Proceedings of the 2019 Winter Simulation Conference
N. Mustafee, K.-H.G. Bae, S. Lazarova-Molnar, M. Rabe, C. Szabo, P. Haas, and Y.-J. Son, eds.

USING COMBINED DISCRETE-EVENT/ CONTINUOUS MODELING FOR SIMULATION OF SAFETY INCIDENTS ON PROJECT NETWORKS

Rana Ead
Hady Elkholosy
Amira Eltahan
Stephen Hague
Simaan AbouRizk

Hole School of Construction Engineering
University of Alberta
4-110 Markin/CNRL Natural Resources Engineering Facility
Edmonton, AB T6G 2W2, CANADA

ABSTRACT

Safety incidents are an expected part of construction; their occurrence leads to cost and schedule overruns, and sometimes severe worker injury. Incidents can be reduced through mitigation in the planning phase. To reduce safety incidents, proper quantification of risk impact is necessary throughout the project. Incorporating continuously occurring safety risks within a project network schedule represented by discrete activities remains challenging. This paper proposes a methodology for building a combined discrete-event continuous simulation model to predict safety-incident related schedule delays. An algorithm uses project information stored in a database to automatically construct a Critical Path Method (CPM) network in *Simphony.NET*. The model simulates daily risk occurrence using a discrete-event process to model reduced project productivity. An illustrative example is explained, and results are compared with a base case CPM network. Activities having a higher risk of incident occurrence are recorded to help practitioners develop proper mitigation strategies during project planning.

1 INTRODUCTION

The construction industry is generally associated with a high risk of injuries and fatalities. The danger inherent to the industry is a result of the dynamic work environment, the presence of heavy equipment on the work-site, and workers performing hazardous operations (Hallowell and Gambatese 2009). According to the US Bureau of Labor Statistics (2017), 79,810 workers were injured and another 971 suffered fatalities in the construction private sector. In addition to the human physical harm that results from these accidents, the economy suffers a massive impact as insurance companies have to pay out accident costs. Consequently, contractors pay indirect expenses in terms of products, materials, and legal costs (Sawacha et al. 1999). Lost-time accidents and injuries lead to reduced productivity, which in turn leads to schedule delays and further cost overruns. Occupational accidents can have a negative impact on the workers' families, contractors' reputations and, depending on the project, society (Garcia-Herrero et al. 2012).

These accidents often occur due to a lack of experienced labor, or insufficient training and supervision carried out by contractors (Sawacha et al. 1999). Choudhry and Fang (2008) mention that accidents are caused by a combination of factors, categorized under unsafe work environment and unsafe work operations; approximately 80% of these accidents are due to human behavior. In practice, project safety is handled reactively. Schedule, cost, and quality control, on the other hand, are managed proactively using integrated computerized systems with proper planning and controlling (Kartam 1997). When safety risks are improperly assessed, and safety precautions poorly planned, workers are exposed to

978-1-7281-3283-9/19/$31.00 ©2019 IEEE

hazardous environments that could reduce their productivity, disrupt their operations, and increase the likelihood of accidents occurring (Kyoo-Jin and Langford 2006). Contractors would benefit from the ability to quantify the impacts of safety incidents and determine higher risk activities. This would allow them to minimize schedule and cost overruns by developing mitigation strategies through advance planning.

2 LITERATURE REVIEW

Various research studies have investigated the effects of risks on project outcomes, such as cost and schedule. Wales and AbouRizk (1996) combined discrete-event modeling of a CPM network with the continuous-event simulation of weather changes and the corresponding effect on the activity productivity; the simulation was completed through a trained neural network fed from historical data. The research tried to overcome the inherent limitation of a CPM network. This limitation is due to the various modeling assumptions attributed to the CPM, such as deterministic branching and fixed (predetermined) activity duration. These limitations alter the function of a CPM in simulating a real project where failure to deliver activities as per the scheduled duration may occur. Other researchers have tried to overcome this limitation of the CPM through running a Monte Carlo simulation (Woorly and Crandall 1983). In this instance, the distribution of factors affecting the activity durations are sampled in addition to sampling the activity duration under optimal conditions. The effect of the factors is combined to the sampled activity duration in order to determine the overall activity duration; this process is applied to each activity separately to obtain the total project duration. The technique implemented by Wales and AbouRizk, however, is more accurate because time is incremented into equal steps, and the weather conditions are resampled in every time-increment to simulate real-life conditions.

Several research studies have focused specifically on safety hazards' effects on project outcomes, and proper safety planning techniques to minimize the number and rate of accidents. Wang et al. (2006) developed a simulation-based model to assess the impact of safety hazards on activity costs. In their approach, they evaluated the likelihood of hazard occurrence based on historical data. Accordingly, a three-point estimation was implemented to simulate the likelihood distribution. They further assessed the costs associated with each incident cause, and integrated both the likelihood and the associated cost of each activity within the schedule network to evaluate uncertainties. Although their research provides practitioners with useful safety information, the implementation of their model is time-consuming, and not user-friendly. Furthermore, they relied on a discrete-event CPM, which fails to account for the continuous nature of risks.

