

Some Aspects about Vagueness & Imprecision in Computer Network Fault-Tree Analysis

D. E. Popescu, M. Lonea, D. Zmaranda, C. Vancea, C. Tiurbe

Daniela Elena Popescu, Doina Zmaranda, Codruta Vancea, Cristian Tiurbe

University of Oradea

Romania, 410087 Oradea, 1 Universitatii St.

E-mail: {depopescu,zdoina,cvancea,ctiurbe}@uoradea.ro

Madalina Lonea

"Politehnica" University of Timisoara

Romania, Timisoara, 2-4 V. Parvan Blvd.

E-mail: madalina_lonea@yahoo.com

Abstract: Based on the available information (eg. multiple functional faults or sensor errors give rise to similar alarm patterns or outcomes), some states in the behaviour of a network can not be distinguished from one another. So, the computer network's fault tree reliability analysis frequently relies on imprecise or vague input data. The paper will use a Dempster-Shafer Theory to accommodate this vagueness and it will show how imprecision can give rise to false-negative, and false-positive inferences; there will be assigned upper and lower bounds for the probability on elements of the state space. After illustrating the computational simplicity of incorporating the Dempster-Shafer Theory probability assignments, we will apply them for analyzing the reliability of the network of our department.

Keywords: reliability analysis, networks, Dempster-Shafer Theory, fault tree.

1 Introduction

The probabilities are no longer appropriate to represent vagueness in risk and reliability analyses; fuzzy set theory was proposed instead to quantify vagueness in this area [1]. Development in DST have shown how probability can be adapted to incomplete or vague information, especially information that is based on human judgment or human-machine interaction.

False positives and false-negatives are often the end products of vagueness or imprecision. They can arise from imprecision due to noise in monitored data, sensor device failure, or from the ambiguity about the logic rules in the fault tree.

So, when the researcher has only imprecise information, he must appeal to fuzzy-set theory techniques, either the common laws or logic his appreciation for the logic relations in fault trees. Fuzzy data can be incorporated through Dempster Shafer Theory (DST) [2] [3] [4] [5] in conventional fault-tree analysis yield meaningful results.

2 Dempster-Shafer Theory mass assignments

DST generalizes classical probability theory by assigning upper and lower bounds for probabilities, as opposed to point values, to both the elements and the subsets of the state space. For a given state space, Ω , mass (probability) is assigned over the set of all possible subsets of Ω . Because each element of Ω is also a subset of Ω (comprising 1 element) any classical probability assignment can be represented in DST. Just as the probabilities of a distribution sum to 1, so do the masses of a DST-distribution.

The standard risk analysis possibility set considered is $\Omega = \{True, False\}$. Therefore, the power set, $2^\Omega = \{\Phi, T, F, (T, F)\}$ contains an element (T,F) that represents an observation that could be either true or false, but not both.

The components of faults trees - the initiating events, consequences, and rules - are typically given fail or not fail possibilities. In some engineering applications, observation of particular initiating events is imprecise, so an analyst cannot tell whether a given event occurred. By using DST method of assigning probability to the (T,F) event, fault tree analysts can quantitatively depict the imprecision [7].

Based on the fact that [sum of all masses] = 1 and based on the fact that in DST $m(\Phi) = 0$, after DST renormalization of the Φ , the state space become: T, F, (T, F), and:

$$m(T) + m(F) + m(T,F) = 1$$

Ordinary fault trees are systems of Boolean equations with components joined by Boolean AND & OR gates. The Boolean AND gate for the members of the effective state space is summarized in Figure 1 [7], and the Boolean OR gate is given in Figure 2.

