

# An Optimized Task Duplication Based Scheduling in Parallel System

**Rachhpal Singh**

Department Of Computer Science, Guru Nanak Dev University, Amritsar-Punjab (India)  
E-mail: rachhpal1969@gmail.com

**Abstract**—By the inherent nature of solving enormous number of problems with the concurrent execution, parallel process methods grow to be a popular technique. The challenges of parallel computing are dealing with the computing resources for the number of tasks and complexity, dependency, resource starvation, load balancing and efficiency. In this paper, the brief discussion about the parallel computation is carried out, and numerous performance issues are also discovered as an open issue. The risk encountered in parallel computing is the motivation to analyze different optimization techniques to accomplish the tasks without risky environment. Genetic Algorithm (GA) is another approach to make the concept of scheduling easy and fast. Here the paper presents a Task Duplication based Genetic Algorithm with Load Balance (TD-GA) approach on parallel processing for effective scheduling of multiple tasks with less schedule length and load balance. TD-GA algorithm truly handles the issues very well and the results show that complexity, load balance and resource utilization are finely managed when compared to the other optimization approaches.

**Index Terms**—Task Duplication, Genetic Algorithm, load balance, Task Scheduling, optimization Task, Parallel Computing.

## I. INTRODUCTION

Parallel processing is an advancement of sequential computing and it becomes the most popular option for simultaneously execute multiple process due to its faster execution time, higher throughput and concurrent high performance nature. Usually, parallel processing is taken as high end processing and so is motivated by simulator having numerical computation of complex systems due to the ability to solve larger problems with concurrency. This is described as heterogeneity computing which means system uses more than one kind of processor and each processor works on different problem. Various applications of parallel processing are data mining, web search engines and networked video and multi-media technologies [1].

Task or job represents a set of instructions or programs and is carried out by a machine or a processor. When huge amount of tasks is available to accomplish by the parallel system then there is a need to co-ordinate these tasks for processing. Scheduling is the optimal design to

sort the number of tasks or activities that are executed for accomplishing the goal within a preset period of time. The reason behind this task scheduling is to improve throughput and to diminish waiting time of process [2].

In heterogeneous parallel system, task scheduling is the challenging issue that deals with the problems such as complexity, dependency, resource starvation and load balancing due to number of tasks execution at the same time. To conquer the problem various numbers of scheduling algorithms and techniques were proposed to manage the listed issues. An effective approach is the task duplication based scheduling that are decreasing many idle machines when number of machines in a system increases to duplicate the task. In task duplication a job is replicated and assigned to huge number of machines i.e. some of the similar task is executed in different processors concurrently and other jobs have dependent relationships with replicated jobs [3].

In this paper, a number of performance issues in section IV discovered and discussed that may encounter when using a parallel system in the scenario of task scheduling. So there is a need to study series of algorithms to overcome the problems found. Techniques enabling is the task scheduling scheme based on GA to resolve scheduling problems like mapping, series of tasks execution and optimal configuration of the parallel system. It reduce the conflict while performing load balancing.

A task duplication scheduling scheme utilized here is to minimize the processor idle time and lessen the mean task response time. Though these techniques are employed to overcome the performance issues, some of them are not fully neglected from the system. Hence another efficient algorithm should be analyzed to demonstrate significant performance improvement using the proposed techniques. With the help of proposed technique, optimized results evaluated and comparatively found best.

Parallel Overhead is still an open issue which is defined as amount of time needed to synchronize parallel jobs that includes job initial time and job final time. The other problem is due to lack of scalability between memory and CPU (processor). It means when adding more processors or machines that increase traffic on the shared memory.

Load balance is another critical problem when the scheduling of multiple tasks assigned to every machine is less or more equivalent to other machines. Balancing

load between parallel processors is computed by the ratio of maximum schedule length or execution time to average execution time of all machines. These are critical issues that are not circumvented by the parallel processing models which directs us to take an effort on various analysis and optimizations techniques to defeat the issues occurred in the parallel system and significantly improves the performance.

Here section II explains the related work studied for this paper and section III describes over all view of parallel processing architecture with its various models. Section IV illustrates discussion of various analyzed issues of task scheduling in parallel processing and mention the need of duplication and heuristic approach. Section V explains basic genetic algorithm and its functions. Further followed by the Task Duplication based Genetic Algorithm (TD-GA) with its performances on task scheduling is described in VI<sup>th</sup> section. VII<sup>th</sup> section explores the performance evaluation and VIII<sup>th</sup> section discussed other related issues of parallel processing. Later, in section IX conclusion and future work is discussed well.

## II. RELATED WORK

D. Asir Antony Gnana Singh, E. JebamalarLeavline, R. Priyanka, P. Padma Priya[4] proposed nature's heuristics Genetic Algorithm. They discussed a heuristic mechanism having selection features to improve the accuracy of data classification.

Sekhri Larbi, Slimane Mohamed[5] proposed regarding scheduling dilemma using heuristic and meta-heuristic methods. The main objective is resources allocation and solving such type of NP-hard problems. With the development of technology, some of the old methods are poor and decrease the performance of the system in terms of inflexibility, inefficient resources etc. Here with the usability of parallel and distributed environment and use of multitasking feature, scheduling if task is easy and also beneficial in mapping of appropriate system processing. Major objective in such type of system is to overall enhance the performance of the system by minimizing the Make-span and balance the load among the machines. Here in this research mechanism solve the scheduling of tasks and mapping of tasks processor relationship. It will reduce the any type of hindrance in scheduling problems. This approach improves the mapping technique by taking control over allocation of resources and balancing the system. It will make the system in parallel and identical mode.

Zhiwei Ye, MingWei Wang, Huazhong Jin, Wei Liu, XuDong Lai [6] proposed to modify the dynamic efficiency of threshold methods based ant colony optimization technique with genetic algorithm. The experimental results shows the efficient techniques for real time applications which improve the search mechanism defined here.

Reza Ebrahimzadeh, Mahdi Jampour[7] proposed the evolutionary optimization algorithms like Genetic Algorithm to improve the result based on random theory.

Important defections here are local convergence and high tolerances in the results due to its randomness in the system. In this paper, they prepared pseudo random numbers by Lorenz chaotic system for operators of Genetic Algorithm to avoid local convergence. The experimental results show that the proposed method is much more efficient in comparison with the traditional Genetic Algorithm for solving optimization problems. .

Raksha Sharma, Vishnu Kant Soni, Manoj Kumar Mishra, and Prachet Bhuyan [8] have discussed the concept of scheduling to get main feasible environment throughput and to equivalent function require having obtainable computing sources. It identifies concept of scheduling and can set in building additional proficient scheduling mechanisms.

Pinky Rosemarry, Payal Singhal and Ravinder Singh[9] have suggested that in parallel environment a variety of tasks and source scheduling mechanisms will help to carry out further research in investigation.

