

Optimal Rule-based Granular Systems from Data Streams

Daniel Leite, Goran Andonovski, Igor Škrjanc, Fernando Gomide

Abstract—We introduce an incremental learning method for the optimal construction of rule-based granular systems from numerical data streams. The method is developed within a multiobjective optimization framework considering the specificity of information, model compactness, and variability and granular coverage of the data. We use α -level sets over Gaussian membership functions to set model granularity and operate with hyper-rectangular forms of granules in nonstationary environments. The resulting rule-based systems are formed in a formal and systematic fashion. They can be useful in time series modeling, dynamic system identification, predictive analytics, and adaptive control. Precise estimates and enclosures are given by linear piecewise and inclusion functions related to optimal granular mappings.

Index Terms—Evolving Systems, Granular Computing, Information Specificity, Online Data Stream, Adaptive Systems.

I. INTRODUCTION

Since the last decade, real-time model adaptation and data stream processing have become key research issues motivated by two broad reasons. First, a proliferation of automated systems, small scale computing devices, sensor networks, social media, and data capture and communication technologies have produced large volumes of data. Currently, the amount of data is increasing at a rate that sometimes outpaces available storage capacity. Often, data are stored only, without an intended purpose in mind. Second, there has been a keen aspiration toward a more human-like, evolving intelligence on chips, softwares, and systems. Broadly stated, the focus of data analysis has shifted from offline batch processing of data to the incremental handling of online data streams [1] [2].

Granular computing methods have been developed to uncover meaningful knowledge from large data sets. These methods examine the information flow in dynamic environments, and produce and keep updated a granular model that can be linguistically understood and be used in pattern classification [3] [4] [5], prediction [6] [7] [8], localization [9], and model-based control [10] [11] [12]. The general idea of granular computing can be traced to Tsau Young Lin and Lotfi Zadeh [13]. The concepts of granules and granular mapping are used in the process of learning from data streams. Granules denote clumps of objects, subsets or elements of a domain drawn

together by similarity, proximity or functionality [16] [14] [15]. Granular mappings are defined over granules and maps them into a collection of granules expressed in some output space. Granular mappings are very often found in rule-based systems, where the mapping is given by If-Then statements [16] [17].

The concept of granule is highly correlated to the concept of information specificity [18]. An increase in the specificity of information tends to increase its usefulness to assist decisions and actions. On one hand, being very specific we risk being both incorrect – since experimental evidence may not be embraced by the information – and flooded by too many details. On the other hand, being very little specific, we can assure that the true values are included. However, we may end up with coarse information and an unhelpful model. In information theory this is called specificity-correctness tradeoff [19]. The granularity [20] of granular models should be legitimized by, and reflect the available data to the highest extent in the sense that granules should maximize data coverage by paving the data space. At the same time, granules should be specific for a more meaningful description.

Granular models built from detailed data streams can be supported by many computational frameworks such as interval mathematics, fuzzy sets, rough sets, shadow sets, cluster analysis, decision trees, neighborhood systems. On top of these are generalized constraints in the sense of Zadeh's general theory of uncertainty [21], which are used to delimit and represent granules within the different frameworks. Computing with granules permit choices of representative objects and handling tools. Regardless of the framework, online granulation aims to retain the essence of stream data as granular objects. While direct application of machine-learning methods to nonstationary data streams is very often infeasible since (i) it is difficult (or impossible) to maintain all the data in memory for multiple training iterations; and (ii) typical nonstationaries usually require a complete offline-redesign of models, the ultimate goal of computing with evolving granules in online environment is to gradually develop more abstract, human-centric representations of the data.

Evolving intelligent systems is a mainstream of research in online data modeling [22] [23] [24] [25]. Here we use the term *evolving* in the sense of gradual development of granules, rule base, and associated parameters. With real-time parametric and structural adaptation, the burden of redesigning models from scratch from time to time is avoided in evolving modeling. A variety of heuristic methods have been proposed over the last ten years to guide the development, and incremental adaptation of rule-based models from nonstationary numerical [4] [11]

Manuscript received on February 21, 2019; revised on April 9, 2019.

Daniel Leite (daniel.leite@deg.ufa.br) is with the Department of Engineering, Federal University of Lavras, Brazil. Fernando Gomide (gomide@dca.fee.unicamp.br) is with the School of Electrical and Computer Engineering, University of Campinas, Brazil. Goran Andonovski (goran.andonovski@fe.uni-lj.si) and Igor Škrjanc (igor.skrjanc@fe.uni-lj.si) are with the Faculty of Electrical Engineering, University of Ljubljana, Slovenia.

[26] [27] [28] and granular [6] [7] [12] [29] data streams. Applications in many distinct domains, such as sensor networks, real-time finance, weather and energy forecasting, control of unmanned vehicles and finger dynamics, chaotic systems, and anomaly detection, have been reported [22] [30] [31] [32] [51]. Interesting and persuasive practical solutions to immediate goals have been achieved. Nonetheless, propositions, lemmas, theorems, and assurance that certain conditions will be fulfilled are still scarce in the field of evolving clustering, evolving neuro-fuzzy, and rule-based modeling from data streams.

The problem we address in this paper is concerned with the optimal construction of rule-based granular models from numerical data streams. The evolving optimal granular system (eOGS) uses piecewise affine and inclusion functions associated with Gaussian and hyper-rectangular forms of granules to give granular and numerical estimates of nonstationary functions. Approximand functions can be time-series models, decision boundaries between classes, control or regression functions in general. eOGS self-adapts its granular structure whenever a new concept emerges, or a drift occurs in the data stream. Adaptation is done by trading-off several conflicting objectives. The approach to choose the model granularity accounts for the variability of the data, level sets, error estimations, and the specificity-correctness tradeoff. Multiple granularities are obtained along the construction of the model. They are justified by the diversity of information occurring in the data. Granulation facilitates incremental updating and maintenance of a statistical summary of the data belonging to a granule with minor storage and processing overhead.

The remainder of this paper is structured as follows. Section II gives the mathematical background needed to support the evolving granular framework and learning algorithm. A specificity measure for Gaussian granules is given, and the ϵ -constraint formulation of the multiobjective design scheme is presented. Section III develops the eOGS algorithm to generate granular structures and mappings from data streams. Section IV addresses an interactive setting of eOGS to achieve particular solutions, and an automatic, fully autonomous eOGS approach. Section V presents application examples, and Section VI concludes the paper and suggests issues for further development.

II. CONTEXTUALIZATION AND BACKGROUND

The fundamental ideas and principles behind evolving optimal granular systems are summarized in this section.

