

Resource-constrained project scheduling with ant colony optimization algorithm

Niken A Savitri^a, I Nyoman Pujawan^a, Budi Santosa^a

Abstract: Resource allocation commonly becomes one of the critical problems in project scheduling. This issue usually occurs because project managers estimate the schedule of activities and network time without considering resource availability. Resource-Constrained Project Scheduling Problem (RCPSP) links to the allocation of resources or set of resources into certain activities in order to accomplish particular objectives. Various approaches have been performed to overcome RCPSP, including the heuristic approach. In this research, Ant Colony Algorithm is used to solve RCPSP. There are 11 examples of projects being investigated with dissimilarity in-network and several activities. The implementation of the Ant Colony Algorithm resulted in the percentage of a near-optimal solution of 63.64%. Besides, the duration obtained from the algorithm above the manual scheduling (assumed optimal) was only 4.29%. Sensitivity analysis was performed to understand how substantially the changes of ACO parameters influenced the result obtained from the algorithm. Based on the result, it could be concluded that the parameters of ACO have no significant effect to project duration.

Keywords: resource-constraint project scheduling, ant colony algorithm

INTRODUCTION

Project scheduling is a critical task in project management. The availability of limited resources and precedence relations between activities make project scheduling a complicated task [1]. A successful project execution depends on optimal and accurate project management. Such a result will impact the increased revenues and decreased costs. One of the critical aspects of project management is scheduling, which has turned into a challenging task due to resource constraints and precedence relationships. As a result, the project scheduling problem is considered by researchers as one of the most generally used and fundamental issues. The development of this issue has been addressed in various aspects [2].

The project scheduling problem is recognized as concluding the time required to implement a project's activities to achieve a definite objective [3]. Proper and realistic scheduling is an essential factor of success for every project. In actuality, project scheduling often involves several objectives that must be achieved instantaneously and have significant uncertainties that may challenge the planned schedule's integrity. Therefore, it is crucial to have effective planning for dealing with such uncertainties.

Resource-Constrained Project Scheduling Problem (RCPSP) is recognized as an NP-hard problem. The solution approach is computationally expensive [4]. Due to the high degree of complexity of RCPSPs, numerous heuristics and metaheuristic methods have been proposed in the literature [5]. This research focuses on approaching RCPSP using an ant colony algorithm. The ACO algorithm is expected to find the optimal solution for the problems.

LITERATURE REVIEW

Resource-constrained project scheduling problems (RCPSPs) comprise assigning jobs to a resource or set of resources with limited capacity to meet some well-defined objectives. The various objectives depending on the

decisionmaker's goals. However, the most common one is to achieve the lowest time to complete the entire project [6].

[6] broadly categorized RCPSP into the following six classifications: single-mode, multi-mode, RCPSP problems with Non-regular objective functions, stochastic, Bin-packing-related RCPSP, and Multi-resource-constrained project scheduling problems [6]. Based on the resources available for completing tasks in the project, RCPSP is classified into a problem with renewable and non-renewable resources [3]. Renewable resources have the same quantity of availability in every period for an unlimited number of periods [7]. Most studies are assuming renewable resources such as [8]. For non-renewable resources, constraints on availability only concern total consumption over the whole project duration and not at each period. The raw material is one of the most noteworthy examples of non-renewable resources because it is available for a project at a specific and settled amount [5].

One of the assumptions linked to the basic RCPSP is that the project's activities cannot be interrupted or discontinued. Namely, an activity must continue to the end without disruptions [3]. However, some activities must be stopped in practice during the processing, which can be due to damages or disabilities of resources, equipment repair, and others [9]. The most frequent objective function in the RCPSP is the minimization of the project completion time. In this objective function, the project activities scheduled to get the project completion time attain the minimum value. [10] offered a model that minimized the expected project completion time by considering the uncertainty of activities duration in the scheduling problem.

