The Ninth International Conference on Construction in the 21st Century (CITC-9)
"Revolutionizing the Architecture, Engineering and Construction Industry through Leadership, Collaboration and Technology"
March 5th-7th, 2017, Dubai, United Arab Emirates

Resource-Constrained Project Scheduling Using Evolutionary Algorithms

Charalampos P. Sinos
Hellenic Open University, Patras, Greece
sinosxaris@yahoo.gr

Athanasios P. Chassiakos
Department of Civil Engineering, University of Patras, Patras, Greece
a.chassiakos@upatras.gr

Abstract

The multi-mode resource-constrained project scheduling problem (RCPSP) is perhaps the most difficult combinatorial optimization problem in project management. This is because the RCPSP includes multiple and conflicting objectives and constraints e.g., leveling the resource profile while keeping resource usage within availability constraints and the project length within acceptable limits. The solution space increases significantly in size as the number of activities and execution modes increase while strict resource availability limitations may extend the project length considerably leading to a wide range of possible activity start times. In such a problem environment, the employment of evolutionary algorithms is necessary to approximate optimal solutions. In this work, a comparative performance analysis of five evolutionary algorithms for solving the RCPSP is presented. The analysis aims to provide insight regarding the algorithm performance in terms of the solution quality and efficacy in model design, implementation, and parameter tuning. The algorithm implementation is done in Matlab environment to facilitate the analysis process. Evaluation results indicate that Genetic Algorithms and Particle Swarm Optimization can provide optimal or near-optimal solutions in most cases, Harmony Search and Simulated Annealing algorithms respond well in medium size problems while Ant Colony Optimization presents a lower performance compared to other algorithms.

Keywords

Resource allocation, Resouce Leveling, Evolutionary algorithms, Project scheduling, Multiobjective optimization

1. Introduction

Construction project scheduling under resource and/or completion deadlines constraints is difficult to solve analytically as the number of activities increases. The problem complexity grows significantly if multiple execution modes (in terms of daily resource units allocated in an activity) are considered. However, the latter provision (if feasible) can improve the resource allocation process and provide solutions which better satisfy (or violate in a lower degree) existing resource and project deadline constraints. Therefore, the resource-constrained project scheduling problem (RCPSP) with single or multiple execution modes remains a problem of interest with a main research objective to develop effective search processes to find an optimal or sub-optimal solution within the vast solution space.

Existing research includes a variety of methods which can be categorized as exact methods, heuristic or meta-heuristic algorithms. Exact methods, such as linear, integer, dynamic or constrained programming (Menesi and Hegazy, 2014), can produce optimal results but scarcely for problems with a wide exploration field. Besides, exact methods are more efficient for continuous problems while project scheduling is a rather discrete problem with scattered nature of results, meaning that two slightly different solutions (schedules) may have considerably different resource allocation profile.

Heuristic approaches, such as branch and bound algorithms, initially produce a list of solutions and then choose someone close to the near -optimal solution while discarding out of bounds solutions. Moukrim, et al. (2015) used a branch and price algorithm with activity preemptiveness to solve the RSPSP for project makespan minimization. The problem with such methods is that there is no universal bounding algorithm working for all problems. According to Kolisch and Drexl (1997), heuristic solution approaches fail to generate feasible solutions when problems become highly resource-constrained.

Metaheuristics are especially suitable when approximate solutions are acceptable, in case that the global optimum is either unknown or computationally expensive to obtain (Warren Liao et al., 2011). Metaheuristic techniques, such as evolutionary algorithms, are stochastic processes that mimic the physical - biological behavior and the evolution of species. The common idea of evolutionary algorithms is that they iteratively search for a near optimum solution producing improved solutions with the process evolution. Evolutionary algorithms have been widely used for the exploration of near optimum solutions in multi -objective RCPSP (Warren Liao et al., 2011, Kaiafa and Chassiakos, 2015). Researchers use multiple types of resources (Zhang, 2012, Menesi and Hegazy, 2014), renewable or non-renewable (Peteghem and Vanhoucke, 2008, Pérez, et al., 2014) in order to create more realistic conditions, and in some studies the resource constraint level is variable according to the resource needs in the construction phases (Kang, et al., 2015). Other researchers use hybrid schemes, taking advantage of two or more algorithms, or using an evolutionary algorithm in combination to another technique. In this direction, Orouji et al. (2014) used a hybrid PSO and shuffled frog leaping algorithm to solve the RCPSP. Compared to other methods, evolutionary algorithms are more flexible to be programmed with execution of multiple types of activities in order to produce solutions with more flattened resource profile and satisfying the resource mobilization plan as well.