Figure 1: Boolean Truth Table for the AND gate

\wedge	T	F	(T,F)
T	T	F	(T,F)
F	F	F	F
(T,F)	(T,F)	F	(T,F)

Figure 2: Boolean Truth Table for the OR gate

\vee	T	F	(T,F)
T	T	T	T
F	T	F	(T,F)
(T,F)	T	(T,F)	(T,F)

The mass for each entry in the table is obtained by multiply-ing the masses of the edge entries and adding all masses of like entries. For example:

Let a_1, a_2, a_3 numbers denoting ma over the possibilities $T, F, (T, F)$ where $a_1 + a_2 + a_3 = 1$

Let b_1, b_2, b_3 numbers denoting mb over the possibilities $T, F, (T, F)$ where $b_1 + b_2 + b_3 = 1$

For the Boolean OR gate from table 2 we have:

$$m\{A \vee B\} = (a_1b_1 + a_1b_2 + a_1b_3 + a_2b_1 + a_3b_1; a_2b_2; a_2b_3 + a_3b_2 + a_3b_3) \tag{1}$$

$$= (a_1 + a_2b_1 + a_3b_1; a_2b_2; a_2b_3 + a_3b_2 + a_3b_3)$$

Thus, for $(A \vee B)$ the T element has mass assignment $(a_1 + a_2b_1 + a_3b_1)$, the F element has mass assignment a_2b_2 , the (T,F) element has mass assignment $(a_2b_3 + a_3b_2 + a_3b_3)$.

Similarly, for the Boolean gate from Figure 1, we have:

$$m\{A \wedge B\} = (a_1b_1; a_1b_2 + a_2b_1 + a_2b_2 + a_2b_3 + a_3b_2; a_1b_3 + a_3b_1 + a_3b_3) \tag{2}$$

$$= (a_1b_1; a_1b_2 + a_2 + a_3b_2; a_1b_3 + a_3b_1 + a_3b_3)$$

3 Fault Tree illustration

We will apply the above techniques for the fault tree that corresponds to the network monitoring system of our department. Our network uses 3 routers and it accesses the Internet through the university

router (Figure 3). It comprises the following elements:

R1	“HN” Router	R1’s fault represent the E1 event
R2	“Cazarma” Router	R2’s fault represent the E2 event
R3	“CNLAB” Router	R3’s fault represent the E3 event
R4	“Univ.Oradea” Router	R4’s fault represent the E4 event
S1	Files Server	S1’s fault represent the E5 event
S2	FTP Server	S2’s fault represent the E6 event
S3	Database Server	S3’s fault represent the E7 event
S4	Domain Server	S4’s fault represent the E8 event

The network’s fault tree representation is shown in Figure 4.