RCPSP has been approximated with various methodologies. [5] use mixed-integer linear programming models to approach the deterministic single and multi-mode RCPSP, with renewable and non-renewable resources. [11] approach RCPSP by using a chance-constrained programming model and a simple two-stage algorithm. In the research, they are taking construction project scheduling problem as the subject with uncertain resource environments. [12] use simulated annealing (SA) adaptations for RCPSP and its multiple mode version. Using the public benchmark instances, the procedures

^aLecturer in the Department of Industrial and Systems Engineering, Institut Teknologi Sepuluh Nopember, ITS Campus, Sukolilo, Surabaya 60111, Indonesia. Corresponding author email address: savitri@its.ac.id

show to be highly efficient and was able to improve the best solutions for the unsolved problems. [13] developed two real-time event-based and discrete time-based disruption recovery models for multiple resource-constrained project scheduling problems. The experimental study discloses that the re-optimization process can reduce the revised makespan. [14] apply an ant colony algorithm and compare the approach with the results of other heuristics for the RCPSp, including genetic algorithms and simulated annealing on a large set of benchmark problems. [9] modified the ACO approach for a multi-constraint (precedence and resource constraints) project scheduling problem. A two-dimension (time and activity) matrix graph is adopted to solve the scheduling problem.

METHODOLOGY

The research methodology in this paper consisted of two sections. The first section presents the mathematical model used to approach the resource-constraint project scheduling problem. The second section explains the ant colony optimization algorithm parameters and the proposed flow diagram.

A. MATHEMATICAL MODEL

The RCPSp can be described as per the following. A single project contains a set $J = \{0, 1, \dots, n\}$ of activities that have to be processed. Invented activities 0 and n correspond to the "project start" and the "project end," respectively. Two kinds of constraints interconnect the activities. First, precedence constraints impose activity j not to be started before all its immediate predecessor activities have been finished. Second, executing the activities needs resources with limited capacities. Activity j requires r_{jk} units of resource $k \in K$ during every period of its non-preemptable duration p_j . Resource type k has a limited capacity of R_k . The parameters p_j , r_{jk} , and R_k are presumed to be deterministic. For the project start and end activities, it is defined that $p_j = 0$ and $r_{j,k} = 0$ for all $k \in K$. The purpose is to find precedence and resource reasonable accomplishment times for all activities with minimum makespan of the project.

In this research, the conceptual decision model for the basic resource-constrained project scheduling problem (RCPSp) defined by [15] and [16] is adopted. The objective function and constraints are given by the following:

$$\text{Minimize } f_n \quad (1)$$

Subject to

$$\text{Startup } f_0 = 0$$

Precedence Relationship

$$f_j \geq f_i + p_i \quad i, j = 1, \dots, n \quad (2)$$

Resource capacity

$$\sum_{i \in S_t} r_{jk} \leq R_k \quad t = 1, \dots, T; k = 1, \dots, K \quad (3)$$

The objective function is to minimize the total duration required to complete the project. Constraint (2) states that activity i should be finished or completed before activity j

can be started. Constraint (3) defines that the resources are used in activity i must respect the availability terms and condition. r_{ik} is defined as the requirement of resource k during activity i and S_t is a group of activity i on a period of t .