A. Justifiable granularity

The realization of information granules from datasets has been discussed [20] [33] [34] [35] [36]. In particular, the principle of justifiable granularity [20] stands for a broad concept to guide the formation of meaningful granules based on experimental evidence, i.e. numerical or detailed granular data. In other words, the more data are included within the bounds of a granule, the better. Data coverage is a desired feature. We want granules to enclose all, or at least most, of the data. However, granules should be as specific as possible to come with an related semantic, and to be more supportive of further

decisions or actions. If many data are included in a granule, the granule may become too wide. By contrary, if the granule is too narrow, few data will be covered. The requirements of experimental evidence and specificity are conflicting. In this paper we resort on a multiobjective optimization based formulation using α -level sets and Pareto fronts to achieve a trade-off between these requirements.

A rule-based model is formed in a formal fashion. First, the essence of the information carried by the data stream is captured by means of recursive equations, fuzzy sets and local functions. We use α -cuts and cylindrical extensions of α -level sets over the Cartesian product space to form hyper-rectangles – a pavement for the data. Given the requirements for the resulting granular construction and for its estimates, a systematic and optimal way of forming granules and granular mappings is developed. We may find a feasible value for α and for other adaptive parameters in such a way that: (i) a maximum number of rules and a minimum specificity for the granules are respected; (ii) numerical or granular estimation errors are treated as priority; or (iii) multiple requirements are balanced. In any scenario, optimally placed granules and optimal inclusion and local functions are obtained.

B. Interval information granules

Constructing models from data streams requires incremental learning to keep an updated summary of the data. Learning should be one-pass, and neglect all previously seen data samples: each data sample should be processed only once and removed from memory. We particularly want to capture spatial and temporal variability of numerical data $x^{[h]} = (x_1, \dots, x_j, \dots, x_n)^{[h]}$, $h = 1, \dots$, using a set of granules $\gamma = \{\gamma^1, \dots, \gamma^c\}$. A local γ^i (a n -dimensional hyperbox) is chosen to fit a sample $x^{[h]}$, that is, to include the information conveyed by $x^{[h]}$. Otherwise, if $x^{[h]}$ is not sufficiently related to any γ^i , then a new granule γ^{c+1} can be created – expanding the current collection γ . Therefore, γ is a granular model that describes the data stream $x^{[h]}$, $h = 1, \dots$

Granules of interval nature depend essentially on the lower l_j^i and upper L_j^i endpoints of axis-aligned hyper-rectangles. Endpoints are determined by spanning α -level subsets of individual features, $\alpha \in (0, 1]$. Assume for simplicity a Gaussian membership function $\Pi_j^i = \mathcal{G}(\mu_j^i, \sigma_j^i)$, as illustrated in Fig. 1, which conveys the representation of the j -th feature of the i -th granule. Gaussians can be easily captured and incrementally adapted from the data stream $x^{[h]}$, $h = 1, \dots$. We are interested in the range of values that data samples rest on most occasions. Parameter α cuts Π_j^i at each tail and produces an interval $[l_j^i, L_j^i]$ whose extension over the Cartesian product space $X_1 \times \dots \times X_j \times \dots \times X_n$ assembles an interval-valued granule. Interval granules cover the data, and fuzzy Gaussians maintain the essence of the information.

Depending on the tightness of Π_j^i and on α , granules and rule-based models may achieve any desired specificity. For example, if a learning algorithm is used to recursively adapt μ_j^i and σ_j^i , then Π_j^i may enlarge, shrink and drift over time to track nonstationarities in data streams. At the same time, we can use α to manage the size of intervals $[l_j^i, L_j^i]$, i.e., to

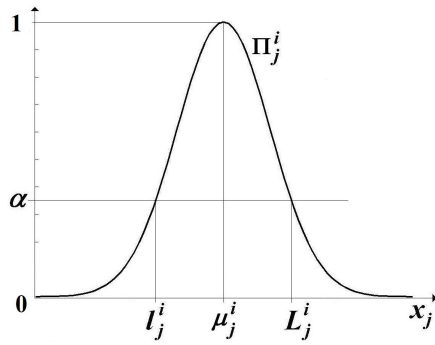


Fig. 1: α -level $[l_j^i, L_j^i]$ of the membership function Π_j^i

control the specificity of local interval granules. Notice that if α is set to 1, then the corresponding hyperbox degenerates to a point. In opposition, if α approaches 0, then a single hyperbox covers the entire data space.

C. Specificity measure

Specificity refers to the amount of information conveyed by a fuzzy set [19].

A measure of specificity of $\Pi \subseteq X$, denoted $\text{Sp}(\Pi)$, assigns to Π a value such that $0 \leq \text{Sp}(\Pi) \leq 1$ and

$$\begin{aligned} \text{Sp}(\Pi) &= 1, & \text{if } \Pi &\equiv \{x\}; \text{ (singleton)} \\ \text{Sp}(\Pi) &= 0, & \text{if } \Pi &= \emptyset; \\ \text{Sp}(\Pi^1) &\leq \text{Sp}(\Pi^2), & \text{if } \Pi^1 &\supseteq \Pi^2. \end{aligned}$$

$\text{Sp}(\Pi)$ measures the specificity of a set in the sense that it indicates the degree to which Π directs to one and only one element as its evidence.

Suppose Π is a subset of X . Yager [19] defines a general class of specificity measure over continuous domains as

$$\text{Sp}(\Pi) = \int_0^{\alpha_{max}} F(\lambda(\Pi_\alpha)) d\alpha, \quad (1)$$

where $\Pi_\alpha = \{x : \Pi(x) \geq \alpha\}$ is the α -level of Π ; α_{max} is the height of Π ; λ is a monotonic measure; and $F : [0, 1] \rightarrow [0, 1]$ is a function such that $F(0) = 1$; $F(1) = 0$; and $0 \leq F(x_1) \leq F(x_2)$, for $x_1 > x_2$.

Let Π be a Gaussian membership function in X with modal value μ and dispersion σ ,

$$\Pi = e^{-(x-\mu)^2/2\sigma^2}. \quad (2)$$

Gaussians are normal (height equal to 1), and have infinite support. Level sets of Gaussian membership functions are intervals Π_α whose centers and radii are of the form $\mu \pm \sqrt{-2\sigma^2 \ln(\alpha)}$.

