Duration-Profit Trade-Off Analysis of Construction Projects

Ahmed B. Senouci
(Associate Professor, Qatar University, Doha, Qatar)

Hassan R.Al-Derham
(Assistant Professor, Qatar University, Doha, Qata)

Abstract
This paper presents a multi-objective optimization model with the capability to generate and evaluate optimal/near-optimal construction scheduling plans that simultaneously minimize the duration of construction projects and maximize their profits. The computations in the present model are organized in three major modules: (1) a scheduling module that develops practical schedules for construction projects; (2) a profit module that computes the total project profit; and (3) a multi-objective module that searches for and identifies optimal/near optimal tradeoffs between project duration and profit. An application example is analyzed to illustrate the use of the model and to demonstrate its capabilities in generating and visualizing optimal trade-offs between construction duration and profit.

Keywords
Genetic algorithms, multi-objective optimization, construction profit, construction duration; planning and scheduling.

1. Introduction

The highly competitive nature of the construction industry places an increasing pressure on decision makers to search for optimal construction plans that simultaneously minimize the duration of construction projects and maximizes their profits.

Significant research advancements have been made in the area of total cost and duration optimization of construction projects. A number of models have been developed using a variety of methods, including heuristics methods, mathematical programming, and genetic algorithms. Heuristic methods, which are based on rules of thumb, do not guarantee optimal solutions (Moselhi 1993). Mathematical programming methods convert the duration-cost trade-off program to mathematical models and utilize linear programming, integer programming, or dynamic programming to solve them (Elmagraby 1993). Mathematical models are usually not suitable for solving large-scale construction project scheduling problems (i.e., several hundred activities). Feng et al. (1997) and Zheng et al. (2004) developed genetic algorithm models for construction duration-cost trade-off analysis. Elazouni and Metwally (2005) developed a genetic programming method that produces finance-based schedules that maximize poject profit through minimizing financing costs and indirect costs. While the above research studies have provided significant contributions to the area of construction scheduling, there has been little or no reported research focusing on multi-objective models that optimize both construction duration and profit.

The objective of this paper is to present the development of a multi-objective genetic algorithm model that supports both duration minimization and profit maximization of construction projects. The model, which is formulated as a two-dimensional duration-profit trade-off analysis, provides contractors/decision makers in construction projects with the capability of generating optimal/near optimal resource utilization plans that optimize both construction duration and profit.

2. Terms and Definitions

The present model requires the following input data:

1. The number of project activities ***NAct***.
2. The number of crew formation strategies for each activity. ***NCrew(n)*** denotes the number of available crew formations for activity ***n***. A crew formation refers to: (1) the construction contract type (i.e., indicates whether the construction job is performed using the contractor's own forces or sub-contracted) and (2) the different crew sizes and acceleration strategies (e.g., overtime options or multiple shifts).
3. The duration of each activity for each crew formation. Each crew formation has its unique output rate and associated cost. For a given crew formation F_n selected for an activity ***n***, the duration of activity ***n*** is denoted as $ActDur(n, F_n)$, where ***n = 1, 2 ,..., NAct*** , and $F_n = 1, 2, \ldots, NCrew(n)$.
4. The direct cost of each activity for each crew formation. The direct cost of activity ***n*** using crew formation F_n is denoted as $ActCost(n,F_n)$.
5. Lag/lead time between an activity and its predecessor may be necessary because of physical or contractual constraints. This lag/lead can be associated with the typical precedence relationships (finish-start, start-start, finish-finish, and start-finish). The lag/lead times between activity n and its predecessor m are described by a three-dimensional array (Lag). The lag/lead between activity ***n*** and its predecessor m is denoted as ***Lag(n,m)***. ***Lag(n,m)*** is positive when representing a lag time and negative when representing a lead time.

3. Optimization Modules

The optimization computations in the present model are organized in three major modules: (1) a scheduling module that develops practical schedules for construction projects; (2) a profit module that computes the profit of construction projects; and (3) a multi-objective module that searches for and identifies optimal/near optimal tradeoffs between project duration and profit. The following sub-sections present a detailed description of these three major modules.

