

Quantifying Success Factors for IT Projects—An Expert-Based Bayesian Model

Liv Gingnell¹, Ulrik Franke^{1,2}, Robert Lagerström¹, Evelina Ericsson¹, and Joakim Lilliesköld¹

¹Industrial Information and Control Systems, KTH Royal Institute of Technology, Stockholm, Sweden

²Swedish Defence Research Agency (FOI), Stockholm, Sweden

Large investments are made annually to develop and maintain IT systems. Successful outcome of IT projects is therefore crucial for the economy. Yet, many IT projects fail completely or are delayed or over budget, or they end up with less functionality than planned. This article describes a Bayesian decision-support model. The model is based on expert elicited data from 51 experts. Using this model, the effect management decisions have upon projects can be estimated beforehand, thus providing decision support for the improvement of IT project performance.

Keywords IT project success factors; decision support; Bayesian networks; Noisy-OR; expert elicitation

1. INTRODUCTION

As the operations of companies and organizations become increasingly automated and computerized, these companies also become increasingly dependent on their, often complex, IT systems. Huge amounts of money are invested in IT projects aiming to develop, improve, and maintain these systems. According to the U.S. Department of Commerce, U.S. investments in IT (Information processing equipment and software) alone went from slightly below \$200 billion in 1990 to slightly above \$400 billion in 2003 (nominal figures; U.S. Department of Commerce, 2003). The trend is similar in the rest of the industrialized world. The ability to successfully manage IT projects is thus as important as ever for companies in any line of business.

In spite of the importance of IT projects, The Standish Group establishes in the latest version of their CHAOS report that the share of IT projects that exceed time and budget, or that fail to fulfill basic systems requirements, continues to be high (The Standish Group, 2009). The scientific rigor of the research performed by The Standish Group has been questioned by some

(e.g., Glass, 2006), as they are unclear about how the projects they audit are selected. Nevertheless, the fact remains that a disturbing number of IT projects actually do fail. The cost of project failure across the European Union was, for instance, 142 billion Euros in 2004 (McManus & Wood-Harper, 2007). Considering the importance of IT systems, this seems serious enough, and many have asked: Why do all these projects fail? Lack of easily available information on how to manage IT projects is no longer a sufficient answer. Researchers and authors within the project management field have written running meters about how to handle different aspects of projects. Coordinators of projects need to consider an almost endless amount of technical, market-related, and organizational interdependencies. Most often, this requires mutual adjustment across many boundaries, both technical and organizational (Adler, 1999).

Moreover, the management of projects is also challenged by an ever-increasing pace of technical development, new standards, new features, product releases, reduced time-to-market, and changing customer demands (Bowersox, Stank, & Daugherty, 1999; Taxén & Lilliesköld, 2008). This requires flexibility to incorporate new technical solutions or functionalities during the development process. Consequently, new circumstances arise throughout the project that must be dealt with dynamically to meet both fixed and emerging targets and requirements. At the same time, companies need to prioritize their project portfolios in order to follow the overall strategy and build the company following a long-term view (Elonen & Artto, 2003). The prioritization is also necessary to make the most out of the company's scarce resources. Moreover, long lists of factors that significantly affect the outcome of the project have been identified. The question is no longer "what to do?" but "where to start?"

In several research fields, Bayesian networks are used to predict outcomes of complex situations (Jensen, 2001). Using Conditional Probability Distributions (CPDs), Bayesian networks make it possible to estimate the outcome of modeled scenarios, based on the probability distributions that underpin the model. Plugging such a Bayesian model into a decision theoretic framework, the effects of various decision scenarios

Address correspondence to Liv Gingnell, Industrial Information and Control Systems, KTH Royal Institute of Technology, Stockholm, Sweden. E-mail: livg@ics.kth.se

Colors versions of one or more of the figures in this article can be found online at www.tandfonline.com/uism.

can be evaluated beforehand, facilitating more rational decision-making (Resnik, 1987).

The contribution of this article is twofold. First and foremost, this study provides an expert-based Bayesian model that enables beforehand analysis of IT projects, based on measurements of widely cited success factors for IT projects. These success factors are derived from previous research on IT project management. A web-based survey with email invitation was answered by 51 experts on IT project management, which provided the probability data needed to build the Bayesian network.

Much has been written about success factors for project management. The same cannot be said about IT project management in specific. The second contribution of this article is therefore the focus on IT projects, summarizing existing knowledge on the topic through a holistic approach.

1.1 Outline

The remainder of the article is structured as follows: Section 2 briefly presents some recent research on IT project success factors, and contrasts the present contribution with some related work. Section 3 presents a literature review, and Section 4 introduces Bayesian networks, survey methodology in general, and the particular methods employed for teaching Bayesian networks from experts. Section 5 is the locus of the main contribution. Here, the results of the expert survey are described, and the resulting Bayesian model for assessment of the success of IT projects is built. The model is further explained through an example in Section 6. A discussion of the strengths and weaknesses of the contribution then ensues in Section 7, followed by some concluding remarks in Section 8.

2. RELATED WORK

This section presents related work in the areas of general project management and specifically IT project management success factors, as well as expert-based Bayesian models.

2.1. Project Management Success Factors

Numerous studies aiming to increase the understanding of factors that affect the outcome of projects have been conducted (Belout & Cauvreau, 2004; Cooke-Davies, 2002; Fortunde & White, 2006; Schindler & Eppler, 2003; Westerveld, 2003). Among these studies, Fortunde and White (2006) give the most extensive literature review. Cooke-Davies (2002) uses empirical data from 70 multinational companies to derive project success factors, while Schindler and Eppler (2003) strive to bridge the gap between theoretical knowledge and managerial reality by turning to learning methodology. Westerveld (2003) maps known success criteria and factors to the European Foundation for Quality Management (EFQM) excellence model (EFQM Organization, 2003) to facilitate improvement of project performance. Belout and Cauvreau (2004) make a profound contribution to the discussion of project success factors by noting that

independent variables for project success vary with the stages of the project life cycle.

All of the above consider projects in general rather than specific types of projects. However, as the present study focuses exclusively on IT projects, only research specifically about IT projects will be included in the literature review presented in Section 3.

2.2. IT Project Management

Different approaches have been applied in the quest to understand causes of IT project failure. This study aims to include an even distribution of sources based on theory, practice, and a mixture of the two. The studies discussed in this section are also used in the literature review and clustering of IT project success factors in Section 3.

Mahaney and Lederer (1999), Neimat (2005), and Reel (1999) use previous literature to investigate success and failure factors of IT projects. In Mahaney and Lederer (1999), the authors formulate a number of hypotheses about escalating IT projects that are to be tested in future research. Both Neimat (2005) and Reel (1999) discuss how critical success factors found in previous research can be used proactively to prevent IT project failures.

The Standish Group (2009), Xia and Lee (2004), and Yeo (2003) analyzed practical cases of project failure. Even though The Standish Group (2009) fails to describe their research methodology in a comprehensive way, the size of the study makes it impossible to overlook. In their widely cited report, The Standish Group (1993) analyzed 30,000 IT projects, and the study is continuously updated with new empirical data, resulting in a new report with a list of top-10 success and failure factors every second year. Xia and Lee (2004) use survey data from 541 IT development projects to analyze the complexity of such projects. In their study, they discuss how the identified complexity components affect the project performance. In Yeo (2003), the author conducts a survey where the respondents have experience from at least one failed IT project and summarizes the critical failure factors according to their experience.

Brocke, Uebernickel, and Brenner (2009) and Keil (1995) use a mixture of theory and empirics to investigate the causes of IT project success or failure. Brocke et al. (2009) analyze five cases of executed IT projects with a literature framework, thereby identifying critical success factors. Keil (1995) presents a model in which he summarizes factors that promote escalation of IT projects. The model is based on literature and tested through a case study where interview data, observations, and historical documents from a single project were used to triangulate the results (Keil, 1995).

Another interesting approach is applied by Wateridge (1997) that uses questionnaires and interviews to investigate how the triple constraint of time, budget, and quality affects the project success. He finds that a too one-sided focus on project time or budget is more likely to fail than projects that focus on commercial success. He also aims to broaden the definition of IT project success to include more than the triple constraint.

2.3. Expert-Based Bayesian Models

The application of Bayesian networks for information system quality analysis is proposed and applied in Lagerström, Franke, Johnson, and Ullberg (2009). In this article, an enterprise architecture evaluation framework for the analysis of information systems modifiability is presented. An expert survey was conducted in order to create a Bayesian model, the details of which are found in Lagerström, Johnson, Höök, and König (2009). A similar method was applied by Franke, Johnson, König & Marcks von Würtemberg (2012), investigating the reasons for IT systems un-availability. The present article uses the same research approach to investigate IT project success factors.

3. LITERATURE REVIEW

This section introduces the success factors proposed by selected literature sources, which were used when identifying and creating the factors employed in the survey presented in this article.

