

## CRASHING-RISK-MODELING SOFTWARE (CRMS)

Nabil Semaan<sup>1</sup>, Najib Georges<sup>2</sup> and Joe Rizk<sup>3</sup>

<sup>1</sup>Assistant Professor, <sup>2</sup>Associate Professor, <sup>3</sup>Senior Lecturer

Civil Engineering Department, Faculty of Engineering, University of Balamand,  
El-Koura District, Lebanon

E-mails: <sup>1</sup>nabil.semaan@balamand.edu.lb <sup>2</sup>najib.georges@balamand.edu.lb  
<sup>3</sup>joe.rizk@balamand.edu.lb

**ABSTRACT:** According to Price Water house Coopers, 80% of the mega projects are delayed in Arab gulf. Crashing a schedule in project management is an important tool that finds solutions to delayed schedules, specifically in time and cost management. Traditional methods in schedule crashing are time consuming, and prone to human error. Moreover, incorporating risk in schedule crashing is even an impossible task for project managers. An automated crashing tool that considers risk and uncertainty in the activity durations and costs is developed and presented in this paper. The Crashing-Risk-Modeling Software (CRMS) is developed. CRMS defines the schedule network, and identifies a triangular distribution for each of the activities normal and crashed duration, and normal and crashed cost as well. CRMS uses Monte Carlo simulation in order to iterate the evaluation of the traditional slope equation. Then CRMS identifies the most occurred critical path, and the least crashed cost. Finally, CRMS shows a normal distribution of the least crashed cost with its respective critical path.

**Keywords:** Crashing Risk Modeling; Monte Carlo Simulation; Schedule Crashing.

### 1. INTRODUCTION

According to Price Water house Coopers, 80% of the mega projects are delayed in Arab gulf (Attwood, 2012). Crashing a schedule is one of the most widely used methods in order to save cost and time in a project. There are many methods that are used for project crashing. Although very lengthy, one of the most effective and accurate methods is the traditional method. The traditional method is a manual one, based on evaluating the normal and crashed durations, normal and crashed cost, then determine the slope of the difference of the crashed to normal cost over the crashed to normal durations for each activity on the critical path. Georges et al. (2014) developed an automated tool entitled 'CRASH' that performs the manual calculation in a fast and transparent manner. However, one major limitation of the CRASH automated tool that it is based on deterministic values of costs and durations. The major problem facing project managers is the need to consider for uncertainty in the schedule

and cost, thus develop stochastic schedules. Monte Carlo simulation is a powerful tool that is applied worldwide in order to model a stochastic schedule.

This paper's objective is to present the CRASH-RISK-MODELING Software (CRMS). This model performs automated crashing of a schedule, while considering the activities durations and costs as probabilities and hence evaluates a stochastic crashed schedule result. The CRMS inputs the schedule network diagram, the activities durations as triangular distributions, the activities costs as triangular distributions, then uses Monte Carlo simulation in order to determine the crashed cost and critical path as a normal distribution.

## **2. BACKGROUND**

Lima *et al.* (2006) compared the use of three schedule crashing methods. The first is the brute force method, the second is a Linear Programming model, and the third is the traditional method. The authors observed that the traditional method resulted in the least increase in the total project direct cost.

Brunnhoeffer *et al.* (2010) developed an algorithm that can be used in Microsoft Excel. The algorithm has nine steps. First, determine all the possible paths through the CPM diagram, then determine the duration of those paths. Afterwards, compute the cost to crash each task, and select the lowest cost that will affect the project duration and modify the project cost. This basic method can be re-run each time. In this manner, he developed an excel spreadsheet, where all those calculations are done. The authors concluded that this method is better than manual calculations and will eliminate the human error, once the number of critical paths increases. Moreover, they stated that it will be helpful if the possible paths were determined a priori.

Celik *et al.* (2011) introduced a new software SPE Beta v.1, based on Brunnhoeffer research. The software requires the user to define a text file where the name of the activity and its predecessors are entered. Then, it will generate all the possible paths for the network diagram that was entered. This software developed with C++ language is still in Beta version.

Li *et al.* (2012) developed a crashing model in Microsoft Excel Solver, where project activities are listed with their information. Possible paths, critical paths and project durations are determined manually a priori, and using a set of constraints, the Excel Solver will determine the added cost when crashing to achieve a project deadline.