3.1 Schedule Module

The main objective of this module is to develop practical schedules for construction projects. The early start and finish times of the project activities are computed in this module. The early start $STime(n,F_n)$ is defined as the earliest time an activity ***n***, using a crew formation F_n can start. Similarly, the early finish $FTime(n,F_n)$ is defined as the earliest time an activity ***n***, using a crew formation F_n can be completed.

The start time $STime(n,F_n)$ of an activity n using a crew formation F_n, which has a preceding activity ***m*** using a crew formation F_m, is computed using one or more of the following precedence relationships equations:

For finish-start precedence relationship:

$$STime(n, F_n) = FTime(m, F_m) + Lag(n,m) \qquad (1)$$

For start-finish precedence relationship:

$$STime(n, F_n) = STime(m, F_m) + Lag(n,m) - ActDur(n, F_n) \qquad (2)$$

For finish-finish precedence relationship:

$$STime(n, F_n) = FTime(m, F_m) + Lag(n,m) - ActDur(n, F_n) \quad (3)$$

For start-start precedence relationship:

$$STime(n, F_n) = STime(m, F_m) + Lag(n,m) \quad (4)$$

The start time ***FTime (n,F_n)*** of an activity n using a crew formation F_n is computed using the following equation:

$$FTime(n, F_n) = STime(n, F_n) + ActDur(n, F_n) \quad (5)$$

3.2 Profit Module

The primary purpose of this module is to determine the project final profit. For a contractor, the cash flow profile of expenses and incomes for a construction project typically follows the work-in-progress for which the contractor will be paid periodically for work completed less retainage. The contractor expenses which occurs more or less continously for the project duration are depicted by a piecewise continuous curve while the progress payments from the owner are represented by a step function. The owner's payments for the work completed are assumed to lag one period behind expenses except that the total retainage withheld is paid at the end of construction.

The equations in this section are presented conforming to those used by Au and Hendrickson (1986). Let ***CE [i]*** be the disbursments performed by the contractor during a typical period period ***t***, typically 1 month, where

$$CE\,[t] = \sum_{n \in AC_t} ActCost(n, F_n) \quad (6)$$

The term AC_t refers to the set of activities performed by the contractor during period ***t***. Let ***SE[t]*** be the disbursments performed by sub-contractors during a typical period ***t***, where

$$SE\,[t] = \sum_{n \in AS_t} ActCost(n, F_n) \quad (7)$$

The term AS_t refers to the set of activities performed by sub-contractors during period ***t***. Let ***IE[t]*** be the contractor's indirect cost during period ***t*** where

$$IE[i] = \begin{cases} C_0 + C_1 & if\ i = 1 \\ C_1 & otherwise \end{cases} \qquad i = 1,2,...., D \quad (8)$$

The terms C_o, C_1, and ***D*** refer to the initial cost (e.g., mobilization, permitting, plan review and similar costs), the indirect cost slope, and the project duration, respectively. Let ***TE[t]*** be the total disbursments during a typical period ***t*** where

$$TE[t] = CE[t] + SE[t] + IE[t] \quad (9)$$

Let ***CP[t]*** be the owner's payment to the contractor performed at the end of period ***t*** where

$$CP[t] = TE[t](1 - Retainage) \quad (10)$$

The term ***retainage*** refers to the percent retainage withheld by the owner and will be returned to the contractor with the last payment. Let ***SP(t)*** the contractor's payment to sub-contractors during period ***t*** where

$$SP[t] = SC[t](1 - Retainage) \quad (11)$$

The retainage withheld by the contractor will be returned to sub-contractors with the last payment. The cumulative cash flow at the end of the period ***t*** ($t \geq 1$) is ***CF[t]*** where

$$CF[t] = NF[t-1] + TE[t] \quad (12)$$

The net cash flow at the end of period ***t*** after receiving the payment is ***NF[t]***. At the end of the previous period (***t-1***) the net cash flow ***NF[t-1]*** is given by

$$NF[t-1] = CF[t-1] + TP[t-1] \tag{13}$$

It is worth noting that the above caluclations are based on the assumption that the contractor pays the interest charges at the end of period. The total interest charges at the end of period ***t*** are ***IP[t]*** where

$$IP[t] = r * NF[t-1] + r * \frac{TE(t-1)}{2} \tag{14}$$

Where r = interest rate per period. The first component of ***IP[t]*** represents the interest per period on the net cash flow ***NF[t-1]*** ; and the second component approximates the interest on **TE[t]** per period.

