



# Review of Solving Software Project Scheduling Problem with Ant Colony Optimization

K.N.Vitekar<sup>1</sup>, S.A.Dhanawe<sup>2</sup>, D.B.Hanchate<sup>3</sup>

ME 2<sup>nd</sup> Year, Dept. of Computer Engineering, VPCOE, Baramati, Pune, Maharashtra, India<sup>1</sup>

ME 1<sup>st</sup> Year, Dept. of Computer Engineering, VPCOE, Baramati, Pune, Maharashtra, India<sup>2</sup>

Assistant Professor, Dept. of Computer Engineering, VPCOE, Baramati, Pune, Maharashtra, India<sup>3</sup>

**ABSTRACT:** SPSP is a problem of scheduling the task and employee. SPSP is a NP-hard (Non Polynomial) problem. SPSP is a problem which is related to RCPSP problem. For solving such problem number of model has been developed. Number of Meta heuristic algorithm is also applied to solve such problem (e.g. GA). This paper presents the survey of methods and models that are put into the historical context. SPSP split the task and distribute dedication of employee to task nodes. Author proposes an ACO Meta heuristics approach to solve the SPSP problem. Author use ACO for solving such problem hence he called it as an ACS: SPSP. Result of this paper is compared with GA to solve SPSP. The proposed algorithm is very efficient and promising and obtains more accuracy.

**Keywords:** Software Project Scheduling Problem(SPSP), Resource Constraint Project Scheduling Problem (RCPSP), Genetic Algorithm(GA), Ant Colony Optimization (ACO), Meta heuristic methods.

## I. INTRODUCTION

Developing effective and efficient tools for project management is a challenging and critical task. Project management is an application of knowledge, skills, tools and technique to solve project scheduling problem. Project scheduling is very critical task in project development. SPSP is a NP-hard problem [4]. This problem consist of number of resources, the important resource is human i.e. Software engineer. To allocate this engineer to a particular task with minimum salary and minimum duration and complete a project is basic requirement in project scheduling. SPSP is related to RCPSP which find an optimal schedule that meets the precedence requirement and minimize the project duration and cost [1]. The SPSP is different than RCPSP is that in SPSP employee is considered with multiple skill and in RCPSP number of resources are considered. Classical Meta heuristic approach is applied to solve SPSP problem such as Genetic Algorithm.

The GA were quite flexible and efficient for software project scheduling [1]. So many works has been done to solve the SPSP using genetic algorithm. In [4] the author describe the time line based model for SPSP with 3-d GA's. The success ratio of GA for SPSP is not satisfactory. To take the advantage of ACO Meta heuristic to propose an effective and efficient algorithm to solve the SPSP.

ACO are used to solve many NP-hard combinatorial problem effectively such as ACS: TSP (Travelling Salesman Problem) [8]. ACO for job shop Scheduling [9], Train Scheduling using ACO technique[7]. The 1<sup>st</sup> application of ACO was RCPSP [2]. The author use two technique to solve the SPSP problem. 1<sup>st</sup> is solve SPSP using ACO and 2<sup>nd</sup> is domain knowledge has been found to be helpful for improving the performance of the scheduling algorithm.

This paper is organized as follows: Software project scheduling problem is covered in section 2. Section 3 presents Ant Colony Optimization, Section 4 describe ACO for SPSP, ACS:SPSP is compared with GA in section 5. We conclude the paper in the section 6.

## II. THE SOFTWARE PROJECT SCHEDULING PROBLEM(SPSP)

### A. Problem Description:

SPSP is a problem of finding an optimal schedule for software project so that all the constraint such as precedence constraint and satisfied and this schedule minimize the salary and duration. SPSP model takes input as a task and the details of task such as task id, task effort, required skill to complete that task, maxhead count of that task. SPSP also take employee as an input and details of employee such as employee id, employee salary and maximum



dedication of employee to each task, skill list of each employee. SPSP take TPG(Task Precedence graph) as input where  $TPG = \{V, A\}$  where  $V$ =set of task and  $A$ =edge between task  $i$  and task  $j$ . Basic Symbols and notations that are used in ACS: SPSP paper is,

Table. 1. Basic symbols and notations for SPSP model.

