Available online at www.sciencedirect.com
ScienceDirect
Data & Knowledge Engineering 63 (2007) 91–107
DATA & KNOWLEDGE ENGINEERING
www.elsevier.com/locate/datak

Strategies for improving the modeling and interpretability of Bayesian networks

Ádamo L. de Santana [a,*], Carlos R. Francês [a], Cláudio A. Rocha [b], Solon V. Carvalho [c], Nandamudi L. Vijaykumar [c], Liviane P. Rego [a], João C. Costa [a]

[a] *Laboratory of High Performance Networks Planning, Federal University of Pará, R. Augusto Côrrea, 01, 66075-110 Belém, PA, Brazil*
[b] *University of the Amazon, Av. Alcindo Cacela, 287, 66060-902 Belém, PA, Brazil*
[c] *Laboratory of Computing and Applied Mathematics, National Institute for Space Research, Av. dos Astronautas 1758, Jd. Granja, 12227-010 São José dos Campos, SP, Brazil*

Available online 13 November 2006

Abstract

One of the main factors for the knowledge discovery success is related to the comprehensibility of the patterns discovered by applying data mining techniques. Amongst which we can point out the Bayesian networks as one of the most prominent when considering the easiness of knowledge interpretation achieved. Bayesian networks, however, present limitations and disadvantages regarding their use and applicability. This paper presents an extension for the improvement of Bayesian networks, treating aspects such as performance, as well as interpretability and use of their results; incorporating genetic algorithms in the model, multivariate regression for structure learning and temporal aspects using Markov chains.

Keywords: Knowledge discovery; Markov chains; Bayesian networks; Multivariate regression

1. Introduction

Bayesian networks stand as one of the best computational intelligence techniques among the existing paradigms, particularly due to their exceptional analytical properties to represent domains. However, just like any other computational algorithm, they also present limitations and disadvantages, regarding their use as well as their applicability.

Among the ''restrictions'' presented by the Bayesian networks we can point out: the difficulty to correlate variables considering the time factor and difficulty to establish which is the optimum combination of states for given variables that would achieve a certain requirement (desirable state for a certain variable of the domain).

* Corresponding author. Tel.: +55 91 3201 7302.
E-mail addresses: adamo@ufpa.br (Á.L. de Santana), rfrances@ufpa.br (C.R. Francês), alex@cci.unama.br (C.A. Rocha), solon@lac.inpe.br (S.V. Carvalho), vijay@lac.inpe.br (N.L. Vijaykumar), liviane@ufpa.br (L.P. Rego), jweyl@ufpa.br (J.C. Costa).

0169-023X/$ - see front matter
doi:10.1016/j.datak.2006.10.005

This paper deals with the mentioned problems by: incorporating a stochastic model such as associating Markov chains to the Bayesian networks; and by combining genetic algorithms with the networks obtained from the data. It also proposes a new and optimized method for learning the Bayesian network graphical representation, in which the creation of the network and the correlation analysis of the variables are conducted through multivariate regressions.

The methods that will be presented here treat the optimization of the Bayesian networks as a whole, considering the performance as well as quality and improvement of the results obtained. This improvement of the results is conducted right from the initial step, the assembly of the network from the data. It is followed with a complement for manipulation and analysis by investigating an optimum combinatory search according to the existing attributes. Then finally the predictive analysis of the network's behavior throughout the time is obtained.

The paper is organized as follows: in Section 2, the basic concepts of KDD (Knowledge Discovery in Database), data mining and Bayesian networks are shown. In Section 3 some related works are presented. Section 4 presents a method for learning the Bayesian network structure using multivariate regression. In Section 5, the use of genetic algorithm is presented for the discovery of the optimum combination of values for the variables of a Bayesian network, as a mean to maximize a target variable. In Section 6 a temporal model using the concepts of Markov chains is presented. The final remarks of the paper are presented in Section 7.

2. KDD, data mining and Bayesian networks

The process of knowledge discovery in database (KDD) [9] stands as a technology capable of widely cooperating in the search of knowledge existing in the data. Therefore, its main objective is to find valid and potentially useful patterns from the data.

The extraction of knowledge from data can be seen as a process with, at least, the following steps: understanding of the application domain, selection and preparation of the data, data mining, evaluation of the extracted knowledge and consolidation and use of the extracted knowledge. Once in the data mining stage, considered the core of the KDD process, methods and algorithms are applied for the knowledge extraction from the database.

This stage involves the creation of appropriate models representing patterns and relations identified in the data. The results of these models, after evaluated by the analyst, specialist and/or final user, are used to predict the values of attributes defined by the final user based on new data [9].

In this work, the computational intelligence algorithm used for data mining was based on Bayesian networks.

A Bayesian network is composed of several nodes, where each node of the network represents a variable, that is, an attribute of the database; arcs connecting them and whose direction implies in the relation of dependency that the variable can possess over the others; and probability tables for each node.

The Bayesian networks can be seen as models that codify the probabilistic relationships between the variables that represent a given domain [32]. These models possess as components a qualitative, representation of the dependencies between the nodes, and a quantitative (conditional probability tables of these nodes) structure, evaluating, in probabilistic terms, these dependencies [3]. Together, these components provide an efficient representation of the joint probability distribution of the variables X of a given domain [24].

One of the major advantages of the Bayesian networks is their semantics, which facilitates, given the inherent causal representation of these networks, the understanding and the decision making process for the users of these models [3]. Basically, due to the fact that the relations between the variables of the domain can be visualized graphically, besides providing an inference mechanism that allows quantifying, in probabilistic terms, the effect of these relations.

A particular type of Bayesian network which we will also be using in this work (Section 5) is the Naive Bayesian network. In a Naive Bayes, it is assumed that all the attributes of the database are mutually independent, but being dependent of one certain father node (one of the attributes of the database, chosen as the main one, from which the remaining attributes present a certain dependency).

The Naive Bayes stands out among the many existing classification methods as one of the simplest and computationally more efficient; being also robust against noises in the data and irrelevant attributes [15], in such a way that they would not influence in the probabilities of the other attributes.

3. Background and related works

In this section, we will present some of the works presented in literature and that also served as basis as well as comparison for the studies presented in this paper. The works are also divided here according to the fundaments of their approaches: whether it is based on the graphical structure learning of the network; on the search for the best configuration, that is, the set of actions or inferences and furthermore its singular values to achieve a specific state; or the temporal analysis for Bayesian networks.

