and electric engineering. Year5. No, 2.

[6] Coise Baude, F., Caromel, D., Leyton, M., Quilici, R.(2004): Grid File Transfer During Deployment, Execution, and Retrieval. INRIA Sophia-Antipolis, CNRS, I3S, UNSA, Route des Lucioles, BP 93, F-06902 Sophia-Antipolis Cedex, France