$SK = \{skill_1, skill_2, \dots, skill_{ sk }\}$	Set of skills required in the project management.
$TK = \{task_1, task_2, \dots, task_{ tk }\}$	Set of task.
$EM = \{emp_1, emp_2, \dots, emp_{ em }\}$	Set of employee.
$V = TK$	Set of Vertex (task) for TPG in project Scheduling
$A = \{(task_1, task_2), (task_3, task_4) \dots\}$	Set of arc(precedence relation between task)
$G(V, A)$	TPG in Project Scheduling
$task_j^{effort}$	Workload of task $task_j$ which is in PM
$task_j^{skills}$	Set of Required skills of task $task_j$ .
$emp_i^{skills}$	Set of Skills Of employee $emp_i$ .
$emp_i^{salary}$	The monthly salary of employee $emp_i$ .
$emp_i^{Maxded}$	The Maximum Dedication Of employee $emp_i$ to task $task_j$ .
$task_j^{start}$	Start time of task $task_j$ .
$task_j^{end}$	End time of task $task_j$ .
$task_j^{cost}$	Cost of task $task_j$ .
$task_j^{Duration}$	Duration of task $task_j$ .
$Project^{Duration}$	Total Duration of Software Project.
$Project^{Cost}$	Total Cost of Software Project.
$Project^{OverWork}$	Total overwork of Whole Software Project.
$emp_i^{Over}$	Overwork of Employee $emp_i$ .
$M = (M_{ij})_{em \times tk}$	Solution Matrix of SPSP.

The effort and staff is calculate by using any COCOMO model[8]. For scheduling task and employee following constraint must be satisfied.

- 1) Every task must involve at least one employee.

$$\sum_{i=1}^E m_{ij} > 0 \quad \forall j \in \{1, 2, \dots, TK\} \tag{1}$$

Where,

$M_{ij}$ - is the degree of dedication of employee  $e_i$  to the task  $t_j$ .

- 2) Employee involved into task should be able to supply all skill required by this task.

$$task_j^{skills} \subseteq \cup emp_i^{skills} \quad \forall j \in \{1, 2, \dots, T\} \quad \{i | m_{ij} > 0\} \tag{2}$$

- 3) Feasible solution should not make employee to work overtime.
- 4) Each task satisfies the precedence constraint.



In SPSP to evaluate the quality of software project schedule, cost of the project and duration of the project is calculated.

For calculating duration, start and finish time of each task is calculate using this equation.

Start time and finish time of each task is calculate:

$$task_j^{start} = \begin{cases} 0 & \text{if } \forall k \neq j, (t_k, t_j) \notin A \\ \max\{task_k^{end} \mid (task_k, task_j) \in A\} & \text{else} \end{cases} \quad (3)$$

$$task_j^{end} = task_j^{start} + task_j^{duration} \quad (4)$$

Calculate the total duration of software project:

Total duration of project is calculated using Gantt chart of the project.

$$task_j^{duration} = \frac{task_j^{effort}}{\sum_{i=1}^E m_{ij}} \quad (5)$$

This equation is used to calculate the duration of task.

$$project^{duration} = \max\{task_j^{end} \mid \forall k \neq j (task_j, task_k) \notin A\} \quad (6)$$

This will calculate the total duration of a software project.

Calculate the total cost of the project:

1<sup>st</sup> calculate the cost of each task using following formula.

$$task_j^{cost} = \sum_{i=1}^E emp_i^{salary} \cdot m_{ij} \cdot task_j^{duration} \quad (7)$$

Once the cost of all task is calculate then sum of all cost of task is the cost of project.

$$project^{cost} = \sum_{j=1}^T task_j^{cost} \quad (8)$$

The overwork of employee is calculated using ramp function.

$$emp_i^{over} = \int_{t=0}^{t=p_{dur}} ramp(e_i^{work}(t) - e_i^{maxded}) dt \quad (9)$$

Where,

$$ramp(x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \end{cases} \quad (10)$$

$$emp_i^{work}(t) = \sum_{\{j \mid task_j^{start} \leq t \leq task_j^{end}\}} m_{ij}$$