First of all, on the matter of *graphical learning*, the construction of a Bayesian network involves the learning of the network structure and the definition of the probabilities associated with its variables. This process can be done directly with the help of experts in the studied domain or automatically, with learning algorithms, which we will focus here. The learning algorithms can be classified as being *constraint based*, where the structure is obtained by identifying the dependencies among the variables; or through a *search and score* of the best network structure.

Here the *search and score* approach is used for the learning of the network topology. The *search and score* works searching through the space of possible existing structures, starting from a graph with no arcs and adding new ones, calculating a score for the given structure until no new arc can be added.

In [22] a search and score method to induce Bayesian networks is proposed, using both fuzzy systems and genetic algorithms. It is proposed a scoring metric based on the evaluation of different quality criteria, which is computed by the fuzzy system; using the genetic algorithm as means to search through the space of possible structures, which, as was also pointed, has already been applied to the learning of Bayesian networks [17].

The fuzzy system uses as input metrics the Bayesian measure, the minimum description length principle [30], Akaike information criteria [1], and the estimated classification accuracy of the network; thus providing the quality of the network as output. The genetic algorithm is used to search the possible network structures.

Comparatives as to the algorithm performance with well-known algorithms (BayesN [21], Bayes9 [28], Tetrad [36] and K2 [6]), which will also be presented as comparative in Section 4, are also presented.

The use of this approach brings however some limitations such as the fact of it being sensitive to the selection of the initial population (for the genetic algorithm) as well as for the different membership functions (for the fuzzy system).

Other recent methods implemented for the learning of the Bayesian graphical structure, usually based on hybrid models can also be seen in [35,19,20], each with its own metric of scoring and evaluation: use of (semantic) crossover and mutation operators to help the evolution process, penalty measure, and Minimum Description Length metric, respectively; [20] proposal however does not involve a need for a complete ordering of the variables as input. Further use of genetic algorithms can also be seen in [10,13].

In [25] the use of a previous ordering of the variables is also studied, proposing a multi-phase approach for the graphical learning based on the use of distinct but easy to implement algorithms, which involves a search method for optimal parents to build the structure, followed by a method to eliminate existing cycles in the graph and finally, an evaluation of the network using structural perturbation.

Aside from the ordering of nodes, the dataset (here we will treat only with *fully observed* cases) size is also an important aspect when considering the network quality and convergence speed of the algorithm. Especially since it is NP-hard [5], exponentially increasing the searching space with the number of variables.

A more thorough overview on the techniques and algorithms for the learning of Bayesian networks can be seen in [4].

With regards to the *optimal configuration search*, we will point here the approach of using computational intelligence algorithms as means to optimize and improve the knowledge discovery process, resulting, in many cases, on hybrid systems. For instance, [37] uses fuzzy systems as a way to improve the Bayesian inference.

In [34] neural networks are treated to improve the backpropagation training algorithm, which, although being one of the most popular training techniques, does not allow discriminating and identifying relevant input variables in the model. To identify these variables would be however an important asset of information for the researchers. Thus, genetic algorithms are implemented together with the neural network as training alternative to extract this knowledge and overcome this deficiency of the algorithm.

In this paper, it will also be presented a hybrid optimization model, here using genetic algorithms to seek and provide further extraction of knowledge and information from Bayesian networks. An application

example on the domain of power systems is given. Genetic algorithm or evolutionary computing, as discussed previously, is one of the most prominently used techniques when it comes to optimization (see [7] for a specific use of genetic algorithms on power systems).

Finally, when it comes to *temporal analysis*, forecasting is accomplished in most works in literature by means of time series analysis [12]. However, techniques such as dynamic Bayesian networks [23], hidden Markov Models [27] or Kalman filters [16] are better suited when it is desired to consider the dependences among the variables, adding also a probabilistic reasoning. In Section 6, a distinct model for temporal analysis of Bayesian networks is proposed for an easy modeling and inference of the observed variables.

4. Structure learning based on multiple regressions

The algorithm, implemented to induce the learning of the structures, searches for the best configuration, amongst the space of possible structures, for the construction of a Bayesian network from the analysis of existing dependences and independences between the variables. The algorithm uses the search and score method, analyzing all the possible graphical combinations that can be set from the variables of the domain.

The analysis for the search of the best Bayesian network that represents the domain is made through a multivariate regression [8,11,12,26,29] on the data. The heuristic of the search follows from the ordinance of the variables, where for each attribute X_i the possible dependencies of the variable with its precedents are examined (variables parents – Pa_i), adding arcs between them and verifying the quality of the network created according to its score; continuing, as follows, with the search of another attribute, that added to the previous one(s) would increase the score of the network.

The validation of the network created by each new added arc made through regressions, that can be single (when analyzing the relation with only one variable) or with multiple variables, as it is usually used.

This algorithm for learning of structures through multivariate regressions (Multiple Regression Structure Learner – MRSL) attributes the score of each network through the value found by the adjusted coefficient of each regression (R^2); which is obtained as described next.

Assuming we are working with a database D with n records and i number of variables, we are searching for the best Bayesian network structure B_S for it. We denote the target variable that we are analyzing as Y, and the k variables candidates for parents as X. The generalized formula of the regression is

$$Y = A_0 + A_1X_1 + A_2X_2 + \cdots + A_kX_k + E \tag{1}$$

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} \tag{2}$$

$$X = \begin{bmatrix} 1 & X_{11} & X_{12} & \cdots & X_{1k} \\ 1 & X_{21} & X_{21} & \cdots & X_{21} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & X_{n1} & X_{n2} & \cdots & X_{nk} \end{bmatrix} \tag{3}$$

Expanding for each instance of the database in Y and X as above we have

$$\begin{aligned} Y_1 &= A_0 + A_1X_{11} + A_2X_{12} + \cdots + A_kX_{1k} + E_1 \\ Y_2 &= A_0 + A_1X_{21} + A_2X_{22} + \cdots + A_kX_{2k} + E_2 \\ \vdots \quad &\quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \\ Y_n &= A_0 + A_1X_{n1} + A_2X_{n2} + \cdots + A_kX_{nk} + E_n \end{aligned} \tag{4}$$

This way, in order to calculate the regression we need to obtain the values of the coefficients:

$$A = [A_1 \quad A_2 \quad \cdots \quad A_n]^{\mathrm{T}} \tag{5}$$

which can be calculated according to

$$A = (X^t X)^{-1} \times X^t Y \tag{6}$$

With the values of A, we can then calculate the value of the regression coefficient (R^2) according to

$$R^2 = \frac{A^t(X^t X)A - n\overline{Y}^2}{y^t y} \tag{7}$$

where $\overline{Y}$ is the mean value of variable Y, and y is obtained by the subtraction of Y by $\overline{Y}$. And thus calculating its adjusted value by

$$\overline{R}^2 = 1 - (1 - R^2)\left(\frac{n-1}{n-k}\right) \tag{8}$$

In the same way, the absence of dependencies ($Pa_i = \phi$) for the variable in question is assigned when obtained, for each possible relation of dependence, a value close to or below zero for $\overline{R}^2$.

