A view of High Dimensional, Large-Scale and Big Data Fuzzy Rule based Regression and Control Systems

Antonio Ángel Márquez
Dept. Tecnologías de la Información
University of Huelva
21007 Huelva, Spain
amarquez@uhu.es

Francisco Alfredo Márquez
C.E. Avanzados en Física, Matemáticas y Computación
University of Huelva
21007 Huelva, Spain
alfredo.marquez@dti.uhu.es

Ana María Roldán
Dept. Tecnologías de la Información
University of Huelva
21007 Huelva, Spain
amroldan@dti.uhu.es

Antonio Peregrín
C.E. Avanzados en Física, Matemáticas y Computación
University of Huelva
21007 Huelva, Spain
peregrin@uhu.es

***Abstract*—This work deals with the methodologies employed in the specialized literature on fuzzy regression and control systems to perform the learning process under conditions of high dimensionality and/or large-scale sets of examples. First, we look at algorithmic approaches in the last decade, where this specific area has experienced its bigger growth, and then at new resources nowadays based on distributed programming with MapReduce. We conclude trying to identify the challenges and future guidelines.**

Keywords—fuzzy regression; fuzzy control; high dimensionality; large-scale; big data

I. Introduction

It is well known that Fuzzy Logic technology was proposed by Zadeh [1], [2], [3] to solve control problems. The first Fuzzy Logic Controllers developed [4], [5], [6], [7] set a milestone in the Intelligent Control area, thanks to the simplicity whereby they tackle non-lineal control problems by using human concepts, knowledge, and experience, as well as approximate reasoning under uncertainty.

Since the early Linguistic or Mamdani Fuzzy Control Systems, whose fuzzy rules came from expert human knowledge, until the later Takagi-Sugeno (or Takagi-Sugeno-Kang, (TSK)) type, currently preferred in this applied area, whose knowledge is basically learned from example datasets, Fuzzy Control Systems have been employed to implement many control systems. They are especially appreciated due to the greater interpretability of the knowledge they achieve compared to other control methods.

However, most of the techniques used in the design of *Fuzzy Rule based Systems* (FRBSs) for regression and control (both of them apply the same technology) find efficiency difficulties when dealing with *High Dimensional* (HD) and/or *Large-Scale* (LS) datasets. The HD problem deals with the large number of variables, resulting in an exponential rule explosion [8], [9]. The LS problem is related with the large amount of data examples, which leads to extraordinary execution times or memory being needed.

The specialized literature addresses the aforesaid obstacles through data reduction, which includes variable and/or instance selection [10], [11], or parallel and distributed methodologies [12], [13], [14], [15], [16], [17]. Nevertheless, the scalability of these techniques continues to be a current research challenge.

Now, a new term, *Big Data* (BD), commands great attention when dealing with the processing of large volumes of data and the scalability of high performance computing resources at affordable cost. This technology is having increasing impact in very different areas, particularly in business intelligence, analytics, machine learning, and progressively giving rise to a few learning models for fuzzy systems in general terms and fuzzy regression and control models in particular, where this new programming paradigm seems to promise valuable contributions.

In this contribution, we present a review of the contributions around HD and LS, as well as the current BD focused on the area of fuzzy regression and control systems, analysing the resources employed to deal with them and, finally, the promising potential of BD in this area.

We organized this work as follows: Section II reviews the current lines for dealing with HD and LS datasets; Section III is specifically dedicated to discussing the new challenges and, finally, Section IV reaches some conclusions and points to future works.

II. Current Approaches for High Dimensional, Large Scale and Big Data for Fuzzy Regression and Control

The purpose of this section is to describe, using different taxonomies, the resources that have been employed in the specialized literature to solve the aforementioned HD and LS problems, as well as those which use the latest Big Data skills.

Likewise, a table is constructed (Table I) where we show the main papers on this area together with some relevant features and characteristics employed by their authors. Specifically, in this table you can find the bibliographic reference, the type of fuzzy system (descriptive linguistic, approximate linguistic, TSK with descriptive or approximate antecedents and its order), the problem addressed (HD, LS, or both together), the technique employed to solve it, the kind of computing (sequential, parallel or distributed), and finally some information about the biggest dataset employed in the experimental study developed by the authors to show the capabilities and performance of the proposal.

