Volume 5, Issue 9, September 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Multi-objective based Event Based Project Scheduling Model Using EMBACO Algorithm

Vidya Sagar Ponnam[1],Dr.N.Geethanjali[2]
[1]Research Scholar, [2]Associate Professor ,
[1, 2] Dept. of Computer Science & Technology, Sri Krishnadevaraya University,
Ananthapuram, Andhra Pradesh, India

Abstract— In this paper, an optimization model was implemented to find the variation of the project dynamic activities and its resource constraint problem. Activities overlapping or resource overlapping is one of the major problem in multi-objective based resource constraint.In the traditional approaches overlapping problem is resolved without considering a whole project with resource constraints.Each project schedule should affect the overlapped activities or resources which are associated with delivering the multiple activities on time.In this proposed work,multiple activites and multiple resources are scheduled without overlapping problem in parallel project scheduling process. Proposed model partitions the large set of projects into subprojects with a limited number of resources and then integrating optimal subprojects to create a feasible solution. Experimental results show that proposed model performs well against the planners to determine an optimal solution with specified resource constraints using various time bounds and decision making.

Keywords— Project scheduling, Multiobjective, Resource Allocation,Probability estimation, Complex dataset.

I. INTRODUCTION

In practical applications, activity resources are limited as well as execution time of some work in project.Traditional activity or resource overlapping problem associated with both resource constraints and precedence relationships in order to minimize the project overall duration. An RCPSP instance consists of a project, which is a set of activities with precedence constraints, and resources. There can be any finite number of resources, each with a finite capacity. The activities have a certain duration, and during that duration they require a certain amount of some of the resources which are released afterwards. Each precedence constraint dictates an ordering between one pair of activities, meaning that one cannot start before the other activity has finished. The goal is to create a schedule that satisfies all constraints and minimizes the makespan. A Partial Order Schedule is another type of schedule which does not provide fixed start times for activities. Instead, a POS defines start times of activities relative to each other. More specific: it uses precedence constraints to only specify the ordering of some pairs of activities. When the capacity of a resource is reduced, it cannot be guaranteed that the set of solutions represented by the POS still contains a valid solution. This is because the POS does not contain information on the resources; only the solutions to possible resource conflicts, or which activities to order sequentially and in what order, are retained. Any information about the exact resource usage is not present anymore, so it is impossible to know from the POS alone whether a solution is still valid with this modification. This can be checked using the original RCPSP instance, but ensuring a valid solution—constructing a new, valid solution when another solution is invalid — requires a new scheduling effort, effectively creating a new Partial Order Schedule.

Traditional scheduling methods, such as PERT (The Program Evaluation and Review Technique) and CPM (Critical Path Method), are not enough for production scheduling, because they consider infinite re-sources, i.e., they cannot take resource constraints into account [3]. Scheduling with infinite resources may give results, which are not feasible. Durations of the tasks are assumed to follow a beta distribution. To compute the expected makespan, they sample task durations from the beta distribution and then apply the suggested tabu search to the drawn task durations. At the end, the expected makespan is estimated as the mean of all the computed makespans. This method violates the requirement of being non-anticipating because in each scheduling sample, it is assumed that the durations of all tasks are known at the beginning of the
scheduling process.

The mapping meta-tasks on a machine are shown to be NP-complete. By using heuristic approach the NP-complete problem can be solved. The processing time and requirements of all applications are assumed to be stochastic. Any given problem is to be scheduled in a given multiprocessor system using multiprocessor scheduling problem to minimize program's execution time and the last task must be completed as soon as possible. For constrained optimization, genetic algorithm is one of the approaches which is widely used to schedule tasks. Execution of genetic algorithm can be optimized with the knowledge implementation of the scheduling problem. In this traditional approach, the challenge of execution, completion time and the precedence order in the parallel processing system were resolved by using the concept of Top –level or bottom-level.It is key to apply an enhanced scheduling algorithm for task sequencing or proper resource allocation on multiprocessors in order to reduce the computing power and to increase scheduling performance in a parallel computing system. The majority of the traditional approaches is still required to execute under resource constraint systems and not capable in normal task scheduling environments. Even moderate scheduling tasks cannot be resolved within the specified execution time by using these techniques[2-5]. These heuristics is to optimize the time and space complexity.The considerable weakness of this heuristic is that they usually apply a greedy mechanism on the scheduling factors such as the structure of the input task graphs and the number of available target processors. Enhancing performance of a heuristically increases its complexity . The static scheduling used to minimize the total project completion time with a limited number of resources and fixed task duration. Static scheduling doesn't consider dynamic task allocation as well as dynamic resource scheduling. A task precedence graph or DAG works accurately for most static and limited parallel task scheduling applications since it depends on the resource and duration dependencies between tasks. A feasible schedule satisfies all of the constraints. An optimal schedule not only satisfies all of the constraints, but also is at least as good as any other feasible schedule. Goodness is defined by the objective measures. When modeling the problem it is often convenient to think of objectives and constraints as equivalent, but when solving the problem they must be treated differently.