Another important aspect of the proposed model is regarding the relevance of the inclusion of new variables in the dependencies model of the attribute. This analysis is made in order to verify whether or not the inclusion of one or more arcs for the variable is indeed relevant for the model, even though with this inclusion a higher $\overline{R}^2$ is obtained. The analysis of this aspect is due, in particular, to the fact of being verified during the evaluation tests of the algorithm, that the $\overline{R}^2$ obtained for the best Pa_i configuration for the variable X_i with a number of arcs x was very close to the one achieved by the best Pa_i configuration with a number of arcs $x + 1$. The same behavior was also observed when comparing the latter with the value obtained with a number of arcs $x + 2$, and so on.

In order to provide the analysis relevance to be generally applicable for datasets disregarding their sizes, the F test, whose formula is presented below, was used:

$$F = \frac{(R_{\mathrm{I}}^2 - R_{\mathrm{R}}^2)/m}{(1 - R_{\mathrm{I}}^2)/(n-k)} \tag{9}$$

where R_{U}^2 and R_{R}^2 are the $\overline{R}^2$ values obtained for the unrestricted (with the inclusion of the new variables) and restricted (without the inclusion of the variables) regressions, respectively, and m is the number of variables added to the model.

In order to exemplify the functioning of the algorithm, let us consider the analysis of the following model: a database D which, for simplification purposes, is composed of 10 records and 4 attributes (Table 1), being each one of them binary (number of possible states is 2). In fact, the number of states that each variable can assume is irrelevant for the functioning of this algorithm; it does not influence in increasing the number of states, as it will be proven later, in the performance of the algorithm. Besides, the algorithm allows dealing with continuous variables as a whole. We also point out that, when dealing with non-numerical discrete variables, the variables have their r states coded to integer values from 1 to r, and then onwards they are treated in the same way by the algorithm.

The algorithm initiates the structure search with a network where all the attributes are mutually independent, that is, there are no arcs connecting them. Given that the first attribute X_1, by definition, does not pos-

Table 1
Database example D

X_1	X_2	X_3	X_4
2	1	1	1
2	2	1	2
1	1	1	2
2	2	1	2
1	1	1	1
1	2	2	2
2	2	2	2
1	1	2	1
2	2	2	2
1	1	2	1

sess parents [14], the search follows immediately to the next node. The node X_2 can then only possess at the most one possible father (X_1). Eq. (8) is then used to verify whether the addition of the arc connecting the two variables is relevant or not (i.e. the $\overline{R}^2$ obtained presents a value close to or below zero) for the structure of the network. Here the variable X_2 becomes Y (2) and only one regression is made, in this case for the node X_1, obtaining an $\overline{R}^2$ of 0.28. According to statistical estimates, an $\overline{R}^2$ value as the one obtained is not significantly enough to indicate the representation of a domain, however, as what it is primordially searched here is the existence of the dependence relations among the variables of the domain (as direct or indirect), such values are assumed here. This way, only the values that, as described previously, are smaller or very close to zero are not accepted.

Continuing the search, we move to X_3, initially verifying the possible connections with only one individual arc, that is, with X_1 and then with X_2, obtaining $\overline{R}^2 = -0.08$ on both. As the results are negative values, no arc is created at this moment. Then the regression is conducted considering as parents both X_1 and X_2, which resulted in a $\overline{R}^2$ of -0.02857. Once those values for the correlations between the node X_3 and the other variables had resulted only in negative values, no relation of dependency is assumed (number of parents is zero) for X_3, and, thus, no arc is directed to it.

The same process follows for X_4, initially testing its connectivity with only one arc, obtaining as results of the regressions for, X_1, X_2 and X_3 the values 0.0625, 0.625 and -0.125, respectively; resulting in the creation of an arc between X_2 and X_4. Considering the existing combinations of size two the following results can be obtained: $\overline{R}^2_{X_1,X_2} = 0.584821$, $\overline{R}^2_{X_1,X_3} = -0.0625$ and $\overline{R}^2_{X_2,X_3} = 0.607143$. As the higher $\overline{R}^2$ found for X_3 considering the existence of two parents – X_2 and X_3 – is smaller than the value considering only one ($\overline{R}^2_{X_2} > \overline{R}^2_{X_2,X_3}$), only the arc connecting X_2 is kept, not adding a new one. Finally, the regression considering all the preceding attributes of X_4 is made, resulting in a $\overline{R}^2$ of 0.59375; thus again, as $\overline{R}^2_{X_2} > \overline{R}^2_{X_1,X_2,X_3}$, no arc is added.

The resulting Bayesian network obtained by the MRSL for the base D is presented in Fig. 1.

The MRSL algorithm acts in an optimized way, with respect to performance, when compared with other existing learning algorithms in the literature. It works directly without considering the number of states of the variables, not suffering from any combinatory impact that can be implied by them in the search and score of the best network structure.

However, in order to optimize even further the performance of the algorithm some considerations and heuristics can also be adopted. In the very first iterations of each variable, a control can be included in order to decrease the combinatory space to be covered and, consequentially its execution time.

Firstly, from the values obtained in the correlations of degree one (number of parents equals to one) of the variable X_i with its precedents ($\overline{R}^2_{X_1}, \ldots, \overline{R}^2_{X_{i-1}}$), it is already possible to observe which, amongst the variables, presents a higher level of correlation with X_i. This is extremely important as, whenever a new arc can be added in the network structure, the new combination of parents found will have as component, compulsorily, the attribute (or combination of attributes, if the new number of arcs is more than 2) found previously. Thus, only the calculations of the future regressions for the combinations that also have as component the nodes whose arcs were already assigned will be made.

Not only that, but if in the correlations of unitary degree the coefficients $\overline{R}^2$ present values close to or below zero, the search for a better configuration can already be finished, as the following ones will also obey the same trend, characterizing an absence of parents for the attribute in question.

