www.czasopisma.pan.pl PAN www.journals.pan.pl
POLSKA AKADEMIA NAUK

Management and Production Engineering Review
Volume 11 • Number 2 • June 2020 • pp. 107–112
DOI: 10.24425/mper.2020.133733

HYBRID GENETIC ALGORITHM FOR BI-CRITERIA OBJECTIVES IN SCHEDULING PROCESS

B.V. Raghavendra

Department of Mechanical Engineering, JSS Academy of Technical Education, India

Corresponding author:
B.V. Raghavendra
JSS Academy of Technical Education
Uttarhalli-Kengeri Road
Bangalore-560060, India
phone: +91 9481776070
e-mail: bvrjss@gmail.com

Received: 26 February 2019
Accepted: 14 November 2019

Abstract

Scheduling of multiobjective problems has gained the interest of the researchers. Past many decades, various classical techniques have been developed to address the multiobjective problems, but evolutionary optimizations such as genetic algorithm, particle swarm, tabu search method and many more are being successfully used. Researchers have reported that hybrid of these algorithms has increased the efficiency and effectiveness of the solution. Genetic algorithms in conjunction with Pareto optimization are used to find the best solution for bi-criteria objectives. Numbers of applications involve many objective functions, and application of the Pareto front method may have a large number of potential solutions. Selecting a feasible solution from such a large set is difficult to arrive the right solution for the decision maker. In this paper Pareto front ranking method is proposed to select the best parents for producing offspring's necessary to generate the new populations sets in genetic algorithms. The bi-criteria objectives minimizing the machine idleness and penalty cost for scheduling process is solved using genetic algorithm based Pareto front ranking method. The algorithm is coded in Matlab, and simulations were carried out for the crossover probability of 0.6, 0.7, 0.8, and 0.9. The results obtained from the simulations are encouraging and consistent for a crossover probability of 0.6.

Keywords

Multi-criteria decisions, genetic algorithm, Pareto method.

Introduction

Scheduling is one of the significant and very important processes in industry. Scheduling is the process of allocation of the resources for effective use of resources. The scheduling process for effective and efficient usage of the resources is studied widely, and many researches have been done in the field. Many researchers considered single objective criteria of minimization of the makespan [1]. However, researchers noticed that in practice, more than one objective function plays a major role and needs to be considered all the objectives for efficient use of the plant resources. The industries such as manufacturing of aircraft, electronics or semiconductors, etc., have a need to trade-off of the objective functions where multiple objective needs to be considered in order to optimize the overall system performance. Obviously, the multi-objective criteria are more intricate than the scheduling with one criteria, and it is difficult to obtain a balanced solution due to the objectives are incompatible or conflicting with each others. Optimizing any one objective generally leads to worsening the solution of another objective. The need to improve the overall system performance for a better solution of the multiobjective criteria and its complexity, drawn the interest to solve the multiobjective problems.

Literature review

Since 1970, researchers are working on multiobjective related optimization problems and its applications in the various fields. In multi-objective problems, overall system performance will be improved by simultaneously considering more than one objective [2]. Decision making in the real world related problems gain the interest and involve multiple and incompatible objectives that need to be addressed while considering the various constraints. A single solution may not exist in multi-objective problems, which is not an excellent solution with respect to the objectives considered. In search of a better solution in the larger space considering stated objectives, non -dominated solution would yield a good result but poor in one or more objectives.

A set of optimal solution is possible to obtain in multi-objective optimizations which are having conflicting objective functions. The optimal set includes the best solution with respect to all objective functions. This set of optimal or non dominated solution in multi-objective optimization (MOPs), is called Pareto optimal solution which is proposed by Vilfredo Pareto in 1896.

Due to the complexity in solving multi-objective optimization problems using deterministic techniques, evolutionary process using computer program would reach the compromised values of various objective functions. Meta-heuristics search tools such as genetic algorithms, simulated annealing, tabu search, and particle swarm optimization have become gained the interest of the researchers to solve the problems with complexity in the multi-objective functions. Significant attention has been given to genetic algorithm (GA) with respect to complexity in the scheduling process. The scheduling in the genetic algorithm domain has the vital research in the field of artificial intelligence and operational research [3]. Researchers have established the four approaches to solve the multi-objective scheduling problems such as:

- Weighting objective method: A single objective function has obtained by combination of weighted values of the objective functions considered in the optimization problems. This method needs the additional information about the comparative importance of the objective function or weights of the decision makers.
- Hierarchical optimization method: In this method ranking the objective functions in decreasing order of their importance by the decision maker. Every individual objective function is then minimized considering to a constraint that prevent the minimum of the new function to exceed minimum of the previously obtained functional value.
- Goal programming method: This method expresses the satisfying goal of the objective functions by considering constraints. The method aim is to find a good solution of pre-defined goals for each objective.
- Pareto approach: The Pareto approach generates the complete non-dominated set or to approximate a set of good solutions. In the Pareto method, improvements is a change in the allocation that makes at least one solution or preference objective function is better off without making any other individual objective worse off.

