A STUDY OF BNP PARALLEL TASK SCHEDULING ALGORITHMS METRIC'S FOR DISTRIBUTED DATABASE SYSTEM

Manik Sharma[1], Dr. Gurdev Singh[2] and Harsimran Kaur[3]

[1] Assistant Professor & Head, Department of Computer Science & Applications,
Sewa Devi S.D. College Tarn Taran, Punjab, India
manik_sharma25@yahoo.com

[2]Professor & Head, Department of Computer Science & Engineering,
Adesh Institute of Engineering & Technology Faridkot, Punjab, India
singh.gndu@gmail.com

[3]Department of Computer Science & Engineering,
Adesh Institute of Engineering & Technology Faridkot, Punjab, India
harsimransamra@gmail.com

ABSTRACT

To solve number of complex scientific problems one must require elevated computation rate comparable to supercomputer. The modernization in latest technologies, communication and information lead to the development of distributed systems and parallel systems as an alternate to Super Computer for solving complex mathematical problems. Parallel processing is a method of executing the multiple tasks alongside on different processors. With the help of parallel processing one is able to solve the complex problems that require huge amount of processing time. In parallel processing or in distributed system task scheduling is one of the major problems. Distributed database system is defined as collection of computer that are connected with one another with the help of some network media over which data and tasks are scheduled for faster execution. The objective of this study is to analyze the various metrics of static (HLFET) and dynamic (DLS) BNP parallel scheduling algorithm in allocating the tasks of distributed database over number of processors. In the whole study the focus will be given on measuring the impact of number of processors on different metrics of performance like makespan, speed up and processor utilization by using HLFET and DLS, BNP task scheduling algorithms.

KEYWORDS

Parallel Processing, Distributed database, HLFET, DLS, Makespan, Speed up.

1. INTRODUCTION

A distributed computing system or parallel systems is defined as the collection of computers (either homogenous or heterogeneous) or workstations. The execution of a program on parallel computer may use different number of processors at different time period during the instruction execution cycle. DDBMS [1][12] consist of single logical database that is decomposed into number of data segment known as fragments; each segment is stored on one or number of sites. There are three major activities [14] in the processing of distributed database system, in the first phase the database is fragmented, in second phase some complex mechanism is used to allocate the database fragment to the different sites and in the third phase the execution of task takes place. It is believed that an effective database fragmentation improves the performance of the database. No doubt fragmentation increases the complexity of physical database design but it significantly impact performance and manageability [2].

Parallel processing [3] is one of the emerging concepts that is used to execute number of tasks on different number of processors at the same time. With the help of parallel processing one is able to solve complex and computation intensive problems in an effective way. Depending upon nature of nodes the parallel processing system can be divided into two categories known as homogenous or heterogeneous parallel system. In homogenous environment the number of processor used for executing the different tasks are similar in capacity and in case of heterogeneous environment the tasks are allocated on various processors of different capacity and speed. Independent of the environment the objective of parallel processing is to improve the execution speed and to minimize the makespan [4][13] of task execution. This is done by using the different precious and competent task scheduling algorithm. The objective of task scheduling algorithm is to allocate the different tasks to different processor so that execution speeds of the task increases and the overall execution time of the task decreases. One of the widespread approaches to decipher task scheduling problem is the use of list scheduling algorithm [5]. List scheduling algorithms are primarily classified as static list scheduling algorithm and dynamic list scheduling algorithm. In this paper the focus is given on one of the important static list scheduling algorithm (HLFET) and dynamic list scheduling algorithm (DLS) by using the concept of BNP (Bounded number of processors) in homogenous environment. BNP task scheduling algorithms are mainly based on the concept of assigned priority [3]. BNP uses b-level and t-level for assigning priority to different nodes for its execution. HLFET [3][4][5][6] (Highest Level First with Estimated Times) is one of the important static list scheduling algorithm that compute the sum of computation cost of call the nodes available in a DAG. It computes the sum by considering the longest path from node to an exit node. The Dynamic Level Scheduling algorithm uses an attribute called dynamic level (DL), which is the difference between the static b-level of a node and its earliest start-time on a processor. The node-processor pair which gives the largest value of DL is selected for scheduling. Performance [7] is one of the important factors of parallel processing that can by measured by using different methodologies like experiments, theoretically, analytically, simulation etc. The various measures of performance are makespan, speed up, processor utilization, efficiency, cost, effort, flexibility, accuracy etc.