We will start by mentioning that, as is well known, in the fuzzy regression and control area we can find two kind of different architectures according to their fundamental desired characteristic, i.e. accuracy or interpretability:

- Interpretable fuzzy regression and control systems: descriptive linguistic (the ones that use common linguistic terms in the set of variables, both antecedents and consequents, of their rules, i.e. a single database for the entire system) or Mamdani, as well as TSK-0 with descriptive linguistic antecedents.
- Accurate fuzzy regression and control systems: they are the approximate linguistics (those using different linguistics terms for the variables of each rule, i.e. as many databases as there are rules) and TSK-0 with approximate linguistics, as well as TSK of order 1 and/or higher with approximate or descriptive linguistics antecedents.

The information regarding this can be found in the second row of Table I, under the legend "*Type of Fuzzy System*".

We must point out that, although interpretable fuzzy systems were originally thought to use expert knowledge, when we talk about HD, LS and Big Data problems, we are focused on situations where the system knowledge came from a learning process based on the use of sets of examples with these characteristics.

The "*HD/LS Problems*" and "*Technique*" columns in Table I show information on the type of problem that the authors were trying to tackle with their proposal, as well as the specific technique they used to do so. The proposals for HD or for HD and LS together are predominant. We can also observe that there are contributions for both interpretable and accurate models with HD and LS, and together combined.

A. Methods employed to address the HD problem

The different authors basically approach the HD problem using any kind of *Search Space Reduction* (SSR). We could also consider *Variable Selection* (VS) per se [10], [15], [16], [18], [19], [20], [21] as a kind of SSR, as it involves a reduction in the number of possible rules, which provides an implicit important reduction of the search space to be considered for the rule learning algorithms.

Frequently, VS is carried out through the evolutionary method that learns the granularity of the variables involved in the process [10], [15], [16], [19], [20] (letting the algorithm not consider a variable where its granularity is equal to 1), whereas in other cases, the VS comes implicit in the reduction of rule conditions [11], [18], which allows us to further reduce the searching space, resulting in a less complex set of rules.

Nonetheless, there are other strategies to achieve SSR in the design of the different parameters of the FRBSs. In this way, a resource frequently employed is the use of a lower number of parameters to tune the variables membership functions, for example, by using lateral tuning of the labels and then with only a single parameter [10], [15], [16], [19], [20], [22].

Another resource sometimes employed to perform SSR in HD problems is *cropping criteria*. It consists of thresholding the maximum number of rules admitted when the rule base is learned [10], [11], [18], [19], for example, if rules are being learned using an evolutionary process, it will be ended when the number of rules found reach the aforesaid threshold (in [10] it is fixed to 50 rules, in [11] and [18] at 30 rules, and in [19] they fix it to 100 rules). Normally, these papers point out that the threshold is determined by carrying out a number of different experimental trials.

To deal with HD problems, two techniques are used to achieve the SSR in paper [23], where the Adaptive Inference System have to be tuned. They are on one hand the use of an adaptive parameterized discrete t-norm, considerably reducing the exploration space, going from a continuous space to a discrete one with only a few selected efficient t-norms covering the different areas, and on the other hand, the use of a computation acceleration mechanism based on a look-up table.

Finally, regarding the SSR, there are other contributions that propose a considerable reduction by using a strategy to learn the rule base heuristically [18], [21].

B. Methods employed to address the LS problem

Concerning the LS problem, it can be observed that when using sequential algorithms two main approaches have been employed:

- Instance Selection (IS) [24] within the training set [10], [11], [15], [16], [18], [19], [20], sometimes inside a co-evolutionary learning process.
- Computation acceleration: Sometimes, when using evolutionary algorithms, the fitness function is estimated quickly [23]; in other cases, when computing the TSK coefficients [15], [16], [19], [20], [25] with estimations close to those performed in the calculation of Kalman filters [26] in [25] and [19], or with Elastic Net regularization [27] in [20], [15] and [16], and by other procedures or heuristics to reduce the computing time linked to the high number of instances.

The use of parallel and distributed algorithms lets us accelerate the computing time, for example, for the aforementioned computation of the fitness function in LS problems. Thus, in [28] authors use Graphic Processing Units (GPUs) hardware to compute the TSK coefficients in a parallelized approach.

