UNDERSTANDING MONTE CARLO METHOD USING @RISK FOR PROJECT

Valentin Petru MĂZĂREANU
The Faculty of Economics and Business Administration, “Alexandru Ioan Cuza” University of Iasi
Iasi, Romania
vali.mazareanu@feaa.uaic.ro

Abstract

All the elements of the risk management cycle are important. But risk assessment provides the foundation for all the other elements. In particular, risk assessment represents the basis for establishing the appropriate policies and for selecting the most efficient techniques from the point of view of the implementing costs of these policies. As long as risks and threats change in time, it is important for the organization to periodically perform a risk re-assessment and a reconsideration of the efficiency of the established policies and control methods.

The risk assessment process involves actions of risk analysis and quantification and of the interaction between risks, with the purpose of anticipating the results of the project. It is a difficult process, because many events can influence it in unexpected ways (i.e. the mathematical techniques used can create a false impression of precision and trust, a single risk can cause multiple effects).

One of the methods of performing the risk assessment within a project or a practical situation is that of simulation. Simulation represents a process for determining the decisions with the help of some models that constitute simplified representations of some real systems.

In the Monte Carlo model, which is based on the decisional simulation, a series of estimations are provided that represent the most favorable case, the most likely case and the most unfavorable case with the purpose of showing how the project will progress and how it will fulfill an estimated closing date.

The Monte Carlo simulation is used by various risk analysis software programs. One such product is also @Risk – an operational risk management solution in projects that transforms Microsoft Project into a strong instrument for risk analysis and assistance in making strategic or operational decisions.

In the present paper we will present the connection between decisional simulation, the Monte Carlo method and the stages of solving a problem with the @Risk for Project instrument having as model a software product development and implementing project.

Keywords
Monte Carlo, Risk Management, Decision Simulation

1. INTRODUCTION

The risk assessment process requires information. The more information goes into this process, the higher the accuracy degree for the assessment of a risk will be. The risk identification process (the previous process according to the methodology proposed by Project Management Institute) [1], has provided the object categories that must be protected, has shown which are the risk sources and the events with risk potential that threat those objects. The connection between the activities of the company or between the activities foreseen in a project and risks is a complex one and represents the starting point in the assessment. An approach in the spirit of the risk centers can be used here, this method ensuring more detailing levels. Then one goes on to estimations of costs (assessments of the cost of the necessary resources to achieve the activity) and estimations of the duration of the activities (assessments of the number of units, hours, days, etc., necessary for the fulfillment of an activity).
For the assessment the sensitivity analysis can be chosen, the technique of expected monetary value [2] or the one of decisional trees, the Delphi method or the utility theory (which also emphasizes the attitude towards risk of the decision-maker), two-dimensional matrix (impact vs. occurrence probability assessment) or, as proposals, a multidimensional matrix in which to also consider the moment when the assessment takes place, a certain moment in the life cycle of the system or the psychological characteristics of the person responsible for the good performance of an activity [3].
There can also be chosen instrument of the decision theory [4] that provide solving techniques for difficult problems: the ones that are complex, contain uncertain aspects, have multiple objectives or meet different perspectives. The decision theory uses probabilities to determine the results. The techniques for structuring difficult problems include decisional trees and computerized simulations. So one method to achieve the risk assessment within a project or a practical situation is the one of simulation.
The simulation represents a decision establishing process with the help of some models that constitute simplified representations of some real systems.
The simulation of economic decisions can be applied to all classes of problems that comprise functioning rules, policies and procedures like the ones regarding decision adaptation, decision control and price policy. The solving of problems with the help of the simulation techniques entails the use of some interactive algorithms and the existence of some well determined steps in view of reaching the objective entailed.
The entry dates are, usually, random variables obtained following their generation by a random numbers generator.
The "Monte Carlo" method [5] is based on the use of such random variables, because for the models that involve the existence of a high number of decisional variables, the method necessarily uses the calculus technique, and the algorithm of the method is presented in the succession of its interactive stages.
The Monte Carlo simulation allows for the attributing of probability distributions to uncertain elements (i.e. of the project activities) following that afterwards thorough the simulations (random sampling of these distributions) to determine the distribution of all potential outcomes that could occur under these uncertainties.

