# EVOLUTIONARY OPTIMIZATION OF RESOURCE ALLOCATION IN REPETITIVE CONSTRUCTION SCHEDULES

SUBMITTED: October 2003
REVISED: September 2004
ACCEPTED: September 2005 at http://www.itcon.org/2005/18/
EDITOR: C. Anumba

***Khaled Nassar, Assoc Professor***
***University of Maryland Eastern Shore, Princess Ann, USA***
***email: knassar@umes.edu***

***SUMMARY:*** *Repetitive construction projects are common in a number of different situations such as highway construction and housing projects. Several approaches and models have been presented for modeling repetitive construction projects as well as for optimizing resource allocation for these types of projects. This paper presents a model that uses genetic algorithms to optimally assign resources to repetitive activities in order to minimize the overall project duration as well as the number of interruption days. The model presented in this paper is a straightforward formulation that can be easily modified and expanded for any problem through its spreadsheet implementation, which reduces the modeling time for typical problems extensively. The model outperforms other models in terms of finding a solution with less interruption days. In addition, users of the proposed model can interactively examine tradeoffs between the number of interruption days and the overall project duration.*

***KEYWORDS:*** *optimization, repetitive construction schedules, resource allocation problem*

## 1. INTRODUCTION

Repetitive activities are common in a number of construction projects. Repetitive construction is commonly found for example, in high-rise buildings, housing projects, highways, pipeline networks, bridges, etc. Several approaches and models have been presented for modeling repetitive construction projects, starting with the traditional network techniques such as Critical Path Method (CPM).

A number of researchers however, recognized early on the problems of using conventional network techniques for schedule such projects. The main draw back of conventional network techniques is their inability to maintain crew work continuity (Birrell 1980; Selinger 1980; Kavanagh 1985; Reda 1990; Russell and Wong 1993). In order to rectify this problem, a number of scheduling techniques have been proposed for repetitive construction projects (Selinger 1980; Johnston 1981; Arditi and Albulak 1986; Charzanowski and Johnston 1986; Russell and Caselton 1988; Al Sarraj 1990; El-Rayes and Moselhi 1998). Dynamic programming, for example, has also been extensively used to optimize the scheduling of repetitive projects, (Moselhi and El-Rayes 1993; Eldin and Senouci 1994, Selinger 1980; Russell and Caselton 1988). These formulations consider the overall project duration (Selinger 1980; Moselhi and El-Rayes 1993) or both the overall project duration and the number of interruption days (Russell and Caselton 1988; Eldin and Senouci 1994; and El-Rayes and Moselhi 2001).

Maintaining crew work continuity leads to maximizing the learning curve effect and minimizing the idle time of each crew (Ashley 1980; Birrell 1980). On the other hand, maintaining crew work continuity might result in longer project duration. Selinger (1980) for instance, suggested that increasing crew work interruptions up to a specific amount for some of the activities might reduce the overall project duration. Therefore, the tradeoff between the overall project duration and the number of interruption days is a crucial factor in assigning resources in repetitive projects.

In this paper, an alternative model that uses Genetic Algorithm (GA) to optimally assign resources for repetitive construction projects. The model provides an optimum duration for repetitive construction projects while applying the minimum crew work interruption days to the repetitive construction activities. The model proposed here is straightforward in that it can be implemented easily on any spreadsheet software and modified and expanded for any problem. Furthermore, using the proposed genetic algorithms optimization, the model achieves solutions that can be better than those achieved using other models.

## 2. RESOURCE ASSIGNMENT IN REPETITIVE CONSTRUCTION SCHEDULES

In order to explain the problem of resource assignment in repetitive construction schedules, consider the concrete bridge example which was first presented in (Selinger 1980). Russell and Caselton (1988), and Moselhi and EL-Rayes (1993 & 2001) also analyzed this same example. In the section 4 of this paper, the results from the model presented in this paper are compared to the solutions from these studies. The project consists of four similar sections or units, and each includes the following repetitive activities in sequence: Excavations, Foundations, Columns, Beams, and Slabs. Each repetitive activity is performed by a crew that progresses from the first to the fourth section sequentially. The job logic (i.e., relationships) among succeeding activities, for the project selected here, is finish to start with no lag time. The same technique however can also work with other kinds of relationships (start-to-finish, start-to-start, etc…).