2. RESEARCH PROBLEM

2.1 Problem

One of the major problems of distributed database system is the allocation of data and sub query to different sites. The allocation of data should be done in such a way that it minimize the cost and maximize the performance. The objective of the distributed database system is to share available resources in an effective way. In case of Distributed Database System, Data & operation allocation are both closely interrelated & highly dependent on each other. The objective of this study is reduce the make span and improve the performance of whole system by using one of the important static list BNP task scheduling algorithm and one dynamic list scheduling (DLS) algorithm for allocating the fragments of database operation (sub query) to different sites of database system. For static list scheduling Highest Level First with Estimated Time (HLFET) algorithm [6][8] is picked up. HLFET is based on task priority; it uses static level as node priority. The working of complete algorithm is as given below:

Step1: Determine the static b-level of each node.

Step2: Make a ready list in a descending order of static b-level. Initially, the ready list contains only the entry nodes. Ties are broken randomly.

Step 3: Repeat until all nodes are scheduled.

Schedule the first node in the ready list to a processor that allows the earliest execution. Update the ready list by inserting the nodes that are now ready.

The concept of DLS [6][8] is parallel to the one used by the ETF algorithm. DLS algorithm leans to schedule nodes in a descending order of static b-levels at the beginning of a scheduling process but tends to schedule nodes in an ascending order of t-levels near the end of the scheduling process. The working of DLS algorithm is outlined below in clear steps.

Step1: First of all determine the b-level of each node in the graph.

Step2: Initially, the ready list includes only the entry nodes.

Step3: Repeat until all nodes are scheduled.

Calculate the earliest start-time for every ready node on each processor. Hence, compute the DL of every node-processor pair by subtracting the earliest start time from the node's static b-level. Select the node-processor pair that gives the largest DL. Schedule the node to the corresponding processor. Add the newly ready nodes to the ready list.

3. PROBLEM ANALYSIS

3.1 Introduction to DAG

Directed acyclic graph (DAG) is used for analysis of distributed task scheduling. In mathematical terms, DAG [4][5][11] is defined as set of four different parameters called (V) nodes that represents the tasks to be scheduled, (W) computation cost, (E) edges the connect two nodes and (c) communication cost. This study assumes that distributed database is distributed over number of sites by using replication technology of distributed database management system.

3.2 Analysis

Case I: There are three major activities in the processing of distributed database system, in the first phase the database is fragmented, in second phase some complex mechanism is used to allocate the database and operation fragment to the different sites and in the third phase the execution of task takes place. In this case the query of distributed databases is supposed to be divided into ten different segments(sub query) that are to be scheduled over three homogenous processors for effective speed up and reduced make span. The operation fragments (sub queries) are represented by a DAG of ten nodes with randomly selected computation and communication cost [10] [11] as follow:

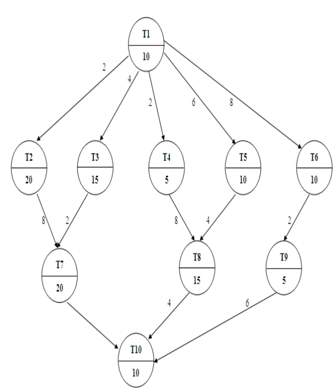

Figure 1: Distributed Database with Ten Segments

For analysis the above said problems one has to calculate one of the important performance metrics known as b-level (bottom level) [9] as given in the following table.