In the Monte Carlo model there are provided a series of estimation that represent the most favorable case, the most likely case and the most unfavorable case. For each of these cases, a probability is also assigned.
The simulation shows how the project will progress and how it will meet and estimated closing date. The project is then decomposed again, this time using different probabilities, resulting a different closing date. The reason for which the model is run several times is to give the risk percentage the possibility to be reflected in the result.

2. The decisional simulation with Monte Carlo in @Risk for Project

The Monte Carlo simulation is used by various risk analysis software programs. Such a product is also @Risk, an operational risk management solution ion projects. More precisely, it is about a suite of applications - Decision Tools Suite, from Palisade Europe – he transforms Microsoft Excel or Microsoft Project in a strong instrument for risk analysis and assistance in the making of strategic or operational decisions. The solution works for any industry, no matter the object of activity or the size of the business. The basic instrument of the suite, @RISK, offers a great calculus power using the Monte Carlo simulation, a mathematical technique for the numerical solving of differential equations. By using the simulation technique @Risk combines all the uncertainties identified in projects to run an „what if" type analysis in view of identifying all the possible results of the project.
In a comparative approach, the stages for solving a problem according to the @risk for project instrument are:

Table 1 Stages for solving problems (a comparative approach)

@risk for project [7]	Monte Carlo Method [8] [9]	Decision Simulation
Developing the model	Designing the structure of the model	Determining the field to be simulated
Defining the uncertainties	Determining the input uncertainty in the model	Establishing the factors that act in the simulated field. Developing the economic and mathematical models that describes the simulated processes
Analyzing the model through running the simulations	Techniques for modeling dependencies	Developing the electronic programs Simulation
Evaluating the results and formulation of a decision	Presentation and interpretation of risk analysis results	Formulating a decision

1) First step – Developing a model

In this stage the developing of a model that defines the problem or the real situation with which the company is confronted is targeted with the purpose of analyzing that problem from all the possible points of view. The curacy of the model (input) that will be submitted to the simulation has a direct reflection in the accuracy of the results (output).

We have chosen for this presentation, the analysis of the risk associated to a project for the implementation of the software solution. The project is realized in Microsoft Project, as presented in the figure below.

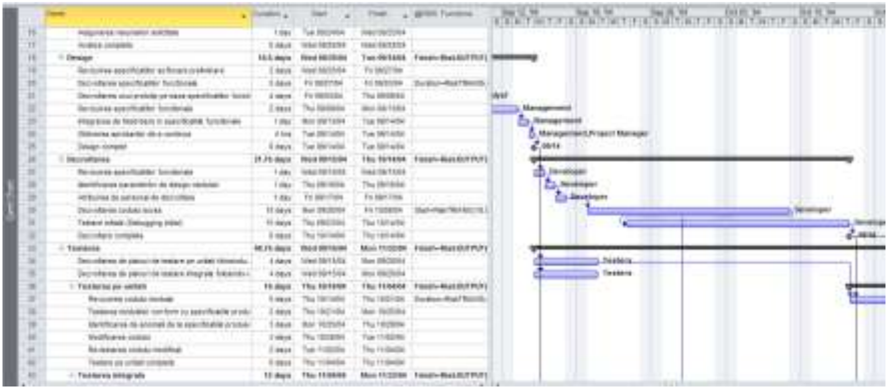


Fig. 1 The Software Development Project (Microsoft Project Planning)

This project is meant to have started on the date of 02.08.2004 and should have been finished on the date of 13.12.2004.

The performance of a project, whatever its nature, is an intimate activity for the project team, thus the complexity of an activity could only be known be the team members. Without performing a risk analysis, it is only possible to identify the critical activities and to separate them from the non-critical ones. With the help of a risk analysis utility, such as @Risk for Project, through which uncertainty can be added to certain activities, one can get to obtaining information of the type: activity 1 has a critical index of 55%, activity 2 a critical index of 85% etc.