*TABLE 1: Sample Problem (from Selinger 1980)*

| | **Activity *i*** | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Unit *j*** | **Excavation** | **Foundation** | | | **Columns** | | | **Beams** | | | | **Slabs** | |
| | **Crew Option # 1** | **Crew option # 1** | **Crew option # 2** | **Crew option # 3** | **Crew option # 1** | **Crew option # 2** | **Crew option # 3** | **Crew option # 1** | **Crew option # 2** | **Crew option # 3** | **Crew option # 4** | **Crew option # 1** | **Crew option # 2** |
| **1** | 12.5 | 11.5 | 14 | 19.2 | 18.1 | 18.1 | 15.1 | 12.9 | 8.6 | 10 | 12 | 15 | |
| **2** | 15.6 | 12 | 15 | 20 | 15 | 15 | 12.5 | 10.7 | 9.3 | 10.8 | 13 | 16.2 | 15.8 |
| **3** | 10.8 | 10.5 | 13 | 17.5 | 22.5 | 22.5 | 18.7 | 16.1 | 10.2 | 11.9 | 14.2 | 17.8 | 13 |
| **4** | 16.7 | 10 | 13 | 16.7 | 17.5 | 17.5 | 14.6 | 12.5 | 8 | 9.4 | 11.2 | 14.1 | 16.7 |

Different crew formations options are available for each activity as listed in Table 1 (excavation for example has only 1 crew option, while the beams activity has 4 crew options). Each crew formation has its own production rate. In addition to determining the optimum crew assignment that minimizes the overall project duration, one also needs to determine the best interruption vector associated with this assignment. Previous researchers have defined the interruption vector is the number of days that the assigned crews will be interrupted between the different work units (Caselton 1988, Moselhi and EL-Rayes 1993, 2001). Allowing crew work interruptions for some of the activities might reduce the overall project duration.

The resource assignment problem in repetitive construction schedules therefore is to determine the optimum crew formation and interruption vector that will reduce the overall project duration. In previous formulations, some of them utilize an arbitrary set of interruption vectors prepared prior to scheduling (Russell and Caselton 1988; Eldin and Senouci 1994). El-rayes and Moselhi 2001 applied an interruption to their model that automatically generates a set of calculated interruption vectors during scheduling to make the interruption more feasible and bounded. In the model presented in this paper however, this is not required, as will be presented in the next section.

## 3. THE DEVELOPED MODEL

### 3.1 Model description

The developed model is a two-state variable model because it considers minimizing the total overall duration of the project as well as minimizing the number of interruption days (in the interruption vector). The model is presented in a way that is similar to traditional CPM calculations and therefore can be easily developed for any project. In addition, the model is developed in a table format which facilitates its implementation in any spreadsheet software as well as utilizing the already developed optimization routines found in such software, such as the genetic algorithm solver in excel, which is utilized in this case.

Table 2 presents the calculations for the sample problem described above. The calculations are divided into three main parts:

- Firstly, part one calculates the earliest start date for each of the activities.
- Secondly part 2 calculates the latest start date.
- Finally, the third part is used to determine when to actually schedule the start of each activity to minimize the overall duration and also minimize the interruptions. The “Interruption Vector”

column (column 13) shows the number of days that each activity is delayed, and the finish day of the last activity is the overall total duration of the project (106.8 days).

