

Stochastic Dynamic Simulation Model Applied to Public Lawyers using Petri Nets

Felisa M. Córdova^{a,*}, Fernando Cifuentes^b

^aSchool of Engineering, Faculty of Engineering and Business,
University Finis Terrae, Av. Pedro de Valdivia 1509, Santiago 05600002, Chile

^b Faculty of Engineering, University of Las Americas, República 71, Santiago 0560002, Chile

Abstract

The aim of this article is to present a method of assignment of defence lawyers to causes, using tools Petri nets, decision trees and Monte Carlo method. It presents a simulation model of assignment applied to law activities, specifically to defender lawyers contracted by tender, using Petri Nets, Montecarlo and decision trees. The main difficulty of this assignment task is how to distribute equally the workload along the year between two kinds of lawyers: defender lawyers' staff and external (tendered) defender lawyer. The system utilized for analyzing task assignment is stochastic, parallel, concurrent and dynamic.

© 2016 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

Peer-review under responsibility of the Organizing Committee of ITQM 2016

Keywords: Petri Nets, Simulation Models, Decision Trees, Montecarlo, Defender Lawyers, Tender.

1. Introduction

The Public Defender Service Office is a state agency whose duty is to provide support to the defendants (accused) in criminal cases at their request, within a period not exceeding 90 days from the indictment. This is organized in Regional defender lawyers' offices (in each region), and each of them, in Local offices. Each Local Office assigns Defender Lawyers (Defence Lawyers) to different courtrooms within its jurisdiction courts, assignment that depends on the Local Chief defender Lawyer [1]. Generally, courtrooms assignments to Defender Lawyers depend on the number of cases they are dealing with and the projection of days that Local Chief Defender perceives that each Defender will take to solve the pending cases. Thus, in an instant during the year each defender has a set of cases that he is processing plus the cases that appear in the courtroom in which he is assigned that day, which can be solved the same day or can be passed to the set of pending lawsuits.

On the other hand, in order to fulfil its mission, Public Defender Service Office annually bids Public Defender Lawyer positions to private law firms, which contribute with a group of defender lawyers. These defender lawyers work in the Public Defender's office as long as the tender contract lasts. So, in a particular period of the year, the Public Defender Service Office has available hired lawyers ("Local Defenders") and

* Felisa Córdova Tel.: +56-02-23260268; Fax: +56-02-23260268.

E-mail address: fcordova@uft.cl

those from tender agreement.

The terms of this tender agreement with these law firms set that they have to contribute with the workforce (Tendered Defenders Lawyers) to the Public Defender's Office according to a fixed number of yearly cases (which were 470 cases/yearly for each Defender Lawyer, reaching a total of 10 Tendered Defender Lawyers in the studied "comuna" (term equivalent to county).

The difficulty is that, for a particular time during year, historically between August and November, the number of cases agreed with these law firms in the tender contract expires, and all the remaining pending work, together with remaining work until year end, lies with Local Defender Lawyers. In the studied year (2006), the cases were assigned with a single national criterion based on the total number of cases that a Defender Lawyer can handle during one year.

On the other hand, the type of crime, the geographical characteristics of each region, the distribution of courts, the type of management associated to the type of crime, legal procedures, investigations, audiences, etc., suggest that there is a dispersion in the effective time intended for similar cases, and especially when comparing between a simple case and a complex one. This phenomenon is beyond the simple randomness of stochastic processes and rather, is due to reasons that could be explained for the nature of the operations in each region, and the type of crime, and may even influence the Defender Lawyer's experience.

The problem modeled in this article is the allocation of cases to Defender Lawyers. The amount of accused for each case is, for simplicity one. The comuna studied matches this condition (in 2006, 94.94% of cases had n accused).

1.1. Montecarlo Generator

As a general rule, it's called Montecarlo any mathematical procedure in which random number generator is extensively used. The utility of this method, for a simulation model, is in the fact that, if there is no distribution for a specific event, you can compare a random number generated by a continuous distribution between zero and one, with a relative frequency of that event. If the generator is reliable, the probability of generation for each number in this range should be equiprobable [2].

The probability for the average absolute error to be less than 14.64% is 99.70%. In practice, what is done, bearing in mind the previous result is:

- To create a partition within the open interval (0, 1), so that the relative frequency of occurrence of a phenomenon corresponds bijectively to this section.
- To generate the appearance of a random number between zero and one, with uniform distribution. This assumes equal probability of appearance of any real number in this section, if the generator is reliable.
- To compare the generated number with the classes and locate its ownership. If it belongs to one of the given partitions, the feature associated with the class "fires".
- The fired property is entered to one of the components of the token that acts as a vector, and travels through the system.
- For another feature, the above steps are repeated, and so on.