3.1. IT Project Management Success Factors

Several authors refer to the core three: (i) On time, (ii) within budget, and (iii) to quality, as the criteria to measure project success. One such source is Wateridge (1997), who also identified some additional factors through a literature study. Finally, Wateridge's personal hit list of project success factors is created based on a questionnaire where respondents were asked to indicate the four most important criteria for project success:

- Inadequacy in incorporating the views of all stakeholders in the project
- Project managers applying factors badly or applying wrong factors
- Perceived success of a project (satisfied key people on the project)
- Unrealistic timescale

Reel (1999) uses the 10 signs of IT project failures identified by Tom Field's analysis of pitfalls in software development (Field, 1997). Based on this list, Reel (1999) interprets the following factors of how to get a successful project start:

- Set realistic objectives and expectations—for everyone
- Build the right team
- Give the team what they think they need
- Developer attrition—keep it low (low rotation of stuff)
- Establish procedures and expectations
- Manage your product more than your personnel
- Track progress
- Establish routines to avoid bad decisions while selecting technologies
- Institutionalize a process for learning from your mistakes

Within a paper introducing ongoing research about failure rates of IT development projects, Mahaney and Lederer (1999) present 14 hypotheses about why IT development projects

fail. These hypotheses will be evaluated in a survey on IT project managers conducted by the researchers. Ten of the presented hypotheses, listed below, deals with different aspects of "runawayness" which derives from the four management disciplines—planning, organizing, controlling, and leading—and thus are relevant for this paper:

- The size of the project will be positively related to project runawayness
- Poorer planning will be positively related to project runawayness
- Poorer control of projects is positively related to project runawayness
- Poor estimation will be positively related to project runawayness
- Poorer monitoring of projects is positively related to project runawayness
- The degree of requirements that are misunderstood are positively related to project runawayness
- The newness of technology is positively related to project runawayness
- The degree of lack of senior management sponsorship is positively related to project runawayness
- The magnitude of problems with requirements specification is positively related to project runawayness
- The rate of turnover of key personnel is positively related to project runawayness

Another attempt to make sense in the field of IT project management studies is presented by Yeo (2003). This paper describes a proposal of expanding Checkland and Holwell's Processes for Organizational Meanings (POM) model, including existing organizing system and formalized information system, with the strategic project planning and delivery process system. Within this "triple-S" framework, an analysis of failure factors identified from 92 survey respondents has been conducted. The first one-third of the failure factors, five factors from each category, are presented in the paper:

- Minimization of timeline
- Weak definitions of requirements and scope
- Inadequate project risk analysis
- Incorrect assumptions regarding risk analysis
- Ambiguous business needs and unclear vision
- Lack of user involvement and inputs from the onset
- Top-down management style
- Poor internal communication
- Absence of an influential champion and change agent
- Reactive and not pro-active in dealing with problems
- Consultant/vendor underestimation the project scope and complexity
- Incomplete specifications when project started
- Inappropriate choice of software
- Changes in design specifications late in project
- Involvement high degree of customization in application

Based on a literature review, interviews, and focus discussions with IT development project managers, Xia and Lee (2004) have developed a taxonomy and measurement of the complexity of IT development projects. The taxonomy has also been evaluated through a survey to North American IT development projects. Finally, the survey result was evaluated according to its correlation with four project performance measures: (i) Project delivery time, (ii) cost, (iii) system functionality, and (iv) end-user satisfaction. The taxonomy identified by Xia and Lee encompasses both organizational/technological and structural/dynamic aspects of IT development projects (ITDPs), including the following factors:

- The project manager did not have direct control over project resources
- Users provided insufficient information
- Project had insufficient staffing
- Project personnel did not have required knowledge/skills
- Top management offered insufficient support
- The project involved multiple user units
- The system involved real-time data processing
- The project involved significant integration with other systems
- The project involved multiple contractors and vendors
- Users' information needs changed rapidly
- Users' business processes changed rapidly
- Organizational structure changed rapidly
- IT infrastructure changed rapidly
- IT architecture changed rapidly
- Software development tools changed rapidly
- The project involved multiple technology platforms
- Rapid technological changes make ITDPs even more complex
- The project involved multiple software environments
- Technical specifications are difficult to estimate and manage
- IT projects are more complex than project teams expect them to be
- Business requirements are difficult to estimate and manage
- Tougher business demands make ITDPs even more complex

The survey of Xia and Lee is rather similar to the one presented in this article. While they base their evaluation survey on responses from IT development project managers, the expert survey in this study is based on responses from IT project manager experts selected based on academic publications.

According to Neimat (2005), common IT project failure causes are related to project management. Based on research by the Coverdale Organization and observations from the Virtual Case File, the following causes are identified:

- Poor planning
- Unclear goals and objectives
- Objectives changing during the project
- Unrealistic time or resource estimates
- Lack of executive support and user involvement
- Failure to communicate and act as a team
- Inappropriate skills

Another perspective on success factors of IT projects is presented by Brocke et al. (2009), who extended their literature review with five case studies of IT projects executed at a European telecommunications company. Results from this study showed that successful projects are highly correlated with efforts to understand customers business and their needs. Specific important factors identified were:

- Poor top management support
- Poor understanding of the business of the customer
- Lack of close collaboration
- Unsuccessful communication
- Lack of alignment of approaches of processes
- Personnel selection
- Late identification of stakeholders

An escalation model based on literature has been created by Keil (1995) and used for analysis of a case study of an IT project. The result from the study suggests escalation being encouraged by a combination of project, psychological, social, and organizational factors. To avoid the escalation problems, the following failure factors are identified within the four factor groups:

- Evidence that continued investment could produce large payoff
- Project regarded as an investment in research and development
- Project setbacks appeared to be temporary problems
- Prior history of success
- High degree of personal responsibility for the outcome of the project
- Errors in the information processing
- Emotional attachment to the project
- Competitive rivalry between sales and manufacturing
- Need for external justification
- Norms for consistency
- Strong advocates who provided continued funding and protection
- Empire building
- Slack resources and loose management controls

The Standish Group CHAOS report (1993) is an important reference for most of the described works, as well as for the present article. An often-cited summary of the CHAOS top-10 factors (The Standish Group, 2009) includes the following success factors:

- User involvement
- Executive support
- Clear business objectives
- Emotional maturity
- Optimization
- Agile process
- Project management expertise
- Skilled resources
- Execution
- Tools and infrastructure

3.2. Clustering of IT Project Success Factors

The IT project success factors from the papers briefly described in Section 3.1 were compared to each other and clustered into new factors. The clustering was carried out by four of the authors in a workshop. Every explicit IT project management success factor found in previous literature was discussed in order to investigate the uniqueness of each factor. The reason for this is twofold. Firstly, the Noisy-OR approach used to build the Bayesian network (cf. Section 4.2), requires independent factors, meaning that it is important not to include several overlapping factors, nor the same factor, differently worded, several times. Secondly, a longer survey with several overlapping

questions would decrease the response rate (Blaxter, Hughes, & Tight, 2006).

The purpose of the clustering was thus not to cover every success factor found in literature, but to separate or merge them into independent factors when necessary. As a consequence of this, the factor *project planning* (Mahaney & Lederer, 1999) was excluded, since the meaning of the success factor as defined by Mahaney & Lederer was covered by other factors, such as *risk analysis* and *good estimations*, both of which are part of the planning process.

A more straight forward example is the success factor *skilled resources* that consists of *personnel selection* (Brocke et al., 2009), *inappropriate skills* (Neimat, 2005), *project personnel did not have required knowledge/skills* (Xia & Lee, 2004) and *project had insufficient staffing* (Xia & Lee, 2004). The factor *project had insufficient staffing* as defined by Xia and Lee (2004) refers both to the skills and the amount of resources in the project team, wherefore it also is a part of the clustered factor *good estimations*; see the factor definitions.

The results from the clustering was reviewed by a fifth author, who is a project management expert according to the same qualifications used to select experts for the survey, see further Section 4.1.1. The result of the clustering can be found in Table 1.