In this work, a comparative performance analysis of five evolutionary algorithms in solving the RCPSP is presented. The experimentation aims to provide insight not only regarding the algorithm performance in terms of the solution quality but also in terms of robustness and user friendliness in model design, implementation, and parameter tuning.

2. Proposed model

The objective function of the proposed model represents the total cost integrating all optimization objectives. The (generally conflicting) objectives are to finish the project as soon as possible, within the resource availability and project deadline (if any), and to let the resource histogram be as leveled as possible in time. Unit cost values are used that represent the “weight” of each objective and the “degree of hardness” of each constraint in a specific problem. In particular, the total cost consists of the following items:

- The direct project cost representing the expenditures for all resources required for project execution.
- The indirect cost (general project expenses) which is proportional to the project duration.
- The cost of exceedance a given project deadline which valuates the consequences of delaying the project beyond the deadline.

- The cost of exceeding the daily resource availability threshold which represents the cost required for recruiting additional resources to the construction site.
- The cost associated with resource usage fluctuations in time which accounts for the cost for moving in and out resources or the cost for keeping resources idle in the construction site.

The analysis includes two types of activity resource adjustments. In the first, a single execution mode is considered indicting that each activity will be executed in a specific way, i.e., at a certain resource rate and duration. The optimization is then done only by rescheduling activities according to their precedence relations. In the second case, multiple execution alternatives are considered following a trade-off between resource rate and activity duration (with the total effort to remain unchanged).

Five types of evolutionary algorithms were implemented and tested with regard to the above problem, namely Genetic Algorithm (GA), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), Simulated Annealing (SA), and Harmony Search (HS). All algorithms were programmed in Matlab to facilitate model development and tuning, parameter setting, and sensitivity analysis of the produced solutions. All algorithms have in common an iterative procedure with an amount of carriers (chromosomes, ants, particles, harmonies etc) starting from random initial solutions to a set of final solutions with lower cost for the objective function. A solution represents a project schedule which is set either by the activity start times or by the slack of each activity from its predecessors. When alternative execution modes are considered, a duration for each activity is chosen among feasible alternatives (activity resources then vary reciprocally to activity duration) and used as an additional input for the schedule and resource histogram development.

3. An application case study

A project with eight activities is considered for an indicative case study with project data presented in Table 1. All activities are supposed to utilize the same type of resource. Figure 1 presents the Gantt chart and the resource graph for the early start project schedule. The project duration is 15 days and the maximum resource requirement is 10 units.

Table 1: Project data for the application example

Activity	Predecessor	Duration	Resource #	Activity	Predecessor	Duration	Resource #
A	-	5	2	E	C	5	2
B	-	10	2	F	A	4	2
C	-	4	2	G	C	6	2
D	A	7	2	H	B, D, E	3	2

A number of scenarios representing different priorities and constraints are evaluated as indicated in Table 2. Case 1 represents the initial project schedule with the earliest start of activities. Case 2 provides the result of the optimization from resource allocation without exceeding the normal project duration (i.e., within available slacks). This is obtained by assigning a high cost value for each day the project is delayed beyond 15 days. Cases 3 and 4 introduce a resource constraint of 4 units per day and this is modeled by a high cost rate for exceeding this limit compared to other costs. In Case 3 each activity is to be executed by the specified number of resources while in Case 4 alternative execution options are considered for each activity based on a trade -off between resource usage and activity durations (so that the total man-hours remain unchanged). For both Cases 3 and 4, a fully leveled resource allocation diagram can be obtained at a project duration of 22 days. Figure 2 indicatively presents the Gantt chart and the resource graph for Case 4.