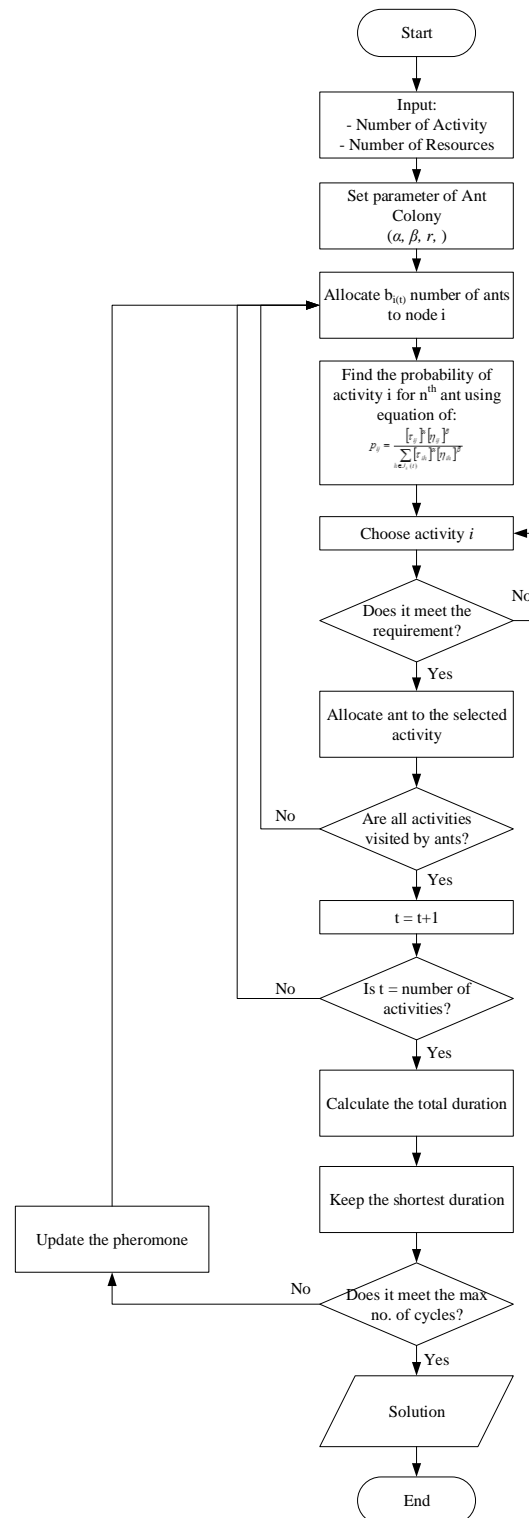


Figure 1 The flow diagram of the proposed ACO algorithm

B. ANT COLONY OPTIMIZATION ALGORITHM

In this paper, the RCPSP is being approached by Ant Colony Optimization (ACO) algorithm. The ACO methodology simulates the behavior of ants in their search for food. Ants create a pheromone trail while traveling around, used by the other ants to define the shortest path. The shortest road trail will be stronger because the ants that return to the colony using this road will refresh the pheromone trail faster than the ants that choose a longer path. The higher pheromone level attracts more ants to choose the short path, and, eventually, the shortest path is established. The flow diagram for the ACO algorithm is proposed in Figure 1.

NUMERICAL EXPERIMENT

In this section, the setting of ACO algorithm parameters and the experiment's computational result are presented.

A. MATHEMATICAL MODEL

The ACO algorithm parameters are the following: α is the relative weight of pheromone trail; β is the relative weight of visibility; the number of iteration; ρ is the evaporation rate, and N is the total number of ants at each iteration.

The role of the parameters α and β is the following. If $\alpha=0$, the closest transfer stations have a higher chance of being selected. This corresponds to a classical stochastic greedy algorithm (with multiple starting points since ants are initially randomly distributed over the stations). If, on the contrary, $\beta=0$, then only pheromone amplification is at work. It will lead to the rapid emergence of a stagnation situation. The latter can be described as a situation in which all ants make the same tour that, in general, is strongly sub-optimal. Accordingly, a proper trade-off has to be set between heuristic value and trail intensity. In this research, the parameters used are evaporation rate $\rho = 0.01$, the number of iteration 10, pheromone index $\alpha = 1$, and visibility index $\beta = 1$

B. COMPUTATIONAL RESULT

The proposed ACO algorithm was tested with Visual Basic 6.0 with ten iterations and an evaporation rate of 0.01. Eleven (11) self-built random projects were created with various activities and resources. The results were

compared with the manual solution to see how far the algorithm value is versus the manual approach.

Table 1. The computational result with Ant Colony

Project	No of activity	No of Resource	Duration with Ant Colony (day)	Duration with manual (day)	% variance
1	4	3	5	5	0.00
2	5	3	11	11	0.00
3	6	4	13	12	8.33
4	7	3	14	14	0.00
5	8	3	9	9	0.00
6	9	3	7	6	16.67
7	10	3	10	9	11.11
8	10	3	11	11	0.00
9	10	3	11	11	0.00
10	10	3	10	9	11.11
11	10	3	9	9	0.00
Average of % variance					4.29

The calculation result shows that the percentage of the near-optimal solution produced by the ACO algorithm is 63.64%. On the other hand, the variance between the ACO algorithm and the manual approach is 4.29%.

B. SENSITIVITY ANALYSIS

Sensitivity analysis was performed to measure how significant impact of ACO algorithm parameters on the total duration. In this analysis, the parameters α (weight of pheromone trail), β (weight of visibility), and ρ (evaporation rate) were adjusted accordingly. The analysis is performed on projects that have divergence on the number of activities and resources. Therefore, it was applied only on project one to project seven because project eight to project eleven have the same number of activities and resources. The value of $\alpha = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, \text{ and } 0.9]$ are being used. The impact of adjustment on parameter α is presented in Table 2.

