ASSESSMENT OF EVOLUTION AND SENSITIVITY IN CONSTRUCTION PROJECTS

Youssef Khoueiry, Graduate student, ytk01@aub.edu.lb
Issam Srour, Assistant Professor, is04@aub.edu.lb
Ali Yassine, Associate Professor, ay11@aub.edu.lb
Faculty of Engineering and Architecture, American University of Beirut, Lebanon, Riad El-Solh, Beirut,

ABSTRACT

Fast-tracking through activity overlapping allows for shortening the total project duration at the expense of an increased likelihood of rework. Activity overlapping allows downstream successor construction activities to start with incomplete information from their upstream predecessor design activities. The nature of information evolution and the strength of dependency between these activities determine the likelihood and amount of potential rework. This paper uses constructs of evolution and sensitivity to characterize the nature and consequences of preliminary (i.e., partial and incomplete) information exchange between upstream and downstream activities, which influence the amount of rework in the downstream activity as a result of a change in upstream information. The paper illustrates the constructs of upstream evolution and downstream sensitivity: what is their influence on the maturity of the information exchanged throughout an overlapped process and how are they assessed in construction projects? The conclusion provides an explanation of the applicability and importance of these constructs in determining the optimal overlapping strategy in fast-track environments.

Keywords: Evolution, sensitivity, overlapping, construction, rework

1. INTRODUCTION

Traditionally, construction projects were executed in a sequential manner, i.e. an activity starts only when its predecessors are fully completed. This method is time consuming and, therefore, cannot meet the obligation of emerging sharp deadlines. Fast-tracking, through overlapping dependent activities, is a viable alternative to the sequential approach. We distinguish between two types of dependencies among various design and construction activities: physical dependency and informational dependency. Physical dependency refers to the case when the downstream successor activity, cannot start until its predecessor upstream activity is entirely completed. A practical example is installing form shutters and pouring concrete. Alternatively, informational dependency refers to the case when the downstream successor activity may start before its predecessor upstream activity is entirely completed. A practical example is starting a construction activity before its related design is completed. As such, overlapping can only be applied on activities with informational dependencies. In this case, the downstream activity starts based on preliminary (i.e., partial and incomplete) information exchanged from the upstream activities, and effectively resulting in overlapping between upstream and downstream activities, thereby reducing the total project execution time (Bogus et al. 2006 and 2009). Nonetheless, starting a downstream activity based on incomplete information introduces the risk of downstream rework should there be a change in upstream information. Future upstream information modifications require rework in the downstream activity to address the changes of the initial information or assumptions. The resulting rework usually consumes resources (e.g. time and money), and is disruptive to the flow of the downstream work.

The constructs of sensitivity, evolution and progress are at the core of the discussion of rework and overlapping upstream and downstream activities. The amount of rework duration and cost are determined by the information exchanged between upstream and downstream teams which are interpreted via the values of upstream evolution and downstream progress and sensitivity. Sensitivity is the difference in percent progress of an activity divided by the perceived progress after a change is introduced due to a change in upstream information (Blacud et al. 2009). Evolution and progress are defined as the rate of generation of upstream (e.g. design) information from the start until the fulfillment of an activity, however, evolution is related to upstream design activities and progress is related to downstream construction activities. The evaluation of sensitivity, evolution and progress yields mathematical expressions for rework cost and duration at each point in time within the overlap period. Hence, these formulations could be used in different types of problems (e.g. optimization type problems) allowing practitioners to make decisions regarding reduction of rework cost and duration for instance or various scheduling decisions.

The goal of this paper is to illustrate the constructs of evolution and sensitivity, and present a method to estimate them in construction projects. This paper, also, discusses the applicability of these concepts in fast-track construction projects. This analysis is particularly useful for contractors, at any of the project's construction life cycle. A clear understanding of the concepts of evolution and sensitivity helps contractors make proper decisions on overlapping pairs of critical activities. Benefits at the level of the whole project are obtained by aggregating similar pairs over the entire schedule.

The rest of the paper is organized as follows. In the next section we define and explain the constructs of evolution and sensitivity. Then, we present a method showing how these constructs can be mathematically formulated and estimated in construction projects. We, then, discuss their usage in optimization models in the purpose of generating an optimal earliest solution for the overlapping problem. We close the paper with a discussion of the importance of evolution and sensitivity and their applicability in optimization models to overlap activities in complex construction projects.