Using a similar approach to integrate safety risks within a project's CPM network, Kartam (1997) proposed a safety total-quality management system. The research incorporates the safety plan within the CPM network as a means of communication and assigning clear safety tasks during the planning phase of the project, which would consequently reduce the occurrence of safety incidents. He developed a database that captures specific project data, which then outputs recommendations. Moreover, his model integrates the user-captured data into the CPM network to assign and correlate safety and health information during the project-planning phase. His research, however, has failed to address or simulate the degree to which hazards affect each activity. While this research provides practitioners with useful data, discrete event processes fail to integrate continuously occurring project risks within the project network.

In order to bridge the existing gap in literature and in practice, this paper presents a combined discrete-event, continuous simulation model capable of simulating the occurrence of safety risks continuously over the project duration. This simulation is achieved through a computer-coded algorithm that reads from a database the project-data regarding activities, durations, relationships, and safety hazards. The program then automatically generates a model simulating the project's CPM network in *Simphony.NET*. As the project advances on a daily basis, the risks associated with the in-progress activities/work packages are modeled through a discrete-event process. When safety risks occur, the productivity of the associated activity is reduced based on the impact the safety incident has on the activity and, consequently, the total project duration time increases. In this manner, the uncertainties associated with each risk are continuously accounted for over the progress of the activity. This approach improves upon the limitations of a discrete-event model, which can only simulate risk occurrence at the start of an activity. Accordingly, this approach provides practitioners with a user-friendly tool to deliver crucial information about high-risk work packages. This information can be effectively used during the planning phase of the project, resulting in schedule and cost savings and overall safety improvement on the jobsite.

3 METHODOLOGY

Generally, simulation tools are effective in modeling different construction operations by mimicking real-time events and providing project managers with both statistical analyses and knowledge of project scheduling, costs, and risks. Combined continuous-discrete event simulation models are useful in modeling risks such as safety incidents, weather conditions, change orders, and other uncertainties that occur during project execution (Wales and AbouRizk 1996). The proposed approach involves the use of a combined continuous-discrete event simulation model to incorporate safety hazards within a project network schedule. The Critical Path Method (CPM) network is a continuous model, and is integrated with a discrete-event process to mimic the occurrence of risks as the project progresses while quantifying their impact on the total project duration. The flowchart presented in Figure 1 provides an explanation of the overall methodology proposed in this paper. A database is constructed containing historical data regarding the likelihood of occurrence of incidents, as well as the project schedule information including tasks, durations and associated safety risks with their impacts. A coded algorithm reads this information from the database and automatically builds the corresponding CPM network for the project in *Simphony.Net*. Incident occurrence is simulated on a daily basis, as the project runs. If an incident occurs, the productivity of the corresponding task is reduced and the project is delayed accordingly.

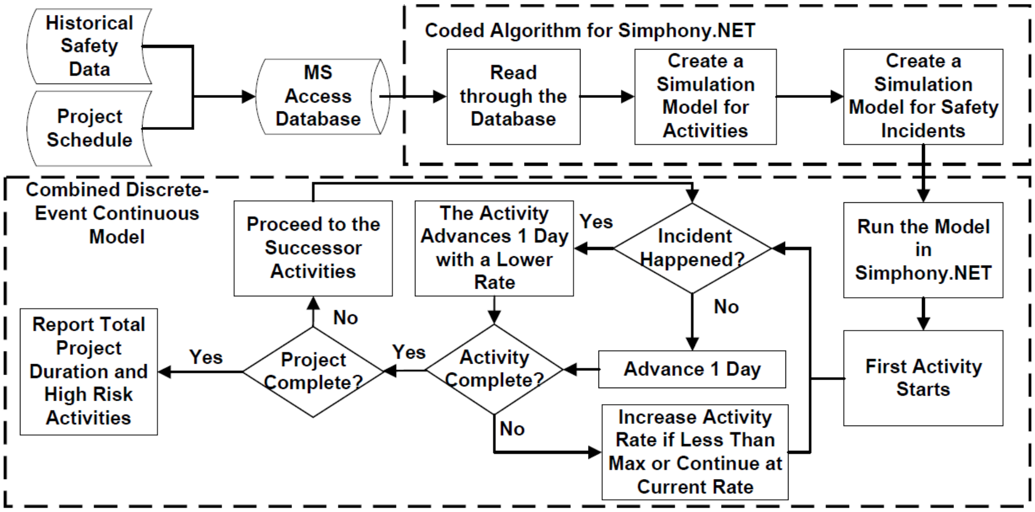

Figure 1: Methodology Flowchart.