*TABLE 2.: Table used in the Proposed Model*

| Unit | *Part 1 (Earliest Start)* | | | | | | | *Part 2 (Latest Start)* | | | *Part 3, Interruption Stage "Optimum Solution"* | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
| | Earliest possble start due to crew avail. SCA | Earliest possible start due Job Logic SJL | Idle Time | Duration | Crew No. | Early Start Date | Early Finish Date | Shift | Late Start date | Late Finish Date | Sched. Start Date | Sched. Finish Date | Interruption Vector |
| **Excavation** | | | | | | | | | | | | | |
| 1 | 0 | 0 | 0.0 | 12.5 | 1 | 0.0 | 12.5 | 0.0 | 0.0 | 12.5 | 0.0 | 12.5 | 0 |
| 2 | 12.5 | 0 | 0.0 | 15.6 | 1 | 12.5 | 28.1 | 0.0 | 12.5 | 28.1 | 12.5 | 28.1 | 0 |
| 3 | 28.1 | 0 | 0.0 | 10.8 | 1 | 28.1 | 38.9 | 0.0 | 28.1 | 38.9 | 28.1 | 38.9 | 0 |
| 4 | 38.9 | 0 | 0.0 | 16.7 | 1 | 38.9 | 55.6 | 0.0 | 38.9 | 56 | 38.9 | 55.6 | 0 |
| | | | | | | | | | | | | | |
| **Foundation** | | | | | | | | | | | | | |
| 1 | 12.5 | 12.5 | 0.0 | 11.5 | 1 | 12.5 | 24.0 | 9.1 | 21.6 | 33.1 | 15.7 | 27.2 | 0 |
| 2 | 24.0 | 28.1 | 4.1 | 12.0 | 1 | 28.1 | 40.1 | 5.0 | 33.1 | 45.1 | 28.1 | 40.1 | 1 |
| 3 | 40.1 | 38.9 | 0.0 | 10.5 | 1 | 40.1 | 50.6 | 5.0 | 45.1 | 55.6 | 40.1 | 50.6 | 0 |
| 4 | 50.6 | 55.6 | 5.0 | 10.0 | 1 | 55.6 | 65.6 | 0.0 | 55.6 | 65.6 | 55.6 | 65.6 | 5 |
| **Columns** | | | 9.1 | | | | | | | | | | 6 |
| 1 | 27.2 | 27.2 | 0.0 | 12.9 | 3 | 27.2 | 40.1 | 0.0 | 27.2 | 40.1 | 27.2 | 40.1 | 0 |
| 2 | 40.1 | 40.1 | 0.0 | 10.7 | 3 | 40.1 | 50.8 | 0.0 | 40.1 | 50.8 | 40.1 | 50.8 | 0 |
| 3 | 50.8 | 50.6 | 0.0 | 16.1 | 3 | 50.8 | 66.9 | 0.0 | 50.8 | 66.9 | 50.8 | 66.9 | 0 |
| 4 | 66.9 | 65.6 | 0.0 | 12.5 | 3 | 66.9 | 79.4 | 0.0 | 66.9 | 79.4 | 66.9 | 79.4 | 0 |
| **Beams** | | | 0.0 | | | | | | | | | | 0 |
| 1 | 40.1 | 40.1 | 0.0 | 8.6 | 1 | 40.1 | 48.7 | 11.2 | 51.3 | 59.9 | 43.4 | 52.0 | 0 |
| 2 | 48.7 | 50.8 | 2.1 | 9.3 | 1 | 50.8 | 60.1 | 9.1 | 59.9 | 69.2 | 52.0 | 61.3 | 0 |
| 3 | 60.1 | 66.9 | 6.8 | 10.2 | 1 | 66.9 | 77.1 | 2.3 | 69.2 | 79.4 | 66.9 | 77.1 | 6 |
| 4 | 77.1 | 79.4 | 2.3 | 8.0 | 1 | 79.4 | 87.4 | 0.0 | 79.4 | 87.4 | 79.4 | 87.4 | 2 |
| **Slabs** | | | 11.2 | | | | | | | | | | 8 |
| | | | | | | | | | | | | | |
| 2 | 61.3 | 61.3 | 0.0 | 15.8 | 1 | 61.3 | 77.1 | 0.0 | 61.3 | 77.1 | 61.3 | 77.1 | 0 |
| 3 | 77.1 | 77.1 | 0.0 | 13.0 | 1 | 77.1 | 90.1 | 0.0 | 77.1 | 90.1 | 77.1 | 90.1 | 0 |
| 4 | 90.1 | 87.4 | 0.0 | 16.7 | 1 | 90.1 | 106.8 | 0.0 | 90.1 | 106.8 | 90.1 | 106.8 | 0 |
| | | | 0 | | | | | | | | | | 0 |
| | | | | | | | | | | | | | 14 |

The inputs to the problem presented here are shown in TABLE 1, which shows the different crew durations for the different activities. The crew duration is the amount of time that each crew *n* would take to finish each unit *j* of activity *i*.

