International Journal of Computer Applications (0975 – 8887)
4th International IT Summit Confluence 2013 - The Next Generation Information Technology Summit

Strategies for Evaluating Risks to Project Schedule

P. K. Suri
Dean, Research and Development,
Chairman CSE/IT/MCA, HCTM Technical Campus,
Kaithal, Haryana, India

Kanchan Narula
Student, HCTM Technical Campus,
Kaithal, Haryana, India

ABSTRACT
Risk management is critical to the success of any software project. The project schedule is the core of the project planning. In the software project development process, risk scheduling is one of the most significant disciplines that cannot be mastered by anyone. So, evaluating risks to the schedule is complex. This paper presents different strategies for schedule risk analysis.

General Terms
Risk Analysis, Project Schedule.

Keywords
Simulation, PERT, Monte Carlo, Estimates.

1. INTRODUCTION
Risk is a probability of occurrence of some unwanted and harmful event to the project. These events can result in cost overruns, schedule slippage, or failure to meet their project goals. Risk is the possibility of loss.

It is a function of both the probability of hazard's occurring and its impact on the project. A risk is the precursor to a problem; the probability that, at any given point in the software life cycle, the desired goals cannot be achieved within available resources and time. Identification of assessment of risks is very cumbersome task. Risk cannot be eliminated from software, but it can be managed.

2. SOFTWARE RISK MANAGEMENT
Risk management is critical to the success of any software project. The objective of risk management is to avoid or minimize the adverse effects of unforeseen events by avoiding the risks or drawing up contingency plans for dealing with them. Boehm defines four major reasons for implementing software risk management [5]:

1. Avoiding software project disasters, including run away budgets and schedules, defect-ridden software products, and operational failures.

2. Avoiding rework caused by erroneous, missing, or ambiguous requirements, design or code, which typically consumes 40-50% of the total cost of software development.

3. Avoiding overkill with detection and prevention techniques in areas of minimal or no risk.

4. Stimulating a win-win software solution where the customer receives the product they need and the vendor makes the profits they expect.

The risk management process starts with the identification of risks. Each of the risks is then analyzed and prioritized. A risk management plan is made that identifies containment actions to reduce the probability of the risk. The plan includes contingency actions that will be taken if the risk turns into a problem. The next step involves monitoring the status of known risks as well as the results of risk reduction actions.

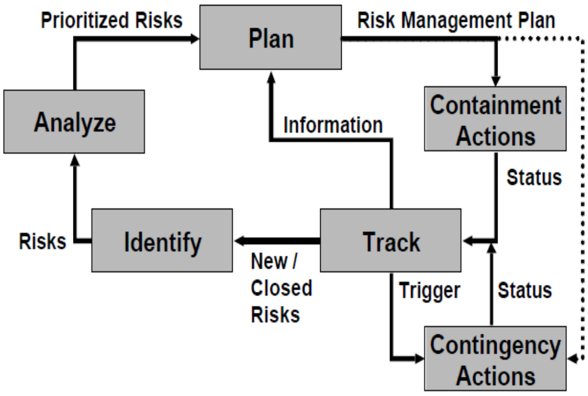

Fig. 1: The Risk Management Process [28]

3. CURRENT TRENDS IN SOFTWARE RISK MANAGEMENT
1987, James G. March and Zur Shapira explores the relation between decision theoretic conceptions of risk and the conceptions held by executives [18].

1991, Boehm describes the emerging discipline of software risk management. Its objectives are to identify, address, and eliminate risk items before they become either threats to successful software operation or major sources of software rework [7].

1992, Roger L. Van Scoy reviews the fundamental concepts of risk and elaborate how these basic concepts apply to the development of large, software-intensive systems, develop strategy for seeing a systematic approach to risk management in software development [42].

1994, Gluch, D. P. establishes a representation of software risk wherein the risks associated with software-dependent development programs are defined as distinct, manageable risk entities [11].

1994, James H. Lambed, Nicholas C. Matalas, Con Way Ling, Yacov Y. Haimes, and Duan Lil makes the case that assessment of the tail of the distribution can be performed separately from assessment of the central values [19].