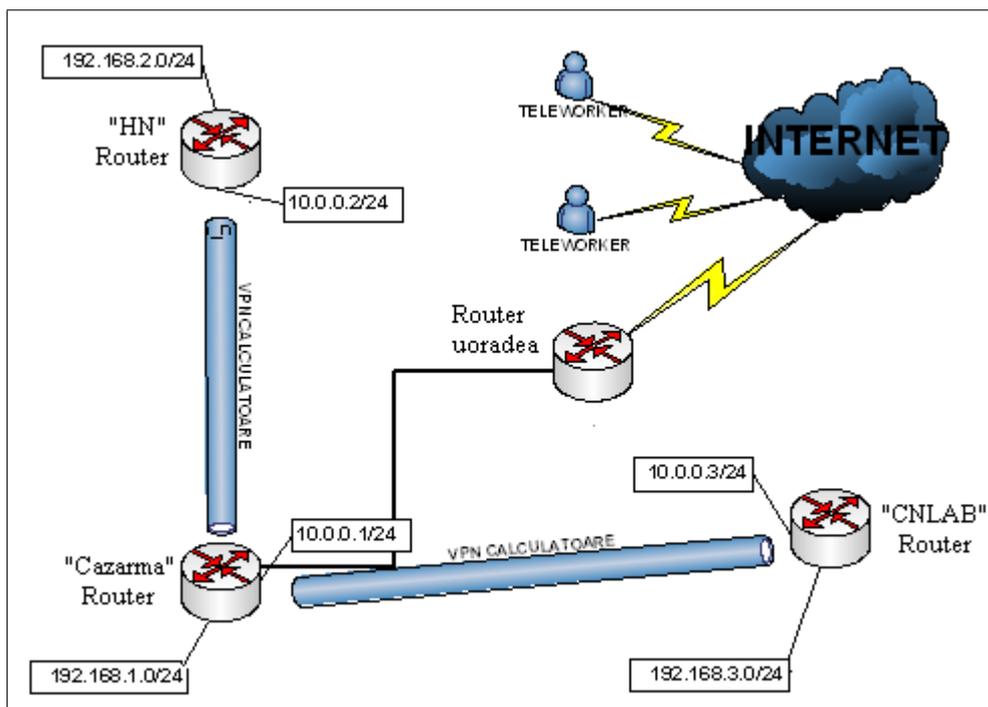


Figure 3: Computer Science Network

The top event is labelled T on the tree represented in Figure 4. The initiating events E_i ($i=1, 2, 3, 4, 5, 6, 7, 8$) lead to intermediate results (levels), I_j ($j=1, 2, 3, 4, 5, 6$). The fault tree comprises the following Boolean equations:

$$T = I_1 \vee I_6 \quad (3)$$

$$I_1 = E_3 \vee E_4 \quad (4)$$

$$I_6 = I_4 \vee I_5 \quad (5)$$

$$I_4 = I_2 \vee I_3 \quad (6)$$

$$I_2 = E_5 \vee E_6 \quad (7)$$

$$I_3 = E_7 \vee E_8 \quad (8)$$

$$I_5 = E_1 \vee E_2 \quad (9)$$

The equal signs in equations (3) ÷ (9) mean that the logical implications works in both directions. To assess the risk of the top event, masses can be assigned to the initiating events E_i , and then propagated

up the tree based on relations (1) and (2). The main problem is the determining of the probabilities of basic initiating events (how are they assigned or determined).

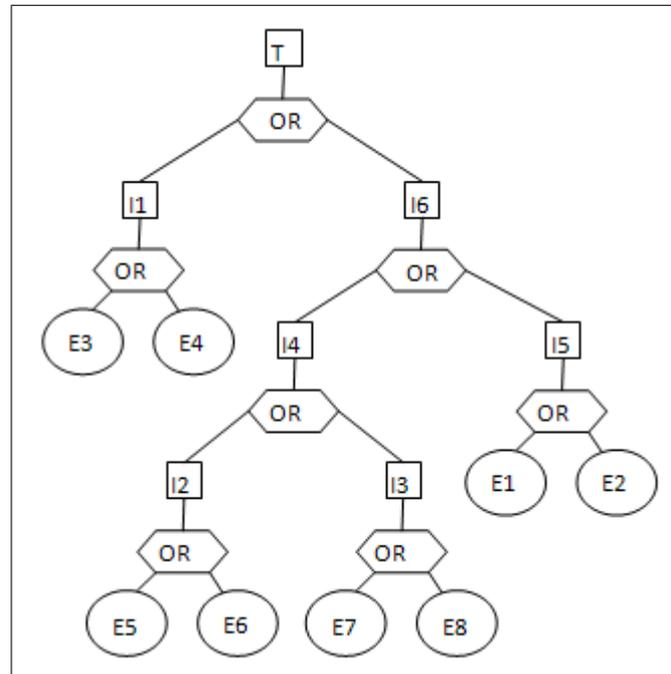


Figure 4: The Fault Tree

Each of (3) ÷ (9) can be interpreted as an "if ... then ..." rule of the fault tree and considered as much part of the fault tree as the initiating events. So, let R_i be a parameter about the failure rates of rule $i = 3, \dots, 9$, where the i -th corresponds to equations i [6]. When joined to our existing fault tree, the R_i depict a further constraint on the fault tree, through which the initiating events must pass before an intermediate level is reached. The R_i can be interpreted in several ways. For example, to model "silent alarms" when the true state of a system is abnormal, the R_i could represent incidents in which the alarms are not working properly. If a rule were incorporated in the design of the fault tree such that the rule were true in only 9 out of 10 trials, then coupling R_i with a Boolean AND gate to the existing tree could filter the number of sure fire observations from the rule to 9/10..