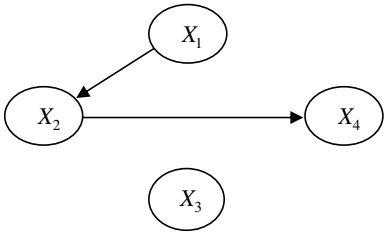

Fig. 1. Bayesian network generated by the MRSL for the base D.

In an analogous manner, there will be no need to continue the search for the admission of new arcs for each attribute when verified that the significance of the admission of a new arc is not relevant over the previous model, given that the posterior structures will also, usually, not present a higher significance.

Another aspect that can be manipulated, together with the previous, is the indication by the user specialist in the domain of a minimum degree of significance to be verified for the admission of a new arc in the structure.

The evaluation of the proposed model was made considering both the quality of the Bayesian network found by the algorithm as well as its computational performance.

For comparing the analysis regarding the quality of the generated network, the *Chest Clinic* [18] database was used as application example (usually known as Asia), which denotes a problem of a fictitious medical diagnosis, of whether a patient has tuberculosis, lung cancer or bronchitis, based on his X-ray, dyspnea, visit to Asia and smoking status. The database possesses eight binary variables and its Bayesian network presents eight arcs connecting them (Fig. 2).

Table 2 compares the result achieved by our algorithm (MRSL) with the one from the original Bayesian network of the Asia database, as well as the results obtained by others five existing algorithms in the literature: K2 [6], Tetrad [36], Bayes9 [28], BayesN [21] and Genetic-Fuzzy [22]. The *Total* column presents the number of arcs found by each algorithms; the column *Correct* contains the number of arcs that were correctly found; the column *Additional* presents the number of arcs that were found and that are not in the original network; and the column *Absent* presents the number of arcs that were not found and are present in the original network.

For the performance evaluation of the algorithm, the analysis was made using as testbed experiment the model presented by the Asia network, which is composed of eight variables and 1000 records, comparing the results obtained with the ones presented by the K2 algorithm.

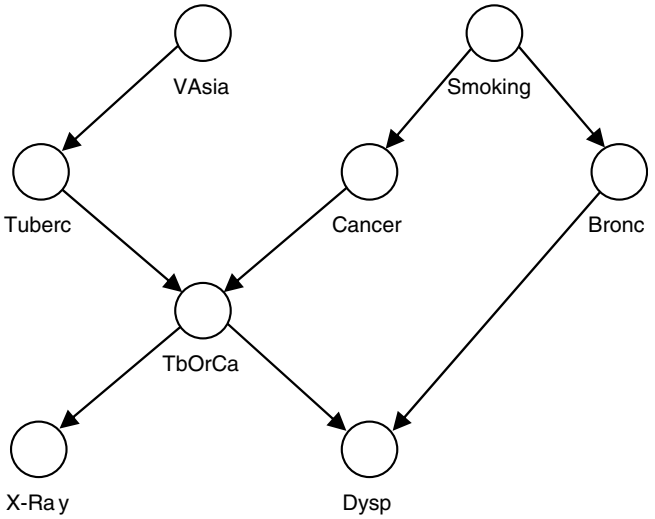

Fig. 2. Bayesian network of the database Asia.

Table 2
Comparative of the results obtained for the Asia database

Algorithms	Total	Correct	Additional	Absent
MRSL	8	8	0	0
Genetic-Fuzzy	9	8	1	0
K2	8	7	1	1
BayesN	8	5	3	3
Bayes9	4	4	0	4
Tetrad	4	4	0	4

The test made here seeks to verify the performance of the algorithm using as parameter the discretized states of the database variables, that is, the number of possible states that each attribute can assume. The performance tests were made analyzing the execution time for both algorithms over the database, with its attributes, initially binary, discretized from the two initial states until a maximum of ten. The obtained results (Table 3) denote the execution times of the algorithms without considering the time spent for reading the database into the memory.

Tables 4 and 5 present the same tests, now also considering an increase in the number of records of the database to 5000 and 10,000, respectively. Table 6 presents the values, considering only the discretization space of 10, for a better visualization of the gradual behavior in the increase of the execution time between the algorithms.

Table 3
Execution times (s) obtained by the algorithms

Number of states	MRSL	K2
2	0.08	0.1
3	0.08	0.14
4	0.08	0.24
5	0.08	0.51
6	0.08	1.35
7	0.08	3.51
8	0.08	9.1
9	0.08	20.05
10	0.08	44.48

Table 4
Execution times (s) obtained by the algorithms for 5000 records

Number of states	MRSL	K2
2	0.42	0.48
3	0.42	0.68
4	0.42	0.93
5	0.42	1.32
6	0.42	2.26
7	0.42	4.53
8	0.42	10.27
9	0.42	21.32
10	0.42	45.42

Table 5
Execution times (s) obtained by the algorithms for 10,000 records

Number of states	MRSL	K2
2	0.84	0.96
3	0.84	1.33
4	0.84	1.78
5	0.84	2.32
6	0.84	3.39
7	0.84	5.81
8	0.84	11.74
9	0.84	22.87
10	0.84	47.05

Table 6
Execution times (s) obtained with a number of states of 10

Number of records	MRSL	K2
1000	0.08	44.48
5000	0.42	45.42
10,000	0.84	47.05

As it could be verified by the obtained results, the structure learning algorithm based on multiple regressions outperforms on both aspects analyzed: with respect to the quality of the Bayesian network induced by the algorithm as well as in its computational performance. The algorithm uses in its structure statistical models with a fundamental theory, especially concerning the prediction and correlation analysis of the variables; and that, due to its nature, works in an optimized way, improving the performance as the number of states assumed for the variables increases; which is a common characteristic for databases that represent real world domains.

5. Maximization model

The objective of this model is to identify the best configuration, among the possible values of the existing variables in the domain, which maximizes a given attribute, identifying initially the other variables that present a dependency from it. It is also worth mentioning that, it is possible to accomplish the same process using not the maximum value, but any other value that may be relevant.

In contrast to the way the genetic algorithms are used in the majority of the hybrid systems proposed in literature, where they are adopted to optimize the process of learning the structure of Bayesian networks, as discussed in Section 3. Here, this technique is used for the discovery of the most probable values of the variables of a Bayesian network, given the value of a key attribute.

The analysis described here were originated from the demands of the research project "PREDICT – Support Decision Tool for Load Prediction of Electrical Systems", financed by the "National Agency of Electric Energy of Brazil – ANEEL" in course since September 2004 [31,33]. This project, together with the Government of the state of Pará and the Power Supplier of the State of Pará, aims at designing and implementing a decision support system, using mathematical and computational intelligence methods, to foresee the demand for energy purchase in the future market and to make inferences on the power system situation from the consumption historical data and its correlations with socio-economic and climatic data.