3.1 Input Modeling

This model is constructed from information provided in the database, which itself is constructed based on the database schema provided in Figure 2. The user first creates a table for "activities" or "work packages" (depending on the level of detail required) called "WORKPACKAGES." The user then provides data such as the work package ID, name, and productivity rates.

The next table created is called "RELATIONSHIPS," which includes the relationships between work packages or activities and has two fields (the work package ID and predecessor ID) obtained as a foreign keys from the "WORKPACKAGES" table. It is important to note that each relationship must be a Finish-to-Start relationship and must be entered in a separate row. Consequently, if one activity has two predecessors, it will appear twice in the work package ID field.

The third table the user creates, "RISK," encompasses safety hazards associated with each work package. These risks are categorized by trade and event type for each work package, which means that a single work-package can have a number of risks associated with it. This paper does not discuss the method by which the fatal and non-fatal impacts are found; however, the illustrative example explained below suggests how they can be obtained.

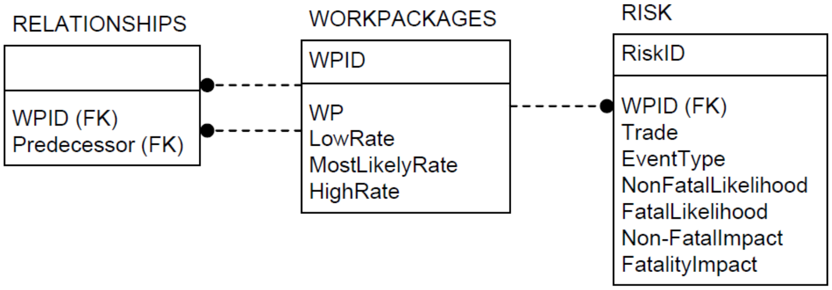

Figure 2: Database Schema.

3.2 *Simphony.NET*

Simphony.NET represents an evolution in computer simulation and its integration into the construction industry. It consists of a foundation library, as well as specialized computer programs that allow for the development of new construction simulation tools in an efficient manner." (AbouRizk et al. 2016). In *Simphony.NET*, modeling elements define the behavior of the model and have properties to control their behavior. Entities flow through and between these modeling elements and have attributes to define their state. Arrows in *Simphony.NET* are responsible for directing the flow of entities Figure 3 shows the design of the modeling elements that are used throughout the paper.

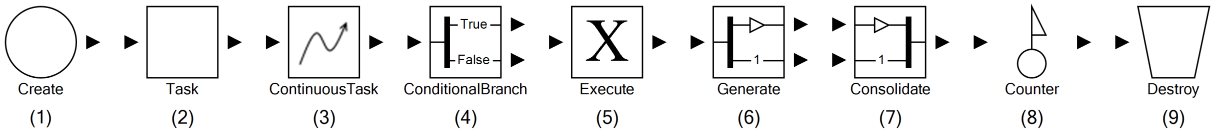

Figure 3: *Simphony.NET* modeling elements.

The following list provides a brief explanation of these entities:

1. The *create element* introduces entities to the model.
2. The *task element* delays an entity by a specified duration, which can be deterministic, stochastic, or user-written code.
3. The *continuous task element* delays an entity for a certain duration determined by a specified progress rate. The rate can change continuously as the simulation runs.
4. The *conditional branch* allows the user to input a certain condition to re-route the entity depending on whether the condition is fulfilled or not.
5. The *execute element* runs a user-written code whenever an entity passes through it.
6. The *generate element* splits one stream of entities into two distinct streams.
7. The *consolidate element* merges two streams of entities into a single stream.
8. The *counter element* counts the entities passing through. It is mainly used to calculate production in a system but can also be used to terminate the model by specifying a count limit.

9. The *destroy element* is used to get rid of elements once they finish the required cycle.

It should be noted that the above explanation is very brief and is intended to familiarize the reader with the modeling elements utilized; however, many functions, properties and elements of *Simphony.NET* are not discussed.

3.3 Algorithm

A code was developed to read off the database (as explained above). The code executes several queries which allow it to do the following:

1. Read the list and create a *continuous task element* for each of the activities/work packages.
2. Create an *execute element* prior to the *task element*, the purpose of which will be explained in detail later in this paper.
3. Create *generate elements* for any work packages that have more than one successor.
4. Create *consolidate elements* for any work packages that have more than one predecessor.
5. Add relationships between all the elements to construct the CPM network.
6. Create a *.sim* file with the generated model.

The purpose of the code is to develop a generic model that can be used with any project schedule. A database would be constructed based on the schema explained above, and the CPM network would be constructed to run the simulation in correspondence to this schedule.