Joining the R_i to the fault tree through an AND gate is useful in modelling false negatives. When the R_i are joined to the fault tree by AND gates and some sensors fail during abnormal conditions, the anticipated consequence might not occur on the fault tree because it has been falsely stopped by the rule parameter.

We will focus only on false-negatives, and thus, the R_i are joined to the existing fault tree through some Boolean AND gates. The new fault tree graphically represents the seven equations:

$$T = (I_1 \vee I_6) \wedge R_3 \tag{10}$$

$$I_1 = (E_3 \vee E_4) \wedge R_4 \tag{11}$$

$$I_6 = (I_4 \vee I_5) \wedge R_5 \tag{12}$$

$$I_4 = (I_2 \vee I_3) \wedge R_6 \tag{13}$$

$$I_2 = (E_5 \vee E_6) \wedge R_7 \tag{14}$$

$$I_3 = (E_7 \vee E_8) \wedge R_8 \tag{15}$$

$$I_5 = (E_1 \vee E_2) \wedge R_9 \quad (16)$$

We consider 3 adjustable assignments of mass for the initiating events and rules variables on the fault tree; these are given in Table 1.

Table 1: Mass Assignment

cwev	Case I	Case II	Case III
E1	(0.9, 0.1, 0)	(0.8, 0, 0.2)	(0.8, 0, 0.2)
E2	(0.9, 0.1, 0)	(0.8, 0, 0.2)	(0.8, 0, 0.2)
E3	(0.9, 0.1, 0)	(0.8, 0, 0.2)	(0.8, 0, 0.2)
E4	(0.9, 0.1, 0)	(0.8, 0, 0.2)	(0.8, 0, 0.2)
E5	(0.8, 0.2, 0)	(0.9, 0, 0.1)	(0.9, 0, 0.1)
E6	(0.8, 0.2, 0)	(0.7, 0, 0.3)	(0.9, 0, 0.1)
E7	(0.8, 0.2, 0)	(0.9, 0, 0.1)	(0.9, 0, 0.1)
E8	(0.8, 0.2, 0)	(0.9, 0, 0.1)	(0.9, 0, 0.1)
R9	(0.8, 0, 0.2)	(0.8, 0, 0.2)	(0.9, 0.1, 0)
R8	(0.7, 0, 0.3)	(0.9, 0, 0.1)	(0.8, 0.2, 0)
R7	(0.9, 0, 0.1)	(0.9, 0, 0.1)	(0.8, 0.2, 0)
R6	(0.9, 0, 0.1)	(0.9, 0, 0.1)	(0.8, 0.2, 0)
R5	(0.9, 0, 0.1)	(0.7, 0, 0.3)	(0.7, 0.3, 0)
R4	(0.8, 0, 0.2)	(0.8, 0, 0.2)	(0.8, 0.2, 0)

1. The initiating events have the usual probability p assigned to true and the remaining $(1-p)$ is assigned to false. The rule variables have the reliability factor assigned to true and the remaining non-true mass assigned to \sphericalangle (T,F). The rule variables correspond to information that each of the rule has held, eg, in 8 out of 10 trials. In the 2 remaining trials, there are not sufficient informations to validate or invalidate the rule.
2. The reliability factor is assigned to true and the remaining non-true mass on both the rule parameters and the initiating events is assigned to the \sphericalangle (T,F) term. There are no sufficient information to conclude whether the event or rule failed. The remaining mass is assigned to \sphericalangle (T,F) term - it is better than assigning the remaining mass to True or False.
3. The evidence is precise on the rule validations but imprecise on the initiating events. There is the situation of a good appreciation of the logic structure of the engineering system but there are problems with the devices. The mass assignments can be obtained from relations (1) and (2) together with the values from Figure 1 For our case study we made our calculation in Microsoft Excel. Our first calculation was made based on relations (3) - (9) and the results are given in Figure 5, and then we made the calculus based on relations (10) - (16) and the results are given in Figure 6.