The case study example, proposed by the domain specialists of the power system market, and used for the optimization model was to discover under which circumstances the power consumption would be maximized. For such, the optimization model was based on a few steps.

Firstly, identify which attributes, among those from the database, influences directly in the power consumption; also this way building the Bayesian network structure.

As a means to particularize each section of the paper, the search and score algorithm K2 [6] will be used here as the learning algorithm instead of the MRSL, searching for the most probable belief network structure B_S given a data set D. The K2 algorithm applies a Bayesian scoring method, according to

$$P(B_S|D) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{\Gamma(r_i)}{\Gamma(r_i + N_{ij})} \prod_{k=1}^{r_i} \Gamma(N_{ijk} + 1) \tag{10}$$

where

n number of nodes
q_i number of configurations of the parents of the variable X_i
r_i number of possible values of X_i
N_{ijk} number of cases in D where X_i is evidenced with its value k, and the configuration of the parents of X_i is evidenced with the value j
N_{ij} number of observations in which the configuration of the parents of X_i is in evidence with the value j, being $N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$

Analyzing the Bayesian network, it was verified, and confirmed by the domain experts, the correlation of the power consumption with some of the remaining variables, especially with the following: the number of employments in the sectors of the transformation industries and agriculture and cattle breeding, and the values of the total turnover and of the dollar; which were shown to be more representative than the others.

Given the knowledge that the variables of number of employments in the transformation industries (*emp_ind*), employments in the agriculture and cattle breeding (*emp_agro*), value of the total turnover (*val_turn*) and the value of the dollar (*val_dol*) are the main influencers in the variation of the power consumption, they were used for the next step, which consisted in the creation of a Naive Bayesian network (Fig. 3), in which all of the remaining attributes (*emp_ind*, *emp_agro*, *val_turn* and *val_dol*) were dependent of only one, in our case, the power consumption.

All the attributes were discretized in ten states, according to the frequency of their values, allowing us to verify the probability associated to each one of them, as well as the conditional probabilities existing between the variables.

Once the network is set, the next step is, by making use of the data given by the Bayesian network, search the network attributes for the states that would maximize the power consumption. In this step we use a modified genetic algorithm.

Here, instead of a cost function to validate the individuals of the population, a Bayesian inference algorithm is implemented (Eq. (11)); each of the individuals of the genetic algorithm represents an inference configuration of the Bayesian network generated randomly (e.g. evidencing the variables *emp_ind* with the state 7, *emp_agro* with state 1, *val_turn* with 7 and *val_dol* with 4 generates the individual 2-1-7-4). Each individual is then, for its classification, submitted to the Bayesian inference module in order to verify the probability in which the power consumption attribute would be maximized, obtaining, at the end of the iterations, the best possible configuration of inferences on the Bayesian network for the maximization of the power consumption:

$$P(x_i|c_1, c_2, \ldots, c_n) = P(x_i) \prod_{k=1}^{n} P(c_k|x_i) \tag{11}$$

where $c_1, c_2, \ldots, c_n$ are possible evidenced events and x_i is the event we want to observe.

However, we would have at the end of this step (after the genetic algorithm analysis) only the respective states (i.e. band of values) for this maximization, instead of a single value (for each attribute), which is what we seek. For such, we make use, again, of a genetic algorithm; but this time a traditional genetic algorithm, whose aptitude function we obtain from the data.

The function used for the genetic algorithm is obtained from a regression of multiple variables made over the attributes of the network. The multivariate analysis is however made over the consumption data, but considering only the data instances located within the ranges found in the previous step. Thus, the obtained equation (presented below) presented a good representativity ($\overline{R}^2$of approximately 0.9039) over the domain:

$$Y = 258,598,510.5 + 3,675.6834 \times X_1 + 4,430.9036 \times X_2 + 0.4701 \times X_3 - 12,182,208.61 \times X_4 \tag{12}$$

where Y represents the power consumption and X_1, X_2, X_3 and X_4 represent the values of the attributes *emp_ind*, *emp_agro*, *val_turn* and *val_dol*, respectively.

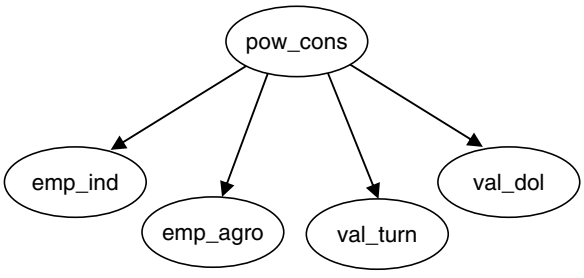

Fig. 3. Naive Bayesian network.

Table 7
Values of the attributes for the maximization of the consumption value

Attribute	Value
emp_ind	5380
emp_agro	3357
val_turn	100,752,576.00
val_dol	2.861

Based on function (12), the genetic algorithm is then used, thus obtaining the values, for each of the attributes that would maximize the power consumption. Mentioning again that the individuals evaluated by the aptitude function (12) are only those within the range of values that maximize the value of consumption. Thus, in order to achieve the occurrence of the maximum consumption, equivalent the 305,760,544 MW h, it is necessary that the values in Table 7 are observed, for the attributes *emp_ind, emp_agro*, *val_turn* and *val_dol*.

It is worth mentioning that the optimization model used is restricted not only to the discovery of the maximum values of consumption, but can also be used to identify the scenarios that cause a minimum, average or any other value to be reached by the power supplier, given the variation of the considered economic aspects.

Among the main results obtained in the "Predict" Project with the use of this model, it is possible to highlight: the extension of the interpretability of the generated Bayesian networks to measure the causal relationship for the consumption and the socio-economic variables, from the discovery of the values that compose an optimum combination given a certain target, for example, the consumption; and the interest of those involved in the Project in using the functionalities of the model for many other scenarios, not only relative to the power consumption, but also to government actions (e.g. discovered of the variables, that would maximize the employment and income), has encouraged and certified the use of the proposed model.

6. Markovian models incorporation with Bayesian networks

Despite allowing the verification of the future behavior of its attributes through inferences, the Bayesian networks do not present means that would allow us to discover how close or distant these events would be from occurring. In other words, they do not allow us to quantify and point out the time it would take for the impact of these inferences to occur. As a reference we point out that, in order to extract these time properties from the Bayesian network and introduce them into a Markovian process we need to be working with a time series study, being thus working with a given time scale.