Osama I. Hassanein, Ayman A. Aly, Ahmed A. Abo-Ismael[10] discussed the genetic-fuzzy control approach with other techniques to meet the required performance objectives using a genetic algorithm. The parameters subject to optimization are the width of the membership functions and scaling factors.

A. Ebaid, R. Ammar, S. Rajasekaran and R. ElKharboutly[11] taken a Directed Acyclic Graph (DAG) for efficient scheduling in parallel computing. They have taken two major parameters schedule length and inter-processor communications to build critical path approach to improve the performance metrics like processor requirement, inter-processor communication, and schedule length etc.

J. Zhao and H. Qiu,[12] discussed scheduling of duplication tasks in cloud computing environment with genetic approach and ant colony algorithm by minimizing the execution time of the job scheduling and to measure energy consumption of every job.

H. A. Bazoobandi, M. Khorashadizadeh and M. Eftekhari, [13] showed the arithmetic approach used in parallel and distributed computing with genetic algorithm. Main motive was to accelerate set of tasks, sub-tasks related applications and their execution concurrently. Also task duplication mechanism was adopted to decrease the overall computation time. Results from the simulator give the effectiveness of defined solution.

S. Shakya and U. Prajapati, [14] have discussed Genetic Algorithm for job shop scheduling.

## III. ARCHITECTURE VIEW

Parallel processing is simultaneous utilization of many resources to resolve computational problems [15][16]. Problem is wrecked into isolated parts and can be solved simultaneously by using multiple processors. Separated problem is then broken into a sequence of instructions. Further instructions are implemented on separate machines in parallel mode [17]. System resources have a lone machine having many CPU and random number of systems interconnected in a group. A computational

dilemmamainly divided into separate parts and can be solved concurrently by executing multiple jobs at any time. The computational problem can be answered within a smaller amount of time with manycomputing devices or sources than single computationalsource. The basic design includes memory and CPU. Both program and data stored by the memory. Data is set of information that is used by the program to execute the process. System takes instructions and/or data from memory, which interprets instructions and executes sequentially.Program execution is carried out serially by statement by statement at a time and takes place one machine. All jobs have a division of parallel instruction setthat contain instructions execution serially.

Fig. 1 represents the parallel computing process

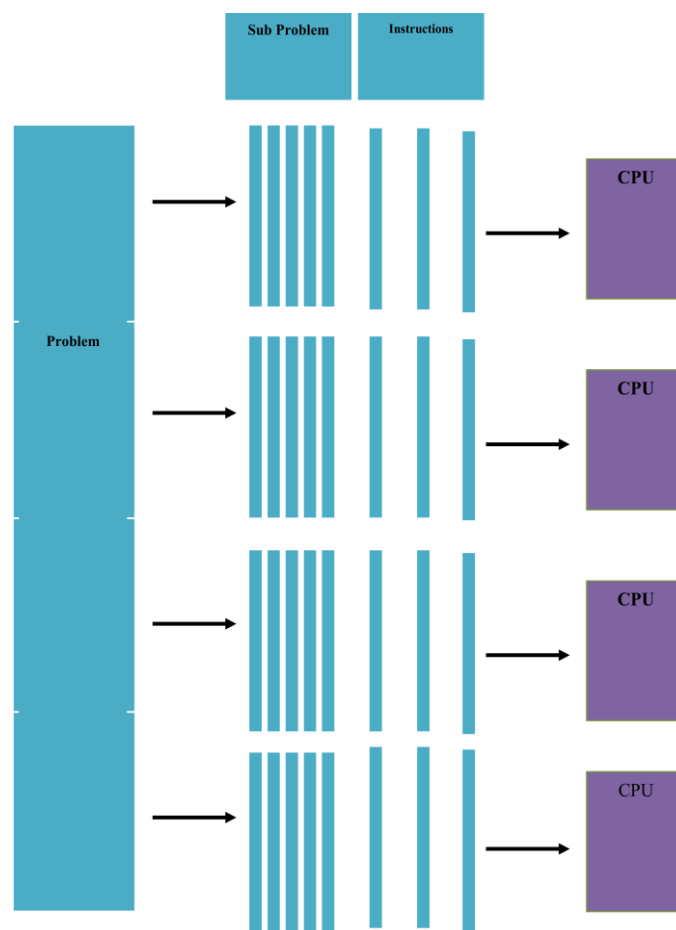


Fig.1. Parallel Computing or parallel Processing

Assume that the parallel processing environment has number of machines, jobs (tasks) and are plotted in graph (called task graph) or DAG (Directed Acyclic Graph) [21] which have  $N$  nodes. Every node in this graph represents the task with the set of instructions which are consecutively executed in identical machines [22].The computation cost of job is calculated by weight of node. The communication cost of jobs is computed by weight of edge connected between jobs in DAG [23]. The communication cost of a job is gain if the jobs are executed on different machines and if two tasksbe executed on same processor, then the communication

showing a problem is separating into sub sections and instructions are given to different CPU for processing each problem simultaneously. Many parallel programming model [18][19] are commonly usedlike sharedMemory, Threads, Message Passing etc. Jobs share a common address space in case of shared memory model whereas a single job hasmany, simultaneous execution paths in thread model.

Similarly message passing has a set of jobs using own local memory. Also using data parallel model, parallel jobs perform operations on a data set and each job execute on a separate partition having same data. Combination of any two or more than two parallel programming models is hybrid model [20].

cost is considered as zero as in Fig. 2(a). Each node is executed to a machine and schedule length is considered as a maximum of task finish time i.e. maximum time taken for the task's execution across the processors [24].

#### IV. SCRUTINIZED ISSUES IN TASK SCHEDULING METHODS

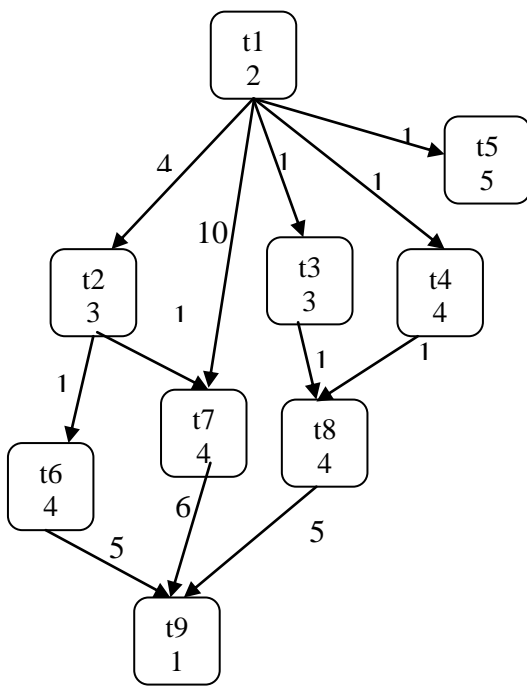
Task scheduling has a vital role in distributed computing system in which many tasks is assigned to available machines by indicate beginning time of each task execution allocate to each processor. The goal of tasks scheduling is to reduce the total execution time and

to handle optimal usage of available processors.