If

$$M \triangleq \sqrt{-2\sigma^2 \ln(\alpha)}, \quad (3)$$

then

$$\Pi_\alpha = [\mu - M, \mu + M]. \quad (4)$$

Using the Lebesgue-Stieltjes measure [37] for λ in a totally bounded domain $X = [c, d]$ we have

$$\lambda(\Pi_\alpha) = \frac{wdt(\Pi_\alpha)}{wdt(X)} = \frac{\mu + M - (\mu - M)}{d - c} = \frac{2M}{d - c},$$

where $wdt(\cdot)$ means the interval width which is equal to the absolute difference between the endpoints of the interval [38]. If all $x \in [0, 1]$, then $[c, d] = [0, 1]$. Assuming $F(z) = 1 - z$ we get

$$F(\lambda(\Pi_\alpha)) = 1 - \frac{2M}{d - c}.$$

Using the Yager definition of specificity (1) and the α -level set Π_α we have

$$\begin{aligned} \text{Sp}(\Pi_\alpha) &= \int_0^1 F(\lambda(\Pi_\alpha)) d\alpha \\ &= 1 - \int_0^1 \lambda(\Pi_\alpha) d\alpha \\ &= 1 - \frac{2}{d - c} \int_0^1 M d\alpha. \end{aligned}$$

Notice that the term $2 \int_0^1 M d\alpha$ is the area under Π_α , and because the width of Π_α is $2M$ we have that $\int_0^1 wdt(\Pi_\alpha) d\alpha = wdt(\Pi_\alpha)$. Therefore,

$$\text{Sp}(\Pi_\alpha) = 1 - \frac{wdt(\Pi_\alpha)}{d - c} = 1 - \left(\frac{L - l}{d - c} \right),$$

where $l = \mu - M$, and $L = \mu + M$ are the endpoints of (4). In terms of the dispersion σ and α ,

$$\text{Sp}(\Pi_\alpha) = 1 - \frac{2\sqrt{2}\sigma\sqrt{-\ln(\alpha)}}{d - c}. \quad (5)$$

Let $\Pi^i = \{\Pi_1^i, \dots, \Pi_j^i, \dots, \Pi_n^i\}$ be a set of Gaussian functions such that Π_j^i has domain X_j , and let $X^n = X_1 \times \dots \times X_j \times \dots \times X_n$. Through α -cuts we may assemble an interval granule γ^i in X^n . We define the specificity of a n -dimensional granule γ^i as the average of the specificity measure of each of its n components, namely,

$$\text{Spa}(\gamma^i) = 1 - \frac{2\sqrt{2}}{n} \sum_{j=1}^n \frac{\sigma_j^i \sqrt{-\ln(\alpha)}}{d_j - c_j}. \quad (6)$$

D. A compromising decision strategy

Data-stream-driven learning methods to build granular models should ideally adapt key decision parameters to provide the requested outcomes in an optimal sense and satisfy constraints. For example, parameter $\alpha \in (0, 1]$, which gives level sets in eOGS is a free quantity representing a decision to be made. The decision maker may either interactively set α – and other key parameters – to force a specific model structure and behavior, or use automatic procedures for the purpose of optimizing an objective function within the feasible region formed by constraints. While evolving methods should capture information from the data stream, and represent the current environment as accurately and stably as possible, the parameters and structure of models should remain flexible to deal with changes (stability-plasticity dilemma [39]). Let \mathbb{P} represent all key eOGS parameters to be set. The remaining parameters will be presented formally in the description of the learning algorithm in the next section.

Possible criteria that might be used to guide the adaptation of the parameters \mathbb{P} include the numerical estimation error (E_n), granular estimation error (E_g), specificity of the granular mapping ($Spa(\gamma)$), and total number of rules (c). Generally, we want to reduce the numerical and granular errors, increase the specificity/meaningfulness of the granules, and maintain a compact rule-base structure. Clear tradeoffs emerge, e.g., a smaller number of rules tends to produce less specific granular mappings, and may generate better or worse estimations depending on the data stream.

A multiobjective function to be minimized is given as

$$\begin{aligned} F(\mathbb{P}) = \min & [f_1(\mathbb{P}), f_2(\mathbb{P}), f_3(\mathbb{P}), f_4(\mathbb{P})] \\ \text{s.t. } & \mathbb{P} \in \Omega \end{aligned} \quad (7)$$

where Ω is the parameter space and

$$\begin{aligned} E_n &\triangleq f_1(\mathbb{P}), \\ E_g &\triangleq f_2(\mathbb{P}), \\ -Spa(\gamma) &= -\sum_{i=1}^c Spa(\gamma^i) \triangleq f_3(\mathbb{P}), \\ c &\triangleq f_4(\mathbb{P}). \end{aligned}$$

The objectives of (7) compete so that the solution is not unique. Improvement in one objective may degrade the others but maintain Pareto optimality. Experts can express preference for a solution along a tradeoff surface. We provide two approaches to deal with this issue in the next sections. The first is an interactive approach to choose \mathbb{P} and produce a set of solutions, which is suggested to the decision maker. The second is an automatic, fully autonomous approach to choose \mathbb{P} in a balanced way.

In the ϵ -constraint method [40], a priority objective is chosen to be optimized while the other objectives are converted into constraints by setting an upper bound to each of them. The problem (7) takes the form

$$\begin{aligned} F(\mathbb{P}) = \min & f_s(\mathbb{P}) \\ \text{s.t. } & f_t(\mathbb{P}) \leq \epsilon_t, \forall t, t \neq s \\ & \mathbb{P} \in \Omega. \end{aligned} \quad (8)$$

The following theorems show that Pareto optimal solutions of (7) can be obtained by solving (8) for distinct choices of ϵ_t [41] [42].

Theorem 1. The solution of (8) is weakly Pareto optimal.

Proof. [41] Let $\mathbb{P}^* \in \Omega$ be a solution of (8). Assume \mathbb{P}^* is not weakly Pareto optimal. Then, there exists some other $\mathbb{P} \in \Omega$ such that $f_r(\mathbb{P}) < f_r(\mathbb{P}^*) \forall r$. This means that $f_t(\mathbb{P}) < f_t(\mathbb{P}^*) \leq \epsilon_t \forall t, t \neq s$. Thus \mathbb{P} is feasible and $f_s(\mathbb{P}) < f_s(\mathbb{P}^*)$. Being so, \mathbb{P}^* is a contradictory solution. Therefore, \mathbb{P}^* needs to be weakly Pareto optimal. **Q.E.D.**

Moreover,

Theorem 2. A decision $\mathbb{P}^* \in \Omega$ is Pareto optimal if it is a solution of (8) for any s and $\epsilon_t = f_t(\mathbb{P}^*)$.

