Project Scheduling: Improved Approach to Incorporate Uncertainty Using Bayesian Networks

VAHID KHODAKARAMI, Queen Mary University of London, United Kingdom
NORMAN FENTON, Queen Mary University of London, United Kingdom
MARTIN NEIL, Queen Mary University of London, United Kingdom

ABSTRACT

Project scheduling inevitably involves uncertainty. The basic inputs (i.e., time, cost, and resources for each activity) are not deterministic and are affected by various sources of uncertainty. Moreover, there is a causal relationship between these uncertainty sources and project parameters; this causality is not modeled in current state-of-the-art project planning techniques (such as simulation techniques). This paper introduces an approach, using Bayesian network modeling, that addresses both uncertainty and causality in project scheduling. Bayesian networks have been widely used in a range of decision-support applications, but the application to project management is novel. The model presented empowers the traditional critical path method (CPM) to handle uncertainty and also provides explanatory analysis to elicit, represent, and manage different sources of uncertainty in project planning.

Keywords: project scheduling; uncertainty; Bayesian networks; critical path method; CPM

Vol. 38, No. 2, 39-49, ISSN 8756-9728/03

Introduction

Project scheduling is difficult because it inevitably involves uncertainty. Uncertainty in real-world projects arises from the following characteristics:

- *Uniqueness* (no similar experience)
- *Variability* (trade-off between performance measures like time, cost, and quality)
- *Ambiguity* (lack of clarity, lack of data, lack of structure, and bias in estimates).

Many different techniques and tools have been developed to support better project scheduling, and these tools are used seriously by a large majority of project managers (Fox & Spence, 1998; Pollack-Johnson, 1998). Yet, quantifying uncertainty is rarely prominent in these approaches.

This paper focuses especially on the problem of handling uncertainty in project scheduling. The next section elaborates on the nature of uncertainty in project scheduling and summarizes the current state of the art. The proposed approach is to adapt one of the best-used scheduling techniques, critical path method (CPM) (Kelly, 1961), and incorporate it into an explicit uncertainty model (using Bayesian networks). The paper summarizes the basic CPM methodology and notation, presents a brief introduction to Bayesian networks, and describes how the CPM approach can be incorporated (using a simple illustrative example). Also discussed is a mechanism to implement the model in real-world projects, and suggestions on how to move forward and possible future modifications are presented.

The Nature of Uncertainty in Project Scheduling

A Guide to the Project Management Body of Knowledge (PMBOK® Guide)—Third edition (PMI, 2004) identifies risk management as a key area of project management:

> "Project risk management includes the processes concerned with conducting risk management planning, identification, analysis, response, and monitoring and control on a project."

Central to risk management is the issue of handling *uncertainty.* Ward and Chapman (2003) argued that current project risk management processes induce a restricted focus on managing project uncertainty. They believe it is because the term "risk" has become associated with "events" rather than more general sources of significant uncertainty.

In different project management processes there are different aspects of uncertainty. The focus of this paper is on uncertainty in project scheduling. The most obvious area of uncertainty here is in estimating duration for a particular activity. Difficulty in this estimation can arise from a lack of knowledge of what is involved as well as from the uncertain consequences of potential threats or opportunities. This uncertainty arises from one or more of the following:

- Level of available and required resources
- Trade-off between resources and time
- Possible occurrence of uncertain events (i.e., risks)
- Causal factors and interdependencies including common casual factors that affect more than one activity (such as organizational issues)
- Lack of previous experience and use of subjective rather than objective data
- Incomplete or imprecise data or lack of data at all
- Uncertainty about the basis of subjective estimation (i.e., bias in estimation).

The best-known technique to support project scheduling is CPM. This technique, which is adapted by the most widely used project management software tools, is purely deterministic. It makes no attempt to handle or quantify uncertainty. However, a number of techniques, such as program evaluation and review technique (PERT), critical chain scheduling (CCS) and Monte Carlo simulation (MCS), do try to handle uncertainty, as follows:

- PERT (Malcom, Roseboom, Clark, & Fazer, 1959; Miller, 1962; Moder, 1988) incorporates uncertainty in a restricted sense by using a probability distribution for each task. Instead of having a single deterministic value, three different estimates (pessimistic, optimistic, and most likely) are approximated. Then the "critical path" and the start and finish date are calculated by the use of distributions' means and applying probability rules. Results in PERT are more realistic than CPM, but PERT does not address explicitly any of the sources of uncertainty previously listed.
- Critical chain (CC) scheduling is based on Goldratt's theory of constraints (Goldratt, 1997). For minimizing the impact of Parkinson's Law (jobs expand to fill the allocated time), CC uses a 50% confidence interval for each task in project scheduling. The safety time (remaining 50%) associated with each task is shifted to the end of the critical chain (longest chain) to form the project buffer. Although it is claimed that the CC approach is the most important breakthrough in project management history, its oversimplicity is a concern for many companies that do not understand both the strength and weakness of CC and apply it regardless of their particular and unique circumstances (Pinto, 1999). The assumption that all task durations are overestimated by a certain factor is questionable. The main issue is: How does the project manager determine the safety time? (Raz, Barnes, & Dvir, 2003). CC relies on a fixed, right-skewed probability for activities, which may be inappropriate (Herroelen & Leus, 2001), and a sound estimation of project and activity duration (and consequently the buffer size) is still essential (Trietsch, 2005).
- Monte Carlo simulation (MCS) was first proposed for project scheduling in the early 1960s (Van Slyke, 1963) and implemented in the 1980s (Fishman, 1986). In the 1990s, because of improvements in computer technology, MCS rapidly became the dominant technique for handling uncertainty in project scheduling (Cook, 2001). A survey by the Project Management Institute (PMI, 1999) showed that nearly 20% of project management software packages support MCS. For example, PertMaster (PertMaster, 2006) accepts scheduling data from tools like MS-Project and Primavera and incorporates MCS to provide project risk analysis in time and cost. However, the Monte Carlo approach has attracted some criticism. Van Dorp and Duffey (1999) explained the weakness of Monte Carlo simulation in assuming statistical independence of activity duration in a project network. Moreover, being event-oriented (assuming project risks as "independent events"), MCS and the tools that implement it do not identify the sources of uncertainty.

As argued by Ward and Chapman (2003), managing uncertainty in projects is not just about managing perceived threats, opportunities, and their implication. A proper uncertainty management provides for identifying various sources of uncertainty, understanding the origins of them, and then managing them to deal with desirable or undesirable implications.

Capturing uncertainty in projects "needs to go beyond variability and available data. It needs to address ambiguity and incorporate structure and knowledge" (Chapman & Ward, 2000). In order to measure and analyze uncertainty properly, we need to model relations between trigger (source), and risk and impacts (consequences). Because projects are usually one-off experiences, their uncertainty is *epistemic* (i.e., related to a lack of complete knowledge) rather than *aleatoric* (i.e., related to randomness). The duration of a task is uncertain because there is no similar experience before, so data is incomplete and suffers from imprecision and inaccuracy. The estimation of this sort of uncertainty is mostly subjective and based on estimator judgment. Any estimation is conditionally dependent on some assumptions and conditions—even if they are not mentioned explicitly. These assumptions and conditions are major sources of uncertainty and need to be addressed and handled explicitly.

The most well-established approach to handling uncertainty in these circumstances is the Bayesian approach (Efron, 2004; Goldstein, 2006). Where complex causal relationships are involved, the Bayesian approach is extended by using Bayesian networks. The challenge is to incorporate the CPM approach into Bayesian networks.