### III. ANT COLONY OPTIMIZATION (ACO)

ACO was designed by Italian author Dorigo and this Meta heuristic approach is used to solve many NP-hard Problems. ACO [10] is an algorithm based on the behavior of the real ants in finding a shortest path from a source to the food. This algorithm utilizes the behavior of the real ants while searching for the food. It has been observed that the ants deposit a certain amount of pheromone in its path while traveling from its nest to the food. Again while returning, the ants subjected to follow the same path marked by the pheromone deposit and again deposit the pheromone in its path. In this way the ants following the shorter path are expected to return earlier and hence increase the amount of pheromone deposit in its path at a faster rate than the ants following a longer path. ACO takes the inspiration from the foraging behavior of the ants. These ants deposit pheromone on the ground in order to mark some favorable path that should be followed by other members of the colony. However, the pheromone is subjected to evaporation by a certain amount at a constant rate after a certain interval and therefore the paths visited by the ants frequently, are only kept as marked by the pheromone deposit, whereas the paths rarely visited by the ants are lost because of the lack of pheromone deposit on that path and as a result the new ants are intended to follow the frequently used paths only. Thus, all the ants starting their journey can learn from the information left by the previously visitor ants and are guided to follow the shorter path directed by the pheromone deposit. In ACO, a number of artificial ants build solutions to the considered optimization problem at hand and exchange information on the quality of these solutions via a communication scheme that is pheromone deposit on the path of the journey performed by it. At the present time, many



algorithms have been suggested based on the improvement of ACS algorithm and used for solving various problems such as TSP, Train scheduling, nurse scheduling, job shop scheduling etc.

#### IV. ACO FOR SPSP

The problem of SPSP is solved using ACO includes the important algorithm is ACS:SPSP algorithm and construction graph for selecting the dedication of each employee to the task and heuristic information is calculated by using one of the six strategies and designing the pheromone to solve the communication problem.

##### Construction Graph-

ACO is widely used to solve the combinatorial problem. The 1<sup>st</sup> step of applying ACO to SPSP is to construct graph that will assign the dedication of each employee to the task. Each employee gives some dedication to the task, so this can be assigned to the task by using construction graph. Construction graph is designed by splitting the task into number of nodes and this nodes depends upon the number of employee and the value if minimum dedication.

First we calculate the density of nodes i.e.

$$Density = minded^{-1} + 1 \quad (11)$$

Where, minded is minimum dedication of employee to task j.

The value of dedication is start from 0 and it is increased in multiple of minded. The value of density is 5 when the value of minded is 0.25. the split operation of a task in TPG is described as follows:

- 1) Select starting node and put into column 0
- 2) According to the number of employee , create E number of column and give names as column<sub>1</sub> to Column<sub>E</sub> , each column include Density number of nodes.
- 3) Identify end node and add to the Column<sub>E+1</sub>
- 4) Construct all possible edges from column<sub>i</sub>to column<sub>i+1</sub> .

After splitting the task ant select one by one edge and reaches to the next task. This path is straight forward. There is no any returning backward path. Ant select edge according to the dedication of employee to the task. Ant select only one edge from each column and go to next column. At the end when ant reaches to the end of node i.e. end task that time the total task is assigned to employee and one tour is complete. After that according to the dedication of employee the quality of solution is evaluated.

##### Implementation of ACS-SPSP-

We have constructed the graph that will help to assign the dedication of employee to the task. Once the tour is completed the quality of solution is checked by using fitness function ACS is employed to obtain the best solution which has the maximum fitness values. The fitness function is defined as the inverse weighted sum of project cost and project duration. We consider the importance of project cost and duration is equal in the fitness function and the weights are used to adjust the project cost and duration to the same order of magnitude. The fitness function is presented as following:

$$f(x) = (w_{cost} * project^{cost} + w_{dur} * project^{duration} )^{-1} \quad (12)$$

The details of ACS-SPSP are as follows:

- 1) Initialize all parameter that are used in ACO. Such as  $\alpha, \beta, q_0, \rho, N_{gen}, N_{ant}$  .  $\alpha$  and  $\beta$  are used to evaluate the importance of history information and heuristic information.  $\rho$  which adjust the pheromone updating,  $q_0$  balance the exploration and exploitation behaviour,  $N_{gen}$  is the number of generation of ACO,  $N_{ant}$  is the number of ant.
- 2) Initialize pheromone values. All values are 0.
- 3) Ant select path to get solution. Each ant select next node in construction graph using selection scheme. Each ant maintains its own solution matrix and store the result dedication values into that matrix.
- 4) Evaluate the quality of a matrix using fitness function. Calculate the cost and duration of project as well as over time load of the employee.
- 5) Select the best solution and update the pheromone values using pheromone update formula.