II. RELATED WORK

Khalizadeh et al presented a metaheuristic algorithm based on a modified Particle Swarm Optimization (PSO) approach introduced by Tchomte and Gourgan which uses a modified rule for the displacement of particles for the multi- mode resource constrained project scheduling problem with minimization of total weighted resource tardiness penalty cost (MRCPSP-TWRTPC). This problem involves for each activity, both renewable and non-renewable resource requirements depending on activity mode. A multimode particle swarm optimization which combines with genetic operator to solve a bio-objective MRCPSP with positive and negative cash flows was developed by Kazemi and Tavakkoli- Moghaddam. Zhang proposed an Ant colony optimization (ACO) algorithm with its effectiveness and efficiency justified through a series of computational analyses. Van Peteghem and Vanhoucke (2011) proposed a scatter search algorithm which is among the best performing competitive algorithms in the open literature after they had in 2009 presented artificial immune systems, a new search algorithm inspired by the mechanisms of a vertebrate immune system performed on an initial population set. The AIS algorithm proves its effectiveness by generating competitive results for the different PSPLIB datasets. Ranjbar et al. (2009) proposed a hybridized scatter search procedure to solve the MRCPSP. The author defines two important problem classes for project scheduling. The first problem, problem A, is defined as a problem of scarce resources where the objective is finding the shortest project duration with a given amount of resources. The second problem, problem B, is defined as a problem of scarce time where the objective is finding the least cost resource requirements within a given project duration. The author uses graph theory to represent and solve these project scheduling problems where they used a special "dual" relation between the scarce resource and scarce time problems.

Once the general resource capacities are decided upon in RPP, the problem becomes the Resource Dedication Problem (RDP), a multi-project problem environment with given resource capacities under RD policy. RDP is defined as the optimal dedication of resource capacities to different projects within the overall limits of the resources and with the objective of minimizing a predetermined objective function.In multi resource constrained project scheduling, each task may require a set of activities or a set of successive operations. For a given activity,several resources may execute in a parallel manner, which means the task can consider any one of the available resources for processing.These issues are often known as machine scheduling problems[3]. In SM-RCPSP, multiple task specifies a minimum and maximum time delay between the project activities. The minimum time delay represents that a task can only start or finish when the predecessor task has already finished in a given time. The conventional scheduling methods such as program evaluation and review technique and critical path method are not enough for multitasking, because they generate infinite schedules which cannot take resource constraints into consideration. Traditional methods generate a large set of feasible and infeasible solutions.

III. PROPOSED ALGORITHM

Initialization: $g = 1,\ t_1 = 0,\ A_0 = \{0\},\ \Gamma_o = \{0\},\ S_0 = \{0\},\ RD_k(0) = R_k\ \ (k \in K)$

while $|S_g| < n + 2$ **repeat**

{

Update E_g

while $E_g \neq \{\}$ **repeat**

{

Select activity with highest priority

$$j^* = \underset{j \in E_g}{\operatorname{argmax}} \{ PRIORITY_j \}$$

Calculate earliest finish time (in terms of precedence only)

$$EF_{j^*} = \max_{i \in P_j} \{F_i\} + d_{j^*}$$

Calculate the earliest finish time (in terms of precedence and capacity)

$$F_{j^*} = \min \left\{ t \in \left[FMC_{j^*} - d_{j^*}, \infty \right] \cap \Gamma_g \mid r_{j^*,k} \leq RD_k(\tau),\ k \in K \mid r_{j^*,k} > 0,\ \tau \in \left[t,\ t + d_{j^*} \right] \right\} + d_{j^*}$$

Update $S_g = S_{g-1} \cup \{ j^* \}$, $\Gamma_g = \Gamma_{g-1} \cup \{ F_{j^*} \}$

Iteration increment: $g = g + 1$

Update $A_g,\ E_g,\ RD_k(t) \mid t \in \left[F_{j^*} - d_{j^*} , F_{j^*} \right],\ k \in K \mid r_{j^*,k} > 0$

}

Determine the time associated with activity g

$$t_g = \min \{ t \in \Gamma_{g-1} \mid t > t_{g-1} \}$$

}

Finite mixture models are used in many applications for project activities density estimation and model based similar activities grouping. In many cases, the number of activities of the mixture model is unknown ad needs to be estimated from the project data. An expectation maximization algorithm which allows to find local maxima of the likelihood function. The traditional EM approach suffers with inconsistent convergence rates across the different areas to locate local optimum.