2. BACKGROUND AND LITERATURE REVIEW

In the context of a pair of dependent activities, as work is occurring in the upstream activity, information is evolving and shared with the downstream dependent activity (see Figure 1). This evolving information is associated with a level of uncertainty depending on the time at which it is exchanged (with the downstream activity): whether at an early or late stage of the upstream activity. If the information was exchanged at an early stage, a high level of uncertainty is associated with the exchanged information, which gives rise to a high likelihood of information changes in subsequent future exchanges. However, at later stages, when the upstream activity is almost completed, the exchanged information is more mature and the amount of uncertainty is minimal.

Along these lines, Krishnan et al. (1997) introduced and explained the concept of evolution as the narrowing of an interval value. Design information is related to defining and fixing the value of design parameters. This value is initially unknown, but an interval for the value is known; that is, a lower and an upper bound. Initially, the interval is wide, but as design work is carried out, the information evolves and the value of the design parameter is gradually narrowed until its final value at the time of activity completion. It is assumed that the final value falls within the initial interval estimated at the outset of the design activity. Krishnan refers to this approach of describing information evolution as narrowing and is shown in Figure 1.

Furthermore, evolution refers to the fact that the content of any information generated throughout an activity evolves from zero percent to account for high uncertainty level, to hundred percent to account for zero uncertainty level. Thus, evolution is defined as the maturity level of information throughout the entire upstream process of a specific activity and is quantitatively interpreted as a function rising from zero at the start of an activity to hundred percent at the end of this activity. According to Krishnan et al. (1997) the degree of evolution is a way to measure how close is the un-finalized design to its final value. Similarly, Loch and Terwiesch (1998) defined ''upstream information evolution'' as the continuous design modification process. Bhuiyan et al. (2004) studied the new product development process model under uncertainty conditions. They defined upstream evolution as "the speed with which complete information is created for transmission". Furthermore, Bhuiyan et al. (2004) distinguished between two types of upstream evolution (fast and slow) and two types of downstream sensitivity (High and low). Blacud et al. (2009) applied the concepts of evolution and sensitivity to overlap dependent design activities in construction. They defined evolution as the "rate at which design information is generated from the start of an activity through the completion of the activity". Similar to other studies (e.g. Bhuiyan et al. (2004)) Blacud et al. (2009) distinguished between the ranges of evolution which can be either "fast" or "slow." They defined a fast evolving activity as an activity which "achieves a high level of definition in an early stage of its duration", whereas a slow evolving activity "will not achieve its final solution until the last moments of its duration". Lin et al. (2009) defined the upstream evolution "as the rate at which preliminary upstream information is modified" and they assumed that it follows a nonhomogeneous Poisson process with a variable rate over time.

Downstream sensitivity is the result of the exchange of unfinished upstream information with downstream activity. Due to the uncertainty of the upstream information exchanged in a preliminary form, an increased likelihood of rework is associated with any expected future change to this information. The amount of downstream rework is dependent on downstream sensitivity. Small changes in the upstream design information will largely disrupt the workflow of a highly sensitive activity. On the contrary, an activity with low sensitivity will not require large rework even with a large change in the upstream design Krishnan et al (1997) define sensitivity as the relationship between downstream iterations and the magnitude of changes of the upstream information. Loch and Terwiesch (1998) defined ''downstream sensitivity'' as the impact of a modification on downstream rework. Bhuiyan et al. (2004) expressed downstream sensitivity as "the degree to which work is changed as the result of absorbing transferred information". Blacud et al. (2009) argued that sensitivity for design activities can either be high or low.

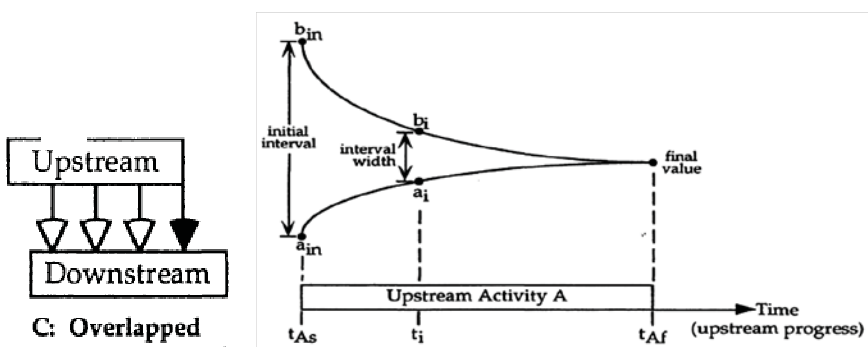

Figure 1: Overlapped process and narrowing concept (adopted from Krishnan et al. 1997)