CPM Methodology and Notation

CPM (Moder, 1988) is a deterministic technique that, by use of a network of dependencies between tasks and given deterministic values for task durations, calculates the longest path in the network called the "critical path." The length of the "critical path" is the earliest time for project completion. The critical path can be identified by determining the following parameters for each activity:

D—duration
ES—earliest start time
EF—earliest finish time
LS—latest start time
LF—latest finish time.

The earliest start and finish times of each activity are determined by working forward through the network and determining the earliest time at which an activity can start and finish, considering its predecessor activities. For each activity j:

$ES_j = \text{Max}\ [ES_i + D_i\ ;$ over predecessor activities $i]$
$EF_j = ES_j + D_j$

The latest start and finish times are the latest times that an activity can start and finish without delaying the project and are found by working backward through the network. For each activity i:

$LF_i = \text{Min}\ [LF_j - D_j\ ;$ over successor activities $j]$
$LS_i = LF_i - D_i$

The activity's "total float" (TF) (i.e., the amount that the activity's duration can be increased without increasing the overall project completion time) is the difference in the latest and earliest finish times of each activity. A critical activity is one with no TF and should receive special attention (delay in a critical activity will delay the entire project). The critical path then is the path(s) through the network whose activities have minimal TF.

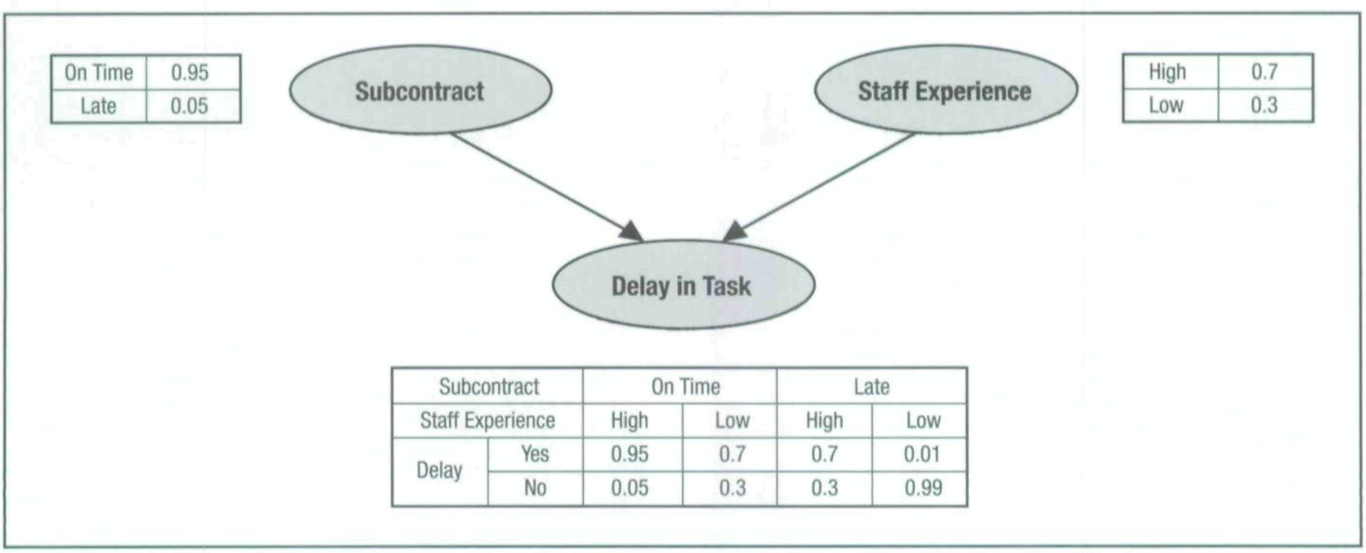
The CPM approach is very simple and provides very useful and fundamental information about a project and its activities' schedule. However, because of its single-point estimate assumption, it is too simplistic to be used in complex projects. The challenge is to incorporate the inevitable uncertainty.

Proposed BN Solution

Bayesian Networks (BNs) are recognized as a mature formalism for handling causality and uncertainty (Heckerman, Mamdani, & Wellman, 1995). This section provides a brief overview of BNs and describes a new approach for scheduling project activities in which CPM parameters (i.e., ES, EF, LS, and LF) are determined in a BN.

Bayesian Networks: An Overview

Bayesian networks (also known as belief networks, causal probabilistic networks, causal nets, graphical probability networks, probabilistic cause-effect models, and probabilistic influence diagrams) provide decision support for a wide range of problems involving uncertainty and probabilistic reasoning. Examples of real-world applications can be found in Heckerman et al. (1995), Fenton, Krause, and Neil (2002), and Neil, Fenton, Forey, and Harris (2001). A BN is a directed graph, together with an associated set of probability tables. The graph consists of nodes and arcs. Figure 1 shows a simple BN that models the cause of delay in a particular task in a project. The nodes represent uncertain variables, which may or may not be observable. Each node has a set of states (e.g. "on time" and "late" for "Subcontract" node). The arcs represent causal or influential relationships between variables. (e.g., "subcontract" and "staff experience" may cause a "delay in task"). There is a probability table for each node, providing the probabilities of each state of the variable. For variables without parents (called "prior" nodes), the table just contains the marginal probabilities (e.g., for the subcontract" node P(on-time)=0.95 and P(late)=0.05). This is also called "prior distribution" that represents the prior belief (state of knowledge) about the variable. For each variable with parents, the probability table has conditional probabilities for each combination of the parents' states (see, for example, the probability table for a "delay in task"



On Time	0.95
Late	0.05

High	0.7
Low	0.3

Subcontract		On Time		Late	
Staff Experience		High	Low	High	Low
Delay	Yes	0.95	0.7	0.7	0.01
	No	0.05	0.3	0.3	0.99

Figure 1: A Bayesian network contains nodes, arcs and probability table

in Figure 1). This is also called the "likelihood function" that represents the likelihood of a state of a variable given a particular state of its parent.

The main use of BNs is in situations that require statistical inference. In addition to statements about the probabilities of events, users have some *evidence* (i.e., some variable states or events that have actually been observed), and can infer the probabilities of other variables, which have not as yet been observed. These observed values represent a posterior probability, and by applying Bayesean rules in each affected node, users can influence other BN nodes via propagation, modifying the probability distributions. For example, the probability that the task finishes on time, with no observation, is 0.855 (see Figure 2a). However if we know that the subcontractor failed to deliver on time, this probability updates to 0.49 (see Figure 2b).

The key benefits of BNs that make them highly suitable for the project planning domain are that they:

- Explicitly quantify uncertainty and model the causal relation between variables
- Enable reasoning from effect to cause as well as from cause to effect (propagation is both "forward" and "backward")
- Make it possible to overturn previous beliefs in the light of new data
- Make predictions with incomplete data
- Combine subjective and objective data
- Enable users to arrive at decisions that are based on visible auditable reasoning.

BNs, as a tool for decision support, have been deployed in domains ranging from medicine to politics. BNs potentially address many of the "uncertainty" issues previously discussed. In particular, incorporating CPM-style scheduling into a BN framework makes it possible to properly handle uncertainty in project scheduling.

There are numerous commercial tools that enable users to build BN models and run the propagation calculations. With such tools it is possible to perform fast propagation in large BNs (with hundreds of nodes). In this paper, AgenaRisk (2006) was used, since it can model continuous variables (as opposed to just discrete).