Proof. [41] Let $\mathbb{P}^* \in \Omega$ be Pareto optimal. Assume \mathbb{P}^* does not solve (8) for some s where $\epsilon_t = f_t(\mathbb{P}^*)$. Then there exists a $\mathbb{P} \in \Omega$ so that $f_s(\mathbb{P}) < f_s(\mathbb{P}^*)$ and $f_t(\mathbb{P}) \leq f_t(\mathbb{P}^*)$, which contradicts the Pareto optimality of \mathbb{P}^* . Therefore, \mathbb{P}^* necessarily solve (8) for any chosen s . Since \mathbb{P}^* is by assumption solution of (8) for every s there is no \mathbb{P} such that $f_s(\mathbb{P}) < f_s(\mathbb{P}^*)$. **Q.E.D.**

The ϵ -constraint method provides Pareto optimal solutions within eOGS if the objective functions are convex. Otherwise, as convexity is never guaranteed in nonlinear nonstationary context, the method provides local noninferior solutions [40] [42].

Consider a parameter variation function

$$v(\psi) = \inf_{\mathbb{P} \in \Omega} \{f_s(\mathbb{P} + \psi_s) \mid f_t(\mathbb{P} + \psi_t) \leq \epsilon_t \forall t\} \quad (9)$$

associated with (8) [41]. Converting (9) into an unconstrained problem by using a Lagrange function we have

$$\min f_s(\mathbb{P} + \psi_s) + \sum_{\forall t, t \neq s} \lambda_t (f_t(\mathbb{P} + \psi_t) - \epsilon_t), \quad (10)$$

where $\lambda_t \forall t$ are Lagrange multipliers. The Karush-Kuhn-Tucker necessary condition for \mathbb{P}^* to be a solution of the ϵ -constraint problem says that there exists $\lambda_t \geq 0$ such that

$$\begin{aligned} \nabla f_s(\mathbb{P}^*) + \sum_{\forall t, t \neq s} \lambda_t \nabla (f_t(\mathbb{P}^*) - \epsilon_t) &= 0, \\ \lambda_t (f_t(\mathbb{P}^*) - \epsilon_t) &= 0, \forall t \neq s. \end{aligned}$$

If the constraint concerning f_t is not active, the corresponding multiplier λ_t is equal to 0 [43].

A systematic and fully autonomous heuristic procedure to change the decision parameters \mathbb{P} of eOGS models is given in Section IV. In this case, the heuristic procedure is the

decision maker, and no human is involved in the process of finding a solution to (7) from (8). In practice, users can also systematically select values for \mathbb{P} manually, and observe if the price that should be paid by the secondary objectives for an improvement of the priority objective, f_s , is acceptable. In this case, the user is the decision maker.

To sum up, once a priority objective, and the bounds for the remaining objectives are chosen, eOGS aims at generating local noninferior solutions in terms of numerical and granular estimation errors, structural compactness, and granular specificity. The eOGS is a heuristic procedure that continuously attempts to solve (8) at each processing step. Approaches for the eOGS heuristics may either involve user preferences in an interactive mode, or be fully autonomous. In both cases, the solutions developed during the processing steps are approximations of the Pareto optimal solutions only. This is because the systems we are concerned with here are nonstationary, and eventually subject to functional changes as well. Details of the eOGS are given in the next section. Section IV summarizes the interactive design, and the fully autonomous operation approaches of the eOGS.

III. EVOLVING OPTIMAL GRANULAR MODELS

Granular evolving models provide numerical and granular estimations of nonlinear and nonstationary functions from data streams [29] [44]. An online learning algorithm builds the model structure and adapts parameters whenever new behaviors are found in the data. eOGS detects concept drift and shift and copes with uncertainty.

The basic working principle is to enclose similar data samples into granules upon which computations are conducted more easily. eOGS models are linguistically interpretable. A granular mapping from a space X to a space Y is a rule which assigns to each granule of X a granule in Y . eOGS models encode granular mappings into If-Then rules. A collection of rules, namely the rule base, is a granular mapping that gives a granular and a numerical output. The granular output is created by an inclusion function P or by the bounds of granules developed in the output space. The numerical output is given by a local algebraic function p associated with each rule. In the following sections, we address the eOGS rule-base structure and learning methodology.

A. From hyper-rectangles to explicit rules

An eOGS granular hyper-rectangle γ^i is defined by spanning α -level sets of membership functions $\Pi_j^i = \mathcal{G}(\mu_j^i, \sigma_j^i)$, $j = 1, \dots, n$, in the space $X_1 \times \dots \times X_n$. Granule γ^i can be explicitly described by the rule

$$\begin{aligned} R^i: & \text{ IF } (l_1^i \leq x_1 \leq L_1^i) \text{ AND } \dots \text{ AND } (l_n^i \leq x_n \leq L_n^i) \\ \text{ THEN } & (u_1^i \leq y_1 \leq U_1^i) \text{ AND } \bar{y}_1^i = p_1^i(x_1, \dots, x_n) \text{ AND} \\ & \dots \\ & (u_m^i \leq y_m \leq U_m^i) \text{ AND } \bar{y}_m^i = p_m^i(x_1, \dots, x_n), \end{aligned}$$

where l_j^i and L_j^i , $j = 1, \dots, n$; $i = 1, \dots, c$, are the j -th lower and upper bounds of the feature x_j according to the i -th rule;

u_k^i and U_k^i , $k = 1, \dots, m$, are the k -th lower and upper bounds of the output y_k ; and p_k^i are numerical affine functions,

$$\bar{y}_k^i = p_k^i(x_1, \dots, x_n) = a_{0k}^i + \sum_{j=1}^n a_{jk}^i x_j. \quad (11)$$

In general, each p_k^i can be of different type and do not need to be linear. The hyper-rectangle γ^i conveys Gaussian membership functions $\Pi_j^i = \mathcal{G}(\mu_j^i, \sigma_j^i)$, $j = 1, \dots, n$, as internal representation – an additional information that summarizes past data samples belonging to γ^i . The modal values μ_j^i and dispersions σ_j^i are captured recursively from the data stream. Moreover, the Recursive Least Squares (RLS) algorithm [12] is used to determine the coefficients a_{jk} , $j = 0, \dots, n$, $k = 1, \dots, m$, of the functions p_k^i whenever the i -th rule is active for an input sample $x = (x_1, \dots, x_n)$.

B. Stiegler and Minimal approaches to create granules

Granules and rules are created either if the current input sample $x^{[h]}$ is not in $[l_j^i, L_j^i] \forall i$ and some j ; or $y^{[h]} \notin [u_k^i, U_k^i] \forall i$ and some k . Notice that the bounds of the underlying intervals depend on the value of α that cuts Π_j^i and Π_k^i . If $\alpha \rightarrow 0^+$, then $[l_j^i, L_j^i] \forall j$ and $[u_k^i, U_k^i] \forall k$ cover the whole input and output spaces, i.e., they form unit n - and m -dimensional hyperboxes, respectively. Contrariwise, if $\alpha \rightarrow 1^-$, then the granule γ^i degenerates in a single point.