1.2. Petri Nets

Petri nets allow the construction of several independent modules that produce a configuration that is determined from a combination of these modules, collaborating in this way to the modeling of the defense process [3]. Petri Networks are being used for modelling dynamic operations of discrete systems, mainly in manufacturing [4-5]. They are also utilized like a very useful tool for modelling, to analyse, to simulate and to control production systems [6]. Hierarchical and temporal Petri Nets also allow the construction of a simulation model with a knowledge based system built in, which is parametric, scalable and adaptable to various

configurations of processes [4].

Three different techniques were used in the development of the proposed model, techniques that have not been used together. Petri Nets and Montecarlo were used to model the generation of values of variables within the model. Decision trees were used for managing the elements.

In Petri Nets, each element represents a characteristic that is present in any system to be modeled. Their operations are based on four elements: *places, transitions, tokens and arcs* [7]. In this model, token acts as a vector of information: every component represents a characteristic (variable).

The two main purposes of this model are: first, emulate a Court for one year, and later, probe different scenarios to obtain well balanced workload. The simulation model was built using the PACE™ software.

1.3. Decision Trees

A decision tree is a predictive model used in the artificial intelligence field. Given a database construction, logical diagrams are constructed pretty similar to systems based prediction rules, which are used to represent and categorize a number of conditions occurring in succession in order to solve a problem. A decision tree has entries which can be an object or a situation described by a set of attributes. In response to this the system sends a response which ultimately is a decision that is taken from the entries. The values which entries and outputs can take can be discrete or continuous. For simplicity discrete ones are more used. When using discrete values in the functions of an application is called classification and when using continuous is called regression [8].

A decision tree performs a test as vectors move towards the leaves so as to reach a decision. The decision tree typically contains internal nodes, probability nodes, leaf nodes and arcs. An internal node contains a test on a value of a property. A probability node indicates that a random event must occur according to the nature of the problem, this type of node is round, and others are square.

A leaf node represents the value that will return the decision tree branches and finally provides the possible paths that are according to the decision. In computer software design purposes, a decision tree indicates the actions to take based on the value of one or more variables. Representation is a tree whose branches diverge depending on the values that variables take and ending with a concrete action. It is typically used when the number of conditions is not very large (in this case, it is better to use a decision table).

2. Method

2.1. Model Assumptions

The way to proceed in modeling is to configure the case (modules one and two), and then assign it to the Defender Lawyer, with the constraint that each defender who attends to a courtroom can only handle a maximum of 18 cases. At this limit, another defender is assigned to the courtroom.

In case that some of the defendants is underage, has to pass to a special court, thus not being considered in this study. For purposes of the model, the defendants are people overage (18 years).

Simulation sets that the initial conditions of the tender contract, concerning to assign Defender Lawyers a fixed number of causes during the year is not efficient.

The sensitized model is used in order to determine the maximum workload that Defender Lawyers could hold under these characteristics, reaching a maximum of 15%. The criterion used is of “type time”. However, at the time of this study there was a formalization of such times, since the Defender Lawyer had not studied this aspect, therefore being created an exclusive time for this type of modeling. In conditions existing in that moment, the type time was between 5.13 and type 5.45 hours / week in the case of the non-sensitized model.

Finally, it is developed a model in which the conditions are removed (provided by the tender contract)

regarding the fixed amount of causes, proposing a model in which the workload is evenly distributed between Local Defender and Tendered Lawyers. The result of this new model is the balance in workloads.

One aspect not mentioned in this study and that the model can effectively perform, is the weekly desegregation as to workload for each Defender Lawyer, Local or Tendered, an unstudied aspect so far.

Some of them are those Reported in [3-10] and developed by [1].

Crimes were grouped in 14 categories, considering 82% of the total. It created a category 15 to contemplate the remaining 18% as shown in Table 1.

2.2. Simulation Model in Petri Nets

There are four main modules, each made up by sub modules, and two considerations: clone restrictions cause and unrestricted causes. The first one is responsible for generating, randomly, weekly crimes, both the type and the weekly amount, using a Monte Carlo generator, which considers the historical frequency of them “GENERATES CRIMES BY TYPE WEEK” (in right side in Figure 1).

