

## On Calculating Activity Slack in Stochastic Project Networks

Gary Mitchell

Pamplin School of Business Administration,  
University of Portland, Portland, OR 97203-5798

**Abstract: Problem statement:** Identifying critical tasks in a project network is easily done when task times are deterministic, but doing so under stochastic task times is problematic. The few methods that have been proposed contain serious drawbacks which lead to identifying critical tasks incorrectly, leaving project managers without the means to (1) identify and rank the most probable sources of project delays, (2) assess the magnitude of each source of schedule risk, and (3) identify which tasks represent the best opportunities for successfully addressing schedule risk? **Approach:** In this study we considered the problem of identifying the sources of schedule risk in a stochastic project network. We developed general expressions for determining a task's late starting and ending time distributions. We introduced the concept of stochastic slack and develop a number of metrics that help a project manager directly identify and estimate the magnitude of sources of schedule risk. Finally, we compared critical tasks identified using the activity criticality index to those found using stochastic slack metrics. **Results:** We have demonstrated that a task may have non-zero probability of negative stochastic slack and that expected total slack for a task may be negative. We also found that while the activity criticality index is effective for calculating the probability that a task is on a critical path, the stochastic slack based metrics discussed in this paper are better predictors of the extent to which a delay in a task will result in a project delay. **Conclusion/Recommendations:** Project managers should consider using stochastic slack based metrics for assessing project risk and establishing the most likely project schedule outcomes. Given the calculation complexity associated with theoretically exact stochastic slack metrics, effective heuristics are required.

**Key words:** Project management, project planning, stochastic activity durations, project risk, activity slack

### INTRODUCTION

The difficulties associated with managing projects under uncertainty are widely recognized. In its controversial Chaos Report, the Standish Group provides a discouraging picture of IT project success rates (Dominguez, 2009). In fact, if the Standish Group's survey results are representative of actual IT performance, there has been a reversal in the improvement trend observed during the late 1990's. Figure 1 presents the results from eight Chaos Reports, beginning with the first report based on surveys taken in 1994. The apparent trend, in spite of early improvements, is an increasing rate of failed projects. The reason for this trend is unclear, but a reasonable assumption is the increasing complexity of modern IT projects.

Although the Chaos Report's results are specific to IT projects, many of the factors increasing IT project complexity today are responsible for increasing the complexity of projects of all kinds including, for example, projects in new product development, supply

chain management and construction. These complications arise from the need to integrate technologies of ever-increasing complexity, a continuing trend toward the use of outsourcing and virtual teams, significant competitive pressures resulting in the need for rapid project deployment and quick completion and a reliance on large project teams.

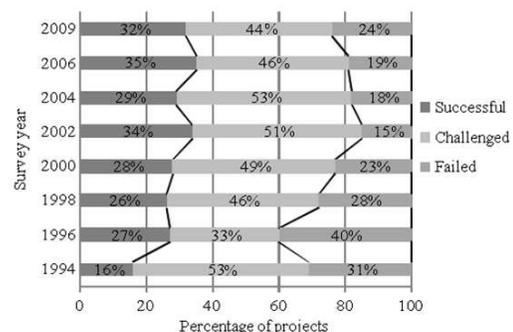


Fig. 1: IT project performance

No one factor can account for, or prevent, failure in a project. It is common practice for a project team to develop a comprehensive risk assessment and risk management plan. Frequently, identified risks are categorized by risk class. A typical project risk taxonomy might include five classes of risk: Budget, political, resource, schedule and technology. Clearly, these are not independent with obvious relationships existing between and among all classes. However, considering that the Standish Group defines a successful project as one completed on time, within budget and delivering substantially all of the original promised scope, two of these risk classes, budget and schedule risk, are of particular consequence in evaluating the project performance patterns demonstrated in Fig. 1 and generalizing to other types of projects.

Schedule risk, in particular, poses a significant hazard for project managers. Prevailing methods fail to fully consider the uncertainty associated with task completion and the interaction of task time uncertainty and project network topology. Identifying the “most critical” tasks with regard to schedule risk is a problem faced by all project managers. Yet, the question, “How can a project manager identify the “most critical” tasks with regard to schedule risk?” remains incompletely answered at best. An effective approach for answering this question is desperately needed by practicing project managers faced with evaluating the schedule impact of hundreds and thousands of tasks.