In the last twenty years, evolutionary algorithms have demonstrated the advantage in obtaining solutions to the MOPs. Researchers have developed various approaches in solving multi-objective problems through evolutionary algorithms (MOEAs). Goldberg [4] recommended the application of the Pareto front method [5] to arrange a set of solutions for MOPs. Pareto method is applied to rank the quality of the solutions which consists of multi-objective values; the procedure to sort out the solution is called non-dominated sorting. In non-dominated sorting, the values of the objective functions are compared and divided into a number of ranks. Smaller ranks in the solution are better than those in a larger rank. In 1995 the non-dominated sorting is first adopted in NSGA [6] then it has become commonly adopted in MOEAs. The most important concerned in the MOEAs are effectiveness and efficiency of the algorithm. In the existing MOEAs, environmental selection is adopted for non-dominated sorting, and additional criteria are suggested to distinguish the solution of the same rank.

Among the many-objective optimization approaches, sorting of non-dominated method has gained the interest and used by many researchers in solving multiobjective problems through evolutionary algorithms (MOEAs). Pareto dominance phenomenon [7], deteriorates the effectiveness of the many-objective optimization problems (MaOPs). Researchers have adopted Pareto non-dominated method to solve MOEAs to address MaOPs, GrEA [8], NSGA-III [9], KnEA [10], and LMEA [11]. Some research papers have reported that decomposition based MOEAs with dominance used in the non-dominated sorting gives the most effective solution in MOEA/DD [12], and BCE-MOEA/D [13]. Due to the efficiency and computational cost of the non-dominated sorting, its application in the MOEAs becomes another issue. It is experimented of NSAG-II consumes more runtime when the population size of 1000 and maximum of 500 generations in the case to solve a bi-criteria objectives DTLZ1. Literatures are reported that the computational cost will be higher in the case of the larger population size and number of objective functions considered. Hence, it becomes an important issue to address the computational cost and efficiency to improve the non-dominated sorting algorithms. Some of the improved algorithms are considered non-dominated ranking approach, Jensen's sort, deductive sort [14], and an efficient non-dominated sorting (ENS) [15]. Researchers have developed few tailored non-dominated sorting algorithms to solve MaOPs, such as corner sort [16], T-ENS [11], and A-ENS [17]. Though the many algorithms developed to address the effectiveness and efficiency of non-dominated sorting, much focus has not been devoted to analyze them in MOEAs, particularly for solving MaOPs.

The Pareto optimization process could be used in the evolutionary methodology. Algorithms such as the genetic algorithm apply evolutionary processes to generate non-dominated set of solutions by the biologically inspired evolutionary method. The solution obtained with the genetic algorithm multiobjective method may not be Pareto optimal, but the algorithm is designed to evolve a good solution to approach the Pareto front to get diversity existing in the Pareto front in order to obtain a reasonably good solution.

At the end of the Pareto analysis, decision makers are needed to select a only one solution from the available large number of sets of optimal solutions. Several researchers have developed various approaches to help the decision maker for selection of the best solution. In the various approaches convergence and diversity are two criteria which need to be balanced to obtain an efficient Pareto front. Two approaches such as one-at-a-time strategy and simultaneous strategy have been used to address the problems [18] in the Pareto -optimal front analysis.

In Fig. 1, the population is ranked in three ranks based on non-dominated sorting. The solutions of each rank are mutually non-dominated; in the population of P, the solutions of first rank cannot be dominated by any other solution in the population; with non-dominated sorting, the quality of results in a population can be greatly distinguished, and this approach has been generally adopted in MOEAs [19].

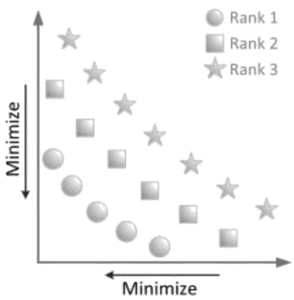

Fig. 1. Non dominated sorting.

