A Fuzzy Approach based on Dynamic Programming and Metaheuristics for Selecting Safeguards for Risk Management for Information Systems

E. Vicente, A. Mateos and A. Jiménez-Martín
Decision Analysis and Statistics Group, Departamento de Inteligencia Artificial, Universidad Politécnica de Madrid,
Campus de Montegancedo s/n, Boadilla del Monte, 28660 Madrid, Spain

Abstract: In this paper we focus on the selection of safeguards in a fuzzy risk analysis and management methodology for information systems (IS). Assets are connected by dependency relationships, and a failure of one asset may affect other assets. After computing impact and risk indicators associated with previously identified threats, we identify and apply safeguards to reduce risks in the IS by minimizing the transmission probabilities of failures throughout the asset network. However, as safeguards have associated costs, the aim is to select the safeguards that minimize costs while keeping the risk within acceptable levels. To do this, we propose a dynamic programming-based method that incorporates simulated annealing to tackle optimizations problems.

1 INTRODUCTION

There are several risk analysis and management methodologies for information systems (IS) that conform to International Organization for Standardization (ISO) standars, specifically the ISO 27000 family of standars. Some examples of these methodologies are MAGERIT, by the Spanish Ministry of Public Administrations (López Crespo et al., 2006); CRAMM (CCTA, 2003), by the Central Computing and Telecommunications Agency (UK); or NIST SP 800-30 (Stoneburner and Gougen, 2002), by the National Institute of Standard and Technology (USA).

These methodologies do not, however, consider uncertain valuations, but use precise values on different, usually percentage, scales. Boolean values are sometimes even used to indicate whether or not assets are dependent on each other regardless of the degree of such dependency. In no case is vague or imprecise information about the input parameters allowed. In our opinion, this is an important drawback of these methodologies.

In (Vicente et al 2013a) we proposed an extension of the MAGERIT methodology based on classical fuzzy computational models. This methodology includes the following milestones:

1. *Identification and Valuation of Assets*

   An asset is anything that is of value to the organization and therefore requires protection. A few data, information or business process assets often account for the total value of an organization's assets. These assets are called *terminal assets.* Other assets (*support assets* such as hardware, software, personnel, facilities, ...) are valuable insofar as they are beneficial to the terminal assets, and they inherit the terminal asset value, according to the resulting benefit. Thus, support assets have no intrinsic value; they take their value from terminal assets.

   The identified assets of the organization are then valued. Some assets may have a monetary value (how much money the organization would lose if this asset stopped working), whereas others require a qualitative assessment (if an asset stops working the losses would be very high, low, medium...).

   As mentioned above, the support assets inherit their values from terminal assets depending on how they influence each other. So, we have to determine the dependency relationships of the terminal assets with respect to support assets, and also dependency relationships between support assets.

2. *Threat Identification*

   A *threat* is an event that can trigger an incident in the organization, causing damage or intangible material loss to assets. Threats may be of natural or human, accidental or deliberate origin. Some threats can affect more than one asset. In such cases, threats can cause different impacts depending on what assets are affected. A detailed list of threats is available in Annex C of ISO IEC 27005. MAGERIT suggests two threat assessment measures: *degradation*, the damage that the threat can cause to the asset, and *frequency*, how often the threat materializes.

3. *Identification and Valuation of impact and Risk Indicators*

   It is then necessary to qualitatively identify the consequences and establish impact and risk indicators for the valued assets and threats. The impact of a threat on an asset is the product of the asset value multiplied by the respective degradation. Risk is the product of the impact of the threat multiplied by the respective frequency.

4. *Selection of Safeguards*

   Safeguards are measures for addressing threats. They can be procedures, personnel policies, technical solutions or physical security measures at the facilities. These safeguards can be *preventive*, if they reduce the frequency of threats; or *palliative*, if they reduce the degradation of assets caused by threats (López-Crespo et al., 2006).

As described below, experts use a linguistic term scale (see Figure 1 and Table 1) to represent asset values, their dependencies and the frequency and asset degradation associated with possible threats. Risk analysis computations are then based on the trapezoidal fuzzy numbers associated with linguistic terms.

However, direct assignment based on a rigid linguistic term scale is not always advisable since the expert has no say in the number of linguistic terms that the scale is to include and about the appearance of their associate trapezoidal fuzzy numbers. In that case we propose the use of the betting and lottery-based method for fuzzy probability elicitation described in (Vicente et al 2013c). Betting and lottery-based methods commonly used to assign probabilities can also be used to assign fuzzy probabilities (Savage, 1954; Finetti, 1964). In this section we briefly describe these methods and show how a fuzzy number representing the probability judgment can be extracted from experts.

Betting Method. For two selected monetary values $x > y$, the expert is given the option between either of the two following gambles:

- *b1:* If event A happens, then you win x\$. Otherwise, you lose y\$.
- *b2:* If event A does not happen, then you win y\$. Otherwise, you lose x\$.

If the expert has no preference for either bet, the respective expected utilities of both bets are equal, and it follows that $p(A) = x/(x+y)$. If the expert chooses one of the two gambles, then the expected utility of the selected gamble should be higher than for the rejected gamble. Then, the analyst has to update monetary values and offer the expert two new gambles. Thus, an interactive process is enacted until two alternative gambles are reached to which the expert is indifferent.

Lottery-based methods. For a given probability and monetary values x\$ and y\$, the expert is given the choice between the following lotteries:

- *l1:* If event A happens, then you win x\$. Otherwise, you lose y\$.
- *l2:* You win x\$ with probability p, or y\$ with probability $1-p$.

If the expert has no preference for either of the lotteries, then the respective expected utilities are equal, and it follows that $p(A) = p$. Otherwise, the expert must readjust the value p, keeping the same monetary values. This again generates an interactive process, enacted until a couple of lotteries are reached to which the expert is indifferent.

The betting and lottery-based methods assume that the expert is able to provide a specific value for the probability of an event. However, a more realistic scenario is where experts have an imprecise and vague idea of that value. Consequently, experts will have an interval rather than a precise value in mind at the point when they are indifferent to either bet or lottery, that is, for the lottery-based method there will be an interval $[a,c]$ such that if $p = [a,c]$, then the expert has no preference for either lottery $l1$ or $l2$. Similarly, the betting method can result in an interval of indifference $[b,d]$.

Current protocols for probability elicitation like the above recommend the use of several methods to test the consistency of the expert and the existence of bias. In this regard, the development of betting and lottery-based methods meets this recommendation and establishes the following:

- If $[a,c] \cap [b,d] = \varnothing$, then the expert's probabilistic judgment was inconsistent.
- If any of the intervals is contained in the other $[a,c] \subseteq [b,d]$ (or $[b,d] \subseteq [a,c]$), then we assume that the trapezoidal fuzzy number (b,a,c,d) (or (a,b,d,c)) designates the expert probabilistic judgment.
- If $[a,c] \cap [b,d] \neq \varnothing$, is uncountable, and none of the intervals is contained in the other, then, assuming that $a \leq b \leq c \leq d, (a,b,c,d)$ designates the expert probabilistic judgment.

Thus, we consider the set of trapezoidal fuzzy numbers with support in [0,1], TF[0,1], i.e.,

Table 1: Linguistic term scale.