The problem of identifying critical tasks in a deterministic network is well understood. Standard Critical Path Method (CPM) analyses can be used to identify the longest path(s), known as the critical path(s), in an activity network. Multiple critical paths may exist, but all will be of equal length. A project’s critical tasks are those that lie along a critical path. These methods are described in many sources including Demeulemeester and Herroelen (2002) and Klastorin (2004).

A task’s Total Slack (TS) is defined as the amount of time a task may be delayed without causing a delay in the project and is calculated for task *i* as:

$$TS_i = LS_i - ES_i = LF_i - EF_i \quad (1)$$

Where:

$LS_i$  and  $ES_i$  = The task’s late and early starting times  
 $LF_i$  and  $EF_i$  = The task’s late and early finish times respectively as determined using the standard forward and backward pass calculations of the critical path method

Tasks on a project’s critical path have total slack equal to zero and are the project’s “critical” tasks. It is possible to identify the critical path(s) in a deterministic project network as an unbroken sequence of tasks with total slack equal to zero.

We immediately encounter difficulties developing concepts analogous to total slack and “critical” tasks for stochastic project networks. Even the most basic concept of a single longest critical path (or multiple longest paths with the same length) through the network no longer applies, as almost any path may be the longest path through the network with non-zero probability. Perhaps the most well-known example of a criticality metric in the stochastic network setting is the criticality index, defined as the probability that a task will lie on a critical path (Demeulemeester and Herroelen, 2002). However, a task may lie on a critical path without introducing risk of project delay (i.e., schedule risk) into the project network. A small example illustrates this concept.

Consider the small two-task series network in Fig. 2. Assume task A completes in 2 weeks with a 40% probability or in 5 weeks with a 60% probability. The expected duration of task A,  $E[t_A]$ , is 3.8 weeks. Assume that task B completes in 2 weeks with probability 1.0. Then the expected makespan of the project,  $E[M]$ , is 5.8 weeks and there is a 60% chance the actual project duration will exceed  $E[M]$ . Both tasks A and B are characterized by a criticality index of 1.0, as both lie on the project’s only critical path. A project manager using the classical criticality index would identify both as “critical” tasks, making them the focus of significant management, tracking and control effort. However, a moment’s consideration clearly indicates that the two tasks do not contribute equally to the project’s schedule risk. In fact, in this example, all of the schedule risk is introduced by task A. Although task B lies on the critical path and may, as the result of delays in task A, begin and end after its expected starting and ending times, Task B is capable only of preserving the delay introduced by Task A along the critical path. A rational project manager would recognize Task A, but not Task B, as a source of significant schedule risk. That is not to say that Task B is of no interest to the project manager. On the contrary, Task B may be a source of recovery for delays introduced into the network by Task A. It is simply that Task B itself is not a source of new delays into the project network.

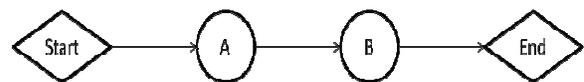


Fig. 2: Two task series network

In this study, we discuss the problems and deficiencies associated with commonly used methods for identifying the most critical tasks with regard to schedule risk. We introduce the concept of stochastic slack, provide analytical expressions for determining the total slack distribution and demonstrate, through simple numerical examples, how stochastic slack can help a project manager effectively identify those tasks which embody the greatest schedule risk.

This study makes three contributions. First, we demonstrate how to calculate the late starting and ending time distributions for every task in the network. To the best of our knowledge, this is the first stochastic project network research to consider this problem. Second, we consider the question of whether the concept of activity slack, well-defined in deterministic project networks, has an analogous measure in stochastic project networks. To the best of our knowledge, this is the first paper to look at this important and relevant question. Finally, we demonstrate that stochastic slack measures can be effective in helping project managers identify tasks representing the largest sources of scheduling risk.

