# STATIC AND DYNAMIC BNP PARALLEL SCHEDULING ALGORITHIMS FOR DISTRIBUTED DATABASE

Manik Sharma
Assistant Professor & Head, Computer Deptt.
Sewa Devi S.D. College
Tarn Taran
Manik_sharma25@yahoo.com

Smriti
Assistant Professor,
Shiv Shanker Institute of Engg. & Technology
Patti, Tarn Taran
smritimahajan30@gmail.com

## ABSTRACT

Parallel processing is a technique of executing the multiple tasks concurrently on different processors. Parallel processing is used to solve the complex problems that require vast amount of processing time. Task scheduling is one of the major problems of parallel processing. The objective of this study is to analyze the performance of static (HLFET) and dynamic (DLS) BNP parallel scheduling algorithm for 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.

## General Terms

Parallel and Distributed Systems

## Keywords

BNP, HLFET , DLS , Parallel Processing , Distributed database.

## 1. INTRODUCTION

Parallel processing [1] is one of the growing concepts that is used to execute number of tasks on different number of processors at the same time. The concept of parallel processing helps in solving complex and computation intensive problems in competent way. Generally parallel processing environment is categorized as homogenous or heterogeneous environment. 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 [2] 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 [3]. 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 shceduling 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 [1]. BNP uses b-level and t-level for assigning priority to different nodes for its execution. HLFET [1][2][3][4] (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 [5] 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. Problem under Study

### 2.1 Problem

One of the major problems of distributed database system is the allocation of data to different sites. The allocation of data should be done in such a way that it minimize the cost and maximize the performance.

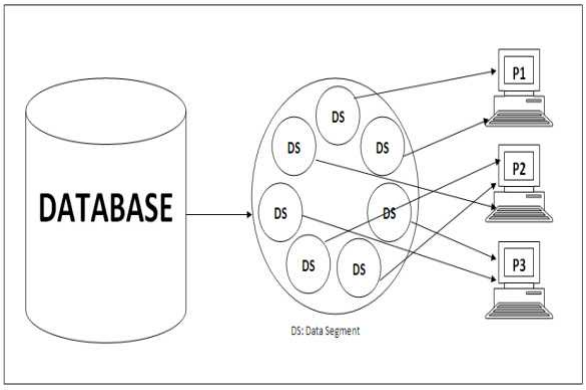


Figure1: Distributed Database

The objective of this study is reduce the maskepan, improve the performance of data allocation and measuring performance of the system by using one of the one static list BNP task scheduling algorithm and one dynamic list sheduling (DLS) algorithm. The Highest Level First with Estimated Time (HLFET) algorithm[4][6] is one of the simplest list-scheduling algorithms. It uses Static level as node priority and is described 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 [4][6] 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: Calculate 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. Results & Discussion

### 3.1 Introduction to DAG

Directed accyclic graph (DAG) is used for analysis of distributed task scheduling. In mathematical terms, DAG [2][3][9] 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.

### 3.2 Analysis

Case I: In this case the distributed databases is supposed to be divided into ten different database segments that are to be scheduled over three homogenous processors for effective speed up and reduced make span. The database instances are represented by a DAG of ten nodes with randomly selected computation and communication cost [8] [9] as follow:

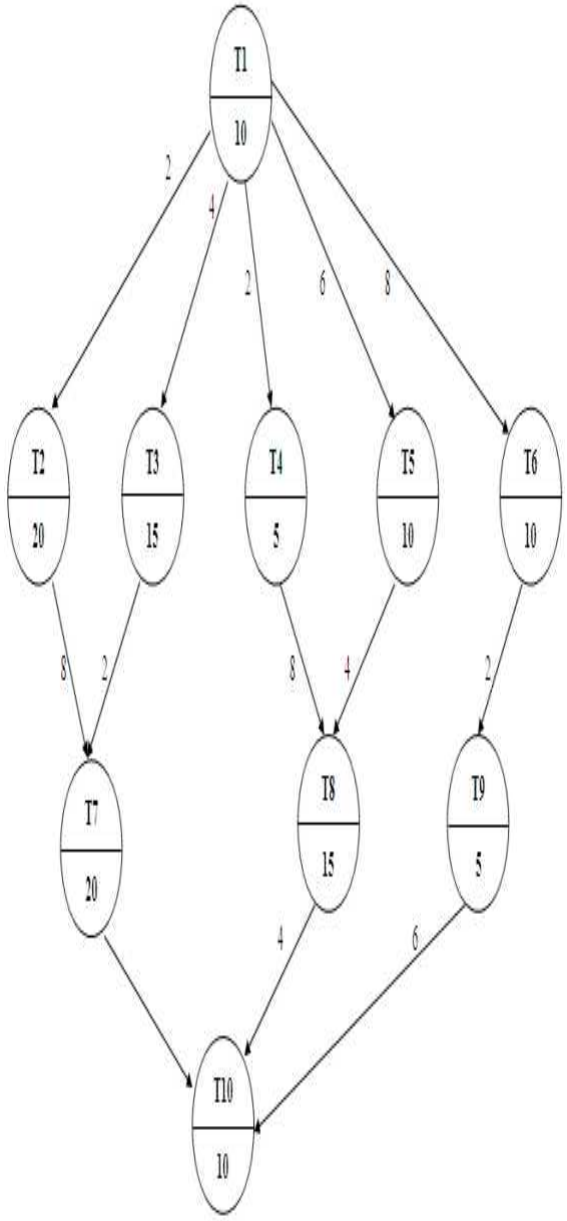


**Figure 2: Distributed Database with Ten Segments**

For analysis the above said problems one has to calculate one of the important performance metrics known as b-level(bottom level) [7] 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 3(a) & 3(b) shows how the above said tasks are scheduled three processor without and with genetic approach.

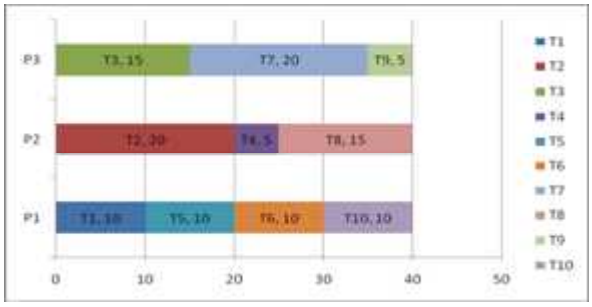


**Figure 3(a): Task scheduled with HLFET**

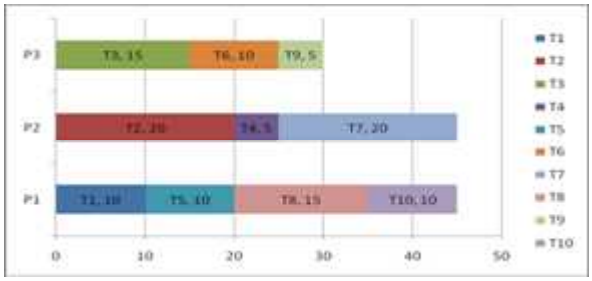


**Figure 3(b): Task scheduled with DLS**

The following chart shows the speed up comparison for execution of distributed database segments with serial processing and parallel processing (HLFET & DLS).

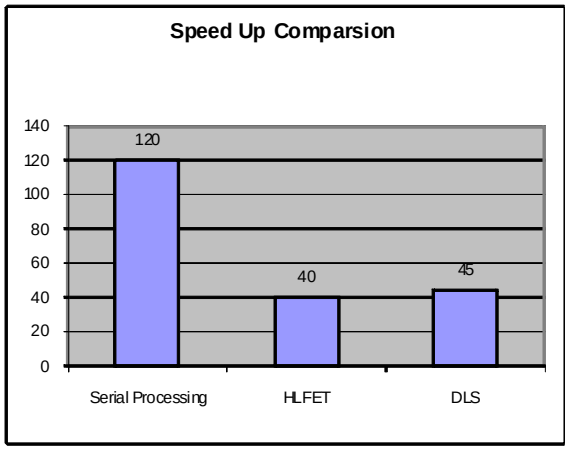


**Figure 3(c): Speed Up**

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.