Term	Trapezoidal fuzzy number
Very Low (VL)	(0, 0, 0, 0.05)
Low (L)	(0, 0.075, 0.125, 0.275)
Medium-Low (ML)	(0.125,0.275, 0.325, 0.475)
Medium (M)	(0.325, 0.475, 0.525, 0.675)
Medium-High (MH)	(0.525, 0.675, 0.725, 0.875)
High (H)	(0.725, 0.875, 0.925, 1)
Very High (VH)	(0.925, 1, 1, 1)

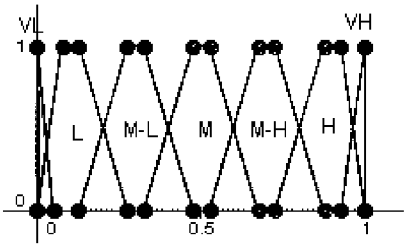

Figure 1: Linguistic term scale.

$\widetilde{A} = (a,b,c,d)$ with $0 \le a \le b \le c \le d \le 1$ and with a trapezoidal function in the vertices $(a,0),(b,1),(c,1),(d,0)$ (Chen, 1996; Chen and Chen, 2003; Chen and Chen, 2009; Vicente et al., 2013b). Note that $\mathbb{R}$ is a subset of TF[0,1] if we consider the injection (Vicente et al 2013a) $\phi : \mathbb{R} \hookrightarrow TF[0,1]$; $a \approx \phi(a) = (a,a,a,a) = \widetilde{a}$.

Consequently, the following operators proposed in (Xu et al 2010) accounting for trapezoidal fuzzy numbers will be used to make computations:

Given $\widetilde{A}_1 = (a_1,b_1,c_1,d_1)$, $\widetilde{A}_2 = (a_2,b_2,c_2,d_2) \in TF[0,1]$, then:

- $\widetilde{A}_1 \oplus \widetilde{A}_2 = (a_1 + a_2 - a_1a_2, b_1 + b_2 - b_1b_2, c_1 + c_2 - c_1c_2, d_1 + d_2 - d_1d_2)$ and
- $\widetilde{A}_1 \otimes \widetilde{A}_2 = (a_1a_2, b_1b_2, c_1c_2, d_1d_2)$.

$\oplus$ and $\otimes$ are two internal composition laws in TF[0,1] that verify the commutative and associative properties and both have a neutral element.

The assets of an IS are elements of value to the organization and therefore require protection (servers, files, personnel, facilities, hardware, software,...).

As cited before, these assets are interrelated (López Crespo et al, 2006), forming an acyclic graph, where just a few data, information items or business process assets often account for the total value of an organization's assets. These assets are called *terminal assets*. Other assets (*support assets*, such as hardware, software, personnel, facilities,...) are valuable insofar as they are beneficial to the terminal assets. In other words, the support assets inherit their values from terminal assets depending on how they influence each other, i.e., depending on the probability of that any failure in an asset being transferred to the terminal assets.

In general, we say asset A_j *directly depends* on asset A_i, denoted by $A_i \rightarrow A_j$, if a failure in asset A_i causes a failure in the asset A_j with any given probability. This probability is usually referred to as the *degree of direct dependency* of A_j with respect to A_i. Note that in this fuzzy adaptation the degrees of direct dependency between assets will be represented by linguistic terms, which have associated trapezoidal fuzzy numbers. We denote these degrees of direct dependency by $\widetilde{d(A_i,A_j)}$.

These dependencies form a directed acyclic graph (to terminal assets), so that there may be intermediate assets between any asset A_i and a terminal asset A_k which can propagate a fault generated in A_i through to the terminal A_k. Our aim then is to compute the transmission probability between A_i and A_k. This probability is called *degree of indirect dependency* between A_i and A_k, which is denoted by $\widetilde{D(A_i,A_k)}$ and can be computed as follows (Vicente et al 2013a).

We denote by $\mathbf{P}=\{P_1,...,P_s\}$ the set of paths in the network connecting A_i with A_k. These paths are a sequence of arcs connecting a sequence of vertices, such that the start vertex and the last vertex are A_i and A_k, respectively. Then,

A) If all assets, excluding A_i and A_k, in the paths in $\mathbf{P}$ are influenced by only one asset, then

$$\widetilde{D(A_i,A_k)} = \bigoplus_{j=1}^{s} \widetilde{D(A_i,A_k|P_j)} \qquad (1)$$

where $\widetilde{D(A_i,A_k|P_j)} =$

$\widetilde{d(A_i,A_{j_1})} \otimes \widetilde{d(A_{j_1},A_{j_2})} \otimes ... \otimes \widetilde{d(A_{j_n},A_k)}$ and $P_j : (A_i \rightarrow A_{j_1} \rightarrow A_{j_2} \rightarrow ... \rightarrow A_{j_n} \rightarrow A_k)$.

B) Otherwise, we assume that the first r paths in $\mathbf{P}$ are formed by assets (excluding A_i and A_k) influenced by only one asset, and the remaining $s-r$ paths include at least one asset simultaneously influenced by two or more assets. Then, for the r first paths, we proceed as in *A)*, and we denote by $\mathbf{S}$ the set including the $s-r$ remaining paths. We proceed with $\mathbf{S}$ as follows:

(i) Consider the set of non-terminal assets in $\mathbf{S}$ influenced by two or more assets, denoted by *I*, and the subset of *I* including assets uninfluenced by any other asset in *I*, denoted by *NI*.

(ii) We consider an asset A_r in *NI*. Then, we simplify the paths in $\mathbf{S}$ that include asset A_r making $A_i \rightarrow A_r \rightarrow ... \rightarrow A_k$, with $\widetilde{d(A_i,A_r)} = \widetilde{D(A_i,A_r)}$ (computed as in *A*).

(iii) Remove repeated paths from **S** and keep only one instance.

(iv) Build I and NI again from **S**.

(v) If NI is not empty, go to (ii). Otherwise, the algorithm finishes.

Let us denote the resulting set of paths by **S**$=\{P'_1,...,P'_m\}$ with $m \leq s-r$. Then, the degree of dependency of A_k regarding A_i is

$$\widetilde{D(A_i,A_k)} = \bigoplus_{j=1}^{r} \widetilde{D(A_i,A_k|P_j)} \bigoplus_{l=1}^{m} \widetilde{D(A_i,A_k|P'_l)}. \quad (2)$$

Once we have computed the degree of indirect dependency between all assets regarding the terminal assets, we can compute the accumulated values for non-terminal assets $\widetilde{v}_l$. These values usually have three components (ISO/IEC serie 27000):

1. *Availability*. How much damage would it cause if the asset is not available or cannot be used? This is a typical services inspection.
2. *Confidentiality*. How much damage would it cause if the asset is disclosed to someone it should not be? This is a typical data inspection.
3. *Integrity* How much damage would it cause if the asset is damaged or corrupt? This a typical data inspection. Data can be manipulated, be wholly or partially false, or even missing.

Therefore,

$$\widetilde{v}_{i_{(l)}} = \sum_{k=1}^{n} ((\widetilde{D(A_i,A_k)} \otimes \widetilde{v}_{k_{(l)}}) \quad (3)$$

where l denotes the lth component.

Once assets have been valueted, the next step in the risk analysis methodology is to identify possible threats and compute the corresponding impact and risk indicators for the IS.

