An Algorithm for Optimized Cost in a Distributed Computing System

Safdar Alam[1], Prof. Ravindar Kumar[2]

[1] P.G Student, Dept. of Electronics & communication, Al-Falah University, Haryana, India
[2] Assistant Professor, Dept. of Electronics & Communication, Al-Falah University, Haryana, India

---***---

Abstract - *A distributed system consists of a collection of autonomous computers, connected through a network which enables computers to coordinate their activities and to share the resources of the system. In distributed computing, a single problem is divided into many parts, and each part is solved by different computers. As long as the computers are networked, they can communicate with each other to solve the problem. DCS consists of multiple software components that are on multiple computers, but run as a single system. The ultimate goal of distributed computing is to maximize performance in a time effective, cost-effective, and reliability effective manner. The problem is addressed of assigning a task to a distributed computing system. The assignment of the modules of tasks is done statically. In this paper We study and formulate an algorithm to solve the problem of static task assignment in DCS,, teach task be assigned to get the more reliable results in lesser cost. The method uses mathematical algorithm by using optimization for optimized cost for task allocation in DCS in MATLAB*

***Keywords*: Distributed Network, Dynamic Allocation, Performance, Residing cost, Reallocation cost**

1.INTRODUCTION

Distributed Computing System (DCS) is a collection of independent computers interconnected by transmission channels that appear to the users of the system as a single computer. Distributed systems are groups of networked computers. The word distributed in terms such as DCS, referred to computer networks where individual computers were physically distributed within some geographical area. The terms are Now days used in a much wider sense. Each node of DCS is equipped with a processor, a local memory,and interfaces. The purpose of the distributed system is to coordinate the use of shared resources or provide communication services to the users. In distributed computing, each processor has its own private memory (distributed memory). The processors in a typical distributed system run concurrently in parallel. The required processing power for task assignment applications in a DCS can not be achieved with a single processor. One approach to this problem is to use (DCS) that concurrently process an application program by using multiple processors. as a means of differentiating between the various components of a project. It can also be understand as usually assigned piece of work to the processor often to be finished within a certain time. A task is a piece of code that is to be executed and task allocation refers to the way that tasks are chosen, assigned, coordinated. Execution time is the time in which single Instruction is executed. Execution cost can be termed as the amount of value of resource used. The execution cost of a task depends on the processor on which it is executed (heterogeneous processors) and the communication between two tasks depends only on whether or not they are assigned to the same processor (homogeneous network). several issues arise such as the minimization of time and cost as well as maximization of system reliability count. By considering that the preference should be given to the idle processor we assign load count as 1 or 0. Now, in each table we will do the addition of each row and will also take the average of each row on the basis of sum of each row. Now, we will subtract the values from average. Negative and zero values will not be considered. For time and cost minimum value will be allocated and for reliability maximum value will be considered. Now the tasks can be allocated for getting the optimized results in terms of cost, also E cost can be calculated. The function for obtaining the overall assignment time, cost and reliability is as follows-Distributed computing is a field of computer science that studies distributed systems. A distributed system is a model in which components located on networked computers communicate and coordinate their actions by passing messages. The components interact with each other in order to achieve a common goal. Three significant characteristics of distributed systems are: concurrency of components, lack of a global clock, and independent failure of components. Examples of distributed system from systems to peer-to-peer applications.

2. Objective

The objective of this research is to find out the optimized cost in Distributed Computing System (DCS) for a task allocation problem or develop a task allocation model that can minimize the overall system cost with the dynamic re-allocation approach. task execution must be completely done before another task takes control of the processor, and the processor environment is homogeneous. This means that the processors have same speeds or processing capabilities.This study offers a mathematical model that allocates the tasks dynamically as tasks executes in various phases. During the particular task execution rest of other task are residing in the particular phase. Execution cost for each phase [EC], inter task communication cost [ITCC], residence cost [RC] of each task on different processors and relocation cost [REC] for each task are considered to design a dynamic tasks allocation model. To achieve cost optimization in DCS allocation method finds an allocation with minimum allocation cost.

