iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 4 **Issue:** IX **Month of publication:** September 2016

DOI:

www.ijraset.com
IC Value: 13.98

Volume 4 Issue IX, September 2016
ISSN: 2321-9653

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

A Genetic Algorithm for Resource Contrained Project Scheduling Problem

Shaikh Shadab Anwar[1], Dr. V. S Bilolikar[2]

[1,2]Department of Production Engineering,
Fr. C .Rodrigues College of Engineering, Bandra(W), Mumbai.

Abstract— Resource constrained project scheduling in a well know NP hard problem of Operation Research which has attracted a lot of research in last decade. This paper deals with a metaheuristic methodology for solving a RCPSP (Resource Constrained Project scheduling Problem) and makes use of Genetic Algorithm with an objective of minimizing the makespan of the project schedule. Problem Instances of PSPLIB were solved and % Average Deviation was calculated.

Keywords— Genetic Algorithm, RCPSP (Resource Constrained Project Scheduling), Metaheuristic, (SGS) Serial Schedule Generation Scheme.

I. INTRODUCTION

Resource constrained Project Scheduling Problem is a scheduling problem for scheduling project activities with precedence and resource constrained with some objective, such as minimization of project makespan, Net Present Value etc. RCPSP serves a backbone of many industries and its application can be found in industries such as Construction, Software Development, Production etc. are some to be named. The RCPSP has attracted many researchers [1] [2].Solution Methodology can be classified in to four categories such as exact, enumerative, heuristic and metaheuristic. Exact method can be used to optimize project with less than 60 activities, for the project with more number of activities heuristic and metaheuristic methods are found to find an optimal or a near optimal solution. Metaheuristic method such as GA [3] [4] [5], Simulated Annealing [7] [8] and Tabu Search (TS) [9] were successfully implemented to solve RCPSP and it was found that SA and GA outplay all other heuristic in the literature. GA and SA were found to perform best on a large size project. [10].

The performance of GA largely depends on the solution encoding and decoding. Activity list representation was proposed by Hartman [4] and Computational Result reveals that this GA outperforms other GA based on other solution encoding techniques. This GA makes use of Serial Scheduling Scheme (SGS) as a decoding method. It was shown by a Hartman [5] that not only Serial Scheduling Scheme decoding procedure for activity list representation. Different encoding procedure produces different schedule thus leading to different makespan. An additional gene for extended representation to an activity list representation called Serial-parallel gene (S/P gene) serves as a transformation from activity list transformation into a schedule was proposed by Hartmann. This is known as a self-adapting GA. Alcaraz and Maroto [6] used a new representation with additional gene known as forward-backward gene (F/B gene) it determines the direction in which activity list is scheduled. This representation leads to a different schedule with probably a different makespan in a forward and a backward way [12] [13]. This paper uses an activity list representation without an additional gene which can be transformed into schedule using a serial scheduling scheme. An extensive computational experiment shows that the algorithms proposed in the works of Hartmann [5] and Alcaraz and Maroto [6] outperform the other heuristic algorithms. This paper proposes a GA for RCPSP with activity list representation and two point crossover. The Serial Scheduling Scheme (SGS) is adopted as a decoding procedure; Elitist Strategy was adopted to retain the best in the next generation. This paper is organized as follows after introduction Section 2 explains the mathematical model Section 3 presents the proposed GA to RCPSP Section 4 gives details of Obtained results with Analysis.

II. RESOURCE CONSTRAINED PROJECT SCHEDULING

A project is visualized as an A-O-N network. The activities are assigned numeric values *1* to *n* where *1* represents the beginning and *n* represents the ending activity. D_i $(1 \le i \le n)$ denotes integer duration of activities, S_i $(1\le i \le n)$ and F_i $(1 \le i \le n)$ denotes an integer starting and finishing time respectively. R_{ik} $(1\le i \le n,\ 1\le k \le K)$ denotes a constant resource required by an activity *i* of type *k* and A_k is the constant availability of resource *k*, Mathematically RCPSP can be described as:

$$Min\ F_n \qquad \ldots \ldots (1)$$

Subject to

$$F_1 = 0; \qquad \ldots \ldots (2)$$

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

$$F_j - D_j \geq F_i \forall (i,j) \epsilon H \quad \ldots..(3)$$

$$\sum_{i \, \epsilon \, S_t} R_{ik} \leq A_k \, , t = 1,2,3 \ldots F_n ; k = 1,2,3 \ldots . K \quad \ldots..(4)$$

