

Software Maintenance Scenarios Simulation with Fuzzy Cognitive Maps

Cristina Lopez
School of Engineering
University Pablo de Olavide
Seville, Spain
clopvar@upo.es

Jose L. Salmeron
School of Engineering
University Pablo de Olavide
Seville, Spain
salmeron@acm.org

Sebastian Lozano
School of Engineering
University of Seville
Seville, Spain
slozano@us.es

Abstract—Successful performance of software, and indeed Enterprise Resource Planning (ERP), depends on proper system maintenance. For this reason, firms should follow a maintenance process that drives the ERP system toward success. However, in general terms, ERP maintenance managers do not know what conditions they should target to successfully maintain their ERP systems. Moreover, numerous risks threaten ERP maintenance, but they are normally dealt trusting on experience. To address this requirement, we have built a FCM-based tool that helps ERP teams to simulate the impact of risks on ERP maintenance.

Keywords: *Simulation; Software maintenance; ERP; Fuzzy Cognitive Maps; Risks Simulation*

I. INTRODUCTION

Enterprise Resource Planning (ERP) is one the most common and critical systems implemented by firms around the world. These systems allow the modeling, automation and integration of company business processes, grouping all data into a single database and providing relevant and updated information for decision-making and control.

Many firms have implemented ERP packages to fully integrate, standardize and coordinate their processes. These systems are more or less customized to fit them to the adopting company through the implementation process. But when the ERP is operative, practitioners undergo changes in the system. These modifications are carried out to correct or prevent failures, migrate the ERP to new releases and align the application regarding the specific business requirements or ERP users' expectations.

If the company does not properly maintain the ERP system, failures will arise, performance will decrease and the expected benefits will not be obtained [1]. This may even lead to early retirement of the ERP application. Hence, ERP maintenance becomes a key process in the post-implementation stage.

Given the significant risks inherent in ERP maintenance and the serious consequences that may follow, it is necessary to manage all risks properly. However, research on this particular issue remains inadequate. Previous studies neither indicate which conditions must be satisfied to support successful ERP maintenance, nor identify the risks that may threaten maintenance.

Accordingly, we assess the effects of risks on the success of ERP maintenance through a systematic approach similar to that proposed elsewhere. With this in mind, we identify risks to ERP maintenance success. Finally, we built a Fuzzy Cognitive Map to model risk effects on ERP maintenance success and simulate distinct scenarios.

Section 2 presents a literature review of theoretical background about ERP maintenance and risk. Section 3 presents the fundamentals of Fuzzy Cognitive Maps. Moreover, this section presents key aspects of the tool. Section 4 describes the process for building a Fuzzy Cognitive Map to model risk in ERP maintenance. Section 5 studies the results of the FCM simulation and the risks impacts on ERP maintenance. For this purpose, we define two scenarios and simulate them using the Fuzzy Cognitive Map. Finally, Section 6 offers the conclusions.

II. BACKGROUND

A. Enterprise Resource Planning

Enterprise Resource Planning (ERP) is one the most adopted enterprise systems. An ERP is a large commercial off-the-shelf configurable application that integrates the core enterprise business functions.

An ERP can be defined as a single software system to support the complete integration of information from all functional areas of a company by means of a single database that is accessible through a unified interface and channel of communication.

The ERP life cycle begins when top management decides to install the software. Following this decision, numerous tasks are carried out to implement the system. However, the ERP life cycle is not finished as soon as the application is operative and the ERP implementation stage has been concluded. At that moment, ERP enters into the post-implementation stage, which continues until its retirement.

B. ERP maintenance

ERP adopters have to perpetually maintain their ERPs until the end of its life span. However, frequent ERP modifications cause high risks and costs, which impact on software performance [2]. In such a case, the daily activity of the firm

would be hindered. Hence, ERP maintenance is considered a critical process, which needs to be managed carefully.

ERP maintenance can be defined as the process addressed to identify and perform changes required for improving or maintaining system performance through the ERP post-implementation stage. The maintenance team implements each maintenance request by means of projects.

