

# QOS, Comparison of BNP Scheduling Algorithms with Expanded Fuzzy System

Amita Sharma  
M.Tech Scholar (CSE)  
SBBSIET, Jalandhar

Harpreet Kaur  
AP, CSE  
SBBSIET, Jalandhar

**Abstract**—Parallel processing is a field in which different systems run together to save the time of the processing and to increase the performance of the system. It has been also seen that it works somewhat up to the load balancing concept. Previous algorithms like HLFET, MCP, DLS, ETF, have shown that they can reduce the burden of the processor by working simultaneous working system. In our research work, we have combined HLFET, MCP, DLS, ETF with FUZZY logic to check out what effect it makes to the parameters which has been taken from the previous work done like Makespan, SLR, Speedup, Process Utilization. It has been found that the fuzzy logic system works better than the single algorithm.

**Keywords**—Parallel Processing; DAG, BNP; Fuzzy logic; Multiprocessor.

## I. INTRODUCTION

Parallel processing is one of the emerging concepts that is used to execute number of tasks on different number of processors at the same [5] 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 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. Task scheduling algorithms take care of all the above said factors and can be represented by Directed Acyclic Graphs (DAG)[15].

## II. MULTIPROCESSOR SCHEDULING ALGORITHMS [3]

A. *Bounded Number of Processors (BNP) scheduling algorithms:* These algorithms schedule the DAG to a bounded number of processors directly. The processors are assumed to be fully-connected.

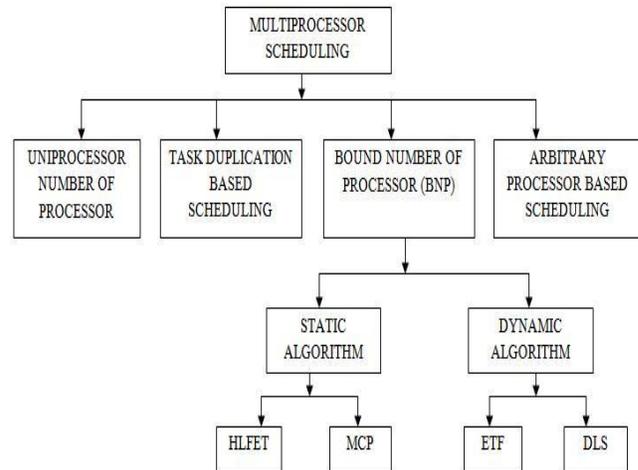


Fig. 1. Multiprocessor scheduling algorithms

B. *Unbounded Number of Clusters (UNC) scheduling algorithms:* These algorithms schedule the DAG to an unbounded number of clusters. The processors [1][26] are assumed to be fully-connected. The technique employed by these algorithms is also called clustering.

C. *Task Duplication Based (TDB) scheduling algorithms:* These algorithms also schedule the DAG to an unbounded number of clusters but employ task duplication technique to further reduce the completion time.

D. *Arbitrary Processor Network (APN) scheduling algorithms:* These algorithms perform scheduling and mapping on the target architectures in which the processors are connected via a network of arbitrary topology. In this paper, BNP scheduling algorithms are discussed and their performance is analyzed.

BNP class of algorithms is categorized into two categories:

**Static Algorithms[2]** : These algorithms use list scheduling approach. Therefore in static algorithms once the task prioritization phase is finished then and only then the processor selection phase begins.

### 1. HLFET Algorithm

It is one of the simplest algorithms. Here the HLFET stands for Highest Level First with Estimated Time.

### Algorithm Steps:

- 1) Calculate the static Level of the nodes in the DAG.
- 2) Insert all the nodes into a list and sort the list according to descending order of Static Level of the nodes.
- 3) While not the end of the list do
  - Remove the node  $n_i$  from the list.
  - Compute the earliest start execution time of  $n_i$  for all the processor present in the system.
  - Map the node  $n_i$  to the processor that has the least early start execution time.

### 2. MCP Algorithm

MCP stands for Modified Critical Path. It uses the Latest Start Time attribute for mapping the nodes to processors.