Where ***H*** represents set of pairs of activities with precedence constraints and S_t represents the set of activities in progress in time interval ***[t - 1, t]:*** $S_t = \{ i \mid F_j - D_j \leq t \leq F_i \}$ (1) denotes an objective function (2) the finish time of dummy activty ***1*** (3) gives the precedence constraints (4) indicates the resource constraints for each resource type ***k.*** The project duration is minimized by minimizing the finish time of ending activity ***n***.

III.GENETIC ALGORITHM

Genetic Algorithm is a metaheuristic strategy use to solve combinatorial and hard optimization problem, It starts by generating an initial population of individual (solution) of some finite size, the individual are encoded by various solution representation scheme, Fitness value of each individual is calculated, A pair of individuals are selected by using some selection techniques and new individual are reproduced, few individuals undergo mutation. The procedure is continued till the defined number of generation

A. Chromosome Representation

The solution is encoded with activity list representation respecting the precedence constraints. Each activity in the representation is followed by its predecessor. Activity list is transformed into a schedule by a decoding procedure that is called the serial schedule generation scheme (SGS).

B. Initial Population

In order to initialize initial population containing individuals as described above, we will consider the construction of an activity list

1) Construction of Activity List: To generate an activity list as described in section *A*, we calculate the heuristic of each activity by using a convex combination with two parameters namely Due Date and Duration. Then the activities are used for making an individual of an initial population, an activity in the decision set with the lowest heuristic value is given the priority over the other activities if all the predecessors of that activity are in the scheduled set (i.e. are scheduled).

C. Selection

Selection Mechanism is a genetic method of survival of the fittest generally observed in a natural phenomenon. In this paper the fitness of all the individuals are calculated (i.e. makespan) and the individuals are sorted according to increasing value of the make span.

1) Elitist Strategy: After calculating the fitness values of the individuals, 50% of the best individuals in the population are carried over to the next generation and the remaining 50% individual recombine and form remaining individual of the next generation for recombination (i.e. crossover) individuals are selected randomly.

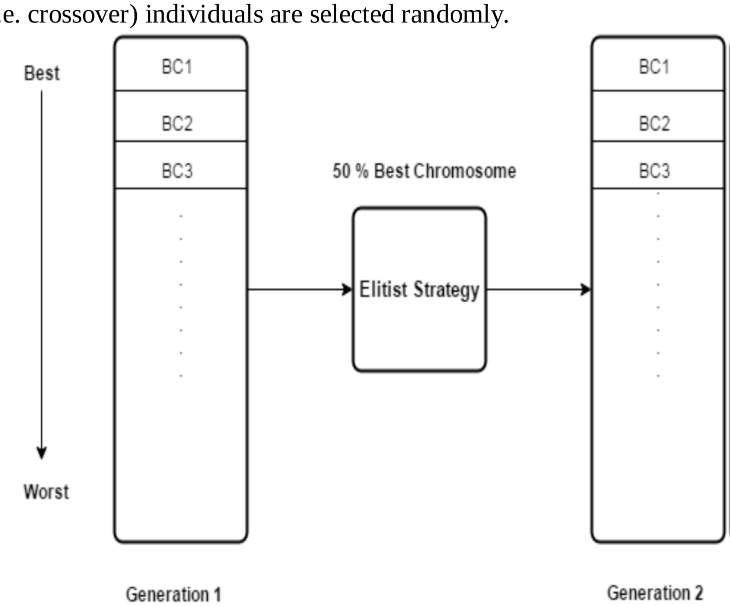


FIGURE 1.

Volume 4 Issue IX, September 2016
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

D. Crossover

Crossover is a technique of creating new individuals from existing individuals; it is synonyms to the reproduction technique in the natural phenomenon. The success of Genetic Algorithm lies in selection of a good crossover operator for creating new individuals.