TABLE I. SUMMARIZATION OF THE MAIN CONTRIBUTIONS AND SOME OF THEIR FEATURES

Reference / Method Name	*Type of Fuzzy System*	*HD/LS Problems*	*Technique*	*Sequential/ Parallel/ Distributed*	*Dataset Characteristics*
[21] SSEM	Descriptive Linguistic	HD	VS and SSR	Sequential	Max samples: 20640 Max variables: 13
[28]	Descriptive ants. TSK-0	LS	Speed-up TSK coeficients	Parallel-GPU	Max samples: 65536 Max variables: 2
[10] $FS_{MOGFS}{}^{e}$+TUN^{e}	Descriptive Linguistic	HD/LS	HD: VS and SSR LS: IS	Seqential	Max samples: 40768 Max variables: 85
[11] PAES-SOGA	Descriptive Linguistic	HD/LS	HD: SSR LS: IS	Sequential	Max samples: 40768 Max variables: 21
[23] ACO-HDP	Descriptive Linguistic	HD	HD: SSR	Sequential	Max samples: 20640 Max variables: 85
[18] PAES-RCS	Descriptive Linguistic	HD/LS	HD: VS and SSR LS: IS	Sequential	Max samples: 40768 Max variables: 40
[25]	Descriptive ants. TSK-1	HD	HD: Speed-up TSK coeficients	Sequential	Max samples: 70000 Max variables: 10
[19] METSK-HD^{e}	Aproximative ants. TSK-0	HD/LS	HD: VS, Speed-up TSK coeficients and SSR LS: IS	Sequential	Max samples: 40768 Max variables: 40
[20] FRULER	Descriptive ants. TSK-1	HD/LS	HD: VS, Speed-up TSK coeficients and SSR LS: IS	Sequential	Max samples: 40768 Max variables: 40
[15] S-FRULER	Descriptive ants. TSK-1	HD/LS	HD: VS, Speed-up TSK coeficients and SSR LS: IS and Splits dataset	Distributed MR-Spark	Max samples: 257560 Max variables: 140
[17] SCALABLE-EAD	Descriptive Linguistic	LS	LS: Split datasets	Distributed MR-Hadoop	Max samples: 40768 Max variables: 85

C. The use of MapReduce distributed programming model

However, it is in the scope of the MapReduce models area where things are more promising, due to its relative novelty and potential. Papers [15], [16], and [17] use this strategy, which divides the training example set between different computers, tackling the scalability of the LS problems very effectively, or distributed co-evolutionary learning with different groups of variables for HD with a reasonable effort in algorithms design. Thus, we shall finally focus on the recent proposals around this methodology

Although nowadays there are only a few proposals for fuzzy regression and control systems around this framework, they have emerged in the last two years. We highlight some recent works that allow scalability of the algorithms for handling big problems. S-FRULER [15] was proposed as the distributed version of FRULER [20], to improve the scalability limitations of FRULER following the MapReduce schema by using Apache Spark. This is an algorithm capable of building a complete TSK-1 fuzzy regression or control system with interpretable antecedents (descriptive linguistic), finding the involved subset of variables, the number of linguistic terms needed and their lateral tuning, and of course, the TSK-1 coefficients. S-FRULER is a complex methodology whose basics consist of splitting the training dataset into a set of smaller ones with the same size and processing them in a distributed Map function using a random VS and then a FRULER algorithm (which performs an IS and an evolutionary process with a double chromosome to learn the granularity, the lateral displacement of the linguistic terms, and the evaluation of the TSK-1 with the previously commented Elastic Net Regularization quick approximation), to finally aggregate the results of the Maps in order to achieve the final Knowledge Base (KB) in a non-trivial Reduce process. The applicability of this proposal was shown in [16], where the authors conducted a study which compared FRULER against S-FRULER. This experimental study was based on datasets and also on the fuzzy modelling of the thermal dynamics of buildings using the information of different sensors to automatically generate a KB of fuzzy rules for control. In this type of problem, one of the main issues is the generation of interpretable and accurate fuzzy models in reasonable time, given the large amount of data generated in the building. Results show that S-FRULER clearly improved on FRULER in terms of number of rules and, of course, in runtime.