### Algorithm Steps:

- 1) Calculate the Latest Start Time (LST) of all the nodes in the DAG.
- 2) Insert all the nodes into a list and sort the list according order of Latest Start Time.
- 3) While not the end of the list do
  - Remove the node from the list
  - Compute the earliest start execution time of  $n_i$  for all the processors present in the system.
  - Map the node  $n_i$  to the processor that has the least early start execution time.

**Dynamic Algorithms:** These algorithms also use list scheduling approach. In Dynamic algorithms both the task prioritization phase and processor selection phase goes on side by side.

### 1. ETF Algorithm

ETF stands for Earliest Task First. This algorithm computes the earliest execution start time for all nodes and selects one with lowest value for scheduling. In this algorithm the ready node stands for that node which has all its parents scheduled.

### Algorithm Steps:

- 1) Calculate the Static Level of each node in the DAG.
- 2) In the beginning the ready node list contains only the entry node.
- 3) While the ready node list is not empty do
  - Compute the earliest start time of all the nodes in the ready node list on each processor.
  - Select the node with earliest start time. If two or more nodes have same earliest execution start time values then the node with highest Static Level is selected.
  - Map the selected node to the processor.
  - Add new ready nodes to the ready node list.

### 2. DLS Algorithm

DLS stands for Dynamic Level Scheduling. It uses the Dynamic Level attribute for scheduling the nodes.

### Algorithm Steps:

- 1) Calculate the Static Level of node in DAG.
- 2) In the beginning the ready node list contains only the entry node.
- 3) While the ready node list is not empty do
  - Compute the earliest start time of every node in the ready node list on each processor.
  - Calculate the Dynamic Level of every node in the list
  - Select the node with the largest Dynamic Level.
  - Schedule the node onto the processor.
  - Add new ready nodes to the ready node list.

### III. DAG MODEL:

The DAG is generic model of a parallel program consisting of a set of processes among which there are dependencies. Each process is an indivisible unit of execution, expressed by node. A node has one or more inputs and can have one or more output to various nodes. When all inputs are available, the node is triggered to execute. After its execution, it generates its output. In this model, a set of node ( $n_1, n_2, n_3 \dots \dots n_n$ ) are connected by a set of a directed edges, which are represented by  $(n_i, n_j)$  where  $n_i$  is called the Parent node and  $n_j$  is called the child node. A node without parent is called an Entry node and a without child node called an Exit node. The weight of a node, denoted by  $w(n_i)$ , represents the process execution time of a process. Since [3] each edge corresponds to a message transfer from one process to another, the weight of an edge, denoted by  $c(n_i, n_j)$ , is equal to the message transmission time from node  $n_i$  to  $n_j$ . Thus,  $c(n_i, n_j)$  becomes zero when  $n_i$  and  $n_j$  are scheduled to the same processor because intraprocessor communication time is negligible compared with the inter processor communication time. The node and edge weights are usually obtained by estimations. Some [15] variations in the generic DAG model are described below:

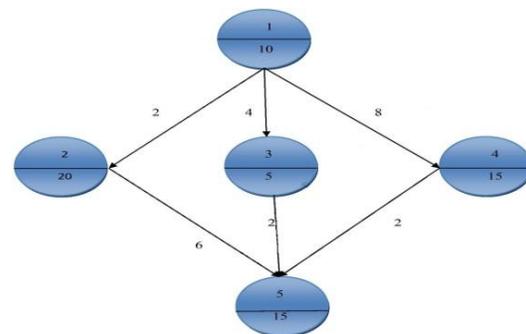


Fig. 2. DAG Model

### IV. FUZZY LOGIC

Fuzzy logic is a rigorous mathematical field, and it provides an effective vehicle for modeling the uncertainty [29] in human reasoning. Fuzzy set is uniquely determined by its membership function (MF), and it is also associated with a linguistically meaningful term. Fuzzy logic provides a systematic tool to incorporate human experience. It is based on three core concepts, namely, fuzzy sets, linguistic variables,

and possibility distributions. The importance of fuzzy logic derives from the fact that most modes of [32] human thinking and especially common sense reasoning are approximate in nature. The essential features of fuzzy logic are as follows:

- In fuzzy logic everything is a matter of degree.
- Any logical system can be fuzzified.
- In fuzzy logic, knowledge is interpreted as a collection of elastic or, equivalently, fuzzy constraint on a collection of variables.
- Inference is viewed as a process of propagation of elastic constraints.