Georges *et al.* (2014) developed the CRASH automated tool. The CRASH automated tool performs the manual crashing procedure using a software. The main advantage of this

automated tool is the simplicity in input, and the speed to perform calculations. In addition, CRASH tool considers several critical paths and points out to the least crashed cost.

Until now, academic research has failed in developing a model for crashing using stochastic activities durations and costs. CRMS solves this major problem by developing software that evaluates a stochastic crashed schedule.

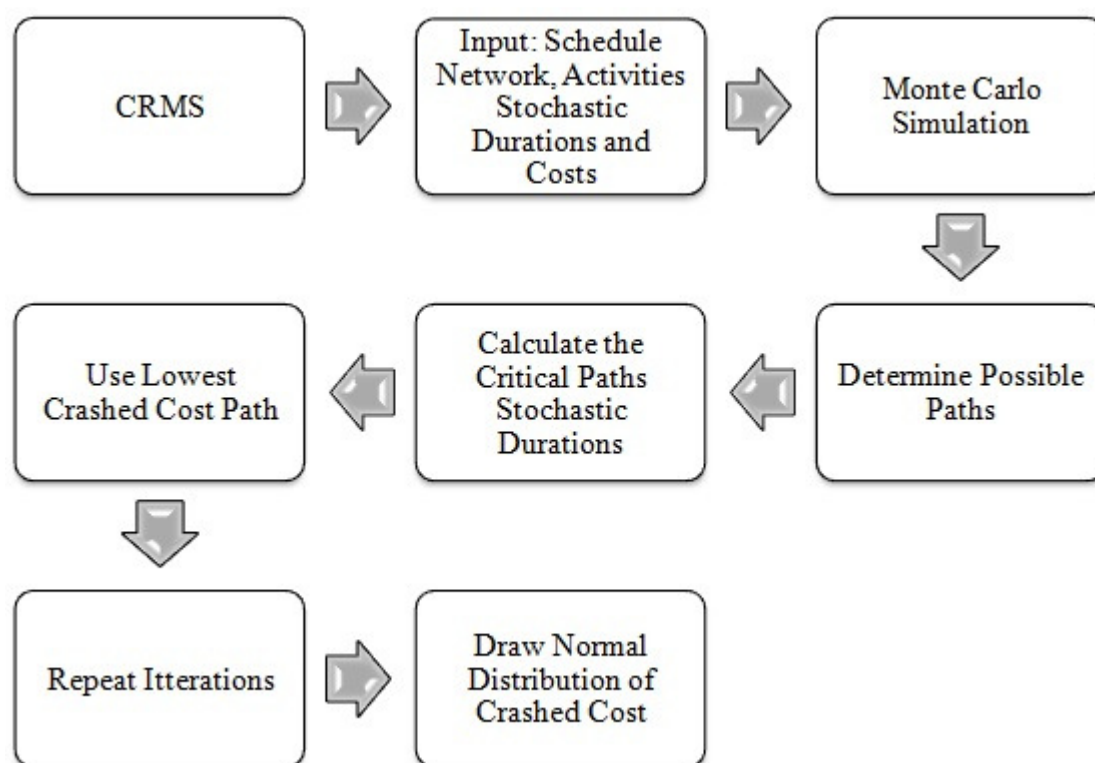
### **3. MONTE CARLO SIMULATION**

Every choice made in engineering or construction management involves a risk factor. Future events can't be predicted by our experience, knowledge or guessing no matter how much we have access to information. Monte Carlo simulation (also known as the Monte Carlo Method) is a good procedure that can be used to predict lots of alternatives of possible outcomes of any one decision made and evaluates the impact of risk, therefore allowing a better choice of decisions. Monte Carlo simulation is characterized by a technique that involves computerized mathematics as means of accounting for risk in decision making. This method is being used in wide areas such as finance, project management, energy, manufacturing, engineering, research and development, insurance, oil and gas, transportation, and the environment. It provides different and various range of possible outcomes and probabilities that may occur for every choice made. The choice of the decision-maker can be in any range of the risk taken from very risky or no risk at all. Monte Carlo allows the investigator to substitute a range of values representing all possible outcomes in a building model. The method calculates over and over the result until it provides a set of different values representing possibilities that can be plotted into a probability function. These probability functions are more realistic in showing the importance and certainty of the risk taken. The Monte Carlo can process thousands of recalculations that result in all possible complete and final outcomes. The most commonly used probability distribution functions are:

- Normal distribution: The user simply defines the mean or expected value and a standard deviation to describe the variation from the mean. Values in the middle near the mean are most likely to occur.
- Uniform distribution: All values have an equal chance of occurring, and the user simply defines the minimum and maximum.
- Triangular distribution: The user defines the minimum, most likely, and maximum values.

#### 4. CRASHING-RISK-MODELING SOFTWARE (CRMS) METHODOLOGY

The developed software provides the lowest crash cost of the activity in order to crash it. Hence the normal duration of the activity should be always greater than the crash duration for it to be crashed. Briefly a normal distribution graph shows all the possible outcomes of the activities according to the mean and the standard deviation of the project. The mean represents the normal duration of the activity. Implementing the risk in CRMS is important because the user has a choice in selecting the level of the risk between 0% and 100% in the project that's being analyzed. The procedural steps that the program uses are: Monte Carlo simulation, finding the critical path, crashing the lowest cost, finding the normal distribution for risk. The steps are described in the flow chart of Figure 1.



**Figure 1.** Flow Chart for CRMS Methodology

#### Input

The input for CRMS is the schedule network. First the activity name – or I.D. - (e.g. “A”), direct successors, (e.g. “B, C, D”) is inputted. Then, the activity normal duration following a triangular distribution is inputted. Hence, a ‘most likely’, an ‘optimistic’ and a ‘pessimistic’ normal duration is inputted. The same is repeated for the crashed duration, the normal cost and the crashed cost. A snapshot of the CRMS input is illustrated in Figure 2.

## Simulation

Following the input, Monte Carlo simulation is used in order to generate stochastic crashed schedule. However, the CRMS evaluates a stochastic slope for each activity, using Variates of the input randomly, as per Equation 1.

$$\text{Var}(\text{Slope}) = \frac{\text{Var}(\text{Crashed Cost}) - \text{Var}(\text{Normal Cost})}{\text{Var}(\text{Normal Duration}) - \text{Var}(\text{Crashed Duration})} \quad [1]$$

Where:  $\text{Var}()$  = Variates of the cumulative distribution function of the input

Monte Carlo repeats the same calculation, using Equation 1, using 1 million iteration. The software checks every path of activities until it determines the longest path. Although it is not simple to save all paths including all critical paths, the CRMS simplifies the process by computing all critical paths and determining which one occur the most. Next, CRMS finds the second longest path in order to crash the project duration. Figure 3 shows a snapshot of this result. Figure 3 points that the critical path A-C-H occurred 100,000 times.