Table 1. Code of crimes

Code of Crime	Name of Crime
103	Simple Theft
66	Crimes against Intellectual Property Law
140	Less Serious Injuries
139	Minor Injuries
208	Robbery with Intimidation
18	Threat of attack against persons and property
213	National Property Theft Public Use
203	Receiving
189	Carry cutting or stabbing weapon (288 bis)
215	Theft in a place not Habited
102	Lack Theft (494 bis CP)
210	Robbery with Violence
97	Homicide
187	Parricide
1000	Other Crimes

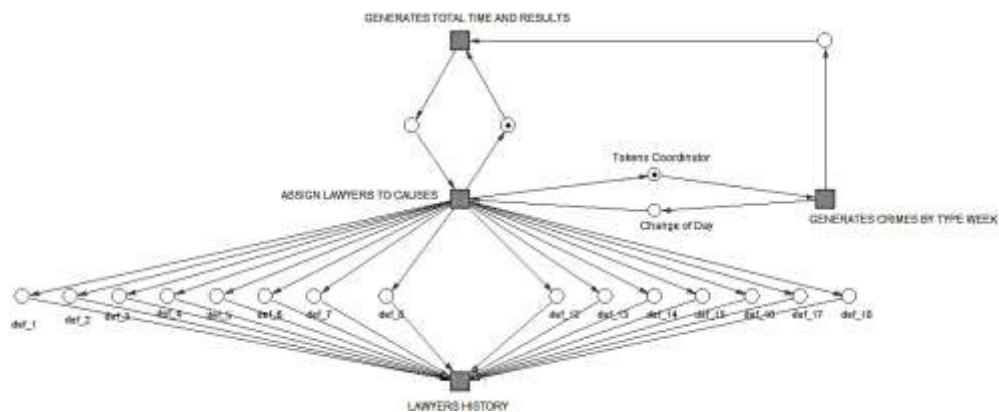


Fig.1. Model Simulation Modules in Petri Nets

The information thus obtained is stored in a vector, which goes to the second module (“GENERATES

TOTAL TIME AND TYPE RESULT”, at the top of Figure 1), in which the processing time is added within the Court and the type of output that will cause. This is done because for the same offense, depending on the circumstances and the defense of it is made, the result may be different. It also means a different processing time.

The third module “ASSIGN LAWYERS TO CAUSES (in the center of the Figure 1) specializes in assigning cases to Defender Lawyers. The issue to face in this module is: given the above data, how to represent the allocation of defender lawyers to cases.

Basically, in both the model with constraints and the one with no constraints, one can speak of two types of defender lawyers, and represent them in modules, independently of the number of defender lawyers that are in each case.

For the model with cases constraints for the Tendered Defenders, the constraint is 470 cases per each tendered defender lawyer, while Local Defender Lawyers are with no constraints.

The last module “LAWYERS HISTORY” (at the bottom of Figure 1) saves the results of the simulation.

- *Module “GENERATES CRIMES BY TYPE WEEK”*

It corresponds to the data used in the network for Montecarlo filter and then it is transmitted as a vector token inside the Petri net. After having the token with the day of the week, the information is filtered as follows. With crimes grouped by day of the week, the following parameters were empirically determined: crimes are grouped by type and its frequency is determined according to the amount accumulated per day, generating disjoint classes for each day of the week. Figure 2 shows the Module GENERATES CRIMES BY TYPE WEEK and Table 2 shows the accumulation for Monday.

After classes are created, a random number between zero and one is generated, using Montecarlo. This number matches one of the classes a set and the number of crimes that occur on that day.

This module focuses on generating total processing time of the case (procedures that the Defender Lawyer should make to handle the case) and total verdict time (or total processing time of the trial in the courtroom). The problem that arises for this module is that for the same crime, processing and total time are not necessarily equal for the same type of crime, largely depending on the evidence filed in the trial. Thereupon, the total case processing time depends on the type of output (or court decision of the trial), along with the involved paperwork.