V. EXPERIMENTAL RESULTS AND DISCUSSIONS

The analytical results of the entire algorithms under all three case scenarios are determined. Following parameters are implemented for given scheduling algorithms and results are shown graphically:

**Makespan:** It is defined as the completion time of the algorithm. Lesser the Makespan less time to execute the algorithm more efficient is the algorithm. Makespan is calculated by measuring the finishing time of the exit task by the algorithm.

**Processor Utilization:** In multiprocessor system, processor work in parallel. There can be some scenario when large amount of work is done by one processor and lesser by others, which the work distribution is not proportionate

Processor Utilization (%) = (total time taken of Scheduled tasks/Makespan)\*100

**Speed Up:** It is defined as the ratio of time taken by serial algorithm work to the time taken by the algorithm to perform the same work.

**Speed Up = Time taken by serial algorithm/Time taken by parallel Algorithm**

**Scheduled length Ratio (SLR):** It is defined as the ratio of Makespan of the algorithm to Critical Path values of the DAG. The lesser the values of SLR the more efficient is the algorithm, but The SLR cannot be less than the Critical path values.

**Scheduled length Ratio = Makespan / Critical Path**

**5.2.1 Scenario 1: 5 Task Nodes :** From the graphs shown below ,we can say that results from all the algorithms using 5 Task Nodes the parameters Makespan, SLR, SpeedUp are throughout the same.

Fig. 3. Values of different parameters for 5 nodes.