BN for Activity Duration

Figure 3 shows a prototype BN that the authors have built to model uncertainty sources and their affects on duration of a particular activity. The model contains variables that capture the uncertain nature of activity duration. "Initial duration estimation" is the first estimation of the activity's duration; it is estimated based on historical data, previous experience, or simply expert judgment. "Resources" incorporate any affecting factor that can increase or decrease the activity duration. It is a ranked node, which for simplicity here is restricted to three levels: low, average, and high. The level of resources can be inferred from so-called "indicator" nodes. Hence, the causal link is from the "resources" directly to observable indicator values like the "cost," the experience of available "people" and the level of available "technology." There are many alternative indicators. An important and novel aspect of this approach is to allow the model to be adapted to use whichever indicators are available.

The power of this model is better understood by showing the results of running it under various scenarios. It is possible to enter observations anywhere in the model to perform not just predictions but also many types of trade-off and explanatory analysis. So, for example, observations for the initial duration estimation and resources can be entered and the model will show the distributions for duration. Figure 4 shows how the distribution of the activity duration in which the initial estimation is five days changes when the level of its available resources goes from low to high. (All the subsequent figures are outputs from the AgenaRisk software.)

Another possible analysis in this model is the trade-off analysis between duration and resources when there is a time constraint for activity duration and it is interesting to know about the level of required resource. For example, consider an activity in which the initial duration is estimated as five days but must be finished in three days. Figure 5 shows the probability distribution of required resources to meet this duration constraint. Note how it is skewed toward high.

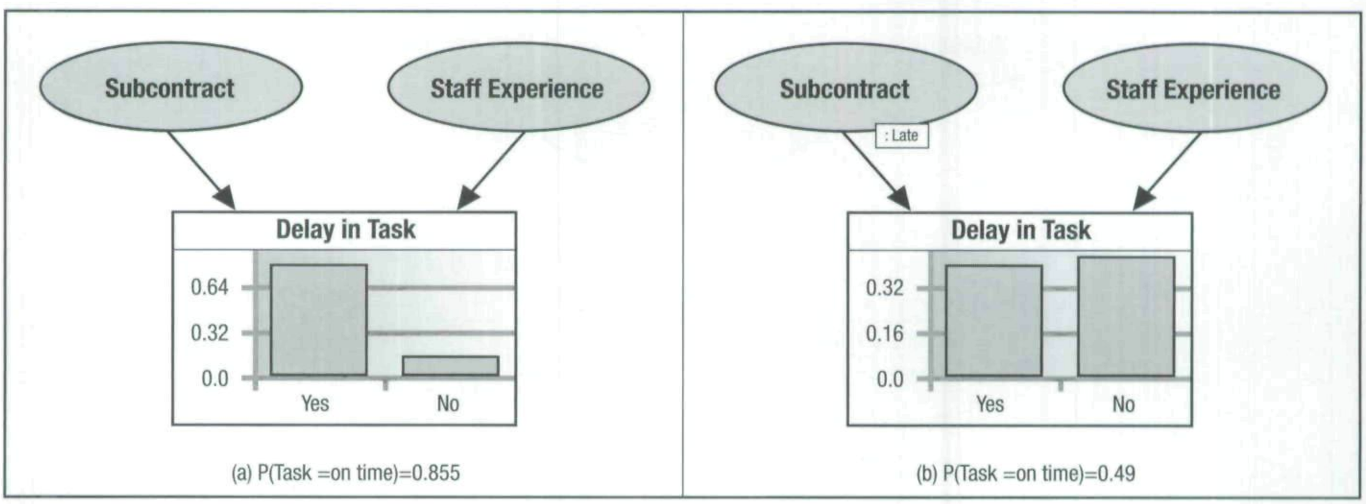


Figure 2: New evidence updates the probability

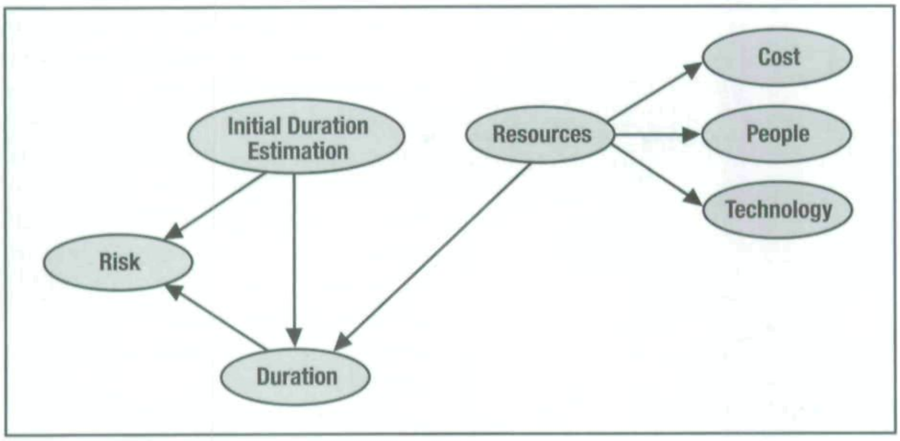


Figure 3: Bayesian network for activity duration

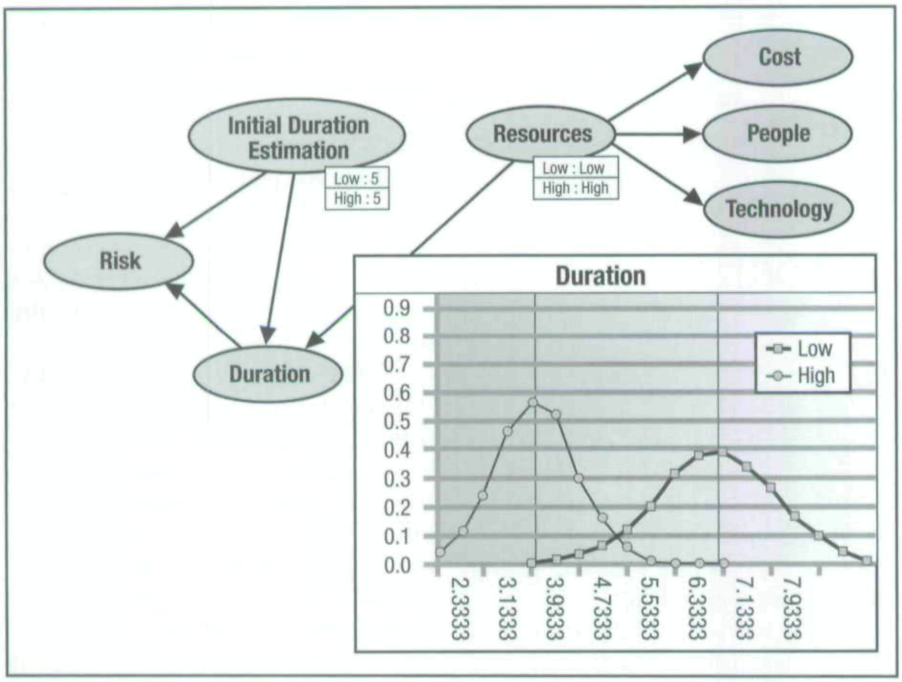


Figure 4: Probability distribution for "duration" (days) changes when the level of "resources" changes