This study is related to previous research in project management, tracking and control. The Classic PERT method, developed as a result of the Polaris Weapons System program in the 1950's (Nahmias, 2005), is probably the best-known mechanism for considering the stochastic nature of task durations and is the mechanism most often implemented by project management software products attempting to support non-deterministic task durations. Using PERT, it is possible to develop estimates of the uncertainty of the project makespan. Elmaghraby (1977) and Klastorin (2004) point out a number of problems with the PERT method, including its determination of project makespan using expected task durations.

Numerous papers have been written about the PERT method, with the earliest appearing very shortly after its introduction. Most of these have dealt with questions and criticisms, about PERT's assumptions of beta distributed activity times, its approximations for activity time means and variances and the subjective nature of the time estimates upon which PERT is based. Fulkerson (1962) discusses the PERT method and develops a procedure for bounding the true expected project makespan from below. MacCrimmon and Ryavec (1964) critically evaluate PERT assumptions and discuss key sources of error in the computed expected makespan. They discuss the relationship between parallelism in the activity network and their relative lengths and the extent of errors in the PERT expected makespan. Littlefield and Randolph (1987), in

reply to a question posed by Sasieni regarding the origin of PERT's formula for the mean time for an activity, summarize the development of the PERT formulae. Williams (1995) discusses the meaning of PERT estimates in the context of management by objectives and Parkinson's Law. He discusses the possibility that activity time distributions may be discontinuous about the mean.

Additional work related to the problem of determining the expected project makespan with stochastic task durations includes papers by Van Slyke (1963) who suggests Monte Carlo Simulation as a viable method for constructing the project makespan distribution, Martin (1965) who defines a network reduction approach for determining the makespan Probability Density Function (PDF), Dodin (1984) who develops a heuristic approach to finding the  $k$  most critical paths through a project network, Dodin (1985a) who develops an approximation for the makespan CDF, Kleindorfer (1971); Robillard and Trahan (1976) and Dodin (1985b) who obtain bounds for the makespan PDF and Kulkarni and Adlakha (1986) who develop the makespan distribution for a project network with exponentially distributed task times using a Markov Pert Networks (MPN). Many of these, including Dodin (1985a), developed approximations using discretization of continuous density functions, simplifying the convolution of task densities. This offers a more practical implementation than Martin's exact method, but introduces error as the result of discretizing the distributions.

Hagstrom (1990) developed a recursive algorithm for determining either the CDF or moments of the project makespan distribution. In a separate paper, Hagstrom (1988) discussed the computational complexity of PERT problems and demonstrated that computing the makespan distribution is #P-complete, computing the expected makespan is at least as difficult and neither can be computed in polynomial time.

In an important related paper, Elmaghraby (2000) reviews the literature on determining the criticality of activities in stochastic project networks, develops a taxonomy of sensitivity issues and evaluates a number of approaches for assessing criticality and sensitivity, including those suggested by Williams (1992) and Cho and Yum (1997). Elmaghraby concludes that existing measures can be misleading and difficult to evaluate and that a need exists for an easier approach. Demeulemeester and Herroelen (2002) conclude that the issue of determining a meaningful indicator of the criticality of an activity has not been settled. Elmaghraby (2005) demonstrates that the use of expected values to analyze deterministic equivalent

projects can lead to incorrect project scheduling and resource allocation decisions.

Finally, this paper is related to work in robust project scheduling which reduces project makespan uncertainty by inserting buffers designed to reduce the project minimize task starting time deviations from a pre-schedule (Herroelen, 2007).

The rest of this study is organized as follows. First, we explicitly define the general problem, including assumptions and develop the analytical expressions for late starting and ending time distributions and the total slack distribution. Next, we use simulation to develop simple examples to illustrate stochastic slack measures and compare them to the common criticality index. Finally, we summarize our findings, discuss implications for practicing project managers and identify potential extensions of present study.

### MATERIALS AND METHODS

**Early and late schedule distributions:** We assume that a project can be represented by a directed acyclic graph  $G = \{N, A, W\}$  with a set of nodes  $N = \{1, \dots, n\}$ , a set of directed arcs  $A = \{(i,j)\}$  and a set of node weights  $W = \{d_i\}_{i \in M}$  where nodes  $i \in M$  represent tasks with durations  $d_i$  and the arcs  $(i,j)$  represent finish-to-start precedence relationships between tasks  $i$  and  $j$  with zero lags. Tasks 1 and  $n$  denote the starting and ending nodes of the project, respectively; by definition,  $d_1 = d_n = 0$ . This notation is generally referred to as AON (Activity-on-Node) project representation (Elmaghraby, 1977; Klastorin, 2004).