3.3 Multi-Objective Optimization Module

The objective of this module is to search for optimal/near-optimal trade-offs between project duration and total cost using a multi-objective genetic algorithm model. Genetic algorithms are search and optimization tools that assist decision makers in identifying optimal or near-optimal solutions for problems with large search space. They are inspired by the mechanics of evolution and they adopt the survival of the fittest and the structured exchange of genetic materials among population members over successive generations as a basic mechanism for the search process (Goldberg 1989). The present model is implemented in three major phases: (1) Initialization phase that generates an initial set of ***S*** possible solutions for the problem; (2) Fitness evaluation phase that calculates the duration and total profit of each generated solution; and (3) population generation phase that seeks to improve the fitness of solutions over successive generations.

3.3.1 Initialization

The main purpose of this phase is to initialize the optimization procedure in the present model, using the following two major steps.

1. Read project and genetic algorithm parameters needed to initialize the search process. The project parameters include: (1) Number of project activities; (2) Number of crew formations for each activity; (3) Activity duration and direct cost for each crew formation; (4) Lag/lead time between successive activities and their precedence relationships. The required genetic algorithm parameters for this initialization phase include: (1) String size; (2) Number of generations; (3) Population size; (4) Mutation rate; and (5) Crossover rate. The string size is determined by the model, considering the total number of activities in the analyzed project. The number of generations ***G*** and population size ***S*** are identified based on the selected string size in order to improve the quality of the solution. Similarly, the mutation and crossover rates are determined considering the population size and the method of selection employed by the algorithm.
2. Generate random solutions (*s= 1 to S*) for the initial population P_1 in the first generation (*g=1*). These solutions represent an initial set of activity crew formations. This set of possible solutions is then evolved in the following two phases in order to generate a set of activity optimal crew formations that establishes an optimal trade-off between project duration and profit.

3.3.2. Fitness function evaluation

The main purpose of this phase is to evaluate the duration and profit for each possible solution (*s*) in generation (*g*) in order to determine the fitness of the solution. This fitness determines the likelihood of survival and reproduction of each solution in following generations. As such, this phase evaluates the two identified fitness functions for each solution using the following two steps.

1. Calculate the project duration (***D(s,g)*** for solution (*s*) in generation (*g*)) using the procedure described in the scheduling module location.

2. Calculate the project total profit (***TotalProfit(s,g)*** for solution (s) in generation (g)) using the procedure described in the profit module location.

3.3.3. Population Generation

The purpose of this phase is to create three types of population in each of the considered generations: (1) Parent; (2) Child; and (3) Combined. For each generation (g), a parent population (P_g) is used to generate a child population (C_g) in a manner similar to that used in traditional genetic algorithms (Goldberg 1989). The purpose of generating this child population is to introduce a new set of solutions by rearranging and randomly changing parts of the solutions of the parent population. This child population can then be combined with the parent population to create an expanded set of possible solutions that forms the combined population (N_g) for generation (g). This combined population (N_g) is used to facilitate the comparison among the initial solutions in the parent population and those generated in the child population. The best solutions in this combined population regardless of their origin are retained and passed to the following generation as a parent population (Zitzler et al. 1999; Deb 2001; Deb et al. 2001; Deb et al. 2002). The computational procedure in this phase is implemented in the following steps.