In the literature, various papers propose models to support ERP maintenance [3], [4]. Despite this, models aware of risk have not been developed. For this reason, the aim of our research was to develop a formal model for managing ERP maintenance risks.

C. Risks within ERP maintenance

ERP system maintenance is a large-scale, unstructured and highly complex undertaking [5]. In some cases, it even requires the use of unfamiliar technologies and tools. Hence many controllable and uncontrollable risks can affect its outcome.

A risk is an event characterized by uncertainty because it may or may not occur. If this risk turns into a real problem, it may cause dangerous failures. In such case, the project performance and the maintained ERP will be seriously damaged. Therefore, risk management becomes a crucial process to ensure ERP success.

A proper risks management helps practitioners to be aware of the real situation of the project, their problematic aspects and potential causes existing of ERP project failure. They will thus resolve more efficiently the threats' projects. Otherwise, improper risks management lead the carrying out of evitable error and the appearance of potential problems, which make the achievement ERP maintenance project outcomes difficult. Therefore, active risks management is key to reach ERP maintenance projects success.

In this way, the literature provides numerous methods, checklists, analytical frameworks, risk assessment tools and risk response strategies to help project managers to handle risk factors more effectively [6]. These studies mainly focus on classic software projects. However, different risks affect ERP project success, because classic software projects and ERP projects are not similar [3]. This has encouraged the appearance of ERP project risk studies in the literature. However, these studies focus on the implementation stage.

Various research efforts have identified risk in the context of ERP projects. Sumner [7] identified, described and categorized the risk factors associated with ERP implementation. Moreover, Sumner also marked which risks are unique in ERP projects. Huang et al. combined a multi-criteria methodology and the Delphi method to identify and prioritize risks [8]. By contrast, post-implementation stage risk research is scarce in the literature.

III. FUZZY COGNITIVE MAPS

FCMs describe a cognitive map model with two significant characteristics. First, causal relationships between nodes have different intensities represented by fuzzy numbers. A fuzzy number is a quantity whose value is uncertain, rather than

exact. It can be thought of as a function whose domain is usually the interval between 0 and 1 (or -1 and 1), inclusive. Each numerical value in the interval represents the degree of membership in a fuzzy set, where 0 is non-membership and 1 represents full membership. The second characteristic is that the system is dynamic – it evolves with time. The system involves feedback, and a change in a concept node may affect other concept nodes, which in turn can impact the node initiating the change. Feedback plays a prominent role in FCMs by propagating causal influences along complicated pathways.

Moreover, FCMs provide an intuitive, yet precise, way of expressing concepts and reasoning about them at their natural level of abstraction. By transforming decision modeling into causal graphs, decision makers with no technical background can understand all of the components in a given situation. In addition, with an FCM, it is possible to identify and consider the most relevant factor that seems to affect the expected target variable. For these reasons, we think that FCMs are a worthy tool.

A. FCM fundamentals

A FCM is a digraph composed of nodes and edges. The nodes (x_i) are the most relevant dynamic variables or factors describing a target real-world dynamic system. These may represent concepts such as costs, diseases, medication, investment, or marketing strategy, to name a few. For any context, the nodes are a set of entities that are relevant to the domain of study.

Relationships between nodes are represented by directed edges. An edge linking two nodes can model the influence of the causal variable on the effect variable [9]. In addition, edges also indicate the type of connection, incorporating either a plus (+) or a minus (-) sign. A positive relationship between two factors indicates that an increase or decrease in a causal variable causes the effect variable(s) to change in the same direction. When the relationship is negative, the change in the effect variable is in the opposite direction (i.e., an increase in the cause variable causes a decrease in the effect variable).

Because FCMs are hybrid methods that mix fuzzy logic and neural networks [10], each cause is assessed by its intensity, $w_{ij} \in [-1, +1]$, where i is the pre-synaptic (causal) node and j is the post-synaptic (effect) node one.