## 3.2 Model Explanation

### 3.2.1 Part One of the Model

Now, consider part 1 of the model in TABLE 2. Column 1 contains the earliest possible start date according to crew availability, $SCA^i_{nj}$. For the first activity in the first unit (no succeeding activity) this is equal to zero. For the next unit, $SCA^i_{nj}$ is equal to the early finish date of the preceding unit of the same activity. It is determined using the following formula:

$$SCA^i_{nj} = \text{Early Finish Date}^i_{n(j-1)} (\text{column7}) \quad (1)$$

The early finish date for a unit *j* of an activity *i* is based on the crew *n* selected. The crew used for each activity is shown in "Crew" column (column 5). For example, the earliest possible time that the third unit of the "foundation" activity can start (40.1) is determined from the earliest finish date of the second unit. Note that since crew number 1 is used (as shown in the "Crew no" column, number 5) for the foundation activity, the duration, Dinj, for any of the units of the activity considered is actually the crew duration (shown in Table 1). If a different crew were selected then the duration (column 4) for the units would reflect the new crew selected in column 5.

Column number 2, on the other hand, contains the earliest possible start date according to job logic, SJLinj. This represents the earliest date the activity could start if there were no constraints due to the crew duration. This is calculated based on the scheduled finish date or the same unit of the preceding activity, and is given by:

$$SJL_{nj}^{1} = \text{Scheduled Finish date}_{nj}^{i-1}(\text{Column12}) \quad (2)$$

Since an activity can not start until its predecessor has finished or until the crew working on the activity is available, the earliest start date (column 6) for each activity would have to be the latest of the (SJL or SCA). This is calculated according to the following formula:

$$\text{Early Start Date}_{nj}^{i} = \text{MAX}(SJL_{nj}^{i}, SCA_{nj}^{i}) \quad (3)$$

For any given activity, if the earliest start date according to job logic, SJL, were later than the earliest start due to crew availability, SCA, then this would mean that the activity could start immediately since the crew would also be available to work immediately. In this case, there is no idle time for the activity. On the other hand, if the start earliest start date according to job logic, SJL, were earlier than the earliest start due to crew availability, SCA, then the activity would have to sit idle until the crew has finished working on the preceding unit. The time the activity sits idle in that case is the difference between the SJL and the SCA. Idle time is shown in column 3 and is calculated according to for the following formula:

$$\text{Idle}_{nj}^{i} = \left\{ \begin{array}{l} 0, \text{if } (SJL_{nj}^{i} - SCA_{nj}^{i}) \le 0 \\ SJL_{nj}^{i} - SCA_{nj}^{i}, \text{if} (SJL_{nj}^{i} - SCA_{nj}^{i}) > 0 \end{array} \right\} \quad (4)$$

Finally, the early finish date (column 7) for any unit in an activity is simply the sum of the early start date and the duration. This is given by the following formula:

$$\text{Early Finish Date}_{\text{nj}}^{\text{i}}(\text{column7}) = \text{Early Start Date}_{nj}^{i}(\text{column6}) + D_{nj}^{i}(\text{column 4}) \quad (5)$$

This concludes the first part of the model, which determines the early starts and early finish dates, as well as the amount of idle time for any units of an activity due to lack of resources (because the crews would be still working on the previous unit of the activity).

### 3.2.2 Part Two of the Model

In the second part of the model the late start and late finish dates are calculated. The idea here is that since some of the units in an activity would have to sit idle anyways (due to the crew availability), one could shift the start of preceding units accordingly. The amount that each unit could be delayed without delaying the overall project finish date is called the shift. The shift (column 8) is equal to the sum of all the idle times of the preceding units in an activity and is given by the following formula:

$$\text{s}hift_{nj}^{i} = \sum_{j+1}^{j} Idle_{nj}^{i} \quad (6)$$

Since the shift determines the amount that each unit could be delayed without delaying the overall project finish date, the late start date (column 9) would simply equal the early start plus the shift, and the late finish is equal to the late start plus the duration:

$$\text{Late Start Date}_{nj}^{i}(\text{column 9}) = \text{Early Start Date}_{nj}^{i}(\text{column 6}) + shift_{nj}^{i}(\text{column 8}) \quad (7)$$

$$\text{Late Finish Date}_{nj}^{i}(\text{column 10}) = \text{Late Start Date}_{nj}^{i}(\text{column 9}) + D_{nj}^{i}(\text{column 4}) \quad (8)$$