General proposed algorithm

In this method randomly generated the population size of 'N' is generated initially. Bi-criteria objective functions, total idleness of the machine cost, and penalty cost for each population is determined and assigned Pareto dominance rank based on non-dominated sorting. Then the priority rank is assigned to every population based on Pareto dominance rank and also its position with respect to the other solutions in the same rank. Parent population for off-spring generation is chosen based on dominance rank. New offsprings are generated using crossover and mutation operator. The population of size N for the next subsequent generation is equal to the selected parent population (based on Pareto rank) and offspring generated. The process is repeated till to the given number of generations/iterations.

General algorithm framework

Input parameters: Population = P, Population size = N:

1) P ← Start the population of size N.
2) Determine Objective functions.
3) F ← Nondominated_sorting (P)
4) Select the best parents based on priority rank. If multiple solutions have same rank then based on the combined objective function the solutions will be selected
5) Generate offsprings ← Crossover operator and Mutation operator.
6) Population size N for next generations ← combining best parents and offspring
7) Repeat 2 to 6 till termination of the algorithm.

Problem formulation

The problem has been formulated considering the data set for experimentation as mentioned in the Table 1. The experimentation has been carried out on parallel machines considering two objective functions such as minimizing the idleness of the machine, and the penalty cost. The algorithm is designed and coded in Matlab to meet the stated objective functions. The input parameters considered for the experimentation are a number of parallel machines (m), the number of setups (Si), J-number of jobs/part types and batch quantity (bj) for each job type. The experimentation is carried out on six parallel machines. The numbers of setups are required to complete the operations are three and ten part types with a batch quantity of '10' are assumed.

Table 1
Data set for experimentation.

Part type (J_i)		1	2	3	4	5	6	7	8	9	10
Operations time in min.	Set up-I	62	53	38	34	32	33	31	75	6.78	17.34
	Set up-II	44	46	38	31	19	31	30	–	–	5.15
	Set up-III	2	–	20	10	10	9	16	–	–	15
Due Day (dbJ_i)		2	2	2	2	2	1	1	1	1	1
Penalty Cost Rs./Day/Batch (pbJ_i)		10	12	12	13	9	11	11	14	8	10

In this work the genetic algorithm is used to determine the best solution for bi-objective criteria based on Pareto front non-dominated sorting method. Initially, population size of 20 are randomly generated. Each population/chromosome is calculated for objective functions and the first best 12 parents have been selected for the next generation based on Pareto front non-dominated sorting. Since the scheduling problem is based on NP hard, combinatorial type and hence partially mapped cross operator is used followed by mutation operator. A large number of trials were conducted to decide and found that the algorithm yields a good solution within the 30 generations. The experimentation was conducted for various crossover probabilities of 0.6, 07, 0.8 and 0.9.

Results and discussion

The study was carried out for various crossover probabilities of 0.6, 0.7, 0.8 and 0.9 and the results are presented in Figs 2 to 7. The number of iterations was conducted for 30 generations and found that the solutions were not converged after 23rd iteration. The results for bi-criteria objective functions of minimizing cost of machine idleness and penalty cost were presented and found that the steady state pattern of results for the crossover probability of 0.6 till to the generation of 17 and further it is converged in the case of machine idleness cost, then the steady state till the end of the generation. The results of objective functions for crossover probability 0.7 and 0.9 are shown and analyzed that it exhibits an unsteady pattern of solutions. The results shown in the graph are steady till 23rd generation and further found non-convergence in the case of cross over probability of 0.8. In Fig. 6, number of non-dominated fronts based on Pareto sorting for each generation are presented. Figure 7 shows the total cost for crossover probability of 0.6, 0.7, 0.8 and 0.9. In the Fig. 6, it is evident that the number of non dominated fronts is varied at a considerable extend from generation to generation and does not yield steady state pattern of results. The crossover probability of 0.6 is giving the best solution for both the objective functions and also better results for a combined cost of both the objective functions. The experiments were conducted on Intel R Core(TM) I 7-6700 CPU @3.4GHZ. The designed algorithm helps the decision maker to analyze the bi-criteria objectives more effectively.