Threats are characterized by how often the threat materializes (*frequency*) $\widetilde{f}$ and by the *degradation* $\vec{D} = (\widetilde{d}_1, \widetilde{d}_2, \widetilde{d}_3)$ that the threat can cause to the three asset components. Note again that the frequency and degradation levels will be selected by the expert from the linguistic term scale and, consequently, a trapezoidal fuzzy number will be associated with each of them.

Then, the *impact* of a threat on an asset A_j is

$$\widetilde{I}_{i_{(l)}} = \widetilde{d}_l \otimes \widetilde{v}_{i_{(l)}}, \quad (4)$$

and the *risk* to the asset is

$$\widetilde{R}_{i_{(l)}} = \widetilde{I}_{i_{(l)}} \otimes \widetilde{f}. \quad (5)$$

The results of these operations will be fuzzy numbers belonging to TF[0,1], which, generally, do not match up with the fuzzy numbers associated with the linguistic terms of the scale. Thus, a similarity function must be used to identify the most similar trapezoidal fuzzy number in the linguistic term scale to the fuzzy number output from computations.

Different similarity functions have been proposed by several authors (Chen and Chen 2003, Chen and Chen 2009, Gomathi and Sivaraman 2012, Xu et al 2010, Zhu and Xu 2012). In (Vicente et al 2013b) a new similarity function was proposed on the basis of the geometric distance between both fuzzy numbers, the distance between their centroids and/or the ratio between the common area and the joint area under the membership functions.

Following the risk analysis and management methodologies for IS, Section 2 deals with the selection of safeguards that can be enforced to reduce the transmission probability of a failure throughout the IS. The aim is to minimize costs while keeping the risk at acceptable levels. To do this, we propose a mixed technique based on dynamic programming and metaheuristics, specifically, simulated annealing.

2 SELECTION OF PREVENTIVE SAFEGUARDS

From equations (3), (4) and (5) and the algorithm for computing degrees of indirect dependency, we can derive the risk for the IS in each component l given a threat with frequency $\widetilde{f}$ and degradation $\widetilde{D} = \left\{\widetilde{d}_l\right\}_{l=1}^{3}$ in the support asset $\widetilde{A}_i$ as

$$\widetilde{R}_{i_{(l)}} = \sum_{k=1}^{n} D\widetilde{D(A_i,A_k)} \otimes \widetilde{v}_{k_{(l)}} \otimes \widetilde{f} \otimes \widetilde{d}_l,$$

$\widetilde{v}_{k_{(l)}}$ being the value (constant) assigned to the terminal asset $\widetilde{A}_k$ in the component l.

Safeguards are measures for addressing threats. They can be procedures, such as incident management and documentation; personnel policies, such as training and awareness of employees operating on the IS; technical solutions, such as identification and authentication mechanisms based on biometrics; or physical security measures of the facilities, such as temperature control systems.

These safeguards can be *preventive*, if they reduce the frequency of threats; or *palliative*, if they reduce the degradation caused by threats on assets (López Crespo 2006). As the degree of dependence between two assets is the transmission probability of failures, a special type of preventive safeguard is that which reduces dependencies between support and terminal assets.

In this section we propose a method for reducing the degrees of dependency from all support assets to terminal assets minimizing the costs for the company.

As mentioned above, the probability of transmission of failure $\widetilde{D(A_i,A_k)}$ is the result of fuzzy operations with the probabilities of transmission of failure through intermediate assets linking the attacked support asset with other asset.

In each of these intermediate assets, safeguards can be enforced to reduce the probability of transmission of a failure. The effect induced for a safeguard in the probability of transmission of failures between two assets A_u and A_v can also be defined as a linguistic term, which is represented by a fuzzy number $\tilde{e}^{u,v} \in TF[0,1]$, so that if the degree of direct dependency between the assets A_u and A_v is $\widetilde{d(A_u,A_v)}$, then, when we implement a safeguard with effect $\tilde{e}^{u,v}$, the degree of direct dependency is reduced to

$$\widetilde{d(A_u,A_v)} \otimes (\tilde{1} \ominus \tilde{e}^{u,v}),$$

where $\ominus$ denotes the usual subtraction operation between trapezoidal fuzzy numbers, i.e., $(a_1,a_2,a_3,a_4) \ominus (b_1,b_2,b_3,b_4) = (a_1-b_4, a_2-b_3, a_3-b_2, a_4-b_1)$.

Note that $\ominus$ is not an internal composition law in TF[0,1], however,

- $\tilde{A},\tilde{B} \in TF[0,1] \Rightarrow \tilde{A} \otimes (\tilde{1} \ominus \tilde{B}) \in TF[0,1]$,
- $\tilde{A} \otimes (\tilde{1} \ominus \tilde{B}) \leq \tilde{A}$ with the partial order of the trapezoidal fuzzy numbers (i.e., $\tilde{A} \leq \tilde{B} \Leftrightarrow a_1 \leq b_1, a_2 \leq b_2, a_3 \leq b_3, a_4 \leq b_4$) and
- $\tilde{A} \otimes (\tilde{1} \ominus \tilde{B})$ decreases with $\tilde{B}$.

We consider the set of safeguards that hinder the direct transmission of failure between A_u and A_v, $S^{u,v}$. Each safeguard $S_p^{u,v} \in S^{u,v}$ has a monetary cost $c_p^{u,v}$ and an effect $\tilde{e}_p^{u,v}$ over $\widetilde{d(A_u,A_v)}$, which is reduced to $\widetilde{d(A_u,A_v)} \otimes (\tilde{1} \ominus \tilde{e}_p^{u,v})$.

The problem of keeping an acceptable level (low or very low) for the failure transmission probabilities among support and terminal assets with minimal costs can be represented as follows:

$$\begin{array}{c} min \sum_{u,v}\sum_{p} c_p^{u,v} x_p^{u,v} \\ s.t. \\ \widetilde{D(A_i,A_k)} \leq \tilde{U}_{ik}\, \forall i,k \\ x_p^{u,v} \in \{0,1\}\, \forall u,v,p \end{array},$$

where i and k in the first set of constraints refer to non-terminal and terminal assets, respectively, $\tilde{U}_{ik}$ is a residual value accepted by the experts, $x_p^{u,v}$ are the decision variables ($x_p^{u,v} = 1$ means that safeguard $S_p^{u,v}$ is selected), and $\widetilde{D(A_i,A_k)}$ is reassessed replacing values $\widetilde{d(A_u,A_v)}$ by the affected values regarding the selected safeguards:

$$\widetilde{d(A_u,A_v)} \otimes \left[\bigotimes_p (\tilde{1} \ominus \tilde{e}_p^{u,v}) \right],$$

where A_u and A_v are two consecutive assets connected by an arc in some path between A_i and A_k.

Note that the fact that the usual order in $TF[0,1]$ is a partial order constitutes a very restrictive constraint in our optimization problem, so we will use the concept of similarity function to relax this constraint.

If we define a threshold $\alpha \in [0,1]$ and a similarity function S, the constraint $\widetilde{D(A_i,A_k)} \leq \tilde{U}_{ik}\, \forall i,k$ can be replaced by $S(\widetilde{D(A_i,A_k)}, \tilde{U}_{ik}) \geq \alpha$. Thus, the restrictiveness of the constraint increases proportionally to the threshold value and the feasible solution set will be composed of solutions that verify these softened/relaxed constraints.