TABLE 1
IT Project Success Factors Found in Literature

Success Factor	Sources
Coworker influence	(Reel, 1999; Yeo, 2003)
Goals and objectives	(The Standish Group, 2009; Reel, 1999; Yeo, 2003; Xia & Lee, 2004; Neimat, 2005; Brocke et al., 2009)
Good estimations	(Wateridge, 1997; Mahaney & Lederer, 1999; Yeo, 2003; Xia & Lee, 2004; Neimat, 2005)
Handling of size and complexity	(The Standish Group, 2009; Mahaney & Lederer, 1999; Yeo, 2003; Xia & Lee, 2004)
Internal communication	(Wateridge, 1997; Reel, 1999; Mahaney & Lederer, 1999; Yeo, 2003; Neimat, 2005; Brocke et al., 2009)
Less commercial pressure	(Xia & Lee, 2004)
Less customized solution	(Yeo, 2003; Xia & Lee, 2004)
Lessons learned	(Reel, 1999)
No immature technology	(Reel, 1999; Mahaney & Lederer, 1999; Xia & Lee, 2004)
No late changes	(Yeo, 2003; Xia & Lee, 2004; Neimat, 2005)
No superfluous activities	(The Standish Group, 2009)
Project monitoring	(The Standish Group, 2009; Reel, 1999; Mahaney & Lederer, 1999; Yeo, 2003; Keil, 1995)
Risk analysis	(Yeo, 2003)
Skilled project manager	(The Standish Group, 2009; Xia & Lee, 2004)
Skilled team	(The Standish Group, 2009; Xia & Lee, 2004; Neimat, 2005; Brocke et al., 2009)
Stakeholder politics	(The Standish Group, 2009; Keil, 1995)
Systems requirements	(Mahaney & Lederer, 1999; Yeo, 2003; Xia & Lee, 2004; Brocke et al., 2009)
Top management support	(The Standish Group, 2009; Mahaney & Lederer, 1999; Yeo, 2003; Xia & Lee, 2004; Neimat, 2005; Brocke et al., 2009)
Use of tools and infrastructure	(The Standish Group, 2009; Yeo, 2003; Xia & Lee, 2004)
User involvement	(The Standish Group, 2009; Yeo, 2003; Xia & Lee, 2004; Neimat, 2005; Brocke et al., 2009)
Working routines	(Reel, 1999)

The clustered success factors are defined as follows:

Coworker influence means that the project manager does not make important decisions without consulting with the team.

Goals and objectives about the project outcome and purpose of the project are clearly formulated and successfully communicated to relevant stakeholders.

Good estimations means that the estimation of needed resources is realistic and measurable.

Handling of size and complexity includes allocation of enough resources to coordinate project stakeholders and personnel.

Internal communication means the communication within the project team.

Less commercial pressure enables a project timeline suited to the project rather than to the market.

Less customized solution enables a higher degree of modular reuse of solutions from previous projects.

Lessons learned means that established routines for knowledge transmission within and between projects are used.

No immature technology means that the system developed only depends on technology that is acknowledged to be operational.

No late changes refer to requirement changes regarding the outcome of the project.

No superfluous activities that do not, directly or indirectly, add value to the project outcome are carried out.

Project monitoring includes active use and continuous updating of the project plan and actions taken immediately when the project moves in unexpected directions.

Risk analysis includes established routines for discovering risk and to take preventive or responsive action to avoid them.

Skilled project manager refers for instance to the ability to adjust the leadership style depending on the group and the situation.

Skilled team refers not only to competent personnel in general, but requires that the team as a whole covers relevant knowledge to perform all tasks in the project.

Stakeholder politics means that relevant stakeholders (e.g., the board of directors) have the best for the project in mind rather than departmental interest.

Systems requirements should be complete, clearly formulated, and measurable.

Top management support means that the project sponsor is actively involved in the project.

Use of tools and infrastructure means that the use of tools and infrastructure is customized to the project needs.

User involvement means that the end user of the project outcome should be consulted throughout the project.

Working routines should be standardized and communicated to relevant personnel.

4. BAYESIAN NETWORKS

Friedman, Linial, and Nachman (2000) describes a Bayesian network, $B = (G, P)$, as a representation of a joint probability

distribution. The first component G is a directed acyclic graph consisting of vertices, V , and edges, E , (i.e., $G = (V, E)$). The vertices denote a domain of random variables X_1, \dots, X_n , also called chance nodes.

Each chance node, X_i may assume a value x_i from the finite domain $Val(X_i)$. The edges denote causal dependencies between the nodes (i.e. the causal relations between the nodes). Whenever an edge goes from a node X_i to a node X_j , X_i is said to be a causal parent of X_j . The second component P of the network B , describes a CPD for each chance node, $P(X_i)$, given the set of its causal parents $Pa(X_i)$ in G . It is now possible to write the joint probability distribution of the domain X_1, \dots, X_n using the chain rule of probability, in the product form

$$P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | Pa(X_i)) \quad (1)$$

In order to specify the joint distribution, the respective conditional probabilities that appear in the product form must be defined. The component P describes the distribution for each possible value x_i of X_i , and $pa(X_i)$ of $Pa(X_i)$, where $pa(X_i)$ is the set of values of $Pa(x_i)$. These conditional probabilities are represented in matrices, here forth called CPDs. Using a Bayesian network, it is possible to answer questions such as: What is the probability of variable X being in state x_i , given that $Y = y_i$ and $Z = z_i$?

In the general case, the relations between variables described by the conditional probability matrices can be arbitrarily complicated conditional probabilities. The model presented in this article uses only a single rather simple relation, leaky Noisy-OR, described in Section 4.2. More comprehensive treatments on Bayesian networks can be found in, for example, Neapolitan (2003), Jensen (2001), Shachter (1988), and Pearl (1988).

4.1. Expert Elicitation

Expert elicitation, in the context of Bayesian statistics, is the process where a person's knowledge and beliefs about one or more uncertain quantities are formulated into a joint probability distribution (Garthwaite, Kadane, & O'Hagan, 2005; i.e., the act of parameter estimation through the use of domain experts). This approach is generally used when available datasets are sparse in comparison with the number of nodes that need to be parameterized (Johansson & Falkman, 2006). Using a well-structured process for expert elicitation is important in order to minimize the bias of the domain expert. A rough outline of such an elicitation process is given in Renooij (2001):

1. Select and motivate the expert
2. Train the expert
3. Structure the questions
4. Elicit and document the expert judgments
5. Verify the results

In the following, we detail how each of these steps were carried out in the present study. The aspect of result verification is addressed in Section 7.

4.1.1. Select and Motivate the Expert

The selection of respondents in the present survey was based upon academic publications. Papers published in *International Journal of Project Management* 2005 – May 2010, *International Journal of Managing Projects in Business* 2008 – May 2010, and *Project Management Journal* 2009 – May 2010 were manually screened based on title and abstract to determine whether the authors should be invited to participate or not. In all, 456 experts were invited to the survey, of which 407 were found through *International Journal of Project Management*. As it was sometimes difficult to distinguish experts on project management in general from experts on IT project management, both categories were invited to the survey. Respondents without experience of IT projects were excluded from the analysis through background questions.

As the experts consulted in this study were widely spread geographically, a web-based survey was used. A personal invitation was sent to all respondents by email. A reason to use personal invitations is that the non-response bias then tend to be directly related to the subject, (i.e., chiefly respondents particularly interested in the subject participate; Fowler, 2002). This effect will be further discussed in Section 5. The internet-based application SurveyMonkey hosted the survey, which was open for four weeks, from June 14 to July 12, 2010. As recommended in Blaxter et al. (2006), reminders were sent to non-responding participants in the middle of the second week and the end of the third week to increase the response rate.

As noted in Renooij (2001), it is important to convince the experts that there is no straightforward way to tell a “right” from a “wrong” answer, but that their assessments should only represent their very own knowledge and experience as faithfully as possible. Furthermore, each question in the present survey included a self-assessment on the credibility of the answer, enabling anyone feeling uncertain to communicate this. As will be discussed in Section 4.2, this self-assessment also plays an important role in the construction of the Bayesian model.

4.1.2. Train the Expert

The validity of the study is highly dependent on the respondents’ comprehension of the questions. Therefore, it is often advisable to spend part of the survey to train the expert (Czaja & Blair, 2005), so that she will not only be a subject matter expert, but also an expert in giving probability estimates. In the present survey, this was accomplished by the use of an initial tutorial question, where the scope and aim of the question was explained using text and figures.

During the training phase, feedback on answers with known correct answers can help experts calibrate their responses (Baecher, 1988). However, in the present study, this was not feasible due to lack of indisputable data of sufficient generality, as one of the main reasons for performing this study is the lack of previous studies about quantified success factors. In the absence of possibilities for actual calibration, the selection of

respondents serves as an indirect calibration. Having published articles in the same forum and often referring to each other, they all belong to the same research field. Thereby, their theoretical knowledge on the topic can be assumed to be aligned with each other’s.

4.1.3. Structure and Questions

There are some different approaches to elicitation, direct elicitation being the most obvious and straightforward one. Here, questions are asked along the lines of “What is the probability that variable A takes state a , given the parent values b_1, c_1 ?” However, these questions can be hard for domain experts to relate to (Garthwaite et al., 2005), forcing the use of alternative approaches, as described for example in Woodberry, Nicholson, Korb, and Pollino (2005). In the present survey, a behaviorally anchored scale (Mangione, 1995) was used. The experts were asked to answer “What is the probability that a currently failed IT project would have been successful regarding (a) time, (b) budget and (c) quality if a best practice factor X had been present?” (Mutatis mutandis, depending on the appropriate grammar of each factor.) The factors were derived from current research about IT project management as described in Section 3. Their completeness is thoroughly discussed in Section 7.