Tasks	Execution Time	Static b-level	t-level	b-level	Dynamic Level
T1	10	60	0	72	60
T2	20	50	12	60	38
T3	15	45	14	49	31
T4	5	30	12	42	18
T5	10	35	16	43	19
T6	10	25	18	33	7
T7	20	30	40	32	-10
T8	15	25	30	29	-5
T9	5	15	30	21	-15
T10	10	10	62	10	-52

Table 1: Analysis of b-level and t-level

Following figures 2(a) & 2(b) shows how the above said tasks are scheduled over three processors with HLFET and DLS task Scheduling algorithms.

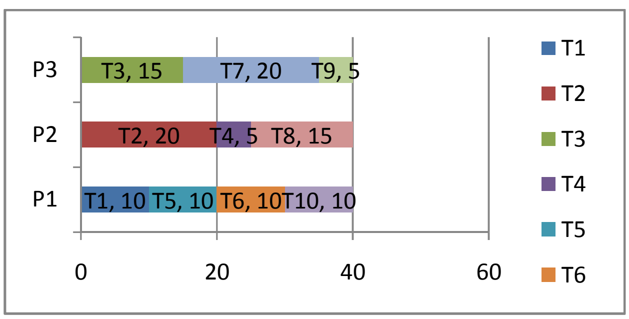

Figure 2(a): Task scheduled with HLFET

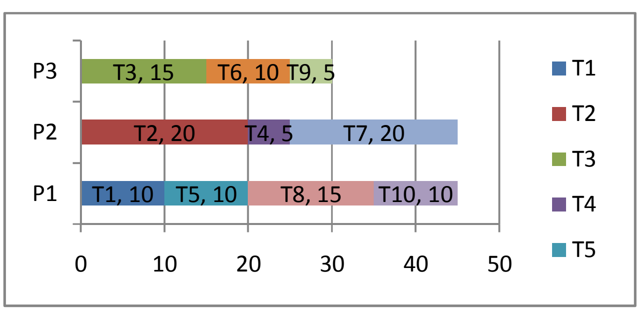

Figure 2(b): Task scheduled with DLS

The following chart 3(a) show the execution time of jobs when scheduled on serial processor and on distributed system with HLFET and DLS Algorithms.

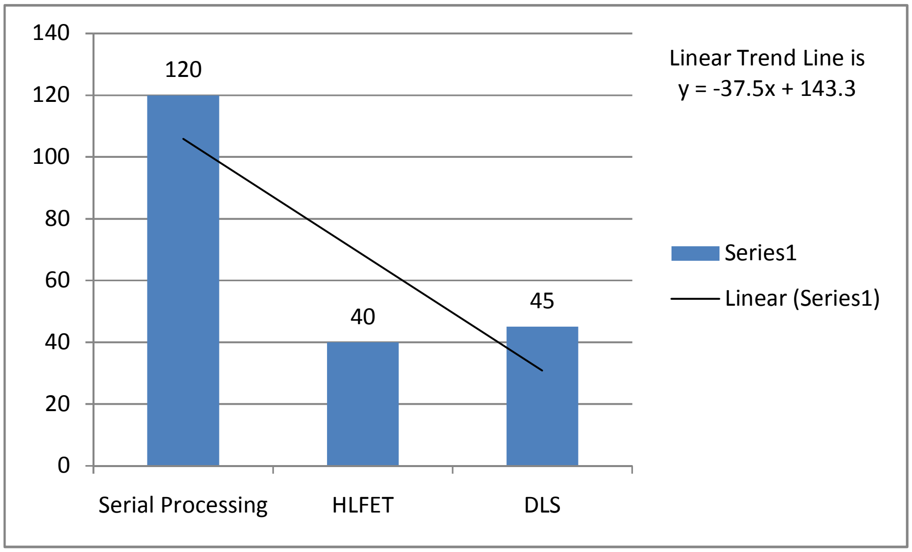

Figure 3(a): Execution Time (Make Span) of Jobs on serial and Distributed System

From the above analysis it is clear the Processor utilization with HLFET and DLS scheduling algorithm is as given below:

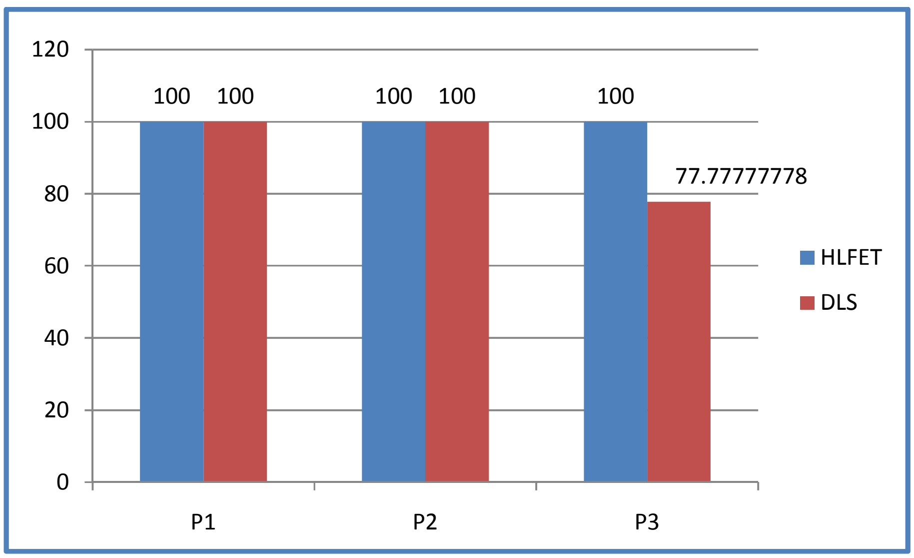

Figure 3(b) Processor Utilization with HLFET and DLS Scheduling Algorithms

Case II: In this case of data allocation in distributed database system in which database is divided into 15 different data segments. The following DAG represented the 15 different data segments that are to be scheduled over three processors.

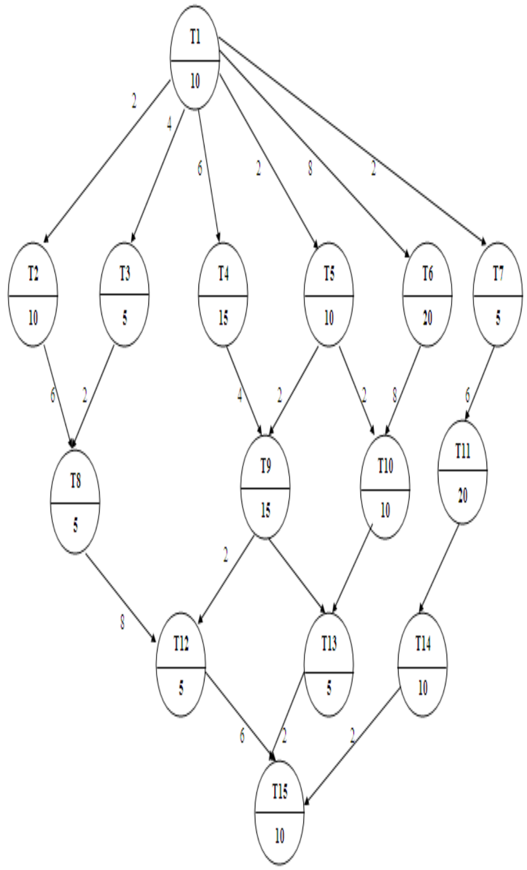

Figure 4: Distributed Database with Fifteen Segments

The following table shows the performance analysis of the above said data.