A classical initial problem when working with the Bayesian networks in the time would be the existing necessity to setup conditional probability tables for each discrete unit of time analyzed. Thus, it is assumed, as well as described in literature, that the focus is to use a stationary random process.

In this work, the time analysis from the modeling of the data and characteristics proceeding from a Bayesian network into a Markov chain is presented. The idea is to establish an isomorphism between a Bayesian network in time and a discrete time Markov chain.

The model used seeks to analyze the forecast, differently as it would be if a dynamic Bayesian network would be used or even a hidden Markov model (HMM); it can be however consider as making use of the concepts of HMM, with respect to its theoretical foundations and assumptions regarding non-regular Markov models and being governed by probability distributions.

The proposal intends then on modeling in a simplified way the Markovian time transition according to a first-order process, but also intrinsically considering, in its transitions, the other variables of the domain that might also influence in the behavior of this attribute. That is, just as a Markov chain, a Bayesian network can be seen as a matrix of attributes that are correlated and that also has an influence over each other throughout time.

To exemplify the model, a simple example of a Bayesian network can be considered (Fig. 4), composed of only two variables: *Grade* and *Study*; where the grade obtained on a given test depends on the amount of study applied. It is also assumed that the tests are taken on a monthly time scale.

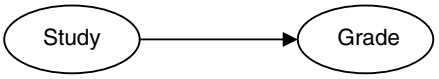

Fig. 4. Bayesian network mounted with the variables *Grade* and *Study*.

It is considered as possible values for the attributes the following: Study (Hard, Medium, Little); and Grade (Excellent, Good, Regular).

In this sense, the Bayesian network would also present the values of initial and conditional (for *Grade* only, given that it is the only attribute that possesses a parent attribute, that is, a dependence relation of the *Grade* given the *Study*) probabilities.

The dependency model and the probability tables would represent all the data the Bayesian network could offer us. Following the Markovian modeling, what we are seeking to obtain is the time instant n that, given an inference, a determined probability configuration of an attribute would happen (e.g. considering our example, given that we study *Hard*, when we would obtain a grade *Excellent* with probability of 70%, *Good* with 25% and *Regular* with 5%).

Given that what we seek is in fact the new configuration of a determined attribute, what we end up needing is to setup the Markovian transition matrix of this attribute. This is done by mapping the transition probabilities for the states of the attribute onto the matrix, based on the conditional probabilities that it possesses given its dependencies with the other attributes (e.g. also considering the example, we must map the transition probabilities of *Grade* for: *Excellent* and pass to *Good*, *Excellent* to *Regular*, *Excellent* and achieving *Excellent* again, etc.). That is, we would have to compute the transition probabilities for the states of a given variable, which Markovianly speaking we can anagously see as the transition probability to achieve an state N_{t+1} based on N_t. Hence we seek to find $P(N_{t+1} = s_y | N_t = s_x) = p_{xy}$; thus creating a Markov transition matrix, according to the model in Table 8.

However, considering only the factor of study in relation to the grade is not enough to verify the relation of the variable *Grade* with itself and to make the transition between its states, as the Markov transition matrix would immediately converge to the stationary state. So, we must also consider the value of the attribute *Grade* at a previous point of time, acting together with the variable *Study* and thus obtaining the transition relations for the variable *Grade*.

For such, the first record in the existing historical database is ignored so that we can insert in the analysis, analogously to a first-order Markovian process, the *Previous Grade* obtained. Tables 9 and 10 present the marginal and conditional (Study, Grade and the Grade in the previous period) probabilities of the *Current Grade* considering the *Study* and the *Previous Grade* (Grade-1).

Table 8
Model of the Markov transition matrix to be mounted

Grade\Grade	*Excellent*	*Good*	*Regular*
Excellent	p_{EE}	p_{EG}	p_{ER}
Good	p_{GE}	p_{GG}	p_{GR}
Regular	p_{RE}	p_{RG}	p_{RR}

Table 9
Initial probabilities of the Bayesian network

Study		Grade	Grade	Grade-1
Hard (Ha)	0.133	Excellent (E)	0.210	0.333
Medium (Me)	0.534	Good (G)	0.467	0.333
Little (Li)	0.333	Regular (R)	0.323	0.333

Table 10
Conditional probabilities of the Bayesian network – P(Grade|Study ∩ Grade-1)

Study ∩ G-1\Grade	*E*	*G*	*R*
Ha ∩ E	0.934	0.033	0.033
Ha ∩ G	0.333	0.333	0.333
Ha ∩ R	0.333	0.333	0.333
Me ∩ E	0.491	0.491	0.018
Me ∩ G	0.033	0.934	0.033
Me ∩ R	0.018	0.491	0.491
Li ∩ E	0.333	0.333	0.333
Li ∩ G	0.018	0.491	0.491
Li ∩ R	0.033	0.033	0.934

The calculations for the Markov transition matrix would follow:

$$p_{\mathrm{EG}} = P(\mathrm{E}) \times [P(\mathrm{G}|\mathrm{Ha} \cap \mathrm{E})P(\mathrm{Ha}) + P(\mathrm{G}|\mathrm{Me} \cap \mathrm{E})P(\mathrm{Me}) + P(\mathrm{G}|\mathrm{Li} \cap \mathrm{E})P(\mathrm{Li})] \tag{13}$$

Generalizing we would have

$$p_{xy} = \frac{\sum_{i=1}^{n} P(s_y|s_x \cap Pa_i) \times P(Pa_i)}{\sum_{j=1}^{m}\sum_{k=1}^{n} P(s_j|s_x \cap Pa_k) \times P(Pa_k)} \tag{14}$$

where

s	observed variable and its respective states
Pa	variable that represents the attributes on which variable *s* is dependent
n	number of possible states and/or combinations that the parents of this attribute can assume
m	number of states the attribute can assume

Calculating from (14), we obtained the Markov transition matrix presented below (Table 11).

The matrix obtained presents the transition probability values for the states of a given variable analyzed. If we apply a solution of the chain to find the probability vector at a given time *n*, we will then have to calculate the *n*th power of the random probability matrix. As described by the equations of Chapman–Kolmogorov [2].

In matrix notation, the expression is

$$P^{(n)} = P^{(m)} \times P^{(m-n)} \tag{15}$$

where $P^{(n)}$ is the transition matrix in the step *n*. From (15) it can be concluded, therefore, that

$$P^{(n)} = P^{n} \tag{16}$$

demonstrating that the matrix in step *n* corresponds to the *n*th power of this matrix. Thus, for example, if the unit of time is discretized in months and if we wanted to obtain the probabilities for the grades occurrence three months from now, we would have to find the power P^3 of the matrix (Table 12).