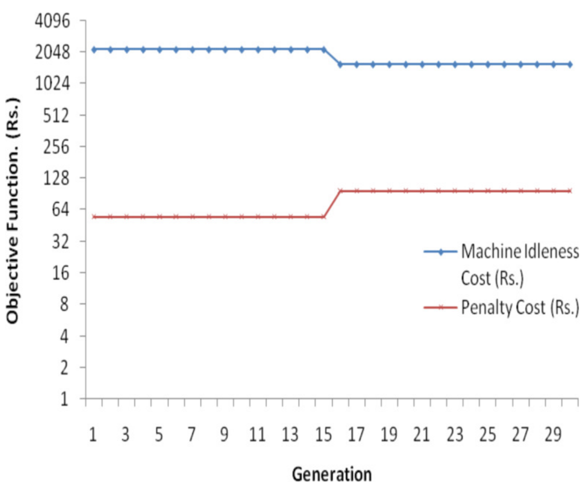

Fig. 2. Crossover probability = 0.6.

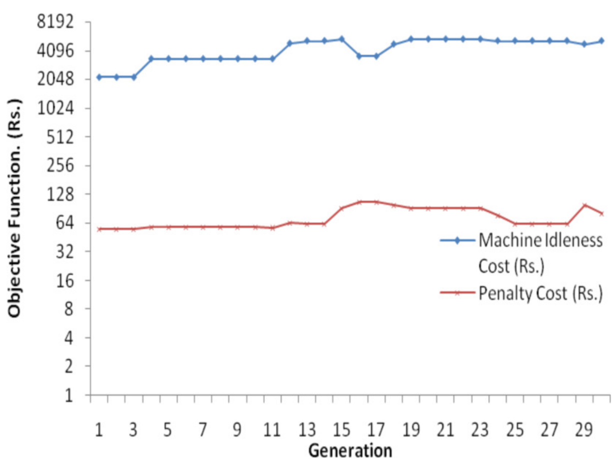

Fig. 3. Crossover probability = 0.7.

www.czasopisma.pan.pl PAN www.journals.pan.pl

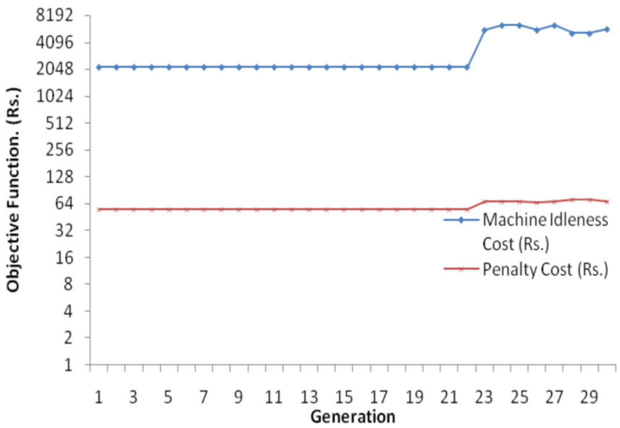

Fig. 4. Crossover probability = 0.8.

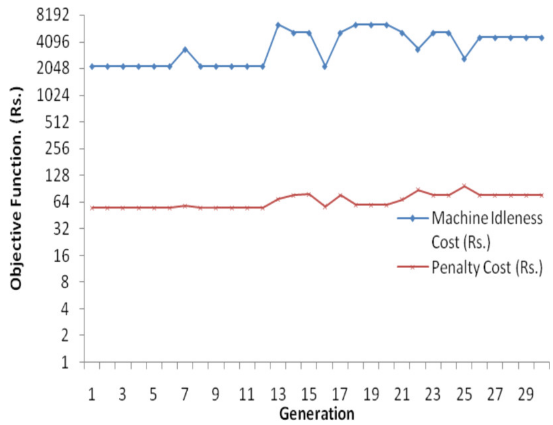

Fig. 5. Crossover probability = 0.9.

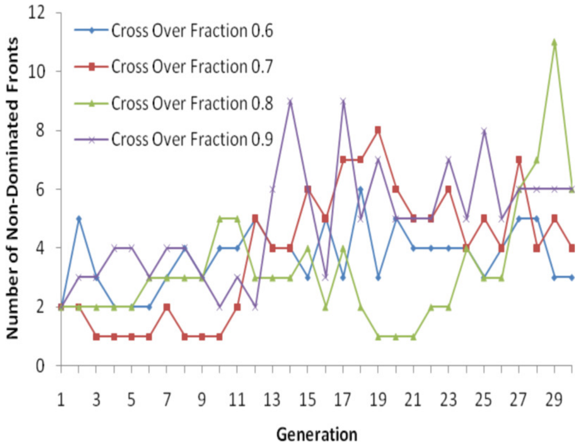

Fig. 6. Number of non-dominated fronts.