3. Technique

This research considers a distributed computing system consisting of a set T = {t1, t2, t3, t4,...tm} of m tasks to be allocated on a set P = {p1, p2, p3, ...pn} of n processors divided into k phases with criteria tasks m are more than the number of processors n (m>n). Execution cost for phase wise of each processor is given in the form of Execution Cost Matrix ECM(,,) of order k x m x n. The Residing Cost for residing the unexecuted tasks on the processor is mentioned in Residing Cost Matrix RCM(,,) of order k x m x n. The Inter Task Communication Cost between executing and non-executing tasks are also considered and is mentioned in the Inter Task Communication Cost Matrix ITCCM(,) of order m x k and during the processing a task is re-allocate from one processor to another processor then it also obtained some cost i.e. reallocation cost and it is given in the Reallocation Cost.

Matrix RECM(,) of order m x k. To calculate ERCM(,) for each phase sum up the values of ECM(,,) and RCM(,,). Compute the average of each row of ERCM(,) and arrange the values in increasing order in AVG_ROW asc (). Now it selects first n number tasks from AVG_ROW asc() and store them in ERCM_I(,) and remaining n number of tasks in ERCM_II by partitioning ERCM(,) into two sub problems. Follow the same process for next n or less than n and solve them using assignment method. Evaluate the Execution Cost, Communication Cost and Reallocation Cost. Follow the same process for all phases and at the end calculate the value of Execution Cost, Communication Cost and Reallocation Cost to obtain the phase wise total execution cost. Calculate the sum of optimal cost of each phase to evaluate the overall optimal cost of distributed computing system.

4. Proposed Method

Step 1: Start Algorithm
Step 2: Input number of tasks as m
Step 3: input the number of processors in n
Step 4: Input the number of phases in k
Step 5: input the Execution Cost Matrix ECM(,,) of order k xm x n
Step 6: input the Residing Cost Matrix RCM(,,) of order k x mx n
Step 7: input the Inter Task Communication Cost Matrix ITCCM(,) of order m x k
Step 8: input the Reallocation Cost Matrix RECM(,,) of order m x k
Step 9: For I = 1 to m
i. K = I:
ii. Add the values for ECM(,,) and RCM(,,) and store the results in ERCM(,,)
iii. Store the average of each row of ERCM(,,) and store it in AVG_ROW()
iv. sort AVG_ROW() in ascending and store the results in AVG_Row asc()
v. While (All tasks of AVG_Rowasc() !=SELECTED)
{
a. Make partition of ERCM(,,) for n tasks, store it in ERCM_I(,) and ERCM_II(,)
b. Apply assignment method on ERCM_I(,) and ERCM_II(,)
}
v. Compute Execution Cost (EC),Inter Task Communication Cost (ITCC) and Reallocation Cost (RC)
vi. Total Cost = EC + ITCC + RC
I = I + 1
10. Applying evolutionary optimization technique
vii. Optimal Cost = (Total Cost)
Step 10: State Results

5. Implementation

This research considers a distributed computing system which is made up of four tasks {t1, t2, t3, t4} to be allocated on two processors {p1, p2} in five phases. The phase wise execution cost of individual processor is known in the form of Execution Cost Matrix ECM(,,) of order k x m x n where k is the number of phases, m is the number of tasks and n is the number of processors. Residing costs for the remaining tasks, except for the executing task, on each processor is also known and mentioned in Residing Cost Matrix (,,) or order k x m x n.Inter Task Communication Cost between the executing task and all other task if they are on different processors also taken into the consideration and mentioned in Inter Task Communication Cost Matrix ITCCM(,) or order m x k. During the execution an allocated task is shifted from one processor to another processor during the next phase then some cost is incurred in reassignment process at the end of each phase andit is known as reallocation cost. Reallocation cost for the given example is also known and it is mentioned in Reallocation Cost Matrix RECM(,) of order m x k.