Recently, in [17], another approach to performing Adaptive Defuzzification of Descriptive Linguistic fuzzy system for LS problems was presented. It is based on a classic data driven method, the Wang and Mendel (WM) algorithm [29] implemented in MapReduce, (named there *Scalable-WM*), followed by the evolutionary process to adapt the defuzzification parameters (also implemented in a second MapReduce phase, named there *Scalable-EAD*) in order to increase the accuracy of the final fuzzy model and obtain a more compact rule base. The aim of this work was then to propose and analyse a scalable pure linguistic FRBS with weighted defuzzification for fuzzy regression and control in

Big Data scenarios following the distributed MapReduce paradigm.

Nevertheless, it can be observed that the size of the datasets employed in the different papers on regression and control in this area, including those using MapReduce, is not large enough in number of examples or variables to be considered Big Data.

D. *Summarization of the different methodologies*

Next, and to finish this section, Table II shows a summary of the different techniques employed in the specialized literature to deal with the HD and LS problems. We consider that this schema could be useful for a rule based system for regression and control design, as a general view and then to select between the different options.

As can be seen in Table II, VS and SSR are arranging techniques employed with HD. Within VS techniques, the possible mechanisms to implement it are shown: either as a straight mechanism to do the VS per se; or by granularity search, accepting a granularity in the process that means that this variable does not take part in the process; or through a mechanism that lets us select the rule conditions in a similar way.

Attending to the mechanisms employed to solve the HD problem through the SSR technique, we can find the lateral tuning (which reduces the search space of the parameters to tune the linguistic terms); the cropping criteria for the number of rules (i.e. the maximum number of rules to be reached); the discretization of the continuous adaptive parameters of the fuzzy operators, i.e. for the adaptive parameterized t-norms or defuzzification; and finally, through the use of heuristics in the rule base learning methods.

As regards the techniques to deal with the LS problems, (continuing with Table II), we can see that they have been tackled either with straight IS, or by improving the efficiency of the fitness function computation. Specifically, speeding up the calculation of the TSK coefficients (when we are talking about them particularly) has been done by using any approximation (like the one used in the Kalman filters or the Elastic Net Regularization, or other); or by using look-up tables to store the previous computations; or by parallel computations using GPUs.

Table II does not include the techniques employed by the MapReduce distributed models, due to we have dedicated Section III to that discussion specifically.

III. Challenges in Scalable Fuzzy Regression and Control Scenarios

This section is focused on the fuzzy models for regression and control that we could name *truly scalable*, as sequential models, even if they use a mechanism to tackle a higher number of instances and/or variables successfully, are not really models that could scale from certain relatively low levels, and far from being considered as Big Data, so this section is devoted to distributed models and their challenges. Therefore, in order to implement *truly scalable* solutions, the easiest way is to resort to distributed programming and particularly MapReduce and its implementations, which allow iterations over data like Apache Spark.

TABLE II. Sumarization of techcnologies and mechanisms

Type of Problem	Techniques	
	Type of Technique	***Mechanism***
HD	VS	straight variable selection
		within granularity search
		within rule condition selection
	SSR	lateral tuning
		cropping criteria for rules
		discrete adaptive parameters
		heuristic rule base learning
LS	IS	straight instance selection
	fitness function speed-up	TSK coefficients
		look-up table
		GPUs

The requirement of scalability due to the high number of instances can be resolved in a simpler way through the distributed programming paradigm MapReduce than the scalability due to high number of variables. In this scenario, the LS problem is tackled through the disjoint split of the training example dataset into subsets, if the engaged algorithm is valid to be used with divide and conquer principles, in other words, if it is possible to aggregate the individual solutions found within the different subsets of instances to compound the global solution in the Reduce phase (Fig. 1 illustrates this simple schema). E.g. if we are learning the rule base by using the WM methodology distributed (each Map function perform the WM algorithm), the Reduce phase only has to merge the received rules with their associate matching in a similar way that the own WM does, this is, based on the matching degree. Other rule base generation methods, like those based on evolutionary algorithms may not be as directly adapted.