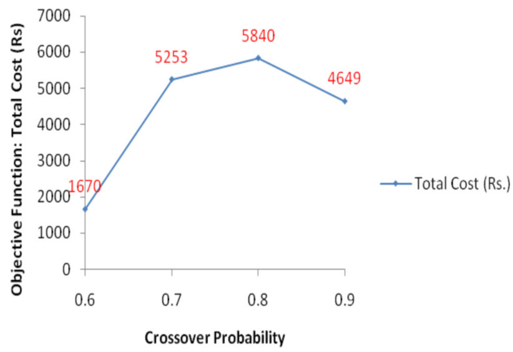

Fig. 7. Crossover probability and total cost.

References

[1] Wang X., Zhang C., Gao L., Li P., *A survey and future trend of study on multi-objective scheduling*, 2008 Fourth International Conference on Natural Computation, 6, 382–391, 2008.

[2] Reddy B.S.P., Rao C.S.P., *A hybrid multi-objective GA for simultaneous scheduling of machines and AGVs in FMS*, International Journal of Advanced Manufacturing Technology, 31, 5–6, 602–613, 2006.

[3] Gen M., Lin L., *Multiobjective genetic algorithm for scheduling problems in manufacturing systems*, Industrial Engineering & Management Systems, 11, 4, 310–330, 2012.

[4] Goldberg D.E., *Genetic algorithms in search, optimization, and machine learning*, Addison-Wesley, Boston, 1989.

[5] Miettinen K., *Nonlinear multiobjective optimization*, Springer, New York, 1999.

[6] Srinivas N., Deb K., *Multiobjective optimization using nondominated sorting in genetic algorithms*, Journal of Evolutionary Computation, 2, 3, 221–248, 1995.

[7] Li B., Li J., Tang K., Yao X., *Many-objective evolutionary algorithms: a survey*, ACM Computing Surveys, 48, 1, 13, 2015.

[8] Yang S., Li M., Liu X., Zheng J., Â grid-based evolutionary algorithm for many-objective optimization, IEEE Transactions on Evolutionary Computation, 17, 5, 721–736, 2013.

[9] Deb K., Jain H., *An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach. Part I: solving problems with box constraints*, IEEE Transactions on Evolutionary Computation, 18, 4, 577–601, 2014.

[10] Zhang X., Tian Y., Jin Y., *A knee point driven evolutionary algorithm for many-objective optimization*, IEEE Transactions on Evolutionary Computation, 19, 6, 761–776, 2014.

[11] Zhang X., Tian Y., Cheng R., Jin Y., *A decision variable clustering-based evolutionary algo-*

rithm for large-scale many objective optimization, IEEE Transactions on Evolutionary Computation, 99, 1–1, 2017, doi: 10. 1109/TEVC.2016.2600642.

[12] Li K., Deb K., Zhang Q., Kwong S., *An evolutionary many-objective optimization algorithm based on dominance and decomposition*, IEEE Transactions on Evolutionary Computation, 19, 694–716, 2015.

[13] Li M., Yang S., Liu X., *Pareto or non-Pareto: bi-criterion evolution in multi objective optimization*, IEEE Transactions on Evolutionary Computation, 20, 5, 645–665, 2016.

[14] McClymont K. Keedwell E., *Deductive sort and climbing sort: new methods for non-dominated sorting*, IEEE Transactions on Evolutionary Computation, 20, 1, 1–26, 2012.

[15] Zhang X., Tian Y., Cheng R., Jin Y., *An efficient approach to non-dominated sorting for evolutionary multi-objective optimization*, IEEE Transactions on Evolutionary Computation, 19, 2, 201–213, 2015.

[16] Wang H., Yao X., *Corner sort for Pareto-based many-objective optimization*, IEEE Transactions on Cybernetics, 44, 1, 92–102, 2014.

[17] Zhang X., Tian Y., Jin Y., *Approximate non-dominated sorting for evolutionary many-objective optimization*, Information Sciences, 369, 14–33, 2016.

[18] Chaudhari P.M, Dharaskar R.V., Thakare V.M., *Computing the most significant solution from Pareto front obtained in multi-objective evolutionary*, International Journal of Advanced Computer Sciences and Applications, 1, 4, 63–68, 2010.

[19] Tian Y., Wang H., Zhang X., Jin Y., *Effectiveness and efficiency of non-dominated sorting for evolutionary multi- and many-objective optimization*, Complex & Intelligent Systems, 3, 4, 247–263, 2017.