Table 2. The result of sensitivity analysis for parameter α

Parameter				Project 1 (day)	Project 2 (day)	Project 3 (day)	Project 4 (day)	Project 5 (day)	Project 6 (day)	Project 7 (day)
α	B	ρ	τ_0							
0,1	1	0,5	0,01	5	11	13	14	9	7	10
0,2				5	11	13	14	9	7	10
0,3				5	11	13	14	9	7	10
0,4				5	11	13	14	9	7	10
0,5				5	11	13	14	9	7	10
0,6				5	11	13	14	9	7	10
0,7				5	11	13	14	9	7	10
0,8				5	11	13	14	9	7	10
0,9				5	11	13	14	9	7	10
1				5	11	13	14	9	7	10

The same value [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9] are used for parameter β with the impact of adjustment presented on Table 3.

Table 3. The result of sensitivity analysis for parameter β

Parameter				Project 1 (day)	Project 2 (day)	Project 3 (day)	Project 4 (day)	Project 5 (day)	Project 6 (day)	Project 7 (day)
α	β	P	τ_0							
1	0.1	0.5	0.01	5	11	13	14	9	7	10
	0.2			5	11	13	14	9	7	10
	0.3			5	11	13	14	10	7	10
	0.4			5	11	13	14	9	7	10
	0.5			5	11	13	14	9	7	10
	0.6			5	11	13	14	9	7	10
	0.7			5	11	13	14	9	8	10
	0.8			5	11	13	14	9	7	10
	0.9			5	11	13	15	10	7	10
	1			5	11	13	14	9	7	10

While for parameter ρ , the applied values are [0.01, 0.02, 0.05, 0.1, 0.25, 0.5, 0.6, 0.75, 0.9, 1]. The impact of adjustment on parameter ρ is presented on Table 4.

Table 4. The result of sensitivity analysis for parameter ρ

Parameter				Project 1 (day)	Project 2 (day)	Project 3 (day)	Project 4 (day)	Project 5 (day)	Project 6 (day)	Project 7 (day)
α	B	ρ	τ_0							
1	1	0.01	0.01	5	11	13	14	9	7	10
		0.02		5	11	13	14	9	7	10
		0.05		5	11	13	14	9	7	10
		0.1		5	11	13	14	9	7	10
		0.25		5	11	13	14	9	7	10
		0.5		5	11	13	14	9	7	10
		0.6		5	11	13	14	9	7	10
		0.75		5	11	13	15	9	8	10
		0.9		5	11	13	14	9	7	10
		1		5	11	13	14	10	8	10

RESULT AND DISCUSSIONS

In this paper, the data of eleven projects with a range of four to ten activities are being used. Using the parameters that have been set, the algorithm produces a near-optimal solution with a total percentage of 63.64%. Sensitivity analysis was performed by changing the value of parameter α (weight of pheromone trail), β (weight of visibility), and ρ (evaporation rate). The analysis aims to evaluate if the changes in ACO parameters affect the optimal solution of the algorithm. Based on the sensitivity result, it is concluded that projects with few activities are not affected by the changes in ACO parameters.

CONCLUSION

This paper presented a heuristic approach to the resource-constrained project scheduling problem (RCPSP) with an ant colony optimization algorithm (ACO). Based on the calculation result, the algorithm produces a near-optimal solution with a total percentage of 63.64%. Sensitivity analysis was performed to evaluate if the changes in ACO parameters affect the solution. The sensitivity result shows that changes in ACO parameters do not affect the optimal solution, particularly for projects with few activities. It is suggested to apply the ACO algorithm for RCPSP in projects with a bigger scale and more activities, for future research. Besides, the trial simulation could be conducted to identify ACO parameter settings that result in a good performance.