5.1 Example

This research paper considers a distributed network system which is made up of four tasks {t1, t2, t3, t4} to be allocated on two processors {p1, p2} in five phases. The phase wise efficiency of individual processor is known in the form of Execution Cost Matrix ECM(,,) of order k x m x n where k is the number of phases, m is the number of tasks and n is the number of processors. The ECM(,,) is as mentioned here:
Residing costs for the remaining tasks, except for the executing task, on each processor is also known and mentioned in Residing Cost Matrix (,,) or order k x m x n. RCM(,,) mentioned here:

ECM (,,) =

Phase	Task	Execution Cost p_1	p_2
1	t_1	5	4
	t_2	-	-
	t_3	-	-
	t_4	-	-
2	t_1	-	-
	t_2	7	6
	t_3	-	-
	t_4	-	-
3	t_1	-	-
	t_2	-	-
	t_3	3	5
	t_4	-	-
4	t_1	-	-
	t_2	-	-
	t_3	-	-
	t_4	4	#
5	t_1	5	6
	t_2	-	-
	t_3	-	-
	t_4	-	-

Inter Task Communication Cost between the executing task and all other task if they are on different processors also taken into the consideration and mentioned in Inter Task Communication Cost Matrix ITCCM(,) or order m x k. ITCCM is mentioned in form of Table 1:

Table 1: Inter Task Communication Matrix

ITCC(,)=	Tasks ↓	Phases 1	2	3	4	5
	t_1	-	4	5	3	-
	t_2	2	-	4	5	2
	t_3	5	3	-	6	3
	t_4	2	3	4	-	0

During the execution an allocated task is shifted from one processor to another processor during the next phase then some cost is incurred in reassignment process at the end of each phase and it is known as reallocation cost. Reallocation cost for the given example is also known and it is mentioned in Reallocation Cost Matrix RECM(,) of order m x k. RECM(,) is shown in Table 2:

RCM (,,) =

Phase	Task	Residing Cost p_1	p_2
1	t_1	-	-
	t_2	2	3
	t_3	3	2
	t_4	4	3
2	t_1	2	3
	t_2	-	-
	t_3	3	4
	t_4	2	5
3	t_1	4	2
	t_2	3	4
	t_3	-	-
	t_4	4	2
4	t_1	2	4
	t_2	3	2
	t_3	2	3
	t_4	-	-
5	t_1	-	-
	t_2	3	2
	t_3	2	3
	t_4	2	4

Table 2: Reallocation Cost Matrix

RECM(,)=	Tasks ↓	Phases 1	2	3	4	5
	t_1	2	2	5	3	-
	t_2	3	3	4	4	-
	t_3	4	3	3	3	-
	t_4	2	4	2	2	-

As per the execution table (ECM), task t_1 will execute in phase 1 while remaining tasks i.e. t_2, t_3 and t_4 will be treat as a residing tasks. On the sum up of ECM(,,) and RCM(,,) will drive another matrix as ERCM:

ERCM (,)=

	p_1	p_2
t_1	5	4
t_2	2	3
t_3	3	2
t_4	4	3

On evaluating the average of each row of ERCM(,) an linear array avg_row() is obtained here:

avg_row() =

t1	t2	t3	t4
4.5	2.5	2.5	3.5

On arranging the values of avg_row() in ascending order a new linear array avg_row_asc() linear is formed:

avg_row_asc() =

t2	t3	t4	t1
2.5	2.5	3.5	4.5

In order to get optimal assignment, allocation technique will divide ERCM(,) by selecting first two tasks from sum_row_asc() and store the values in ERCM_I(,) and last two tasks into ERCM_II respectively:

ERCM_I(,)=

	p_1	p_2
t_2	2	3
t_3	3	2

ERCM_II(,)=

	p_1	p_2
t_4	4	3
t_1	5	4

On derived two matrices i.e. ERCM_I and ERCM_II apply assignment method to allocate the tasks and the allocation and their costs is present in Table 3.

Phase	Task	Processor	Assigned Task	Execution Cost (EC)	Communication Cost (CC)	Reallocation Cost (RC)	Phase-wise Total Cost (EC + CC + RC)
1	t_1	p_1	$t_2 * t_4$	6	4	0	
		p_2	$t_3 * t_1$	6			

By applying the same process on the remaining phases final allocation is obtained as present in Table 4 for given example.