2) Step 2 – Defining uncertain elements

- In order for the results of the simulation to have relevance it is necessary to take into consideration all the possible values for these variables (uncertain elements).
- In this stage the Probability Distributions Functions – PDF are used, which describe the range of possible values for a variable and the probabilities associated with these values.
- Let us assume that from the discussions with the members of the software product implementing / realization team (expert consultation) and from the historical data we have, we identified the following risk areas (we will mention in the end what each function represents):

- Determining the scope of the project – activity that also represents the starting point in realizing the project. Although on the date of 15 May 2004 it was decided that this project will start on the date of 02.08.2004, this is still uncertain. Also it is wanted that this activity is framed within a 4 hour time period, with a deviation of minimum/maximum one hour.

Start=RiskTRIANG(7.28.2004, 8.03.2004, 8.05.2004)
Duration=RiskNORMAL(4, 1)

- Analysis of needs – it is wanted that this activity be framed within a period of maximum 5 days, minimum 2 days.

Duration=RiskUNIFORM(2, 5)

- Elaboration of the preliminary specifications – it is assumed that this operation should last 3 days, but this is again uncertain.

Duration=RiskTRIANG(2, 3, 5)

- Development of the functional specifications – although it is presupposed that this activity should last 5 days, this is not yet certain.

Duration=RiskTRIANG(4, 5, 6)

- Development of the source code – the importance of this activity determines the development team to try and start this activity on the date of 20.09.2004, which is also uncertain.

Start=RiskTRIANG(9.18. 2004, 9.20.2004, 9.22.2004)

- Revision of the modular code – activity wished to last for 5 days, also uncertain.

Duration=RiskTRIANG(4, 5, 6)

- Testing the integrity of the modules – activity wanting to last 5 days, uncertain.

Duration=RiskTRIANG(4, 5, 6)

- Development of the specifications for end users – activity wished to last for 3 days, also uncertain.

Start=RiskTRIANG(9.14.2004, 9.15.2004, 9.16.2004)

- Revision of the documentations for all users – activity wished to last for 2 days with a one day deviation.

Duration=RiskNORMAL(2, 1)

- Installing/running the software - an activity wished to last for one day and to start on the date of 22 November 2004.

Start=RiskTRIANG(11/01/2004, 11/22/2004, 11/25/2004)

- Determining the final strategy for running – activity that must fit into the duration of minimum one day, maximum 3 days.

Duration=RiskUNIFORM(1, 2)

- Performing the transfer of knowledge/learning from mistakes – activity wished to last for one day, also uncertain; a deviation of half a day is allowed.

Duration=RiskNORMAL(1, 0.5)

We left for this analysis only these areas of uncertainty. Notice that fact that these uncertainties can only be “thought” for each activity in part and it all depends on the project realization team and on the way each sees his/her part of the project. Uncertainties can also be applied on the method of administration of material or human resources.

For the analysis of the uncertainties of a project, whether it is about the starting date for an activity or sub-activity, or the duration or an activity / sub-activity, or about uncertainties applied to the production / acquiring of material costs, of the costs with salaries etc, the @Risk for Project software uses a series of functions to determine its margin of error and to generate certain results that will be later taken into account, in the Monte Carlo simulations, such as [7], [5]:

- NORMAL (a, b), where a means normal value and b standard deviation.
- PARETO (theta, a), Pareto distribution.
- UNIFORM (min, max), where min and max represent the minimum and maximum value of the field.
- TNORMAL (mean, standard deviation, min, max), where mean represents the most probable value, standard deviation as it say and min and max the minimum and maximum value.
- SIMTABLE (val.1, val.2, val.3, ..., val.n), represents a list of values that a field can take; these values will be taken into account in simulation one by one.

The functions used in this analysis are: *RiskTRIANG*, *RiskNORMAL*, *RiskUNIFORM*.

Start=RiskTRIANG(28.07.04, 03.08.04, 05.08.04), takes into consideration the fact that the activity (in our case is the start of the project)may actually start somewhere on:

- 28.07.2004 – the minimum value;
- 03.08.2004 – most likely;
- 05.08.2004 – maximum value.