REFERENCES

- [1] S. Hartmann and D. Briskorn, "A survey of variants and extensions of the resource-constrained project scheduling problem," *Eur. J. Oper. Res.*, vol. 207, no. 1, pp. 1–14, 2010, doi: 10.1016/j.ejor.2009.11.005.
- [2] F. Habibi, F. Barzinpour, and S. J. Sadjadi, "A multi-objective optimization model for project scheduling with time-varying resource requirements and capacities," *J. Ind. Syst. Eng.*, vol. 10, no. special issue on scheduling, pp. 92–118, 2017, [Online]. Available: http://www.jise.ir/article_53352.html.
- [3] F. Habibi, F. Barzinpour, and S. J. Sadjadi, "Resource-constrained project scheduling problem: review of past and recent developments," *J. Proj. Manag.*, vol. 3, pp. 55–88, 2018, doi: 10.5267/j.jpm.2018.1.005.
- [4] D. Arkhipov, O. Battaïa, and A. Lazarev, "An efficient pseudo-polynomial algorithm for finding a lower bound on the makespan for the Resource Constrained Project Scheduling Problem," *Eur. J. Oper. Res.*, vol. 275, no. 1, pp. 35–44, 2019, doi: 10.1016/j.ejor.2018.11.005.
- [5] T. S. Kyriakidis, G. M. Kopanos, and M. C. Georgiadis, "MILP formulations for single- and multi-mode resource-constrained project scheduling problems," *Comput. Chem. Eng.*, vol. 36, no. 1, pp. 369–385, 2012, doi:

- 10.1016/j.compchemeng.2011.06.007.
- [6] B. Yang, J. Geunes, and W. O'brien, "Resource constrained project scheduling: Past work and new directions," *Dep. Ind. ...*, pp. 1–28, 2001, [Online]. Available: <http://ie406.cankaya.edu.tr/uploads/files/RCPS.pdf>.
- [7] P. Wuliang, H. Min, and H. Yongping, "An improved ant algorithm for Multi-mode Resource Constrained Project Scheduling Problem," *RAIRO - Oper. Res.*, vol. 48, no. 4, pp. 595–614, 2014, doi: 10.1051/ro/2014025.
- [8] M. B. Abello and Z. Michalewicz, "Multiobjective resource-constrained project scheduling with a time-varying number of tasks," *Sci. World J.*, vol. 2014, 2014, doi: 10.1155/2014/420101.
- [9] R. Chen and S. Lo, "Using an Enhanced Ant Colony System to Solve Resource- Constrained Project Scheduling Problem," *J. Comput. Sci.*, vol. 6, no. 11, pp. 75–84, 2006.
- [10] S. Creemers, "Minimizing the expected makespan of a project with stochastic activity durations under resource constraints," *J. Sched.*, vol. 18, no. 3, pp. 263–273, 2015, doi: 10.1007/s10951-015-0421-5.
- [11] Z. Wan and J. He, "Construction project scheduling problem with uncertain resource constraints," *Chinese J. Eng.*, no. 70371032, pp. 1–12, 2004, [Online]. Available: http://en.cnki.com.cn/Article_en/CJFDTOTAL-GCSX200503003.htm.
- [12] K. Bouleimen and H. Lecocq, "A new efficient simulated annealing algorithm for the resource-constrained project scheduling problem and its multiple mode version," *Eur. J. Oper. Res.*, vol. 149, no. 2, pp. 268–281, 2003, doi: 10.1016/S0377-2217(02)00761-0.
- [13] R. K. Chakraborty, A. Abbasi, and M. J. Ryan, "Reactive scheduling of projects with unreliable resources," *Proc. Int. Conf. Comput. Ind. Eng. CIE*, vol. 2018-December, no. January, 2018.
- [14] D. Merkle, M. Middendorf, and H. Schmeck, "Ant colony optimization for resource-constrained project scheduling," *IEEE Trans. Evol. Comput.*, vol. 6, no. 4, pp. 333–346, 2002, doi: 10.1109/TEVC.2002.802450.
- [15] E. De Frene, D. Schatteman, W. Herroelen, and S. Van de Vonder, "A Heuristic Methodology for Solving Spatial a Resource-Constrained Project Scheduling Problems," *SSRN Electron. J.*, 2011, doi: 10.2139/ssrn.1089355.
- [16] R. Kolisch and S. Hartmann, "Heuristic Algorithms for the Resource-Constrained Project Scheduling Problem: Classification and Computational Analysis," pp. 147–178, 1999, doi: 10.1007/978-1-4615-5533-9_7.