Lin et al. (2008) discussed the concepts of evolution and sensitivity but did not mathematically evaluate these parameters because they considered that information changes between dependent activities are implanted in the tasks carried out. Roemer and Ahmadi (2004) followed the same methodology used by Krishnan et al. (1997) to build an optimization model for both, crashing and overlapping. However, their work built on the probability and impact of rework without crossing into a mathematical formulation of evolution and sensitivity. Bogus et al. (2005 and 2006) adopted the concepts of evolution and sensitivity to generate a conceptual framework for reducing design durations through overlapping activities based on the patterns of evolution and sensitivity (fast/slow and high/low).

As shown earlier, a remarkable part of the work in the literature defined the concepts of evolution and sensitivity. Some of studies (e.g. Krishnan et al (1997), Lin et al. (2008)) evaluated these parameters and used them in mathematical optimization models, whereas others (e.g. Bogus et al. (2005) and (2006)), only used specific patterns for evolution (fast and slow) and sensitivity (high and low) to develop a basic and an enhanced overlapping strategy framework. More recently, Bogus et al. (2011) examined the relationship between evolution and sensitivity concepts and the probability of rework in different design strategies including overdesign and early release. They, also, presented a simulation model for overlapping sequential activities, through defining the rework probability as a function of the percent of overlap under different scenarios (e.g., fast evolution and low sensitivity). Nonetheless, Bogus et al. (2011)'s study stopped short of presenting an empirical model for illustrating how evolution and sensitivity can be constructed using real world data .

In summary, the literature does not present a comprehensive methodology on how to define and estimate the mathematical expressions of evolution and sensitivity from real projects' data. This paper attempts to do so by focusing on the nature of construction activities and the type of relationships that exists between these activities. The outcome of the proposed methodology is an explicit mathematical expression of evolution and sensitivity for each pair of activities in a project schedule.

3. MATHEMATICAL EVALUATION OF EVOLUTION, PROGRESS AND SENSITIVITY

As mentioned earlier, the goal of this paper is to illustrate the concepts of evolution, progress and sensitivity in fast-track construction projects. The following section provides a methodology to collect and analyze field data in order to derive mathematical expressions for evolution, progress and sensitivity.

3.1 Evolution and progress of activities

Several mathematical functions and patterns can be used to define the path of an evolution or progress function from 0% to 100% (e.g. polynomial functions, exponential functions). In this paper, we focus on only two paths obtained by two exponential functions, which will be explained in subsequent paragraphs. We distinguish between two types of evolution functions, high evolution and low evolution. High evolution refers to an increase in the maturity of the information generated in a decreasing rate and low evolution refers to an increase in the maturity of the information generated in an increasing rate. In other words, an activity with high evolution is an activity where the major part of the relevant information is acquired at early stages, e.g. installation of firefighting pipes where majorly cutting the pipes to real dimensions and installing them comes first and the painting and pressure testing come at the end. Conversely, an activity with low evolution is an activity where the work in early stages requires a lot of data collection or data analysis and coordination; thus, the major part of the work is done at the later stages of the activity, e.g. issuing MEP shop drawings, which requires coordination among all trades before having the final layout approved. Figure 2.a and 2.b show the cases of high and low evolution. Evolution functions can be mathematically modeled through an exponential function which requires different parameters. Those parameters are calculated and estimated from site data and will be explained later in subsequent paragraphs.

To distinguish between upstream and downstream evolution functions, we introduce the progress function. The progress function, F, is an evolution function; however, it is dedicated only for downstream successor activities. Progress functions do not give an indication of the information generated in an activity; however, they are an indication of the amount of work performed for a specific activity until a certain point in time. Progress functions abide by the same rules of evolution functions; and thus, they are classified in two categories as well: low and high progress functions.

To mathematically model evolution and progress, we introduce an exponential generic function, Q, presented in Equation (1) to represent the different types of evolution and progress functions. This increases the flexibility of these constructs by allowing for different combinations of functions (e.g. low and high progress) using only one functional form. One could argue that other mathematical functions such as polynomial or square root functions could be used to solve the same problem. While this might be possible, such functions, once multiplied by each other, will yield complex functions which could be hard to solve with regular software packages such as MATLAB or MATHCAD.