The function *Duration=RiskNORMAL(4, 5)* states the fact that the normal duration of the activity is 4 hours, but may have a standards deviation of 5 hours.

The function *Duration=RiskUNIFORM(2, 5)* states that the duration of this activity has to be somewhere between a minimum value of 2 (days) and a maximum of 5 (days).

The function *Duration=RiskTRIANG(2, 3, 5)* states that the duration of this activity may be:

- 2 – the minimum value is 2 (days);
- 3 – most likely it will be 3 (days);
- 5 – the maximum value is 5 (days).

As noticed in figure 1, in the project worksheet another column emerged, called *@Risk:Functions*, column that comprises the uncertainty functions applied to each activity taken into calculation of the risk analysis.

In order the see which is the result of this analysis, we will need the identification of the result activity of the project. This can also be done for intermediary activities of the project.

We will mention thus that the main activity is Project – developing a software, and the field *@Risk: Functions* will have the value, *Finish=RiskOUTPUT().*

The same value will also be mentioned for each of the main activities of the project.

3) Step 3 – Analysis of the model by running a simulation

Through the simulation, the model is calculated repeatedly, @RISK using each time a different set of possible values of the variables in the input.

The instrument tries all the valid combinations of the variables detected in the input in order to simulate all the possible effects. It is like running hundreds or thousands of „what if" analyses, changing all the variables in the input at the same time, for each iteration.

These being mentioned, we will start the project analysis, by running the simulations on the uncertain values we specified, and we determine that the application performs a number of 100 recalculations**.**

In order to see a list of the inputs (declared functions of uncertainty) and of the declared outputs (result activities), we can turn to the option List Inputs by Outputs (Fig. 2). Also from here certain distributions can be blocked so as not to be taken into account in a future simulation.

Fig. 2 List Inputs by Outputs

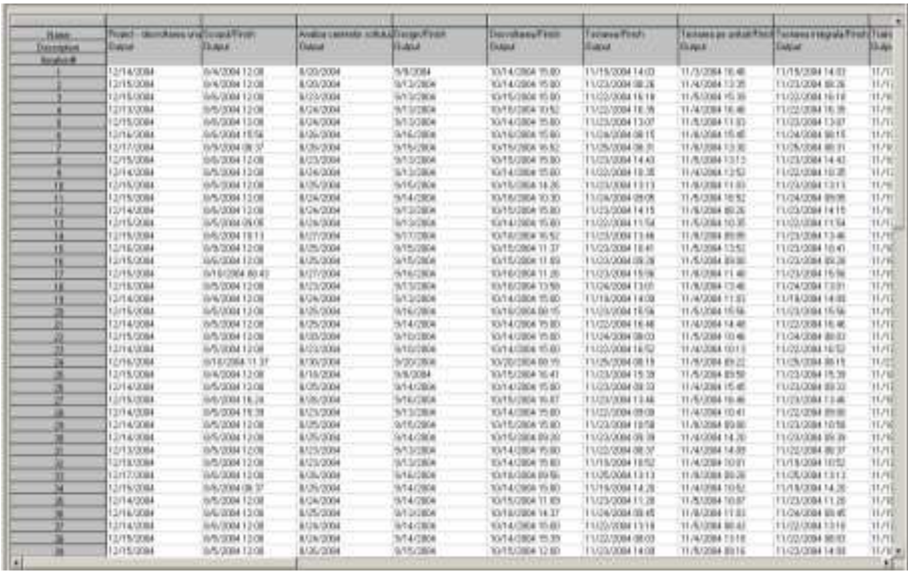
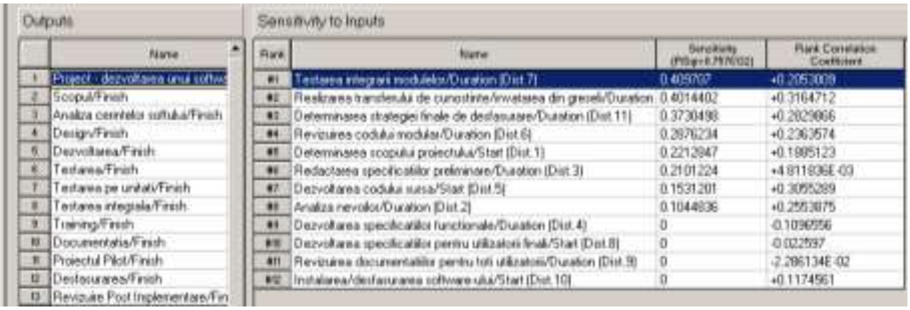