1) *Two point Crossover:* Two random integers are drawn let us say $r_1$and r_2 such that $\mathbf{1 \leq r_1 < r_2 \leq J}$ which represent the position on the activity list representation of an individual.

Let us say $\mathbf{r_1=2}$ and $\mathbf{r_2= 5}$, after crossover

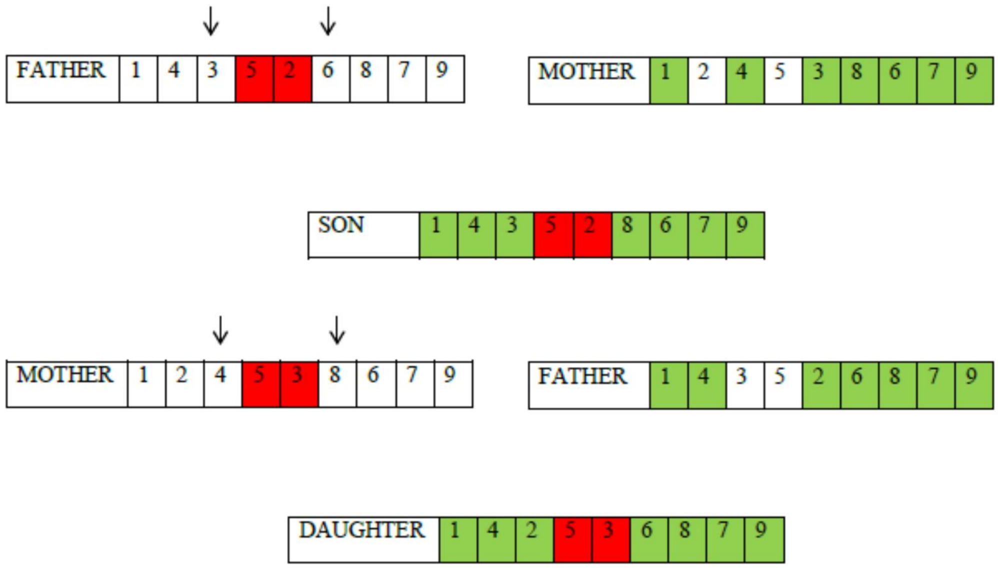


FIGURE 2.

E. Mutation

In an activity list representation an activity is randomly chosen with a mutation probability of P_m and it is swapped with another activity respecting a precedence constraints. Below diagram illustrates the mutation procedure

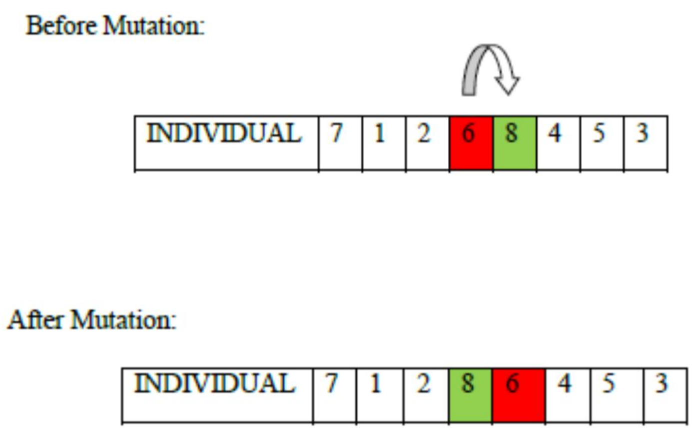


FIGURE 3.

IV. COMPUTATIONAL EXPERIMENT

A. Test Design

The experiments have been conducted on an Intel64 Family 6 Model 58 Stepping 9 Genuine Intel i3 Processor compatible personal computer with 2.4 GHz clock-pulse and 4 GB RAM. The GA has been coded in C++ 11 Programming Language, compiled with the Microsoft Visual C++ 2012 Compiler, and run under Windows 7 Operating System. As test instances, standard sets constructed by the project instance generator ProGen of Kolisch have been used. These instances can be downloaded in the project scheduling problem library PSPLIB TUM management website [14].