1996, Ronald P. Higuera, Yacov Y. Haimes presents a holistic vision of the risk-based methodologies for Software Risk

Management (SRM) developed at the Software Engineering Institute (SEI) [43].

1996, Rodrigues, A. contrasts the characteristics of the two approaches and provides an overview of various areas of application of system dynamics in project management [41].

1997, Gemmer, A. describes a plan to elicit the behaviour "communicate risk more effectively" [10].

1999, Jerry Banks Marietta, Georgia, introduced Modeling concepts in simulation based on the example and discussed the advantages and disadvantages of simulation [37].

2000, Jiang et al. examines the relationship between project risks and information systems project team performance [22].

2001, Jiang et al. proposes and tests a model based on literature that relates sources of risk to strategies and success [21].

2002, Yacoub and Ammar describe a heuristic risk assessment methodology that is based on dynamic metrics [48].

2004, Y.H. Kwak, J. Stoddard addresses lessons learned from implementing project risk management practices in software development environment [49].

2004, Kwan-Sik Na, Xiaotong Li, James T. Simpson, Ki-Yoon Kim conduct a comparative study to determine how risk management strategies impact software product and process performance in countries with dissimilar IT capabilities [24].

2004, Jakub MILER, Janusz GÓRSKI presents a systematic approach to software risk identification based on risk patterns, demonstrates how this approach can be applied using RUP as the reference model of software processes [17].

2005, Young Hoon Kwak, Kenneth Scott LaPlace defines that Risk tolerance is often misunderstood or overlooked by project managers [50].

2005, Dragan Milosevic, Peerasit Patanakul undertook an exploratory study into the impact of SPM on project performance in development projects in high-velocity industries [9].

2006, Mei-yuan Wang, Yao-bin Lu, Jin-long Zhang established an index evaluation system to evaluate and select outsourcees using the principles of Analytical Hierarchy Process (AHP) and Cluster Analysis based on Group Decision Making [33].

2007, Dr P K Suri, Manoj Wadhwa provides a quantitative means to assess the risk associated with software development, by outlining the different factors which introduce the risk, assigning weightages to each factor [20].

2007, Kwan-Sik Na, James T. Simpson, Xiaotong Li, Tushar Singh, Ki-Yoon Kim investigate the impact of two alternative conceptualization of software development risk on both objective performance and subjective performance [25].

2008, Gupta, D. Sadiq, M. proposed a software risk assessment and estimation model (SRAEM) [12].

2008, Hassan Mathkour, Ghazy Assassa, and A. Baihan apply risk management to software development that uses extreme programming approach. A risk tool is designed and developed using MS Excel [14].

2008, Dr.Ing. Tilo Nemuth implement risk analysis tool for construction project evaluation in the tender phase based on Monte Carlo Simulation [16].

2009, Karel de Bakker, Albert Boonstra, Hans Wortmann presents a meta-analysis of the empirical evidence that either supports or opposes the claim that risk management contributes to IT project success [23].

2009, P. K. Suri, Bharat Bhushan, Ashish Jolly design a simulator for analyzing the performance measurement of Software Risk Assessment using Markov process [36].

2009, Undram Chinbat, Soemon Takakuwa focuses on the devel-opment of a simulation method which provides an engineering tool for managing risks associated with the development of open mining improvement projects [46].

2010, M.S. ROJABANU, Dr. K. Alagarsamy proposed a model for the Software Risk Management based on the Developer, Development process and the customer [34].

2010, Ayse Kucuk Yilmaz and Triant Flouris presents an integrative conceptual framework for sustainability risk management in enterprise-wide [4].

2011, Sneh Prabha, R.L.Ujjawal proposed a technique to evaluate the risk based on the source code as well as on the changes in the requirements of the user [45].

2011, Malaya Kumar Nayak, Sanghamitra Mohanty, Rachna Soni describes an approach used to reveal dynamics operating in supply of, and demand for, skilled Information and Communications Technology (ICT) employees [31].