For deterministic project networks, we define a feasible project schedule to be a set of task starting and ending times that satisfy all precedence and due date constraints. Let  $\pi_k$  represent the  $k^{\text{th}}$  path through the network and  $|\pi|$  represent the length of path  $\pi$  through a deterministic project activity network. A path:

$$\pi_k \in P = \{\pi_1, \pi_2, \dots, \pi_n\}$$

where,  $P$  is the set of all paths through the network  $G$ , is a critical path if  $|\pi_k| = \max(P)$ . Define the project makespan,  $M = \max(P)$ , as the project completion time. Then given the definitions above and the total slack definition in Eq. 1,  $TS_i \geq 0$  for all tasks  $i \in N$ . Moreover, given a feasible schedule for a deterministic project network,  $TS_i = 0 \forall i \in \pi_k$  for all  $k: |\pi_k| = M$ .

Now consider a stochastic project network such that task activity times (durations) are random variables. Task starting and ending times, as well as activity slack times, are therefore random variables. We

define the following notation for use in the subsequent discussion:

- $\Phi_i$  = The set of immediate predecessors of task  $i$
- $\Gamma_i$  = The set of immediate successors of task  $i$
- $a_i(t)$  = The activity time Probability Density Function (PDF) for task  $i$
- $A_i(t)$  = The activity time Cumulative Distribution Function (CDF) for task  $i$
- $c$  = Schedule type,  $c \in \{E(\text{arly}), L(\text{ate})\}$
- $s_{c,i}(t)$  = The  $c$  start time PDF for task  $i$
- $S_{c,i}(t)$  = The  $c$  start time CDF for task  $i$
- $f_{c,i}(t)$  = The  $c$  finish time PDF for task  $i$
- $F_{c,i}(t)$  = The  $c$  finish time CDF for task  $i$

The early start schedule distributions, assuming continuous distributions, can now be defined similarly to Martin (1965) and Dodin (1985a):

$$S_{E,i}(t) = \prod_{j \in \Phi_i} F_{E,j}(t) \tag{2}$$

$$s_{E,i}(t) = \frac{dS_{E,i}(t)}{dt} \tag{3}$$

$$F_{E,i}(t) = \int_0^t A_i(t-x) s_{E,i}(x) dx \tag{4}$$

$$f_{E,i}(t) = \frac{dF_{E,i}(t)}{dt} \tag{5}$$

Assuming, without loss of generality, lexicographic ordering of the tasks, such that  $i < j \forall i \in \Phi_j$  and  $j > i \forall j \in \Gamma_i$ , Eq. 2 and 5 may be used to determine the early starting and ending time distribution for every task in the network. This may be done by proceeding sequentially forward through the network, in a manner analogous to Mitchell and Klastorin (2007) and similar to Dodin (1985a).

When the early start distributions have been determined, the late start schedule distributions can be calculated using the following, by setting  $F_{L,n} = F_{E,n}$  and proceeding sequentially backwards through the network, beginning with task  $n$  and terminating with task 1, using Eq. 8 and 7:

$$S_{L,i}(t) = \int_{x=0}^t \int_{y=t}^{\infty} a_i(y-x) f_{L,i}(y) dy dx \tag{6}$$

$$s_{L,i}(t) = \frac{dS_{L,i}(t)}{dt} \tag{7}$$

$$F_{L,i}(t) = \prod_{j \in \Gamma_i} \{1 - S_{L,j}(t)\} \tag{8}$$

$$f_{L,i}(t) = \frac{dF_{L,i}(t)}{dt} \tag{9}$$

The computational intractability of determining the makespan distribution  $F_{E,n} = F_{L,n}$  is well known and was discussed previously. However, these expressions are useful, representing the exact approach for which practical heuristics may be developed.