Optimized ExpectationMaximization ACO:

EM Algorithm

1. **Initialization:** Choose means at random
2. Set up a model;
3. Estimate the model parameters a_k, μ_k, and Σ_k such that the likelihood of the data under these parameters is maximized;
4. Based on the estimated parameters, divide data points into clusters according to the posterior probabilities of the cluster labels given the observed data.
5. **E step:**
    a. For all points and means, compute Prob(point|mean)
    b. Prob(mean|point) =Prob(mean) Prob(point|mean) / Prob(point)

$$p_{ij} \equiv p(j|\mathbf{x}_i) = \frac{p(\mathbf{x}_i|j)c_j}{\sum_{k=1}^{K} p(\mathbf{x}_i|k)c_k} = \frac{p(\mathbf{x}_i|j)c_j}{p(\mathbf{x}_i)}.$$

6. **M step:**
    a. Each mean = Weighted avg. of points
    b. Weight = Prob(mean|point)

$$c_j^{\text{new}} = \frac{1}{\ell}\sum_{i=1}^{\ell} p(j|\mathbf{x}_i) = \frac{p_j}{\ell},$$

$$\boldsymbol{\mu}_j^{\text{new}} = \frac{\sum_{i=1}^{\ell} p(j|\mathbf{x}_i)\mathbf{x}_i}{\sum_{i=1}^{\ell} p(j|\mathbf{x}_i)} = \frac{\sum_i p_{ij}\mathbf{x}_i}{\sum_i p_{ij}} = \frac{\sum_i p_{ij}\mathbf{x}_i}{\ell c_j^{\text{new}}},$$

$$\boldsymbol{\Sigma}_j^{\text{new}} = \frac{\sum_{i=1}^{\ell} p_{ij}(\mathbf{x}_i - \boldsymbol{\mu}_j^{\text{new}})(\mathbf{x}_i - \boldsymbol{\mu}_j^{\text{new}})^{\mathrm{T}}}{\ell c_j^{\text{new}}}.$$

7. Repeat until convergence

Experimental Results

The proposed system implemented on PSPLIB data set with a large number of project activities.From the experimental results it has been found that performance of the proposed work is better than traditional activities selection models.Proposed approach efficiently minimizes the search space due to the correlation between the project activities.This system helps to find the local minima of the time duration for resource allocation.The computational time required to find the best solution is significantly minimized by using the optimization function.Efficient performance of the proposed algorithm is checked accordingly by using the objective function,makespan and project constraints.

Sample Data:

PRECEDENCE RELATIONS:

jobnr.	#modes	#successors	successors
1	1	3	2 3 4
2	3	2	8 15
3	3	3	6 8 10
4	3	3	5 12 14
5	3	3	7 10 11

REQUESTS/DURATIONS:

jobnr.	mode	duration	R 1	R 2	N 1	N 2
1	1	0	0	0	0	0
2	1	1	8	0	8	0
	2	1	7	0	0	6
	3	8	0	7	8	0
3	1	3	10	0	0	5
	2	7	0	9	0	4
	3	10	6	0	0	4
4	1	3	9	0	7	0

2 6 0 8 4 0
3 9 5 0 2 0
5 1 1 3 0 10 0
2 2 2 0 7 0
3 10 0 2 3 0
RESOURCEAVAILABILITIES:
R 1 R 2 N 1 N 2
11 11 54 62

Results:

Table 1: Job performance with traditional algorithms

JOBS	Resources	PSO-ACO	Event-ACO	ProposedAvgTime(secs)
50#	12	217	311	135
100#	27	388	356	194
150#	31	521	519	419
200#	43	785	712	642
250#	53	1233	1089	727

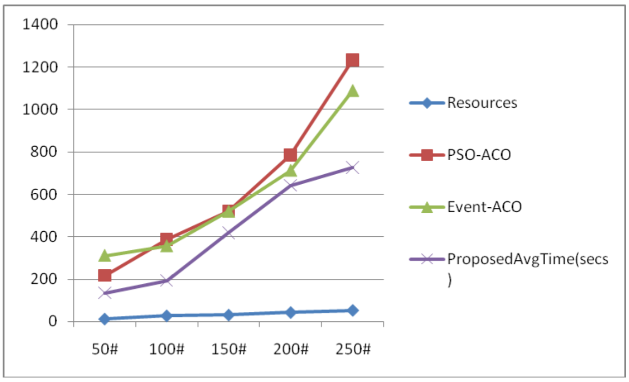

Fig 4: Performance analysis of traditional scheduling algorithm with proposed algorithm