### 3.2.3 Part Three of the Model

Now that the late and early start dates for each unit have been calculated, the third part of the model is used to determine the overall finish date of the project given specific interruptions (or shifts) for each unit of the activities. The work on specific units can be interrupted for a number of days without increasing the overall project duration. The number of days that each activity can be interrupted is shown in the "Interruption" column

(column 13). In fact by varying the number of interruption days (i.e. the interruption vector) and the crews, a reduction in the overall project duration can be achieved. The scheduled start date of any unit in the project would equal the late start date minus those interruption days. Although the scheduled start date can also be determined as the earliest start plus the interruption days, the way the spreadsheet is setup makes it more readable to use the late start minus the interruption days. The scheduled start and finish dates are therefore defined as:

$$\text{Scheduled Start Date}^{i}_{nj}\,\text{(column 11)} = \text{Late Start Date}^{i}_{nj}\,\text{(column 9)} - \sum_{j+1}^{j}\text{Inter}^{i}_{nj}\,\text{(column 13)} \qquad (9)$$

$$\text{Scheduled Finish Date}^{i}_{nj}\,\text{(column 12)} = \text{Scheduled Start Date}^{i}_{nj}\,\text{(column 11)} + D^{i}_{nj}\,\text{(column 4)} \qquad (10)$$

*TABLE 3: Comparison between the results of the proposed model and other similar models for the same problem*

| | Activity *i* | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Unit *J* | Excavation | | Foundation | | Columns | | Beams | | Slabs | |
| | Inter | S | Inter | S | Inter | S | Inter | S | Inter | S |
| One-state variable formulation (Selinger 1980) | | | | | | | | | | |
| 1 | - | 0 | - | 13.7 | - | 32.6 | - | 47.3 | - | |
| 2 | - | 12.5 | - | 28.1 | - | 45.5 | - | 59.3 | - | 72.3 |
| 3 | - | 28.1 | - | 43.1 | - | 56.2 | - | 72.3 | - | 88.1 |
| 4 | - | 39 | - | 56.2 | - | 72.3 | - | 86.6 | - | 101.2 |
| Finish | - | 55.6 | - | 68.7 | - | 84.8 | - | 97.99 | - | 117.9 |
| Optimum Crew Formation | 1 | 1 | 2 | 2 | 3 | 3 | 3 | 3 | 1 | 1 |
| Two-state variable formulation (Russel and Cselton 1988) | | | | | | | | | | |
| 1 | 0 | 0 | 0 | 17.6 | 0 | 29.1 | 0 | 43 | - | - |
| 2 | 0 | 12.5 | 0 | 29.1 | 0 | 42.1 | 4 | 55.6 | 0 | 64.9 |
| 3 | 0 | 28.1 | 0 | 41.1 | 0 | 52.8 | 4 | 68.9 | 0 | 80.7 |
| 4 | 0 | 39 | 4 | 55.6 | 0 | 68.9 | 4 | 83 | 0 | 93.7 |
| Finish | - | 55.6 | - | 65.6 | - | 81.4 | - | 91 | - | 110.4 |
| Optimum Crew Formation | 1 | 1 | 1 | 1 | 3 | 3 | 1 | 1 | 1 | 1 |
| Two-state variable formulation (El-Rayes and Moselhi 2001) | | | | | | | | | | |
| 1 | 0 | 0 | 0 | 15.6 | 0 | 27.2 | 0 | 43 | - | - |
| 2 | 0 | 12.5 | 1 | 28.1 | 0 | 0.1 | 0 | 51.6 | 0 | 61.3 |
| 3 | 0 | 28.1 | 0 | 40.1 | 0 | 50.8 | 6 | 66.9 | 0 | 77.1 |
| 4 | 0 | 39 | 5 | 55.6 | 0 | 66.9 | 3 | 80.1 | 0 | 90.1 |
| Finish | - | 55.6 | - | 65.6 | - | 79.4 | - | 88.1 | - | 106.8 |
| Optimum Crew Formation | 1 | 1 | 1 | 1 | 3 | 3 | 1 | 1 | 1 | 1 |
| Proposed Two-state variable formulation | | | | | | | | | | |
| 1 | 0 | 0 | 0 | 15.6 | 0 | 27.2 | 0 | 43 | - | - |
| 2 | 0 | 12.5 | 1 | 28.1 | 0 | 0.1 | 0 | 51.6 | 0 | 61.3 |
| 3 | 0 | 28.1 | 0 | 40.1 | 0 | 50.8 | 6 | 66.9 | 0 | 77.1 |
| 4 | 0 | 39 | 5 | 55.6 | 0 | 66.9 | 2 | 80.1 | 0 | 90.1 |
| Finish | - | 55.6 | - | 65.6 | - | 79.4 | - | 88.1 | - | 106.8 |
| Optimum Crew Formation | 1 | 1 | 1 | 1 | 3 | 3 | 1 | 1 | 1 | 1 |