Table 4: Final Dynamic Allocation Table

Phase	Executing Task	Processor	Assigned Task	Phase-wise Total Cost (EC + CC + RC)
1	t_1	p_1	$t_2 * t_4$	16
		p_2	$t_3 * t_1$	
2	t_2	p_1	$t_1 * t_4$	22
		p_2	$t_3 * t_2$	
3	t_3	p_1	$t_2 * t_3$	19
		p_2	$t_1 * t_4$	
4	t_4	p_1	$t_4 * t_3$	24
		p_2	$t_2 * t_1$	
5	t_1	p_1	$t_3 * t_4$	15
		p_2	$t_2 * t_1$	
Total Task scheduling cost				96

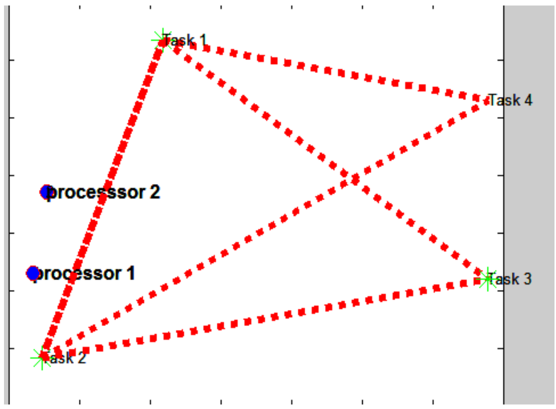

Fig.1: Distributed environment in MATLAB

Overall results also evaluated and compared with past algorithm and found to be very less as shown in Table

Table:5 Comparison Table

	Total cost
Proposed Algorithm	96
Earlier Algorithm	115

6. Conclusion

This research designed a task allocation model with dynamic reallocation technique for execution of tasks in Distributed Computing System (DCS) and provides the optimal solution in order to get optimized costs for task allocation. This allocation model considered the several factors of dynamic environment i.e. execution cost, residing cost, reallocation cost, inter task communication cost and most important execution phases. In dynamic model a tasks completes its execution in various phase so presented dynamic allocation model provide optimal solution phase wise. The presented model is tested in MATLAB platform by creating distributed environment as mentioned in Fig. 1. Optimal cost is calculated for each phase and every task. Communication cost between executing and non-executing task, reallocation cost of task are also added to evaluate final optimal cost of each phase. Phase wise results are generated in MLATLAB for presented algorithm and algorithm [17], results are compared for both algorithms, on comparing phase wise execution cost and total execution cost, it is found presented model shows the better results as mentioned in Table 7

Table 7: Algorithm results derived in MATLAB environment and compare with algorithm [17]

7. References

[1] A.Farinelli, L. Iocchi, D. Nardi, V. A. Ziparo.2005. Task Assignment with dynamic perception and constrained tasks in a Multi-Robot System, Proc. of Intern. Conf. on Robotics and Automation (ICRA'05)
[2] Faizul Navi Khan, KapilGovil. 2014. A TRICKY TASK SCHEDULING TECHNIQUE TO OPTIMIZE TIME COST AND RELIABILITY IN MOBILE COMPUTING ENVIRONMENT, International Journal of Research in Engineering and Technology, Vol. 3 Issue 5, 823-829
[3] Faizul Navi Khan, KapilGovil. 2014. AN EFFICIENT TASK SCHEDULING ALGORITHM TO OPTIMIZE RELIABILITY IN MOBILE COMPUTING, International Journal of Advances in Engineering & Technology, Vol. 7 Issue 2, 635-641
[4] Faizul Navi Khan, KapilGovil. 2014. A Static approach to optimize time cost and reliability in Distributed Processing Environment. International Journal of Scientific & Engineering Research, Vol. 05, Issue 5, 1016-1021
[5] Faizul Navi Khan, KapilGovil. 2013. Cost Optimization Technique of Task Allocation in Heterogeneous Distributed Computing System, Int. J. Advanced Networking and Applications, Vol. 5 Issue 3, 1913-1916
[6] Faizul Navi Khan, Kapil Govil. 2014. Cluster based optimization routing strategy for data communication in