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

B. Computational Results

The algorithm solves each one of the instances with a maximum number of 1,000 schedules, computed as terminating condition. To study the performance of the algorithm different problem set instances were solved i.e. J30, J60 and J90 and % average deviation was calculated for each set. As the optimal solution has not been calculated for projects in J60 and J90, the average increase over an upper bound for each project have been measured i.e. the lowest makespan found by any of the heuristics. The results obtained are tabulated below

$P_{Crossover}$		0.8			0.85			0.9		
$P_{Mutation}$		0.03	0.04	0.05	0.03	0.04	0.05	0.03	0.04	0.05
J30	Avg. Dev	2.34	2.35	2.29	2.29	2.29	2.37	2.56	2.3	2.27
	Std. Dev	15.39	15.37	15.33	15.4	15.47	15.4	15.49	15.4	15.52
J60	Avg. Dev	4.97	5.06	5.06	4.96	4.84	4.85	4.86	4.8	4.81
	Std. Dev	22.75	22.83	22.81	22.72	22.49	22.62	22.6	22.52	22.5
J90	Avg. Dev	6.17	6.08	6.15	5.86	5.82	6.07	5.96	5.83	5.86
	Std. Dev	29.02	28.98	29.12	28.71	28.81	29.05	28.78	28.72	28.64

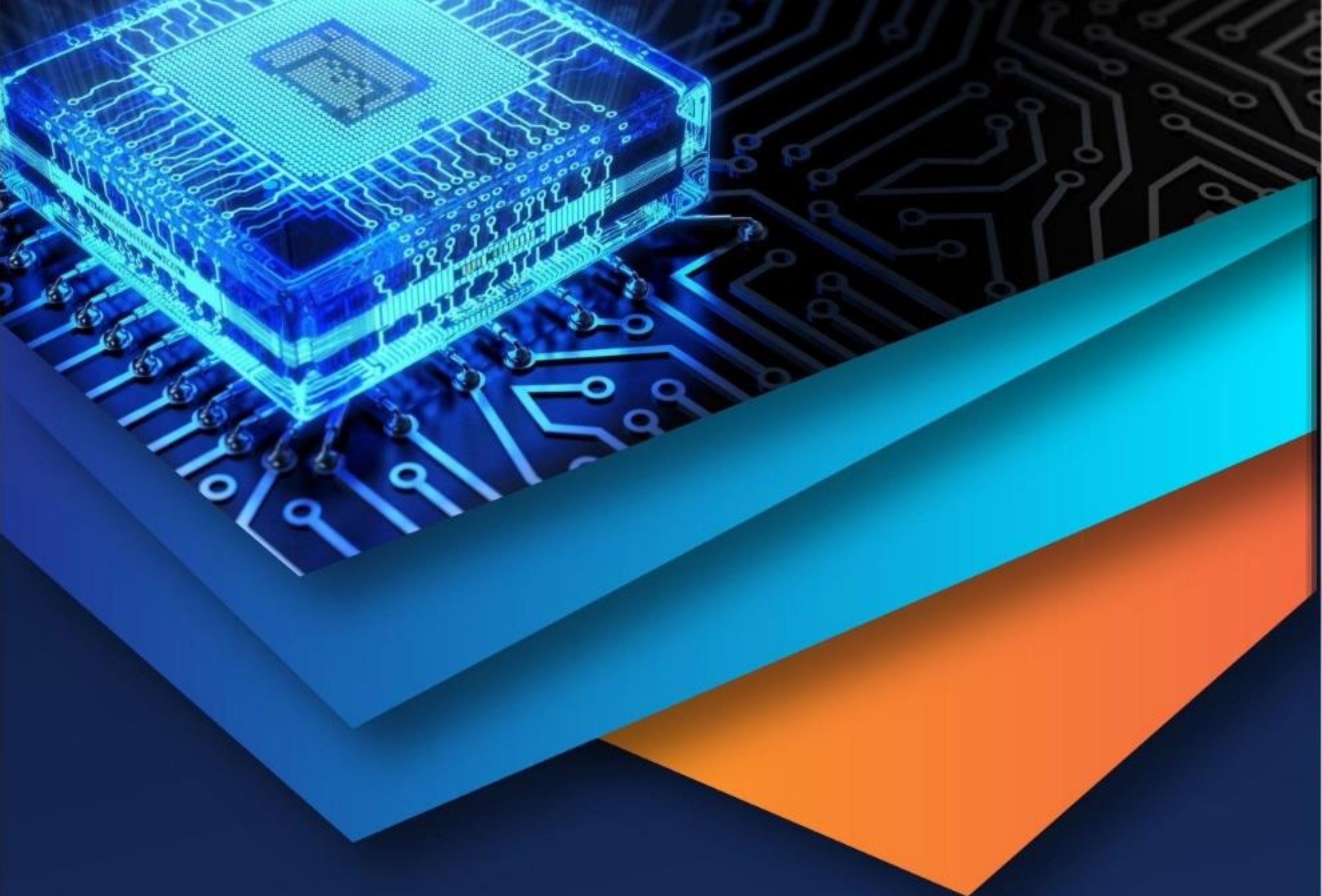
TABLE 1.