Remember that indirect dependencies are recursively computed following the algorithm described in Section 1. Thus, the degree of dependency of the support assets further away from the terminals can be computed from the degree of dependency of the closest assets. Therefore, the problem can be solved in stages, and the principle of optimality in dynamic programming is verified: Given an optimal sequence of decisions, every subsequence is, in turn, optimal. Then we proceed as follows:

- Let L_0 be the set of terminal assets.
- Consider L_1 including support assets whose children belong to L_0 only (L_1 is not empty because the graph is acyclic). Identify safeguards that minimize costs keeping the degrees of dependency over their children at an acceptable level.
- Consider L_2 including support assets whose children belong to $L_0 \cup L_1$ only. Identify safeguards that minimize costs keeping the degrees of dependency over L_0 under an acceptable level. Note that the degrees of indirect dependency from the children of L_2 to terminal assets have already been computed in the previous stage, so we just need to identify the direct degree of dependency over assets in $L_0 \cup L_1$.
- ...
- Consider L_i including support assets whose children belong to $L_0 \cup L_1 \cup ... \cup L_{i-1}$ only. Identify safeguards that minimize costs keeping the degrees of dependency over L_0 under an acceptable level. Note that again we just need to identify the direct degree of dependency on assets of $L_0 \cup L_1 \cup ... \cup L_{i-1}$.
- ...

Simulated annealing (Kirkpartick et al 1983, Cerny 1985) is applied in each step of the algorithm to derive the optimal selection of safeguards. It is a trajectorial metaheuristic which is named for and inspired by annealing in metallurgy.

An initial feasible solution is randomly generated. In each iteration a new solution y is randomly generated from the neighborhood of the current solution, $y \in N(x_i)$. If the new solution is better than the current one, then the algorithm moves to that solution ($x_{i+1} = y$), otherwise the movement to the worst solution is performed with certain probability.

Note that accepting worse solutions allows for a more extensive search for the optimal solution and avoids trapping in local optima in early iterations.

The probability of accepting a worse movement is a function of both the temperature factor and the change in the cost function.

The initial value of temperature (T) is high, which leads to a diversified search, since practically all movements are allowed. As the temperature decreases, the probability of accepting a worse movement falls. If the temperature is zero, then only better movements will be accepted, which makes simulated annealing work like hill climbing.

The pseudocode of simulated annealing for a minimization problem is as follows:

```
• Generate an initial feasible solution $x_0$. Do $x^* = x_0$, $f^* = f(x_0)$, $i = 0$.
  Select the initial temperature $t_0 = T$ ($t_i$ temperature in the step $i$)
• Repeat until stopping criterion is satisfied:
  – Randomly generate $y \in N(x_i)$
    ∗ If $f(y) - f(x_i) \leq 0$, then
      · $x_{i+1} = y$
      · If $f(x^*) > f(y)$, then $x^* = y, f^* = f(y)$
    ∗ Else
      · $p \sim U(0,1)$
      · If $p \leq e^{-(f(y)-f(x_i))/t_i}$, then $x_{i+1} = y$
      · Else $x_{i+1} = x_i$
  – $i = i + 1$
  – Update temperature
```

3 AN ILLUSTRATIVE EXAMPLE

Let us consider the IS shown in Figure 2 with the direct degrees of dependency assessed by the experts considering the linguistic terms of Table 1, which has only one terminal asset, A_6.

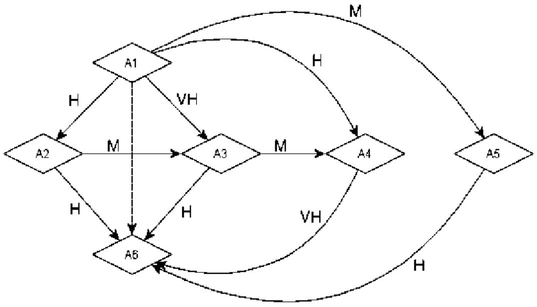

Figure 2: Direct dependencies in the IS.

The set of paths in the analysis of the influence of A_1 over A_6 is $\mathbf{P} = \{$

- $P_1 : (A_1 \rightarrow A_2 \rightarrow A_6)$,
- $P_2 : (A_1 \rightarrow A_2 \rightarrow A_3 \rightarrow A_6)$,
- $P_3 : (A_1 \rightarrow A_2 \rightarrow A_3 \rightarrow A_4 \rightarrow A_6)$,
- $P_4 : (A_1 \rightarrow A_3 \rightarrow A_6)$,
- $P_5 : (A_1 \rightarrow A_3 \rightarrow A_4 \rightarrow A_6)$,
- $P_6 : (A_1 \rightarrow A_4 \rightarrow A_6)$,
- $P_7 : (A_1 \rightarrow A_5 \rightarrow A_6)\}$.

Asset A_3 is influenced by A_1 and A_2, and A_4 is influenced by A_1 and A_3. Therefore, we proceed as in B) of the algorithm described in Section 2, with $r = 2$ and $\mathbf{S}= \{P_2, P_3, P_4, P_5, P_6\}$, as follows:

(i) $I = \{A_3, A_4\}$ and $NI = \{A_3\}$.

(ii) Select A_3, then simplify paths P_2, P_3, P_4 and P_5 to

- $P_2' : (A_1 \rightarrow A_3 \rightarrow A_6)$,
- $P_3' : (A_1 \rightarrow A_3 \rightarrow A_4 \rightarrow A_6)$,
- $P_4' : (A_1 \rightarrow A_3 \rightarrow A_6)$ and
- $P_5' : (A_1 \rightarrow A_3 \rightarrow A_4 \rightarrow A_6)$,

respectively, with $\widetilde{d(A_1,A_3)} = \widetilde{D(A_1,A_3)} = \left(\widetilde{d(A_1,A_2)} \otimes \widetilde{d(A_2,A_3)}\right) \oplus \widetilde{d(A_1,A_3)}$.

(iii) $\mathbf{S}= \{P_2', P_3', P_6\}$ since $P_2' = P_4'$ and $P_3' = P_5'$.

(iv) $I = \{A_4\}$ and $NI = \{A_4\}$.

(v) Go to (ii).

(ii) Select A_4, then simplify paths P_3' and P_6 to

- $P_3'' : (A_1 \rightarrow A_4 \rightarrow A_6)$, and
- $P_6' : (A_1 \rightarrow A_4 \rightarrow A_6)$,

respectively, with $\widetilde{d(A_1,A_4)} = \widetilde{D(A_1,A_4)} = \left(\widetilde{d(A_1,A_3)} \otimes \widetilde{d(A_3,A_4)}\right) \oplus \widetilde{d(A_1,A_4)}$.

(iii) $\mathbf{S}= \{P_2', P_3''\}$ since $P_3'' \equiv P_6'$.

(iv) $I = \varnothing$ y $NI = \varnothing$.

(v) The algorithm finishes since $NI = \varnothing$.

Finally, $\mathbf{S}=\{P_2', P_3''\}$ and the degree of dependency of A_6 regarding A_1 is $D\widetilde{(A_1,A_6)} = D\widetilde{(A_1,A_6|P_1)} \oplus D\widetilde{(A_1,A_6|P_7)} \oplus D\widetilde{(A_1,A_6|P_2')} \oplus D\widetilde{(A_1,A_6|P_3'')} = (d\widetilde{(A_1,A_2)} \otimes d\widetilde{(A_2,A_6)}) \oplus (d\widetilde{(A_1,A_5)} \otimes d\widetilde{(A_5,A_6)}) \oplus (d\widetilde{(A_1,A_3)} \otimes d\widetilde{(A_3,A_6)}) \oplus (d\widetilde{(A_1,A_4)} \otimes d\widetilde{(A_4,A_6)})$

The degree of dependency of A_6 regarding A_1 is $D(A_1,A_6) = (0.980, 0.999, 0.999, 1)$ if we consider the linguistic terms of Table 1 show in Figure 3.