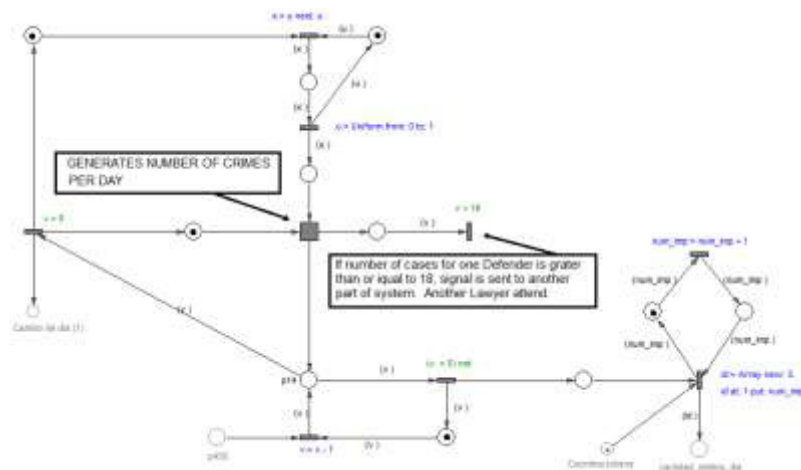


Fig.2. Module GENERATES CRIMES BY TYPE WEEK.

The input of this module corresponds to the vector coming from the module “GENERATES CRIMES BY

TYPE WEEK”, and the output of the vector contains the nested vector, the processing time (third component), and the total time or output (fourth component).

Table 2. Cumulative frequency of crimes for Monday

Cumulative Frequency for Monday [a , b [Amount of Crimes
0.0000	0.0213	9
0.0213	0.0426	17
0.0426	0.0638	18
0.0638	0.0851	19
0.0851	0.1064	21
...
0.9574	0.9787	106
0.9787	1.0000	134

- *Module “GENERATES TOTAL TIME AND TYPE OF RESULT”*

The objective is to assign the type of crime for which the accused is blamed. The *Module GENERATES TOTAL TIME AND TYPE OF RESULT* is showed in Figure 3. Data is showed in Table 3, and is related with Table 1.

At this point there must be a distinction between two stages: the *total processing time*, which is the one that Defender Lawyer takes in the various procedures related to the case, and the *total time out*, which is the total time taken to reach a verdict.

The total processing time and total time out generator modules are classified into two categories: Those which represent crimes, not resolved on the same day in the courtroom because of the nature of their process (verdict time nonzero). This category includes 210 = Robbery with violence, 97 = Homicide and 187 = Parricide. Those who's processing can be solved the same day or on subsequent days. This category includes all the other crimes.

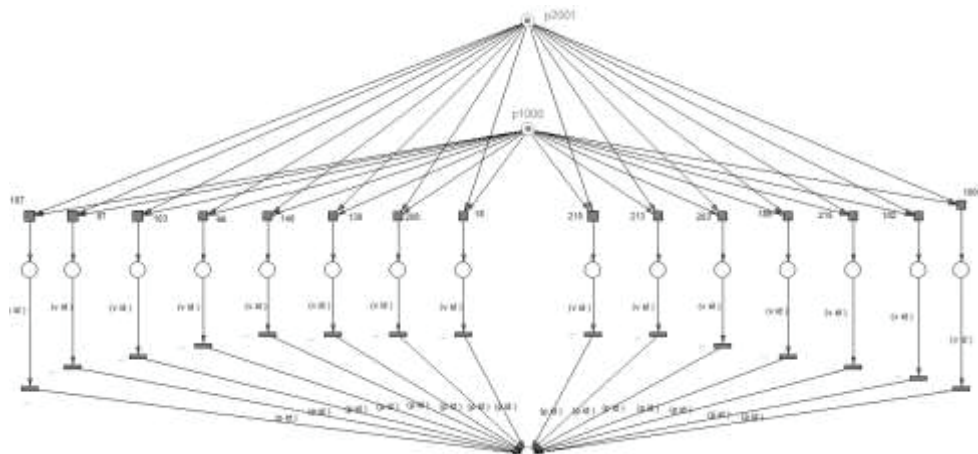


Fig.3. Module GENERATES TOTAL TIME AND TYPE OF RESULT

Table 3. Cumulative frequency for type of crime

Cumulative Frequency by Type [a , b [Type of Crime
0.000	0.535	103
0.535	0.588	66
0.588	0.629	140
0.629	0.658	139
0.658	0.685	208
0.685	0.709	18
0.709	0.728	213
0.728	0.746	203
0.746	0.764	189
0.764	0.780	215
0.780	0.796	102
0.796	0.808	210
0.808	0.809	97
0.809	0.809002	187
0.809002	1.000	1000

- *Category of crimes not resolved during the same day.*

It was decided to separate the processing time and the time out to more easily modify them in future studies. The data for these times (in days) are in Table 4.

At the time of the model performance there was not a formal study of processing times, whereby it was devised, for purposes of the simulation, a type time used in case management, established in man hours per week. For the crime 210 these times are exposed in Table 5.