Including subjective formulations such as “best practice” in a survey has both advantages and disadvantages. On the one hand, it is possible to interpret the question in several ways, which makes it more difficult to compare the answers with each other. On the other hand, the answers are to a lower extent limited to assumptions specified in the question (Mangione, 1995).

In this particular case, the authors of this article do not claim to know IT project management “best practice” better than the respondents, which is the reason that this subjective wording was deemed acceptable.

As noted in Cooke (1991), experts dislike writing numbers for subjective probabilities and prefer to check scales, place an “X” in a box, and the like. In the present survey, this was accommodated by using predefined scales in drop-down lists for the alternatives. To provide the respondent with a bird’s-eye view of the survey, a figure illustrating all question categories in order of appearance was continuously displayed throughout the survey.

4.1.4. Elicit and Document the Expert Judgments

Since elicitation is taxing for the expert, Cooke (1991) recommends that sessions should not exceed one hour. The present survey being web-based, with the possibility for the respondent to take a break or withdraw at his or her discretion, this problem can be considered of marginal importance. However, if a survey is too long or too complex, the response rate of the questionnaire decreases (Blaxter et al., 2006). The level of detail in this study was therefore limited by the expected response rate considered acceptable. A survey of the responses ex post indicates that a typical full response required about 20–30 minutes.

4.2. Building Bayesian Networks

Bayesian networks are a powerful formalism, but their use requires the specification of CPDs. As the number of variables X_1, \dots, X_n causally affecting a target variable Y grows, fully specifying these distributions becomes increasingly cumbersome. As noted by Onisko, Druzdzel, and Wasyluk (2001), a binary variable with n causal parents requires 2^n independent parameters to exhaustively describe the conditional probabilities. As n grows, 2^n parameters quickly become a prohibitive number. Often, however, canonical parameter-based distributions can be used to decrease the modeling effort, yet still give a sufficiently good approximation of the true distribution (Onisko et al., 2001).

The solution described in Onisko et al. (2001) is the use of a Noisy-OR gate. Using this formalism, the number of parameters required from expert estimation becomes only n , a significant gain. The underlying assumption is that instead of investigating every combinatorial interaction among the X_1, \dots, X_n causal parent variables, their interactions are modeled by a Noisy-OR gate. Furthermore, since Noisy-OR distributions approximate CPDs using fewer parameters, the resulting distributions are in general more reliable, being less susceptible to overfitting (Friedman & Goldszmidt, 1999).

The Noisy-OR gate (Onisko et al., 2001; Pearl, 1988) is typically used to describe the interaction of n causes X_1, \dots, X_n to an effect Y . In the present article, of course, this effect Y is the success of IT projects. Two assumptions are made, viz. (i) that each of the causes has a probability p_i of being sufficient for producing Y , and (ii) that the ability of each cause X_i to bring about Y is independent. Mathematically, the following holds:

$$p_i = P(y|\bar{x}_1, \bar{x}_2, \dots, \bar{x}_i, \dots, \bar{x}_n) \quad (2)$$

where x_i designates that causal factor X_i is present and \bar{x}_i that it is absent. It follows that the probability of y given that a subset $X_p \subseteq \{X_1, \dots, X_n\}$ of antecedent causes are present can be expressed as:

$$P(y|X_p) = 1 - (1 - p_0) \prod_{i: X_i \in X_p} (1 - p_i) \quad (3)$$

This is a compact specification of the CPD.

A natural extension proposed by Henrion (1989) is the so called *leaky* Noisy-OR gate. The rationale for the leakage is that models typically do not capture all causes of Y . If some potential X_i have been left out, as is the case in “almost all situations encountered in practice” (Onisko et al., 2001), this shortcoming can be reflected by adding an additional parameter p_0 the leak probability, such that

$$p_0 = P(y|\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n) \quad (4)$$

In words, this reflects the probability that Y will occur spontaneously, in the absence of all the explicitly modeled

causes X_1, \dots, X_n . In a leaky Noisy-OR gate, the CPD becomes

$$p(y|X_p) = 1 - (1 - p_0) \prod_{i: X_i \in X_p} \frac{1 - p_i}{1 - p_0} \quad (5)$$

The aim of the expert elicitation survey can now be stated more explicitly. For each of the factors X_1, \dots, X_n identified in the survey, a probability p_i of X_i causing a successful outcome of an IT project can be estimated. The respondents also provided a numeric value of p_0 , see further Section 5.

5. RESULTS

5.1. The Respondents

Figure 1 and Figure 2 display some basic data about the respondents who chose to complete the entire survey, viz. their affiliation and their number of years of experience of IT projects. Respondents with less than a year of experience with IT projects have been excluded from the analysis.

5.2. The Bayesian Model

The main results of the survey are presented in Tables 2–4. As described in Section 4, the respondents not only answered each question but also stated their certainty. In these tables, the N specified excludes respondents who stated to be unsure of their answers, and retains only those who were 50%, 90%, or

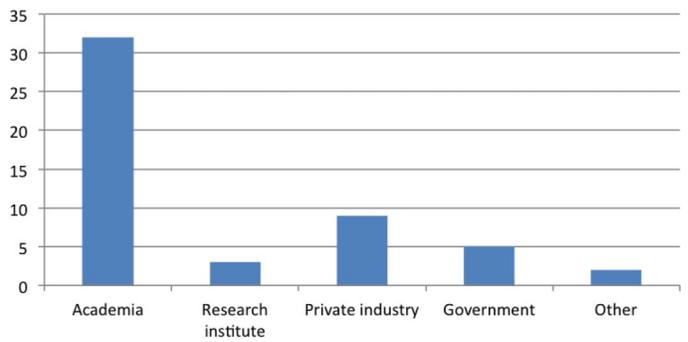


FIG. 1. Data on respondents' affiliations.

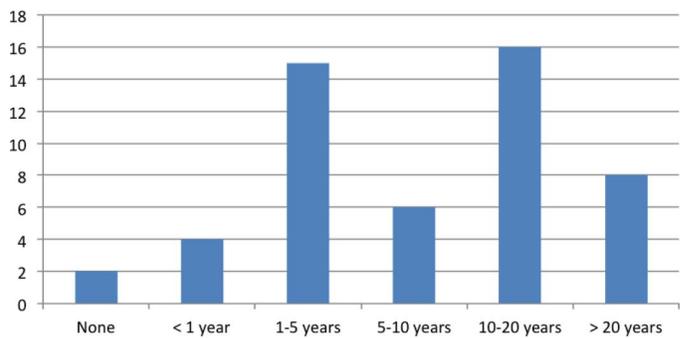


FIG. 2. Data on respondents' years of experience related to IT projects.

TABLE 2
 What is the Possibility That a Currently Failed IT Project Would Have Been Successful Regarding **TIME** if a Best Practice *FACTOR X* Had Been Present?

Causal Factor X	< 1%	1%–5%	5%–10%	10%–25%	25%–50%	> 50%	N
Coworker influence	5	12	18	15	3	1	54
Goals and objectives	3	3	5	13	13	21	58
Good estimations	4	6	3	7	10	21	51
Handling of size and complexity	3	5	9	10	12	12	51
Internal communication	3	7	6	13	11	14	54
Less commercial pressure	10	11	10	10	9	8	58
Less customized solution	6	7	10	9	11	8	51
Lessons learned	5	5	11	7	17	13	58
No immature technology	6	7	6	11	13	8	51
No late changes	5	4	7	8	14	13	51
No superfluous activities	4	13	13	9	8	4	51
Project monitoring	3	5	5	13	14	11	51
Risk analysis	3	4	12	17	20	20	76
Skilled project manager	1	5	12	8	8	20	54
Skilled team	2	6	7	12	7	20	54
Stakeholder politics	4	7	8	8	14	13	54
Systems requirement	4	6	13	10	10	8	51
Top management support	1	5	8	10	13	21	58
Use of tools and infrastructure	5	13	12	9	7	5	51
User involvement	4	6	9	12	12	11	54
Working routines	6	9	11	8	14	3	51

TABLE 3
 What is the Possibility That a Currently Failed IT Project Would Have Been Successful Regarding **BUDGET** if a Best Practice *FACTOR X* Had Been Present?