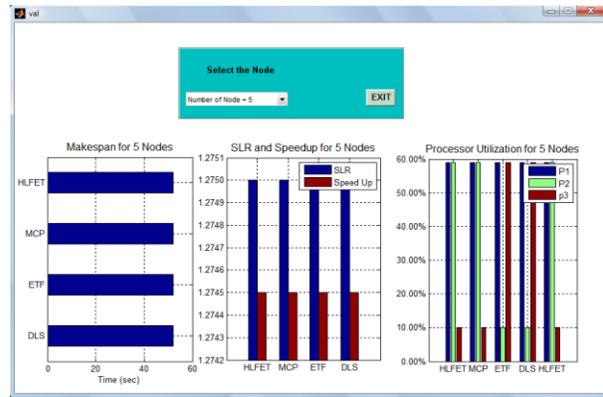


Fig.3.1 Makespan for 5 nodes

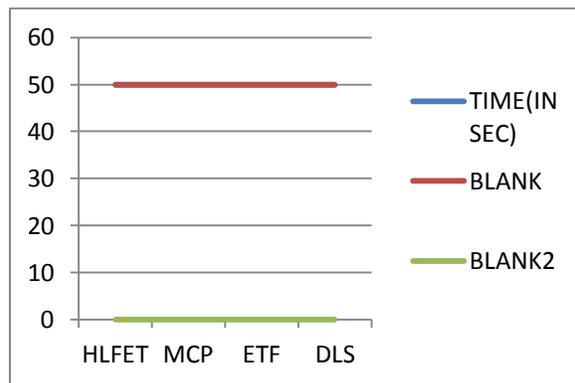


Fig.3.2: SLR & Speed Up for 5 Nodes

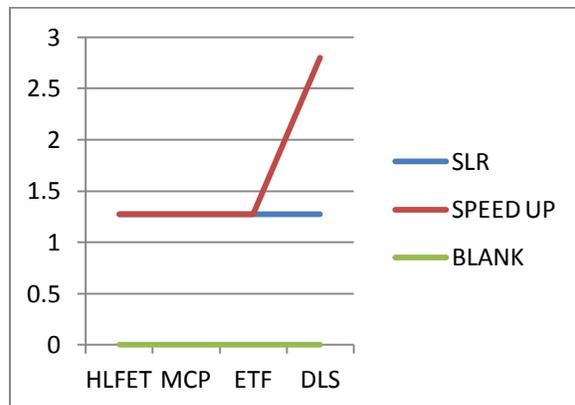
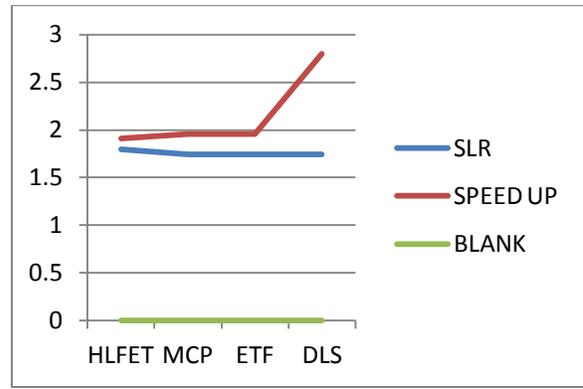
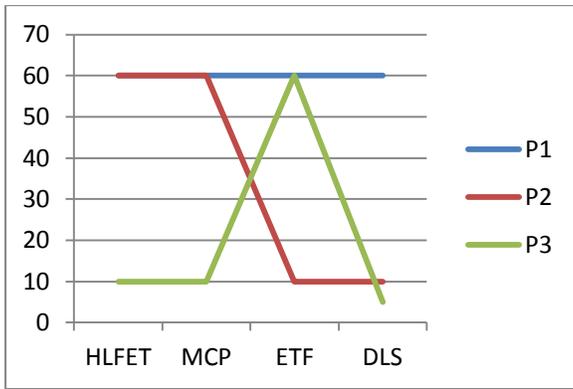


Fig.3.3: Process Utilization for 5 Nodes



5.2.2 Scenario 2: 10 Task Nodes: Makespan of HLFET,MCP,ETF is equal but Makespan of DLS is higher than the others. Similarly, SLR and Speedup is same for MCP,ETF, DLS but different for HLFET. The same applies for Process Utilization.

Fig.4.2:SLR & Speed up for 10 nodes

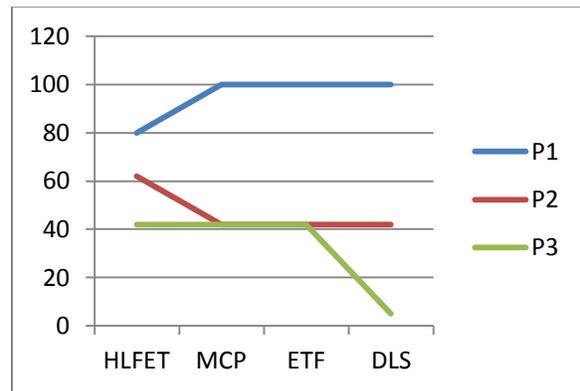
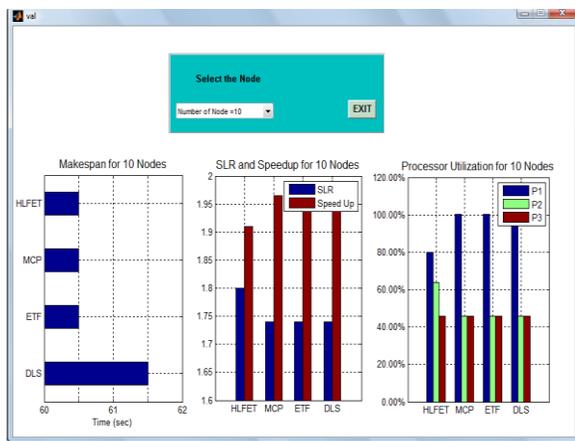


Fig.4.3: Process Utilization for 10 Nodes

5.2.3 Scenario 3: 15 Task Node: MCP shows the highest Makespan and Process Utilization values than others whereas ETF shows the lowest Speedup and SLR values.

Fig. 4. Values of different parameters for 10 nodes.