Tasks	Execution Time	Static b-level	t-level	b-level	ALAP Time	Dynamic Level
T1	10	55	0	79	0	55
T2	10	30	12	50	29	18
T3	5	25	14	41	38	11
T4	15	45	16	53	26	29
T5	10	35	12	45	34	23
T6	20	45	18	61	18	27
T7	5	45	12	55	24	33
T8	5	20	38	34	45	-18

T9	15	30	35	34	45	-5
T10	10	25	46	33	46	-21
T11	20	40	13	44	35	27
T12	5	15	41	21	58	-26
T13	5	15	62	17	62	-47
T14	10	20	35	22	57	-15
T15	10	10	60	10	69	-50

Table2: Analysis of b-level and t-level

Following figures 5(a) & 5(b) shows how above said segments of distributed database are scheduled on three processors with HLFET and DLS task Scheduling algorithms.

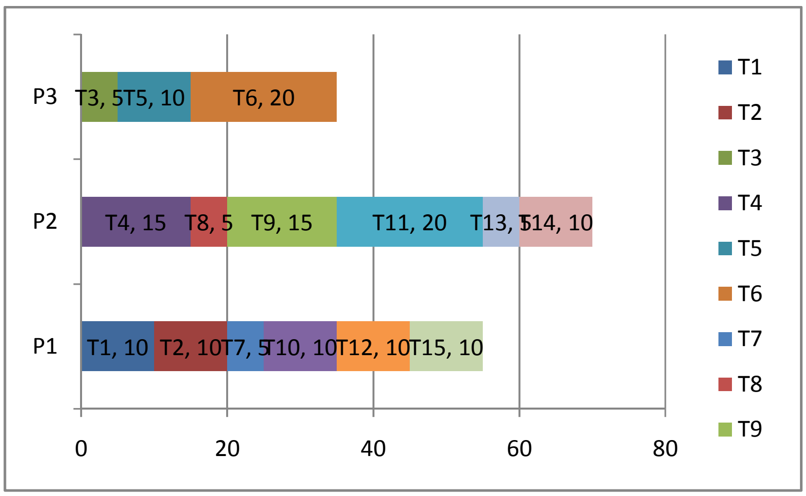

Figure 5(a): Task scheduled with HLFET

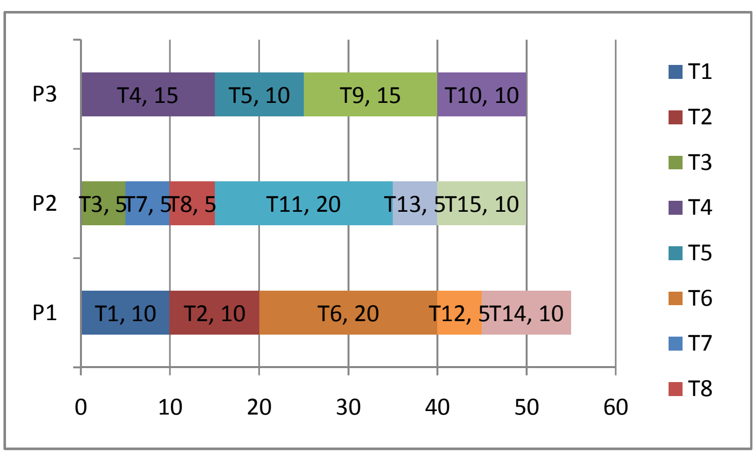

Figure 5(b): Task scheduled with DLS

The following Figure 6(a) shows the comparison of execution time of jobs when executed serially or on distributed database system by using HLFET & DLS task scheduling algorithms.