Let us consider a threat on asset A_1 with frequency $\widetilde{f} = M$ and degradation $\widetilde{d} = (H,H,H)$, then the risk to asset A_1 is $\widetilde{R}_{1_{(l)}} = (0.23, 0.415, 0.485, 0.675)$, $l = 1,2,3$.

We consider the asset network and the fuzzy direct dependencies shown in Figure 2 corresponding to an IS. Besides, the set of available safeguards of failure transmission between support assets are shown in Tables 2-5.

We also consider the fuzzy threshold $\widetilde{U} = (0, 0, 0.1, 0.2)$ below which the degree of dependency between all assets and terminal assets will be acceptable, and let $\alpha = 0.95$. In other words, the similarity of the degree of dependency after applying the selected safeguards for the given $\widetilde{U}$ must be at least 0.95.

The set of solutions in each stage is represented by binary matrices, in which each row represents the safeguards of S_p^{uv}, which prevents the failure transmission from asset u to v considered in that stage.

We use the similarity function proposed by (Chen 1996): Given two trapezoidal fuzzy numbers $\widetilde{A} = (a_1, a_2, a_3, a_4)$ and $\widetilde{B} = (b_1, b_2, b_3, b_4)$,

$$S(\widetilde{A}, \widetilde{B}) = 1 - \frac{\sum\limits_{i=1}^{4} \mid a_i - b_i \mid}{4}.$$

Although other similarity functions have been proposed in the literature (Chen and Chen 2003, 2009, Sridevi and Nadarajan 2009, Xu et al 2010, Hejazi et al 2011, Gomathi and Sivaraman 2012, Zhu and Xu 2012, Vicente et al 2013b), we have decided to use the geometric distance between both fuzzy numbers due to its low computational cost.

Dynamic programming is then executed as follows:

First, note that $L_0 = \{A_6\}$, since the only terminal asset in the IS in Figure 2 is A_6.

- Stage 1: $L_1 = \{A_4, A_5\}$. We adjust the degrees of dependency $D\widetilde{(A_4,A_6)} = d\widetilde{(A_4,A_6)} \otimes \left[\bigotimes_{p}^{10}(\widetilde{1} \ominus \widetilde{e}_p^{4,6} x_p^{4,6})\right] = VH \otimes \left[\bigotimes_{p}^{10}(\widetilde{1} \ominus \widetilde{e}_p^{4,6} x_p^{4,6})\right]$ and $D\widetilde{(A_5,A_6)} = d\widetilde{(A_5,A_6)} \otimes \left[\bigotimes_{p}^{15}(\widetilde{1} \ominus \widetilde{e}_p^{5,6} x_p^{5,6})\right] = H \otimes \left[\bigotimes_{p}^{15}(\widetilde{1} \ominus \widetilde{e}_p^{5,6} x_p^{5,6})\right]$, such that $S\left(D\widetilde{(A_4,A_6)}, \widetilde{U}\right) \geq 0.95$ and $S\left(D\widetilde{(A_5,A_6)}, \widetilde{U}\right) \geq 0.95$, $\widetilde{e}_p^{4,6}$ being the effect induced for the safeguard $S_p^{4,6}, p = 1, ..., 10$, $\widetilde{e}_p^{5,6}$ the effect induced for the safeguard $S_p^{5,6}, p = 1, ..., 15$, $x_p^{4,6} = 1$ or $x_p^{4,6} = 0$ if the safeguard $S_p^{4,6}, p = 1, ..., 10$, is selected or not,

Table 2: Safeguards for A_1.

Tag	Effect	Cost	Tag	Effect	Cost
$S_1^{1,2}$	L	100	$S_1^{1,3}$	MH	356
$S_2^{1,2}$	M	300	$S_2^{1,3}$	H	324
$S_3^{1,2}$	MH	550	$S_3^{1,3}$	L	110
$S_4^{1,2}$	M	430	$S_4^{1,3}$	ML	345
$S_5^{1,2}$	ML	125	$S_5^{1,3}$	VL	87
$S_6^{1,2}$	L	240	$S_6^{1,3}$	MH	345
$S_7^{1,2}$	VL	100	$S_7^{1,3}$	M	200
$S_8^{1,2}$	MH	324			
$S_9^{1,2}$	VH	570			
$S_1^{1,4}$	M	209	$S_1^{1,5}$	M	230
$S_2^{1,4}$	M	267	$S_2^{1,5}$	M	345
$S_3^{1,4}$	MH	342	$S_3^{1,5}$	L	187
$S_4^{1,4}$	VH	789	$S_4^{1,5}$	M	321
$S_5^{1,4}$	M	234	$S_5^{1,5}$	MH	345
$S_6^{1,4}$	M	356	$S_6^{1,5}$	H	543
$S_7^{1,4}$	M	276	$S_7^{1,5}$	MH	356
$S_8^{1,4}$	M	200	$S_8^{1,5}$	M	206
$S_9^{1,4}$	H	467	$S_9^{1,5}$	M	342
$S_{10}^{1,4}$	H	342			
$S_{11}^{1,4}$	L	127			
$S_{12}^{1,4}$	M	207			

Table 3: Safeguards for A_2.

Tag	Effect	Cost	Tag	Effect	Cost
$S_1^{2,3}$	M	356	$S_1^{2,6}$	M	348
$S_2^{2,3}$	L	87	$S_2^{2,6}$	L	187
$S_3^{2,3}$	ML	267	$S_3^{2,6}$	ML	254
$S_4^{2,3}$	M	320	$S_4^{2,6}$	ML	367
$SS_5^{2,3}$	ML	156	$S_5^{2,6}$	ML	567
$S_6^{2,3}$	M	320	$S_6^{2,6}$	M	390
$S_7^{2,3}$	M	256	$S_7^{2,6}$	ML	256
$S_8^{2,3}$	M	300	$S_8^{2,6}$	M	307
$S_9^{2,3}$	L	200	$S_9^{2,6}$	L	235
			$S_{10}^{2,6}$	ML	124
			$S_{11}^{2,6}$	M	400
			$S_{12}^{2,6}$	L	278
			$S_{13}^{2,6}$	ML	260

Table 4: Safeguards for A_3.

Tag	Effect	Cost	Tag	Effect	Cost
$S_1^{3,4}$	M	345	$S_1^{3,6}$	M	267
$S_2^{3,4}$	H	650	$S_2^{3,6}$	M	356
$S_3^{3,4}$	M	200	$S_3^{3,6}$	M	378
$S_4^{3,4}$	M	367	$S_4^{3,6}$	M	324
$S_5^{3,4}$	M	388	$S_5^{3,6}$	M	345
$S_6^{3,4}$	H	453	$S_6^{3,6}$	M	231
$S_7^{3,4}$	L	189	$S_7^{3,6}$	MH	453
$S_8^{3,4}$	L	256			
$S_9^{3,4}$	M	345			

Table 5: Safeguards for A_4 and A_5.