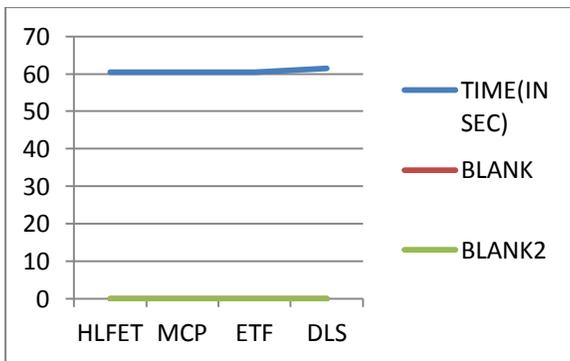
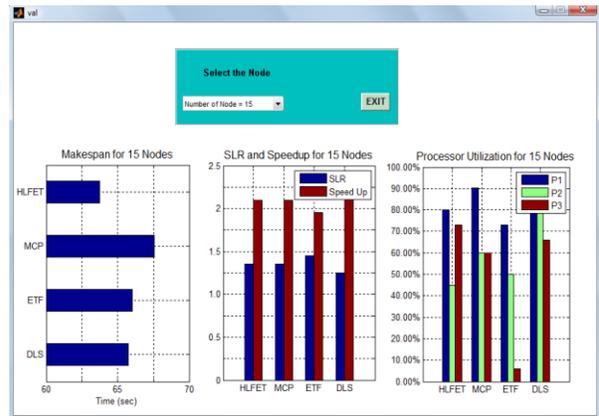


Fig.4.1:Makespan for 10 nodes

Fig. 5. Values of different parameters for 15 nodes.

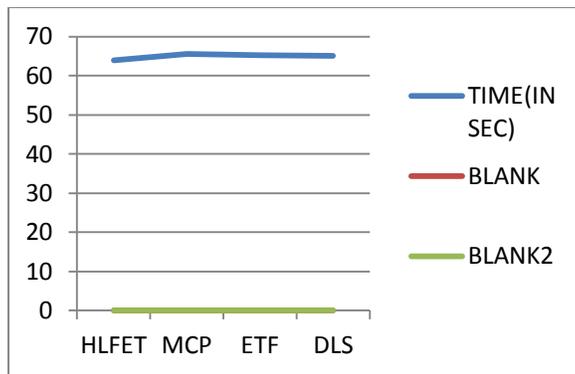


Fig.5.1: Makespan for 15 nodes

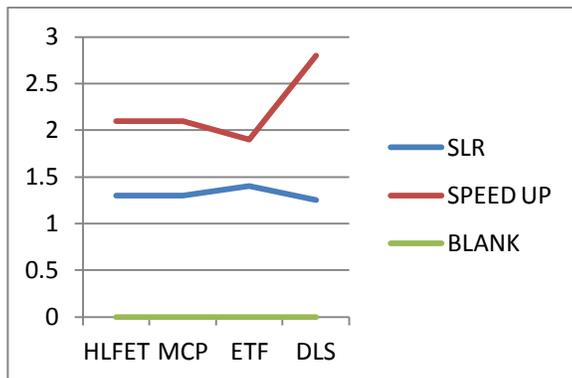


Fig.5.2: SLR & Speed up for 10 nodes

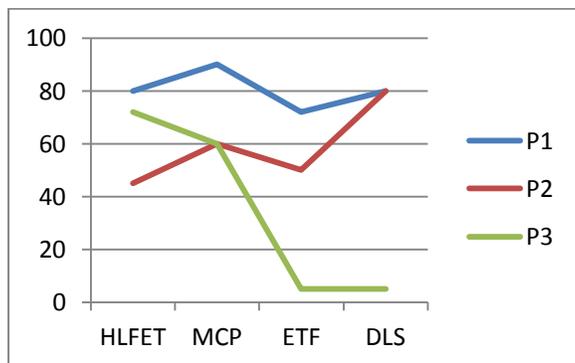


Fig.5.3: Process Utilization for 15 nodes

## VI. CONCLUSION

After comparative analysis, following results were obtained:

- Makespan of DLS showed large increase in amount with 10 nodes and MCP for 15 nodes.
- Process Utilization remained same for 5 nodes. MCP, ETF, DLS utilized process efficiently than HLFET for 10 nodes with 15 nodes. MCP proved to be better than other algorithms whereas ETF showed large drop in utilization rate.
- SLR remained the same for 5 nodes while HLFET for 10 nodes and ETF for 15 nodes showed lesser SLR.
- ETF was the algorithm with higher Speedup.