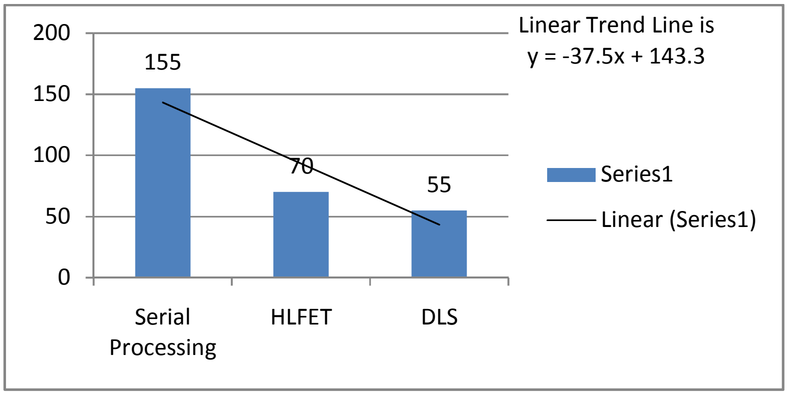

Figure 6(a): Execution time (Make Span) of jobs

From the above chart it is clear that the processor utilization with HLFET and DLS task scheduling algorithm with fifteen segments is as given below:

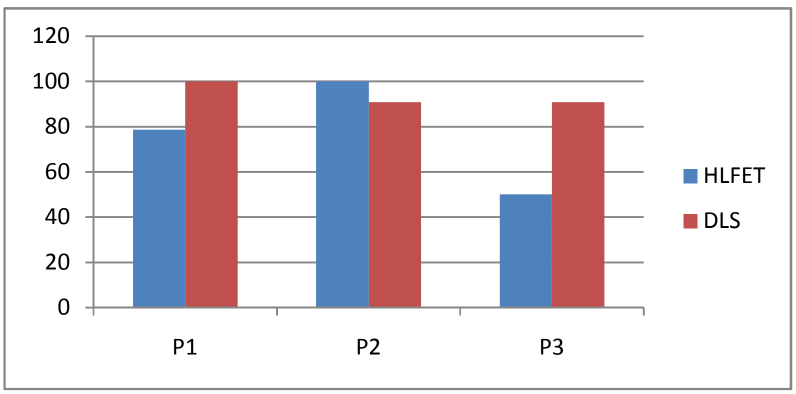

Figure 6(b): Processor Utilization with HLFET and DLS Task Scheduling algorithms

The following figure 7 shows how much improvement in the execution time is observed when the jobs are implemented by using the BNP parallel task scheduling algorithms.

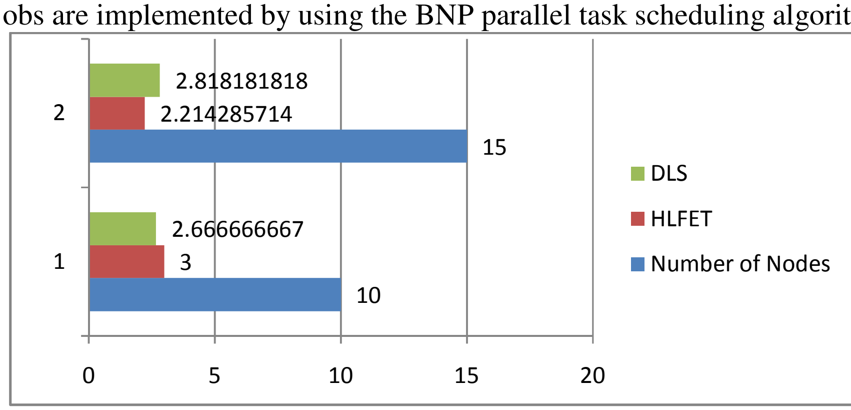

Figure 7: Speed Up of execution time with HLFET and DLS Task Scheduling Algorithms

4. CONCLUSIONS

The study shows that distributed system or parallel system helps in reducing the execution time of jobs in execution. From the above analysis it is concluded that distributed system can faster the jobs execution up to twice, thrice or even more. By scheduling the jobs or task in an effective way the distributed database system is able to process more number of jobs in lesser time as compare to traditional serial system in which jobs are executed serially. It is also concluded that all the processors involved in distributed system are not utilized cent percent. The study has shown that some processors are underutilized.