An approach to initialize the parameters of a new granule is to consider the Stiegler's standard Gaussian function [45]. The new granule γ^{c+1} has modal value

$$\mu_{j,k}^{c+1} = (x_j, y_k)^{[h]} \forall j, k, \quad (12)$$

and dispersion

$$(\sigma_{j,k}^{c+1})^2 = 1/2\pi \forall j, k. \quad (13)$$

A hyper-rectangle centralized in $(x, y)^{[h]}$ is obtained as the Cartesian product of unidimensional α -cuts.

Based on (3) and (4), granule bounds are given as

$$[l_j^{c+1}, L_j^{c+1}] = x_j^{[h]} \pm \sqrt{-2(\sigma_j^{c+1})^2 \ln(\alpha)} \quad (14)$$

and

$$[u_k^{c+1}, U_k^{c+1}] = y_k^{[h]} \pm \sqrt{-2(\sigma_k^{c+1})^2 \ln(\alpha)}, \quad (15)$$

$0 < \alpha < 1$. Polynomial coefficients are set as

$$a_{0k}^{c+1} = y_k^{[h]}, \forall k; \text{ and } a_{jk}^{c+1} = 0, \forall k, j = 1, \dots, n. \quad (16)$$

Initial granule specificity is obtained from (6).

The Stiegler (maximal) initialization approach may require some iterations for the new granule to shrink and be sized similarly to the other granules. During this period, all new samples within the bounds of the new granule should be used

to adapt its modal value, dispersion and consequent coefficients. Notice that, the RLS algorithm to adapt consequent coefficients requires at least $n+1$ linearly independent samples falling within a granule to achieve consistent values for $n+1$ coefficients. If desired, insufficiently mature granules can be considered apart from the others and temporarily ignored as a contributor to the global estimation. Particularly, this may be an approach to detect outliers as, after a number of iterations, a granule prompted by an outlier will not achieve maturity. Outlier detection is not on the scope of this paper.

An alternative to the Stiegler approach is to initialize the dispersion of the new granule γ^{c+1} using a small (minimal) value for $(\sigma_{j,k}^{c+1})^2$. A default value is suggested as

$$(\sigma_{j,k}^{c+1})^2 = 10^{-2} \quad \forall j, k. \quad (17)$$

In the Minimal approach, a new granule tends to expand over time and develops itself driven by samples resting within its bounds or relatively closer to its center compared to the centers of other granules. Overlapping of granules is reduced; more specific granular mappings are stimulated; and the evolution process and eOGS overall estimates tend to be smoother.

C. Adaptation over time

Real-world data streams change over time. Adapting eOGS rules consists in enlarging or contracting the bounds of granules and simultaneously changing the coefficients of local functions to better fit new behaviors.

Consider samples within a time window, $(x, y)^{[h-\nu]}$, $(x, y)^{[h-\nu+1]}$, ..., $(x, y)^{[h]}$; ν is the length of the window, h is the time step. If a new $x^{[h]}$ fits $[l_j^i, L_j^i] \quad \forall j$ and $y^{[h]}$ fits $[u_k^i, U_k^i] \quad \forall k$ and some i , then the parameters of the local Gaussians Π^i are updated recursively using

$$\mu_{j,k}^i(\text{new}) = \frac{\nu^i \mu_{j,k}^i(\text{old}) + (x_j, y_k)^{[h]}}{\nu^i + 1}, \quad \forall j, k \quad (18)$$

and

$$(\sigma_{j,k}^i(\text{new}))^2 = \beta^i (\sigma_{j,k}^i(\text{old}))^2, \quad \forall j, k, \quad (19)$$

where

$$\beta^i \triangleq \frac{\nu^i (\mu_{j,k}^i - (l_j, u_k)) + \Psi (|\mu_{j,k}^i - (x_j, y_k)^{[h]}|)}{(\nu^i + 1)(\mu_{j,k}^i - (l_j, u_k))},$$

ν^i is the number of samples belonging to γ^i out of the past ν samples; $\Psi = 2$ is a default value. Superior or inferior values for the parameter Ψ compel expansion or contraction of Gaussians over the iterations, and less or more specific granular mappings.

Combining (4), (18) and (19), the α -level set of the updated granule, γ^i , is found as

$$[(l_j^i, u_k^i), (L_j^i, U_k^i)] = \mu_{j,k}^i(\text{new}) \pm \sqrt{-2 \ln(\alpha) (\sigma_{j,k}^i(\text{new}))^2} \quad (20)$$

$\forall j, k$. Moreover the specificity of the underlying granule, $\text{Spa}(\gamma^i)$, is calculated as in (6).

Given a priority objective f_s (numerical or granular estimation error, specificity of the granular mapping or total number of rules), the ϵ -constraint method (8) finds α and Ψ to satisfy the constraints f_t and minimize f_s . The values of α and Ψ are used in (19) and (20) to update the dispersions Π_j^i and Π_k^i and determine the bounds $[l_j^i, L_j^i]$ and $[u_k^i, U_k^i]$. Users manifest their preferences by choosing the main objective and setting admissible values for the constraints $\epsilon_t \forall t$.

Notice that the adaptation procedures described are recursive, i.e., values are accumulated. Therefore, a sample $(x, y)^{[h]}$ can be discarded after being processed. The RLS algorithm [12] adapts the coefficients $a_{j,k}^i \quad \forall j, k$, for samples that activate γ^i . Notice also that the size of the time window, ν , means the lifetime of information within the short-term memory of eOGS models. Models consider the last ν samples only to keep evolution active. If model convergence is expected, then $\nu = \infty$ provides more static and solid granules after a number of iterations. In this case, the model is still flexible to changes since new granules and rules may be created for new situations.

D. Merging granules

Merging neighbor granules, say γ^{i_1} and γ^{i_2} , is helpful to reduce the number of rules and eliminate partially overlapping granules conveying similar information. A variety of decision criteria and heuristic ways of merging granules can be envisioned. The approach we adopted takes the 2-norm of the difference between midpoints of all pairs of granules, i.e.,

$$\arg \min_{i_1, i_2=1, \dots, c; i_1 \neq i_2} \frac{\|\mu^{i_1} - \mu^{i_2}\|}{n}, \quad (21)$$

where n is the number of features, or dimensions of μ . Given the closest granules, γ^{i_1} and γ^{i_2} , if

$$\frac{\|\mu^{i_1} - \mu^{i_2}\|}{n} \leq \omega, \quad (22)$$

being ω a constant or time varying threshold, then the granules are merged. Manual and automatic adaptation approaches for ω according to an objective and constraints will be described in the Methodology section.