## VII. FUTURE SCOPE

Although fuzzy logic is very interesting to be implemented but it has a loop hole into it. There is a rule base for each and every type of architecture into it. If we would be introducing a new random node into the system, the fuzzy logic won't have any information about it unless and until we don't mention it into the program by creating a new rule base. Hence in future, if somebody can find a method to optimize the rule set on run time so that the system takes a slight less time to create those rule sets which would make a system little smoother. This task can be achieved with the help of some of the latest methods formally known as Neural Classifications. There are some other methods also which can be tried for genetic algorithm such as PS, PSO, BFO.

## REFERENCES

- [1] T. Hagrais; J. Janeček ( 2003) "Static vs. Dynamic list-scheduling performance comparison" Acta Polytechnica vol. 43 no.6.
- [2] Parneet Kaur; Dheerendra Singh; Gurbinder Singh ; Navneet singh (2011) "Analysis, comparison and performance evaluation of BNP scheduling algorithms in parallel processing International journal of information technology and knowledge management" volume 4, no. 1, pp. 279-284.
- [3] Ranjit Rajak (2012) "Comparison of bounded number of processors (BNP) class of scheduling algorithms based on matrices" no.2(34) issn 1512-1232 3 5.
- [4] Hui Jin; Xian-he sun; Ziming zheng; Zhiling lan ; Bing xie (2009) "International symposium on cluster computing and the grid performance under failures of Dag-based parallel computing".
- [5] Er. Navneet Singh; Er. Gagandeep Kaur; Er. Parneet Kaur ;Dr. Gurdev Singh(2012) "Analytical performance comparison of bnp scheduling Algorithms".
- [6] Samriti, Sandeep Gill , Ankur Bharadwaj , Navpreet singh, Harsimran Singh, Jashwinder Singh (2012 ) "Analysis of hlfet and mcp task scheduling algorithms", International journal of modern engineering research (ijmer) vol.2, issue.3, may-june 2012 pp-1176-1180 issn: 2249-6645.
- [7] Rinkle Aggarwal, Lakhwinder Kaur, Himanshu Aggarwal(2009) "Design and Reliability analysis of a new fault-tolerant multistage Interconnection network", Icgst-cnir journal, volume 8, issue 2.
- [8] Albert p. C. Chan; Daniel w. M. Chan, m.asce; and John f. Y. Yeung (2009) "Overview of the application of "fuzzy techniques" In construction management research journal of construction engineering and management" asce / november / 1241
- [9] Mutasim nour, Shireen y.m (2006) "Adaptive fuzzy logic speed controller with torque adapted gains function for pmsm drive", Journal of engineering science and technology vol. 1, no. 1 59-75 © school of engineering, taylor's college .
- [10] Amrita Sarkar ;G.Sahoo; U.C.Sahoo(2012) "Application of fuzzy logic in transport Planning International journal on soft computing" (ijsc) vol.3, no.2, may 2012
- [11] José m. Merigó, anna m. Gil-lafuente(2009) "Some basic results of fuzzy research in the isi web of knowledge", ninth international conference on intelligent systems design and applications
- [12] T. Trigo de la vega, p. Lopez-garcía, s. Muñoz-hernandez "Towards fuzzy granularity control in Parallel/distributed computing".
- [13] Manik Sharma, Dr. Gurdev Singh, Harsimran Kaur (2012) "A study of BNP parallel Task scheduling algorithms Metric's for Distributed database system", International journal of distributed and parallel systems (ijdps) vol.3, no.1.
- [14] M.kaladevi ; Dr.s.sathiyabama (2010) "A comparative study of scheduling algorithms for real time task" ,International journal of advances in science and technology, vol. 1, no. 4.
- [15] Nidhi Arora (2012) " Comparative study of task duplication based scheduling algorithms for parallel systems department of computer science and engineering" ,International journal of computer applications (0975 – 8887) volume 58– no.19.
- [16] Ali m. Alakeel (2012) "A fuzzy dynamic load balancing algorithm For homogenous distributed systems".