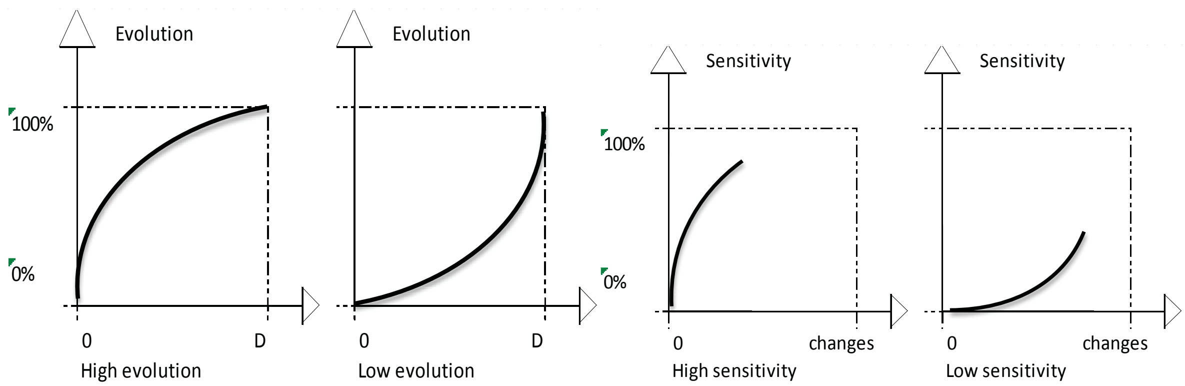

Figure.2: (a) High evolution, (b) Low evolution, (c) High sensitivity, (d) Low sensitivity