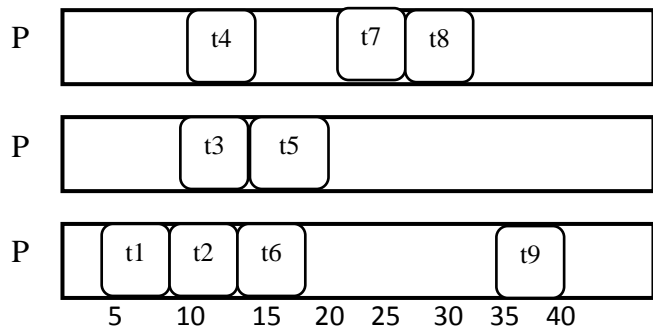
Task scheduling is assigning the various jobs in parallel computing system having a set of similar processor, in other words all the machines are in the heterogenous forms. When such type of processing be done, then the main goal is to determine the series of execution of jobs of every machine in this system. As the execution time of every task or job is determined with the help of series of jobs order in the system that be executed in a the given order. All the jobs are processing according to the assigned processors and tasks from the set of given jobs in the parallel environment. Also it will evaluate the performance of the system and jobs in this system. It will effects the total finish time. So an execution of task scheduling depends upon following

major components:

- i). Order of the jobs in particular sequence.
- ii). Number of machines or nodes in the system.
- iii). Computation and evaluation of the jobs in the parallel computing environment.
- iv). Scheduling of the jobs or tasks.
- v). Mapping of the jobs onto the processors.
- vi). Relationship between processors and tasks.
- vii). Types of scheduling.
- viii). Scheduling criteria to achieve the objective of job processing.
- ix). Evaluate the makespan.
- x). To find the setup of the processors, i.e. weather these are of homogenous or heterogenous.



(a)



(b)

Fig.2. (a) Tasks represented in Directed Acyclic Graph and (b) represents Task Scheduling and Task Execution Time

- Task Scheduling: Suppose that the tasks and the processors need to allocated are too high then the optimal Task Scheduling would be more time-consuming than other random execution of tasks. If T tasks are allocated to M processors, then the number of tasks scheduling be like  $|M|^{[T]}$  and number of states required to execute them will be  $|T|!$  [25].
- Schedule length: The objective of scheduling the task is to reduce the scheduling time and length of the schedule by assigning proper tasks into the appropriate processor.
- Load balance: Load balance is the other problem encountered in the scheduling of multiple tasks

when number of jobs assigned to every machine is less or more equivalent to other machines.

In order to solve problem related with scheduling having many jobs and machines, two approaches are selected that are discussed as below:

- First approach is the **Task Duplication** based scheduling to handle the usage of available machines or systems.
- Second approach is an essential use of **Powerful Heuristic Algorithms** to minimize process completion time as well as optimized scheduling of jobs to appropriate processors.

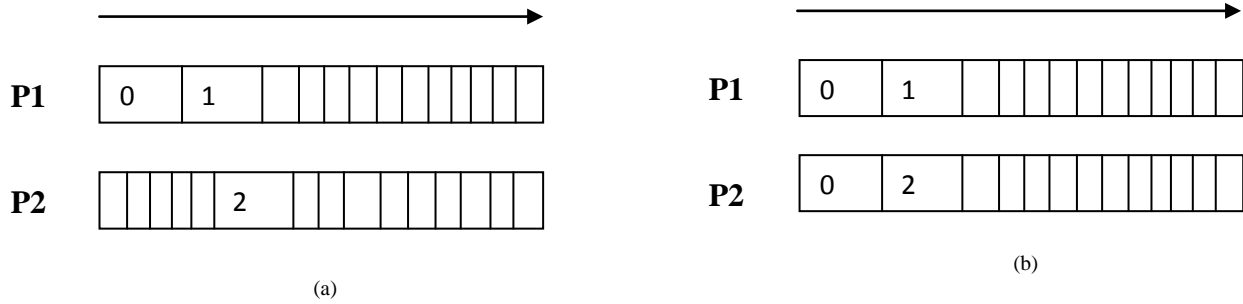


Fig.3. (a) Task scheduling (b) Task Duplication based Scheduling

Why Choosing Task Duplication and GA in Scheduling?

Duplication of tasks is having a parallel mechanism of scheduling jobs by replicating assignments of jobs on which other jobs of program depends critically. It potentially reduces initial times of tasks waiting to receive the data from the critical tasks. So it improves overall execution time of whole program. Scheduling having task duplication is illustrated in Fig. 3(b) which represents that a duplicate task (i.e. task zero) is running in the processor P2 in order to minimize idle time of processor. Thus, duplication has the potential to decrease schedule length by efficiently utilizing processors. Task duplication is used to minimize inter-processor communication overhead that improves total execution time.

Genetic Algorithm (GA) is a heuristic approach [26] [27] [28] [29] [30]. After doing comparisons to other scheduling algorithms, GA has been the growing technology due to the nature of evolution on large number of solutions to obtain the most favorable outcome [30]. Hence genetic algorithm is satisfied as a promising method which is designated for scheduling number of tasks in parallel computing systems.

V. BASIC GENETIC ALGORITHM AND ITS FUNCTIONS

Genetic algorithm is the well-known heuristic approach which is useful and efficient, yet the problem is huge and difficult. GA is based on process of natural selection which means that “selects the best and discards the worst” to find a best solution. GA begins with the population generation of solutions for the problem and performs evolution by repeatedly apply the set of genetic operators. The three search operators used in the genetic algorithm is one selection, second crossover and third one mutation operators that significantly helped for obtaining optimal solution. Each generation population is represented as a chromosome which is corresponding to bit strings like 1010110100.

➤ Population generation:

If the population is randomly generated with string  $n=5$  and length = 10, the population generated is as follows:  
 $G1 = 1111010101$ ;  $G2 = 0110100110$ ;  $G3 = 1110110001$ ;

$G4 = 0100010011$ ;  $G5 = 1110111101$

➤ Evaluate fitness for each individual:

The fitness function is a value assigned to each solution depending on how close it actually from current solution to the expected optimal solution. If fitness function is assumed as the highest distance i.e.  $f(x) = (d)$ , then the best fitness is selected where the distance is high.