Activity	Time 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
A	2	2	2	2	2										
B	2	2	2	2	2	2	2	2	2	2					
C	2	2	2	2											
D						2	2	2	2	2	2	2			
E					2	2	2	2	2						
F						2	2	2	2						
G					2	2	2	2	2	2					
H													2	2	2
Total	6	6	6	6	8	10	10	10	10	6	2	2	2	2	2

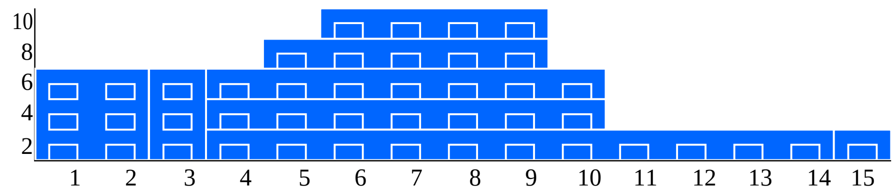

Figure 1. Gantt chart and resource graph for the early start schedule

Table 2. Optimal results for example project under different scenarios

Case no	Case description	Resource constraint	Project duration	Max resource usage	Resource standard deviation	Cumulative resource fluctuations
□	Initial early time schedule	-	15	10	3.25	12
□	Resource allocation within available slacks	-	15	6	0.52	2
□	Resource allocation under resource constraint – no activity duration adjustment permitted	R ≤ 4	22	4	0	0
4	Resource allocation under resource constraint – activity duration adjustment permitted	R ≤ 4	22	4	0	0

Activity	Time 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
A			2	2	2	2	2															
B			2	2	2	2	2	2	2	2	2	2										
C	4	4																				
D													2	2	2	2	2	2	2			
E								2	2	2	2	2										
F													2	2	2	2						
G																	2	2	2	2	2	2
H																				2	2	2
Total	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4

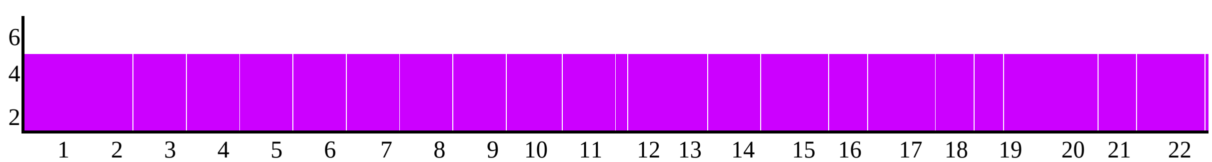

Figure 2. Gantt chart and resource graph for Case 4 scenario

The convergence progress for an indicative case is presented in Figure 3. The genetic algorithm applied to Case 3 scenario reached the optimal solution after 140 generations with all chromosomes used. Table 3 summarizes the performance results of the five algorithms with regard to the case study (discussion is provided in the following section).

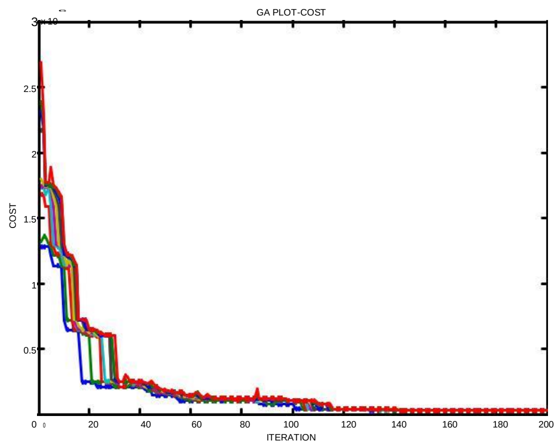

Figure 3. Optimization function convergence with genetic algorithm in Case 3 scenario

Table 3. Case study evaluation results

Method		Case 2	Case 3	Case 4
Genetic Algorithm	Project cost-duration	10900-15	32000-22	32000-22
	Simulation time	1 sec	8 sec	240 sec
Ant Colony	Project cost-duration	12100-15	39100-23	62600-29
	Simulation time	1 sec	4 sec	10 sec
Particle Swarm	Project cost-duration	10900-15	32000-22	32000-22
	Simulation time	1 sec	2 sec	145 sec
Simulated annealing	Project cost-duration	10900-15	32000-22	45300-24
	Simulation time	1 sec	3 sec	29 sec
Harmony search	Project cost-duration	10900-15	32000-22	48200 -25
	Simulation time	1 sec	36 sec	50 sec