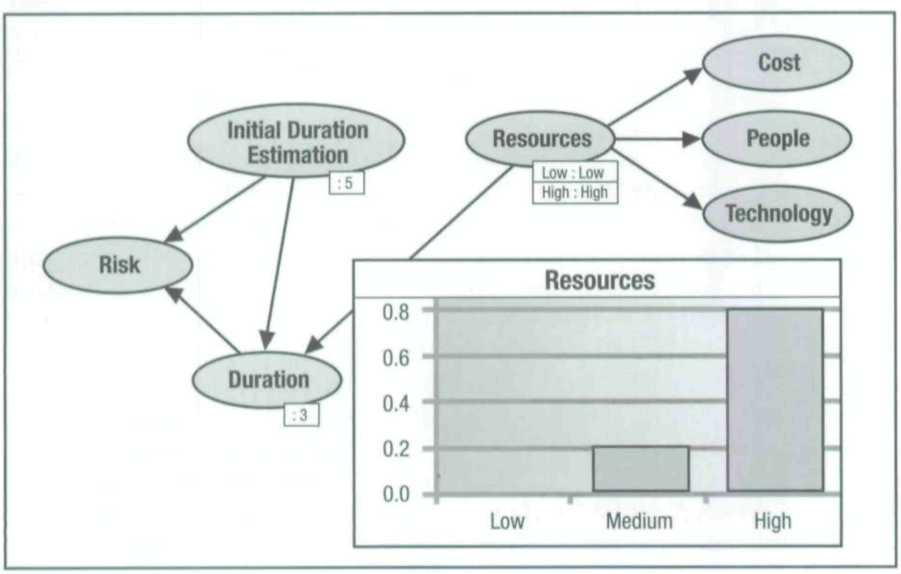


Figure 5: Level of required "Resources" when there is a constraint on "Duration"

Mapping CPM to BN

The main components of CPM networks are *activities*. Activities are linked together to represent dependencies. In order to map a CPM network to a BN, it is necessary to first map a single activity. Each of the activity parameters are represented as a variable (node) in the BN.

Figure 6 shows a schematic model of the BN fragment associated with an activity. It clearly shows the relation between the activity parameters and also the relation with predecessor and successor activities.

The next step is to define the connecting link between dependent activities. The forward pass in CPM is mapped as a link between the EF of each activity to the ES of the successor activities. The backward-pass in CPM is mapped as a link between the LS of each activity to the LF of the predecessor activities.

Example

The following illustrates this mapping process. The example is deliberately very simple to avoid extra complexity in the BN. How the approach can be used in real-size projects is discussed later in the paper.

Consider a small project with five activities—A, B, C, D, and E. The activity on arc (AOA) network of the project is shown in Figure 7.

The results of the CPM calculation are summarized in Table 1. Activities A, C, and E with TF=0 are critical and the overall project takes 20 days (i.e., earliest finish of activity E).

Figure 8 shows the full BN representation of the previous example. Each activity has five associated nodes. Forward pass calculation of CPM is done through the connection between the ES and EF. Activity A, the first activity of the project, has no predecessor, so its ES is set to zero. Activity A is predecessor for activities B and C so the EF of activity A is linked to the ES of activities B and C. The EF of activity B is linked to the ES of its successor, activity D. And finally, the EF of activities C and D are connected to the ES of activity E. In fact, the ES of activity E is the maximum of the EF of activities C

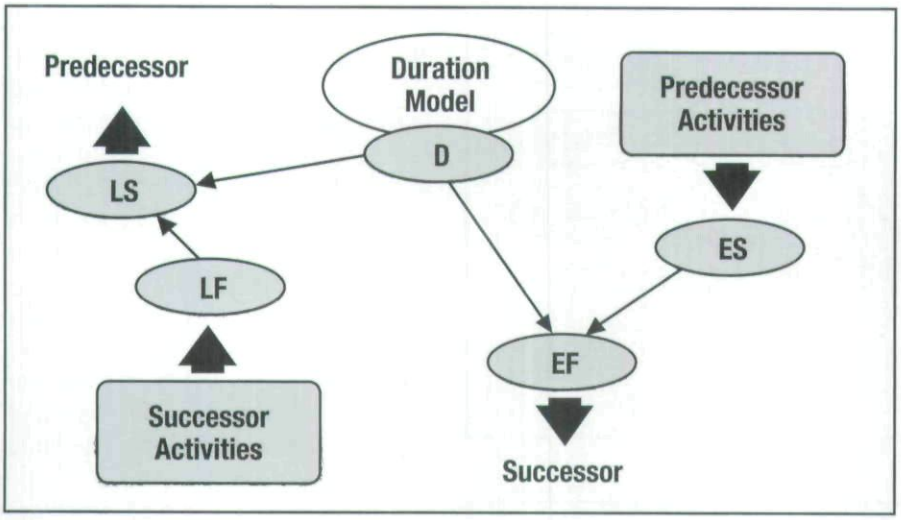


Figure 6: Schematic of BN for an activity

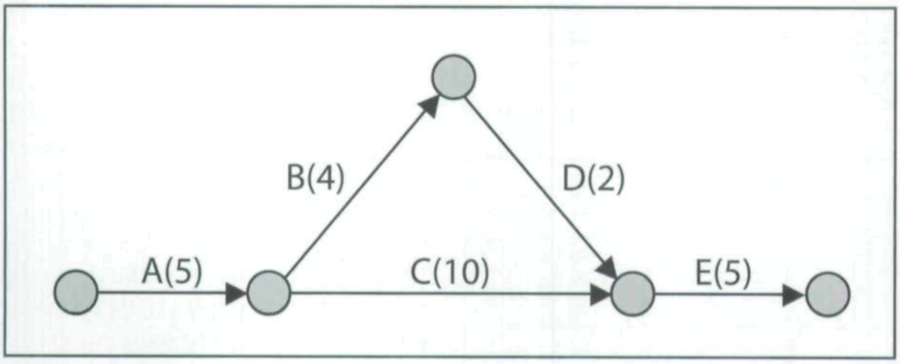


Figure 7: CPM network

and D. The EF of activity E is the earliest time for project completion time.

The same approach is used for backward CPM calculations connecting the LF and LS. Activity E is the last activity of the project and has no successor, so its LF is set to EF. Activity E is successor of activities C and D so the LS of activity E is linked to the LF of activities C and D. The LS of activity D is linked to the LF of its predecessor activity B. And finally, the LS of activities B and C are linked to the LF of activity A. The LF of activity A is the minimum of the LS of activities B and C.

For simplicity in this example, it is assumed that activities A and E are more risky and need more detailed analysis. For all other activities the uncertainty about duration is expressed simply by a normal distribution.

Results

This section explores different scenarios of the BN model in Figure 8. The main objective is to predict the project completion time (i.e., the earliest finish of E) in such a way that it fully characterizes uncertainty.

Suppose the initial estimation of activities' duration is the same as in Table 1. Suppose the resource level for activities A and E is medium. If the earliest start of activity A is set to zero, the distribution for project completion is shown in Figure 9a. The distribution's mean is 20 days as was expected from the CPM analysis. However, unlike CPM, the prediction is not a single point and its variance is 4. Figure 9b illustrates the cumulative distribution of finishing time, which shows the probability of completing the project before a given time. For example, with a probability of 90% the project will finish in 22 days.

In addition to this baseline scenario, by entering various evidence (observations) to the model, it is possible to analyze the project schedule from different aspects. For example, one scenario is to see how changing the resource level affects the project completion time.

Figure 10 compares the distributions for project completion time as the level of people's experience changes. When people's experience changes from low to high, the mean of finishing time changes from 22.7 days to 19.5 days and the 90% confidence interval changes from 26.3 days to 22.9 days.

Another useful analysis is when there is a constraint on the project completion time and we want to know how many resources are needed. Figure 11 illustrates this trade-off between project time and required resources. If the project needs to be completed in 18 days (instead of the baseline 20 days) then the resource required for activity A most likely must be high; if the project completion is set to 22, the resource level for activity A moves significantly in the direction of low.