3.4 Project Network Model

The Activity-on-Node (AON) diagramming technique has been used to represent the project network in *Simphony.NET*. Figure 4 compares a typical AON diagram with a Simphony.NET AON model. In the typical model, activities are represented by nodes, and the arrows show the precedence relationships between these activities. In *Simphony.NET*, activities are represented by continuous task elements. The *create element* is used to fire an entity into the model to start the simulation. Activities that have more than one successor require a *generate element* to clone the entity and allow each clone to flow through one path to each successor. Activities with more than one predecessor require a *consolidate element* to prevent the activity from starting until all predecessor activities have been completed. The counter at the end of the network keeps track of the time at which the last entity passes through, indicating the total project duration. Finally, the destroy element is used to end the simulation. The arrows between the modeling elements also represent the precedence.

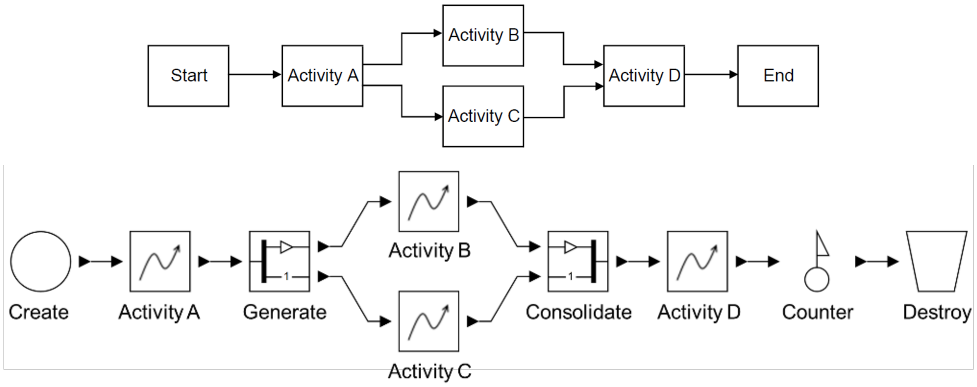

Figure 4: Typical AON diagram (top) vs *Simphony.NET* AON diagram (bottom).

Two CPM networks are constructed by the code, one representing the "base case" where the conditions are ideal (i.e. assuming no safety incidents occur) referred to herein as the "*base case model.*" The second model shows the effect of introducing safety hazards and risks on the project duration, and is called the "*risk model.*"

3.5 Global Variable Initialization

Global variables defined in *Simphony.NET* are a collection of attributes shared among all of the entities. Any change made by one entity is visible to others. The following are the three main types of global variables initialized in this model:

1. *Initial Rate Global Variables:* This type of global variable is created for each work package. The rate of the work package is sampled using a Pert distribution. The parameters (optimistic rate, pessimistic rate, and most likely rate) are pre-defined in the database. The value of the sampling is then stored in the global variable, and is referred to by the corresponding work package present in both the base case and the risk model. The index of each of these global variables equals the negative value of the work package ID. For a concreting work package having an ID of 2, the corresponding global variable storing the sampled rate of this work package is GX(-2).
2. *Multiplier Global Variables:* This type of global variable is also created for each work package. Its value is initialized to 1 and multiplied by the rate for each work package. When an incident happens, its impact is stored in the global variable to simultaneously decrease the rate of the activity and increase the total project duration. The index for these global variables is the work package ID; therefore, the multiplier global variable for the same work package is GX(2).
3. *Progress Rate Global Variable:* The current progress rate of the work package is calculated and stored in this global variable. Initially, the value of this global variable is equal to the sampled value of the rate global variable for the same work package. The index of this global variable is equal to 1000 + the ID of the work package; therefore, the progress rate global variable for the concreting work package will be GX(1002).

Although it would have been better to represent the indices of the global variables using a three-dimensional array, one of the limitations of *Simphony.NET* is that it only supports using one-dimensional arrays for global and local variable indices. Accordingly, the indices were selected in the manner explained above.

3.6 Execute Element Expression

Each *task element* built in *Simphony.NET* is associated with an *execute element* and represents an activity or work package based on the definition in the database. The *execute element* first sets the values of the entity's two local attributes (LX(1) and LX(2)) flowing through the CPM network to the values of the global variable indexes for the corresponding work package that it precedes.

Since each safety incident for each work package is represented by an entity, the execute element first creates these entities and then assigns local variables to them with the definitions shown in Table 1. These entities are passed on to the risk sub-model to simulate risk occurrence.

Table 1: Local variable definitions for risk entities.

