Int. J. Advanced Networking and Applications
Volume: 05, Issue: 03, Pages:1913-1916 (2013) ISSN : 0975-0290

Cost Optimization Technique of Task Allocation in Heterogeneous Distributed Computing System

Faizul Navi Khan
Department of Computer Application, Teerthanker Mahaveer University, Moradabad - 244 001, UP, INDIA
Email: faizulnavi@yahoo.co.in
Kapil Govil
Department of Computer Application, Teerthanker Mahaveer University, Moradabad - 244 001, UP, INDIA
Email: drkaplilgovil@gmail.com

---ABSTRACT---
A Distributed Computing System (DCS) is a network of workstations, personal computer and /or other computing systems. Such system may be heterogeneous in the sense that the computing nodes may have different speeds and memory capacities. A DCS accepts tasks from users and executes different modules of these tasks on various nodes of the system. Task allocation in a DCS is a common problem and a good number of task allocation algorithms have been proposed in the literature. In such environment an application runs in a DCS can be accessible on every node present within the DCS. In such cases if number of tasks is less than or equal to available processors in the DCS, we can assign these task without any trouble. But this allocation becomes complicated when numbers of tasks are greater than the numbers of processors. The problem of task allocation for processing of 'm' tasks to 'n' processors (m>n) in a DCS is addressed here through a new modified tasks allocation technique. The model, presented in this paper allocates the tasks to the processor of different processing capacity to increase the performance of the DCS. The technique presented in this paper is based on the consideration of processing cost of the task to the processors. We have tried a new technique to assign all the tasks as per the required availability of processors and their processing capacity so that all the tasks of application get execute in the DCS.

Keywords - **Distributed Computing System, Optimization Technique, Performance, Processing Cost, Task allocation.**

Date of Submission: November 21, 2013 Date of Acceptance : January 02,2014
----

1. INTRODUCTION

Distributed Computing System (DCS) aims at achieving higher execution speed than the one obtainable with uniprocessor system by exploiting the collaboration of multiple computing nodes interconnected in some fashion. The best possible speed up will obviously be obtained if the various partitions of the given computational task can run independently in parallel. In DCS a single large problem is broken into multiple small problems and processed the smaller part and combines the solution into one solution for the problem. The purpose of the DCS is to coordinate the use of shared resources or provide communication services to the users. In DCS several set of processors handles multiple tasks and ensure the execution of all the requested tasks modules. A DCS can be seen as virtual uniprocessor. In the present research paper, we have taken a domain of DCS where there are 'n' heterogeneous processors and 'm' tasks (where m>n). This scenario looks like a FIFO (First in First out) ordered queue. When first most tasks are assigned to the processor, remaining tasks have to wait until first task has been completed. As given in figure 1.

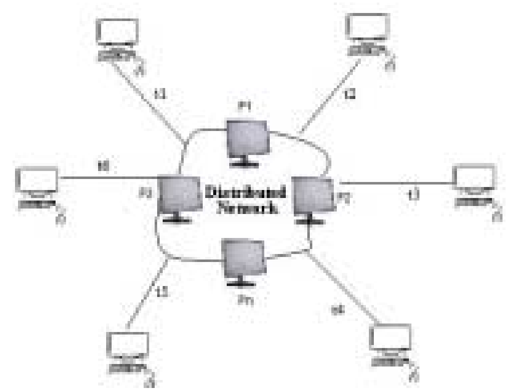

Figure 1: Allocation problem in Distributed Computing System

As per the presented scenario initially all the available processor will be assigned with one task for each processor and remaining tasks will be on hold until the allocated processor will be free. To solve such task allocation problem in DCS, will have to make such arrangement where more than one task can be assigned to a single processor in order to achieve minimum execution cost. Some of the task allocation schemes have been reported in the literature, such as Optimal execution cost [1,2,5,7,8,9,12,14], Task Allocation Model [6,10,11,20], Task Allocation for maximum reliability [15,16,17,19], Task Scheduling [4,13], Decentralized management of bi-modal network [3] and Disaster recovery mechanism using widely distributed

Volume: 05, Issue: 03, Pages:1913-1916 (2013) ISSN : 0975-0290

networking [18]. In this research paper we have taken the problem of task allocation in DCS based on execution cost. Here we have proposed a new task allocation scheme to get minimum execution cost by allocating task. And overall allocation should also be balanced that avoid the situation of overloading to make the proper utilization of processors of the DCS.