The next scenario investigates the impact of risk in activity A on the project completion time as it is shown in Figure 12. When there is a risk in activity A, the mean of distribution for the project completion time changes from 19.9 days to 22.6 days and the 90% confidence interval changes from 22.5 days to 25.3 days.

One important advantage of BNs is their potential for parameter learning, which is shown in the next scenario. Imagine activity A actually finishes in seven days, even though it was originally estimated as five days. Because activity A has taken more time than was expected, the level of resources has probably not been sufficient.

By entering this observation the model gives the resource probability for activity A as illustrated in Figure 13. This can update the analyst's belief about the actual level of available resources.

Assuming both activities A and E use the same resources (e.g., people), the updated knowledge about the level of available resources from activity A (which is finished) can be entered as evidence in the resources

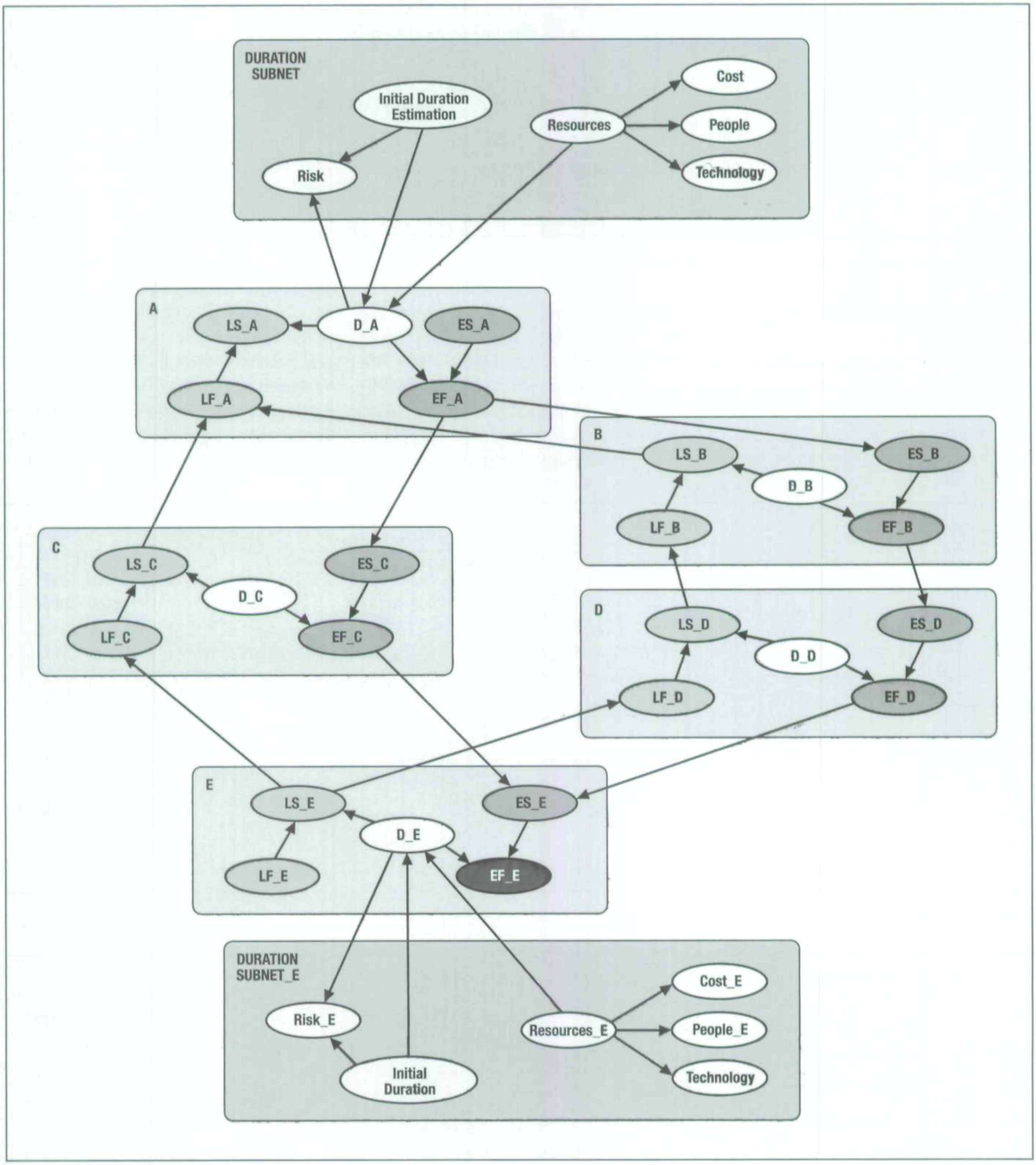


Figure 8: Overview of BN for example (1)

for activity E (which is not started yet) and consequently updates the project completion time. Figure 14 shows the distributions of completion time when the level of available resource of activity E is learned from the actual duration of activity A.

Another application of parameter learning in these models is the ability to incorporate and learn about bias in estimation. So, if there are several observations in which actual task completion times are underestimated, the model learns that this may be due to bias rather than unforeseen risks, and this information will inform subsequent predictions. Work on this type of application (called dynamic learning), is still in progress and can be a possible way of extending the BN version of CPM.

Activity	D	ES	EF	LS	LF	TF
A	5	0	5	0	5	0
B	4	5	9	9	13	4
C	10	5	15	5	15	0
D	2	9	11	13	15	4
E	5	15	20	15	20	0

Table 1: Activities' time (days) and summary of CPM calculations

Object-Oriented Bayesian Network (OOBN)

It is clear from Figure 8 that even simple CPM networks lead to fairly large BNs. In real-sized projects with several activities, constructing the network needs a huge effort, which is not effective especially for users without much experience in BNs. However, this complexity can be handled using the so-called object-oriented Bayesian network (OOBN) approach (Koller & Pfeffer, 1997). This approach, analogous to the object-oriented programming languages, supports a natural framework for abstraction and refinement, which allows complex domains to be described in terms of interrelated objects.

The basic element in OOBN is an object; an entity with an identity, state, and behavior. An object has a set of attributes each of which is an object. Each object is assigned to a class. Classes provide the ability to describe a general, reusable network that can be used in different instances. A class in OOBN is a BN fragment.

The proposed model has a highly repetitive structure and fits the object-oriented framework perfectly. The internal parts of the activity subnet (see Figure 6) are encapsulated within the activity class as shown in Figure 15.