Local Variable	***Definition***	***Local Variable***	***Definition***
LX(1)	Incident likelihood	LN(1)	Index of rate global variable
LX(2)	Fatality likelihood	LS(1)	Name of incident
LX(3)	Incident impact	LS(2)	Occupation type
LX(4)	Fatality impact	LS(3)	Work package name

3.7 Risk Sub-Model

As explained in the previous section, each entity representing a safety hazard is fired by the execute elements into the risk sub-model shown in Figure 5. The conditional branch element checks whether an incident has happened or not by sampling a random deviate uniformly distributed from 0-1. If the sampled value is less than or equal to the likelihood of the incident happening (a local attribute of the entity) the formula returns "true"; otherwise, it returns "false." Another conditional branch then checks whether or not the incident resulted in a fatality by sampling another random deviate and comparing it to the fatality likelihood (another local attribute of the entity). Depending on whether or not the incident resulted in a fatality, the work package's multiplier global variable, to which the entity belongs, is equated to the impact value of the incident or fatality. Using the concreting work package as an example, the work package rate is changed by multiplying the current progress rate global variable by the multiplier global variable as shown in Equation (1).

$$GX(1002) = GX(1002) \times GX(2) \tag{1}$$

The entity is then advanced one day using a *discrete task element*. Another *execute element* uses Equation (2) to calculate the rate of the activity so that:

1. If an incident occurs, the rate begins to increase linearly over the coming days or,
2. If no incident takes place, the rate continues at the initial sampled value carried by the rate global variable.

$$GX(1002) = Min((GX(1002) + 1\% \times GX(-2)), GX(-2) \tag{2}$$

Finally, a third conditional branch will check whether or not the activity is complete. This happens by using the *ServersAvailable* formula built into *Simphony.NET*, and checking if the servers available are more than 0 (i.e. the task is complete). If the work package is complete, the entity is destroyed. Once it is destroyed, there are no more safety incidents that can take place for this work package. If the work package is incomplete, the entity goes through the discrete-event sub-model again.

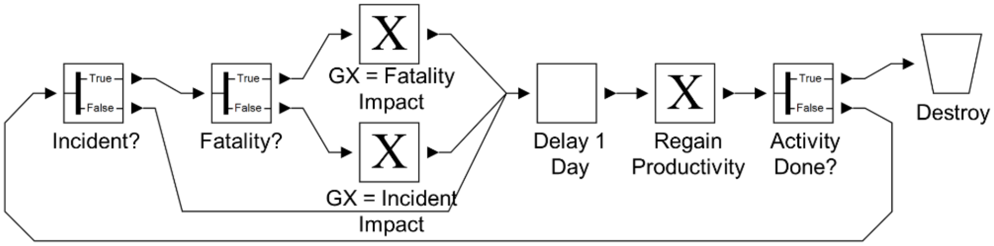

Figure 5: Risk sub-model.

4 ILLUSTRATIVE EXAMPLE

4.1 Data Collection

Data regarding the likelihood of safety hazards or risks were obtained from the US Bureau of Labor Statistics. The Injuries, Illnesses, and Fatalities (IIF) program collects annual data on the number and rate of work-related injuries and fatal injuries. The statistics provided vary by incident, geography, industry, occupation, and others (United States Department of Labor 2017). The data used for the model were organized into two tables:

1. The incident rate per 10,000 workers by occupation (for e.g. brick masons) and events leading to injury or illness (for e.g. falls, slips, and trips).
2. The fatality rate per 100,000 workers by occupation and event.

Using this data, the incident likelihood and fatality likelihood of each risk are calculated as follows:

$$(Incident\ Rate/100{,}000) \times number\ of\ workers \quad (3)$$

$$(Fatality\ Rate/10{,}000) \times number\ of\ workers \quad (4)$$

Table 2 shows the values for incident and fatality impacts. A qualitative scale is assumed for ease of application and with the help of a former safety manager, translated into the quantitative values shown. It should be noted, however, that these values can be modified or other techniques could be used to model the impact more accurately. Suggesting other techniques is beyond the scope of this paper.

Table 2: Impact score.

Qualitative Impact	Very Low (VL)	Low (L)	Moderate (M)	High (H)	Very High (VH)
Quantitative Impact	0.90	0.80	0.50	0.30	0.00

4.2 Project Description

The main work packages involved in the construction of a 3-storey building were identified. Table 3 shows a summary of these work packages and their optimistic, most likely, and pessimistic rates as pre-defined in the database.

Table 3: Work package definition.

Work Package	***WP ID***	***Predecessor(s)***	***Optimistic Rate***	***Most Likely Rate***	***Pessimistic Rate***
Earthworks	1	-	0.167	0.143	0.100
Concreting	2	1	0.020	0.017	0.014
Masonry	3	2	0.045	0.040	0.036
MEP	4	2	0.038	0.033	0.029
Roofing	5	8	0.125	0.100	0.071
Doors and Windows	6	3	0.333	0.250	0.167
Finishing	7	4, 8	0.025	0.022	0.020
Thermal & Moisture Insulation	8	3	0.077	0.067	0.056

Using the database schema, a table was constructed of the risks associated with each work package. A total of 66 safety incidents were added to the database. Table 4 shows a sample of these risks.

Table 4: Risks (sample table).