The modules structure for the remaining set of crimes, that is, {103, 66, 140, 139, 208, 18, 213, 203, 189, 215, 102, 1000} is similar. The purpose is to determine how many processing days the case will be in order to add, on that day, the number of hours spent by the defender lawyer in the defense of his client.

Table 4. Processing Time and Time Out for Robbery with Violence (210), Homicide (97) and Parricide (187)

Type of Crime	Cumulative Frequency [a , b [Processing Time (in days)	Time Out (in days)
210	0.00	0.25	77	145
	0.25	0.50	90	75
	0.50	1.00	125	155
97	0.00	0.25	5	15
	0.25	0.50	18	175
	0.50	1.00	23	15
187	0.00	1.00	96	265

Table 5. Weekly Man Hours for Crime 210

Processing Days	Man-Hours Weekly
0	77
78	90
91	125

- *Category of crimes which processing can be solved on the same day or on subsequent days.*

This category is more difficult to model because a crime can be solved on the same day it is exposed in the courtroom, or may require further processing. Part of the complexity can be seen in Table 6, which shows the days of processing.

Table 6. Processing days for Simple Theft (103)

Cumulative Frequency [a , b [Processing Time (in days)	
		min	max
0.000	0.769	0	0
0.769	0.837	1	19
0.837	0.879	20	29
0.879	0.918	30	49
0.918	0.965	50	99
0.965	0.994	100	199
0.994	1.000	200	322

- *Module “ASSIGN LAWYERS TO CAUSES”*

This module specializes in assigning cases to Defender Lawyers. The issue to face in this module is: Given the above data, how to represent the allocation of defender lawyers to cases. Basically, in both the model with constraints and the one with no constraints, one can speak of two types of defender lawyers, and represent them in modules, independently of the number of defender lawyers that are in each case. Figure 4 shows this module.

For the model with cases constraints for the Tendered Defenders, the constraint is 470 cases per each tendered defender lawyer, while Local Defender Lawyers are with no constraints.

When the Defender Lawyer has handled 18 cases (that is the estimate for a defender lawyer by day), it changes to the next Defender Lawyer.

For the model with constraints, once a defender lawyer has completed 470 cases, this module ignores the defender lawyers whose share of tendered case has been fulfilled and immediately continues to assign the defender lawyer who still has the capacity to work, according to the agreement. When all of the tendered defender lawyers have met the agreement, the model ignores them and continues to work only with local ones.

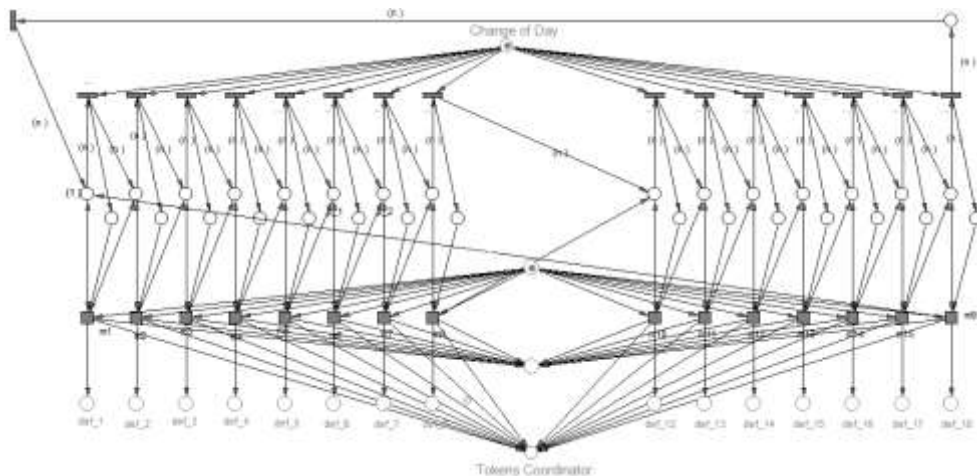


Fig.4. Module ASSIGN LAWYERS TO CAUSES

In the model with no constraints, since there are no a number of cases as an upper limit, the model works

with all of them. The reason for which defender lawyers do not carry a correlative number is that a model with 18 defender lawyers was reused and then three of them were removed until there were 15 left.

Defender Lawyers with no constraints: this module checks that the number of cases does not exceed 18 per day.

Defender Lawyers with Constraints: the difference is that a cases counter has been added for not exceeding the limit agreed by the tender contract, which is, monitoring the number of cases that can be daily processed by a defender lawyer, and also allows the passage of the vector.