- [17] Ravi Rastogi; Nitin Durg Singh Chauhan ; Mahesh Chandra Govil (2011) "Disjoint paths multi-stage interconnection networks stability problem", Ijcsi international journal of computer science issues, vol. 8, issue 4, no 1.
- [18] Ravi rastogi ; Nitin ; Durg singh chauhan ; Mahesh Chandra Govil (2011) " On stability problems of omega and 3-disjoint paths omega multi-stage interconnection networks", Ijcsi international journal of computer science issues, vol. 8, issue 4, no 2
- [19] Rinkle Rani Aggarwal (2012) "Design and performance evaluation of a new irregular fault-Tolerant multistage interconnection network", Ijcsi international journal of computer science issues, vol. 9, issue 2, no 3.
- [20] Ms. Rita Mahajan, Dr. Renu Vig , Ms. Preeti Abrol (2012) " Ftelmin network: a new min with Fault- tolerance characteristics", vol 5.
- [21] Erick Cantú-Paz (2008) "A survey of parallel genetic algorithms", Vol 5.
- [22] Yudan Liu<sup>1</sup>, Raja Nassar, Chokchai (box) leangsuksun , Nichamon naksinehaboon, Mihaela Paun, Stephen L. Scott (2008) " An optimal checkpoint/restart model for a Large scale high performance computing System"
- [23] Michael Isard Mihai Budiu, Yuan Yu Microsoft Research, Silico (2004) , "Dryad: Distributed Data-Parallel Programs from Sequential Building Blocks" Vol 5.
- [24] Mamta Ghai ;Karamjit Kaur Cheema (2010) " Design and reliability analysis of new fault-tolerant Irregular multistage interconnection network International journal of computer applications" (0975 – 8887) Volume 10– no.1.
- [25] Jorge R. Ramos; Vernon Rego (2004) "Efficient implementation of Multiprocessor scheduling Algorithms on a simulation Testbed" .
- [26] Ishfaq Ahmad, Yu-Kwong Kwok; Min-You Wu (1996) "Analysis, Evaluation, and Comparison of Algorithms for Scheduling Task Graphs on Parallel Processors".
- [27] S. N. Sivanandam, S. Sumathi and S. N. Deepa (2007) "Introduction to Fuzzy Logic using MATLAB", Springer-Verlag Berlin Heidelberg.
- [28] Li Xin Wang (1997) "A course in Fuzzy system & control".
- [29] Yue Wu; Biaobiao Zhang; Jiabin Lu; K. -L. Du(2011) "Fuzzy Logic and Neuro-fuzzy Systems: A Systematic Introduction", International Journal of Artificial Intelligence and Expert Systems (IJAE), Volume (2) : Issue (2).
- [30] Claudio Moraga(2005) "Introduction to Fuzzy Logic" FACTA UNIVERSITATIS (NI 'S) SER.: ELEC. ENERG. vol. 18, No. 2, 319-328.
- [31] Z.Zalila;A.Cuquemella;C.Penet;A.Chika;B.Lorentz;D.Deschamps;C.Assemat (2007)" Fuzzy Logic & Fuzzy Interference System: Introduction & properties" .
- [32] Srinivasa Rao D; Seetha M; Krishna Prasad MHM(2012) "Comparison of Fuzzy and Neuro Fuzzy Image Fusion Techniques and its Applications" International Journal of Computer Applications (0975 – 8887) Volume

## BIOGRAPHIES:



She is Convener of CSE Dept. at Sant Baba Bhag Singh Institute of Engineering & Technology ,Padhiana. She has completed her B.Tech in 2003 and M.Tech in 2006 and pursuing Ph. D from Punjab Technical University, Jalandhar. She has published more than 15 papers in National and International Conferences and 7 papers in International Journals. She has guided more than 100 students in B.Tech Major Projects and 30 in M.Tech Major Projects.



Amita Sharma, A dynamic lecturer, who just not has a flair for teaching but a passion to learn. She has worked at K.C. Group of Institutes, Nawanshahr. She has Completed her B.tech (Computer Science & Engineering) in 2008 and M.tech (Computer Science & Engineering) in 2012. She has published more than 5 national and international papers. She is an active researcher and her areas of interest are Parallel Computing, Software testing and MANET (Moblile Aadhoc Network). She has guided more than 60 students in B. tech Major Projects