Fitness value for the generated population will be as follows:

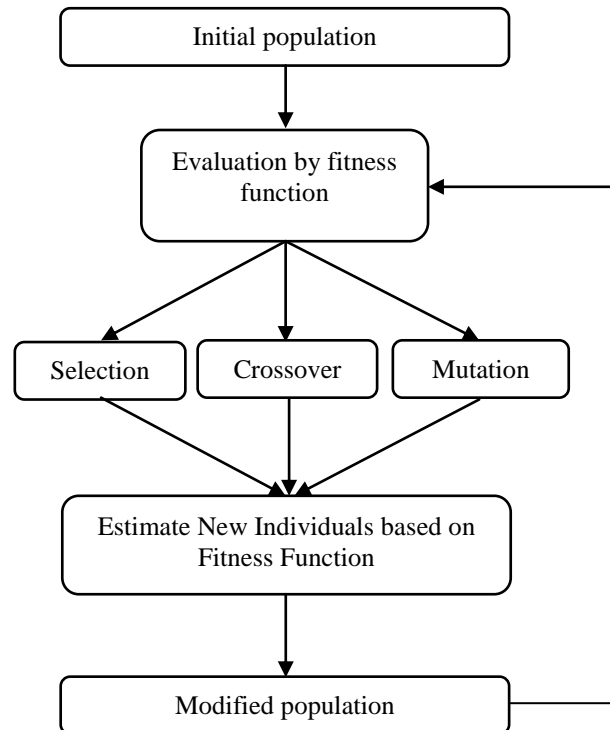


Fig.4. Genetic Algorithm Cycle

$f(G1) = 7$ ;  $f(G2) = 5$ ;

$f(G3) = 6$ ;  $f(G4) = 4$ ;  $f(G5) = 8$ ,

Here after evaluation, the total fitness value is 30.

➤ Selection:

Selection based on several strategies, here taken as ranking metric i.e. depending on highest fitness function or value. G4 chromosome has lowest fitness and is neglected and is replaced by other generation set. The population is now rearranged as follows:

$$G1' = 1111010101; G2' = 1110110001;$$

$$G3' = 1110111101; G4' = 0110100110;$$

$$G5' = 1110111101$$

➤ Crossover:

Crossover selects strings from previous chromosomes and creates new chromosomes or strings by choosing a crossover probability. The crossover taken between the pair G1, G2 and G3, G4, is discussed in table below and the probability taken as 0.6. On the basis of above parameters the crossover will be shown as:

Parent	Before Crossover	After Crossover
1	G1' = 1111010101 G2' = 111011 0001	G1'' = 1111010001 G2'' = 1110110101
2	G3' = 111011 1101 G4' = 011010 0110	G3'' = 1110110110 G4'' = 011010 1101

➤ Mutation:

It is used to keep up the genetic diversity from one generation of a population of chromosomes to next i.e. reverse randomly selected bits based on the mutation probability.

Before Mutation	After Mutation
G1'' = 1111 10101	G1! = 1111110101
G2'' = 11101 0001	G2! = 1110100001
G3'' = 11101 1101	G3! = 1110101101
G4'' = 011 100110	G4! = 0111100110
G5'' = 1110 11101	G5! = 1110011101

The newly generated populations after applying the genetic operators are normally like

$$G1! = 1111110101,$$

$$G2! = 1110100001,$$

$$G3! = 1110101101,$$

$$G4! = 0111100110,$$

$$G5! = 1110011101.$$

The fitness value for each new generated individual is

calculated and computed value is:

$$f(G1!) = 8, f(G2!) = 5, f(G3!) = 7, f(G4!) = 6, f(G5!) = 7.$$

The total fitness value is 35. After a single iteration the fitness value is varied from 30 to 35. This cycle is repeated until the highest solution is reached.

### VI. TASK DUPLICATION BASED GENETIC ALGORITHM (TD-GA)

Using an efficient scheduling technique, some machines might be out of use or have idle throughout and execution of parallel job assignment. Jobs allocated to these might be waiting to collect some data from jobs executing on some other machines. For this scenario a best schedule of tasks having task duplication mechanism has been proposed and named as Task Duplication based Genetic Algorithm (TD-GA). The contribution of Duplication Based Genetic Algorithm is to optimize the best scheduling of tasks with minimized schedule length and high speed, high utilization of processors and to guarantee efficient load balance between parallel processing systems.

Proposed **TD-GA** has been executed similarly like genetic algorithm and is modified with few steps having the task duplication based approach.

1. Initially each tasks weight and communication cost is calculated and are randomly allocated in different parallel processors.
2. The population is generated up to population size N by applying duplication of some tasks randomly in processors.
3. The fitness function should be calculated based on schedule length and load balance parameter for each selected individuals to optimize the best schedule of tasks.
4. Perform the three GA operators (selection, crossover and mutation) on generated population set to reach best population with the minimized schedule length and load balance. The cycle is in loop until specified number of generations G or iteration is arrived (if G=100, then 100 iterations should be carried to optimize the best fitness). It is represented in Fig. 4.

If all the jobs  $t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9$  are scheduled on the machine  $P_1$ , then total completion time of **task execution is 65** and the **communication cost is zero** (as shown in Fig. 2(a)).

If the scheduling of tasks on different processors i.e. tasks  $t_1, t_4, t_6$  and  $t_9$  are allocated in processor  $P_1$  and tasks  $t_2, t_3, t_7$  are allocated in processor  $P_2$  and tasks  $t_5, t_8$  are scheduled in processor  $P_3$  with the indication of their execution time, then the total **execution time is reduced to 43 time units** (It is as shown in Fig. 5).

#### Way to Generate the Population in TD-GA:

The genetic algorithm carried out a problem and maps them to a set of strings called chromosomes. The chromosome is represented by the mapping of jobs to appropriate machines and sequence of jobs for task

scheduling. The population is randomly generated by TD-GA depends on number of jobs allocated to machines. For example, if the generation size is selected as G=5, the

random generated populations with Tasks=9 and Processors=3 with any of the duplicated tasks are represented in Fig. 6.

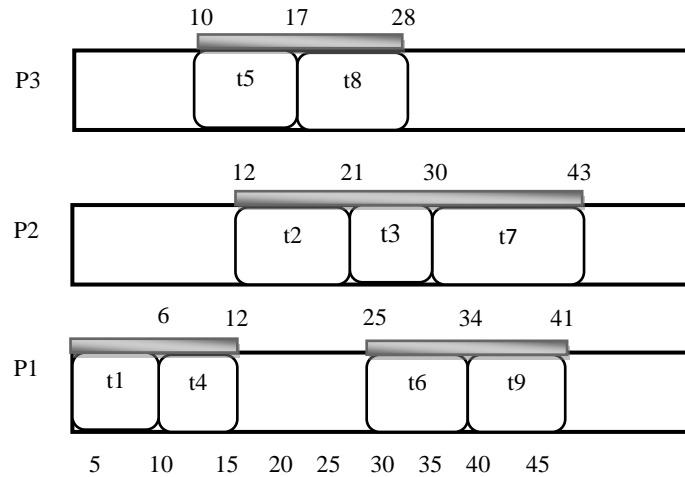


Fig.5. Tasks Scheduling on Different Processors

What is the Fitness Function of TD-GA?