2011, Aurangzeb Khan, Dr. Farooque Azam, Muhammad Shoaib Zafar describes how to mitigate the risk in requirement gathering phase of Global software development (GSD) [2].

2011, Dhirendra Pandey, Ugrasen Suman, A. K. Ramani proposed a method to match requirement engineering approaches with risk assessments approaches [8].

2011, Liu Jun, Wang Qiuzhen, Ma Qingguo develops an integrative model to explore the moderating effects of uncertainty on the relationship between risk management and IS development project performance from a vendor perspective [29].

2011, Ms Manisha.Ingle, Dr.Mohommad Atique, Prof. S. O. Dahad reports the methodology to solve risk analysis problems with the purpose of determining the project's attractiveness [32].

2011, Lazaros Sarigiannidis, Prodromos D. Chatzoglou observed positive impact of adopting risk management strategies on projects [26].

2011, Averill M. Law give a three-activity approach for choosing the theoretical distribution that best represents a set of observed data [3].

2011, Abdullah Al Murad Chowdhury and Shamsul Arefeen recognizes the increasing role of risk management in present software projects [1].

2011, Lazaros Sarigiannidis, Prodromos D. Chatzoglou investigates a wide range of relevant literature, proposes a new conceptual framework for managing risk in software development projects [27].

2012, Haneen Hijazi, Thair Khdour, Abdulsalam Alarabeyyat investigate the state of risk and risk management in the most popular software development process models [13].

2012, P.K. Suri, Pallavi Ranjan discussed several existing methods for software cost estimation and their aspects [38].

2012, Vinay Kumar Nassa, Sri Krishan Yadav present an approach for creating a robust risks classifications and measurement system [47].

2012, Iqbal Felani developed a risk management information system. This information system will be supported by some simple methods to reduce subjectivity and qualitative result [15].

2012, P.K. Suri, Pallavi Ranjan implement a simulator in C language which estimates the probability of a successful project completion [39].

2012, Malaya Kumar Nayak, Dr Sanghamitra Mohanty deals with the benefit of conducting schedule risk analysis on an ICT Infrastructure Project [30].

2012, Paul Clarke a, Rory V. O'Connor consolidated a substantial body of related research into an initial reference framework of the situational factors affecting the software development process [35].

Linda Westfall reviews the basic concepts, terminology, and techniques of Software Risk Management [28].

Severin V. Grabski., Stewart A. Leech., Bai Lu. identify the risks and controls used in ERP implementations [44].

4. RISK ANALYSIS OF PROJECT SCHEDULE

The project schedule is the core of the project planning. As the time needed to complete a project activity is hard to estimate, scheduling a project is extremely difficult task. In the software project development process it is very obvious that risk scheduling is one of the most significant disciplines that cannot be mastered by anyone who normally has project risk scheduling responsibilities. So, evaluating risks to the schedule is complex. Once the activities duration ranges have been determined, the schedule risk analysis can determine the risk during the project schedule. Normal risk schedules are designed using single point approximations of activity durations. When the uncertainty of activity durations is taken into account, the duration of a schedule path is likely to differ from that computed. To identify the risk involvement during scheduling, various tools and techniques have been discovered. Simulation is an important technique for schedule risk analysis.

5. STRATEGIES FOR SCHEDULE RISK ANALYSIS

Analysis of the project's schedule network can be used to identify risk factors associated with the project in the following ways:

- PERT
- Monte Carlo Simulation
- Simulation of PERT

5.1 Pert

PERT (Project Evaluation and Review Technique) was developed to take into account uncertainty of estimates of task durations. Instead of using a single estimate for the duration of each task, PERT requires three estimates - Optimistic Time, Most Likely Time, Pessimistic Time.

- **Optimistic Time (t_o)**: The optimistic time is the shortest possible time in which the activity can be completed.
- **Most Likely Time (t_m)**: The most likely time is the normal amount of time the activity would take.
- **Pessimistic Time (t_p)**: The pessimistic time is the longest time the activity could take if everything goes wrong.

The main objective of PERT is to find out the completion time for a particular event and to determine what are the chances of completing a job and the risk of not completing a job in time. In the network analysis, it is assumed that the time values are deterministic or variations in time are insignificant. It is difficult to get a reliable time estimate because the technology is changing rapidly. Time values are subject to chance variations [39].