V. CONCLUSION

This research paper discusses RCPSP (Resource Constrained Project Scheduling Problem) in detail under the light of Literature review mentioning various works done so far in the view of providing the solution. A genetic algorithm has been proposed as a probable solution to the said problem. Effect of different parameters i.e. Mutation Probability and Crossover Probability in the process of finding out an optimal solution was studied. Simulations were conducted and results obtained were tabulated to show the % average deviation varying crossover Probability from 0.8 to 0.9. It was found that for a J30 problem set lowest % average deviation was 2.27 by taking mutation probability as 0.05 and crossover probability 0.09, while for a J60 and J90 problem set % average deviation was found to be 4.8 and 5.82 respectively, with Mutation Probability of 0.04 and crossover probability of 0.9 for J60, mutation probability 0.04 and crossover probability 0.85 for J90 problem set.

REFERENCES

[1] Demeulemeester, E., & Herroelen, W. (2002). Project scheduling-A research handbook. Vol. 49 of International Series in Operations Research & Management Science.
[2] Hartmann, S. (2012). Project scheduling under limited resources: models, methods, and applications (Vol. 478). Springer Science & Business Media.
[3] Alcaraz, J., & Maroto, C. (2001). A robust genetic algorithm for resource allocation in project scheduling. Annals of Operations Research, 102(1-4), 83-109.
[4] Hartmann, S. (1998). A competitive genetic algorithm for resource-constrained project scheduling. Naval Research Logistics (NRL), 45(7), 733-750.
[5] Hartmann, S. (2002). A self-adapting genetic algorithm for project scheduling under resource constraints. Naval Research Logistics (NRL), 49(5), 433-448.
[6] Lee, J. K., & Kim, Y. D. (1996). Search heuristics for resource constrained project scheduling. Journal of the Operational Research Society, 47(5), 678-689.
[7] Boctor, F. F. (1996). Resource-constrained project scheduling by simulated annealing. International Journal of Production Research, 34(8), 2335-2351.
[8] Bouleimen, K. L. E. I. N., & Lecocq, H. O. U. S. N. I. (2003). A new efficient simulated annealing algorithm for the resource-constrained project scheduling problem and its multiple mode version. European Journal of Operational Research, 149(2), 268-281
[9] Thomas, P. R., & Salhi, S. (1998). A tabu search approach for the resource constrained project scheduling problem. Journal of Heuristics, 4(2), 123-139.
[10] Hartmann, S., & Kolisch, R. (2000). Experimental evaluation of state-of-the-art heuristics for the resource-constrained project scheduling problem.European Journal of Operational Research, 127(2), 394-407.
[11] Kolisch, R. (1996). Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation. European Journal of Operational Research, 90(2), 320-333.
[12] Li, K. Y., & Willis, R. J. (1992). An iterative scheduling technique for resource-constrained project scheduling. European Journal of Operational Research, 56(3), 370-379.
[13] Özdamar, L., & Ulusoy, G. (1996). A note on an iterative forward/backward scheduling technique with reference to a procedure by Li and Willis.European Journal of Operational Research, 89(2), 400-407.
[14] Kolisch, R., & Sprecher, A. (1997). PSPLIB-a project scheduling problem library: OR software-ORSEP operations research software exchange program. European journal of operational research, 96(1), 205-216.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)