The fitness function is based on the objective parameter to be optimized for improving the system performance. Here in the parallel processor scheduling having multiple jobs, the main objective of scheduling is to minimize length of the schedule and effective handling task balance to improve system's efficiency as well as utility. The major goal is to get chromosome's best fitness function in terms of load balance and schedule length.

Schedule length (SL) is ending time of current job running on parallel processors equation (1).

$$SL = Max [FTJ](1)$$

Where Max [FTJ] is maximum final time of jobs or tasks.

Load or task balance is used to set load balance between parallel processors. Many techniques used the optimized schedule length, but task or job balance in heterogeneous environment might not be satisfied. To remove this hindrance proposed TD-GA algorithm is used to optimize the best load balance and best schedule length to minimize execution time and to balance the jobs. Load Balance (LB) can be computed as illustrated in equation (2).

$$LB = SL / AVG (Ex_{tp}) (2)$$

Where SL is Schedule Length and  $AVG (Ex_{tp})$  is average execution time of processors.

$AV_{exe}$  (Average execution time) of a system is defined as ratio of sum of final time of jobs executing on machines to no. of parallel machines which is illustrated in equation (3).

$$AV_{exe} = T_{exe} / Nop (3)$$

Where  $T_{exe}$  is Total execution time of processors and  $Nop$  is the Number of Processors or machines.

Therefore the Fitness function used in TD-GA is represented with the optimization of load balance and schedule length as depicted in equation (4).

$$Fitness\ of\ TD-GA = 1 / (O_{SL} + O_{LB}) (4)$$

Where  $O_{SL}$  is optimized schedule length and  $O_{LB}$  is optimized load balance.

The fitness function should optimize the best chromosomes i.e. what are the tasks to be allocated and to which parallel processor.

Effects of Genetic Operators on TD-GA performance:

➤ Selection

From the initial population, the individuals or chromosomes are selected to do the operations crossover and mutation (genetic operations). During selection operation, the proposed scheme can choose any individuals having the highest fitness value.

Before moving into next loop of population, selected chromosomes will take place genetic operations to get child (offspring) chromosomes that is used to produce generations of population further.

For example, if two chromosomes are selected based on high fitness among the initial generated populations, then that are selected for the next chromosome generation.

For example the selected chromosome 1 and 2 are having highest fitness value to reach the best solution, then that are represented in Fig. 7.

➤ Crossover

It is operation of changing chromosome structure by choosing individuals from initial set of population having the crossover rate (probability)  $R_{CR}$ . Probability of crossover  $R_{CR}$  is taken between 0.5 and 0.9 in order to receive the best results. In the proposed method, one-point crossover is taken among several strategies which

produce the new chromosomes by taking the probability 0.6.

After this crossover point the individuals interchanged between them to produce the child (offspring) chromosomes. This whole operation is shown in Fig. 8. Sometimes solution does occur, then there be need of two point crossover operation, but here is no need.

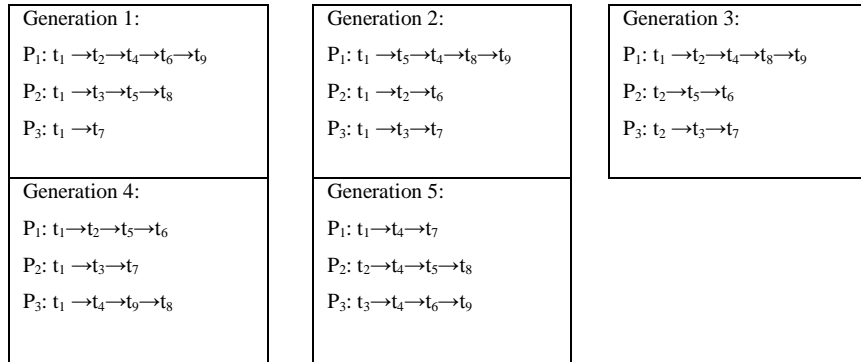


Fig.6. Generation of Population

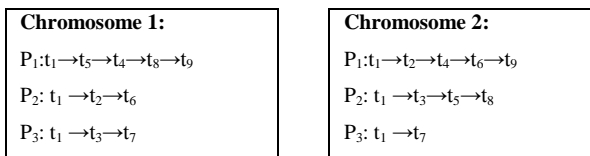


Fig.7. Selection of chromosomes

chromosomes. The cycle represented in Fig. 4 is repeated until the number of iteration is performed to get the best solutions from the generated populations.

If the ending point arrived, then the scheduled jobs are assigned to their corresponding systems. It is best optimal results from the algorithm implementation.

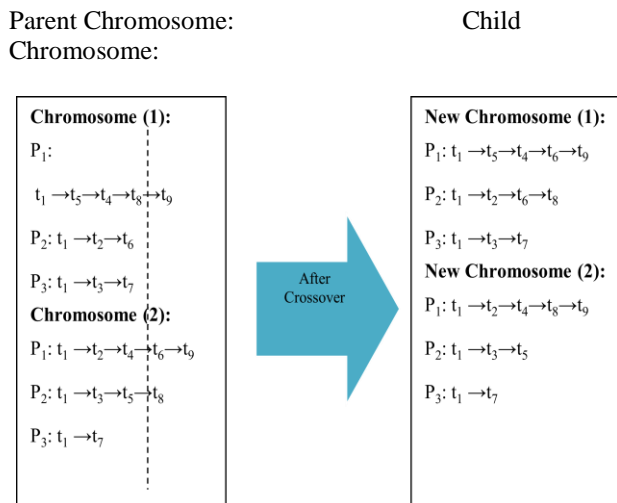


Fig.8. Crossover of chromosomes

➤ Mutation

The mutation is used to turn over and re-establish the lost information to the chromosomes with a little probability to produce the effect of errors during duplication. It is as shown in Fig. 9.

Here the mutation probability taken is 0.01. Point where operation mutation occurs is known as mutation point (chosen randomly in an individual).