Causal Factor X	< 1%	1%–5%	5%–10%	10%–25%	25%–50%	> 50%	N
Coworker influence	4	16	13	17	2	2	54
Goals and objectives	5	1	9	15	11	17	58
Good estimations	2	5	5	9	13	17	51
Handling of size and complexity	3	7	8	8	12	13	51
Internal communication	3	8	8	16	9	10	54
Less commercial pressure	9	5	14	12	10	8	58
Less customized solution	6	10	12	6	9	8	51
Lessons learned	4	5	11	13	11	14	58
No immature technology	5	6	7	13	11	9	51
No late changes	4	3	9	11	12	12	51
No superfluous activities	6	14	11	9	6	5	51
Project monitoring	3	5	4	16	14	9	51
Risk analysis	6	6	14	13	16	21	76
Skilled project manager	2	2	11	9	9	21	54
Skilled team	4	10	5	10	12	13	54
Stakeholder politics	4	6	10	9	14	11	54
Systems requirement	3	8	12	9	10	9	51
Top management support	3	5	4	20	9	17	58
Use of tools and infrastructure	4	16	13	6	9	3	51
User involvement	4	4	9	18	9	10	54
Working routines	4	8	15	10	11	3	51

TABLE 4
What is the Possibility That a Currently Failed IT Project Would Have Been Successful Regarding **QUALITY** if a Best Practice **FACTOR X** Had Been Present?

Causal Factor X	<1%	1%–5%	5%–10%	10%–25%	25%–50%	>50%	N
Coworker influence	5	13	13	16	5	2	54
Goals and objectives	3	3	6	9	17	20	58
Good estimations	1	9	13	7	9	12	51
Handling of size and complexity	4	4	7	8	12	15	51
Internal communication	3	8	6	10	12	15	54
Less commercial pressure	8	8	10	14	10	8	58
Less customized solution	8	10	13	4	9	7	51
Lessons learned	4	7	8	8	14	17	58
No immature technology	7	7	10	5	12	10	51
No late changes	4	8	10	9	8	12	51
No superfluous activities	7	17	8	13	4	2	51
Project monitoring	4	10	6	10	9	12	51
Risk analysis	5	11	8	15	14	23	76
Skilled project manager	3	5	9	10	8	19	54
Skilled team	3	6	6	9	11	19	54
Stakeholder politics	5	5	8	8	10	18	54
Systems requirement	4	8	7	9	11	12	51
Top management support	2	5	12	11	10	18	58
Use of tools and infrastructure	3	17	11	10	5	5	51
User involvement	3	5	5	11	7	23	54
Working routines	6	11	10	10	10	4	51

9% certain. This corresponds to the answers actually used in building the Bayesian model.

As can be seen, the number of useful answers vary from 51 to 76 over the causal factors, with a mean of 54. While Tables 2–4 give a good overview of the results, they do not show the self-assessed uncertainties associated with each answer by the respondents. These, however, play a vital role in determining the Noisy-OR probabilities p_1, \dots, p_n associated with each of the causes X_1, \dots, X_n listed in the table. To weight these judgments into a single probability p_i for use in the Noisy-OR model, the respondents' answers were added as weighted mean samples

$$p_i = \frac{w_{i1}P_{i1} + w_{i2}P_{i2} \dots + w_{in}P_{in}}{W_{i1} + w_{i2} \dots + w_{in}} \quad (6)$$

where P is the midpoint of the i th answer interval and w_i is the respondents' certainty of their answers. p_i is thus an arithmetic mean (Lind, Marchal, & Wathen, 2008) of the answers, weighted by the certainty levels $w_i \in \{0.5, 0.9, 0.99\}$ of the answers (these figures were the alternatives used in the survey). Respondents who stated to be unsure of their answers are excluded from the weighting. The procedure is iterated for each and every causal factor, resulting in probabilities p_1, \dots, p_n as illustrated in Table 5– Table 7 (rounded to one decimal). Each p_i reflects the share of currently failed IT projects that would,

in the experts' opinion, be successful if the *factor* X_i had been managed according to best practice. It might seem counterintuitive that $\sum p_i > 100\%$, but consider a project that failed because of a poor risk analysis which did not identify a risk that subsequently occurred. Then it might also be true that this risk could have been identified, and failure averted, if appropriate attention had been paid to lessons learned from previous projects. Thus, the factors need not be mutually exclusive.

As can be seen from Tables 5–7, judging from the respondents' answers, top management support is the most important factor to finish on time, a skilled leader (project manager) is the most important factor to finish on budget, and user involvement is the most important factor to achieve acceptable quality.

One factor is still missing in order to obtain a complete leaky Noisy-OR model, viz.~the leakage p_0 . To obtain a numeric value of p_0 , the respondents were asked to what extent they believe that the aspects covered by the survey can explain the success or failure of an IT project. A leakage of $p_0 = 28\%$ corresponds to the answers and will be used throughout the remainder of the article. However, several respondents have explicitly commented that they think the survey "covers every important aspect," at the same time as they assess the leakage to 10–25 %. A possible interpretation of this is that every project includes unique success factors that cannot easily be generalized, but perhaps it is more plausible that the respondents have not fully understood the question on leakage.

TABLE 5
Causal Factors Regarding **TIME** With Probabilities for Noisy-OR Model

Causal Factor X_i Best Practice . . .		p_i
1	. . . top management support	46.7%
2	. . . good estimations	46.0%
3	. . . skilled team	44.8%
4	. . . goals and objectives	44.7%
5	. . . skilled project manager	44.6%
6	. . . stakeholder politics	38.7%
7	. . . risk analysis	38.6%
8	. . . no late changes	37.9%
9	. . . internal communication	36.0%
10	. . . project monitoring	35.9%
11	. . . handling of size and complexity	35.3%
12	. . . lessons learned	35.2%
13	. . . user involvement	31.2%
14	. . . systems requirement	29.0%
15	. . . less customized solution	27.4%
16	. . . no immature technology	25.0%
17	. . . less commercial pressure	22.9%
18	. . . working routines	22.4%
19	. . . no superfluous activities	19.9%
20	. . . use of tools and infrastructure	19.3%
21	. . . coworker influence	12.7%

TABLE 6
Causal Factors Regarding **BUDGET** With Probabilities for Noisy-OR Model

Causal Factor X_i Best Practice . . .		p_i
1	. . . skilled project manager	45.3%
2	. . . good estimations	43.7%
3	. . . top management support	38.5%
4	. . . goals and objectives	37.8%
5	. . . handling of size and complexity	36.8%
6	. . . risk analysis	36.6%
7	. . . lessons learned	36.4%
8	. . . no late changes	35.7%
9	. . . stakeholder politics	35.5%
10	. . . skilled team	34.8%
11	. . . project monitoring	32.9%
12	. . . user involvement	31.0%
13	. . . internal communication	30.5%
14	. . . no immature technology	29.8%
15	. . . systems requirements	27.0%
16	. . . less customized solution	25.4%
17	. . . less commercial pressure	24.2%
18	. . . working routines	21.6%
19	. . . no superfluous activities	20.4%
20	. . . use of tools and infrastructure	18.2%
21	. . . coworker influence	14.2%

6. A SCENARIO-ANALYSIS EXAMPLE

This section is intended to illustrate how the leaky Noisy-OR model presented in the previous section can be used for actual assessment on the outcome of an IT project and how it can guide decision-making with an impact on IT project management. The figures in the example are based on the assumed performance level of the IT project success factors. These are then used as input to the Leaky Noisy-OR model presented in Section 5.2 in order to analyze different scenarios. All figures in this section are derived from the calculations described in Section 5.2, given the stated performance of the different success factors.

To improve the performance of the IT project success factors defined in 3.1 is in most cases something that affects the project management process rather than specific projects. The Leaky Noisy-OR model presented in Section 5.2 could thus be used as decision basis for process improvement or portfolio management, comparing the expected outcome of possible investments, such as project sponsor educations or software modularization initiatives.

The model could also be used as input to specific projects. Consider as an example a project that is just about to start. The project comprises the development of the support system *Pegasus* and its delivery to an external customer. During the pre-study, the customer has claimed that the most important thing is the functionality of *Pegasus*, but that both time and budget

also are important. The project manager *PM* is considered to be one of the best project managers at the company and has many years of managing experience. She has worked together with the members of the project team before and has a good idea of how to establish working routines that will be well suited for everyone. She will also conduct a thorough risk analysis with the project team. None of the other factors in the Noisy-OR model however can be said to match best practice at the time. The project team for instance consists of only capable co-workers, but lack some technical competences that will be needed to realize the project successfully. *PM* therefore assumes that the success factors *skilled project manager*, *working routines*, and *risk analysis*, but no others match best practice. The project outcome can then be estimated as shown in Figure 3.