a. Calculate Pareto optimal rank and crowding distance for each solution ($s=1$ to S) in the parent population (P_g). First, this is done by ranking the solutions in the population according to their Pareto optimal domination of other solutions, where a solution is identified as dominant if it is better than all other solutions in all of the optimization objectives considered simultaneously. Second, this step calculates the crowding distance of each solution, which represents the closeness of neighboring solutions to the solution considered. The crowding distance values help the algorithm spread the obtained solutions over a wider Pareto front instead of converging to points that cover only a small part of the tradeoff surface (Deb et al. 2001).
b. Create a new child population (C_g) using the genetic algorithm operations of selection, crossover, and mutation. The selection operation chooses the individuals that will go through the reproduction process, by favoring those with higher optimal ranks and wider crowding distances. The crossover operation, on the other hand, crosses each pair of the selected individuals at a randomly determined point and swaps the variables coded in the springs at this point, resulting in two new individuals. The mutation operation randomly changes the value of one of the variables in the string to induce innovation and to prevent premature convergence to local optima (Goldberg 1989). The fitness of the generated child population is then analyzed using the earlier described steps of Phase 2 in order to obtain the values of project duration and total profit for each solution.
c. Combine child population (C_g) and parent population (P_g) to form a new combined population (N_g) of size ***2S***. This combined population acts as a vehicle for the elitism, where good solutions of the initial parent population are passed on to the following generation to avoid the loss of good solutions of the initial parent population once they are found (Deb et al. 2001).
d. Calculate Pareto optimal rank and crowding distance for each solution ($s=1$ to $2S$) of the newly created combined population (N_g). This step performs the same operations as Step 1 of this phase on the new combined population (N_g).
e. Sort the new combined population (N_g) using the niched comparison rule. This sorting rule selects solutions with higher Pareto optimal ranks and breaks ties between solutions with the same rank by favoring solutions with higher crowding distances.
f. Keep the top ***S*** solutions from the combined population (N_g) to form the parent population (P_{g+1}) of the next generation. This parent population is then returned to Step 1 of this phase for generating a new child population. This iterative execution of the second and third phases of the model continues until the specified number of generations are completed.

4. Illustrative Example

A construction project is analyzed in order to illustrate the use of the present model and demonstrate its capabilities in generating and evaluating optimal trade-offs between project duration and profit. The project consists of 12 activities as shown in the project planning network of Figure 1. The activity durations and direct costs are shown in Table 1. The precedence relationships between successive activities were chosen to be Finish-Start with zero time lag. The indirect cost ($\boldsymbol{C_1}$) is estimated at $1500 per day with an initial indirect cost ($\boldsymbol{C_0}$) of $5000. The rate of crossover and mutation were set equal to 0.8 and 0.005, respectively. The population size and the number of generations were selected equal to 100 and 1500, respectively.

Two case studies were considered herein. In the first case study, all the activities were assumed to be performent by the contractor. In other words, all activity crew formations were assumed to belong to the contractor. In the second case study, the third actiovity crew formations were assumed to belong to sub-contractors. The project profits were determined for both case sudies using the presented multi-objective genetic algorithm model. Figure 2 summarize the results for both case studies. The results show that the profits obtained in the first case study were about 16% lower than those obtained in the second case study, which indicates that sub-contractors have increased the contractor's profit. This is due to the fact that delaying sub-contractors' payments reduces the interests paid by the contractor during the project execution.

This example illustrates the capability of the present model of (1) generating a set of feasible construction plans that establish optimal trade-offs between project duration and profit, (2) identifying optimum construction plans that support minimizing project duration and maximizing project profit, and (3) helping contractors/decision makers in construction projects to select an optimal plan that satisfies the specific requirements of the project being considered.

Table 1: Activity Costs and Durations

Activity No.	1			2			3			4			5			6			7			8			9			10			11			12		
Crew No.	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
Duration (Days)	2	3	4	4	5	6	7	8	9	8	9	10	2	3	4	4	5	6	7	8	9	10	11	12	2	3	4	5	6	7	4	5	6	7	8	10
Direct Cost (x $1000)	40	30	20	50	40	30	45	35	30	60	50	40	35	30	25	70	60	50	40	30	20	45	30	25	40	35	25	55	50	40	65	60	50	60	55	40