The values used in the “Interruption Vector” column (column 13), as well as those in the “Crew” column (column 5), which specify which of the different crews will be used for each activity, will determine the overall project duration. These values were determined through an optimization routine as described in the next section. The spreadsheet cells in the “Interruption Vector” column (column 13) were used as the variables in the optimization problem, which was used to minimize the overall project duration.

### 3.2.4 Computerized Solutions using Evolutionary Add-in

One of the advantages of the formulation described above is that it can be easily developed for any size problem on any commercial spreadsheet application. Spreadsheets provide a transparent way of modeling the problem, giving the user a complete understanding and control of the different variables and constraints in the problem. For the majority of construction problems, whose size usually ranges from small to medium, spreadsheet solutions would be the most efficient and economical method to solve the problem. Most contractors already use spreadsheets extensively. Furthermore, the model can be easily modified and expanded for any problem by simple “formula dragging” which reduces the modeling time for typical problems extensively.

The developed spreadsheet uses standard spreadsheet formulae to model the problem. For example, an HLOOKUP function is used to look up duration of each unit based on the crew used. Spreadsheet, such as Excel, provides a solver add-in, which can be used for optimizing problems like the one presented here. However, traditional solvers can not be used to optimize this problem, because the spreadsheet involves discontinuous and non-smooth functions such as the MAX() and LOOKUP() functions.

Fortunately, a number of commercial genetic solver add-ins are available for excel that can maximize the speed and robustness for solving optimization problems such as the one presented above, e.g. Evolver and the Premium Solver which is used here. The Premium Solver Platform works with existing Excel Solver models to solve much larger problems up to hundreds of times faster. This evolutionary solver uses the powerful optimization capabilities of generic algorithm (GA) techniques. GA techniques are inspired by biological systems that improve fitness through evolution. This is done through mimicking Darwinian principles of natural selection by creating an environment where hundreds of possible solutions to a problem can compete with one another and only the "fittest" survive. Just as in biological evolution, each solution can pass along its good "genes" through "offspring" solutions so that the entire population of solutions will continue to evolve better solutions. In terms of our application, good “genes” would represent the schedule with the least duration and with the least interruption days.

The objective of the model presented here is to determine the optimum crew work formation and interruption vector that will minimize the overall project duration, while also keeping the number of interruption days to a minimum. Therefore the objective function is set as the overall project duration plus the sum of the interruption days. The crew work formation type and work interruption for each unit were set as the changing variables.

It is a requirement when using evolutionary solver that independent variable must have an upper and lower limit. Therefore upper limit and lower limit constraints were specified for all variables. For example, the upper limit on the crew variable was set as 3 for the foundations, 3 for the columns, 4 for the beams and 2 for the slabs. Another constraint to satisfy is that the activity in each repetitive unit cannot start at an earlier date than that established in part 1 of the model because of the job logic and crew availability constraints. After running the optimization, the optimum overall project duration was determined as 106.8 with associated total interruption of 14 days.