FCMs are dynamic systems involving feedback whereby change in a node may affect other nodes, which in turn can affect the node initiating the change. The analysis begins with the design of the initial vector state (C^0), which represents the initial value of each variable or concept (node). The initial vector state with n nodes is denoted as:

$$C^0 = (C_1^0 \quad C_2^0 \quad \dots \quad C_n^0) \quad (1)$$

The new values of the nodes are computed in an iterative vector-matrix multiplication process with an activation function, which is used to monotonically map the node value into a normalized range [0,1]. The sigmoid is the most common function [11] when the concept (node) value maps onto the range [0,1].

B. FCM dynamics

The vector state C_i^{t+1} at the instant $t+1$ would be computed as:

$$C_i^{t+1} = f \left(\sum_{\substack{j=1 \\ j \neq i}}^n C_j^t \cdot w_{ij} \right) \quad (2)$$

where $f(x)$ is the transformation function.

The state changes along the process. Usually, two kinds of transformation functions are used in FCM dynamics. The first one is the unipolar sigmoid:

$$f(x) = \frac{1}{1 + e^{-\lambda x}} \quad (3)$$

The second is the hyperbolic tangent:

$$f(x) = \tanh(x) = \frac{e^{\lambda x} - e^{-\lambda x}}{e^{\lambda x} + e^{-\lambda x}} \quad (4)$$

Both functions use λ as a constant for function slope (degree of fuzzification). The FCM designer has to specify the lambda value. For large values of lambda (e.g., $\lambda \geq 10$) the sigmoid approximates a discrete function that maps its results to interval (0,1); for smaller values of lambda (e.g., $\lambda \leq 1$) the sigmoid approximates a linear function; while values of lambda closer to 5 provide a good degree of fuzzification in the [0,1] interval [11].

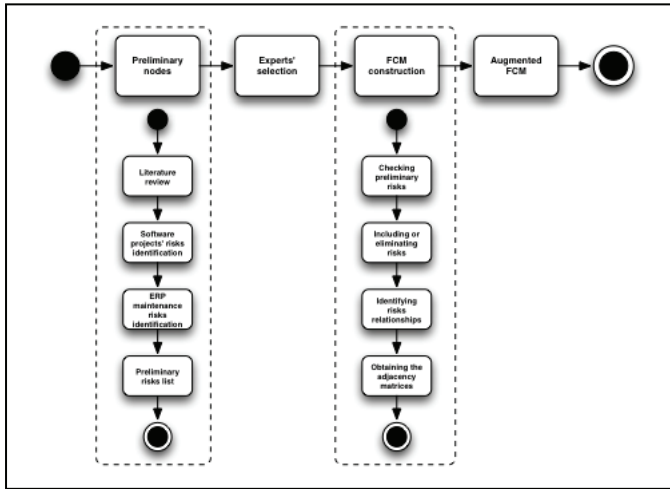


Figure 1. FCM construction process

The FCM inference process finishes when stability is reached. The final vector state shows the effect of the change in the value of each node on the FCM. After the inference process, the FCM reaches one of three states following a number of iterations. It settles down to a fixed pattern of node values, the so-called hidden pattern or fixed-point attractor.

Alternatively, the state can continue cycling between several fixed states, known as a limit cycle. With a continuous function, a third possibility is a chaotic attractor. This occurs when, instead of stabilizing, the FCM continues to produce different results (state vector values) for each cycle. In this case, the technique fails to offer a useful outcome for risk analysis.

IV. FCM CONSTRUCTION

The aim of this study was to create a FCM to simulate the risks' impact on ERP maintenance. We carried out the steps indicated in Figure 1 to build the FCM model. Firstly, we identified the preliminary nodes to help the experts create their FCMs. To do so, we performed a broader literature review on software projects risks and ERP projects. We thus achieved an initial large risks list. But, the initial list does not solely gather ERP maintenance risks.

