Modelling based on Rule Induction Learning

RUIZ-GOMEZ, J. (*); RAMOS-JIMENEZ, G. (**); VILLALBA-SORIA, A. (**)
* Systems Engineering and Automation Department
** Languages and Computer Science Department
University of Malaga
Complejo Tecnologico. Campus de Teatinos. 29071. Malaga
SPAIN

Abstract: - Many different approaches exist in the field of systems modelling, but in the case of non-linear systems is difficult to obtain conventional mathematical models. As an alternative way there are approaches based on Artificial Intelligence techniques; among this methods those based on if-then rules are the more well known and spread. One conventional approach in modellig is black-box modelling, that works with the only knowledge of a set of experimental input/output data.

In this paper two rule-based methods are applied to the problem of systems modelling trying to obtain a rule-based model from a data set. Both methods use an inductive learning technique based on the well known algorithm Quinlan's ID3. The first one uses fuzzy logic to represent the rules. The second one (CIDIM) is a non fuzzy clustering method. A significant feature, specially of the second method, is that it achieves models with a small number of rules. This is relevant for the transparency of the rule based-model.

The experimental data obtained from the mobile robot RAM velocity and curvature are the only knowledge used to obtain a qualitative model that reflects the robot dynamic behaviour. This model will be useful to design rule-based controllers.
IMACS/IEEE CSCC'99 Proceedings, Pages:5991-5996

Key-Words: - Inductive Learning, Qualitative Modelling, Data Mining, Fuzzy Clustering, Rule-based Control.

1. Introduction

Modelling, by means of rules, a non linear process arising from a set of data that reflect past behaviour of a system is an alternative approach to conventional I/O modelling of dynamic processes.

The aim of this work is to model a system by using both fuzzy and no fuzzy modelling techniques, based on the automatic rule generation from experimental data.

Due to the fact that perhaps the most successful Artificial Intelligence algorithms are a family of entropy-based techniques for inducing decisions trees, this is the base of the automatic learning mechanism for the rule generation. Decision trees organize tests hierarchically such that running the tests along a path from the top to any leaf classifies an example within a category.

On inductive learning the family of TDIDT (Top Down Induction Decision Tree) algorithms is well known one [2, 12]. From a set of experiences, these algorithms build a decision tree [3, 7]. Others TDIDT algorithms use different measures [16, 10, 11] but all TDIDT algorithms [6, 20, 24] always build a decision tree.

The best known algorithm for inductive learning is Quinlan ID3 [15]. Developed by J. R. Quinlan in 1979, this algorithm builds decision trees by using the entropy [21] as a measure of information.

This paper layout is: first an introduction about modelling and related works, next an overview on the two methods, the first by using fuzzy sets and the second a non fuzzy approach, followed by their application to a table of data of velocities and curvatures obtained from the mobile robot RAM, developed in Malaga University [13, 14].

In both methods, the maximum number of sets to divide the output ranges has been limited to 7 because this is an usual value for a fuzzy consequent partition.

2. Fuzzy Modelling

Fuzzy set approaches always seem to be appropriate when the modeling of human knowledge is necessary. This also happens when human evaluations are needed, in decision making, in project planning, in control, and so on.

This is the case in control systems design. Perhaps the main feature of fuzzy controllers and fuzzy models is their transparency, i.e. their easy comprehension by a human agent. Besides, fuzzy identification is adopted due to the difficulty of conventional mathematical modelling on certain systems, specially on non-linear ones, and due to the ability, settled by Wang [25] and Buckley *et al.* [1] to approximate any real function by means of a set of fuzzy rules.

Fuzzy identification begins with the Takagi and Sugeno approaches in 1985, based on Sugeno inference method [22]. Afterwards, entropy based learning was applied by Tani *et al*. [23] that identify the premises structure and selects the effective input variables by use ID3 or Delgado and Gonzalez, [4] that worked on an inductive learning process based on the frequency of appearance of certain patterns from raw data. Fuzzy clustering algorithms as the fuzzy c-means or the hyper-ellipsoidal clusters of Klawonn and Kruse [8, 9] have been used for deriving classification rules from data.