4. Discussion and concluding remarks

On the basis of results presented in Table 3 and from other case studies, the following provisional observations can be made:

- The comparison among algorithms in terms of solution effectiveness indicates that GA and PSO algorithms present higher performance than the rest of the algorithms. On the other side, ACO algorithm seems to have the poorest efficiency among tested algorithms.
- Parameter tuning is necessary for improving the algorithm performance. The GA algorithm requires the least effort for parameter tuning while PSO and ACO algorithms are mostly dependent on proper parameter settings.

2 The consideration of alternative activity execution types (Case 4), although improving the potential for obtaining better solutions with regard to optimization objectives and problem constraints, increases the number of possible solutions and thus the difficulty to obtain the optimal solution.

3 The best solution is not guaranteed in every run. Repetitive runs of each algorithm indicate that as the problem becomes more complex, the success rate (i.e., the percent of optimal or sub-optimal solutions obtained) decreases. The genetic algorithm presents the best performance with regard to the success rate criterion.

4 With all effectiveness measures considered, the genetic algorithm appears to be the most robust method for solving the resource-constrained scheduling problem. Further testing, however, is still required.

The resource leveling problem is one of the most challenging in the area of project management. The optimization problem incorporates conflicting objectives and constraints, e.g., leveling the resource usage while keeping the project duration within acceptable limits and resource usage within availability constraints. The solution space grows significantly in size as the number of activities and execution modes increase while strict resource availability limitations lead to project length extension and to several alternative activity scheduling solution schemes. In such cases, the employment of evolutionary algorithms is necessary to approximate an optimal solution. Although results from such experimentation are promising, more work is needed to solve this problem effectively and especially as the project size increases.

References

Kaiafa, S. and Chassiakos, A.P. (2015). “A genetic algorithm for optimal resource-driven project scheduling”. *Procedia Engineering* 123, 260 – 267

Kang, L., Moon, H., Min, C., Kim, S., and Kim, H. (2015). “Developing an active resource allocation algorithm considering resource supply and demand in a construction site”. *KSCE Journal of Civil Engineering*, Vol. 19, pp 17-27.

Kolisch, R. and Drexl, A. (1997). “Local search for nonpreemptive multimode resource-constrained project scheduling”. *IIE Transactions*, Vol. 29, pp 987–999.

Menesi, W and Hegazy, T. (2015). “Multimode resource-constrained scheduling and leveling for practical-size projects”. *Journal of Management in Engineering*, 31(6): 04014092

Moukrim, A,, Quilliot, A., and Toussaint, H. (2015). “An effective branch-and-price algorithm for the Preemptive Resource Constrained Project Scheduling Problem based on minimal Interval Order Enumeration”. *European Journal of Operational Research*, Vol. 244, No 2, pp 360–368.

Orouji, H., Bozorg Haddad O., Fallah-Mehdipour E, Marino M.A. (2014). “Extraction of decision alternatives in project management: Application of hybrid PSO-SFLA”. *Journal of Management in Engineering*, Vol. 30, No 1, pp 50-59.

Pérez, Á., Quintanilla, S., Lino, P. and Valls, V. (2014). “A multi-objective approach for a project scheduling problem with due dates and temporal constraints infeasibilities”. *International Journal of Production Research*, Vol. 52, No 13, pp 3950-3965.

Peteghem, V. and Vanhoucke, M. (2008). “A genetic algorithm for the multi-mode resource-constrained project scheduling problem”. Working paper, Gent University, Belgium.

Warren Liao, T., Egbelu, P.J., Sarker, B.R., and Leu, S.S. (2011). “Metaheuristics for project and construction management – A state-of-the-art review”. *Automation in Construction*, Vol. 20, pp. 491-505.

Zhang, H. (2012) . “Ant colony optimization for multimode resource-constrained project scheduling”. *Journal of Management in Engineering*, Vol. 28, No 2, pp 150-159.