Fig. 3 Simulation Data

The final result of the analysis can be tracked by using the *Run Simulation* command. All the calculations (the 100 mentioned iterations) performed by the @Risk utility can be visualized from the window Simulation Data (Fig. 3).
The Sensitivity option presents the (most) critical variables for each of the wanted result activities, obviously including for the final result (Fig.4).



Outputs

	Name
1	Project - dezvoltarea unui softw
2	Scopul/Finish
3	Analiza cerintelor sufului/Finish
4	Design/Finish
5	Dezvoltarea/Finish
6	Testarea/Finish
7	Testarea pe unitati/Finish
8	Testarea integrala/Finish
9	Training/Finish
10	Documentatia/Finish
11	Proiectul Pilot/Finish
12	Desfasurarea/Finish
13	Revizuire Post Implementare/Fin

Sensitivity to Inputs

Rank	Name	Sensitivity	Rank Correlation Coefficient
#1	Testarea integrarii modulelor/Duration (Dist.7)	0.409707	+0.2053009
#2	Realizarea transferului de cunostinte/invatarea din greseli/Duration	0.4014402	+0.3164712
#3	Determinarea strategiei finale de desfasurare/Duration (Dist.11)	0.3730498	+0.2029866
#4	Revizuirea codului modular/Duration (Dist.6)	0.2876234	+0.2363574
#5	Determinarea scopului proiectului/Start (Dist.1)	0.2212947	+0.1885123
#6	Redactarea specificatiilor preliminare/Duration (Dist.3)	0.2101224	+4.811836E-03
#7	Dezvoltarea codului sursa/Start (Dist.5)	0.1531201	+0.3055289
#8	Analiza nevoilor/Duration (Dist.2)	0.1044836	+0.2553875
#9	Dezvoltarea specificatiilor functionale/Duration (Dist.4)	0	-0.1096556
#10	Dezvoltarea specificatiilor pentru utilizatorii finali/Start (Dist.8)	0	-0.022597
#11	Revizuirea documentatiilor pentru toti utilizatorii/Duration (Dist.9)	0	-2.286134E-02
#12	Instalarea/desfasurarea software-ului/Start (Dist.10)	0	+0.1174561

Fig. 1 Sensitivity Analysis (most critical variables for the mail output)

From here there can be generated the chart that presents the relative meaning of each inserted variable (Inputs) in determining the value of the result (Output). The chart for the final result of the project is presented in Fig.5.