Considering the fact that the customer values quality over money, *PM* decides to add four weeks to the pre-study phase of the project, dedicated to specify the goals of the project and to communication with the end user of *Pegasus*. Thereby, she increases the probability that *Pegasus* is delivered with the right qualities from 79% to 94%, see Figure 4. She uses the model to motivate the procrastination of the planned delivery and the additional cost these changes bring about, arguing that they also increase the chances of keeping the timeline and budget of the project. The changes can therefore be seen as necessary investments for a successful development of *Pegasus*. It might be objected that this could have been read straight off Tables 5–7,

TABLE 7
Causal Factors Regarding **QUALITY** With Probabilities for Noisy-OR Model

	Causal Factor X_i Best Practice . . .	P_i
1	. . . user involvement	45.7%
2	. . . skilled team	45.5%
3	. . . goals and objectives	44.6%
4	. . . stakeholder politics	43.4%
5	. . . skilled project manager	42.2%
6	. . . top management support	39.0%
7	. . . handling of size and complexity	38.9%
8	. . . lessons learned	38.5%
9	. . . internal communication	37.2%
10	. . . risk analysis	36.1%
11	. . . systems requirements	34.8%
12	. . . good estimations	33.1%
13	. . . no late changes	32.3%
14	. . . project monitoring	31.5%
15	. . . no immature technology	30.7%
16	. . . less customized solution	24.5%
17	. . . less commercial pressure	23.4%
18	. . . working routines	21.7%
19	. . . use of tools and infrastructure	19.3%
20	. . . coworker influence	16.8%
21	. . . no superfluous activities	14.9%

by themselves, but the full leaky Noisy-OR model is needed. Another strength of the full model is that the expected cost of project failure (delayed delivery, loss of expected functionalities, etc.) can be compared to the estimated costs for getting the various factors up to best practice level.

7. DISCUSSION

From the results presented in Tables 5–7, it is worth noting that only 13 factors make it to the top-10 list for any of the success criteria time, budget, and quality. The remaining eight factors are, accordingly, considered less important by the experts for all the criteria. Indeed, the least-important factors are almost identically ordered for each of the success criteria. Among the top-10 success factors, however, the rankings differ over time, budget, and quality. The most drastic difference is *user involvement* that is considered the most important factor for the quality of an IT project, whereas it does not even make it to the top-10 lists for the other success criteria. This seems reasonable, as *user involvement* is a well-acknowledged (Brocke et al., 2009; Charette, 2006; Neimat, 2005; The Standish Group, 2009; Xia & Lee, 2004; Yeo, 2003) condition for assuring that the outcome of the project is possible to use, but also requires investments in time and thereby also in money.

At first sight, the probabilities derived from the expert opinions and presented in Tables 5–7 look surprisingly high. Could it really be that there are several different factors, each of which would suffice to increase the probability of success with almost 50% if performed according to best practice? If so, why are there still so many failed IT projects? On a second thought, however, managing any of these factors according to the best practice level is not that easy, and in some cases not even possible. For instance, many IT projects are performed under circumstances that necessitate requirement changes late in the project. On the other hand, education as well as experience

that finding the most promising candidates requires only an ordinal ranking.

However, a key strength of the Bayesian method is the possibility to investigate the impact of getting several factors up to best practice level at the same time. To evaluate these interactions of several factors, Tables 5–7 are not sufficient

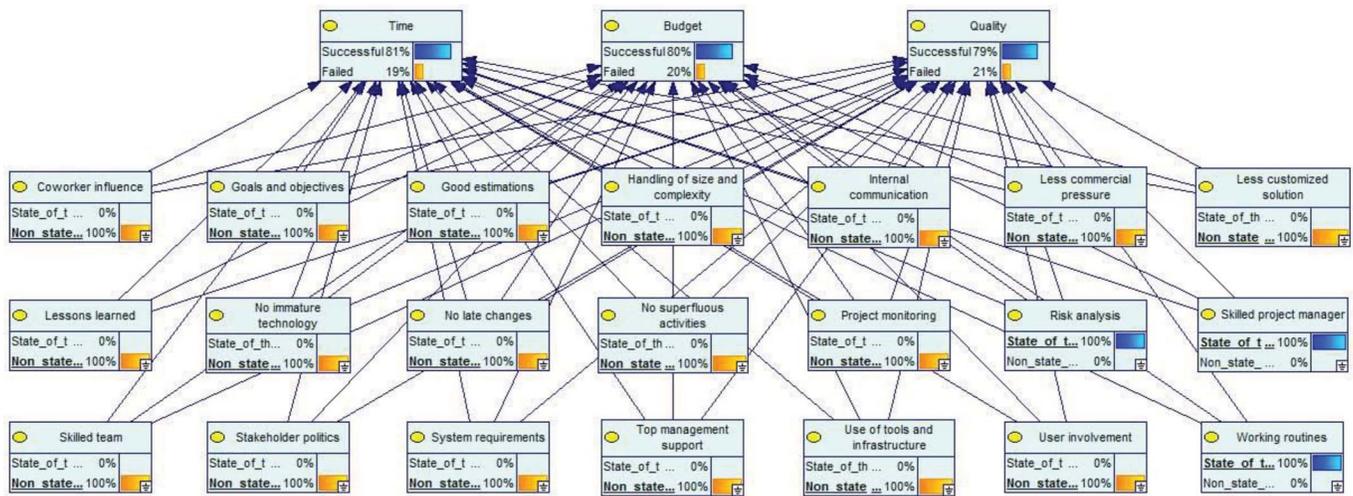


FIG. 3. Estimation of project outcome.

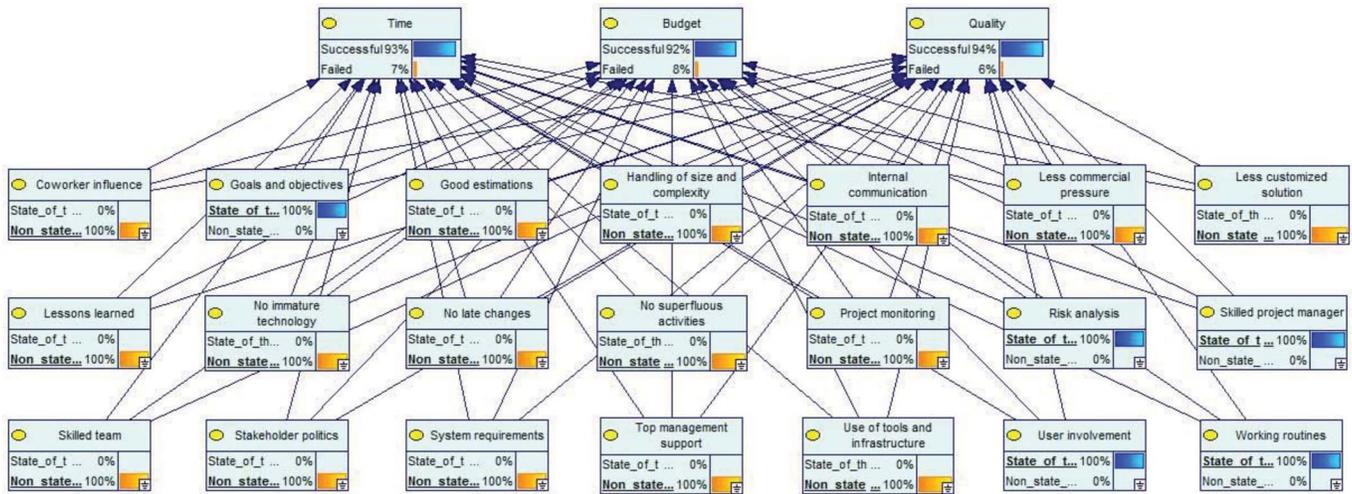


FIG. 4. Updated estimation of project outcome.

improves the performance of many of the apparently important factors. This model could thus be used to give indications where additional training of the coworkers would be beneficial for an organization.

Several of the top-10 success factors (e.g., *user involvement*, *top management support* and *goals and objectives*) are addressed in process improvement frameworks such as ISO 9000 (European Committee for Standardization, 2005) and EFQM (EFQM Organization, 2003), to some extent also in IT specific frameworks like COBIT (IT Governance Institute, 2007), ITIL (ITSM Library, 2007), and CMMI (Software Engineering Institute, 2009). Whether this indicates a real causal effect or whether the experts consulted in this study have been influenced by their knowledge about quality certification frameworks is a question that only can be answered through an empirical validation study.

Equally striking is the high degree of compliance between the top-10 success factors in Tables 5–7 and the number of citations per success factor shown in Table 1. Of the 13 consolidated top-10 factors derived from the expert survey, only four (*risk analysis*, *stakeholder politics*, *skilled leader*, and *no late changes*) did not make it to the top-10 most well-cited factors, though this might also depend on bias from respondent familiarity with the most-cited studies.

Interesting exceptions from the alignment between literature sources and the results are the factors *systems requirements* and *immature technology* that, even though well cited, were considered relatively unimportant by the experts. One possible interpretation of this is that these factors have been highly important, but that their importance have decreased over the years. The general approach to IT projects has undergone many changes during the last decade. One difference is that use of agile methods has grown more common, supposedly dealing with uncertainties such as unclear systems requirements in a better way throughout the project. This interpretation

is somewhat confirmed by the fact that a “clear statement of requirements” was considered the third-most-important success factor by the CHAOS report from 1993 (The Standish Group, 1993), whereas in the report from 2009 (The Standish Group, 2009), systems requirements are not mentioned at all.