## 4. COMPARISON TO EXISTING MODELS

The same problem was analyzed by using dynamic programming formulations. The initial problem formulation by Slinger (1980) Russell and Caselton (1988) did not consider interruption as a second state variable and resulted in a minimum duration of 117.9 days. Russel and Caselton (1988), which expanded the formulation allowing 16 interruption days, provided an improvement by minimizing the overall duration days to 110.4 days. The last formulation (El-Rayes and Moselihi (2001) provided an additional improvement by reducing the overall project duration to 106.8 days, and interrupting the crew work continuity by a total of 15 days, as shown in Table 4.2.

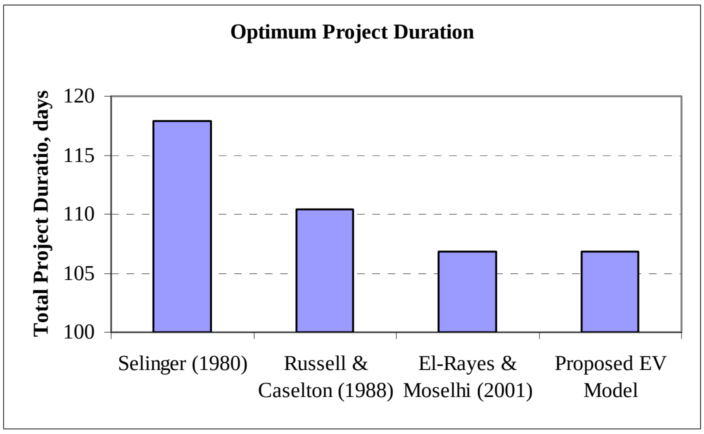


*FIG. 1: Comparison of optimum project duration*

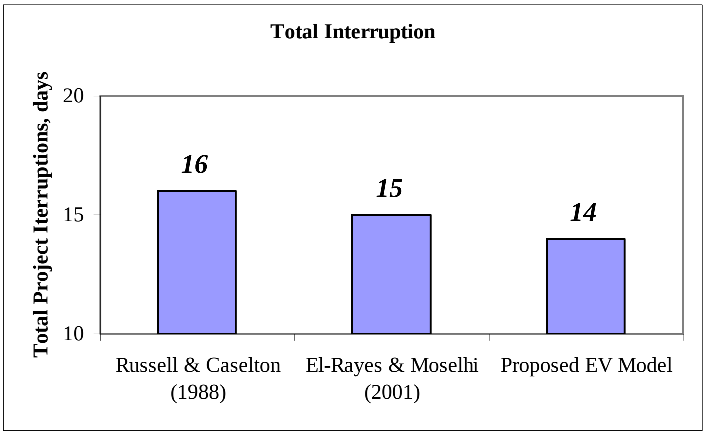


*FIG. 2: Comparison of total interruptions (days)*

*TABLE 4: Optimum Project Duration & Interruption*

| **Formulation Model** | **Total Duration** (days) | **Interruption** (days) |
|---|---|---|
| Selinger (1980) | 117.9 | - |
| Russell & Caselton (1988) | 110.4 | 16 |
| El-Rayes & Moselhi (2001) | 106.8 | 15 |
| Proposed GA Model | 106.8 | 14 |

The results obtained using dynamic formulations (Selinger 1980; Russell and Caselton 1988; El-Rayes and Moselhi 2001) are compared to those provided by the spreadsheet model as shown in Table 3, Fig 1, Fig 2 and Table 4, confirm that the GA model provides a more efficient solution than those obtained by others, the minimum overall duration is the same as that of (El-Ryes and Moselhi 2001), but with less interrupting days for crew work continuity.

In addition, one of the advantages of the model described here is the ability to interactively modify the variable in the model, such as the interruption vector, and observe the effects on the overall duration. For example, by increasing the interruption days for the fourth unit in the foundation activity from 5 to 7, the overall duration of the project can still be further reduced to 105.5, while increasing the interruption days by those two days. This allows users to examine tradeoffs between the number of interruption days and the overall project duration.