**Stochastic slack:** We now develop an exact expression of stochastic total slack using the definitions in Eq. 2-9. The CDF for the Total Slack (TS) distribution for task  $i$  is calculated as follows:

$$G_{ts,i}(t) = \int_0^\infty \int_{-\infty}^{x+t} s_{E,i}(x) s_{L,i}(y) dy dx \tag{10}$$

from which we can easily find the probability density function:

$$g_{ts,i}(t) = \frac{dG_{ts,i}(t)}{dt} \tag{11}$$

Using early and late starting time distributions, expected total slack for task  $i$  can be calculated as follows:

$$E[TS_i] = \int_0^\infty \int_{-\infty}^\infty (y-x) s_{E,i}(x) s_{L,i}(y) dy dx \tag{12}$$

### RESULTS

It is clear from Eq. 12 that expected total slack can be negative. This results from the calculation approach. We set the late finish distribution equal to the early finish distribution and work backwards through the network, calculating the late start and late finish distributions for every task in the network. Clearly then, there will be some potential outcomes for which at least one path through the network is longer than at least one other path through the network. This results in the possibility for  $TS_i < 0$  and, in fact, late start realizations for some tasks such that the late start time is less than zero.

It is our contention that the existence of negative total slack can be used by project managers to identify

tasks which, if delayed, are most likely to delay the project. In fact, expected total slack need not be negative to provide valuable insight into a project. If expected total slack is calculated using Eq. 12 for every task  $i$  in the network, then a simple ranking of tasks by  $E[TS]$  can identify the tasks most likely to introduce a delay into the project. The task with the lowest  $E[TS]$  would be the most likely to introduce a delay, while the task with the largest  $E[TS]$  would be least likely to delay the project.

While such a task ranking is valuable, Eq. 10 may be used to calculate  $\Pr\{TS_i < 0\}$  for every task  $i \in N$ . For example, we might have a task with  $E[TS_i] = 1$  but  $\Pr\{TS_i < 0\} = 0.45$ . In this case, a project manager would recognize that even though he expected slack for a task is non-negative, the task has a 45% chance of exhibiting negative total slack and therefore introducing delay along its path(s).

### DISCUSSION

While the expressions given in equations (10) through (12) pose no theoretical difficulties, they are computationally intractable for even small sample project networks. Therefore, we use simulation to illustrate the concepts previously discussed using a simple numerical example and compare stochastic slack to the commonly calculated activity criticality index. We also introduce an additional key metric for a practicing project manager; the probability that the project is late given negative total slack for task  $i$ :

$$\Pr\{L > 0 | TS < 0\}$$

where,  $L$  represents project lateness, is a direct measure of the schedule risk posed by the uncertainty associated with task  $i$ 's activity time and its location within the project network.

**Example details:** Consider the project network in Fig. 3. By convention, tasks 1 and 6 have zero activity duration with probability 1.0. Therefore, any delay introduced into the project schedule will be introduced by one of the remaining tasks 2-5. Clearly, on the basis of the criticality index, Task 5 would be considered the most critical task in this network with  $CI_5 = 1.0$ , where  $CI_i$  is the criticality index for task  $i$ .

Whereas the criticality index for task 5 could be determined strictly from its location within the project network, those for tasks 2 through four depend on both their locations within the project network and their activity time distributions given in Table 1.

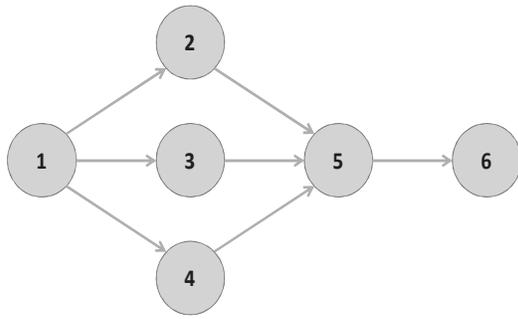


Fig. 3: Sample project network

Table 1: Sample project task activity time distributions

	Task					
	1	2	3	4	5	6
Distribution	Det	Beta	Beta	Beta	Beta	Det
Mean	0	51.00	43.00	52.00	19	0
SD	0	18.55	10.33	18.49	8.94	0