7.1. Why Use a Bayesian Model?

So far, the results from the expert survey have been discussed pretty straightforwardly. An intuitive question might thus be: What does the Bayesian network approach add to the IT project management research field? The procedure of letting the respondents specify the importance of each success factor using intervals of percentages obviously adds another level of subjectivity to the results, compared to a simpler ranking of the factors.

As explained in Section 6, a Bayesian network enables beforehand analysis of certain scenarios. By adding information on which causal factors that can be performed in accordance with best practice, the model is able to predict the probability that the project will succeed regarding time, budget, and quality. The outcome of planned changes to a company’s project management routines could thereby be estimated beforehand. It is also possible to include the cost of going from non-best practice to best practice and thus calculate the optimal investment level in project management practice for a company wishing to improve the performance of IT projects.

The price that has to be paid for these possibilities is the additional uncertainties attached to the expert elicitation. This uncertainty can, however, be reduced by validation studies reviewing the implementation and outcome of IT projects. The study at hand shall be viewed only as a first version of the model that will be iteratively improved by future empirical studies.

7.2. Validity of the Model

As noted in Section 4, proper use of the Noisy-OR gate makes two assumptions regarding the structure of the interaction of causes (X_1, \dots, X_n) and effect (Y ; Onisko et al., 2001; Pearl, 1988). These are (i) that each of the causes has a probability p_i of being sufficient for producing Y , and (ii) that the ability of each cause X_1, \dots, X_n to bring about Y is independent. In the present study, Y is successful outcome of an IT project regarding the success criteria *time*, *budget*, and *quality*. X_1, \dots, X_n are causal factors bringing the success about.

Arguing for (i) in this case is straightforward. Indeed, all of the factors included in this study have been discussed in the literature as having an immense impact on the project outcome. However, this impact is not always deterministic (e.g., best-practice top management support will not infallibly lead to project success, but will do so with a certain probability p). Arguing for (ii) is harder. In many cases, it is reasonable to assume that factors are independent, but this is not always the case. One project manager could for instance promote extensive use of risk analysis in the project planning, as could any of the project team members. Therefore, the impact of factors such as *risk analysis* and *top management support* depend to some extent on other factors. In general terms, the distinction between proactive and reactive factors indicate that (ii) is an approximation that does not hold in all circumstances. In the full model, such dependencies could be modeled by rescaling different factors p_i with different factors α_i accounting for interactions. However, to accurately reflect these phenomena, more empirical data is needed. To conclude, while the assumptions required for the Noisy-OR model are reasonable as a first approximation, the model should certainly be open to further refinement. Such refinement is a matter of empirical investigation, where the handling of the success factors from IT projects can be analyzed and statistically checked for independence and compared to the actual project outcome. It should be noted, however, that there is no need to refashion the entire Bayesian model if some cause variables X_1, \dots, X_n turn out to be dependent. The bulk of the causes can remain modeled in a Noisy-OR relation to each other, while a select few can be modeled using different CPDs.

Another potential weakness of the model is the absence of respondent calibration (Baecher, 1988) in the training phase of the survey. As one of the main reasons for performing this study is the lack of previous studies about quantified success factors, there are no figures at all to use for calibration, and even less so indisputable figures of sufficient generality. In the absence of possibilities for actual calibration, the selection of respondents serves as an indirect calibration. Having published articles in the same forum and often referring to each other, they all belong to the same research field. Thereby, their theoretical knowledge on the topic can be assumed to be aligned with each other's. Nevertheless, the results from this study are to be seen as a first attempt to quantify IT project success factors. The model could and should be validated through empirical data, which is on the agenda for future research.

7.3. Limitations of the Model

The model presented in this article is based on theoretical expertise on IT project management, thereby providing a general quantification of the success factors. The reason for this is to make the model general enough to be adaptable in all types of organizations. Few metrics-based models—which are designed with generalized empirical data or expert elicited data—that work perfectly for the specific case can be found. Most models are built around the general case but require calibration with case-specific data in order to work well.

COCOMO, CONstructive COSt MODEL, was in its first version released in the early 1980s. It became one of the most frequently used and most appreciated software cost estimation metrics-based models (Boehm, Abts, & Chulani, 2000; Chulani, Boehm, & Steece, 1999). Although the latest version is based upon both empirically collected data from software projects and expert information, Chulani et al. (1999) recommend that organizations using COCOMO calibrate it using company-specific data to increase the model accuracy and usability.

TEAMATe, The Enterprise Architecture Modifiability Analysis Tool, was proposed in its current form in 2010. It contains a metrics-based architecture metamodel for change cost analysis of software projects. The model is based on expert elicited data and tested in industrial case studies. This model is built as a general model but requires a categorization of company-specific project measures in order to perform well (Lagerström, Johnson, Höök, & König, 2010).

Chiesa, Frattini, Lazzarotti, and Manzini (2009) have studied Performance Measurement Systems (PMS) in R&D. They found in both a literature study and a multiple case study that the performance measurement system, and especially the metrics used, need to be customized based on some contextual factors of the firm in which the system is used. Similar results were also found by Vanek, Jackson, and Grzybowski (2008), who present a literature study focusing on systems-engineering metrics in product development. One of their main conclusions were that there is no ultimate key set of metrics for the product development process, but there is probably a set of metrics that are the most appropriate for the product development process at one business unit in a company.

Diez (1993) presents an algorithm for parameter adjustment in noisy OR-gate based Bayesian models. According to the author, “the knowledge engineer must resort to subjective assessments of probability obtained from human experts, who make use of their memory and of the literature published in their field.” Diez (1993, p. 99) concludes that these expert-based models often need refinement based on real cases.

This is also the case for the model proposed in this article. It can and most likely it will need a company-specific calibration in order to perform as required. Thus, in future work, the Bayesian model will be calibrated with industry cases using the method presented by Diez (1993).

AUTHOR BIOS

Liv Gingnell (born Marcks von Württemberg) holds a Master of Science in Engineering and of Education from the Royal Institute of Technology and Stockholm University. She joined the department of Industrial Information and Control Systems on the Royal Institute of Technology (Stockholm, Sweden) for PhD studies in November 2009. Her research interests are project management, Lean Product Development, Design for Six Sigma and the implementation of development frameworks in the industry. In addition to her research interests, she supervises master's thesis students and teaches the course Business Development and Quality Management.

Ulrik Franke works as a scientist at the Swedish Defence Research Agency (FOI) in Stockholm, within the Decision Support Systems group. His research interests include Enterprise Architecture, the theory and practice of decision-making, information fusion, and the impact of ICT on politics and national security. He received his PhD in Industrial Information and Control Systems in 2012 from the Royal Institute of Technology (KTH) in Stockholm. He has published articles in various journals, for example: IEEE Transactions on Network and Service Management, Software Quality Journal, and Enterprise Information Systems.

Robert Lagerström is an assistant professor at the Royal Institute of Technology (KTH) in Stockholm, Sweden. During 2010/2011, Robert did an industrial post-doc at ABB Corporate Research in Västerås, Sweden. He received his PhD degree in Industrial Information and Control Systems in 2010 and his MSc degree in Computer Science in 2005, both at KTH. His topic of research is enterprise architecture and systems modifiability. Robert has written a number of academic publications in the field of enterprise architecture, modifiability, IT-management, and he is also a co-author of the book *Enterprise Architecture: Models and Analyses for Information Systems Decision Making*.

Evelina Ericsson is a PhD student at the department of Industrial Information and Control System, Royal Institute of Technology (Stockholm, Sweden). She holds a Master of Science in Engineering and of Education from the Royal Institute of Technology and Stockholm University. Her research interests are Design for Six Sigma, Lean Product Development, and the integration and implementation of these frameworks in organizations product development processes.

Joakim Lilliesköld is an assistant professor at the Royal Institute of Technology (KTH). During the last 13 years, he has studied complex system development projects in many different settings. His thesis, "Managing Complex Industrial Projects - A Comparison Between Holistic Models" was based on best-practice models found in Swedish industry. Joakim also has a strong interest in education and has many years of experience teaching project management. He has

authored and co-authored several books. Today, he is the Director of Undergraduate Studies at the School of Electrical Engineering at the Royal Institute of Technology (KTH).