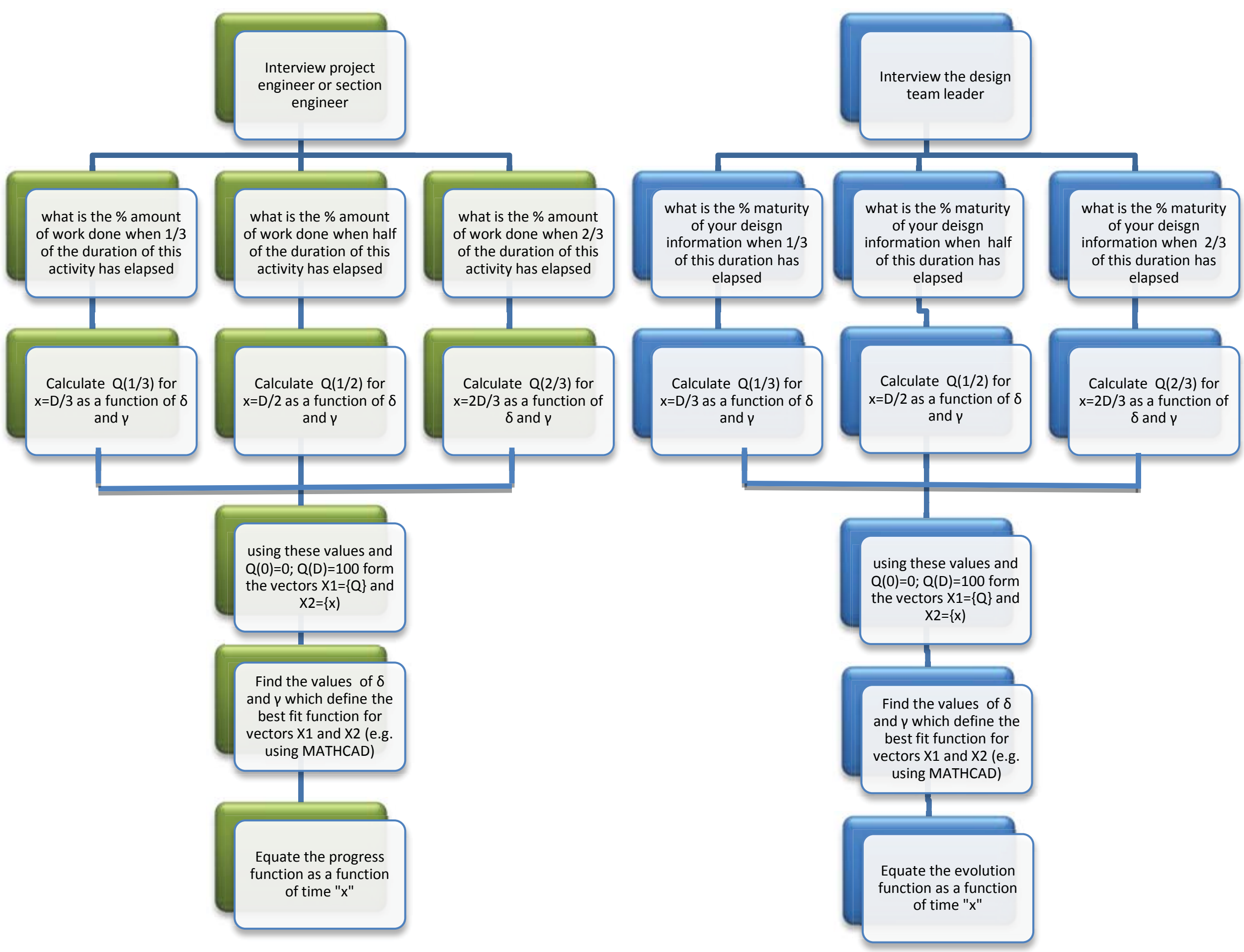

Figure 3. Flow chart for evaluating upstream evolution (right) and downstream progress (left).

$$Q(\gamma, \delta, x) = \gamma * \left(1 - e^{\delta * x}\right) \quad (1)$$

Q is a function of three parameters: "x" is a time variable, whereas "γ" and "δ" are constants that define the shape of Q, and are estimated based on the problem's conditions. "γ" Indicates the magnitude of the function and "δ" indicates the slope of the function. To numerically evaluate these two parameters, a site visit should be arranged to interview site staff to take realistic data about the performance of the studied activities through a brief questionnaire. Figure 3 shows two flow charts illustrating the method used to numerically evaluate progress and evolution. For instance, to evaluate the progress function of a downstream construction activity, a site engineer or a section engineer is interviewed. the downstream construction activity is first identified. Then, the engineer is requested to estimate the amount of downstream work performed when: 1) one-third of the total activity's duration elapses, 2) half of the activity's duration elapses and 3) two-third of the activity's duration elapses. Knowing that an amount of 0 percent progress is recorded when the activity hasn't started, and a 100 percent progress is recorded when the activity is done, we build two vectors $X_1$and X_2 representing the values of durations and their corresponding progress percentages. Finally, we use equation (1) and vectors $X_1$and X_2 to get a best fit function for progress. MATHCAD is used to generate the best fit function by defining the corresponding values of γ and δ.

Similarly, evolution is estimated following the same methodology, however, the interviewed person is a design team leader and he is required to estimate the percentage amount of valuable information which could be transferred to the construction team on the same dates defined in the previous section. Referring to Figure 3, the flow chart describes the evaluation method we developed to measure progress and evolution functions using five points to estimate each vector (described in the above flow chart) in order to have a reasonable amount of accuracy for the parameters. However, for simplicity, one could use any set of points within the allowable domain and build the vectors X_1 and X_2 with a minimum dimension of 3 for a unique vector. D is the total duration of an activity, for which we are estimating the evolution or sensitivity function. n is the total number of points in time selected to evaluate the Q function. In turn, these vectors are denoted as follows:

$$X_1 = \{x_1; x_2; \dots; x_i; \dots; x_n\}$$
$$X_2 = \{Q_1; Q_2; \dots; Q_i; \dots; Q_n\}$$
$$0 \le x_i \le D\,; 0\% \le Q_i \le 100\%\,; 3 \le i \le n.$$

To numerically illustrate this method, we consider a simple example using only three points. For instance, to study upstream activity "issuance of mechanical shop drawings", we interviewed the MEP project engineer of an ongoing residential tower in Lebanon. The project is a 26 story building with six basements and a total built up area of $35000\ m^2$. The project engineer stated that the information released from the issuance of all mechanical shop drawings is minimal in the first half of the activity's duration, and maximal in the second half. This is because; data for each mechanical plan, especially the inverted level and routing of all utilities, are finalized only after the coordination with all trades together is performed. Using this information and following the flow chart process shown in Figure 3, we estimate the evolution function using the minimum requirement of three points. The collected constructed vectors are as follows:

$$X_1 = \left\{0; \frac{2}{3}D; D\right\}\ and\ X_2 = \{0; 30\%; 100\%\}$$

Vector "X_1" represents different times during the entire activity process where the maturity level of the information is evaluated. Vector "X_2" represents the maturity level of the information evaluated at the times of vectors "X_1" respectively. For instance, for this specific example, when two-third of the activity's duration has passed, the information generated has matured only 30%. Similarly, to illustrate the evolution profile we followed the same methodology using the flow chart shown in Figure 3 using also three different points. The two vectors constructed are:

$$\mathrm{Y}_1 = \left\{0; \frac{1}{2}D; D\right\} and\ "Y_2" = \{0; 80\%; 100\%\}$$

According to these vectors, the information generated from the upstream activity has a maturity level of 80% midway through the activity duration.