- 6) Step 3-5 repeat till the termination condition is not satisfied. Termination condition is either the number of generation or the quality of solution.
- 7) Select the best solution whose cost and duration is less according to the fitness function.

**Pheromone management-**

Ant select path in construction graph using selection scheme and when tour is finished that means all task are successfully assigned to the employee, by using fitness function we check the quality of solution and if solution is best then the pheromone values are updated. Pheromone value are updated in two times. The update of pheromone  $\tau_{ij}$  on the edge ended with  $N_{ij}$  is implemented by the following equation:

$$\tau_{ij} = (1 - \rho) \cdot \tau_{ij} + \rho * \Delta\tau \tag{13}$$

Where,  $\rho$  is the  $0 < \rho < 1$ , is a factor which indicate show much influence old pheromone makes on new pheromone and  $\Delta\tau$  is an offset related to the quality of current solution. If  $\rho$  is large, the new pheromone value is less affected by old pheromone value but much affected by  $\Delta\tau$  and vice versa. In local pheromone updating scheme, only the duration and cost of a single task are used. The offset  $\Delta\tau$  in global pheromone updating rule is given by:

$$\Delta\tau = (w_{cost} * project^{cost} + w_{dur} * project^{duration} + w_{over} * project^{overWk})^{-1} * sgn\Delta\tau$$

Where

$$sgn = \begin{cases} 0 & \text{if project can be done} \\ 1 & \text{else} \end{cases}, w_{cost}, w_{dur}, w_{over} \tag{14}$$

The offset  $\Delta\tau$  in local pheromone updating rule is given by:

$$\Delta\tau = (w_{t\_cost} * t_{cost}^k + w_{t\_dur} * t_{dur}^k)^{-1} \tag{15}$$

Where  $w_{t\_cost}$  and  $w_{t\_dur}$  are certain constants which are used to control the relative importance of task cost When ant select next node using selection scheme, such selection scheme are as follows:

$$p_k(i, j) = \frac{[\tau(i, j)]^\alpha \times [\eta(i, j)]^\beta}{\sum_{s=0}^{density} [\tau(i, s)]^\alpha \times [\eta(i, j)]^\beta} \tag{16}$$

Where,

$\tau(i, j)$  is the pheromone on the path ended with node  $N_{ij}$  and  $\eta(i, j)$  is the heuristic information of the problem.  $\alpha$  and  $\beta$  are both fixed parameters.

**Heuristic information-**

Ant select next node in construction graph when task is split, using the heuristic information and selection scheme. This paper use six different heuristic strategies to calculate heuristic information. Out of those Allocated Dedication is the best strategy and whose result is best than the other strategies. The basic formula used for calculating the heuristic information is as,

$$Heur[i] = \begin{cases} \frac{Temp[density - i - 1]}{Total} & \text{if } AllocDedi[k] > 0.5 \\ \frac{Temp[i]}{Total} & \text{else} \end{cases} \tag{17}$$

Where,

$$Temp[i] = \begin{cases} Dedi[i] + AllocDedi[k] - 0.5 & \text{if } AllocDedi[k] > 0.5 \\ Dedi[i] + AllocDedi[k] & \text{else} \end{cases} \tag{18}$$

TOTAL is the sum of all temporary values as calculated as

$$Total = \sum_{i=0}^{Density} Temp[i] \tag{19}$$

AD strategy means that the heuristic information is related to the dedication of employee  $e_i$  has already been contribute to other task.