Table 2: Proposed resources vs Activities Overlap Allocation Time

JOBS	Resources	ActivitiesOverlapTime
50#	12	21
100#	27	30
150#	31	38
200#	43	71
250#	53	87

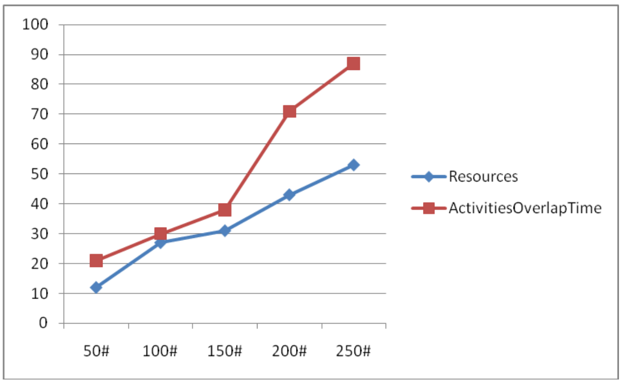

Fig 52: Proposed resources vs Activities Overlap Allocation Time

V. CONCLUSION

Activities overlapping or resource overlapping is one of the major problem in multi-objective based resource constraint.In the traditional approaches overlapping problem is resolved without considering a whole project with resource constraints.Each project schedule should affect the overlapped activities or resources which are associated with delivering the multiple activities on time.In this proposed work,multiple activites and multiple resources are scheduled without overlapping problem in parallel project scheduling process. Proposed model partitions the large set of projects into subprojects with a limited number of resources and then integrating optimal subprojects to create a feasible solution. Experimental results show that proposed model performs well against the planners to determine an optimal solution with specified resource constraints using various time bounds and decision making.

REFERENCES

[1] K. C. Ying, S. W. Lin, and Z. J. Lee, "Hybrid-directional planning: Improving improvement heuristics for scheduling resource-constrained projects," International Journal of Advanced Manufacturing Technology, vol. 41, pp. 358– 366, 2009 .

[2] L. Deng, Y. Lin, and M. Chen, "Hybrid ant colony optimization for the resource-constrained project scheduling problem," Journal of Systems Engineering and Electronics, vol. 21, pp. 67–71, 2010.

[3] K. Moumene and J. A. Ferland, "Activity list representation for a generalization of the resource-constrained project scheduling problem," European Journal of Operational Research, vol. 199, pp. 46–54, 2009.

[4] P. Brucker and S. Knust. A linear programming and constraint propagationbased lower bound for the RCPSP. European Journal of Operational Research, 127:355–362, 2000.

[5] J. Carlier and E. N´eron. A new lp based lower bound for the cumulative scheduling problem. Technical report, Universit´e Technologique de Compi`egne, 2001.

[6] Colak S. Agarwal A. Erenguc S. ;"Resource Constrained Scheduling Problem: A Hybrid Neural Approach"; s/f; Recovered from: http://www.cba.ufl.edu/dis/docs/papers/ResourceConstrainedProjectSchedulingAHybridNeuralApproach. pdf ; Last viewed on : March de 2006.

[7] Stinson, J.P., Davis, E.W. and Khumawala, B.M., "Multiple Resourceconstrained Scheduling Using Branch-and-Bound", AIIE Transactions, Vol. 10 No. 3, 1978, pp. 252 [10] Xinggang Luo 1,2, Dingwei Wang 2, Jiafu Tang 2, Yiliu Tu 3ResourceConstrained Software project Scheduling Problem , Proceedings of the 6th World Congress on Intelligent Control and Automation, June 21 23, 2006, Dalian, China.

[8] O. Bellenguez and E. Ne´ron, "Methods for the Multi-Skill Project Scheduling Problem," Proc. Ninth Int'l Workshop Project Management and Scheduling, 2004.

[9] P. Brucker, A. Drexl, R. Mohring, K. Neumann, E. Pesch, "Resource-Constrained Project Scheduling: Notation, Classification, Models and Methods," European J. Operational Research, 1999.

[10] O. Bellenguez and E. Ne´ron, "A Branch-and-Bound Method for Solving Multi-Skill Project Scheduling Problem," RAIRO-Operations Research, 2007.