2. NOTATIONS

p	Processor
t	Task
n	Number of Processors
m	Number of Tasks
TCM	Task Cost Matrix
PCM	Processing Task Cost Matrix
TPCM	Temporary Processing Cost Matrix
STPCM	Sum of Temporary Processing Cost Matrix

3. OBJECTIVE

The objective of this research is to minimize the overall processing cost for a Distributed Computing System (DCS) through optimally assigning the tasks of various processors in heterogeneous environment, so that the performance of the DCS is to be enhanced. Processing of task is referred here as task along with all its sub tasks are executed. Assignment type would be static in DCS that will also insure the processing of all the modules of task as task modules are more than the numbers of processors of the environment. Here the performance is measured in term of processing cost of the task that have to be get processed on the processors of the environment and it have to be efficiently processed i.e., cost to be minimized.

4. TECHNIQUE

In order to evaluate the overall optimal processing cost of a DCS, we have consider the problem where a set $P = \{p_1, p_2, p_3, \ldots\ldots\ldots p_n\}$ of 'n' processors with different configuration and a set $T = \{t_1, t_2, t_3, \ldots\ldots\ldots t_m\}$ of 'm' tasks, where m>n. processing cost are known for all task modules for every processor and will be arrange the processing cost for each task for different processor in a Processing Task Cost Matrix (PCM) of order (n*m). After arranging the task modules we will break PCM Matrix into another frame called Temporary Processing Cost Matrix TPCM [3][3] and will also initialize the processing load for all processors by zero. In the next step, will form all possible combination of elements column wise by using TPCM (,). And will find the total number of possible combination 27 and will also calculate the sum of all possible combination and store in another matrix Sum of Temporary Processing Cost Matrix STPCM(,). After calculate the sum of all possible combination in the matrix will find minimum value of sum and will compare processing load on within the available processor if task match with the appropriate processor where the processing load is also minimum then assign the task otherwise search for next minimum value of sum and the task will get assign, as per the allocation scheme these steps will be repeated until all the task module will assign to the processors in DCS.

5. ALGORITHM

1. Start Algorithm
2. Read the number of task in m
3. Read the number of processor in n
4. Store task and Processing Cost into Matrix PCM(,) n x m of order
5. While (All task != Assigned)

{

i. Consider n tasks in sequence and store in TPCM(,) in order of n x m.

ii. Create all possible combination and calculate the sum for each combination and store values in STPCM().

iii. Search the minimum value of sum and draw Combination Cost Matrix (CCM)

iv. Check the processing load of matching processor for eligible task (with minimum processing cost).

v. If it is minimum then the task will get assign otherwise will go to step iii and search next minimum value to task get assigned.

}

6. State the results
7. End of algorithm

6. IMPLEMENTATION

Here in this research paper, we have taken Distributed Computing System (DCS) which consist a set P of 3 processors $\{p_1, p_2, p_3,\}$ with different configuration, and a set T of 6 tasks $\{t_1, t_2, t_3, t_4, t_5, t_6\}$. Two imaginary tasks are also added to complete the frame with 0 processing cost (t_7 and t_8). It is shown in the Table 1. The processing cost of each task varies processor vise and known and mentioned in the processing cost matrix namely PCM of order 3 x 8.

Table 1: Processing Cost Matrix

PCM[3X8] =

	t_1	t_2	t_3	t_4
P_1	1.522	1.256	2.025	1.523
P_2	2.584	1.866	2.015	0.985
P_3	0.955	1.658	2.001	1.221

	t_5	t_6	t_7	t_8
P1	1.352	1.555	0	0
P2	2.034	1.235	0	0
P3	1.944	2.009	0	0

To accomplish task allocation process in optimize way, we take first three task to complete first frame and will create a temporary matrix TPCM of [3][3] as follows:

Table 2: Temporary Processing Cost Matrix

TPCM[,] =

	t_1	t_2	t_3
P_1	1.522	1.256	2.025
P_2	2.584	1.866	2.015
P_3	0.955	1.658	2.001

Here we will make all the possible combinations of elements and find out the smallest sum of possible combinations.
For the first frame the combination, we have as follows:

Table 3: Sum of Temporary Cost Matrix

	t_1	t_2	t_3
P_1	-	1.256	-
P_2	-	1.866	-
P_3	0.955	-	-

Here t_1 will get assign to p_3 processor. Allocation table 1 will be as follows for the first frame:

Allocation Table 1

Processor	Task	Processing Cost
P_3	t_1	0.955

By the following the same procedure for the rest frames, we find following allocation table:

Allocation Table 2

Processor	Task	Processing Cost
P_1	**t_2 * t_5**	**1.256 + 1.352 = 2.608**
P_2	t_3 * t_4	2.015 + 0.985 = 3.000
P_3	t_1 * t_6	0.955 + 2.009 = 2.964
Total Processing Cost		8.572

7. CONCLUSION

In this research paper we have taken a problem where m number of tasks needs to assign n number of processor where m is always greater than n. We did task allocation in optimize way in terms of processing cost in a Distributed Computing System (DCS). The technique introduced in research paper applied on several sets of input data and that verified the objective of get minimum processing cost for given tasks for their execution. Here we measured the performance in terms of processing cost of the tasks that has been processed by the processor of the network. The outcome of the given example as mentioned below here:

Resultant Allocation Table

Processor	Task	Processing Cost
P_1	**t_2 * t_5**	**2.608**
P_2	t_3 * t_4	3.000
P_3	t_1 * t_6	2.964
Total Processing Cost		8.572

Figure 2 shows the final task allocation as mentioned in allocation table:

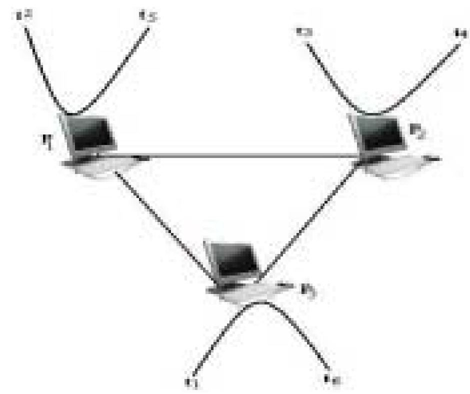

Figure 2: Final task allocation

We have also drawn graphical representation of stated outcome of the given input as mentioned below:

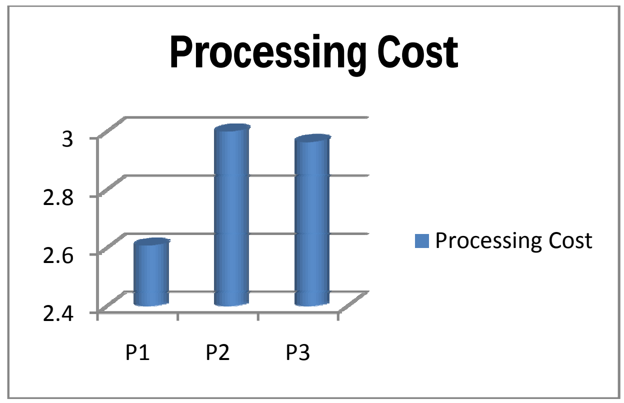

Figure 3: Processing cost for available processors in Distributed Computing Network

REFERENCES

[1] Anju Khandelwal, "Optimal Execution Cost of Distributed System through Clustering", International Journal of Engineering Science and Technology, Vol: 3, issue: 3, pp: 2320-2328, 2011.

[2] Ahmed Younes, Hamed, "Task Allocation for Minimizing Cost of Distributed Computing Systems Using Genetic Algorithms", International Journal of Advanced Research in Computer Science and Software Engineering, vol: 2, issue: 9, pp: 1202-1209, 2012.

[3] Asaduzzaman Shah, Maheswaran Muthucumaru, "Decentralized management of bi-modal network resources in a distributed stream processing platform", Journal of Parallel and Distributed Computing, vol: 71, issue 6, pp: 774-787, 2011.