Table 2: Stochastic slack measures for sample project given  $L > 0$

	Task			
	2	3	4	5
Criticality index	0.2277	0.0851	0.6871	1.00000
$E[TS L]$	10.8369	17.3785	-1.3717	-0.07810
Delta $E[TS]$	10.9151	17.4566	-1.2936	-0.07810
$Pr\{TS < 0 L\}$	0.3327	0.2040	0.5426	0.65030
$Pr\{L TS < 0\}$	0.5015	0.5988	0.7851	0.48290
Corr(d,M)	0.3471	0.0095	0.1239	0.35870

For the purpose of the example, we set the project due date  $D = E[M] = 56.4$  and calculated lateness  $L = E[M] - D$  so that  $L > 0$  when the project completes after its due date. Using 1000 simulated project realizations, we determined that  $Pr\{L > 0\} = 0.505$  and  $Pr\{L \leq 0\} = 0.495$ . Thus, even for a fairly small number of trials, we see that there is essentially no bias toward delayed or early completions for the project (as expected given  $D = E[M]$ ).

**When the project is delayed:** Perhaps the most significant insight is drawn from considering stochastic slack metrics calculated only from trials in which the project experienced a delay. Table 2 presents these results for the sample project. As expected, the criticality index for task 5 is 1.0. The next most critical task is task 4 with  $CI_4 = 0.6871$ , indicating that task 4 is on the critical path 68.71% of the time. Looking first at  $E[TS|L]$ , we see that task 4 has a significantly larger negative expected slack than task 5, suggesting that task 4 is responsible for introducing more delay into the

project than task 5. On the basis of the CIs, we would rank the tasks (from most likely to least likely to delay the projects) as 5,4,2,3. However, the expected total slacks suggest a ranking of 4,5,2,3 is more appropriate. The  $\Delta E[TS]$  values describe the extent to which an activity reduces or restores flexibility into the network. Thus, Task 4 with  $\Delta E[TS_4] = -1.3717$  has the largest negative delta, while task 3 has the largest positive delta. These  $\Delta E[TS]$  values can be useful for identifying the tasks most likely to delay a path and which tasks along the same path, offer the best opportunities for a project manager to recover from a delay introduced by another task.

The most interesting results in Table 2 are the values for  $Pr\{L|TS < 0\}$ . These tell us, for example, that the project was late 78.51% of the time, when  $TS_4 < 0$ , but only 48.29% of the time when  $TS_5 < 0$ . In fact, all three of tasks 2 through 4 have higher values for  $Pr\{L|TS < 0\}$  than task 5, indicating that task 5 is the least critical of the four tasks (although the differences between the probabilities for tasks 2 and 5 is small). This is a significant difference from the information offered by the criticality index. Thus, a rational project manager would provide the strongest focus on tasks 3 and 4, with a complete ranking of 4, 3, 2, 5 (compared to 5, 4, 2, 3 as suggested by the criticality index).

Significantly, we observe that  $E[TS_2|L] = 10.3869$  and  $Pr\{L|TS_2 < 0\} = 0.5015$  while  $E[TS_3|L] = 17.3785$  and  $Pr\{L|TS_3 < 0\} = 0.5988$ . In other words, a delay in task 3 is approximately 19.4% more likely to delay the project than is a delay in task 2, even though task 3 has a larger expected total slack. This finding suggests that considering the size of the summary statistic,  $E[TS|L]$ , is insufficient for characterizing the schedule risk associated with a task. A rational project manager must consider  $Pr\{TS < 0|L\}$ , which identifies the tasks most likely to be delayed when the project is late and  $Pr\{L|TS < 0\}$ , which directly indicates the likelihood of a late project given a task delay.

Finally, the  $corr(d,M)$  values shown in the last row of Table 2 represent the correlation of task activity time realizations with project makespan realizations. This measure has been reported by Elmaghraby (2000) as a potential metric for identifying the most critical tasks in a project network. For the sample network, the correlation between task duration and project makespan yields a different ranking than criticality index,  $Pr\{TS < 0|L\}$  and  $Pr\{L|TS < 0\}$ .