After completion of genetic operators there will be better chromosomes than the previous generated

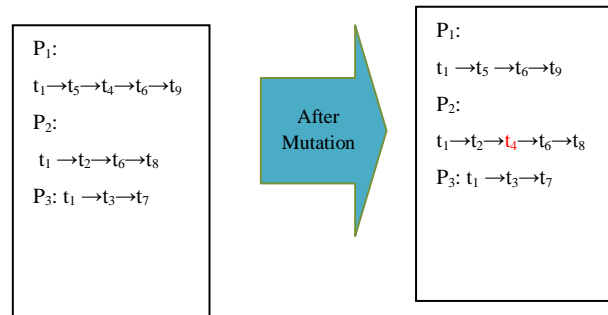


Fig.9. Mutation of chromosomes

VII. PERFORMANCE EVALUATION

ProposedGA with TD in parallel processing is experimented with simulator in Java Net Beans IDE. It is setup having jobs cc (communication cost) and jobs wt (weight) that are taken randomly generated jobs cc as well as jobs wt in task table. Simulated task duplication mechanism improved by replication of more jobs dynamically in order to decrease load balance and schedule length. Here consider variation of tasks like 10, 20, and 30 with the increased number of processors i.e. 3 and 4processors. Duplication is taken from 3 tasks to 5 tasks along with the processors. The generation is taken as 10, i.e. the population size is 10 and the crossover point considered is 0.5 with mutation probability 0.08. The fitness function is based on the schedule length and



load balance represented in equation (4). Proposed mechanism dynamically assigned jobs to machines according to fitness objective functions. As per operation first job or task is replicated on every available machine and so other jobs be replicated as per requirement and user’s reliability. Jobs in machine are assigned by jobs weight and communication cost with adjacent tasks (creating adjacent task graph). So proposed technique find ideal time of every machine which is executing in parallel mode and replicates jobs in ideal time of machine in order to decrease load balancing and schedule length.

The ultimate goal here is to decrease schedule length by replication of jobs in parallel environment. Fig. 10 describes execution time or completion time and can be minimized with the help of a new heuristic technique and genetic approach occurs as the best algorithm.

When there is no duplication by GA scheduling, then the schedule length i.e. the total execution time is up to 190 units for 30 tasks with 4 processors. If the duplication of tasks is 3,(duplication=3), then the schedule time is significantly reduced to 50 time units.

This demonstrates that the GA along with duplication can absolutely reduce the schedule length in an effective manner. The fitness value for schedule length is calculated by the TD-GA approach and is plotted in graph, which represents that the schedule length is

minimized after 10 iterations with 30 tasks and 3 duplicated tasks on 4 processors.

In the first iteration the schedule length is in top i.e. up to 60 time units, and is decreased after the number of iterations. Finally it reaches the optimized level in the 10th iteration.

This proposed method is also developed for balancing the load among processors (Assignment of jobs to parallel system is a balanced approach for proper utilization of processors).

Here Fig. 11 shows that load balance is the measure of processors with number of varied tasks with the help of TD-GA algorithm. The graph plotted for 30 tasks allocated into 4 processors with 3 duplicated tasks and the fitness of load balance is calculated.

This demonstrates that the GA along with duplication can entirely reduce the load balance in an effective manner. During the first iteration the fitness value for load balance is normally in lower level and after the number of iteration which is progressively improved for number of processors and tasks. At last the load balance level reaches the optimized fitness value after 10th iteration from 1.02 to 1.15.

The entire analysis shows that the optimization of schedule length and fitness is individually calculated and the total fitness is evaluated according the equation (4).

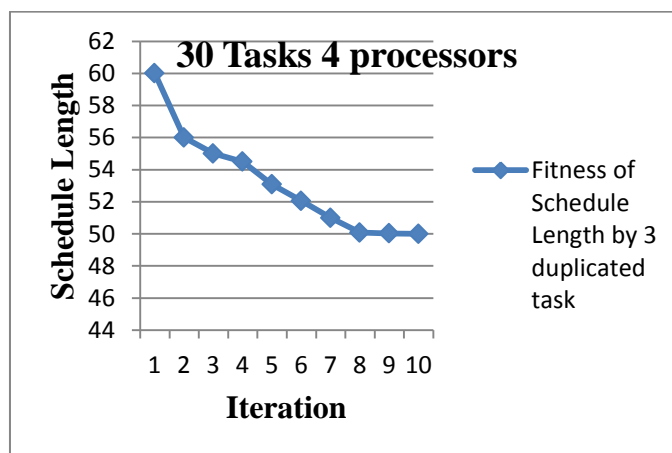


Fig.10. Optimization of Schedule length by Fitness function

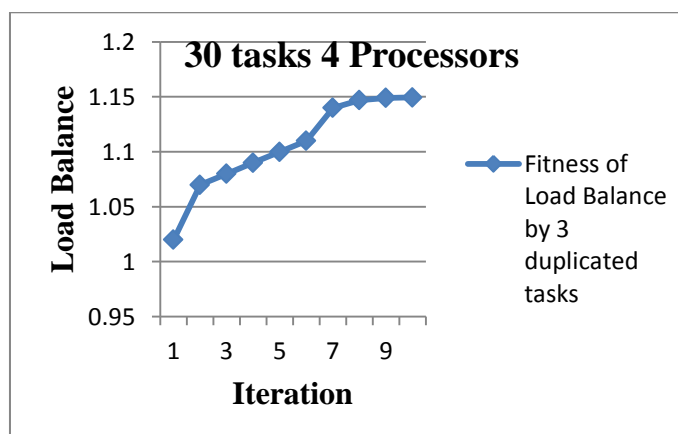


Fig.11. Optimization of Load Balance by Fitness function

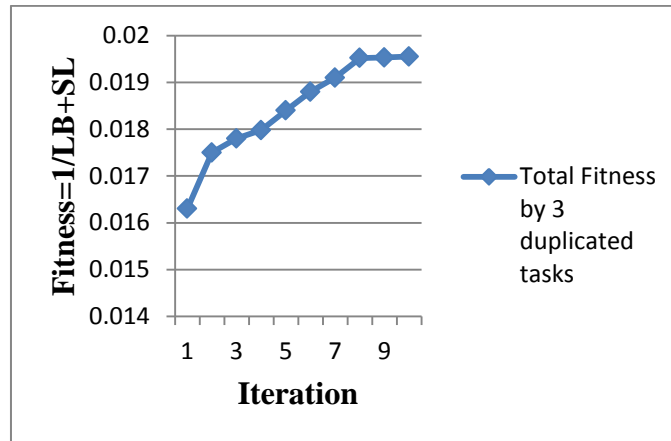


Fig.12. Optimization of Load Balance and Schedule Length by Total Fitness

The Fig. 12 represents the optimized schedule length and load balance by the fitness of TD-GA approach. Thus the system performance is increased for each generation and finally reaches the best from the fitness evaluation.

The Fig. 13(a) describe comparison of schedule length between proposed methods by varying replication jobs with genetic method and carried without job replication. After applying the new scheduling genetic algorithm TD-GA, the execution time can be reduced by using the GA to optimize the tasks, the performance becomes better than other type of scheduling algorithm. The Fig. 13(b) illustrates the handling of load balance measure when the

task is increased with minimized number of processors.

For 3 duplicated tasks, among 10 tasks along with 3 processors, the task balance is 1.0206. At the same time, when the task is increased up to 30 along with 3 duplicated tasks, the task is also balanced very well as 1.0526. Since the load balance could be efficiently handle by the proposed TD-GA approach.

Various estimation of TD-GA shows that, the total execution time or completion time and load balance is effectively minimized, and it improves the throughput of system. It is noticed that the TD-GA is the best algorithm as by the parameter evaluation.