The resulting granule, say γ^{c+1} , is formed by $n+m$ Gaussian membership functions (n inputs and m outputs) whose modal values depend on the amount of data samples each of the merged granules used to represent. Formally,

$$\mu_{j,k}^{c+1} = \frac{\nu^{i_1} \mu_{j,k}^{i_1} + \nu^{i_2} \mu_{j,k}^{i_2}}{\nu^{i_1} + \nu^{i_2}}, \quad \forall j, k. \quad (23)$$

If the Gaussians are extended to the n -dimensional space, the midpoint of the new granule is in a straight line that connects the midpoints of the previous granules.

The maximum dispersion approach takes

$$\sigma_{j,k}^{c+1} = \max(\sigma_{j,k}^{i_1}, \sigma_{j,k}^{i_2}), \quad \forall j, k, \quad (24)$$

as the dispersions of the new granule γ^{c+1} . The idea is to preserve information of a variety of samples in the merging region. Given (23) and (24), granule bounds and specificity can be obtained from (20) and (6), respectively, by analogy. Polynomial coefficients are set as

$$a_{jk}^{c+1} = \frac{a_{jk}^{i_1} + a_{jk}^{i_2}}{2}, \forall j, k. \quad (25)$$

E. Deleting granules

Inactive rules for a number of iterations can be deleted. This may mean that the underlying system changed and removing granules is practical to keep the rule base size as compact as possible.

Remember ν^i , the amount of samples that activated γ^i out of the newest ν samples, $x^{[h-\nu]}, \dots, x^{[h]}$. For inactive granules, $\nu^i = 0$, and the respective granule and rule can be removed. Applications in which rare events are more important than the usual ones, e.g., in anomaly detection, or when cyclical drifts are expected, deletion of inactive granules may be inappropriate, and therefore ignored.

F. Interval approximation function

The image of a granule γ^i through a multivariable real function p_k^i is defined as:

$$p_k^i([l_1^i, L_1^i], \dots, [l_n^i, L_n^i]) = \{p_k^i(x_1, \dots, x_n) : x_j \in [l_j^i, L_j^i], j = 1, \dots, n\}.$$

Generally, the image of γ^i through p_k^i is not a hyperrectangle and it may be difficult to obtain in closed form. In practice, p_k^i can be approximated by an inclusion function P_k^i , which is a hyperrectangle in the range of p_k^i , namely $[u_k^i, U_k^i]$.

An interval function P_k^i is called inclusion function if $p_k^i \subseteq P_k^i \forall i$. Inclusion functions are not unique; they depend on how we choose P [46].

An inclusion function P_k^i is optimal if it is the hull of p_k^i . In other words, the optimal inclusion function for p_k^i is the smallest hyperrectangle P_k^{i*} that contains p_k^i . Figure 2 illustrates the idea. P_k^{i*} is unique. Its specificity is the highest possible that guarantees inclusion.

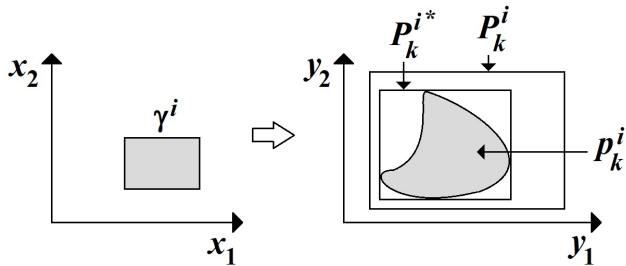


Fig. 2: Image p_k^i of granule γ^i and inclusion functions P_k^i and P_k^{i*}

Let p_k^i be monotonically increasing in $[l_j^i, L_j^i]$, $j = 1, \dots, n$. Then we can obtain p_k^i from

$$p_k^i(x) = [p_k^i(l_1^i, \dots, l_n^i), p_k^i(L_1^i, \dots, L_n^i)].$$

Consequently, for any $x \in \gamma^i$, $p_k^i(x) \subseteq [p_k^i(l^i), p_k^i(L^i)]$. For monotonic decreasing functions we have

$$p_k^i(x) = [p_k^i(L_1^i, \dots, L_n^i), p_k^i(l_1^i, \dots, l_n^i)].$$

An interval function p_k^i is thin if it involves only degenerate interval parameters or, equivalently, numerical parameters. For instance, the interval function

$$p_k^i(x \in \gamma^i) = a_{0k}^i + \sum_{j=1}^n a_{jk}^i [l_j^i, L_j^i] \quad (26)$$

is thin for $(a_{0k}^i, \dots, a_{nk}^i)$ degenerated intervals. When an interval function has at least one interval parameter of nonzero width, it is called thick [47]. We consider thin interval functions only and the linear form (26). If the inclination $a_{jk}^i < 0$, then the bounds should be inverted, i.e. $[L_j^i, l_j^i]$.

G. eOGS learning algorithm

The eOGS learning algorithm is summarized as follows:

```

BEGIN
Set Interactive or Automatic (fully autonomous) mode;
Choose the output smoothness level  $\mathcal{U}$ ;
if (Interactive)
    Set parameters  $\mathbb{P} = \{\alpha, \Psi, \nu, \omega\}$ ;
    Set Stiegler or Minimal approach for the initial dispersions;
end
if (Automatic)
    Choose a priority objective  $f_s$ ;
    Choose admissible values for the remaining objectives  $\epsilon_t \forall t$ ;
end
for  $h = 1, \dots$ 
    Read input data  $x^{[h]}$ ;
    if  $h = 1$ 
        Create granule  $\gamma^1$  and rule  $R^1$  (Eqs. (12)-(17) and (6));
        Provide singular  $\bar{y}^{[h]}$  and granular  $[u^1, U^1]^{[h]}$  predictions;
    else
        Provide singular  $\bar{y}^{[h]}$  and granular  $[u^i, U^i]^{[h]}$  predictions;
        // The actual output  $y^{[h]}$  becomes available;
        if  $x_j^{[h]} \notin [l_j^i, L_j^i] \forall i$  and some  $j$ 
            Create granule  $\gamma^{c+1}$  and rule  $R^{c+1}$  (Eqs. (12)-(17) and (6));
        else
            Adapt active granules  $\gamma^i$  (Eqs. (18)-(20) and (6));
        end
    end
    Delete inactive granules and rules;
    Merge granules and rules (Eqs. (21)-(25), (6), (20));
    if (Automatic)
        Adapt parameters  $\mathbb{P} = \{\alpha, \Psi, \nu, \omega\}$  to minimize  $f_s$  respecting  $\epsilon_t \forall t$ ;
    end
end
END

```

IV. OPTIMAL DESIGN METHODOLOGY

We assume numerical and granular error measures to evaluate system accuracy. Additionally, a guideline on how to set eOGS parameters interactively to achieve particular outcomes is described. In this case, the user is the decision maker. The user and the eOGS algorithm described in Section III are involved in the process of finding a solution for (7) using (8).