Each activity is specified by its starting node, finishing node, and three time estimates. The risk index for the activity network can be computed as -

1. **Compute mean time (μ_k) and variance (σ_k^2) for each activity** - Mean time and variance of each activity/ node is computed on basis of Optimistic, Most Likely, Pessimistic timings of completion of project estimated by software development experts. The mean time for each activity can be approximated using the following weighted average:

Mean time = (Optimistic + 4 * Most likely + Pessimistic)/6 $\boldsymbol{\mu_k = (t_o + 4t_m + t_p)/6}$

The variance is given by:
Variance = [(Pessimistic - Optimistic) / 6]²

$$\boldsymbol{\sigma_k^2 = ((t_p - t_o)/6)^2}$$

2. **Determine the critical path and critical activities through network** - Critical path is the longest path through the network. The whole project falls behind schedule if something falls behind schedule on the critical path. Critical activities are the activities that lie on the critical path.

3. **Estimate the probability of risk during project completion** - (i) Calculating the z values - Given a

scheduled time (ST) for completing the project, the z value can be computed as -

$$\mathbf{z = \frac{\underline{(Scheduled\ time - \sum mean\ time\ of\ critical\ activities)}}{\sqrt{\sum variance\ of\ critical\ activities}}}$$

(ii) Converting z values to probabilities - The z value can be converted to probability of risk of not completing the project on time by using standard normal probability table or graph.

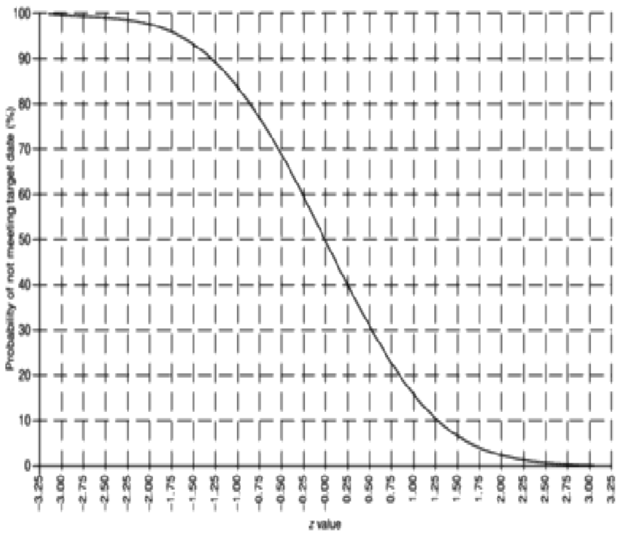

Fig. 2: The probability of obtaining a value within Z standard deviations of the mean for a normal distribution

5.1.1 Benefits

- PERT explicitly defines and makes visible dependencies (precedence relationships) between the activities of the schedule network.
- PERT facilitates identification of the critical activities and critical path.
- PERT provides for potentially reduced project duration due to better understanding of dependencies.

5.1.2 Limitations

- There can be hundreds or thousands of activities and individual dependency relationships.
- When the PERT charts become unwieldy, they are no longer used to manage the project.

5.2 Monte Carlo Simulation

As an alternative to the PERT technique, and to provide flexibility in specifying activity durations, Monte Carlo simulation techniques can be used to evaluate the risks of not achieving deadlines. The basis of this technique is to calculate activity times for a project network a large number of times, each time selecting activity times randomly from a set of estimates. The Monte Carlo method thus produces range of estimates with associated probabilities.

For example, In the COCOMO II estimation model, equation that relates estimated schedule to estimated effort is of the form:

$$\mathbf{S = c * (E)^{d}}$$

where E is estimated effort in staff-months, S is the estimated schedule in months, c and d are constants derived from historical data. The Monte Carlo technique can be used produce estimates of the probabilities of achieving various project milestones, including the completion milestone as depicted in Fig. 3.