The analysis and results presented (Tables 11 and 12), considered the behavior of the domain, given the available data, in time without any inference being made; when considering this aspect, in order to make the analysis in time given the evidence of a determined state of a variable – as example, considering as fact that the level of *Study* applied to make the test was *Medium* – we would have (Table 13).

Thus, considering the inference made, we would have in a step $n = 3$ the following matrix (Table 14).

Table 11
Markov transition matrix obtained

Grade\Grade	*Excellent*	*Good*	*Regular*
Excellent	0.497	0.378	0.125
Good	0.068	0.707	0.225
Regular	0.065	0.318	0.618

Table 12
States transition matrix in the step $n = 3$

Grade\Grade	*Excellent*	*Good*	*Regular*
Excellent	0.1878	0.5274	0.2851
Good	0.1085	0.5561	0.3359
Regular	0.1071	0.4976	0.3974

Table 13
Transition matrix considering the inference made – study: medium

Grade\Grade	*Excellent*	*Good*	*Regular*
Excellent	0.491	0.491	0.018
Good	0.033	0.934	0.033
Regular	0.018	0.491	0.491

Table 14
Transition matrix in the step $n = 3$ considering the inference made – study: medium

Grade\Grade	*Excellent*	*Good*	*Regular*
Excellent	0.150	0.805	0.045
Good	0.054	0.892	0.054
Regular	0.045	0.805	0.150

Finally, in order to go back from the Markovian transition matrix to the probability table of the variable we use the following:

$$P(s_x) = \frac{\sum_{i=1}^{n} p_{ix}}{\sum_{j=1}^{n}\sum_{k=1}^{n} p_{kj}} \tag{17}$$

where P is the probability for a given state of the observed variable s; n is the number of possible states that s can assume; and p represents the transition probabilities among the n states of variable s. Thus finding the probabilities for each state of the attribute *Grade* in a time period $n = 3$ given the inference of *Medium Study* applied. The probabilities for the attribute *Grade* considering the example given here are as follow: *Excellent* 0.083, *Good* 0.834 and *Regular* 0.083.

7. Final remarks

The possibility to represent graphically the structure of the patterns obtained from the data, as well as the exploratory character of the analysis allowed by the Bayesian networks, enables to indicate more deeply the relationship between the variables of a domain, favoring the increase of the comprehensibility of the discovered patterns, as well as the identification of the usefulness and relevance of these patters.

In this paper, three techniques to optimize the functioning of the Bayesian networks were presented, seeking, amongst other things, the improvement of its interpretability. The implemented models act in three stages in respect to the Bayesian networks for a more complete and extended use of its resources.

A new technique was initially presented for the modeling of the graphical structure of a Bayesian network using multiple regressions as method for the correlation analysis of the attributes. A hybrid model was also studied. This was based on the association of the interpretation given by the Bayesian networks with genetic algorithms, in order to obtain, given the value of a parameter-target, the Bayesian combination necessary to achieve it. Added to that, a Markovian approach to represent correlations in time is also proposed; it introduces innumerable advantages, amongst which we can point out that, the Markovian models possess relatively

simple solutions compared to its computational effort and to the mathematical complexity involved, which stimulates and facilitates its use.

With these strategies it is possible to extend the interpretability of the Bayesian networks and adjust them even further for applications of the real world, providing the decision support systems with innumerable other possibilities of interpretation and inferences.

As future works, the aspects for the implementation of a wider optimization and applicability will be more deeply studied, as well as other aspects and problems that are present in the search for the optimum structure of a Bayesian network, amongst which is the task of learning the structure without the necessity of a previous ordinance of the variables.

References

[1] H. Akaike, A new look at the statistical model identification, IEEE Transactions on Automatic Control 19 (6) (1974) 716–723.
[2] G. Bolch, S. Greiner, H. de Meer, K.S. Trivedi, Queuing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications, John Wiley & Sons Inc., New York, USA, 1998.
[3] Z. Chen, Data Mining and Uncertain Reasoning – An Integrated Approach, John Wiley Professional, 2001.
[4] J. Cheng, D. Bell, W. Liu, Learning Bayesian networks from data: an efficient approach based on information theory, Technical Report, Department of Computer Science, University of Alberta, 1998.
[5] D.M. Chickering, D. Heckerman, Christopher Meek, Large-sample learning of Bayesian networks is NP-hard, Journal of Machine Learning Research 5 (2004) 1287–1330.
[6] G. Cooper, E. Herskovitz, A Bayesian method for the induction of probabilistic networks from data, Machine Learning 9 (1992) 309–347.
[7] K.P. Dahal, C.J. Aldridge, S.J. Galloway, Evolutionary hybrid approaches for generation scheduling in power systems, European Journal of Operational Research, in press.
[8] W.R. Dillon, M. Goldstein, Multivariate Analysis – Methods and Applications, John Wiley & Sons, 1984.
[9] U. Fayyad, G. Piatesky-Shapiro, P. Smyth, R. Uthurusamy, Advances in Knowledge Discovery and Data Mining, AAAI Press, 1996.
[10] J.A. Gamez, L.M. de Campos, S. Moral, Partial abductive inference in Bayesian belief networks – an evolutionary computation approach by using problem-specific genetic operators, IEEE Transactions on Evolutionary Computation 6 (2) (2002) 105–131.
[11] J.F. Hair Jr., R.E. Anderson, R.L. Tatham, W.C. Black, Multivariate Data Analysis, Prentice-Hall, 1998.
[12] J.D. Hamilton, Time Series Analysis, Princeton University Press, 1994.
[13] H. Handa, O. Katai, Estimation of Bayesian network algorithm with GA searching for better network structure, in: Proceedings of the 2003 International Conference on Neural Networks and Signal Processing, 1, 2003, pp. 436–439.
[14] E. Herskovits, Computer-based probabilistic networks construction, Ph.D. Thesis, Medical Information Sciences, University of Pittsburgh, 1991.
[15] J. Huang, J. Lu, C.X. Ling, Comparing Naive Bayes, decision trees, and SVM with AUC and accuracy, in: Third IEEE International Conference on Data Mining, ICDM 2003, pp. 553–556.
[16] R.E. Kalman, A new approach to linear filtering and prediction problems, Transactions of the ASME – Journal of Basic Engineering 82 (1960) 35–45.
[17] P. Larrañaga, Structure learning of Bayesian networks by genetic algorithms: a performance analysis of control parameters, IEEE Journal on Pattern Analysis and Machine Intelligence USA 18 (9) (1996) 912–926.
[18] S.L. Lauritzen, D.J. Spiegelhalter, Local computations with probabilities on graphical structures and their application to expert systems, Journal of Royal Statistics Society B 50 (2) (1988) 157–194.
[19] Gang Li, Fu Tong, Honghua Dai, Evolutionary structure learning algorithm for Bayesian network and penalized mutual information metric, in: Proceedings IEEE International Conference on Data Mining, ICDM 2001, pp. 615–616.
[20] Xiao-Lin Li, Sen-Miao Yuan, Xiang-Dong He, Learning Bayesian networks structures based on extending evolutionary programming, in: Proceedings of 2004 International Conference on Machine Learning and Cybernetics, 3, 2004, pp. 1594–1598.
[21] M.M. Morales, N.C. Ramírez, J.L.J. Andrade, R.G. Domínguez, Bayes-N: an algorithm for learning Bayesian networks from data using local measures of information gain applied to classification problems, in: MICAI 2004: Advances in Artificial Intelligence, Lecture Notes in Artificial Intelligence, vol. 2972, Springer Verlag, Germany, 2004, pp. 517–526.
[22] M.M. Morales, R.G. Dominguez, N.C. Ramirez, A.G. Hernandez, J.L.J. Andrade, A method based on genetic algorithms and fuzzy logic to induce Bayesian networks, in: Proceedings of the Fifth Mexican International Conference in Computer Science, 2004, ENC 2004, pp. 176–180.
[23] K. Murphy, Dynamic Bayesian networks: representation, inference and learning, PhD Thesis, Computer Science Division, UC Berkeley, 2002.
[24] J. Pearl, Probabilistic Reasoning in Intelligent System, Morgan Kaufman Publishers, 1988.
[25] H. Peng, C. Ding, Structure search and stability enhancement of Bayesian networks, in: Third IEEE International Conference on Data Mining, ICDM 2003, pp. 621–624.
[26] R.S. Pindyck, D.L. Rubinfeld, Econometric Models and Economic Forecasts, Irwin/McGraw-Hill, 1998.
[27] L.R. Rabiner, B.H. Juang, An introduction to Hidden Markov models, IEEE ASSP Magazine 3 (1) (1986) 4–16.
[28] N.C. Ramírez, Building Bayesian networks from data: a constraint based approach, PhD Thesis, University of Sheffield, 2001.