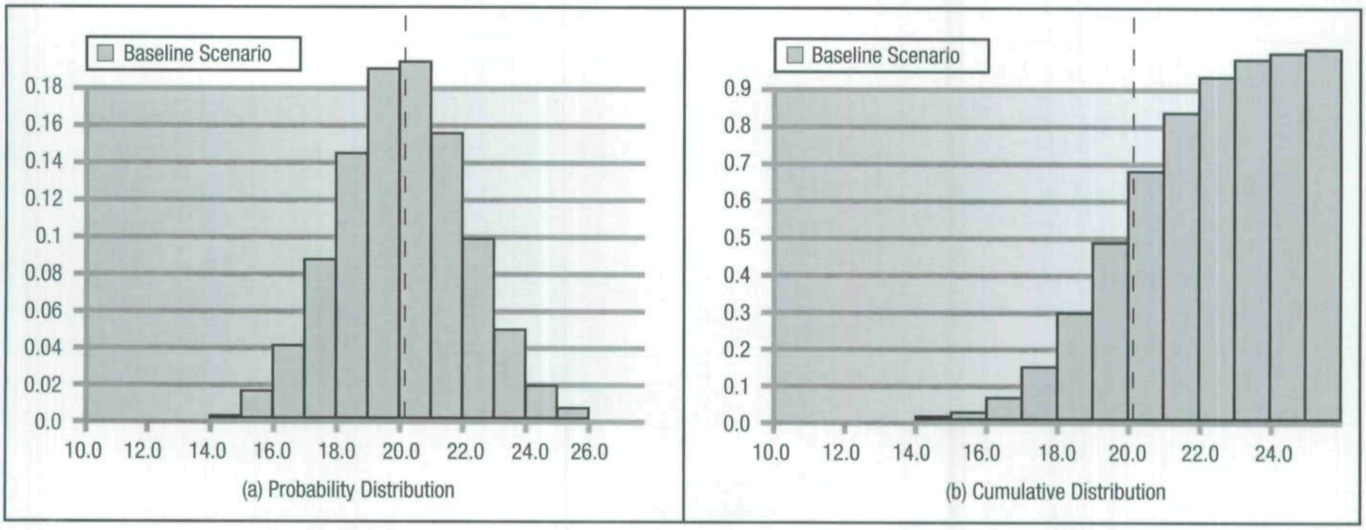


Figure 9: Distribution of project completion (days) for main scenario in example (1)

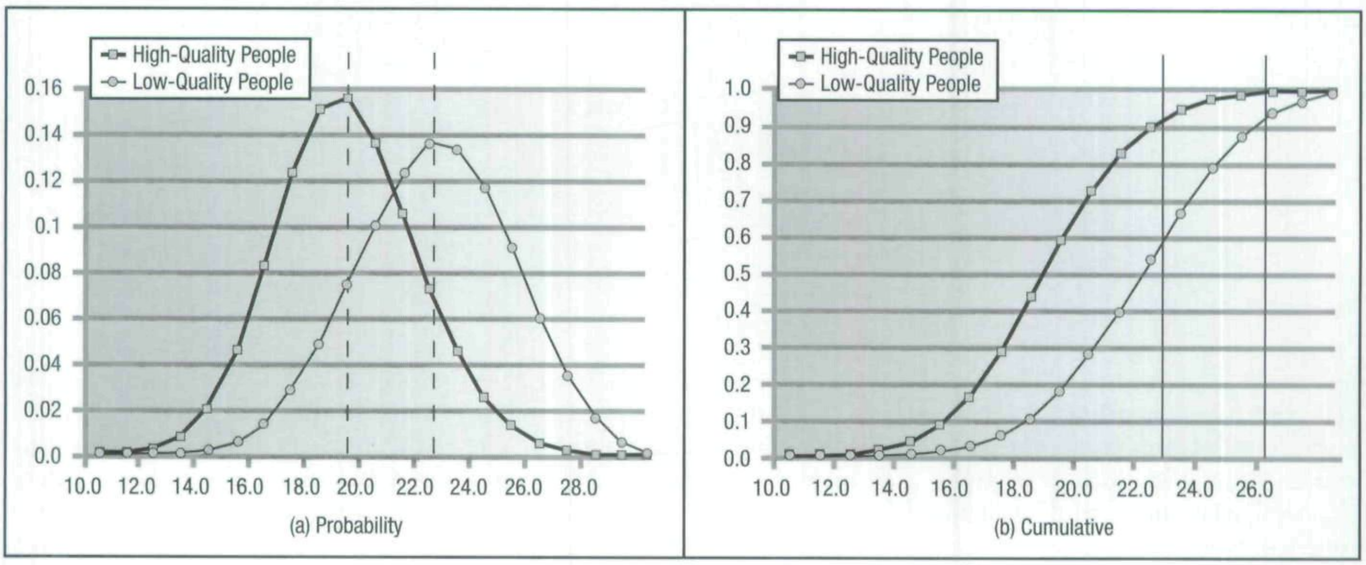


Figure 10: Change in project time distribution (days) when level of people's experience changes

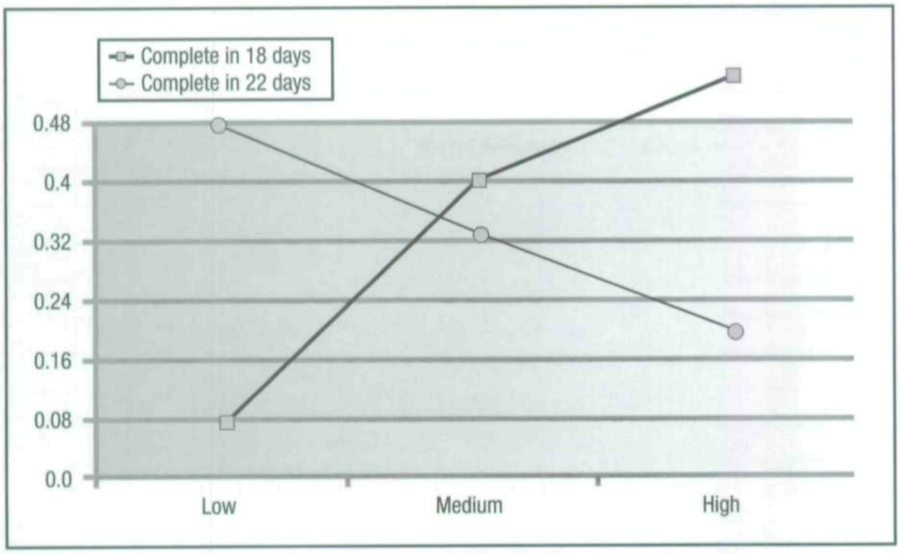


Figure 11: Probability of required resource changes when the time constraint changes

Classes can be used as libraries and combined into a model as needed. By connecting interrelated objects, complex networks with several dozen nodes can be constructed easily. Figure 16 shows the OOBN model for the example previously presented.

The OOBN approach can also significantly improve the performance of inference in the model. Although a full discussion of the OOBN approach to this particular problem is beyond the scope of this paper, the key point to note is that there is an existing mechanism (and implementation of it) that enables the proposed solution to be genuinely "scaled-up" to real-world projects. Moreover, research is emerging to develop the new generation of BNs tools and algorithms that support OOBN concept both in constructing large-scale models and also in propagation aspects.