Risk ID	***WP ID***	***Trade***	***Event***	***Incident Likelihood***	***Fatality Likelihood***	***Incident Impact***	***Fatality Impact***
19	2	Carpenters	Contact with objects	0.07164	0.0000702	0.5	0
34	4	Electricians	Harmful exposure	0.0015	0.0000576	0.5	0
39	5	Roofers	Falls, slips & trips	0.03447	0.001116	0.5	0
58	7	Painters	Falls, slips & trips	0.0285	0.0001641	0.5	0

4.3 Combined Discrete-Event Continuous Model

Figure 6 shows a portion of the generated CPM network models for the base case and risk scenarios built using the coded algorithm, as discussed above. The risk model integrates safety incidents into the project's network schedule, whereas the base case model assumes ideal conditions to compare the effect of risks on the total project duration. The work packages described in Table 3 are modeled as continuous task elements within *Simphony.NET*.

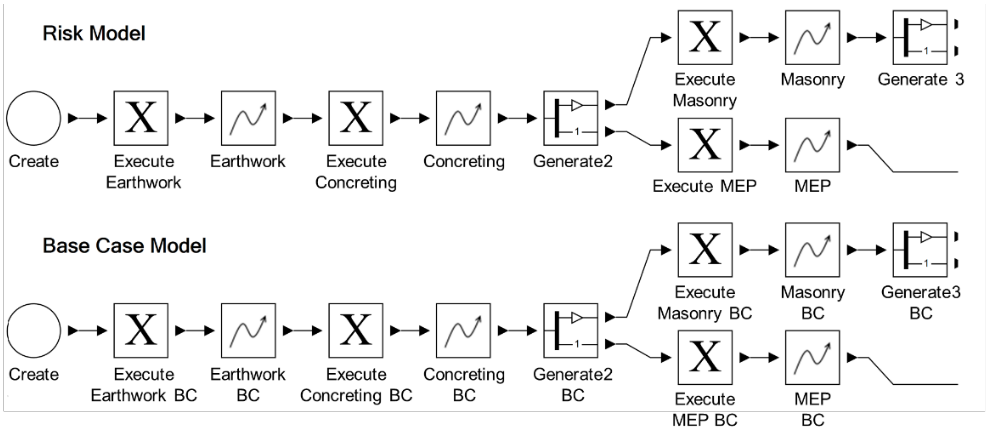

Figure 6: Risk model and base case model CPM networks (only a portion is shown for legibility).

5 OUTPUT ANALYSIS

5.1 Project Duration and Productivity Analysis

The base case and risk model are run 1000 times, sampling each work package's duration from a Pert distribution using the rejection method described in Cheng (1978) to determine the total project duration, and identify when an incident will occur. This will help determine which activities have higher risks and unsafe work environments. Figure 7 compares the histograms for total project duration in the base case and risk models, respectively.

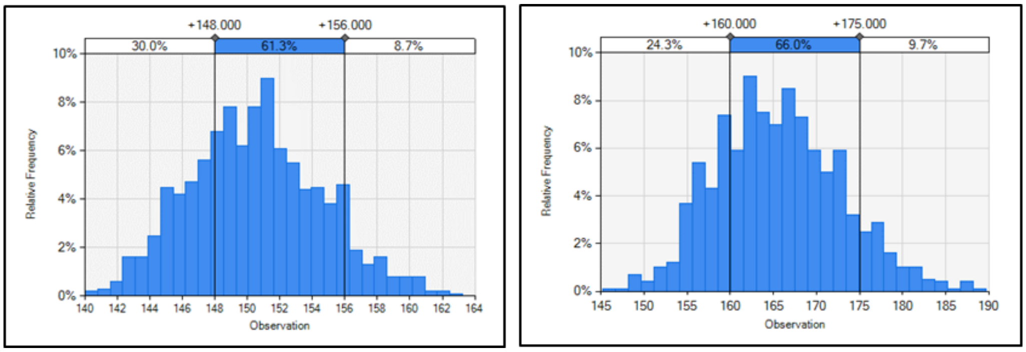

Figure 7: Base case model histogram (left) and risk model histogram (right).

Table 5 indicates that 61.0% of project durations are between 148 and 156 days with a mean of 150.4 days. Based on the cumulative probability, the project takes 153 days or less 70% of the time. The risk model has a mean project duration of 165.6 days; 66% of the simulated project durations are between 160 and 175 days. Finally, 69.5% of the simulation runs give a total project duration less than 169 days.

Table 5: Total project duration for the two cases.

Project Duration	***Base Case Model (days)***	***Risk Model (days)***
Mean	150.4	165.6
Maximum	164	190
70% cumulative probability	153	169

When an incident occurs during the execution of work, the productivity rate of the work package is reduced by multiplying current progress rate by the multiplier global variable. For example, concreting work in the base case model is completed in approximately 55.5 days with a rate of 1.8% per day, as shown in Figure 8. In the risk model, however, multiple incidents occur during concreting work, resulting in a fluctuation of the productivity rates between 0.6% and 1.8% per day. After the occurrence of the incident, the crew starts to regain its original productivity during execution of the remaining works. Concreting work is finished in approximately 65.5 days in the risk model; 10 days longer than in the base case.