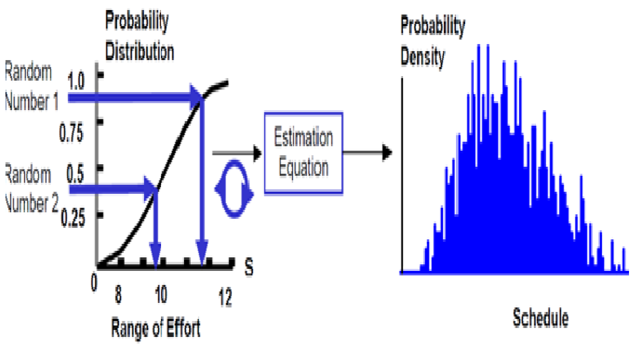

Fig. 3. A depiction of Monte Carlo estimation [40]

5.2.1 Benefits

The primary advantage of using Monte Carlo simulation in projects is that it is an extremely powerful tool to understand and quantify the potential effects of uncertainty of the project. Without the consideration of uncertainty in both project schedules and budgets, the project manager puts oneself at risk of exceeding the project targets. Monte Carlo simulation aids the project manager in quantifying and justifying appropriate project reserves to deal with the risk events that will occur during the life of the project.

5.2.2 Limitations

The primary drawbacks of Monte Carlo simulation is high use of computing power and the amount of time and resources spent to complete the simulation activity. A lack of easy-to use software to to run complex simulation against project schedules was also a problem.

5.3 Simulation of PERT

Mean time (μ_k) and standard deviation (σ_k) for each activity/node are evaluated on the basis of emperical data available (greater than 50) from different software development houses for a particular sector. For eg- banking sector. the timings for completion of each activity/node can be computed by using Box- Muller transformation.

$$\mathbf{t_k = s * \sigma_k + \mu_k}$$

where σ_k and μ_k are the standard deviation and mean, respectively, for the k_{th} activity and s is the desired sample from the standardized normal distribution.

$$\mathbf{s = sqrt\ (- 2\ ln\ (rn1)\ )\ cos\ (2\ pi * rn2)}$$

Where (rn1, rn2) is a pair of random numbers in the range (0, 1).

Using simulation of PERT, one can compute critical indexes for each activity/node and thus accordingly due importance can be given to that particular activity/node. this will minimize the risk factor involved in each and every activity and it will enable software development houses to develop the project in time.

6. CONCLUSION

Project failures are the result of multiplicity of risks inherent in software project environment. When risks are not managed properly, they leave projects vulnerable to factors that can cause major rework, major cost or schedule over-runs. Risk scheduling is a difficult discipline. Analysis of the project's schedule network can be used to identify risk factors associated with the project. In this paper, we have reviewed different strategies for evaluating risks to the project schedule.

7. ACKNOWLEDGMENTS

Sincere Thanks to HCTM Technical Campus Management, Kaithal, Haryana, India for their constant encouragement.

8. REFERENCES

[1] Abdullah Al Murad Chowdhury and Shamsul Arefeen (2011): 'Software Risk Management: Importance and Practices', IJCIT, Volume 02, Issue 01.

[2] Aurangzeb Khan, Dr. Farooque Azam and Muhammad Shoaib Zafar (2011): 'Managing Requirement Risks in Global Software Development', International Journal of Computer Applications (0975 – 8887) Volume 33– No.7.

[3] Averill M. Law (2011): 'How to Select Simulation input Probability Distributions', Proceedings of the Winter Simulation Conference.

[4] Ayse Kucuk Yilmaz and Triant Flouris (2010): 'Managing corporate sustainability: Risk management process based perspective', African Journal of Business Management Vol.4 (2), pp. 162-171.

[5] Barry W. Boehm (1989): 'Software Risk Management', Les Alamitos, CA, IEEE Computer Society.

[6] Bob Hughes and Mike Cotterell (2004): 'Software Project Management', Tata McGraw-Hill.

[7] Boehm, B.W. (1991): 'Software risk management principles and practices', IEEE Software 8 (1), 32–41.

[8] Dhirendra Pandey, Ugrasen Suman and A. K. Ramani (2011): 'Security Requirement Engineering Issues in Risk Management', International Journal of Computer Applications (0975 – 8887) Volume 17– No.5.