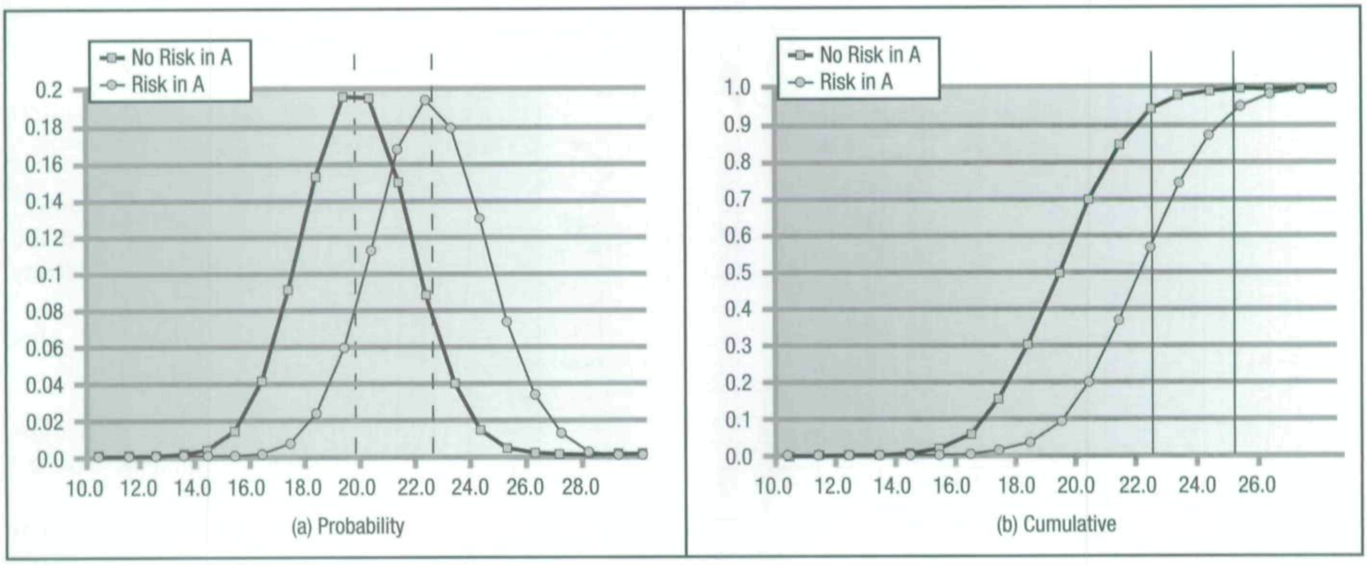


Figure 12: The impact of occurring risk in activity A on the project completion time

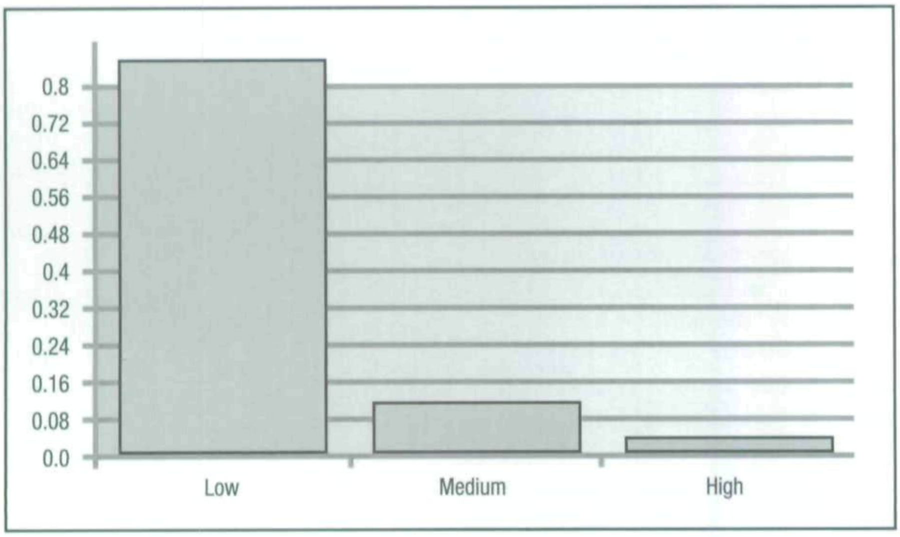


Figure 13: Learnt probability distribution "resource" when the actual duration is seven days

Conclusions and How to Move Forward

Handling risk and uncertainty is increasingly seen as a crucial component of project management and planning. One classic problem is how to incorporate uncertainty in project scheduling. Despite the availability of different approaches and tools, the dilemma is still challenging. Most current techniques for handling risk and uncertainty in project scheduling (simulation-based techniques) are often

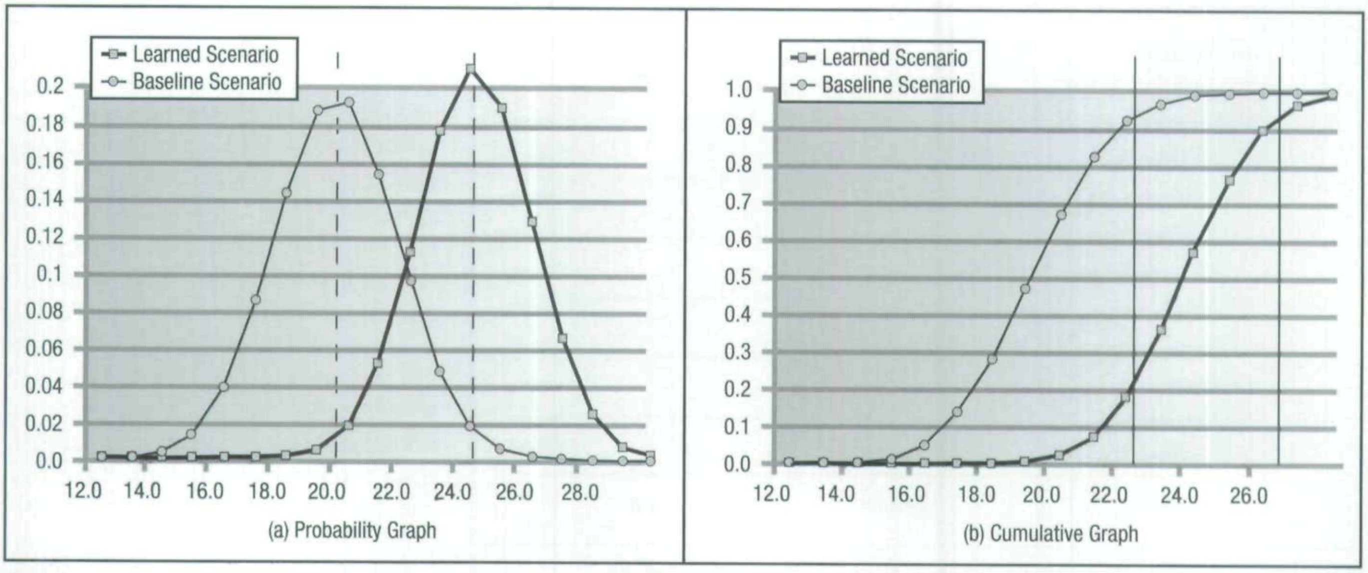


Figure 14: completion time (days) based on learned parameters compare with baseline scenario

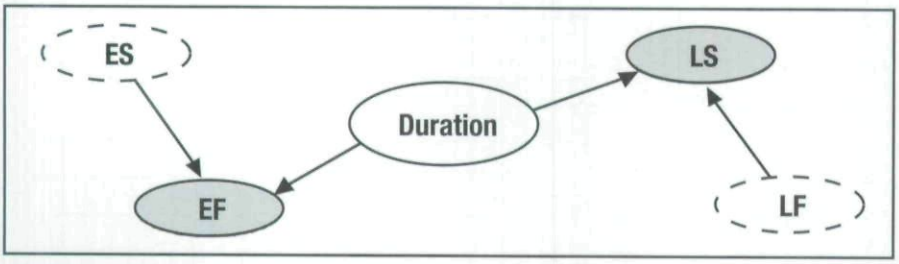


Figure 15: Activity class encapsulates internal parts of network

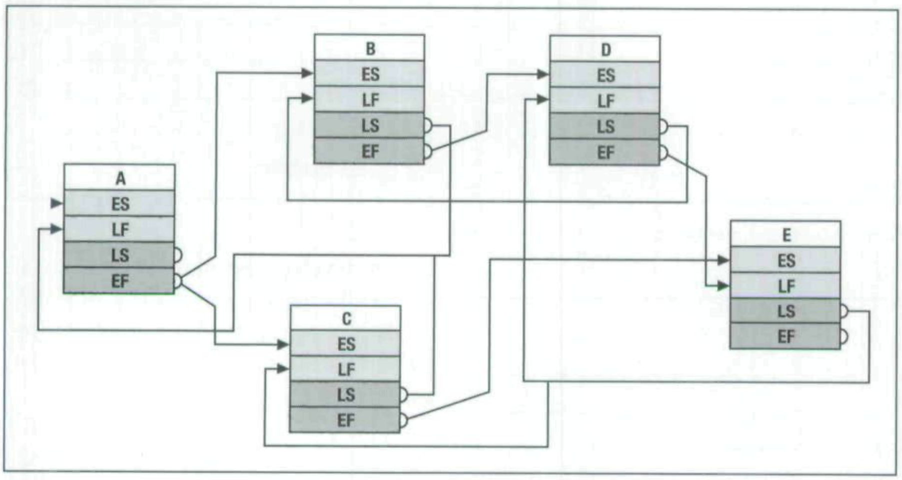


Figure 15: OO model for the presented example

event-oriented and try to model the impact of possible "threats" on project performance. They ignore the source of uncertainty and the causal relations between project parameters. More advanced techniques are required to capture different aspects of uncertainty in projects.