**V. COMPARISON**

The SPSP is a NP-hard problem, so the time complexity of such problem is very high. Such problem is solved by using many techniques. Manual calculation of such problem is not a solution. This problem is solved by using Meta heuristic approach. In Meta heuristic approach SPSP is solved by using GA and ACO. By comparing the result of ACO to the GA, ACO is better than the GA. The average hit rate and project duration of ACO solution is always better than GA. But ACS-SPSP or GA cannot be obtain the feasible solution for the instances with 30 tasks.



## VI. CONCLUSION

SPSP is NP-hard problem. For scheduling task and employee is not a simple task. Its time complexity is very high. Due to its complexity there are very few algorithm is implemented to solve such problem. There are also few algorithm of Meta heuristic approach is presented such as Genetic Algorithm. GA is also very good method to solve NP-hard problem. GA is also applied to solve SPSP problem but its success rate is not satisfactory. Compared with GA approach the proposed approach such as ACS: SPSP is effective and promising in the following aspects.

- 1) Graph based searching method is used i.e. splitting the task by distributing the dedication of employee to a task and construction graph is generated.
- 2) Six heuristic mechanism are used to use the domain knowledge including total dedication in task and allocated dedication of employees.

This aspects are considered and in six heuristic mechanism allocated dedication of employees to other task heuristic achieves the best result. So the ACS:SPSP is outperform the GA approach for the SPSP.

## REFERENCES

- [1]. Alba E, Chicano JF. Software project management with GAs. Information Sciences 2007;177(11):2380–401.
- [2]. Brucker P, Drexl A, Mohring R, Neumann K, Pesch E. Resource-constrained project scheduling: notation, classification, models and methods. European Journal of Operational Research 1999;112(1):3–41.
- [3]. Chang C, Christensen & Zhang T. Genetic algorithms for project management. Annals of Software Engineering 2001;11(1):107–39.
- [4]. Garey M, Johnson D. Computers and intractability: a guide to the theory of NP-completeness. W. H. Freeman; 1979.
- [5]. Heerkens GR, editor. McGraw-Hill; 2002
- [6]. Jing Xiao, Xian-TingAo, YongTang, “Solving software project scheduling problems with ant colony optimization”.
- [7]. K. Sankar, Train Scheduling using ACO technique.
- [8]. Marco Dorigo& Luca Maria Gambardella. ACS: A cooperative learning Approach to the Travelling Salesman Problem.
- [9]. Sjoerd van der zwaan&carlos Marques, ACO for job shop Scheduling.
- [10]. Wei-Neng Chen, Jun Zhang, “Ant Colony Optimization for Software Project Scheduling and Staffing with an Event-Based Scheduler”.

## BIOGRAPHY



**Kishor Namdeo Vitekar** received the Bachelor’s degree (B.E.) in Computer Engineering in 2011 from VPCOE, BARAMATI. He is now pursuing ME degree, in Computer Engineering at VPCOE, BARAMATI. Kishor has qualified NTSE exam in 2003 as well as GATE exam in 2011 and 2012. Kishor has participated and received certificate of participation in “Writing Effective conference Papers” organized by MHRD and IIT, Bombay His current research interests include project scheduling, Image processing, cloud computing and network security.



**Suraj Anil Dhanawe** Obtained Computer Engineering degree (B.E.) in 2011 from VPCOE, BARAMATI. He is now pursuing ME degree, in Computer Engineering at VPCOE, BARAMATI. Suraj has qualified GATE exam in 2012. Research Interest are project scheduling, Data mining.



**Dinesh Bhagwan Hanchate** Birth Place :- Solapur, B.E. Computer from Walchand College of Engineering, Sangli (1995), Lecturer in Gangamai College Of Engineering, Dhule (1995-96), Lecturer in S.S.V.P.S.’s B.S.D. College Of Engineering, Dhule In Computer & IT dept. ( 1996-2005), M.Tech. Computer from Dr. Babasaheb Ambedkar Technological University, Lonere(2002-05), Currently Asst. Prof. Computer Engineering, former H.O.D. ( Computer & IT ) in Vidya Pratishthan’s College Of Engineering, Baramati , currently doing research (SGGS’s Institute of Technology and Engg, Nanded affiliated to SRTMU, Nanded) under the guidance of Dr. Bichkar R.S. .G.H. Raisoni’s College Of Engineering and Management, Wagholi, Pune.