By analyzing the results from Figure 5 and Figure 6 we conclude that it is obvious more realistic to compute the probability of the T event with DST than without it.

On the other hand, the results of the three cases considered by us are given in Figure 7, Figure 8 and Figure 9.

In Case I (Figure 7) the mass is distributed on all 3 elements. The top event occurs with probability 0,6213 with the remaining non-true mass on the \sphericalangle (T,F) and on the false elements.

In Case II (Figure 8) the remaining non-true mass on the intermediate and final events of the fault tree is assigned to the false element of the consequences. This is due to the allocation of zero mass to the

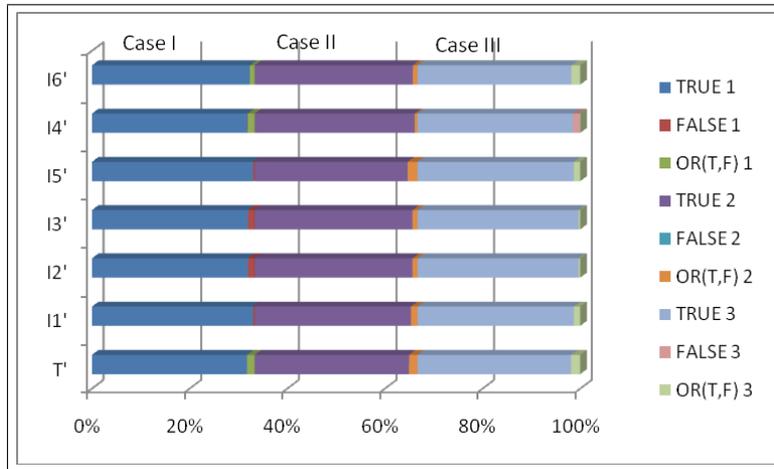


Figure 5: Calculus without imprecision on rules

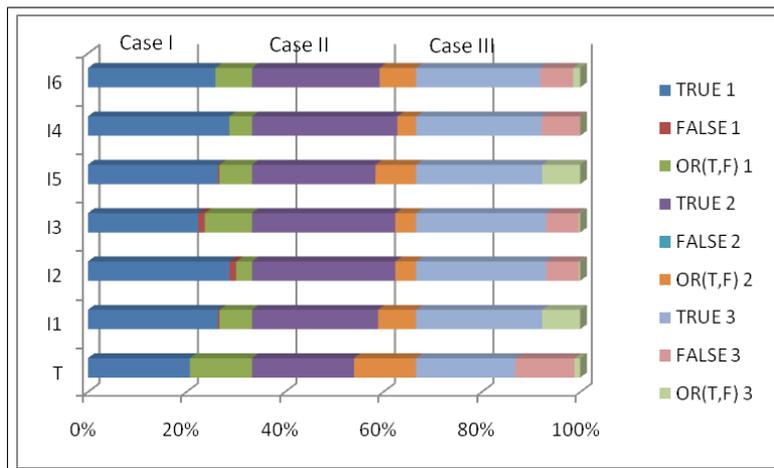


Figure 6: Calculus with imprecision on rules

false elements of the initiating events and rules. The mass of the true element of T is 0,6218, with the remaining 0,3782 mass distributed to the uncommitted term.

The Case III (Figure 9) reflects the most realistic case with fuzzy input data on the initiating events but clear knowledge of the logic rules. The resulting probability assignments from Figure 9 reflect the distribution of all 3 elements.