So, we carefully analyzed the initial list to identify ERP maintenance risks. Subsequently, those risks affecting ERP maintenance were renamed, while the other risks were eliminated. In this way, we obtained the preliminary risks nodes of FCM.

The following activity consisted of selecting experts who were willing to participate in our research. In no case, did we select the experts because they were easily accessible. The main selection criterion considered was profound knowledge in the ERP maintenance and absence of conflicts of interest. A total of 10 leading experts decided to take part with us in this study. These experts belong to different professional scales. Moreover, they had participated or participate in the maintenance of ERP solutions supplied by different vendors. Therefore, we achieved the making up of a heterogeneous group of experts.

Thirdly, each expert individually built an FCM. To help them, we sent the list of preliminary risks nodes, which were only indicative. So, they checked them and eliminated those risks nodes, which do not affect ERP maintenance. Moreover, those experts, who concluded that we had not identified all ERP maintenance risks, added the non-identified risk(s) node(s) to the initial list. Subsequently, the experts state the relationship between the nodes. They also pointed out the risks impacts on ERP performance. We thus achieved one adjacency matrix for each expert.

We finally constructed the augmented matrix (A^*) adding the adjacency matrix of each one of them [10]. The elements' aggregation depend on if there are or are not common nodes. The experts' adjacency matrix had common nodes. So, the elements (w_{ij}^*) in the augmented matrix are computed by [12]:

$$w_{ij}^* = \frac{\sum_{k=1}^n A_k}{n} \quad (5)$$

$$= \frac{\sum_{k=1}^n w_{ij}^k}{n}$$

n being the number of FCMs added, one per expert. The A^* obtained is the following:

$$\begin{aligned}
& \begin{matrix} 0 & .06 & 0 & 0 & .07 & 0 & 0 & 0 & .12 & .04 & 0 & 0 & -.05 \\ 0 & 0 & 0 & 0 & 0 & .03 & 0 & 0 & 0 & 0 & .01 & 0 & -.108 \\ 0 & 0 & 0 & .255 & .325 & .185 & .05 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & .15 & 0 & .105 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & .04 & 0 & .625 & 0 & .17 & 0 & 0 & 0 & 0 & -.018 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & .06 & 0 & 0 & 0 & 0 & -.109 \\ A^* = 0 & 0 & 0 & 0 & 0 & 0 & .23 & 0 & .08 & 0 & 0 & 0 & -.038 \\ 0 & 0 & 0 & 0 & 0 & .11 & 0 & 0 & 0 & 0 & .08 & 0 & -.029 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -.025 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -.032 \\ 0 & .14 & 0 & 0 & 0 & .28 & .315 & .035 & 0 & .02 & .02 & 0 & -.134 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{matrix} \\
& (6)
\end{aligned}$$

Figure 2 shows the FCM obtained, representing how existing risks in the implementation of maintenance request impacted on ERP performance.

V. FCM SIMULATION

Firstly, we defined what-if scenarios at the instant $t=0$. In other words, the values of all variables of the FCM are represented in an $1 \times n$ initial state vector C^0 .

We formulate several questions as following: which risks will impact the strongest on ERP performance during the system maintenance: risks derived from external or internal agents or both at the same time? How will these impacts on the rest of the risks? To answer the previous questions, we created three Scenarios and simulated them:

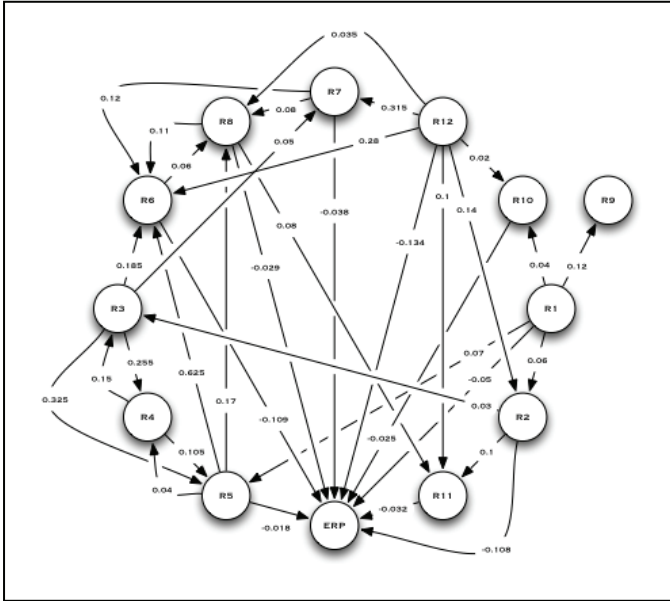


Figure 2. FCM-based ERP maintenance risks model

In Scenario 1 ($C_{S_1}^0$), we activated the risk derived from the external agents. In Scenario 2 ($C_{S_2}^0$), we activated risks derived from the internal agents.

$$C_{S_1}^0 = (0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0) \quad (7)$$

$$C_{S_2}^0 = (0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0) \quad (8)$$

Subsequently, we simulated these Scenarios computing A^* and $C_{S_i}^0$. Specifically, Scenarios were transformed using hyperbolic tangent function (4). We select this activation function (7) because the values of the nodes can fall within the range $[-1, 1]$, 5 being the value assigned to λ .

The simulation finishes when the vector state reaches the stability (at a fixed point or limit cycle) or a chaotic behavior. In this research, both scenarios reached the stability. Table 1 presents the results obtained simulating each Scenario.

TABLE 1. Scenarios simulation results

Node	Description	Scenario 1	Scenario 2
R1	Continuous organizational changes	0	0
R2	Requirements incomplete and/or unclear	0.023556521	0
R3	Instability in maintenance team members	0	0.50123746
R4	Lack group spirit and attitude in maintenance team members	0	0.575731388
R5	Inadequate mix of team skills	0	0.637508649
R6	Quality of original programming	0.255256472	0.830216721
R7	Methodology and maintenance process are not defined	0.030705553	0.327804561
R8	Maintenance tasks are excessively hard	0.226292501	0.641248801
R9	Project milestones cannot be defined	0	0
R10	Use of inadequate technology for system testing	0.012737097	0
R11	Wrongly-fit system with pre-existing applications	0.229341662	0.442749349
R12	Lack active participation by external parties	0.005477003	0
ERP		-0.308378881	-0.619552233

The findings of Scenario 1 show that the impact of the unsuitable involvement of external agents in the project was positive and varied from null to low on the rest of the risks. The quality of the original programming (R6) and fit software with pre-existing applications (R11) were the ones highest affected by R12, both technical risks. The intensity of influence of R12 on ERP performance was higher than on the risks. This specifically was negative and slightly moderate. Therefore, R12 directly affected ERP performance instead of indirectly, that is, by means of other risks.

The results of the simulation of Scenario 2 show that the effect of risks derived from internal agents in the project was also positive. However, in this simulation, the effects on ERP maintenance risks were higher by far. In fact, the intensity of impacts was usually high. The intensities of impacts were from

2 to 6 times higher in Scenario 2 with respect to Scenario 1. Specifically, the influence of R3, R4 and R5 on other risks was from null to strong. As well as Scenario 1, R6 was the highest impacted risk. But the intensity of impact of risks activated in Scenario 2 was three times greater than the risk activated in Scenario 1.

The intensity of influence of risks activated in Scenario 2 on ERP performance was two times greater than the risk activated in Scenario 1. This was negative and slightly strong. R3, R4 and R5 impact directly less on ERP performance than on some other risks. Therefore, risks derived from internal agents directly and indirectly affect ERP performance.

VI. CONCLUSIONS

ERP maintenance is necessary to maintain or improve the usability and system performance. With this in mind, risky and complex maintenance projects follow one another continuously through the ERP life span.

However, numerous risks can arise during the implementation of maintenance requests. To prevent these risks negatively affecting system performance, practitioners have to manage them properly. But, in many cases, they deal these risks with intuitively and inefficiently.