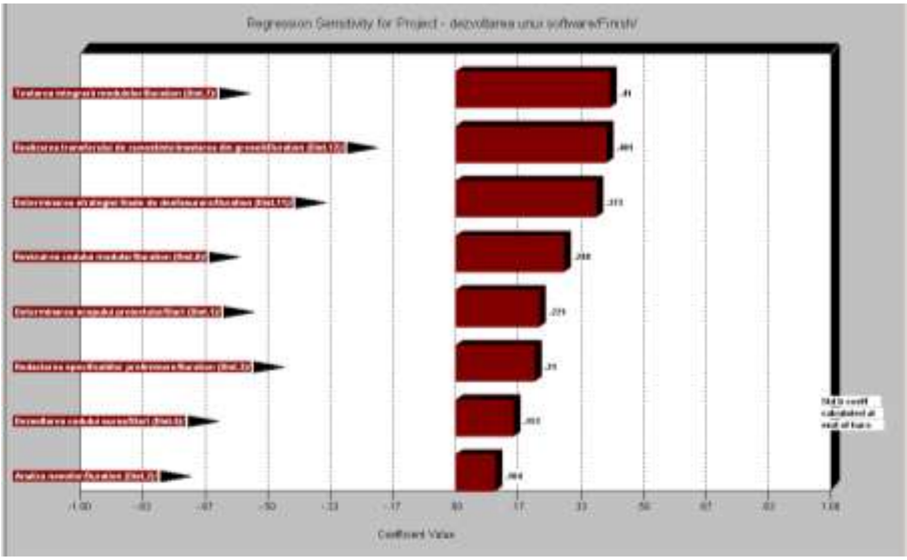


Fig. 2 Regression Sensitivity Graph

When two activities are in close connection, there is the possibility to perform a correlation between their values. This is only necessary when one wants to eliminate the excess iterations. The best example would be the connection between duration and cost applied on the same activity. The values a field can take

(represented by the meeting place of two probability functions) in this situation is comprised between –1 and +1.
In the case of our project, we specified the fact that the activity Elaborating preliminary specifications has a variable duration, mentioned through the function *Duration=RiskTRIANG(2, 3, 5).*
To this activity we also allocate an initial budget (Budget necessary for elaborating the preliminary specifications) of 1000 USD, also mentioning though the uncertainty to respect this roof through the function *Std.Rate=RiskUNIFORM(1000, 2000).*

4) Step 4 – Evaluation of the results and proposing a decision

@RISK generates a distribution of the possible effects a model can have and indicates the probability that each of these effects occurs. The range of probabilities that emerges following the Monte Carlo simulation is possible to be different and more accurate than the results of some traditional analyses, of the type “worst case - best case".
With the help of the Monte Carlo method there are emphasized the results that are more likely than the others and that should gain more weigh in the assessment.
Certain scenarios can also be automatically generated (Fig. 6)
In our case there are generated 3 scenarios, each presenting certain stages to be followed, activities to be performed within a certain term (respectively, budget) that ensures the fulfillment of the project within a certain percentage (Scenario 1 – for a probability starting with 75%, Scenario 2 – for a probability of up to 25%, Scenario 3 – that could ensure the performance of the project in a percentage of more than 90%). For each of these scenarios the (most) critical activities are also presented.

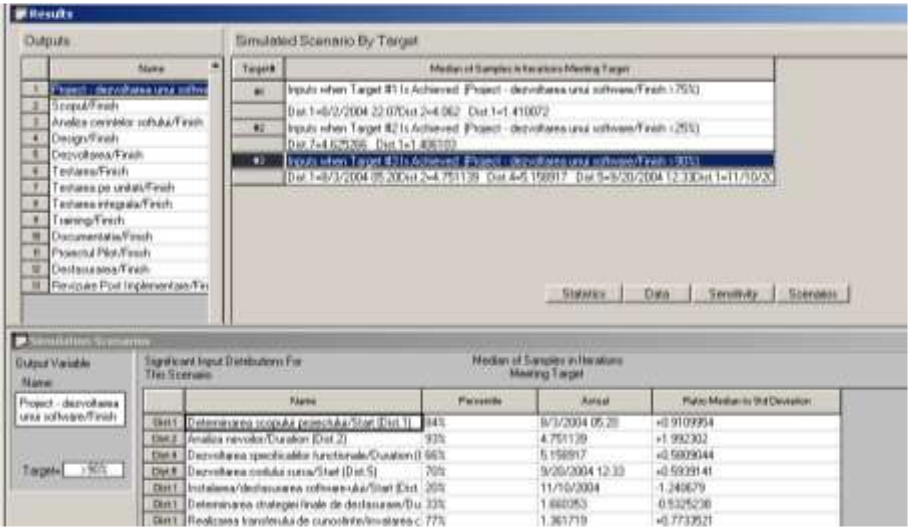


Fig. 6 Scenarios Options

Thus, the probability that the project is finished on the date of 13 December 2004, varies between 3,84% and 15%, depending on the various functions taken into consideration.
With a percentage of 95% of finality of the project in normal standards is presented the date of 17 December 2004. The results of this simulation can also be identified in Microsoft Project, in the column *@Risk:95% Perc*.
Also from this window there can be generated a chart of the probability of finality of the project (Fig. 7).