3. Problem Formulation

Our goal is to obtain a set of rules that describe the behaviour of a non-linear dynamic system. This means to get a model of a system without any knowledge about its components and relations between its variables (a black-box model). The only knowledge is a set of experimental I/O data. This data reflect a first order behaviour, where the output *y* in time *k+1* is a non-linear function of the inputs in instant *k*:

$$y(k+1) = \varphi\left[\upsilon(k), \rho(k)\right] \qquad (1)$$

The desired set of rules is achieved by means of an inductive learning process. It is relevant to obtain models with few rules in order to increase the transparency or human comprehension of the system behavior. This transparency improves on the human understanding of complex non-linear dynamic processes.

Both modelling methods are applied to model a mobile robot dynamic behaviour.

4. First method: fuzzy identification

Though it shares with the classical Takagi-Sugeno method two main features of a fuzzy clustering approach: the partition of variables ranges, and data fuzzification by means of these partitions, the rules are not stated in the Sugeno format, with the consequents as linear function of the input variables. In this case, the consequents are performed in the Mamdani type, i.e. as fuzzy sets. This is an advantage that improves on the model comprehension or transparency. But this method differs from that of Takagi-Sugeno in: the way that decision tree is built in and in the rules generation and input spaces partitioning techniques. A detailed description of this approach can be found in [18, 19].

As the measured data from a physical systems usually are continuous ones, the problem of translating them into fuzzy sets is critical. Two important points on numeric-fuzzy conversion of data arise: the number of sets to split every variable range, and the way the intersection between two adjacent fuzzy sets is treated.

To begin with, every variable range have been divided in two equal sets, by the middle of the range value, with a variable overlap between adjacent sets varying from 0 to 100 per cent. Also it is possible to perform a second way to split the variable range on the basis of critical points of the I/O curve [19], owing a derivative of zero value or not possessing derivative at all.

As ID3 is a learning from classes method, we need first a classification of all variables. All the numerical data are fuzzified. The raw data obtained by experimentation, give raise to a set of fuzzy experiences as these shown in the following Table I.

Table 1. Fuzzy Experiences

Experienc e #	Attribute x_1	Attribute x_2	Consequent y
1	NB	NS	NS
2	PS	Z	PS
3	NS	PB	PB
4	Z	PS	Z

5	...	...	..
...	..	...	...

NB: negative big; NS; negative small; PS: positive small; Z: zero; PB: positive big.

Afterwards a link between “experiences” and rules must be established, to apply ID3 to a fuzzy rules generation process. As facts or “experiences” we consider a set of data resulting in one measurement, i.e. a row in a numerical data table.

For every input variable, we find the consequents that reflect the action of every fuzzy set of the selected input. The probability *Pi* is obtained dividing the number of experiences in the input related with a same consequent, by the total number of times the antecedent is in the table.

After computing the amount of information, we can made a tree based on attributes or input variables. The tree development continues by the node with greater entropy. In a leaf node all its elements belong to the same class. A model rule is given by the path from the root node to a leaf node.

The sequence of models stops when the performances for a certain model achieves a desired value.

Model validation can be accomplished by an numeric index based on root mean square error between experimental and generated data, as well by using an error histogram or last by viewing a 3D graphic of experimental and modelled surfaces.