A fully autonomous no-human-in-the-loop heuristic procedure to adapt the key eOGS parameters is also outlined. In this case, the heuristic procedure is the decision maker. Based on (8), the eOGS algorithm together with either the interactive or fully autonomous approach given in this section attempt to solve (7) approximately since systems are nonstationary and evolve continuously over time. Finally, a method to smooth the predictions over time is delineated.

A. Numerical and granular error indices

The root mean square error of numerical predictions,

$$RMSE = \frac{1}{H} \sum_{h=1}^H \sqrt{\frac{1}{m} \sum_{k=1}^m (y_k^{[h]} - \bar{y}_k^{[h]})^2}, \quad (27)$$

where H is the current number of iterations and m is the number of output variables, is a measure of accuracy to compare different models for a particular dataset [48].

An error measure for granular predictions that takes into consideration inclusion of the actual values and the narrowness of the enclosure is proposed. The mean granular error is

$$MGE = \frac{1}{kH} \sum_{h=1}^H \sum_{k=1}^m 1 - \mathfrak{z}_k^{[h]} (1 - (U_k^{[h]} - u_k^{[h]})), \quad (28)$$

where

$$\mathfrak{z}_k^{[h]} = \begin{cases} 1 & \text{if } y_k^{[h]} \in [u_k, U_k]^{[h]} \\ 0 & \text{otherwise.} \end{cases} \quad (29)$$

If the actual output $y_k^{[h]}$ is out of the prediction bounds $[u_k, U_k]^{[h]}$, then $\mathfrak{z}_k^{[h]} = 0$, giving the maximum MGE . The MGE index is less than 1 only if the granular prediction encloses $y_k^{[h]}$. The narrower is the bounds $[u_k, U_k]^{[h]}$ that encloses $y_k^{[h]}$, the smaller the MGE .

B. Interactive design

A guideline to set eOGS parameters \mathbb{P} interactively is given in this section. We assume default parameters as first attempt. They are $\alpha = 0.1$, $\Psi = 2$, $\nu = 500$ and $\omega = 0.01$. If a prior set of samples is available, a data stream can be simulated sometimes in a matter of seconds and, perhaps, more appropriate parameters can be found from the default values. Convenient boundary values are $0.01 \leq \alpha \leq 1$; $1.5 \leq \Psi \leq 2.5$; $\nu \geq n$, i.e., greater than the number of features; $0.01 \leq \omega \leq 0.05$.

Parameter α (the cut point of the Gaussians), and the initial dispersions when granules are created are the most influential eOGS parameters. The Minimal approach to initialize dispersions ($\sigma_{jk}^{2[0]} = 0.01 \forall j, k$) is recommended if consecutive samples of a stream are highly correlated in time, e.g., for time series in which the inputs are lagged values of the outputs; and continuous dynamical systems, in which sensor measures tend to change smoothly over time. Otherwise, if samples are not time indexed, but collected independently,

or for switched systems or systems subject to anomalies, the Stiegler approach ($\sigma_{jk}^{2[0]} = 1/2\pi \forall j, k$) should be employed. In the former case, the dispersions will automatically expand along the iterations, whereas in the latter case, dispersions will shrink to an appropriate value.

After finding a proper value for α interactively (according to a tradeoff among objectives or focusing on the best value of a specific objective) using the small set of samples, parameters ω , ν and Ψ can be chosen differently from the default values as a way to fine tune the results.

Parameter ω is related to the minimum distance between granules to allow their merging. Higher values of ω tend to give a more compact rule base. Generally, the price is a higher granular error and a smaller specificity of the granular mapping. Then, parameter ν , the inactivity threshold for deleting granules, can be set to higher values if a greater memory from the past is desired. Smaller values of ν are not recommended. Often the numerical estimation error can improve with a smaller ν , but this means that the most recent granules are tracking the current environment quickly; however, rules representing the past are being deleted. Parameter Ψ , used in the adaptation of the Gaussians dispersions, can be reduced or increased to force the Gaussians to shrink or expand faster over the iterations. Values of Ψ different from the default one may temporarily distort the true variance of the data within a granule to accelerate the transient stage, and this may improve marginally the average estimation errors.

C. Fully autonomous operation

A typical form of the optimization problem (8) takes the numerical estimation error as primary objective, i.e.,

$$\begin{aligned} F(\mathbb{P}) &= \min \quad RMSE \\ \text{s.t.} \quad MGE &\leq \epsilon_1 \\ c &\leq \epsilon_2 \\ Spa(\gamma) &\geq \epsilon_3 \end{aligned} \quad (30)$$

In general, the $RMSE$ can be a constraint while MGE , c and $Spa(\gamma)$ may be chosen as primary objective. If any constraint is violated, the algorithm should respond as soon as possible by means of the parameters $\mathbb{P} = \{\alpha, \Psi, \nu, \omega\}$.

During the online operation of the eOGS algorithm if a constraint is violated, parameters are changed as follows:

- If MGE is greater than ϵ_1 , then (i) α is stepped up; and (ii) σ^2 is stepped down for newly created granules, but kept within the Minimal and Stiegler range of values, i.e. $0.01 \leq \sigma^2 \leq 1/2\pi$.
- If c is greater than ϵ_2 , then (i) α is stepped down; and (ii) ω is stepped up. After some iterations, when c assumes a feasible value, ω is reset to default.
- If $Spa(\gamma)$ is less than ϵ_3 , then (i) α is stepped up; and (ii) Ψ is stepped down. After some iterations, when $Spa(\gamma)$ assumes a feasible value, Ψ is reset to default.

Additionally,

- If $RMSE$ is greater than ϵ_4 , then (i) α is stepped down; and (ii) σ^2 is stepped up for newly created granules, but kept within the Minimal and Stiegler range of values, i.e. $0.01 \leq \sigma^2 \leq 1/2\pi$.

If the values ϵ_t are set in an unrealistic way, parameters \mathbb{P} reach a limit and the best possible solution is given. Convenient boundary values were described in the last section as $0.01 \leq \alpha \leq 1$; $1.5 \leq \Psi \leq 2.5$; $\nu \geq n$, i.e., greater than the number of features; $0.01 \leq \omega \leq 0.05$.