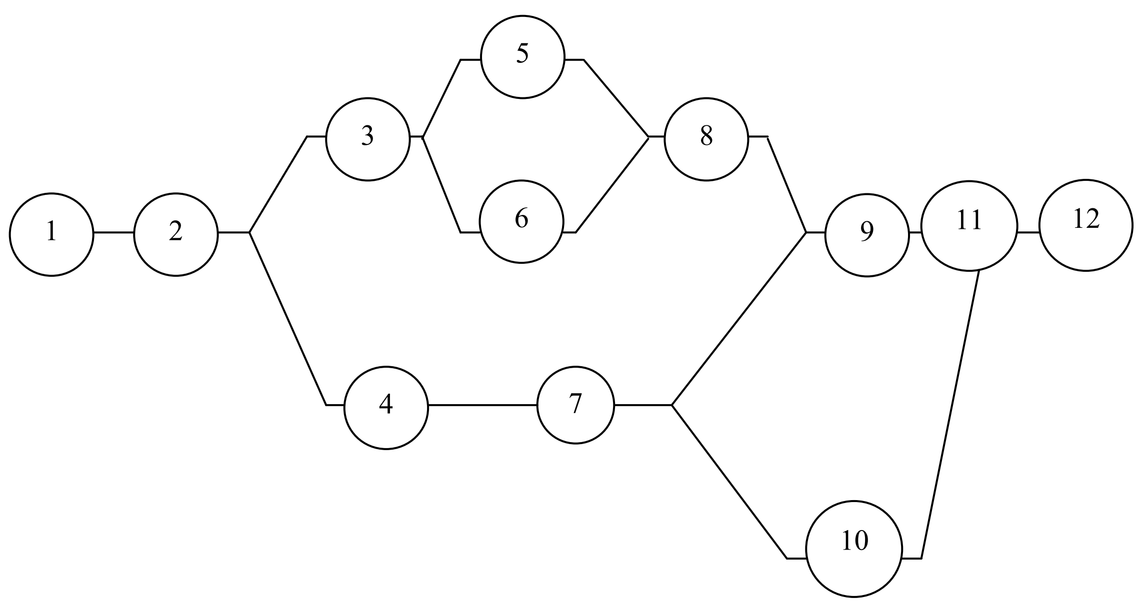

Figure 1: Project Planning Network

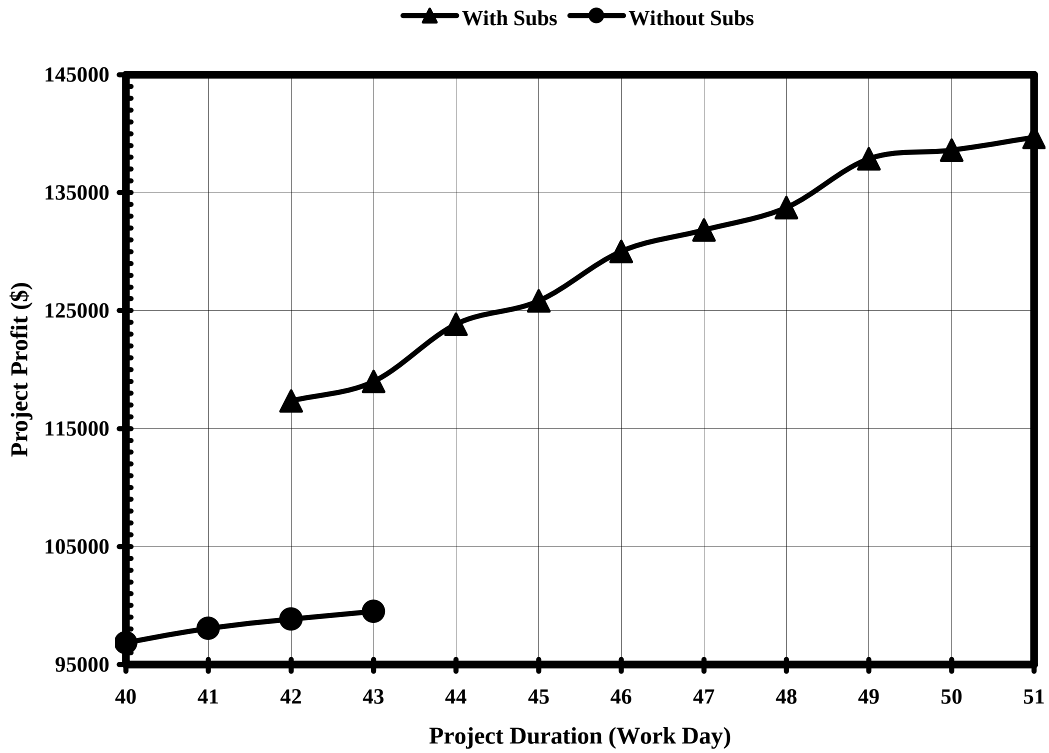

Figure 2: Project Duration-Profit Trade-off

5. Conclusions

A robust multi-objective optimization model was developed to support scheduling of construction projects. The model enables construction planners to generate and evaluate optimal construction plans that establish optimal trade-offs between project duration and profit. Each of these plans identifies, from a set of feasible alternatives, an optimal crew formation for each activity in the project. To accomplish this, the model incorporates (1) a scheduling module that calculate the project duration; (2) a profit module that computes the project total profit; and (3) a multi-objective optimization module that searches for and identifies near optimal construction plans. An application example was analyzed to illustrate the capabilities of the developed model in generating all optimal trade-off solution between project duration and profit in a single run, where each provides the maximum profit that can be achieved for a given project duration. This new capability should prove useful to construction planners and is expected to advance existing scheduling practices for construction projects.

6. References

Deb, K. (2001). *Multi-Objective Optimization using Evolutionary Algorithm,* John Wiley & Sons, LTD, New York, NY.

Deb, K., Agrawal, S., Pratap, A., and Meyarivan, T. (2001). "A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-objective Optimization", *KANGAL Report 200001, Genetic Algorithm Laboratory, Indian Institute of Technology,* Kanpur, India.