5. Second method: CIDIM

CIDIM method [17] builds a decision tree too, but for constructing the tree, CIDIM splits the experiences set in two ones: training set and control set. The training set is used for build the tree. The control set verifies if the expansion of a node improves the prediction. This supervision is achieved by means of two indexes, absolute (I_A) and relative (I_R). In each step of the CIDIM method a node is expanded only if these index increases.

$$I_A = \frac{\sum_{i=1}^{N} CORRECT(e_i)}{N} \qquad (2)$$

$$I_R = \frac{\sum_{i=1}^{N} P_{C(e_i)}(e_i)}{N} \qquad (3)$$

I_A : absolute index
I_R : relative index
N : number of experiences
e : experience
$C(e)$: class of e experience
$P_m(e)$: probability of m class for e experience

$$CORRECT(e) = 1 \quad \text{if } P_{C(e)} = max\{P_1(e), P_2(e), ..., P_k(e)\} \qquad (4)$$

$$CORRECT(e) = 0 \quad \text{if } P_{C(e)} \neq max\{P_1(e), P_2(e), ..., P_k(e)\} \qquad (5)$$

CIDIM can be applied to any problem with a finite number of attributes, each one is finite and ordered, and a finite number of classes.

6. Application

From thousands of data measured from the mobile robot RAM representing in each raw: input or desired velocity, input or desired curvature and real or output measured curvature a set of data of 388 raws has been random selected. The goal is to model the robot dynamic behaviour obtaining a set of rules that describes the relations between the asked or desired variables (input velocity and input curvature) and the actual measured output variable (output curvature). These rules can be afterwards used to design a rule-based controller

6.1 Fuzzy method

From the 388 randomly chosen data, a subset of 70 data has been selected, newly by using a random selection. From these data, 55 form the training set and 15 the test set. The values have been scaled to avoid negative curvatures. Searching for a minimum number of rules, the model obtained where the input velocity variable is considered as a single fuzzy sets (i.e. the influence of this input it is not very relevant to the output), the input curvature is split in four fuzzy sets, and the output results in seven sets model. The model has ten rules. Fuzzy set splitting is depicted in Fig. 2:

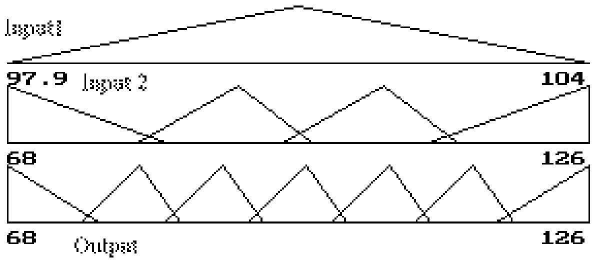

Fig. 1 Fuzzy model inputs and output.

A report showing the numerical values of the triangular fuzzy sets, performance index (about 4.71 % of error between measured and predicted values) and the resulting rules is shown in the following Table 2:

Table 2. Fuzzy model report.

Report: nd5i10s.inf
Model error index: 4.5929
MFS Matrix (I/O fuzzy sets):
Input 1:

1	97.9	101.1	101.1	104.2

Input 2

1	68	68	68	83.95
1	81.05	91.2	91.2	98.45
1	95.55	105.7	105.7	112.9
1	110.1	126	126	126

Output:

1	67.97	67.97	67.97	77.09
1	75.43	81.23	81.23	85.37
1	83.72	89.52	89.52	93.66
1	92	97.8	97.8	101.9
1	100.3	106.1	106.	110.2
1	108.6	114.4	114.	118.5
1	116.9	126	126	126

Model Rules (RLS Matrix):

10	10	10	10
10	1	1	1
10	1	1	2
10	1	2	2
10	1	2	3
10	1	2	4
10	1	3	4
10	1	3	5
10	1	3	6
10	1	4	6
10	1	4	7

The first column value 1 in inputs and output tables means the set in that row is active. The same meaning has values 10 in rules table.

6.2 CIDIM method

Input attributes (desired velocity and desired curvature) and the output attribute (difference between output and input curvatures) have been split in the following intervals:

Input velocity (v)	14 intervals
Input curvature (ρ)	7 “
Output difference	7 “

Intervals on input velocity are numbered from 1 to 14, intervals on input curvature from 1 to 7 and those of the output difference have labels: NB, NM ,NS, Z, PS, PM, PB

The experimental data set (388 experiences) has been random divided in a 80 % training set for build the classification tree and the 20 % set for testing the results.