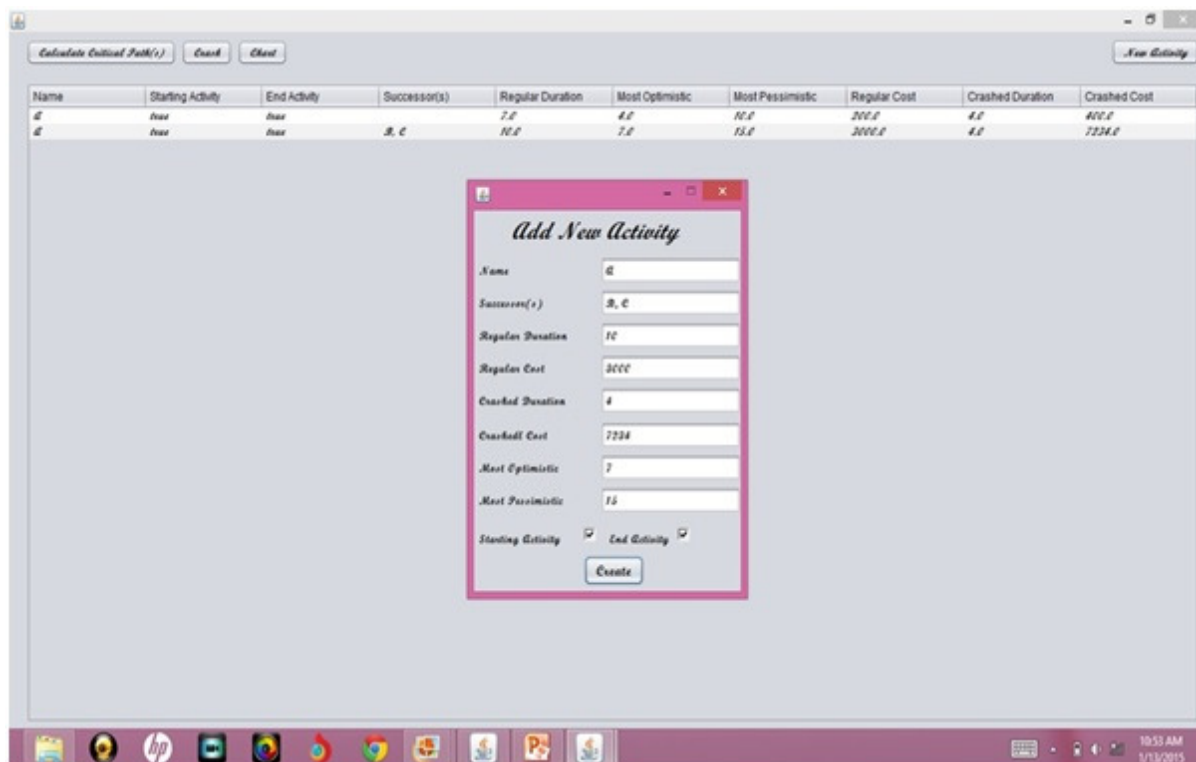


Figure 2. CRMS Input Snapshot

The screenshot shows a software interface with a table of activity data and a dialog box. The table has columns for Name, Starting Activity, End Activity, Successor(s), Regular Duration, Most Optimistic, Most Pessimistic, Regular Cost, Crashed Duration, and Crashed Cost. The dialog box, titled 'Critical Paths', contains the message 'A > C > H > J occurred 100000 times' and an 'OK' button.

| Name | Starting Activity | End Activity | Successor(s) | Regular Duration | Most Optimistic | Most Pessimistic | Regular Cost | Crashed Duration | Crashed Cost |
|------|-------------------|--------------|--------------|------------------|-----------------|------------------|--------------|------------------|--------------|
| A    | true              | false        | C            | 2.0              | 1.0             | 3.0              | 5000.0       | 1.0              | 10000.0      |
| A    | true              | false        | B            | 2.0              | 1.0             | 3.0              | 5000.0       | 1.0              | 10000.0      |
| B    | true              | false        | C            | 4.0              | 3.0             | 6.0              | 6000.0       | 2.0              | 12000.0      |
| B    | true              | false        | F            | 4.0              | 3.0             | 6.0              | 6000.0       | 2.0              | 12000.0      |
| C    | false             | false        | G            | 3.0              | 2.0             | 7.0              | 2025.0       | 2.0              | 6000.0       |
| C    | false             | false        | H            | 3.0              | 2.0             | 7.0              | 2025.0       | 2.0              | 6000.0       |
| D    | false             | false        | J            | 6.0              | 5.0             | 8.0              | 1704.0       | 4.0              | 4000.0       |
| E    | false             | false        | J            | 3.0              | 4.0             | 8.0              | 4000.0       | 2.0              | 8000.0       |
| F    | false             | true         |              | 18.0             | 12.0            | 18.0             | 6628.0       | 10.0             | 14000.0      |
| G    | false             | false        | J            | 6.0              | 3.0             | 7.0              | 176.0        | 4.0              | 1600.0       |
| H    | false             | false        | J            | 4.0              | 3.0             | 7.0              | 184.0        | 2.0              | 2400.0       |
| I    | false             | true         |              | 1.0              | 1.0             | 7.0              | 6655.0       | 2.0              | 14000.0      |
| J    | false             | true         |              | 18.0             | 12.0            | 18.0             | 1942.0       | 2.0              | 24000.0      |