Table 3: Stochastic slack measures for sample project when  $L \leq 0$

	Task			
	2	3	4	5
Criticality index	0.2277	0.0851	0.6871	1.00000
$E[TS E]$	10.5186	21.2111	21.4822	-1.12710
$\Delta E[TS]$	11.6457	22.3382	22.6093	-1.12710
$\Pr\{TS > 0 E\}$	0.6626	0.8606	0.8485	0.45050
$\Pr\{E TS > 0\}$	0.5015	0.5145	0.6452	0.47050
$\text{Corr}(d,M)$	0.3265	0.2171	0.4349	0.32910

Given the direct relationship between the two probability measures and project lateness, the example suggests this is a poor metric for assessing task criticality. The likely reason for the metric's poor performance is that it ignores the project network topology, focusing only on the distribution of activity times compared to the makespan distribution. Network topology plays a fundamental role in the translation of task time uncertainty to schedule risk. The shape of the network may either insulate a project from or exacerbate the impact of task time uncertainty.

**When the project is completed early:** A rational project manager is interested in more than the conditions that are likely to delay a project. Proactive management of a project requires identifying opportunities for changing the risk characteristics of the project. Metrics similar to those calculated to identify late completion risk, but where the project completes early, can be valuable for this purpose.

Here, again, we see the criticality index emphasizing task 5 as the most critical. However, both  $\Pr\{TS > 0|E\}$  and  $\Pr\{E|TS > 0\}$  indicate that task 5 has the least chance of influencing an early project completion. In this case, a rational project manager desiring an early project completion, or even a reduction in schedule risk, might choose to target an early completion of task 4 or 3. These offer the highest likelihood of effecting an early completion (Table 3).

The correlation between activity duration and makespan demonstrates similarly poor performance here as in the late delivery case.

### CONCLUSION

In this study, we address an important problem faced by many project managers; specifically, we introduce the concept of stochastic slack to address the problem of identifying the critical tasks in a stochastic project network. Our stochastic slack based metrics can be used by a project manager to directly assess the likelihood that a delay in a task will result in a project delay. While present study is related to previous

research on criticality and sensitivity in stochastic activity networks, our approach is the first to consider the concepts of stochastic slack, negative expected slack and the use of stochastic slack based metrics to directly assess the extent to which tasks introduce schedule risk or flexibility.

We developed expressions for determining PDFs and CDFs for the late start and late finish distributions. To the best of our knowledge, this is the first research to consider the distributions of late starting and ending times; previous research has implicitly addressed only early starting and ending time distributions. Using the starting and ending time distributions, we developed general expressions for determining the total slack distribution and expected total slack for a task. While these distributions are theoretically straightforward, their use is computationally intractable and the development of efficient and accurate heuristics will be necessary in order to extend these results.

Due to the inherent difficulties associated with calculating starting and ending time distributions in stochastic activity networks, we used a simulation-based example to illustrate stochastic slack concepts. We compared Stochastic Slack Based (SSB) metrics to the commonly used criticality index and found that the SSB metrics provided superior insights into the likely sources of project delays. We also showed that the correlation between task duration and project makespan compares unfavorably to the SSB metrics in identifying likely sources of project delays.

This research represents the first of a number of research projects related to the problem of replenishing non-stationary intermittent demand items. We are currently working on a related project to further develop the concept of Due Date Based (DDB) slack. We are also developing a large scale simulation to evaluate SSB and DDB metric performance in identifying and prioritizing sources of schedule risk. In another planned extension to this study, we will consider whether stochastic analogs exist for the deterministic concepts of free slack, safety slack and independent slack and what they imply for a project manager.

Many applications of slack measures can be found in the deterministic project planning and scheduling literature. Numerous heuristics for the time-cost trade-off and resource constrained project scheduling have been developed using deterministic slack measures. Extending this study to the development of stochastic time-cost tradeoff and resource constrained project scheduling heuristics is an obvious area for future research.