[9] Dragan Milosevic, Peerasit Patanakul (2005): 'Standardized project management may increase development projects success', International Journal of Project Management, 181–192.

[10] Gemmer, A. (1997): 'Risk management: moving beyond process', Computer 30 (5), 33–43.

[11] Gluch, D. P. (1994): 'A Construct for Describing Software Development Risks', Software Engineering Institute.

[12] Gupta, D. Sadiq, M. (2008): 'Software Risk Assessment and Estimation Model', IEEE International Conference on Computer Science and Information Technology, 963-967.

[13] Haneen Hijazi, Thair Khdour and Abdulsalam Alarabeyyat (2012): 'A Review of Risk Management in Different Software Development Methodologies', International Journal of Computer Applications (0975 – 8887) Volume 45– No.7.

[14] Hassan Mathkour, Ghazy Assassa, and A. Baihan (2008): 'A Risk Management Tool for Extreme Programming', International Journal of Computer Science and Network Security, Vol. 8 No.8.

[15] Iqbal Felani (2012): 'Developing Objective-Quantitative Risk Management Information System', Proceedings of the World Congress on Engineering.

[16] Dr.-Ing. Tilo Nemuth (2008): 'Practical Use of Monte Carlo Simulation for Risk Management within the International Construction Industry', Proceedings of the 6th International Probabilistic Workshop, Darmstadt.

[17] Jakub MILER and Janusz GÓRSKI (2004): 'Risk Identification Patterns for Software Projects', Foundations of Computing and Decision Sciences Vol. 29, No. 1-2, pp. 115-13.

[18] James G. March and Zur Shapira (1987): 'Managerial Perspectives on Risk and Risk Taking', Management Science Vol. 33, No. 11.

[19] James H. Lambed, Nicholas C. Matalas, Con Way Ling, Yacov Y. Haimes, and Duan Li (1994): Selection of Probability Distributions in Characterizing Risk of Extreme Events, Risk Analysis, Vol. 14, No. 5.

[20] Jerry Banks Marietta, Georgia (1999): 'Discrete Event Simulation', Proceedings of the Winter Simulation Conference pp. 7-13.

[21] Jiang, J.J., Klein, G. and Discenza, R. (2001): 'Information system success as impacted by risks and development strategies', IEEE Transactions on Engineering Management 48 (1), 46–55.

[22] Jiang, J.J., Klein, G. and Means, T.L. (2000): 'Project risk impact on software development team performance', Project Management Journal 31 (4), 19–26.

[23] Karel de Bakker, Albert Boonstra, and Hans Wortmann (2009): 'Does risk management contribute to IT project success? A meta-analysis of empirical evidence', International Journal of Project Management.

[24] Kwan-Sik Na, Xiaotong Li, James T. Simpson and Ki-Yoon Kim (2004): 'Uncertainty profile and software project performance: A cross-national comparison', The Journal of Systems and Software, 155–163.

[25] Kwan-Sik Na, James T. Simpson, Xiaotong Li, Tushar Singh, Ki-Yoon Kim (2007): 'Software development risk and project performance measurement', The Journal of Systems and Software 80, 596–605.

[26] Lazaros Sarigiannidis, Prodromos D. Chatzoglou (2011): 'Software Development Project Risk Management: A New Conceptual Framework', Journal of Software Engineering and Applications, 4, 293-305.

[27] Lazaros Sarigiannidis, Prodromos D. Chatzoglou (2011): 'Software Development Project Risk Management: A New Conceptual Framework', Scientific Research, Vol.4 No.5, May 2011, PP. 293-305.

[28] Linda Westfall: 'Software Risk Management', The Westfall Team PMB 383, 3000 Custer Road, Suite 270 Plano, TX 75075

[29] Liu Jun, Wang Qiuzhen and Ma Qingguo (2011): 'The effects of project uncertainty and risk management on IS development project performance: A vendor perspective', International Journal of Project Management, 923–933.