[29] J.A. Rice, Mathematical Statistics and Data Analysis, second ed., Duxbury Press, 1995.
[30] J. Rissanen, Modeling by shortest data description, Automatica 14 (1978) 465–471.
[31] C.A. Rocha, A. Tupiassu, C.R.L. Francês, Á. L. Santana, V. Gato, L. Rego, Support decision system for load forecast on power systems, in: III Congress of Technological Innovation on Power Systems – III Citenel, 2005 (in Portuguese).
[32] S. Russel, P. Norvig, Artificial Intelligence, Prentice Hall, 2003.
[33] Á.L. Santana, C.R. Francês, C. Rocha, E. Favero, L. Rego, U. Bezerra, J. Costa, Load forecasting and learning of influence patterns of the socio-economic and climatic factors on the power consumption, Wseas Transactions on Mathematics 4 (3) (2005) 176–184.
[34] R.S. Sexton, S. McMurtrey, D.J. Cleavenger, Knowledge discovery using a neural network simultaneous optimization algorithm on a real world classification problem, European Journal of Operational Research 168 (2006) 1009–1018.
[35] S. Shetty, M. Song, Structure learning of Bayesian networks using a semantic genetic algorithm-based approach, in: Third International Conference on Information Technology: Research and Education, 2005, ITRE 2005, pp. 454–458.
[36] P.R. Spirtes, R. Shcheines, G. Clark, TETRAD II: Tools for Discovery, Lawrence Erlbaum Associates, Hillsdale, NJ, USA, 1994.
[37] C.C. Yang, Fuzzy Bayesian inference, in: Proceedings of IEEE International Conference on Systems, Man, and Cybernetics, Orlando, Florida, 1997.

Ádamo L. de Santana received the BS degree in Computer Science from the University of the Amazon in 2002 and in 2004 the MS degree in electrical engineering from the Federal University of Para, Brazil, where he is a PhD candidate. His research areas include computational intelligence, pattern recognition, data mining and time series.

Carlos R. Francês received the BS degree from the Federal University of Para in 1995, and the MS and PhD degree from the University of São Paulo in 1998 and 2001, respectively, all in Computer Science. He is a full professor in the Post Graduation Program in Electrical Engineering at the Federal University of Para. His research areas include performance evaluation, Markov chains, queuing theory and discrete simulation.

Cláudio A. Rocha graduated from the University of the Amazon in Data Processing (1991), receiving the masters degree in Computer Science from the University of São Paulo in 1999. He is currently a professor at the University of the Amazon and Federal Center of Technological Education of Para. His research areas include data mining, Bayesian networks and uncertainty.

Solon V. de Carvalho graduated in Mechanics-Aeronautics Engineering from the Technological Institute of Aeronautics in 1982. He received the MS degree in Analysis of Systems and Applications from the National Institute for Space Research in 1987, and the PhD degree in Automation-Production from the University of Toulouse III (Paul Sabatier) in 1991. He is a researcher at the National Institute for Space Research with interest in the area of Operational Research, focusing on Stochastic Modeling.

Nandamudi L. Vijaykumar graduated in Computing Technology from the Technological Institute of Aeronautics in 1978. He received the MS degree in Applied Computing from the National Institute for Space Research in 1984, and the PhD in Electronic and Computing Engineering from the Technological Institute of Aeronautics in 1999. He is currently a researcher at the National Institute for Space Research. His areas of interest include performance evaluation, time series analysis and computational modeling.

Liviane P. Rego is a student of the Electrical Engineering course in the Federal University of Para. Her research areas include computational intelligence, data mining and web development.

João W. Costa graduated in Electrical Engineering from the Federal University of Para in 1981. He received the MS degree in Electrical Engineering from the Pontifical Catholic University of Rio de Janeiro in 1989, and the PhD in Electrical Engineering from the State University of Campinas in 1994. He is a professor at the Federal University of Para and a researcher of the Brazilian Research Funding Agency. His research interests are toward the areas of electrical engineering, telecommunication and computing.