Deb, K., Pratap, A., Agrawal, S., and Meyarivan, T. (2002). “A Fast Elitist Multi-objective NSGA-II”, *IEEE Transactions on Evolutionary Computation,* IEEE, Vol. 6, No.2, pp. 182-197.

Elmaghraby, S. E. (1993). "Resource allocation via dynamic programming in activity networks", *European Journal of Operational Research*, Vol. 64, pp. 199-215.

Elazouni, A. M., and Metwally, F. G. (2005). "Finance-based scheduling:tool to maximize project profit using improved genetic algorithms", *Journal of Construction Engineering and Management*, ASCE, Vol. 131, No. 4, pp. 400-412.

Feng, C., Liu, L., and Burns, S. A. (1997). “Using Genetic Algorithms to Solve Construction Time-Cost Trade-Off Problems”, *Journal of Computing in Civil Engineering***,** ASCE, Vol. 11, No. 3, pp. 184-189.

Goldgerg, D. E. (1989). *Genetic Algorithms in search, optimization, and machine learning,* Addison-Wesley Publishing Company, Inc., Reading, Mass.

Hendrickson, C. and Au, T. (1986). *Project Management for Construction,* Prentice Hall,Englewood Cliffs, New Jersey

Leu, S. S and Yang, C. H. (1999). “GA-Based Multicriteria Optimal Model For Construction Scheduling”, *Journal of Construction Engineering and Management*, ASCE, Vol. 125, No. 3, pp. 420-427.

Moselhi, O. (1993). "Schedule Compression using the direct stiffness method", *Canadian Journal of Civil Engineering*, Vol. 20, pp. 65-72.

Zitzler, E., and Thiele, L. (1999). “Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach”, *IEEE Transactions on Evolutionary Computation*, IEEE, Vol. 3, No. 4, pp. 257-271.

Zheng, D., Ng, T., and Kumaraswamy, M. (2004). “Applying a Genetic Algorithm-Based Multiobjective Approach for Time-Cost Optimization”, *Journal of Construction Engineering and Management*, ASCE, Vol. 130, No. 2, pp. 168-176.