Cases I, II and III offer interesting comparisons. In case I we have only vague information on the accuracy of the rules, while in case III we have vague observations of the initiating events. The numerical estimates for the true elements are the same in both cases because both assume the same mass assignments for the true elements of the initiating events and rule parameters. Apparently, the effects of the uncertainty on the rules do not begin to offset the certainty on the initiating events until the mass is propagated up through the majority of the Boolean gates.

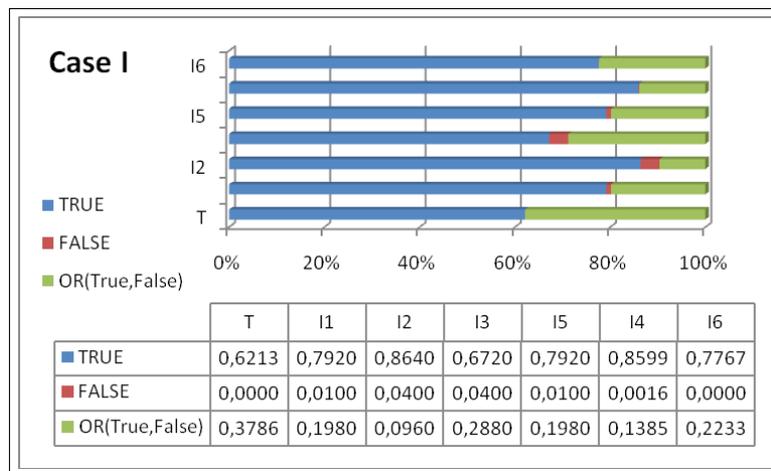


Figure 7: Mass Assignment for Case I

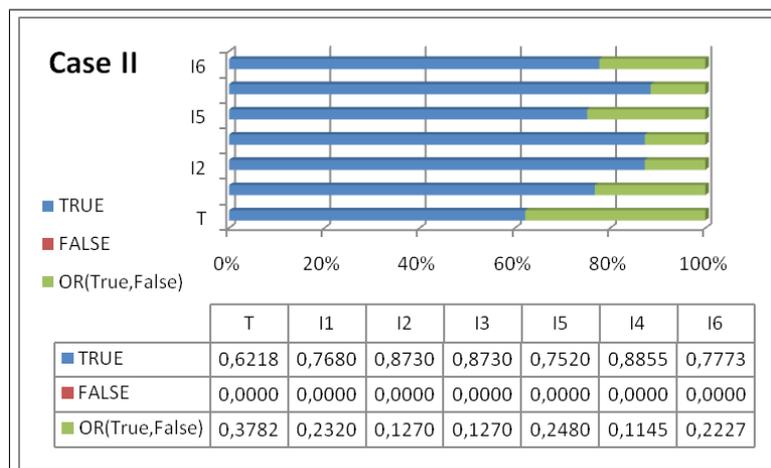


Figure 8: Mass Assignment for Case II

4 Conclusions

This paper illustrate the advantages of using DST methodology (acting on binary state space with either true or false elements) for representing vagueness and imprecision in reliability analysis of net-

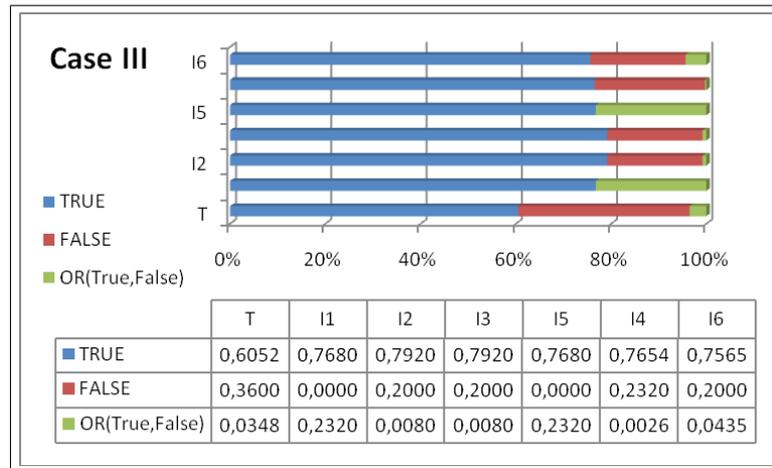


Figure 9: Mass Assignment for Case III

works. We used DST probability assignments for the component of fault trees and a separate parameter on each Boolean rule to show how a pattern of false-negative can be observed.