[4] Boeres, Cristina1, Rebello, Vinod E.F, "A versatile cost modeling approach for multicomputer task scheduling" , journal of Parallel Computing, vol: 25, issue :1 , pp: 63-86, 1999.

[5] Bo yang, huajun hu, suchang guo, "Cost-oriented task allocation and hardware redundancy policies in heterogeneous distributed computing systems considering software reliability", Journal Computers and Industrial Engineering, vol: 56, issue: 4, pp: 1687-1696, 2009.

[6] G.sagar, anil K, sarj E, “Task allocation model for distributed systems”, International Journal of Systems Science, vol: 22, issue: 9, pp: 1671-1678, 1991.

[7] Hsieh, Chung-Chi, Hsieh, Yi-Che, “Reliability and cost optimization in distributed computing systems”, journal of Computers & Operations Research, vol: 30, issue: 8, pp: 1103-1119, 2003.

[8] Ioannis A Moschakis,Helen D Karatza , “Evaluation of gang scheduling performance and cost in a cloud computing system”, Journal of Supercomputing , vol: 59, issue: 2, pp: 975-992 , 2012.

[9] Keren A, Barak A, “opportunity cost algorithms for reduction of I/O and inter process communication overhead in a computing cluster”, IEEE transaction on parallel and distributed systems, vol: 14, issue: 1, pp: 39-50, 2003.

[10] Manisha Sharma, Harendra Kumar, Deepak Garg, “An Optimal Task Allocation Model through Clustering with Inter-Processor Distances in Heterogeneous Distributed Computing Systems”, International Journal of Soft Computing and Engineering, vol: 2, issue: 1, pp: 50-55, 2012.

[11] Mostapha zbakh, said el hajji, “Task allocation problem as a non cooperative game”, Journal of Theoretical and Applied Information Technology, vol: 16, issue: 2, pp: 110-115, 2010.

[12] Najjar Faïza, Slimani, Yahya, “Extension of the one-shot semi join strategy to minimize data transmission cost in distributed query processing”, journal of Information Sciences, vol: 114, issue 1-4, pp: 1-21, 1999.

[13] Nirmeen A. Bahnasawy, Fatma Omara, Magdy A. Koutb, Mervat Mosa , “A new algorithm for static task scheduling for heterogeneous distributed computing system” , International Journal of Information and Communication Technology Research , vol: 1, issue: 1, pp: 10-19, 2011.

[14] Pradeep Kumar Yadav, M.P. Singh, Kuldeep Sharma, “Task Allocation Model for Reliability and Cost optimization in Distributed Computing System”, International Journal of modeling, simulation and scientific computations, vol: 2, issue:2, pp. 1-19,2011.

[15] Qin-Ma Kang, Hong He, Hui-Min Song, and Rong Den, “ Task allocation for maximizing reliability of distributed computing systems using honeybee mating optimization”, Journal of Systems and Software, Volume 83, Issue 11, 2165–2174, 2010

[16] Santhanam Srinivasan, and Niraj, K. , Jha, “Safety and Reliability Driven Task Allocation in Distributed Systems”, IEEE Transactions on Parallel and Distributed Systems, Volume 10, Issue 3, 238-251, 1999

[17] Shatz, S. M. , Wang, J. P. , Goto, M, “Task Allocation for Maximizing Reliability of Distributed Computer Systems”, IEEE Transaction on Computers, Volume 41, Issue 9, 1156-1168, 1992

[18] Ueno Yoichiro, Miyaho, and Suzuki Shuichi, “Disaster recovery mechanism using widely distributed networking and secure metadata handling technology”, Workshop on Use of P2P, GRID and agents for the Development of Content Networks, 45-48, 2009

[19] Vidyarthi, D. P. , and Tripathi, A. K., “Maximizing reliability of distributed computing system with task allocation using simple genetic algorithm”, Journal of System Architecture, Volume 47, Issue 6, 549-554, 2001

[20] Zubair Khan, Ravinder Singh, and Jahangir Alam, “Task allocation using fuzzy inference in parallel and distributed system”, Journal of Information and Operations Management, Volume 3, Issue 2, 322-326, 2012