As mentioned earlier, the elements in the previous vectors reflect the amount of work performed (progress), or the amount of generation of information (evolution) with respect to the total duration of the related activity. Thus, when a different problem is studied, the previous assumptions should be modified accordingly. Table 1 summarizes the general formulation of the high and low evolution and progress. It is essential to mention that table 1 only builds on the vectors mentioned in the previous section which only apply to the specific example considered in this paper. For other problems one should follow the flow chart depicted in Figure 3 and recalculate the parameters in table 1 to suit the addressed problem.

Table1: Values of parameters "δ" and "γ"

	$\boldsymbol{Q(\gamma,\delta,x)=\gamma*\left(1-e^{\delta*x}\right)}$	
	High evolution/Progress	**Low evolution/Progress**
$\boldsymbol{E};\boldsymbol{Pr}$	$\boldsymbol{\gamma}=1.07$ $\boldsymbol{\delta}=-\frac{2.77}{D}$	$\boldsymbol{\gamma}=-0.04$ $\boldsymbol{\delta}=\frac{3.38}{\mathrm{D}}$

3.2 Sensitivity

Sensitivity refers to the amount of additional work or rework required in a downstream activity as a result of a change in the information exchanged from its (predecessor) upstream activity. We describe the amount of rework as a percentage of downstream completed work up until the time at which a modification in the upstream information is recorded. We model this dependency through a mathematical function, starting from 0 percent corresponding to an evolution 100 percent, to a certain percentage amount, not necessarily equal to zero corresponding to an evolution of 0 percent. We define sensitivity as a function of the magnitude of changes occurring in the upstream activity. That is, the more changes are recorded in the upstream activity the more sensitive the activity is. As shown in Figure 2.c and 2.d, sensitivity does not necessarily reach 100 percent. This is the case of activities which require preparation work which could be leveraged even in case of a complete information change. A practical example of such event is a change in the steel reinforcement of a slab with respect to setting the slab. In this case, a major part of the work done is leveraged no matter how large the changes in the slab are, scaffolds and form work remain intact with limited change.

Similar to evolution and progress functions, we distinguish between two types of sensitivity functions, high and low sensitivity functions. High sensitivity function refers to the case where a small amount of change in the upstream information contributes to a large amount of downstream rework. Low sensitivity function refers to the case where a large amount of change in the upstream information contributes only to a small amount of downstream rework. A practical example of activities with high sensitivity functions is the execution of the masonry walls inside an apartment towards the receipt of a modified interior layout for some of the rooms. On the other hand, a practical example of activities with low sensitivity functions is the previously discussed example of the setting of scaffolds and formwork to a slab and a design change of the steel reinforcement of that slab.

We describe sensitivity, S, as the amount of rework to be performed in a downstream activity during the overlap period as a result of a change in the information generated in the upstream activity. In other words, sensitivity is the percentage amount of downstream work to be redone as a result of modified information. Thus, S is a function of the amount of un-evolved information at each point in time, i.e. it is a function of the evolution of the information at each point in time. We assume sensitivity to be a linear function of the likelihood of change, which is the amount of un-evolved information at each point in time, which is also an indication of the likelihood of having rework. This assumption could be relaxed and more complex relationships between sensitivity and the likelihood of rework could be used. For instance, one could use exponential functions or polynomial functions; however this will increase the complexity of the problem and will, thus, require advanced software to solve it.

Sensitivity does not necessarily converge to 100% as we previously mentioned because some activities leverage some of downstream work regardless of any change in the upstream activity. Therefore, we introduce a parameter "α" to account for this issue, "α" represents the overall expected amount of downstream work leveraged despite the changes in the upstream activity. The amount of un-evolved information at each point in time is equal to $(1-E)$ (Yassine 2007). For instance, if the information at a certain point in time has evolved 80% this means that there is still only 20% likelihood of change. As shown in equation (2), the sensitivity and evolution functions have opposite slopes. Therefore, high evolution and high sensitivity cannot be recorded conjunctionally. However, we argue that the sensitivity function is only related to the generic formulation of the evolution function and not the specific value of the function. Thus, one could have a high evolution and a high sensitivity by only using the generic function of low evolution in the sensitivity expression. Figure 4 illustrates a flow chart describing the method of evalua-