So, DST offers a more accurate representation of knowledge.

Therefore, when there is incomplete knowledge or a limited database upon which to make probability assignments, DST offers a clear advantage over binary (true, false) assignments in representing vagueness.

Bibliography

- [1] Stephen D.Unwin, *A fuzzy set theory foundation for vagueness in uncertainly analysis*, Risk Analysis vol.6, num I. 1986, pp.27.34
- [2] Arthur P. Dempster, *Upper and Lower probabilities induced by a multi-valued mapping* Ann Mathematical Statistics, vol.38, 1967, pp.325-339
- [3] Glenn Shafer, *A Mathematical Theory of Evidence*, 1976, Princeton University Press
- [4] Glen Shafer, *Bayes's two arguments for the rule conditioning*, Ann.Statistics, vol.10, 1982, pp 1075-1089
- [5] Glen Shafer, *The Combination of evidence*, Int'l J. Intelligent Systems, vol.I, num.3, 1986, pp.155-176
- [6] Henry Prade, *A computational approach to approximate and plausible reasoning with applications to expert systems*, IEEE Trans. Pattern Analysis and Machine Intelligence, vol.PAMI-7, 1985, May
- [7] Michael A.S.Guth, *A Probability Foundation for Vagueness & Imprecision in Fault Tree Analysis*. IEEE Trans.on Reliability, vol.40, no.5, 1991, dec.

Daniela Elena Popescu (b. June 27, 1961) received her PhD in Computer Science (1998) from the "Politechnica" University Timisoara, Romania. Since 1990 she is working within the Department of Computer Science, Faculty of Electrical Engineering and Information Technology, University of Oradea, Romania, currently position occupied being professor. She is member in the Computer Architecture and Computer Testing research group and she has written more then 80 papers in international journals. Her current main research field of interest is in Computer Architecture and

Digital Circuits Design, Computers Networks and Computational Intelligent Methods. She is also member in Hungarian Technical Academy.

Lonea Alina-Madalina (b. July 16, 1984) is PhD Student at “Politehnica” University of Timisoara. She is studying “Optimisation in Grid Systems”. Her previous projects are in the following fields: Network Server Management, Network Systems, Web Application Development, Advanced Databases, Bash Scripting and Research Management Skills.

Doina Zmaranda (b. July 14, 1967) received her MSc in Computer Science (1990) and PhD in Computer Science (2001) from the "Politehnica" University Timisoara, Romania. Since 1990 she is working within the Department of Computer Science, Faculty of Electrical Engineering and Information Technology, University of Oradea, Romania, currently position occupied being professor. Her scientific research is focusing on real-time application development and programming. In addition, she also investigates issues related to reliability of complex control systems. She has (co-)authored 5 books and more than 20 papers in international journals in the last five years, participating also within several research projects.

Codruta Vancea (b. August 7, 1967) received her MSc in Informtics (1990) and PhD in Mathematics (2003) from the University "Babes Bolyai" of Cluj Napoca, Romania. Since 1991 she is working within the Department of Electrical Engineering, Electrical Measurement and Electric Power Use, Faculty of Electrical Engineering and Information Technology, University of Oradea, Romania. Her current position is assistant professor. Her scientific research is focusing on modeling of electromagnetic problems and parallel computing.

Cristian Tiurbe (b. October 22, 1979) received his MSc in Computer Science (2003) from the University of Oradea, Romania. Since 2002 he is working within the Department of Computer Science, Faculty of Electrical Engineering and Information Technology, University of Oradea, Romania, currently position occupied being assistant. His scientific research is focusing on computer networks security.