Fig. 7 The probability for the project to end

Also now various filters can be introduced for a new simulation, in which certain activities take values only between a declared minimum and maximum.
Also, there can be verified the finality probability of the project on a certain date that interests. Thus, for the date of 20 December 2004 we obtain a probability of 97.89%. for the date of 21 December 2004 the probability is of 100%.
These analyses can also be applied on each of the declared result activities (Output).

3. CONCLUSIONS

When we speak about risk management we understand that it means we are dealing with a complex activity which involves, among others, a strong relationship between members of the project team in terms of project information sharing or applying some complex risk management models in the project plan. All these arguments, but not exclusively, justify the use of the specialized tools that can assist the risk management activities.
The great benefit is provided by speed of work. Once the data are collected and filled into the system, any operations (budget sheets, schedules, plans etc) can be done in minutes. And with intranet/internet technologies all these can be done from outside the decisional office.

Another major benefit is economy. In most of the cases the computer is providing important advantages in terms of cost comparing with the manual system. Supposing that the data were filled in correctly, the possibility to make mistakes in processing them are reduced to minimum and updating them can be done with low cost.

There are many software solutions for project risk management: RiskID (Accela Research), a tool for identifying and evaluating risks in IT projects, RiskRadar (Integrated Computer Engineering Directorate), a tool used in quantification, categorization, prioritization, tracking, controlling and managing risks, Cobra - Consultative, Objective and Bi-functional Risk Analysis (C&A Systems Security), tool for cost management, budget analysis and what if analysis. And many more [10].

Such a tool is @Risk for Project from Palisade Corporation, which uses a technique called simulation to combine all the uncertainties identified in a project. This tool allows defining uncertain values in a project as probability functions using standard @Risk distribution functions.

Although widely accepted and used, Monte Carlo methods and so the @Risk for Project has some weakness because is a unidirectional model and does not offer some interactive link between data and parameters. Even so, properties like simultaneous consideration of threats and opportunities and probability of selecting various criteria [11] make from Monte Carlo simulation one of the preferred methods of generating probability distributions of exposure and risk.

References

[1] Project Management Institute, A Guide to the Project Management Body of Knowledge, 4th Edition, Project Management Institute Inc., Pennsylvania, 2008

[2] Crouhy, M., Galai, D., Mark, R. Risk Management, McGraw-Hill, New York, 2000

[3] Măzăreanu, V.P., Economia Digitală și Managementul Riscurilor, Tehnopress, Iași, 2010

[4] Hall, E.M., Managing Risk: Methods for Software Systems Development, Addison Wesley, 1998

[5] Jorion, P., Financial Risk Management Handbook, 4th Edition, John Wiley & Sons, New Jersey, 2007

[7] @Risk for Project, Advanced Risk Analysis for Project Management, Palisade Corporation, 2000

[8] Pandian, C.,R., Applied Software Risk Management - A Guide for Software Project Managers, Auerbach Publications, Boca Rato, 2007

[9] Molak, V. (ed.), Fundamentals of Risk Analysis and Risk Management CRC Press, Inc, Boca Raton, 1997

[10] ENISA, Technical Department, Section Risk Management, Risk Management: Implementation principles and Inventories for Risk Management/Risk Assessment methods and tools, June 2006

[11] Rezaie K., Amalnik, M.S., Gereie, A. , Ostadi, B., Shakhseniaee, M., Using extended Monte Carlo simulation method for the improvement of risk management: Consideration of relationships between uncertainties, Applied Mathematics and Computation, Elsevier, 2007

Acknowledgements

The results presents in this paper were obtained in the framework of the postdoctoral school programme financed by the *"Developing the Innovation Capacity and Improving the Impact of Research through Post-doctoral Programs", No: POSDRU/89/1.5/S/49944* project.