Both ID3 and CIDIM methods has been applied to the training set. In CIDIM case, the training set has been newly split in two equal sets: the first one for training and the second one to test.

The results are summarized in the following Table 3:

Table 3. ID3 and CIDIM results

	Nodes	Rules	Success Index %
ID3	57	49	67.94
CIDIM	5	3	70.51

It can be seen the significant reduction in the number of rules and the tree size achieved by CIDIM, without diminishing the success index.

The generated tree and rules are shown in the following Fig 2 and Table 4:

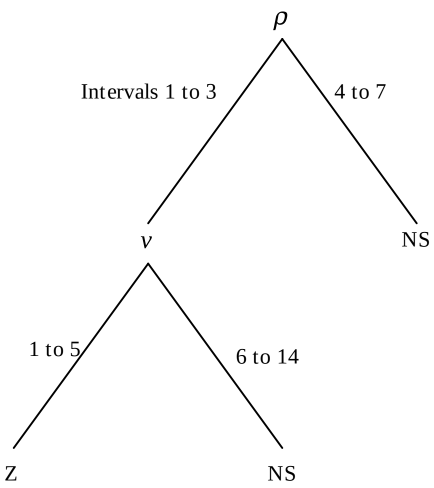

Fig. 2 CIDIM Decision tree

Table 4. CIDIM rules

R1: IF $\rho = 1$ to $3 \wedge v = 1$ to 5 THEN Z class
R2: IF $\rho = 1$ to $3 \wedge v = 6$ to 14 THEN NS class
R3: IF $\rho = 4$ to 7 THEN NS class

where rule R1 is supported by 115 associates experiences, R2 by 22 and R3 by 18.

7. Conclusions

Two qualitative modelling methods, based on inductive learning, have been presented, a fuzzy one and a non fuzzy one.

Both methods obtain a small number of rules, what is a relevant feature for any qualitative modelling approach. So the qualitative inductive leaning alternative for modelling problems looks like an actual alternative to classical mathematical modelling.

Though it is well known that fuzzy rules approaches achieves transparent or easy to understand models, in this case, a non fuzzy modelling technique obtains better results than the fuzzy modelling one.

The second method (CIDIM) gets a significant reduction in the number of rules, as can be seen when confronted with a classical inductive learning method as ID3. This rules reduction is an outstanding feature of the CIDIM approach.

References:

[1] Buckley, J.; Hayashi, Y., Fuzzy input-output controllers are universal approximators, *Fuzzy Sets and Systems* 58, 1993, 273-278.

[2] Buntine, W. and Nibblett, T. A Further Comparison of Splitting Rules for Decision-Tree Induction. *Machine Learning* 8, 1992, pp. 75-85.

[3] Cuena, J., *Inteligencia Artificial: Sistemas Expertos*. Alianza Editorial, 1987.

[4] Delgado, M. and González, A., An Inductive learning procedure to identify fuzzy systems, *Fuzzy Sets and Systems 55*, 1993, pp. 121-132,.

[5] Ettes, D. and Van den Berg, J., Representation and Learning Capabilities of Additive Fuzzy Systems, *Proceedings of the X-th Netherlands/Belgium Conference on Artificial Intelligence (NAIC'98)*, Amsterdam, 1998, pp. 193-202.

[6] Fayyad, U. and Irani, K., Technical Note. On the Handling of Continuous-Valued Attributes in Decision Tree Generation. *Machine Learning*. 8, 1992, pp. 87-102.

[7] Hunt, E. B., Marin, J. and Stone, P. T. *Experiments in induction*. Academic Press, 1996.

[8] Klawonn, F. and Kruse, R., Derivation of Fuzzy Classification rules from Multidimensional Data, in Lasker, G.E. and Liu, X. (Eds.), *Advances in Intelligent Data Analysis*. The International Institute for Advanced Studies in System Research and Cybernetics, Windsor, Ontario, 1995.