5. ACKNOWLEDGEMENTS

Authors are highly grateful to Dr. Gurvinder Singh, Associate Professor & Head, DCSE, GNDU, Amritsar for his precious guidance from time to time.

6. REFERENCES

[1] Thomas Connolly, Carolyn Begg, "Database System A Practical Approach to Design, Implementation, and Management" published by Pearson education, Fourth Edition, Page No. 689.Z

[2] Sanjay Agarwal, Vivek Narasayya, Beverly Yang, "Integrating Vertical and Horizontal Partitioning into Automated Physical Database Design", proceeding of SIGMOD, 2004.

[3] Yu-Kwong Kwok, Ishfaq Ahmad, "Static Scheduling Algorithm for Allocating Directed Task Graph to multiprocessors", ACM Computing Surveys, Vol. 31, no. 4, December 1999.

[4] Ishfaq Ahmad, Yu-Kwong Kwok, Min-You Wu, "Analysis, Evaluation, and Comparison of Algorithms for Scheduling Task Graphs on Parallel Processors", Proceedings of the 1996 International Symposium on Parallel Architectures, Algorithms and Networks, IEEE Computer Society Washington, DC, USA.

[5] T. Hagras, J. Janecek, "Static versus Dynamic List-Scheduling Performance Comparison", Acta Polytechnica Vol. 43 No. 6/2003.

[6] Parneet Kaur, Dheerandra Singh, Gurvinder Singh, "Analysis Comparison and Performance Evaluation of BNP Scheduling Algorithm in parallel Processing", International journal of Knowledge engineering.

[7] Amit Chhabra , Gurvinder Singh, "Parametric Identification for comparing performance evaluation techniques in parallel system",

[8] Gurvinder Singh, Kamaljit Kaur, Amit Chhabra, "Heuristics Based Genetic Algorithm for Scheduling Static Tasks in Homogeneous Parallel System", International Journal of Computer Science and Security (IJCSS), Volume (4): Issue (2).

[9] Ishfaq Ahmad, Yu-Kwong Kwok, Min-You Wu, "Performance Comparison of Algorithms for Static Scheduling of DAG to Multiprocessors", ACM Computing Surveys, Vol. 31, no. 4, December 1999.

[10] Yu-Kwong Kwok, Ishfaq Ahmad, "Dynamic Critical-Path Scheduling: An Effective Technique for allocating Task Graphs to Multiprocessors", IEEE Transactions on Parallel and Distributed System, Vol. 7, No. 5.

[11] DI George, G.J. Joyce Mary, "A New DAG Based Task Scheduling Algorithm for multiprocessor System", IJCA Vol. 19, No. 8, April 2011.

[12] M. Tamer Ozsu, Patrick Valduriez, "Principles of Distributed Database Systems", Published by Pearson Education, Second Edition (Sixth Impression), 2009, Page No. [3]

[13] Alaa Ismail El-Nashar, "To Parallelize or Not To Parallelize, Speed Up Issue", International Journal of Distributed and Parallel Systems (IJDPS) Vol.2, No.2, March 2011.

[14] Rajinder Singh Virk, Dr. Gurvinder Singh, "Optimizing Access Strategies for a Distributed Database Design using Genetic Fragmentation", IJCSNS International Journal of Computer Science and Network Security, VOL.11 No.6, June 2011

Authors

Manik Sharma (MCA) is currently working as an Assistant Professor & Head of Computer Department, Sewa Devi S.D College, Tarn Taran, Punjab, India having teaching experience of more than six years. In publishing an author has written books on Computer Networks and Database Management System with reputed publisher.

Dr. Gurdev Singh is currently working as prorfessor and Head, Department of Computer Science and Engineering, Adesh Institute of Engineering and Technology, Faridkot. Author has huge experience in teaching as well as in research. An area of interest of author is Software Engineering, Computer Network and Distributed system. An author has number of research paper to his credits.