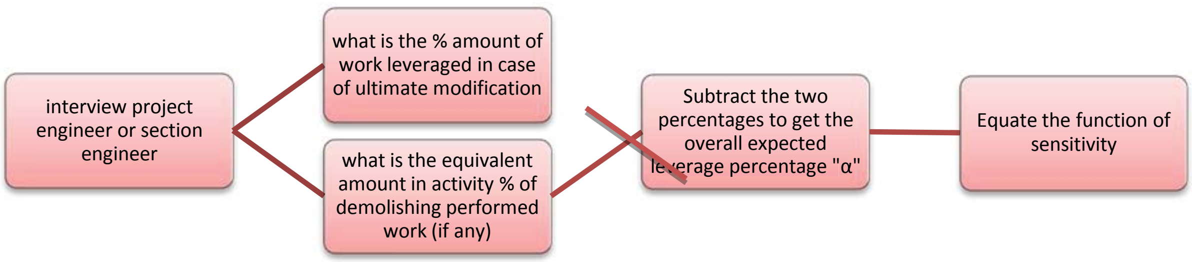

Figure 4. Flow chart for evaluating sensitivity.

tion of parameter "α". The interviewed engineer is requested to estimate the total amount of the work leveraged in case of an extreme change in the information on which he has started downstream work. Also, he is required to estimate, in percentage of downstream work, the amount of demolishing in case there were any.

Equation 2 denotes the mathematical expression of the sensitivity function.

$$S(1-E) = (1-\alpha) * (1-E) \qquad (2)$$

A simple example is considered to numerically illustrate the methodology presented in Figure 4. We interviewed a civil section engineer of the same residential tower described in the previous section. The project engineer stated that for a downstream activity, such as the preparation of a floor slab for pouring concrete, an adequate amount of work (say 40% of the total activity) is leveraged. However, he estimated a trivial value of dismantling and demolishing of wrong work (about 10% of total activity). Thus, this activity will have an expected leveraged amount of 30% towards a change in the steel reinforcement layout. Consequently, the sensitivity function of this downstream activity with respect to changes in the upstream "steel design" activity is mathematically denoted by the following equation:

$$S = 0.7 * \left(1 - \left(\gamma * \left(1 - e^{\delta * x}\right)\right)\right)$$

4. APPLICABILITY OF EVOLUTION, PROGRESS AND SENSITIVITY

In a fast-track environment, one should model the cost and time impact of the increased likelihood of rework as a result of overlapping dependent activities. In fact, rework is directly dependent on the evolution of the information generated from the upstream activity and the sensitivity of the downstream work with regards to change in the upstream information. Consequently, deriving a mathematical function for rework cost and duration allows for modeling the overlapping process. Rework duration, R, is a function of progress of the downstream activity and sensitivity of the downstream work with regards to changes in the information of the upstream activity, because rework duration increases as the amount of work to be performed increases. Additionally, rework duration is the expected value of the performed work; it is subject to a probabilistic distribution to account for the uncertainty of occurrence of the change in the upstream. Thus, rework duration is also a function of the probability of having these changes, which is in turn equal to the amount of un-evolved information at each point in time $(1-E)$. Equation (3) represents the mathematical form of the expression of the expected rework duration.

$$R = f(S; F; 1-E) \tag{3}$$

Similarly, the expected rework cost recorded as a result of the overlapping process is a function of the same probability as the rework duration. It is also a function of: 1) the sensitivity of the downstream activity with respect to changes from the upstream activity, 2) the progress of downstream work at each point in time and 3) the total cost of the entire downstream activity. In fact, the progress of downstream work and the total cost of the entire downstream activity are an indication of the total amount paid to perform the work. Sensitivity is an indication of the amount of rework and the probability is an indication of the expected value of the cost. Thus, getting all these parameters in one function allows for estimating the expected rework cost at each point in time. As previously shown, we consider that there is a linear relationship between the progress function of an activity and the cost of performing this activity. This means that an evolution of 80% in an activity reflects expenditures of 80%. This assumption was made to reduce the complexity of the problem. In turn, any cost figure could be applied to configure the cost expenses in a certain activity. Exponential or polynomial mathematical functions could be used to describe the cost and thus utilized in the expected rework cost function. Similarly, it may appear that the cost of reworking an activity is the same as performing it the first time, however, we argue that the wasted materials or the demolition and evacuation activities are accounted for in the value of the work leveraged presented in parameter "α". Equation (4) represents a mathematical expression of the expected rework cost.

$$H = f(C; S; F; 1-E) \tag{4}$$

Equations (3) and (4) represent a mathematical configuration of the expected rework duration and rework cost without explicitly evaluating their mathematical expressions. These two equations define the core parameters of any optimization problem taking into consideration the overlapping process. Thus, these equations could be used to build and solve an optimization model to get an optimal overlapping policy for any two dependent activities. By running the optimization model on several pairs of dependent sequential activities, one could reshuffle an original project schedule into a fast-tracked schedule and, thus, remarkably reduce the entire project duration.