Mobile Computing, International Journal of Computer Applications, Volume 99, Issue 2, 19-24
[7] Faizul Navi Khan, Kapil Govil. 2013. Distributed Task Allocation Scheme for Performance Improvement in Mobile Computing Network, International Journal of Trends in Computer Science, Vol. 2 Issue 3. 809-817
[8] Faizul Navi Khan, Kapil Govil, AlokAgarwal. 2014 Performance enhancement of distributed network system by Phase-wise dynamic task allocation, 2014, International Conference on Parallel, Distributed and Grid Computing (PGDC 2014), IEEE Proceedings, ISBN. 978-1-4799-7681-2
[9] Faizul Navi Khan, KapilGovil. 2013. Static Approach for Efficient Task Allocation in Distributed Environment, International Journal of Computer Applications, Vol. 81 Issue 15, 19-22
[10] Harendra Kumar, M. P. Singh, P. K. Yadav. 2013. Optimal Tasks Assignment for Multiple Heterogeneous Processors with Dynamic Re-assignment, International Journal of Computers & Technology, Vol. 4, No. 2, 528-535
[11] Kapil Govil. 2011. A Smart Algorithm for Dynamic Task Allocation for Distributed Processing Environment, International Journal of Computer Applications, Vol. 28, No. 2, 13-19
[12] M.P, Singh, P.K.Yadav, H. Kumar, B.Agarwal. 2012. Dynamic Tasks Scheduling Model for Performance Evaluation of a Distributed Computing System through Artificial Neural Network, Proceedings of the International Conference on Soft Computing for Problem Solving (SocProS 2011) (Advances in Intelligent and Soft Computing: Published by Springer) Vol.130, 321-331
[13] Manisha Sharma, Harendra Kumar, Deepak Garg. 2012. An Optimal Task Allocation Model through Clustering with Inter-Processor Distances in Heterogeneous International Journal of Computer Applications (0975 – 8887) Volume 122 – No.22, July 2015 35 Distributed Computing Systems, International Journal of Soft Computing and Engineering, Vol. 2 No.1, 50-55
[14] Monika Choudhary, Sateesh Kumar Peddoju. 2012. A Dynamic Optimization Algorithm for Task Scheduling in Cloud Environment, International Journal of Engineering Research and Applications (IJERA), Vol. 2, Issue 3, 2564-2568
[15] N. Beaumont. 2009. Using dynamic programming to determine an optimal strategy in a contract bridge tournament, Journal of the Operational Research Society, Vol 61, Issue 5, 732-739
[16] Martin Grajcar, "Strengths and Weaknesses of Genetic List Scheduling for Heterogeneous Systems", IEEE Computer Society, Proceedings of the Second International Conference on Application of Concurrency to System Design, Page: 123, ISBN: 0-7695-1071-X, IEEE Computer Society Washington, DC, USA, 2001.
[17] Hesam Izakian, Ajith Abraham, Vaclav Snasel, "Comparison of Heuristics for scheduling Independent Tasks on Heterogeneous Distributed Environments", Proceedings of the 2009 International Joint Conference on Computational Sciences and Optimization, Volume 01, Pages: 8-12, 2009, ISBN:978-0-7695-3605-7, IEEE Computer Society Washington, DC, USA.
[18] Yi-Hsuan Lee and Cheng Chen, "A Modified Genetic Algorithm for Task Scheduling in Multiprocessor Systems", Proc. of 6th International Conference Systems and Applications, pp. 382-387, 1999.
[19] Amir Masoud Rahmani and Mojtaba Rezvani, "A Novel Genetic Algorithm for Static Task Scheduling in Distributed Systems", International Journal of Computer Theory and Engineering, Vol. 1, No. 1, April 2009, 1793-8201.
[20] Michael Rinehart, Vida Kianzad and Shuvra S. Bhattacharyya, "A modular Genetic Algorithm for Scheduling Task Graphs", Technical Report UMIACS-TR-2003-66, Institute for Advanced Computer Studies University of Maryland at College Park, June 2003.