[30] Malaya Kumar Nayak, Dr Sanghamitra Mohanty (2012): 'Schedule Risk Analysis of ICT Infrastructure Projects', International Journal of Computer Applications (0975 – 8887) Volume 38– No.5.

[31] Malaya Kumar Nayak, Sanghamitra Mohanty and Rachna Soni (2011): 'People and Skills: A Potential Risk in ICT Infrastructure Management', International Journal of Software Engineering and Its Applications Vol. 5, No. 3.

[32] Ms Manisha.Ingle, Dr.Mohommad Atique, Prof. S. O. Dahad (2011): Risk Analysis Using Fuzzy Logic, International Journal of Advanced Engineering Technology, Vol.II, Issue III, 96-99

[33] Mei-yuan Wang, Yao-bin Lu, Jin-long Zhang, Software outsourcing risk management (2006): 'Establishing outsourcee evaluation item systems', Journal of Zhejiang University Science A, Volume 7, Issue 6, pp 1092-1098.

[34] M.S. Rojabanu and Dr. K. Alagarsamy (2010): 'A Conceptual Model for the reduction of Time based risk in the Software Project Management', International Journal of Computer Applications (0975 – 8887) Volume 3 – No.5.

[35] Paul Clarke a, Rory V. O'Connor (2012): 'The situational factors that affect the software development process: Towards a comprehensive reference framework', Journal of Information Software and Technology, Vol. 54, Issue 5, pp. 433-447.

[36] P. K. Suri, Bharat Bhushan, and Ashish Jolly (2009): 'Simulator for Risk Assessment of Software Project based on Performance Measurement', International Journal of Computer Science and Network Security, Vol.9 No.6.

[37] Dr P K Suri and Manoj Wadhwa (2007): 'Identification and Assessment of Software Project's Risk', International Journal of Computer Science and Network Security, Vol.7 No.8.

[38] P.K. Suri, Pallavi Ranjan (2012): 'Comparative Analysis of Software Effort Estimation Techniques', International Journal of Computer Applications (0975 – 8887) Volume 48– No.21.

[39] Dr. P.K.Suri and Pallavi Ranjan (2012): 'Estimating the Probability of Project Completion by SIM_DEL Estimator', International Journal of Computer Science and Information Technologies, Vol. 3 (4), 4938-4945.

[40] Richard E. (Dick) Fairley (2008): 'Risk Management for Software Projects', IEEE Computer Society.

[41] Rodrigues, A. (1996): 'The role of system dynamics in project management', International Journal of Project Management, Vol. 14, No. 4, pp. 213–220.

[42] Roger L. Van Scoy (1992): 'Software Development Risk: Opportunity, not problem', Technical report.

[43] Ronald P. Higuera and Yacov Y. Haimes (1996): 'Software Risk Management', Technical Report CMU/SEI-96-TR-012, Software Engineering Institute.

[44] Severin V. Grabski., Stewart A. Leech., Bai Lu.: 'Risks and Controls in the Implementation of ERP Systems', The International Journal of Digital Accounting Research Vol. 1, No. 1, pp. 47-68

[45] Sneh Prabha and R.L.Ujjawal (2011): 'Software Risk Evaluation and Assessment using Hybrid Approach', Proceedings published in International Journal of Computer Applications.

[46] Undram Chinbat, Soemon Takakuwa (2009): 'using Simulation Analysis for Mining Project Risk Management', Proceedings of the Winter Simulation Conference.

[47] Vinay Kumar Nassa and Sri Krishan Yadav (2012): 'Project Management Efficiency using Soft computing and Risk Analysis', International Journal of Computer Applications (0975 – 8887) Volume 50 – No.16.

[48] Yacoub, S.M. and Ammar, H.H., (2002): 'A methodology for architecture level reliability risk analysis', IEEE Transactions on Software Engineering 28 (6), 529–547.

[49] Y.H. Kwak and J. Stoddard (2004): 'Project risk management: lessons learned from software development environment', Technovation.

[50] Young Hoon Kwak and Kenneth Scott LaPlace (2005): 'Examining risk tolerance in project-driven organization', Technovation, 691–695.