Figure 3. CRMS Most Occurred Critical Path Snapshot

### Crashing the Lowest Cost

At this stage, CRMS crashes all the latter critical paths on a one million iterations. After that, CRMS compares the normal duration of the critical path to the 2<sup>nd</sup> longest path and obtains a limiting duration value for crashing. Hence, the user is reminded that the normal duration should be larger than the crashed duration. Then, if possible, it will crash the lowest cost until it reaches its crash limit, and if the activity it is crashing exists in the 2<sup>nd</sup> longest path, it subtracts each day it can crash from the total number of days of both the critical path and the 2<sup>nd</sup> longest path. If the difference between the crashed duration and normal duration is 0, it skips to the second least cost. Till now CRMS has computed the least crash cost and not every crash cost, so then it saves the largest crash cost in order to plot the result as a normal distribution curve. Figure 4 shows a snapshot of the least crashed cost result.

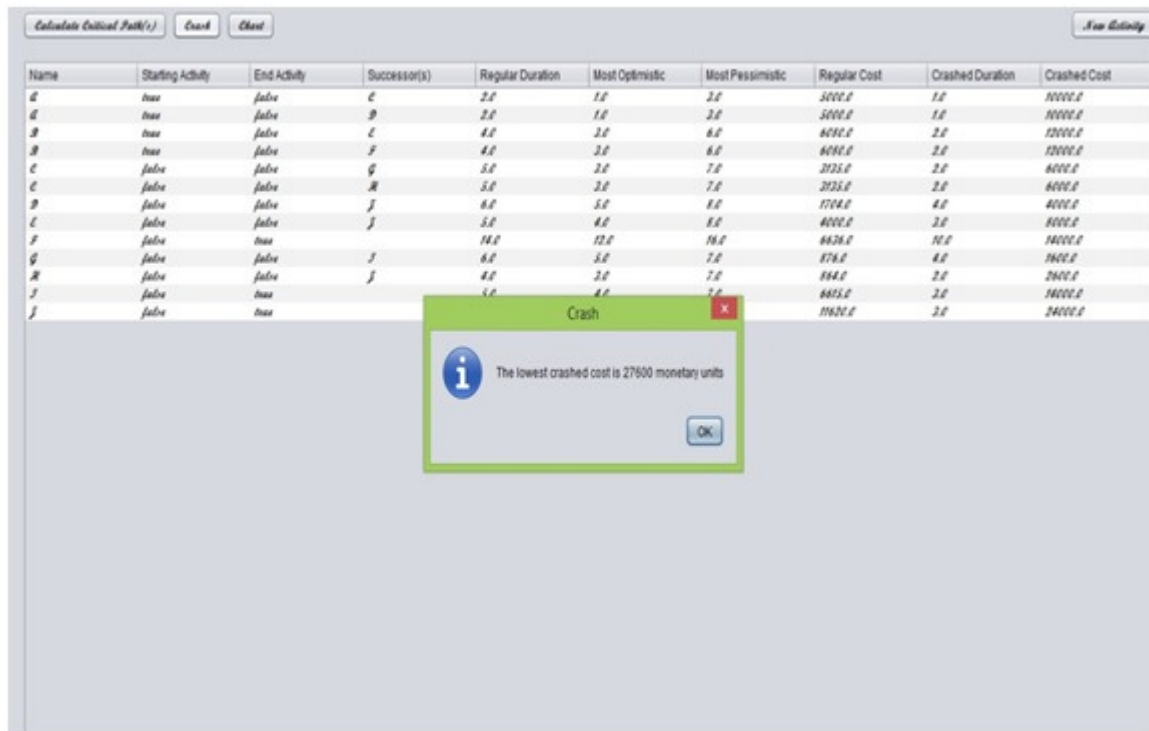


Figure 4. CRMS Least Crashed Cost Snapshot

**Normal Distribution of Result**

Hence, the values that CRMS outputs are very huge and needs computers with enormous memories only accessible to large construction firms. This problem was solved by making CRMS plot half the data that represents having no risk (0% risk level) equivalent to incurring the largest maximum construction cost, then plots the data that represents having a 50% risk level, equivalent to incurring the most probable cost. Figure 5 show the stochastic result.