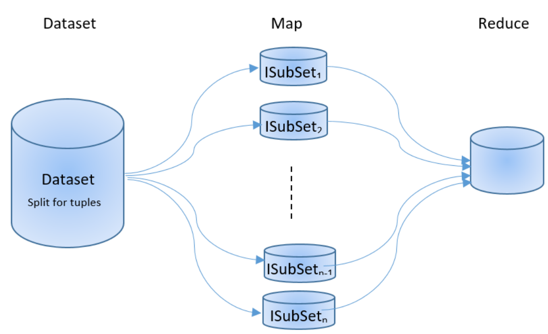

Fig. 1. Handling a LS dataset by partitioning it into some different subsets of instances

On the other hand, to achieve scalability if we are dealing with an HD problem, the regression and control literature is still very limited, and as far as we know, in order to achieve scalability, only the use of random subsets of variables (with replication) has been the method employed within each distributed subset, although this technique requires that the number of distributed processes be high enough to have sufficient information to merge the results previously learned in the distributed manner in an additional non-trivial aggregation process (Fig.2 show this method). E.g. to learn rule bases, this procedure is based on combining rules learned with partially different variables, so they have incompletely the same antecedents and values for them, that could be merged, although not elementally.

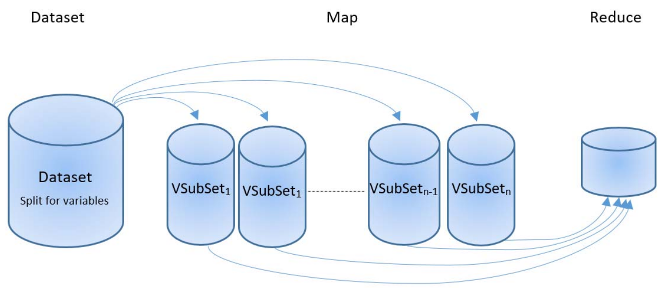

Fig. 2. Handling a HD dataset by creating some subsets of variables (with the whole set of instances)

If the we have to confront a dataset with both LS and HD circumstances simultaneously, the combination of both resources cited in the previous paragraph, i.e. subsets of variables with repetition within each disjoint subset of examples, allow us to resolve the hitch, but it may suffer from limitations related with the complexity of the problem, due to the fact that all the variables would not be really treated with all the examples. It is more similar to a pre-processing that performs an IS followed by a VS procedure like the one depicted in the previous paragraph. It is showed in Fig.3.

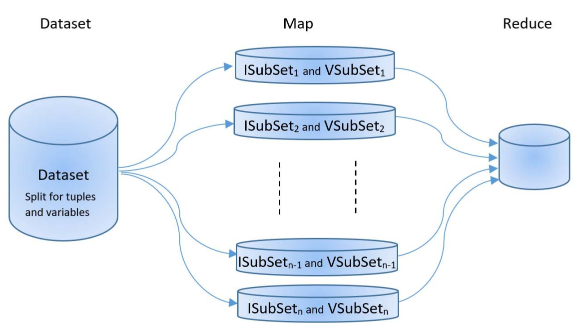

Fig. 3. Handling a LS and HD dataset by partitioning it into some different subsets of instances and with not disjoint subsets of variables each one

Perhaps due to the previously cited reason, the datasets still employed in regression and control are not large enough, so the contributions for Big Data environments are still scarcer than, for example, in fuzzy classifier systems. Hence, we can indeed expect an important growth in the next few years.

In our opinion, what makes fuzzy regression and control systems particularly interesting compared to other non-linguistic options is the interpretability they offer. In this sense, we are not only talking about the most interpretable fuzzy regression and control systems, i.e. descriptive linguistics, but also at the other extreme, the TSK of order greater than 0, as their antecedents are interpretable and they thus provide some comprehensive information about their behaviour: i.e. when they act; and between some and others, there may also be different balances among the precision required in each case and the interpretability they can provide. Furthermore, the interpretability in fuzzy systems, which have been well studied in the last decade, could reformulate its theories based on truly big datasets as, for example, thousands of descriptive linguistic rules do not seem to be very interpretable, although they are *a priori* interpretable knowledge structures.