5. CONCLUSION

This paper explained and illustrated the constructs of upstream evolution and downstream sensitivity and progress in the context of construction projects. An emphasis is placed on the relationship between these parameters and the maturity of the information generated in the upstream activity and the rework duration and cost resulting from the change of this information. The paper also discussed the applicability of these parameters in finding an expected value of rework cost and duration resulting from overlapping. This information can be used in building an optimization model to identify the best overlapping strategy of sequential dependent activities.

This study relies on several assumptions. For example, we assumed a linear relationship between sensitivity and likelihood of rework. We also assumed that the amount of un-evolved information is a proxy for the probability of change or rework. Future work should address these assumptions. In addition, we assumed that the progress of an activity is correlated with the corresponding expenditures, i.e. 80% of the budgeted cost for an activity is spent when 80% of the work is performed. Finally, the study does not address the effect of overlapping on the quality of work performed. However, the presented models and associated survey instruments constitute a solid foundation for starting to think more analytically about fast-tracking and the underlying constructs (of evolution and sensitivity) that enable the development of any future analytical techniques. For instance, we are currently using the presented methodology to build an optimization model which is a non-linear program (NLP) to define the optimal overlapping strategy.

ACKNOWLEDGMENTS

The authors would like to thank the American University Of Beirut's University Review Board for funding this study.

REFERENCES

Bhuiyan, N., Gerwin, D., and Thomson, V., (2004). "Simulation of the New Product Development Process for Performance Improvement" *Management Science* 50(12), pp. 1690–1703, ©2004 INFORMS

Blacud, N., Bogus, S., Diekmann, J., and Molenaar, K., (2009)."Sensitivity of construction activities under design uncertainty," *Journal of Construction Engineering and Management* 135(3): 199-206.

Bogus, S., Diekmann, J.,Molenaar, K., R., Harper, C., Patil, S., and Lee, J., S., (2011). "Simulation of Overlapping Design Activities in Concurrent Engineering" *Journal of Construction Engineering and Management* ASCE / NOVEMBER 2011 / pp.950-957.

Bogus, S., Molenaar, K., R., and Diekmann, J., (2005). "Concurrent Engineering Approach to Reducing Design Delivery Time" *Journal of Construction Engineering and Management* ASCE / NOVEMBER 2005 / pp.1179-1185

Bogus, S., Molenaar, K., R., and Diekmann, J., E., (2006). "Strategies for Overlapping Dependent Design Activities," *Construction Management and Economics* 24: 829-837.

Krishnan, V., Eppinger, S., D., and Whitney, D., E., (1997). "A Model-Based Framework to Overlap Product Development Activities," *Management Science* 43(4): 437-451.

Lin, J., Chai, K., H., Brombacher, A., C., and Wong, Y., S., (2009). "Optimal Overlapping and Functional Interaction in Product Development," *European Journal of Operational Research* 196 (3): 1158-1169.

Lin, J., Chai, K., H., Wong, Y., S., and Brombacher, A., C., (2008). "A Dynamic Model for Managing Overlapped Iterative Product Development," *European Journal of Operational Research* 185(1): 378-392.

Loch, C., H., and Terwiesch, C., (1998). "Communication and Uncertainty in Concurrent Engineering," *Management Science* 44(8): 1032-1048.

Peña-Mora, F., and Li, M., (2001). "Dynamic Planning and Control Methodology for Design/Build Fast=Track Construction Projects" *Journal of Construction Engineering and Management* JANUARY/FEBRUARY 2001/pp.1-17.

Roemer, T., and Ahmadi, R., (2004). "Concurrent Crashing and Overlapping in Product Development," *Operations Research* 52(4), pp. 606-622, ©2004 INFORMS.

Yassine, A., (2007). "Investigating product development process reliability and robustness using simulation," *Journal of Engineering Design*18 (6): 595-561.