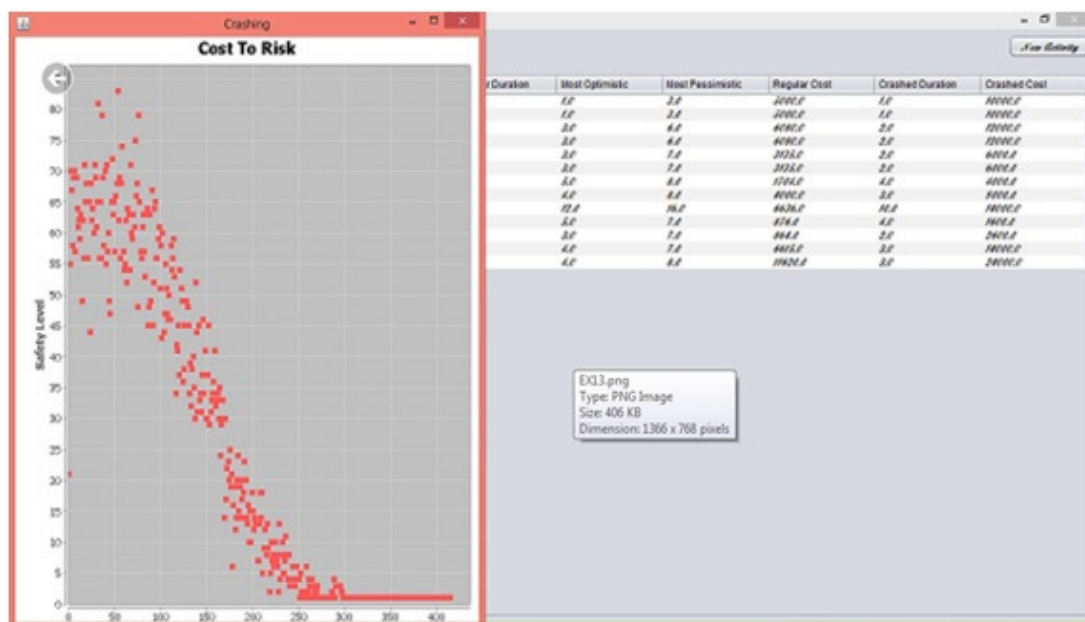


Figure 5. CRMS Result Snapshot

## 5. CONCLUSION

This article develops and overviews a model for crashing schedule considering uncertainty and risk, the CRMS. The uncertainty sources in the risk factors are many, and to figure ways of estimating them is very difficult. The software illustrates several possible scenarios of risk where the user can try to better plan and foresee the possible future problems and the cost increase that may have to be accounted for. As part of any software development, improvements and additions to the program are possible according to different needs from the construction industry.

## REFERENCES

- [1] Attwood E. (2012), “*Gulf Mega Projects Facing Overruns and Delays Survey*”, Retrieved May 20, 2013, from <http://www.arabianbusiness.com/gulf-mega-projects-facing-overruns-delays-pwc-survey-482670.html>
- [2] Brunnhoeffler, G.C. (2010). *Crashing the Schedule – An Algorithmic Approach with Caveats and comments*. Retrieved February 9, 2012, From <http://ascpro0.ascweb.org/archives/cd/2010/paper/CEUE137002010.pdf>
- [3] Georges N., Semaan N., Rizk J. (2014), “*CRASH: An Automated Tool For Schedule Crashing*”, International Journal Of Science, Environment, and Technology, Vol. 3, No. 2, pp. 374 – 394.
- [4] Celik, B., Celik, S., & Brunnhoeffler, G.C. (2011), “*Toward a teaching software application for crashing the schedule*”, SPE<sup>TM</sup> Beta v.1. Symposium conducted at the 47<sup>th</sup> ASC Annual International Conference by the Associated Schools of Construction, Omaha Nebraska, USA.
- [5] Li K., Shao B., and Zelbst P., (2012), “*Project Crashing Using Excel Solver: A Simple AON Network Approach*” International Journal of Management & Information Systems, Volume 16, Number 2, p. 177.
- [6] Lima M.B.F., Silva L.B. & Vieira R.J., (2006), “*Project crashing and costs laws in the knowledge age*”, Symposium conducted at the Third International Conference on Production Research – Americas’ Region (ICPR-AM06), Curitiba, Parana, Brazil.