[9] Kruse, R. and Nauck, D., Learning Methods for Fuzzy Systems, Proceedings of the 3rd German GI Workshop Fuzzy-Neuro-Sisteme’95, Darmstadt (Germany), 1995, pp. 7-22.

[10] LÛpez de M·ntaras R., El problema de la selecciÛn de atributos en aprendizaje inductivo: Nueva propuesta y estudio experimental. *Nuevas tendendencias en Inteligencia Artificial.* Universidad de Deusto, 1992.

[11] LÛpez de M·ntaras R., Technical Note. A Distance-Based Attribute Selection Measure for Decision Tree Induction, *Machine Learning*. 6, 1991, pp. 81-92.

[12] Michalski, R., Carbonell, J. G and Mitchell, T. M. (Eds.), *Machine Learning, An Artificil Intelligence Approach,* Tioga Press, 1983.

[13] Muñoz, V.; Ollero, A. et al., Mobile Robot Trajectory Planning with Dynamic and Kinematic Constraints, *Procc. IEEE Int. Conf. Robotic and*

Automation. San Diego (CA). 1994, pp. 2802-2807.

[14] Ollero, A et al., Integrated Mechanical Design and Modelling of a New Mobile Robot, *Preprints of the SICICA'92 IFAC Symposium on Intelligent Components and Instruments for Control Applications*, Malaga, 1992, pp. 557-562.

[15] Quinlan, J. R., Discovering rules by induction from large collections of examples. In Michie, D., (Ed.), *Expert Systems in the microelectronics age*. Edinburgh University Press, 1979.

[16] Quinlan, J. R., Induction of Decision Trees. *Machine Learning* 1, 1986, pp. 81-106.

[17] Ramos-JimÈnez, G., Villalba-Soria, A., Aprendizaje con CIDIM (Control de InducciÛn por DivisiÛn Muestral) Una mejora de los algoritmos TDIDT, *Tech. Report LCC-ITI 97/08.* Departamento de Lenguajes y Ciencias de la ComputaciÛn. Universidad de M·laga, 1997.

[18] Ruiz-Gomez J.; Garcia-Cerezo, A. and Fernández-Ramos, R., Curvature Fuzzy Modelling for a Mobile Robot, in A. Ollero (Ed.), *Intelligent Components for Vehicles*. Pergamon Press. 1998.

[19] Ruiz-Gomez, J. and Garcia-Cerezo, A., An Inductive Inference Rule Generation Tool for Fuzzy Modelling, *Proceedings of IDEA'97, Intelligent Design in Engineering Application Symposium*, Aachen (Germany), 1997, pp. 113-117.

[20] Schlimmer, J. C., Ficher, D., A case study of incremental concept induction. *Proc. of the Fifth National Conference on Artificial Intelligence.* Morgan Kaufmann Publishers, 1986, pp. 496-501.

[21] Shannon, C. E., The mathematical theory of communication, *The Bell Systems Technical Journal* 27, 1948, pp. 379-423.

[22] Takagi, T. and Sugeno, M., Fuzzy Identification of Systems and Its Applications to Modeling and Control. *IEEE Trans. Syst., Man , Cybernetics*, 15,1985, pp. 116-132.

[23] Tani, T.; M. Sakoda and K. Tanaka, Fuzzy Modelling by ID3 algorithm and its application to prediction of a heather outlet temperature, *Proc. 1992 IEEE Int. Conf. on Fuzzy Systems*, San Diego (CA), 1992, pp. 1719-1723.

[24] Utgoff, Paul, Incremental Induction of Decision Trees. *Machine Learning*, 4, 1989, pp. 161-186.

[25] Wang, L. X., Fuzzy systems are universal approximators, *Proc. Int. Conf. on Fuzzy Systems*, San Diego (CA), 1992, pp. 1163-1170.