REFERENCES

- Adler, N. (1999). *Managing complex product development three approaches*. (Doctoral dissertation). Stockholm School of Business, Stockholm, Sweden. Retrieved from <http://hhs.diva-portal.org/smash/get/diva2:220593/FULLTEXT01>
- Baecher, G. (1988). *Judgemental probability in geotechnical risk assessment*. Technical report. New Orleans, LA: The Office of the Chief, U.S. Army Corps of Engineers.
- Belout, A., & Cauvreau, C. (2004). Factors influencing project success: The impact of human resource management. *International Journal of Project Management*, 22(1), 1–11.
- Blaxter, L., Hughes, C., & Tight, M. (2006). *How to research*. Berkshire, UK: Open University Press.
- Boehm, B., Abts, C., and Chulani, S. (2000). Software development cost estimation approaches: A survey. *Annals of Software Engineering*, 10(1–4), 177–205.
- Bowersox, D. J., Stank, T. P., & Daugherty, P. J. (1999). Lean launch: Managing product introduction risk through response-based logistics. *Journal of Product Innovation Management*, 16(6), 557–568.
- Brocke, H., Uebernickel, F., & Brenner, W. (2009). Success factors in it-projects to provide customer value propositions. In *20th Australasian Conference on Information Systems*, Australia.
- Charette, R. N. (2006). Why software fails. *IEEE Spectrum*, 42(9), 42–49.
- Checkwell, P., & Holwell, S. (1993). Information management and organizational processes: an approach through soft systems methodology. *Information Systems Journal*, 3(1), 3–16.
- Chiesa, V., Fratini, F., Lazzarotti, V., & Manzini, R. (2009). Performance measurement of research and development activities. *European Journal of Innovation Management*, 12(1), 25–61.
- Chulani, S., Boehm, B., & Steece, B. (1999). Calibrating software cost models using Bayesian analysis. In *IEEE Transactions on Software Engineering*, pp. 573–583.
- Cooke, R. M. (1991). *Experts in uncertainty. Opinion and subjective probability in science*. Oxford, UK: Oxford University Press.
- Cooke-Davies, T. (2002). The "real" success factors on projects. *International Journal of Project Management*, 20(3), 185–190.
- Czaja, R., & Blair, J. (2005). *Designing surveys*. Thousand Oaks, CA: Sage Publications Inc.
- Diez, F. (1993). Parameter adjustment in Bayes networks—The generalized noisy or-gate. In *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence*, pp. 99–105, Washington D.C.
- EFQM Organization (2003). *The EFQM excellence model*. Technical report. Brussels, Belgium: EFQM.
- Elonen, S., & Arto, K. A. (2003). Problems in managing internal development projects in multi-project environments. *International Journal of Project Management*, 21(6), 395–402.
- European Committee for Standardization. (2005). *SS-EN ISO 9000 Quality management systems—Fundamentals and vocabulary*. Technical report. Stockholm, Sweden: ISO.
- Field, T. (1997). When bad things happen to good projects. *CIO*, 11(Oct.), 55–62.
- Fortunde, J., & White, D. (2006). Framing of project critical success factors by a systems model. *International Journal of Project Management*, 24(1), 53–65.
- Fowler, F. J. J. (2002). *Survey research methods*. Thousand Oaks, CA: Sage Publications, Inc.
- Franke, U., Johnson, P., König, J., & Marcks von Württemberg, L. (2012). Availability of enterprise IT systems: An expert-based Bayesian framework. *Software Quality Journal* 20(2), 369–394.
- Friedman, N., & Goldszmidt, M. (1999). Learning Bayesian networks with local structure. In M. I. Jordan (Ed.), *Graphical models*, (pp. 421–459). Cambridge, MA: MIT Press.

- Friedman, N., Linial, M., & Nachman, I. (2000). Using Bayesian networks to analyze expression data. *Journal of Computational Biology*, 7, 601–620.
- Garthwaite, P. H., Kadane, J. B., & O'Hagan, A. (2005). Statistical methods for eliciting probability distributions. *Journal of the American Statistical Association*, 100, 680–701.
- Glass, R. (2006). The Standish report: Does it really describe a software crisis? *Communications of the ACM*, 49(8), 15–16.
- Henrion, M. (1989). Practical issues in constructing belief networks. In J. Lemmer, T. Levitt, and L. Kanal (Eds.), *Proceedings of the Third Conference on Uncertainty in Artificial Intelligence (UAI1987)*. Corvallis, OR: AUAI Press.
- IT Governance Institute. (2007). *Control objectives for information and related technology* (4.1 edition). Technical report. Rolling Meadows, IL: IT Governance Institute.
- ITSM Library. (2007). *ITIL: It service management based on ITIL v3: A pocket guide*. Technical report. Buckinghamshire, UK: Author.
- Jensen, F. V. (2001). *Bayesian networks and decision graphs*. Secaucus, NJ: Springer-Verlag New York, Inc.
- Johansson, F., & Falkman, G. (2006). Implementation and integration of a Bayesian network for prediction of tactical intention into a ground target simulator. In *Information Fusion, 2006 9th International Conference on*, pp. 1–7.
- Keil, M. (1995). Pulling the plug: Software project management and the problem of project escalation. *MIS quarterly*, 19(4), 421–447.
- Lagerström, R., Franke, U., Johnson, P., & Ullberg, J. (2009). A method for creating Enterprise Architecture metamodels—Applied to systems modifiability analysis. *International Journal of Computer Science & Applications*, 6, 89–120.
- Lagerström, R., Johnson, P., Höök, D., & König, J. (2009). Software change project cost estimation—A Bayesian network and a method for expert elicitation. In *Third International Workshop on Software Quality and Maintainability (SQM 2009)*.
- Lagerström, R., Johnson, P., Höök, D., & König, J. (2010). Architecture analysis of enterprise systems modifiability—Models, analysis and validation. *Journal of Systems and Software*, 83(8), 1387–1403.
- Lind, D., Marchal, W., & Wathen, S. (2008). *Statistical techniques in business and economics with global data sets*. New York: McGraw-Hill.
- Mahaney, R. C., & Lederer, A. L. (1999). Runaway information systems projects and escalating commitment. In *1999 ACM SIGCPR conference on computer personnel research*. New York: ACM.
- Mangione, T. W. (1995). *Mail surveys. Improving the quality*. Thousand Oaks, CA: Sage Publications, Inc.
- McManus, J., & Wood-Harper, T. (2007). Understanding the sources of information systems project failure. *Management Services*, 51(Autumn), 38–43.
- Neapolitan, R. E. (2003). *Learning Bayesian networks*. Upper Saddle River, NJ: Prentice-Hall, Inc.
- Neimat, T. A. (2005). Why IT projects fail. *White paper collection*. Retrieved from http://www.projectperfect.com.au/info_it_projects_fail.php
- Onisko, A., Druzdel, M. J., & Wasyluk, H. (2001). Learning Bayesian network parameters from small data sets: Application of noisy-or gates. *International Journal of Approximate Reasoning*, 27(2), 165–182.
- Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. San Francisco: Morgan Kaufmann Publishers Inc.
- Reel, J. S. (1999). Critical success factors in software projects. *IEEE Software*, 16(May/June), 18–23.
- Renooij, S. (2001). *Qualitative approaches to quantifying probabilistic networks*. (Unpublished doctoral dissertation). Utrecht, the Netherlands: Utrecht University.
- Resnik, M. D. (1987). *Choices: An introduction to decision theory*. Minneapolis, MN: University of Minnesota Press.
- Schindler, M., & Eppler, M. J. (2003). Harvesting project knowledge: A review of project learning methods and success factors. *International Journal of Project Management*, 21(3), 219–228.
- Shachter, R. D. (1988). Probabilistic inference and influence diagrams. *Operations Research*, 36(4), 589–604.
- Software Engineering Institute. (2009). *CMMI for services, version 1.2*. Technical report. Pittsburgh, PA: Author.
- The Standish Group. (1993). *Chaos report 1993*. Technical report. Boston, MA: Author.
- The Standish Group. (2009). *Chaos summary 2009*. Technical report. Boston, MA: Author.
- Taxén, L., & Lilliesköld, J. (2008). Images as action instruments in complex projects. *International Journal of Project Management*, 26, 527–536.
- U.S. Department of Commerce. (2003). *Digital economy 2003*. Technical report. Washington, DC: Author.
- Vanek, F., Jackson, P., & Grzybowski, R. (2008). Systems engineering metrics and applications in product development: A critical literature review and agenda for further research. *Systems Engineering*, 11(2), 107–124.
- Wateridge, J. (1997). How can is/it projects be measured for success? *International Journal of Project Management*, 16(1), 59–63.
- Westerveld, E. (2003). The project excellence model: Linking success criteria and critical success factors. *International Journal of Project Management*, 21(6), 411–418.
- Woodberry, O., Nicholson, A. E., Korb, K. B., & Pollino, C. (2005). Parameterising Bayesian networks. In *AI 2004: Advances in Artificial Intelligence* (pp. 1101–1107). Berlin/Heidelberg: Springer.
- Xia, B. W., & Lee, G. (2004). Grasping the complexity of is development projects. *Communications of the ACM*, 47(5), 68–74.
- Yeo, K. (2003). Critical failure factors in information system projects. *International Journal of Project Management*, 20(3), 241–246.