Tag	Effect	Cost	Tag	effect	Cost
$S_1^{4,6}$	M	260	$S_1^{5,6}$	M	200
$S_2^{4,6}$	M	245	$S_2^{5,6}$	M	210
$S_3^{4,6}$	ML	170	$S_3^{5,6}$	L	120
$S_4^{4,6}$	M	256	$S_4^{5,6}$	ML	234
$S_5^{4,6}$	M	367	$S_5^{5,6}$	M	267
$S_6^{4,6}$	M	289	$S_6^{5,6}$	MH	367
$S_7^{4,6}$	M	278	$S_7^{5,6}$	MH	366
$S_8^{4,6}$	M	345	$S_8^{5,6}$	M	254
$S_9^{4,6}$	M	240	$S_9^{5,6}$	ML	145
$S_{10}^{4,6}$	MH	435	$S_{10}^{5,6}$	L	206
			$S_{11}^{5,6}$	M	306
			$S_{12}^{5,6}$	M	345
			$S_{13}^{5,6}$	M	280
			$S_{14}^{5,6}$	L	178
			$S_{15}^{5,6}$	MH	377

respectively, and $x_p^{5,6} = 1$ or $x_p^{5,6} = 0$ depending on whether or not the safeguard $S_p^{5,6}, p = 1,...,15$, minimizing the cost.

As L_1 contains two elements, two optimization problems must be solved in this stage, associated with A_4 and A_5, respectively.

Regarding asset A_4, solutions are represented by the vector $x^{4,6} = (x_1^{4,6}, x_2^{4,6}, ..., x_{10}^{4,6})$, see Table 5, where $x_p^{4,6} = 1$ if the safeguard $S_p^{4,6}$ is selected. The respective optimization problem to be solved using simulated annealing is:

$$\begin{array}{c} min \;\; c_1^{4,6} x_1^{4,6} + \ldots + c_{10}^{4,6} x_{10}^{4,6} \\ s.t. \\ S\left(\widetilde{D(A_4,A_6)}, \widetilde{U}\right) \geq 0.95 \\ x_p^{4,6} \in \{0,1\}, p = 1,...,10 \end{array} \quad . \tag{6}$$

The optimal solution and the associated costs are shown in the second row of Table 6, corresponding to vector $x^{4,6^*} = (0,1,1,1,0,0,0,0,1,0)$.

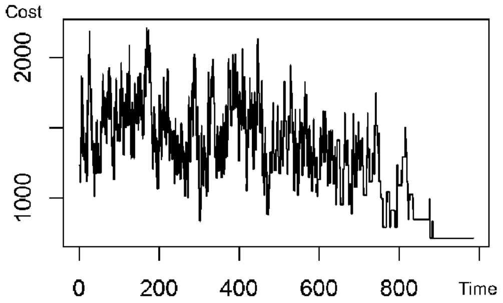

Figure 3: Objective function evolution in the optimum setting of $\widetilde{D(A_5,A_6)}$.

Regarding asset A_5, solutions are now represented by the vector $x^{5,6} = (x_1^{5,6}, x_2^{5,6}, ..., x_{15}^{5,6})$, see Table 5. The optimization problem to be solved is:

$$\begin{array}{c} min \;\; c_1^{5,6} x_1^{5,6} + \ldots + c_{15}^{5,6} x_{15}^{5,6} \\ s.t. \\ S\left(\widetilde{D(A_5,A_6)}, \widetilde{U}\right) \geq 0.95 \\ x_p^{5,6} \in \{0,1\}, p = 1,...,15 \end{array} \quad . \tag{7}$$

The evolution of the objective function over time for the best solution found is shown in Figure 3. The optimal solution and the associated costs are shown in the first row of Table 6, corresponding to vector $x^{5,6^*} = (1,0,0,0,0,0,1,0,1,0,0,0,0,0,0)$.

The new degrees of dependency after the application of the selected safeguards and the respective similarity to the fixed threshold, $\widetilde{U}$, are shown in the first two rows of Table 7.

The purpose of this paper is to describe how a mixture of dynamic programming techniques and metaheuristics can efficiently solve the problem and not to detail or compare the applied metaheuristic (simulated annealing) with others. However, we do think it is worthwhile to describe some parameters used in the implementation and to report a sensitivity analysis analyzing the effects caused by the changes to these parameters.

- We randomly generate a sequence with binary values and check if the similarity constraint is verified to derive the initial solution. The length of the binary sequence depends on the problem (15 when dealing with $x^{5,6}$, 10 when dealing with $x^{4,6}$...).
- The neighborhood of a solution is composed of any solutions that can be derived by changing the value of one of the binary elements of the solution, selected at random. If the resulting

solution is not feasible (does not verify the similarity constraint), then it is discarded and another solution is generated in the neighborhood until a feasible solution is found.

- The initial temperature assures acceptance probabilities of worse solutions close to 0.9 in the initial iterations of the algorithm. The initial temperature is computed to obtain a high probability of acceptance (≥ 0.9) of any neighbor of the initial solution, i.e., given the initial solution x_0, the minimum value T is computed such that

$$e^{-(f(y)-f(x_0))/T} \geq 0.9,\ \forall y \in N(x_0) \text{ and feasible,}$$

with:

$$f(y) - f(x_0) > 0.$$

In other words,

$$T = \max_{\substack{y \in N(x_0)\\ feasible}} \left\{ \frac{-(f(y)-f(x_0))}{ln(0.9)} \right\}$$

because if we have $T \geq \frac{-(f(y)-f(x_0))}{ln(0.9)}$ $\forall y \in N(x_0)$ and feasible, with$f(y) - f(x_0) > 0$, then $ln(0.9) \leq \frac{-(f(y)-f(x_0))}{T}$ $\forall y \in N(x_0)$ and feasible, with $f(y) - f(x_0) > 0$, and since e^x is an increasing function, $0.9 \leq e^{\frac{-(f(y)-f(x_0))}{T}}$ $\forall y \in N(x_0)$ and feasible, with $f(y) - f(x_0) > 0$.

The pseudocode, starting from $x_0 = (x_0[1], ..., x_0[n])$, as follows:

```
* $y = x_0, T = 0, i = 1.$
* While $i \leq n$. Do $y[i] = 1 - y[i]$.
    · If $y$ is a feasible solution then, if $\frac{-(f(y)-f(x_0))}{ln(0.9)} > T$, we have
          $$T = \frac{-(f(y)-f(x_0))}{ln(0.9)}.$$
    · $y = x_0, i = i + 1.$
```

The solution x_0 has at most n feasible neighboring solutions. We have evaluated all neighboring solutions that are worse than the initial solution in those n steps.

In the unfortunate event that the initial solution is the worst of its neighborhood, the initial value of the resulting T is null. Therefore we must start from another initial solution. This does not degrade the algorithm, because it can return to the neighborhood of the discarded solution at any time.

Thus the initial temperature that leads to the optimal solution over A_5 (for optimization problem (7)) is 3578.191.

Table 6: Optimal solutions and costs for each asset.