3. Results

3.1. Scenarios Resulted under the Current Working Constraints

On average, the number of cases handled by the defender lawyers' staff is 780 versus 398 handled by a tendered defender lawyers (for any iteration). The proportion is showed in Figure 5(a). However, the amounts of time consumed by the defender lawyers staff (5.45 h/wk) and tendered defender lawyers (5.13 h/wk) are approximately equal. The proportion is showed in Figure 5(b).



Fig.5. Proportion of cases handled by defenders lawyers under current conditions

3.2. Without the constraints of number of cases (during the year), 18 defender lawyers and proportional increase in the number of cases.

Constraints are removed over the previous model on the number of cases referred to amount of cases which tendered defender lawyers could attend, that is, 470 cases per year. It also increased proportionally the number of cases in the corresponding period of year in 20%, compared to its equivalent constraint (currently tested with successive increases until reaching 20%, which is the limit defender lawyers can daily handle, given the workload they can handle).

On average, the number of cases handled by defender lawyers' staff is 525 versus 524 handled by a tendered defender lawyer (for any iteration). The proportion is showed in Figure 6(a). However, the amounts of time consumed by a defender lawyers' staff (4.26 h/wk) and tendered defender lawyers (4.67 h/wk) are approximately equal. The proportion is showed in Figure 5(b).

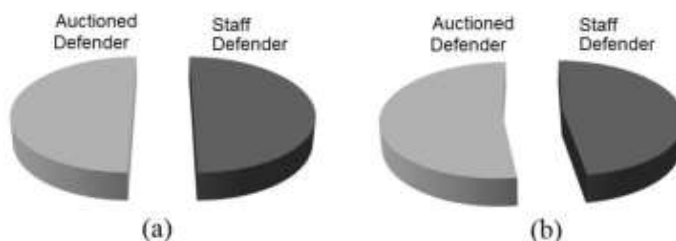


Fig.6. Proportion of cases handled without the constraints and 18 defender lawyers

4. Conclusions

Under the current tendered scheme performs DPP, in terms of limiting the caseload for tendered defender lawyers, generates inefficiency in various ways.

Although the number of hours worked daily by both types of defender lawyers is similar, the annual workload is notably higher for the defender lawyers' staff. This directly affects the quality of the work.

Changing work conditions in the sense of removing the limit of annual cases in the tender, increasing the number of defender lawyers and the workload, all defender lawyers are under equal work conditions.

One aspect not reported in this paper is that the simulation model allows investigating the behaviour of the system disaggregated into weeks.

Acknowledgements

This work was supported by Universidad Finis Terrae and Universidad de Las Américas.

References

- [1] Cifuentes F. *Modelo de Simulación para el manejo de Licitaciones de Defensores en la Defensoría Penal Pública de Chile, Utilizando Redes de Petri Estocásticas y temporizadas*, Eng. Thesis, Universidad de Santiago, Chile; 2009.
- [2] Raychaudhuri S., 2008. "Introduction to Montecarlo Simulation", Proceedings of the 2008 Winter Simulation Conference, S. J. Mason, R. R. Hill, L. Möhch, O. Rose, T. Jefferson, J. W. Fowler eds.
- [3] Meldman J. A Petri-Net Representation of a civil Procedure. *IDEA, The Journal of Law and Technology* 1977; **19**: 123–148.
- [4] Cordova F, Canete L, Quezada L, Yanine F. An Intelligent Supervising System for the Operation of an Underground Mine. *Int. Journal of Computers, Communications and Control* 2008; **3**:259-269.
- [5] Zhang W, Freiheit Th, Yang H. Dynamic scheduling in flexible assembly system based on timed Petri nets model, *Robotics and Computer-Integrated Manufacturing* 2005; **21**:550-558.
- [6] Ghaeli M, Bahri AB, Lee P, Gu T. Petri-net based formulation and algorithm for short-term scheduling of batch plants, *Computers & Chemical Engineering* 2005; **29**: 249-259.
- [7] Zadeh LA. et al. *From Natural Language to Soft Computing: New Paradigms in Artificial Intelligence*. Romania: Editing House of Romanian Academy; 2008.
- [9] Yuan Y, Shaw MJ. Induction of fuzzy decision trees. *Fuzzy Sets and systems* 1995; **69**:125-139.
- [10] Purvis M. *Dynamic Modeling of Legal Processes with Petri Nets*, Ph.D. Thesis, University of Otago, Dunedin, New Zealand; 1998.