Scalability may be desirable for both the number of instances, LS problems, and number of variables, HD problems, as commented throughout this work. We think that a fuzzy regression and control system scalable in both LS and HD problems may not always be necessary, and therefore, it could be left out due to its associated complexity.

It should also be noted that the development of distributed technologies in the area of regression and control not only has a slope of HD and LS problems but also in conventional problems. The learning of complete regression and control systems, i.e. using significant variables, learning the granularity and the knowledge base, is a problem that thanks to the skills of distributed programming can find new procedures to obtain more compact and accurate fuzzy regression and control systems.

IV. Concluding Remarks

In this paper, we have reviewed the literature around HD and LS problems specifically in the area of fuzzy regression and control: from the previous sequential methods to the new currently distributed ones in the framework of MapReduce. True scalability seems to be reachable only following the current models, or at least, in a reasonably easy way, although some of the lessons learned previously for the HD and LS problems are also useful to design more efficient future models. Therefore, the specialized literature in this area will presumably see considerable growth in the next few years.

The technological challenge is to develop models that are as simple and effective as possible, but there is also a challenge in small data, where distributed technology can achieve better results than the traditional models have provided until now.

From the outset, fuzzy systems have been a technology of a purely practical nature: the current challenge consists of facilitating and fulfilling their potential for application, including the new technological developments, to real world problems, to maintain a similar significance within the intelligent control area.

Regression and control systems have basically been sustained by their abilities in non-linear control problems, by the close handling of concepts, knowledge and experience of how humans use them, and by reasoning under uncertainty. The interpretability of fuzzy systems in general terms is always of paramount importance in them. In a future work, we plan to study the extent to which these abilities are maintained or not under Big Data environments, both by the different methods and by their own global sense.

ACKNOWLEDGMENT

This work was partially supported by the Spanish Ministry of Education and Science under Project TIN2017-89517-P

REFERENCES

[1] L.A. Zadeh. Fuzzy sets. Information and Control, vol. 8, pp. 338–353, 1965.

[2] L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes,". IEEE Trans. Syst., Man, Cybern., vol. 1, p.p. 28–44, 1973.

[3] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning,". Partes I, II y III Inf. Sci., vol. 8, 8, 9, pp. 199–249, 301–357, 43–80, 1975.

[4] E. H., Mamdani, "Application of fuzzy algorithms for control of simple dynamic plant," in proc. of the Institution of Electrical Engineers, vol. 121. IET, pp. 1585–1588, 1974.

[5] E. H Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller,". Int. J. Man-Machine Studies vol. 7, no.1, pp. 1–13, 1975.

[6] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control,". IEEE Trans. Syst., Man, Cybern., vol. 1, pp.116–132, 1985.

[7] M. Sugeno and G. Kang, "Structure identification of fuzzy model,". Fuzzy Sets Syst., vol. 28 no.1, pp. 15–33. 1988.

[8] W. E. Combs and J. E. Andrews, "Combinatorial rule explosion eliminated by a fuzzy rule configuration," IEEE Trans. Fuzzy Syst., vol. 6, no. 1, pp. 1–11, Feb. 1998.

[9] Y.Jin, "Fuzzy modeling of high-dimensional systems: Complexity reduction and interpretability improvement," IEEE Trans. Fuzzy Syst., vol. 8, no. 2, pp. 212–221, Apr. 2000.

[10] R. Alcalá, M.J. Gacto and F. Herrera, "A fast and scalable multi-objective genetic fuzzy system for linguistic fuzzy modeling in high-dimensional regression problems,", IEEE Trans. Fuzzy Syst., vol. 19, no. 4, pp. 666–681, 2011.

[11] M. Antonelli, P. Ducange and F. Marcelloni, "Genetic training instance selection in multiobjective evolutionary fuzzy systems: a coevolutionary approach,", IEEE Trans. Fuzzy Syst, vol. 20, no. 2, pp. 276–290, 2012.

[12] I. Robles, R. Alcalá, J.M. Benítez and F. Herrera, "Evolutionary parallel and gradually distributed lateral tuning of Fuzzy Rule-Based Systems,", Evol. Intell., Special Issue on GFS: New Advances,vol. 2, no. 5, pp. 5-19, 2009.