Asset	Solution	Cost
A_5	$S_1^{5,6}, S_7^{5,6}, S_9^{5,6}$	711
A_4	$S_2^{4,6}, S_3^{4,6}, S_4^{4,6}, S_9^{4,6}$	911
A_3	$S_1^{3,6}, S_4^{3,6}, S_6^{3,6}, S_7^{3,6}$	1275
A_2	$S_7^{2,3}, S_1^{2,6}, S_5^{2,6}, S_7^{2,6}, S_{10}^{3,6}$	1551
A_1	$S_1^{1,2}, S_2^{1,3}, S_{10}^{1,4}$	1236
	Total cost	5684

The temperature is maintained constant for $L = 20$ iterations and then it decreases after multiplying by 0.95, so that, after $h * L$ iterations, the temperature is $t_{h*L} = 0.95^h t_0$.

- The algorithm stops if f has not improved in the last 100 iterations.

Table 7 shows the best solutions reached after running the algorithm with different values for α to minimize $D\widetilde{(A_5,A_6)}$. Note that if the constraint is more restrictive, allowing only minor differences with the threshold $\widetilde{U}$, the set of safeguards for implementation will be larger. The same effect occurs when we use a more accurate (with a smaller support) threshold $\widetilde{U}$. Therefore, experts must choose lower or higher levels of acceptable accuracy regarding the dependency between assets, i.e., the accepted risk considering this fact.

- Stage 2: $L_2 = \{A_3\}$. The degrees of dependency $d\widetilde{(A_3,A_6)}$ and $d\widetilde{(A_3,A_4)}$ are adjusted by minimizing costs and incorporating the soft constraint $S\left(D\widetilde{(A_3,A_6)}, \widetilde{U}\right) \geq 0.95$, where

$$D\widetilde{(A_3,A_6)} = \left[d\widetilde{(A_3,A_6)} \otimes \left(\bigotimes_{p=1}^{7} (\widetilde{1} \ominus \widetilde{e}_p^{3,6} x_p^{3,6}) \right) \right] \oplus \left[d\widetilde{(A_3,A_4)} \otimes \left(\bigotimes_{p=1}^{9} (\widetilde{1} \ominus \widetilde{e}_p^{3,4} x_p^{3,4}) \right) \otimes D\widetilde{(A_4,A_6)} \right].$$

Note that $D\widetilde{(A_4,A_6)}$ was computed in Stage 1, $D\widetilde{(A_4,A_6)} = VH \otimes \left[(\widetilde{1} \ominus \widetilde{e}_2^{4,6}) \otimes (\widetilde{1} \ominus \widetilde{e}_3^{4,6}) \otimes (\widetilde{1} \ominus \widetilde{e}_4^{4,6}) \otimes (\widetilde{1} \ominus \widetilde{e}_9^{4,6}) \right] = (0.016, 0.072, 0.104, 0.269)$. The optimization problem to be solved in this stage is

$$\begin{array}{rl} min & c_1^{3,6} x_1^{3,6} + ... + c_7^{3,6} x_7^{3,6} + \\ & c_1^{3,4} x_1^{3,4} + ... + c_9^{3,4} x_9^{3,4} \\ & s.t. \\ & S\left(D\widetilde{(A_3,A_6)}, \widetilde{U}\right) \geq 0.95 \\ & x_p^{3,6} \in \{0,1\}, p = 1, ..., 7 \\ & x_q^{3,4} \in \{0,1\}, q = 1, ..., 9 \end{array}$$

The optimal solution and the associated cost is

Table 7: $\widetilde{D(A_5,A_6)}$ and associated costs for different α levels.

α	$\widetilde{D(A_5,A_6)}$	Similarity	Cost
0.8	(0.05,0.23,0.27,0.46)	0.81	554
0.9	(0.02,0.09,0.12,0.30)	0.93	653
0.95	(0.01,0.07,0.11,0.28)	0.95	711
0.98	(0.00,0.03,0.06, 0.20)	0.98	1021

shown in the third row of Table 6, corresponding to vectors $x^{3,6^*} = (1,0,0,1,0,1,1)$ and $x^{3,4^*} = (0,0,0,0,0,0,0,0,0)$. The new degree of dependency after the application of the selected safeguards and the corresponding similarity to the fixed threshold, $\widetilde{U}$, are shown in the third row of Table 7.

- Stage 3: $L_3 = \{A_2\}$. The degrees of dependency $\widetilde{d(A_2,A_3)}$ and $\widetilde{d(A_2,A_6)}$ are adjusted minimizing costs and incorporating the soft constraint $S\left(\widetilde{D(A_2,A_6)},\widetilde{U}\right) \geq 0.95$, where
$$\widetilde{D(A_2,A_6)} = \left[\widetilde{d(A_2,A_6)} \otimes \left(\bigotimes_{p=1}^{13} (\widetilde{1} \ominus \widetilde{e}_p^{2,6} x_p^{2,6})\right)\right] \oplus \left[\widetilde{d(A_2,A_3)} \otimes \left(\bigotimes_{p=1}^{7} (\widetilde{1} \ominus \widetilde{e}_p^{2,3} x_p^{2,3})\right) \otimes \widetilde{D(A_3,A_6)}\right].$$
Note that $\widetilde{D(A_3,A_6)}$ was computed in Stage 2, $\widetilde{D(A_3,A_6)} = [\widetilde{d(A_3,A_6)} \otimes (\widetilde{1} \ominus \widetilde{e}_1^{3,6}) \otimes (\widetilde{1} \ominus \widetilde{e}_4^{3,6}) \otimes (\widetilde{1} \ominus \widetilde{e}_6^{3,6}) \otimes (\widetilde{1} \ominus \widetilde{e}_7^{3,6})] \oplus [\widetilde{d(A_3,A_4)}) \otimes \widetilde{D(A_4,A_6)}] = (0.008,0.059,0.096,0.301)$.
The optimal solution and the associated cost are shown in the fourth row of Table 6, corresponding to vectors $x^{2,3^*} = (0,0,0,0,0,0,1,0,0)$ and $x^{2,6^*} = (1,0,0,0,1,0,1,0,0,1,0,0,0)$. The new degree of dependency and similarity to $\widetilde{U}$, are shown in the fourth row of Table 8.
- Finally, $L_4 = \{A_1\}$. The degrees of dependency $\widetilde{d(A_1,A_2)}$, $\widetilde{d(A_1,A_3)}$, $\widetilde{d(A_1,A_4)}$ and $\widetilde{d(A_1,A_5)}$ are adjusted minimizing the cost and considering the soft constraint $S\left(\widetilde{D(A_1,A_6)},\widetilde{U}\right) \geq 0.95$, where
$$\widetilde{D(A_1,A_6)} = \left[\widetilde{d(A_1,A_2)} \otimes \left(\bigotimes_{p=1}^{9} (\widetilde{1} \ominus \widetilde{e}_p^{1,2} x_p^{1,2})\right) \otimes \widetilde{D(A_2,A_6)}\right] \oplus \left[\widetilde{d(A_1,A_3)} \otimes \left(\bigotimes_{p=1}^{7} (\widetilde{1} \ominus \widetilde{e}_p^{1,3} x_p^{1,3})\right) \otimes \widetilde{D(A_3,A_6)}\right] \oplus \left[\widetilde{d(A_1,A_4)} \otimes \left(\bigotimes_{p=1}^{12} (\widetilde{1} \ominus \widetilde{e}_p^{1,4} x_p^{1,4})\right) \otimes \widetilde{D(A_4,A_6)}\right] \oplus \left[\widetilde{d(A_1,A_5)} \otimes \left(\bigotimes_{p=1}^{9} (\widetilde{1} \ominus \widetilde{e}_p^{1,5} x_p^{1,5})\right) \otimes \widetilde{D(A_5,A_6)}\right].$$
Note that $\widetilde{D(A_2,A_6)}$, $\widetilde{D(A_3,A_6)}$, $\widetilde{D(A_4,A_6)}$ and $\widetilde{D(A_5,A_6)}$ were computed in previous stages,
$\widetilde{D(A_2,A_6)} = [\widetilde{d(A_2,A_6)} \otimes ((\widetilde{1} \ominus \widetilde{e}_1^{2,6}) \otimes (\widetilde{1} \ominus \widetilde{e}_5^{2,6}) \otimes (\widetilde{1} \ominus \widetilde{e}_7^{2,6}) \otimes (\widetilde{1} \ominus \widetilde{e}_{10}^{2,6}))] \oplus [\widetilde{d(A_2,A_3)} \otimes \left((\widetilde{1} \ominus \widetilde{e}_7^{2,3})\right) \otimes \widetilde{D(A_3,A_6)}] = (0.008,0.057,0.094,0.316)$,
$\widetilde{D(A_3,A_6)} = (0.008,0.059,0.096,0.301)$,
$\widetilde{D(A_4,A_6)} = (0.016,0.072,0.104,0.269)$ and
$\widetilde{D(A_5,A_6)} = (0.01,0.07,0.11,0.28)$.
The optimal solution in this stage is shown in the last row of Tables 6 and 7.