Table 6: Work package duration and productivity rate.

Concreting Work Package	***Base Case Model***	***Risk Model***
Productivity rate	1.8%/day	0.6% - 1.8%/day
Duration	55.5 days	65.5 days

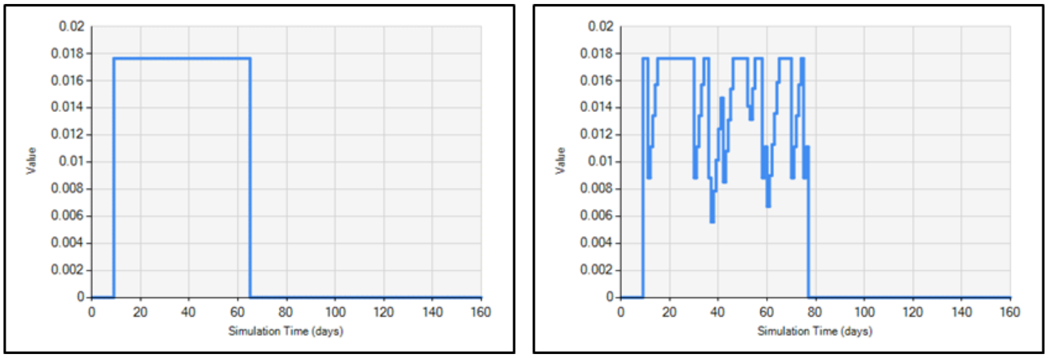

Figure 8: Concreting work package base case model productivity rate *(left)* and risk model productivity rate *(right)*.

5.2 High-Risk Work Packages

For each simulation run, incidents occur at different time periods based on the likelihood of the hazard. While the model advances daily, it identifies the incidents that occur during the project. Accordingly, the information extracted from the simulation model can be helpful to project managers to determine high-risk work packages and assist them in developing the proper mitigation plans to avoid any delays to the project. Table 7 shows the type of incidents that occur in each work package and the days on which each incident happens. It is clear from the results that concreting poses the highest safety risks to workers; consequently, proper safety plans can be incorporated for this work package during the planning phase.

6 MODEL VALIDATION AND VERIFICATION

The methodology proposed in this paper is based on that proposed by Wales and Abourizk (1996).While their model used combined discrete-event/continuous modeling to address the effect of weather changes on crew productivity (Wales et al. 1996), the model in this paper focuses on the effect safety incidents have on production rates.

Table 7: High-risk work packages.

Work Package	*Incident*	*Day*	*Work Package*	*Incident*	*Day*
Concreting	Falls, slips & trips	10	Concreting	Contact with objects	59
Concreting	Falls, slips & trips	29	Concreting	Falls slips & trips	69
Concreting	Falls, slips & trips	35	Concreting	Contact with objects	74
Concreting	Falls, slips & trips	36	Masonry	Contact with objects	94
Concreting	Falls, slips & trips	41	Thermal & Moisture Protection	Contact with objects	102
Concreting	Falls, slips & trips	51	Doors & Windows	Contact with objects	104
Concreting	Falls, slips & trips	52	Thermal & Moisture Protection	Falls, slips & trips	105
Concreting	Contact with objects	57	Finishing	Contact with objects	119

Model validation took place using several techniques based on the abovementioned illustrative example. Initially, the model was verified by comparing the risk model output with other verified model outputs (Sargent 2007). The same project network discussed above was constructed on MS Project. MS Project, however, requires that durations be deterministic to calculate the total project duration using the CPM. Accordingly, the most likely rates presented in Table 3 were used in both models as constant durations. The total project duration output of 150 days matched the output obtained using the MS Project model.

Sensitivity analysis, another technique described by Sargent (2007), was applied to check the validity of the risk sub-model. The risk impact values were altered to make them more intense, and the resulting total project duration increased significantly as shown in Table 9.

Table 8: Sensitivity Analysis Results.

Total Project Duration	*Sensitivity Test (days)*	*Risk Model (days)*
Mean	180.5	165.6
Maximum	476	190
70% confidence	190.8	169

7 CONCLUSION AND RECOMMENDATIONS

Overall, this paper provides construction practitioners with an effective simulation tool that allows them to proactively manage safety incidents. This will improve the overall safety culture on construction sites, increase productivity, and ultimately reduce schedule delays and indirect cost overruns. The main advantages of this model are that it can be easily integrated with other scheduling software, and that it can be modified based on user input for different types of projects.