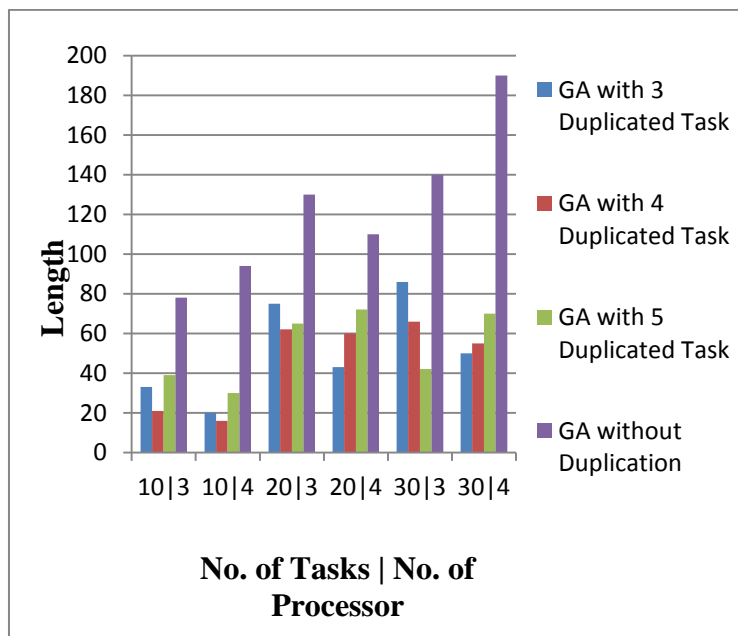


Fig.13. (a) Schedule length of tasks on processors

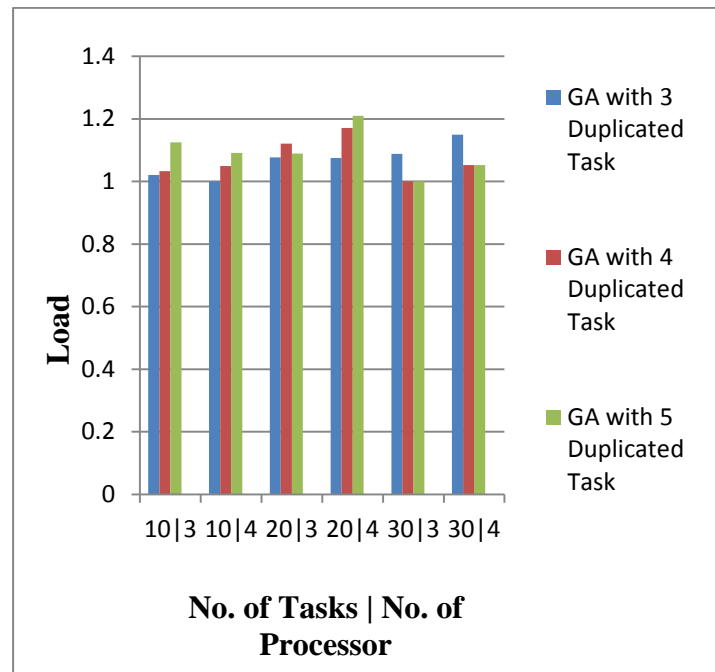


Fig.13. (b) Load Balance by TD-GA

### VIII. OTHER ISSUES

In general, parallel operations are much more intricate than sequential mode operations due to multiple instruction streams executing at the same time. Dependency occurs between instruction sets, when sequence of job execution influences solution of method. Data dependence solutions from multiple use having same position in storage by dissimilar jobs. Dependency is vital to parallel environment because it is one of the prime inhibitors to parallel system.

Resource starvation is a problem where a process continuously avoids the essential resources that are most effective part of the system to finish the task.

Load balancing defines the exercise of allocating job load between jobs so that all jobs become busy all the time and job idle time be minimized. It is vital to parallel system for showing the best performance and these reasons optimize it.

### IX. CONCLUSION

When Task scheduling is elected for distributed computing system, then there should be number of problems encountered in it due to large number of tasks and resources. In order to handle the issues occurred during scheduling in parallel computing, lot of heuristic algorithms have been proposed. This paper addresses the various problems occurred in task allocation to the processors and studied the effective heuristic approach to overcome the issues of schedule time, load balance and utilization of resources. The work mainly focuses on handling load balancing when numerous numbers of multiple tasks are entered into the system and proposed Task Duplication based Genetic Algorithm (TD-GA) for

faster and efficient execution. Mostly, all the mentioned issues are handled very well by the proposed algorithm which is shown in the results. In the future work, the discussion would like to enlarge the optimization method to evaluate the problem on the parallel system for improving the scalability.

### ACKNOWLEDGMENT

The author wishes to thank Prof. (Dr.) Karanjeet Singh Kahlon (Supervisor) and Prof. (Dr.) Gurbinder Singh (Co-Supervisor), both from Department of Computer Science, Guru Nanak Dev University, Amritsar (Punjab-India) for their guidance.

### REFERENCES

- [1] Frank Willmore, "Introduction to Parallel Computing", 2012.
- [2] Haluk Topcuoglu, Salim Hariri and Min-You Wu, "Performance-Effective and Low-Complexity Task Scheduling for Heterogeneous Computing", *IEEE Transactions on Parallel and Distributed Systems*, Vol.1, No.3, March 2002.
- [3] Koichi Asakura, Bing Shao and Toyohide Watanabe, "A Task Duplication Based Scheduling Algorithm for Avoiding Useless Duplication", July 18, 2010.
- [4] D. Asir Antony Gnana Singh, E. Jebamalar Leavline, R. Priyanka, P. Padma Priya, "Dimensionality Reduction using Genetic Algorithm for Improving Accuracy in Medical Diagnosis", *International Journal of Intelligent Systems and Applications (IJISA)*, IJISA, Vol. 8, No. 1, January 2016, ISSN: 2074-904X (Print), ISSN: 2074-9058 (Online), DOI: 10.5815/ijisa, Published By: MECSPublisher
- [5] Sekhri Larbi, Slimane Mohamed, "Modeling the Scheduling Problem of Identical Parallel Machines with Load Balancing by Time Petri Nets", *International*