## 5. DISCUSSIONS AND CONCLUSIONS

In this paper a model for resource allocation in repetitive construction schedules was described. Using the proposed spreadsheet GA implementation, users can minimize the overall project duration while keeping the interruptions as low as possible. While most optimizers keep only the best solution found during its search, the evolutionary optimizer keeps a large set of results, called a population of candidate solution. The population is used to help create new starting solution points not necessarily in the neighborhood of the current best solution, and thus helping to avoid Evolutionary Solver getting trapped at a local optimum. Therefore, the model users can experiment with the different variables in the model to perform a sensitivity analysis on the model.

While the model presented here can easily be implemented for any similar problem as has been found to outperform similar models in terms of minimizing interruption days, a number of points need to be pointed out. The model is non-deterministic as it relies in part on randomly determined starting point, this may yield different solutions for different runs. The model also has the same restrictions as similar models in the literature: Labor requirement and feasible crew size are determined in advance. Each crew formation has a unique daily output and associated duration for completing the activity as well as the assumption that work interruptions may reduce total construction time, and so general expenses.

## 6. REFERENCES


Al Sarraj, Z. M. (1990). "Formal development of-Line Balance Technique." J. Constr. Engrg. and Mgmt., ASCE, 116(4), 689-704.

Arditi, D., and Albulak, M. Z. (1986). "Line of-Balance Scheduling in Pavement Construction." J. Constr. Engrg. and Mgmt., ASCE, 112(3), 411-424.

Ashley, D. B. (1980). "Simulation of Repetitive-Unit Construction." J. Constr. Engrg. and Mgmt., ASCE, 106(2), 185-194.

Beans, J. C. (1994). "Genetic Algorithms and Random Keys for Sequencing and Optimization." ORSA J. on Comp., 6(2), 154-160.

Birrell, G. S. (1980). "Construction Planning - Beyond the Critical Path." J. Constr. Engrg. and Mgmt., ASCE, 106(3), 389-407.

Chrzanowski, E. N. Jr., and Johnston, D. W (1986). "Application of Linear Scheduling." J. Constr. Engrg. and Mgmt., ASCE, 112(4), 476-491.

Eldin, N. N., and Senouci, A. B. (1994). "Scheduling and Control of Linear Projects." Can. J. Civ. Engrg., Ottawa, 21, 219-230.

El-Rayes, K., and Moselhi, O. (1998). "Resource-Driven Scheduling of Repetitive Activities on Construction Projects." J. Constr. Engrg. and Mgmt., ASCE, 107(2), 18-27.

El-Rayes, K., and Moselhi, O. (2001). "Optimizing Resource Utilization for Repetitive Construction Projects." J. Constr. Engrg. and Mgmt., ASCE, 16(4), 433-446.

Frontline System Inc., developers of Spreadsheet Solver/Optimizer in Microsoft Excel, http://www.frontsys.com.

Goldberg, D. E. (1989). "Genetic Algorithm in Search, Optimization, and Machine Learning, Addison- Wesley, Reading, Mass.

Johnston, D. W. (1981). "Linear Scheduling Method for Highway Construction." J. Constr.Div., ASCE, 107(2), 247-261.

Kavanagh, D. P. (1985). "SIREN: A Repetitive Construction Simulation Model." J. Constr.Div., ASCE, 111(3), 308-323.

O'Brien, J. J., Kreitzberg, F. C., and Mikes, W. F. (1985). "Network Scheduling Variations for Repetitive work." J. Constr. Engrg. and Mgmt., ASCE, 111(2), 105-116.

Reda, R. M. (1990). "RPM: Repetitive Project Modeling." J. Constr. Engrg. and Mgmt., ASCE, 116(2), 316-330.

Russell, A. D., and Caselton, W. F. (1988). "Extensions to Linear Scheduling Optimization." J. Constr. Div., ASCE, 114(1), 36-52.

Russell, A. D., and Wong, W. C. M. (1993). “New Generation of Planning Structures.” J. Constr. Div., ASCE, 119(2), 196-214

Senouci, A., and Eldin, N. (1996). “A Time-Cost Trade-Off Algorithm for Nonserial Linear Projecrs.” Can. J. Civ. Engrg., Ottawa, 23, 134-149.