[13] M. A. De Vega, J. M. Bardallo, F. A. Márquez and A. Peregrín, "Parallel distributed two-level evolutionary multiobjective methodology for granularity learning and membership functions tuning in linguistic fuzzy systems," in proc. of ISDA'09 Int. Conf. on Intelligent System Design and Applications, Pisa, Italy, 2009, pp. 134–139.

[14] J. M. Bardallo, M. A. De Vega, F. A. Márquez and A. Peregrín, "Parallel evolutionary multiobjective methodology for granularity and rule base learning in linguistic fuzzy systems" in proc. of FUZZ-IEEE'09 IEEE Int. Conf. on Fuzzy System, Jeju Island, Korea, 2009, pp. 1700–1705.

[15] I. Rodríguez-Fdez, M. Mucientes, and A. Bugarín, "S-FRULER: Scalable fuzzy rule learning through evolution for regression," Knowl. Based Syst. , vol. 110, pp. 255–266, 2016.

[16] P. Rodriguez-Mier, M. Mucientes and A. Bugarín, "Scalable modeling of thermal dynamics in buildings using fuzzy rules for regression," in proc. of FUZZ-IEEE'17 IEEE Int. Conf. on Fuzzy System, Naples, 2017, pp. 1-6.

[17] A. A. Márquez, F. A. Márquez and A. Peregrín, "A scalable evolutionary linguistic fuzzy system with adaptive defuzzification in big data," in proc. of FUZZ-IEEE'17 IEEE Int. Conf. on Fuzzy System, Naples, 2017, pp. 1-6.

[18] M. Antonelli, P. Ducange and F. Marcelloni, "An efficient multi-objective evolutionary fuzzy system for regression problems,", Int. J. Approx. Reason.,, vol. 54, no.9, pp. 1434-1451, 2013.

[19] M.J. Gacto , M. Galende , R. Alcalá and F. Herrera , "METSK-HDe: A multiobjective evolutionary algorithm to learn accurate tsk-fuzzy systems in high-dimensional and large scale regression problems," Inf. Sci., vol. 276, pp. 63–79 , 2014.

[20] I. Rodríguez-Fdez, M. Mucientes, and A. Bugarín, "FRULER: Fuzzy rule learning through evolution for regression," Inf. Sci., vol. 354, pp. 1–18, 2016.

[21] D. Wang, X-J. Zeng and J.A. Keane, "A simplified structure evolving method for Mamdani fuzzy system identification and its application to highdimensional problems," Inf. Sci., vol. 220, pp. 110–123, 2013.

[22] R. Alcalá, J. Alcalá-Fdez, and F. Herrera, "A proposal for the genetic lateral tuning of linguistic fuzzy systems and its interaction with rule selection," IEEE Trans. Fuzzy Syst., vol. 15, no. 4, pp. 616–635, Aug. 2007.

[23] A.A. Márquez, F.A. Márquez, A. Roldán, and A. Peregrín, "An efficient adaptive fuzzy inference system for complex and high dimensional regression problems in linguistic fuzzy modelling," Knowl. Based Syst., vol. 54, pp. 42–52, 2013.

[24] H.Liu and H.Motoda, "On issues of instance selection," DataMin. Knowl. Discov., vol. 6, no. 2, pp. 115–130, 2002.

[25] M. Cococcioni, B. Lazzerini and F. Marcelloni, "On reducing computational overhead in multi-objective genetic Takagi–Sugeno fuzzy systems," Appl. Soft Comput., vol. 11, no. 1, pp. 675–688, 2011.

[26] R. E. Kalman, "A new approach to linear filtering and prediction problems," Trans. ASME – J. Basic Eng. 82 (Series D) pp. 35–45, 1960.

[27] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," J.R.Statist.Soc.B, vol. 67, no. 2, pp. 301–320, 2005.

[28] B. B. Ferreira and A. J.O. Cruz, "A parallel method for tuning Fuzzy TSK systems with CUDA," in: SBC – Proc. of SBGames, Brazilian Computer Society (SBC), pp. 5–8, 2012.

[29] L. Wang and J. Mendel, "Generating fuzzy rules by learning from examples," IEEE Trans. Syst., Man, Cybern., vol. 22, no. 6, pp. 1414–1427, 1992.