One of the model limitations is that the relationships between the work packages (or activities) in the project schedule must be finish-to-start (FS) relationships, which may not be realistic because most construction work overlaps throughout the project. To overcome this problem, other scheduling techniques such as the Precedence Diagraming Method (PDM) and Beeline Diagramming Method (BDM) can be used. PDM and BDM can represent overlapping relationships between activities and provide proper schedule computation methods (Kim 2012).

Another important factor to consider, which has not been addressed in this paper, is the effect of two or more safety incidents occurring on the same day, and on the same work package or activity. The current model does not assume that their impacts should be summed; however, the equation used to calculate the rate does not explicitly account for this case. Furthermore, the incremental increase in productivity after an incident is assumed to be linear, which may not be realistic.

ACKNOWLEDGMENTS

The research work is funded by a Collaborative Research and Development (CRD) Grant (CRDPJ 492657) from the National Science and Engineering Research Council (NSERC). The authors would like to thank Elisia Snyder for her assistance with manuscript editing.

REFERENCES

AbouRizk, S. M., S. A. Hague, and R. Ekyalimpa. 2016. *Construction Simulation: An Introduction using SIMPHONY.*1st ed. Edmonton, Alberta: University of Alberta.

Cheng, R.C.H.1978. "Generating Beta Variates with Nonintegral Shape Parameters". *Communications of the ACM.* 21(4):317-322.

Choudhry, R. M. and D. Fang. 2008. "Why Operatives Engage in Unsafe Work Behavior: Investigating Factors on Construction Sites". *Safety Science* 46(4):566-584.

García-Herrero, S., M.A. Mariscal, J.García-Rodríguez, and D. O. Ritzel. 2012. "Working Conditions, Psychological/Physical Symptoms and Occupational Accidents: Bayesian Network Models". *Safety Science* 50(9):1760–1774.

Hallowell, M. R. and J. A. Gambatese. 2009. "Construction Safety Risk Mitigation". *Journal of Construction Engineering Management* 135(12):1316-1323.

Kartam, N. A. 1997. "Integrating Safety and Health Performance into Construction CPM". *Journal of Construction Engineering and Management* 123(2):121-126.

Kim, S. 2012. "CPM Schedule Summarizing Function of the Beeline Diagramming Method". *Journal of Asian Architecture and Building Engineering* 11(2):367-374.

Yi, K. and D. Langford. 2006. "Scheduling-Based Risk Estimation and Safety Planning for Construction Projects". *Journal of Construction Engineering and Management* 132(6): 626-635.

Sargent, R. G. 2007. "Verification and Validation of Simulation Models". In *Proceedings of the 2007 Winter Simulation Conference,* edited by S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, 124-137 Dec 9th-12th Washington, DC: Institute of Electrical and Electronics Engineers, Inc. .

Sawacha, E., S. Naoum, and D. Fong. 1999. "Factors Affecting Safety Performance on Construction Sites". *International Journal of Project Management* 17(5):309-315.

AbouRizk, S. and R. J. Wales. 1997. "Combined Discrete-Event/Continuous Simulation for Project Planning". *Journal of Construction Engineering and Management* 123(1): 11-20.

United States Department of Labor. 2017. *Injuries, Illnesses, and Fatalities*. https://www.bls.gov/iif/.

Wales, R. J. and S. M. AbouRizk. 1996. "An Integrated Simulation Model for Construction". *Simulation Practice and Theory* 3(6):401-420.

Wang, W. C., J. J. Liu, and S. C. Chou. 2006. "Simulation-Based Safety Evaluation Model Integrated with Network Schedule". *Automation in Construction* 15(3):341-354.

Woorly, J. and K. Crandall. 1983. "Stochastic Network Model for Planning Scheduling". *Journal of Construction Engineering and Management* 109(3):342-354.

AUTHOR BIOGRAPHIES

RANA EAD is an M.Sc. student at the University of Alberta studying Construction Engineering and Management. She received her B.Sc. in Construction Engineering from the American University in Cairo in 2018. Her email address is rsead@ualberta.ca.

HADY ELKHOLOSY is an M.Sc. student at the University of Alberta studying Construction Engineering and Management. He received his B.Sc. in Construction Engineering from Cairo University in 2016. His email address is elkholos@ualberta.ca.

AMIRA ELTAHAN is an M.Sc. student at the University of Alberta studying Construction Engineering and Management. She received her B.Sc. in Construction Engineering from the American University in Cairo in 2018. Her email address is eltahan@ualberta.ca

STEPHEN HAGUE is the Senior Programmer for the Hole School of Construction, Department of Civil and Environmental Engineering, University of Alberta. His email address is shague@ualberta.ca

SIMAAN ABOURIZK is a Distinguished University Professor and Canada Research Chair in operations simulation at the University of Alberta. His research is focused on developing an improved framework for the planning and control of construction projects through advancements in simulation. His email address is abourizk@ualberta.ca.