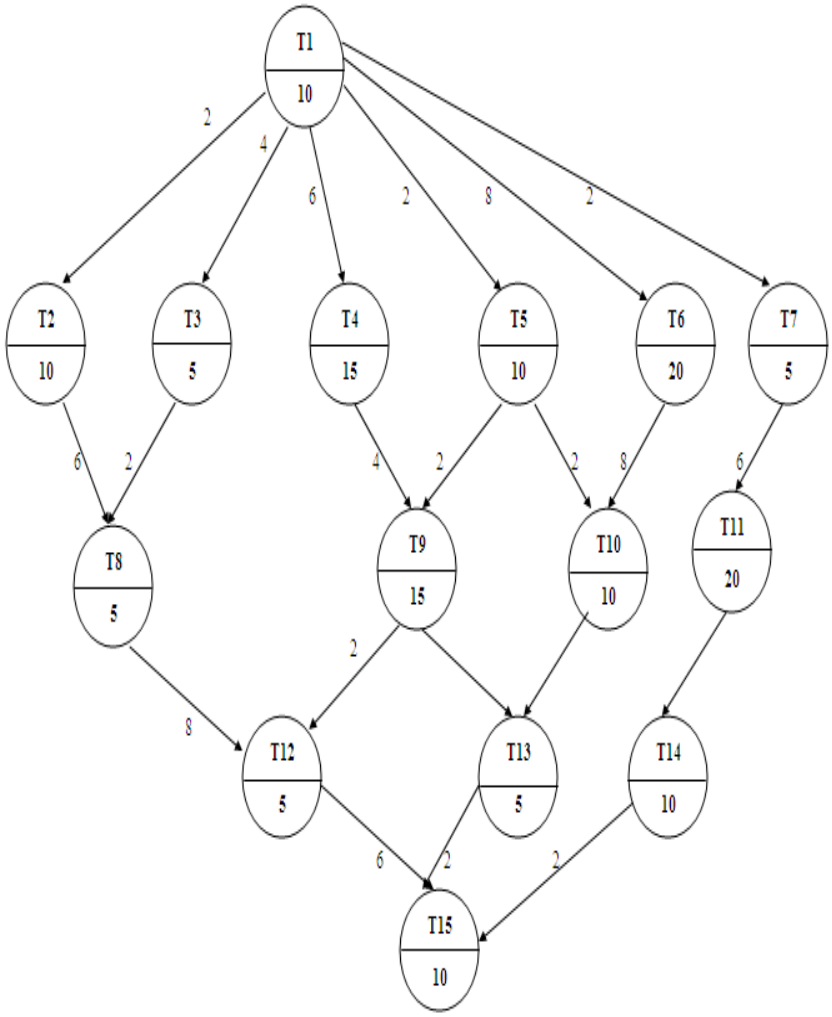


**Figure 4: Distributed Database with Fifteen Segments**

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 without and with genetic approach.

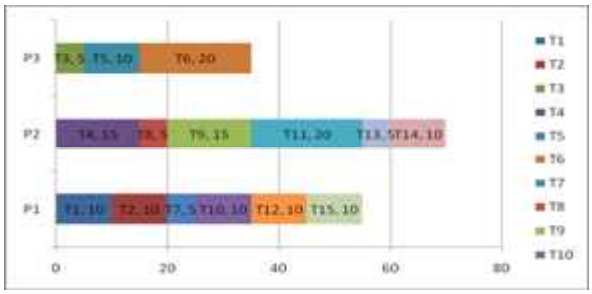


**Figure 5(a): Task scheduled with HLFET**

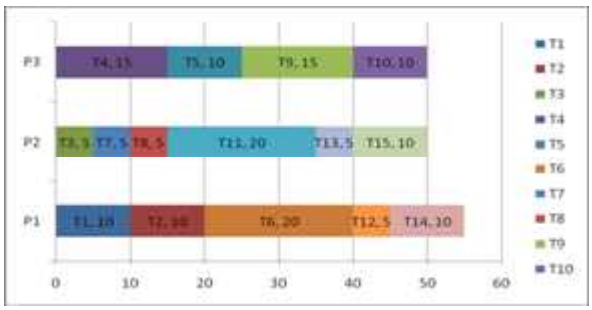


**Figure 5(b): Task scheduled with DLS**

The following chart shows the speed up comparison for execution of distributed database segments with serial processing and parallel processing (HLFET & DLS), when tasks in distributed environment are scheduled over three processors.

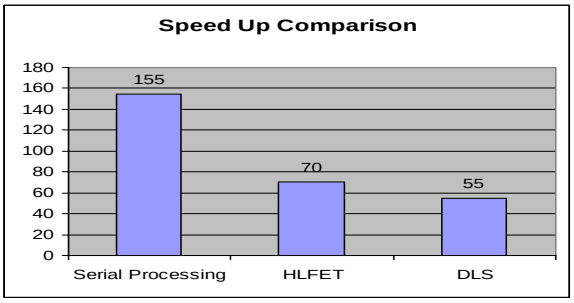


**Figure 5(c): Speed Up**

## 4. CONCLUSIONS

From the above analysis it is clear that BNP static (HLFET) and dynamic (DLS) list scheduling algorithms improves the performance of distributed database by allocation and executing the jobs on different homogenous processors simultaneously as compare to serial processing. From the above analysis it is also concluded that the HLFET and DLS list scheduling algorithm can speed up the exuection by three times or even more under constrained envrionment.

## 5. ACKNOWLEDGEMENT

Authors are highly grateful to Dr. Gurvinder Singh, Associate Professor & Head, DCSE, GNDU, Amritsar and Dr. Gurdev Singh, Professor & Head, DCSE, Adesh Institute of Engineering & Technology, Faridkot for their precious guidance from time to time.

## 6. REFERENCES

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

[2] 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.

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

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

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

[6] 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)

[7] 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.

[8] 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.

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