**REFERENCES**

- Cho, J.G. and B.J. Yum, 1997. An uncertainty importance measure of activities in PERT networks. *Int. J. Prod. Res.*, 35: 2737-2757. DOI: 10.1080/002075497194426
- Demeulemeester, E. and W. Herroelen, 2002. *Project Scheduling: A Research Handbook*. 1st Edn., Kluwer Academic Publishers, Boston, MA., ISBN 10: 1402070519, pp: 549.
- Dodin, B., 1984. Determining the K Most critical paths in PERT networks. *Operat. Res.*, 32: 859-877. DOI: 10.1287/opre.32.4.859
- Dodin, B., 1985a. Approximating the distribution functions in stochastic networks. *Comput. Operat. Res.*, 12:251-264. DOI: 10.1016/0305-0548(85)90024-3
- Dodin, B., 1985b. Bounding the project completion time distribution in PERT networks. *Operat. Res.*, 33: 862-881. DOI: 10.1287/opre.33.4.862
- Dominguez, J., 2009. The curious case of the CHAOS report. <http://www.projectsmaart.co.uk/the-curious-case-of-the-chaos-report-2009.html>
- Elmaghraby, S.E., 1977. *Activity Networks: Project Planning and Control by Network Models*. 1st Edn., John Wiley and Sons, New York, ISBN: 10: 0471238619, pp: 229-238.
- Elmaghraby, S.E., 2000. On criticality and sensitivity in activity networks. *Eur. J. Operat. Res.*, 127: 307-313. DOI: 10.1016/S0377-2217(99)00483-X
- Elmaghraby, S.E., 2005. On the fallacy of averages in project risk management. *Eur. J. Operat. Res.*, 165: 220-238. DOI: 10.1016/j.ejor.2004.04.003.
- Fulkerson, D.R., 1962. Expected critical path lengths in PERT networks. *Operat. Res.*, 10: 808-817. DOI: 10.1287/opre.10.6.808
- Hagstrom, J., 1988. Computational complexity of PERT problems. *Networks*, 18: 139-147. DOI: 10.1002/net.3230180206
- Hagstrom, J.N., 1990. Computing the probability distribution of project duration in a PERT network. *Networks*, 20: 231-244. DOI: 10.1002/net.3230200 208
- Herroelen, W., 2007. Generating robust project baseline schedules. *TutORials in Operations Research. INFORMS National Meeting, Seattle, WA.*, ISBN:13: 978-1-877640-22-3, pp: 124-144.
- Klastorin, T.D., 2004. *Project Management: Tools and Trade-Offs*. 1st Edn., John Wiley and Sons, Inc., New York, ISBN: 10: 0471413844, pp: 139-143.
- Kleindorfer, G.B., 1971. Bounding distributions for a stochastic acyclic network. *Operat. Res.*, 19: 1586-1601. DOI: 10.1287/opre.19.7.1586
- Kulkarni, V.G. and V.G. Adlakha, 1986. Markov and Markov-regenerative PERT networks. *Operat. Res.*, 34: 769-781. DOI: 10.1287/opre.34.5.769
- Littlefield, T.K. and P.H. Randolph, 1987. Reply: An answer to Sasieni's question on PERT times. *Manage. Sci.*, 33: 1357-1359. DOI: 10.1287/mnsc.33.10.1357
- MacCrimmon, K.R. and C.A. Ryavec, 1964. An analytical study of the PERT assumptions. *Operat. Res.*, 12: 16-37. DOI: 10.1287/opre.12.1.16
- Martin, J.J., 1965. Distribution of time through a directed acyclic network. *Operat. Res.*, 13: 46-66. DOI: 10.1287/opre.13.1.46
- Mitchell, G. and T. Klastorin, 2007. An effective methodology for the stochastic project compression problem. *IIE Trans.*, 39: 957-969. DOI: 10.1080/07408170701315347
- Nahmias, S., 2005. *Production and Operations Analysis*. 5th Edn., McGraw-Hill Irwin, New York, USA., ISBN: 10: 0072865385, pp: 524-525.
- Robillard, P. and M. Trahan, 1976. Expected completion time in PERT networks. *Operat. Res.*, 24: 177-182. DOI: 10.1287/opre.24.1.177
- Van Slyke, R.M., 1963. Monte carlo methods and the PERT problem. *Operat. Res.*, 11: 839-860. DOI: 10.1287/opre.11.5.839
- Williams, T.M., 1992. Criticality in stochastic networks. *J. Operat. Res. Soc.*, 43: 353-357. DOI: 10.1057/jors.1992.50
- Williams, T.M., 1995. What are PERT estimates? *J. Operat. Res. Soc.*, 46: 1498-1504. DOI: 10.1057/jors.1994.209