- Journal of Intelligent Systems and Applications(IJISA), ISSN: 2074-904X (Print), ISSN: 2074-9058 (Online), Vol. 7, No. 1, December 2014, DOI: 10.5815/ijisa, Published By: MECS Publisher.
- [6] Zhiwei Ye, MingWei Wang, Huazhong Jin, Wei Liu, XuDong Lai, "An Image Thresholding Approach Based on Ant Colony Optimization Algorithm Combined with Genetic Algorithm", *International Journal of Intelligent Systems and Applications(IJISA)*,ISSN: 2074-904X (Print), ISSN:2074-9058(Online), DOI: 10.5815, Vol. 7, No. 5, April 2015, Published By: MECS Publisher.
- [7] Reza Ebrahimzadeh, Mahdi Jampour, "Chaotic Genetic Algorithm based on Lorenz Chaotic System for Optimization Problems",*International Journal of Intelligent Systems and Applications(IJISA)*, ISSN: 2074-904X (Print), ISSN: 2074-9058 (Online), Vol. 5, No. 5, April 2013,PP.19-24,DOI: 10.5815/ijisa.2013.05.03, Publisher:MECS.
- [8] Raksha Sharma, Vishnu Kant Soni, Manoj Kumar Mishra and PrachetBhuyan, "A Survey of Job Scheduling and Resource Management in Grid Computing", *World Academy of Science, Engineering and Technology*, Vol.4, 2010
- [9] Pinky Rosemarry, PayalSinghal, and Ravinder Singh, "A Study of Various Job & Resource Scheduling Algorithms in Grid Computing", *International Journal of Computer Science and Information Technologies*, Vol. 3, pp. 5504-5507, 2012.
- [10] Osama I. Hassanein,Ayman A. Aly,Ahmed A. Abo-Ismael, "Parameter Tuning via Genetic Algorithm of Fuzzy Controller for Fire Tube Boiler",*International Journal of Intelligent Systems and Applications*, ISSN: 2074-904X (Print), ISSN: 2074-9058 (Online), Vol.4, No.4, April 2012, PP.9-18.
- [11] A. Ebaid, R. Ammar, S. Rajasekaran and R. ElKharboutly, "An enhanced scheduling algorithm using a recursive critical path approach with task duplication," *2012 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, Ho Chi Minh City, 2012, pp. 000107-000113, doi: 10.1109/ISSPIT.2012.6621269.
- [12] J. Zhao and H. Qiu, "Genetic algorithm and ant colony algorithm based Energy-Efficient Task Scheduling," *2013 IEEE Third International Conference on Information Science and Technology (ICIST)*, Yangzhou, 2013, pp. 946-950.doi: 10.1109/ICIST.2013.6747695
- [13] H. A. Bazoobandi, M. Khorashadizadeh and M. Eftekhari, "Solving task scheduling problem in multi-processors with genetic algorithm and task duplication,"*Intelligent Systems (ICIS)*, *2014 Iranian Conference on*, Bam, 2014, pp. 1-4.doi: 10.1109/IranianCIS.2014.6802528.
- [14] S. Shakya and U. Prajapati, "Task scheduling in Grid computing using Genetic Algorithm," *Green Computing and Internet of Things (ICGCIoT)*, *2015 International Conference on*, Noida, 2015, pp. 1245-1248.doi: 10.1109/ICGCIoT.2015.7380654
- [15] Oliver Sinnen, Andrea To, ManpreetKaur, "Contention-Aware Scheduling with Task Duplication", *Journal of Parallel and Distributed Computing*, Volume 71, Issue 1, January 2011, Pages 77–86.
- [16] Ian Karlin et al, "Exploring Traditional and Emerging Parallel Programming Models using a Proxy Application", *IEEE 27th International Symposium on Parallel & Distributed Processing (IPDPS)*, 2013.
- [17] He,Y.Hongyang Sun and Wen-Jing Hsu, "Adaptive Scheduling of Parallel Jobs on Functionally Heterogeneous Resources",*International Conference on Parallel Processing*, 2007.
- [18] "Introduction to Parallel Computing", George Karypis, *Parallel Programming Platforms*.
- [19] Satoshi Ohshima, Kenji Kise, Takahiro Katagiri and Toshitsugu Yuba "Parallel Processing of Matrix Multiplication in a CPU and GPU Heterogeneous Environment",2006.
- [20] Henry Kasim, Verdi March, Rita Zhang1, and Simon See, "Survey on Parallel Programming Model", 2008.
- [21] Wei-Ming Lin and QiuyanGu, "An Efficient Clustering-Based Task Scheduling Algorithm for Parallel Programs with Task Duplication", *Journal Of Information Science And Engineering*, Vol. 23, 589-604, 2007.
- [22] Mohammad I. Daoud and NawwafKharna, "An Efficient Genetic Algorithm for Task Scheduling in Heterogeneous Distributed Computing Systems",*2006 IEEE Congress on Evolutionary Computation*, 2006.
- [23] Heejo Lee, Jong Kim, Sung Je Hong,Sunggu Lee, "Task scheduling using a block dependency DAG for block-oriented sparse Choleskyfactorization",*Parallel Computing (ELSEVIER)*, Vol. 29, pp. 135–159, 2003.
- [24] HeikkiOrsila, TeroKangas, and Timo D. Hamalainen, "Hybrid Algorithm for Mapping Static Task Graphs on Multiprocessor SoCs", 2012.
- [25] Amir MasoudRahmaniandMojtabaRezvani, "A NovelGenetic Algorithm for Static Task Scheduling in Distributed Systems",*International Journal of Computer Theory and Engineering*, Vol. 1, No. 1, April 2009.
- [26] Singh, Jasbir, Singh Gurvinder, "Improved Task Scheduling on Parallel System using Genetic Algorithm", *International Journal of Computer Applications*, Feb2012, Vol. 39, p17-22, 2012.
- [27] Dahal, K.Hossain, A., Varghese, B, "Scheduling in Multiprocessor System Using Genetic Algorithm, *Computer Information Systems and Industrial Management Applications*, 2008.
- [28] Jaspal Singh, Harsharanpal Singh, "Efficient Tasks scheduling for heterogeneous multiprocessor using Genetic algorithm with Node duplication", *Indian Journal of Computer Science and Engineering (IJCSSE)*, 2011.
- [29] RituVerma, SunitaDhingra, "Genetic Algorithm for Multiprocessor Task Scheduling", *CiteSeer*, 2011.
- [30] Yu-Kwong Kwok, Ishfaq Ahmad, "Efficient Scheduling of Arbitrary Task Graphs to Multiprocessors Using a Parallel Genetic Algorithm", *Journal of Parallel and Distributed Computing*, Vol. 47, Issue 1, 25 November 1997, Pages 58–77.

#### Authors' Profiles



**Rachhpal Singh** completed his MCA in Computer Science in 1994 from Thapar Institute of Engineering and Technology, Patiala (Punjab) -India, under T.I.E.T. as Deemed University, Patiala (Now as Thapar University, Patiala). Currently, He is pursuing his Ph.D. degree from Guru Nanak Dev University, Amritsar-Punjab (India).Rachhpal Singh passed Ph.D. course work with 9.3 CGPA out of 10 CGPA. Rachhpal Singh is working as Senior Assistant Professor in the PG department of Computer Science and Applications, Khalsa College, Amritsar. His research areas include (Parallel and Cloud Computing). Singh is having more than 22 years teaching experience and has authored more than 10 popular books in field of Computer Science.