Table 8: New degrees of dependency after applying safeguards.

Asset	$\widetilde{D(A_j,A_6)}$	Similarity $\widetilde{U}$
A_5	(0.015,0.077,0.114,0.280)	0.953
A_4	(0.016,0.072,0.104,0.269)	0.959
A_3	(0.008,0.059,0.096,0.301)	0.956
A_2	(0.008,0.057,0.094,0.316)	0.953
A_1	(0.005,0.045,0.082,0.327)	0.951

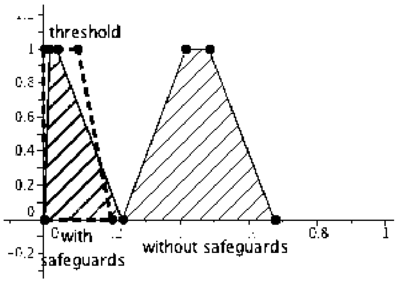

Figure 4: Risk in each component of A_1 before and after implementation of optimal safeguards

After implementing the best safeguards, the risk caused by the previously considered threat over asset A_1 in each component is $\widetilde{R}_{1_{(l)}} = (0.001,0.018,0.039,0.22)$, $l = 1,2,3$.

The risks associated with this threat before and after implementation of safeguards are illustrated along with the risk threshold in Figure 4.

4 CONCLUSIONS

We propose a model for selecting safeguards to reduce risks in information systems based on the reduction of the degree of dependency between support assets and terminal assets. As safeguards have associated costs, our aim is to select safeguards that minimize costs while keeping the risk with acceptable levels.

Although a metaheuristic could be used to solve this optimization problem, dynamic programming combined with simulated annealing was used because of the special structure of the constraint set. This leads to a more computationally efficient solution to the safeguard selection problem. Also the fuzzy environment allows experts to provide imprecise and vague failure propagation probabilities.

Another way to reduce system risk is to act on the probability of threats to each asset materializing or reducing the degradation of assets caused by threat materialization. This is a multiobjective problem (degradation has three components), which will be considered in future research.

ACKNOWLEDGEMENTS

The paper was supported by Madrid Regional Government project S-2009/ESP-1685 and Spanish Ministry of Science and Innovation project MYTM2011-28983-C03-03.

REFERENCES

Cerny, V. (1985). Thermodynamical Approach to the Traveling Salesman Problem: An Efficient Simulation Algorithm, *Journal of Optimization Theory and Applications*, 45, 41-51.

Chen, S.-M. (1996). New Methods for Subjective Mental Workload Assessment and Fuzzy Risk Analysis, *Cybernetics Systems*, 27, 449-472.

Chen, S.-J. and Chen, S.-M. (2003). Fuzzy Risk Analysis Based on Similarity Measures of Generalized Fuzzy Numbers. *IEEE Transactions on Fuzzy Systems*, 11, 45-56.

Chen, S.-J. and Chen, S.-M. (2009). Fuzzy Risk Analysis Based on the Ranking of Generalized Trapezoidal Fuzzy Numbers. *Applied Intelligence*, 26, 1-11.

CCTA Risk Analysis and Management Method (CRAMM), Version 5.0. London: Central Computing and Telecommunications Agency (CCTA), 2003.

Finetti, B. (1964). Foresight: its Logical Laws, its Subjective Sources. In: H.E. Kyburg and H.E. Smokler (eds.), Studies in Subjective Probability. New York: Wiley.

Gomathi, V.L. and Sivaraman, G. (2012). A Novel Similarity Measure between Generalized Fuzzy Numbers. *International Journal of Computer Theory and Engineering*, 4, 448-450.

ISO/IEC Serie 27000 International Organization for Standardization.

Hejazi, S. R., Doostparast, A. and Hosseini, S.M. (2011). An Improved Fuzzy Risk Analysis based on a New Similarity Measures of Generalized Fuzzy Numbers. *Expert Systems with Applications*, 38, 9179-9185.

Kirkpatrick, S., Gelatt., C.D. and Vecchi, M. P. (1983). Optimization by Simulated Annealing. *Science*, 220 (4598), 671-680.

López Crespo, F., Amutio-Gómez, M.A., Candau, J. and Mañas, J.A. (2006). *Methodology for Information Systems Risk. Analysis and Management (MAGERIT version 2). Book I, Book II and Book III.* Madrid: Ministerio de Administraciones Públicas.

Savage, L. J. (1954). The Foundations of Statistics. New York: Wiley.

Sridevi, B. and Nadarajan, R. (2009). Fuzzy Similarity Measure for Generalized Fuzzy Numbers. *International Journal of Open Problems in Computer Science and Mathematics*, 2, 111-116.

Stoneburner, G. and Gougen, A. (2002). *NIST 800-30 Risk Management. Guide for Information Technology Systems*. Gaithersburg: National Institute of Standard and Technology.

Vicente, E., Jiménez, A. and Mateos, A. (2013a). A Fuzzy Approach to Risk Analysis in Information Systems. *Proceedings of the 2nd International Conference on Operations Research and Enterprise Systems*, 130-133.

Vicente, E., Mateos, A. and Jiménez, A. (2013b). A New Similarity Function for Generalized Trapezoidal Fuzzy Numbers. *Lecture Notes on Computer Science*, 7894, 400-411.

Vicente, E., Jiménez, A. and A. Mateos, A. (2013c). An interactive method of fuzzy probability elicitation in risk analysis, *Intelligent Systems and Decision Making for Risk Analysis and Crisis Response*, New York: CRC Press, 223-228.

Xu, Z., Shang, S., Qian, W. and Shu, W. (2010). A Method for Fuzzy Risk Analysis based on the New Similarity of Trapezoidal Fuzzy Numbers. *Expert Systems with Applications*, 37, 1920-1927.

Zu, L. and R. Xu (2012). Fuzzy risk analysis based on similarity measure of generalized fuzzy numbers. *Fuzzy Engineering and Operations Research.* Berlin/Heidleberg: Springer, 569-587.