In order to avoid this, this research aims at supporting practitioners work managing ERP maintenance risks. To do so, we decided to create an FCM. This identifies ERP maintenance risks as well as the existing connections between them. In addition, at any time the practitioner can easily modify the model, adding or eliminating further variables and/or relationships, to faithfully represent the phenomena.

The dynamic analysis of the FCM will help practitioners to take more efficient measures to minimize ERP maintenance risks. Static analysis specifically indicates which risks have more influence on other risks or impact stronger on ERP performance.

In contrast, the dynamic analysis of the FCM allows ERP practitioners to predict the effects of risks activated in the implementation of maintenance requests on ERP performance over time.

The result of dynamic analysis reveals that the risks derived from team maintenance have strongest impact on ERP performance. Hence, in order to achieve successful ERP maintenance, managers should create teams made up of highly qualified professionals in ERP technology, reduce the instability in maintenance team members, as well as improving

the motivation and training of the members of maintenance teams.

Looking to the future, it is necessary to undertake further research on the existing risks in ERP maintenance. To manage such risks most effectively, managers should know which risks are most important to ERP maintenance success. Therefore, research that answers this question is required. The development of techniques to forecast the risks appearance within ERP maintenance projects is also very worthwhile.

ACKNOWLEDGMENT

The authors wish to thank the Spanish Ministry of Science and Innovation (MICINN-ECO2009.12853) and the Andalusian Regional Government (CICE-P07-SEJ-03080) for their financial support.

REFERENCES

- [1] C.S.P. Ng, G. Gable and T. Chan, "A ERP client-benefits-oriented maintenance taxonomy", *Journal of Systems and Software*, vol. 64, no. 2, pp. 87-109, 2002.
- [2] D. Chand, G. Hachey, J. Hunton, V. Owosho and S. Vasudevan, "A balanced scorecard based framework for assessing the strategic impacts of ERP systems", *Computers in Industry*, vol. 56, no. 6, pp. 558-572, 2005.
- [3] C.S.P. Ng, G.G. Gable and T. Chan, "An ERP maintenance model", *Proc. 36th Annual Hawaii International Conference on System Science*, pp. 1-10, 2003, doi:10.1109/HICSS.2003.1174609.
- [4] O.B. Kwon and J.J. Lee, "A Multi-agent intelligent system for efficient ERP maintenance", *Expert System with Applications*, vol. 21, no. 4, pp. 191-202, 2001.
- [5] R. Mookerjee, "Maintaining enterprise software applications", *Communications of the ACM*, vol. 48, no. 11, pp. 75-79, 2005.
- [6] P.L. Bannerman, "Risk and risk management in software projects: A reassessment", *Journal of Systems and Software*, vol. 81, no. 12, pp. 2118-2133, 2008.
- [7] M. Sumner, "Risk factors in enterprise-wide/ERP projects", *Journal of Information Technology*, vol. 15, no. 4, pp. 317-327, 2000.
- [8] Sh.-M. Huang, I.-Ch. Chang, Sh.-H. Li and M.-T. Lin, "Assessing in ERP projects: identify and prioritize the factors", *Industrial management & data systems*, vol. 8, no. 8, pp. 681-688, 2004.
- [9] G. Xirogiannis and M. Glykas, "Fuzzy cognitive maps in business analysis and performance-driven change", *IEEE Transactions on Engineering Management*, vol. 51, no. 3, pp. 334-351, 2004.
- [10] B. Kosko, *Fuzzy Engineering*. Prentice Hall, 1996.
- [11] S. Bueno and J.L. Salmeron, "Benchmarking main activation functions in fuzzy cognitive maps", *Expert Systems with Applications*, vol. 36, no. 3, pp. 5221-5229, 2009.
- [12] J.L. Salmeron, "Augmented fuzzy cognitive maps for modelling LMS critical success factors", *Knowledge-Based Systems*, vol. 22, no. 4, 2009, pp. 275-278.