This paper has proposed a new approach that makes it possible to incorporate risk, uncertainty, and causality in project scheduling. Specifically, the authors have shown how a Bayesian network model can be generated from a project's CPM network. Part of this process is automatic and part involves identifying specific risks (which may be common to many activities) and resource indicators. The approach brings the full weight and power of BN analysis to bear on the problem of project scheduling. This makes it possible to:

- Capture different sources of uncertainty and use them to inform project scheduling
- Express uncertainty about completion time for each activity and the whole project with full probability distributions
- Model the trade-off between time and resources in project activities
- Use "what-if?" analysis
- Learn from data so that predictions become more relevant and accurate.

The application of the approach was explained by use of a simple example. In order to upscale this to real projects with many activities the approach must be extended to use the so-called object-oriented BNs. There is ongoing work to accommodate such object-oriented modeling so that building a BN version of a CPM is just as simple as building a basic CPM model.

Other extensions to the work described here include:

- Incorporating additional uncertainty sources in the duration network
- Handling dynamic parameter learning as more information becomes available when the project progresses
- Handling common causal risks that affect more than one activity
- Handling management action when the project is behind its plan.

References

AgenaRisk. (2006). Available at www.agenarisk.com

Chapman, C., & Ward, S. (2000). Estimation and evaluation of uncertainty: A minimalist first pass approach. *International Journal of Project Management, 18*, 369–383.

Cook, M. S. (2001). Real-world Monte Carlo analysis. *Proceeding of PMI Annual Seminars and Symposium*, Nashville, TN.

Efron, B. (2004). Bayesians, frequentists, and scientists. *Journal of the American Statistical Association, 100*(469), 1–5.

Fenton, N. E., Krause, P., & Neil, M. (2002). Software measurement: Uncertainty and causal modeling. *IEEE Software 10*(4), 116–122.

Fishman, G. S. (1986). A Monte Carlo sampling plan for estimating network reliability. *Operations Research, 34*(4), 581–594.

Fox, T. L., & Spence, J. W. (1998). Tools of the trade: A survey of project management tools. *Project Management Journal, 29*, 20–28.

Goldstein, M. (2006). Subjective Bayesian analysis: Principle and practice2. *Bayesian Analysis, 1*(3), 403–420.

Goldratt, E. M. (1997). *Critical chain*. Great Barrington: The North River Press Publishing Corporation.

Heckerman, D., Mamdani, A., & Wellman, M. (1995). Real-world applications of Bayesian networks. *Comm ACM, 38*(3), 25–26.

Herroelen, W., & Leus, R. (2001). On the merits and pitfalls of critical chain scheduling. *Journal of Operations Management, 19*, 559–577.

Kelly, J. E. (1961). Critical path planning and scheduling mathematical bases. *Operations Research, 9*, 246–320.

Koller, D., & Pfeffer, A. (1997). Object-oriented Bayesian networks. In D. Geiger & P. Shenoy, (Eds.), *Proceedings of the Thirteenth annual Conference on Uncertainty in Artificial Intelligence* (UAI-97) (pp. 302–313), San Francisco, CA.

Malcolm, D. G., Roseboom, J. H., Clark, C. E., & Fazer, W. (1959). Application of a technique for research and development program evaluation. *Operations Research, 7*(5), 646–669.

Miller R. W. (1962). How to plan and control with PERT. *Harvard Business Review*, 93–104.

Moder, J. (1988). Network techniques in project management. *Project Management Handbook*, 2nd edition (pp. 324–373). New York: Van Nostrand Reinhold.

Neil, M., Fenton, N., Forey, S. & Harris, R. (2001). Using Bayesian belief networks to predict the reliability of military vehicles. *IEE Computing and Control Engineering Journal 12*(1), 11–20.

Pinto, J. K. (1999). Some constraints on the theory of constraints: Taking a critical look at the critical chain. *PM Network, 13*(8), 49–51.

Pertmaster. (2006). *Project risk software*. Available at www.pertmaster.com

PMI. (1999). Project management software survey, Newtown Square, PA: Project Management Institute.

PMI. (2004). *A guide to the project management body of knowledge (PMBOK® guide* – 3rd edition). Newtown Square, PA: Project Management Institute.

Pollack-Johnson, B. (1998). Project management software usage patterns and suggested research directions for future development. *Project Management Journal, 29*, 19–29.

Raz, T., Barnes, R., & Dvir, D. (2003). A critical look at critical chain project management. *Project Management Journal, 34*, 24–32.

Trietsch, D. (2005). Why critical path by any other name would smell less sweet? *Project Management Journal, 36*, 27–36.

Van Dorp, J. R., & Duffey, M. R. (1999). Modelling statistical dependence in risk analysis for project networks. *International Journal of Production Economics, 58*, 17–29.

Van Slyke, R. M. (1963). Monte Carlo methods and the PERT problem. *Operations Research, 11*(5), 839–860.

Ward, S., & Chapman, C. (2003). Transforming project risk management into project uncertainty management. *International Journal of Project Management, 21*, 97–105.

VAHID KHODAKARAMI is a PhD student at RADAR group at Queen Mary University of London. He earned a BSc in industrial engineering from Tehran Polytechnic and an MSc in industrial engineering from Sharif University of Technology in Iran. He has more than 10 years experience in both academia and industry. He has also consulted for several companies in project management and system design. He earned his second MSc in information technology from Queen Mary. His research interests include project management, project risk management, decision-making and Bayesian networks.

NORMAN FENTON is a professor of computing at Queen Mary (London University) and is also chief executive officer of Agena, a company that specializes in risk management for critical systems. At Queen Mary he is the computer science department director of research and he is the head of the Risk Assessment and Decision Analysis Research Group (RADAR). His books and publications on software metrics, formal methods, and risk analysis are widely known in the software engineering community. His recent work has focused on causal models (Bayesian nets) for risk assessment in a wide range of application domains such as vehicle reliability, embedded software, transport systems, TV personalization and financial services. He is a chartered engineer and chartered mathematician and is a fellow of the British Computer Society. He is a member of the editorial board of the *Software Quality Journal*.

MARTIN NEIL is a reader in "systems risk" at the Department of Computer Science, Queen Mary, University of London, where he teaches decision and risk analysis and software engineering. He is also a joint founder and chief technology officer of Agena Ltd., which developed and distributes AgenaRisk, a software product for modeling risk and uncertainty. His interests cover Bayesian modeling and/or risk quantification in diverse areas: operational risk in finance, systems and design reliability, project risk, decision support, simulation, artificial intelligence and personalization, and statistical learning. He earned a BSc in mathematics, a PhD in statistics and software metrics and is a chartered engineer.