While the eOGS algorithm – Section III-G – attempts to minimize the $RMSE$ and MGE indices, and the number of rules, c ; and to maximize the specificity, $Spa(\gamma)$, the procedure described in this section monitors the constraints. If a constraint is violated, then parameters \mathbb{P} are updated to reestablish the feasibility of the optimization problem.

D. Smoothing the predictions

In case the actual input and output data change smoothly over time we may introduce smoothness on the eOGS predictions. For example, (i) if the input data stream is available from measures of a continuous dynamical system and the sampling time is sufficiently small compared to the dominant time constant of the system, then only gradual changes are expected as model output; (ii) if differential or difference equations are used to produce a data sequence by numerical simulation, then smooth the predictions may be interesting; (iii) if model inputs consist of lagged values of a univariate time series, then smooth predictions are appealing; (iv) if model outputs will be directly used in the real-world to feed actuators, then smoothness may increase their lifetime.

By contrary, if data samples are available as a stream, but are independent of each other, then smoothed predictions may be detrimental. For example, (i) multivariate dynamical systems and time series with exogenous inputs should disregard smoothness and accept predictions $\bar{y}^{[h]}$, $h = 1, \dots$, as given by the model; (ii) switched systems and time series in which discontinuities are expected should ignore smoothness; (iii) in anomaly detection applications, shifts are the most interesting events and should not be smoothed.

Let $\bar{U} \in \mathbb{N}^+$ be the smoothness level of the predictions. If $\bar{U} = 1$, then $\bar{y}^{[h]}$ is given as model numerical estimation. For $\bar{U} > 1$, the smoothed estimation results from

$$\bar{y}_{smooth}^{[h]} = y^{[h-1]} + \frac{\bar{y}^{[h]} - y^{[h-\bar{U}]}}{\bar{U}}. \quad (31)$$

Increasing \bar{U} gradually reduces shifts or jumps on the model estimations. Therefore, if the input samples from the underlying application are time independent, then \bar{U} should be set to 1 since model outputs should also be time independent. In the latter case, jumps or shifts are desired.

Lower and upper prediction bounds (granular estimations) are smoothed similarly, but using past predictions only. They are obtained from

$$u_{smooth}^{[h]} = u^{[h-1]} + \frac{u^{[h]} - u^{[h-\bar{U}]}}{\bar{U}} \quad (32)$$

and

$$U_{smooth}^{[h]} = U^{[h-1]} + \frac{U^{[h]} - U^{[h-\bar{U}]}}{\bar{U}}, \quad (33)$$

respectively, for $\bar{U} > 1$.

V. APPLICATION EXAMPLES

A. Mackey-Glass time series prediction

The Mackey-Glass equation,

$$\frac{dx}{dt} = \frac{ax^{[t-\tau]}}{1 + (x^{[t-\tau]})^c} - bx^{[t]}, \quad a, b, c > 0, \quad (34)$$

is a time delay differential equation that behaves chaotically or periodically depending on the values of its parameters and on the time delay τ . The equation may represent a feedback control system. We assume that Eq. (34) is unknown. A time series is produced by numerical integration using the fourth-order Runge-Kutta method [49].

From the data we aim to construct a fuzzy granular model f of the Mackey-Glass system to map previous values on a future value, i.e., eOGS plays the role of f in

$$x^{[h+\xi]} = f(x^{[h]}, x^{[h-\Delta]}, \dots, x^{[h-D\Delta]}). \quad (35)$$

Similar to many studies on this series we choose $\xi = 85$, $\Delta = 6$, $D = 3$; and $a = 0.2$, $b = 0.1$, $c = 10$, $\tau = 17$ for the parameters of the Mackey-Glass equation to generate data samples. Samples are presented sequentially to the eOGS algorithm, one at a time. The model is built from scratch, with no rules nor pre-training. We resort to the sample-per-sample testing-before-training approach from $h = 105$ to $h = 11898$. In other words, at each iteration: (i) a prediction is given; (ii) the actual output becomes available; and (iii) adaptation of model parameters and structure are done if necessary – see pseudocode in Section III.

1) *Interactive Design*: We start with the default parameters $\alpha = 0.1$, $\Psi = 2$, $\nu = 500$, $\omega = 0.01$ and the Minimal approach to initialize dispersions.

As the data come from a continuous dynamical system, it is expected that the values change smoothly over time. Therefore, firstly, the smoothness constant \bar{U} is increased. The results obtained for different eOGS constructs considering the numerical, $RMSE$, and granular, MGE , indices; number of rules c ; and specificity of the granular mapping $Spa(\gamma)$ are shown in Table I. The table also shows the total CPU time to process 11793 data samples using a i7-8550U CPU dual-core 1.88-1.99GHz processor with 8GB of RAM.

From Table I we notice that both error indices become quite low for $\bar{U} = 30$, especially the $RMSE$. The error indices saturate for higher values of \bar{U} . This is the usual behavior for univariate time series. Model structure and granule specificity are naturally the same due to the smoothness introduced on the numerical and granular predictions.

The interactive setting of the eOGS algorithm could be considered terminated at this point. Nonetheless, the next step is to change the α -level as an attempt to improve the MGE

TABLE I: eOGS results for the Mackey-Glass time series: increasing the smoothness of the predictions

\bar{U}	<i>RMSE</i>	<i>MGE</i>	<i>c</i>	<i>Spa</i> (γ)	Time(s)
1	0.0452	0.4579	11	0.7926	7.1
5	0.0082	0.3998	11	0.7926	7.1
10	0.0046	0.3834	11	0.7926	7.2
15	0.0037	0.3859	11	0.7926	7.3
20	0.0034	0.3638	11	0.7926	7.5
30	0.0032	0.3578	11	0.7926	7.8
40	0.0033	0.3667	11	0.7926	8.0
50	0.0033	0.3863	11	0.7926	8.2

index and the specificity of the granular mapping. Table II shows the results for different α -level sets.

TABLE II: eOGS results for the Mackey-Glass time series: increasing the α level

α	<i>RMSE</i>	<i>MGE</i>	<i>c</i>	<i>Spa</i> (γ)	Time(s)
0.1	0.0032	0.3578	11	0.7926	7.8
0.2	0.0032	0.3714	13	0.7913	8.4
0.3	0.0032	0.3453	13	0.8023	8.4
0.4	0.0032	0.3080	13	0.8167	8.4
0.5	0.0032	0.3244	13	0.8185	8.5
0.6	0.0032	0.3215	13	0.8221	8.2
0.7	0.0032	0.3281	13	0.8259	8.5

Notice that $\alpha = 0.4$ preserved the *RMSE* while provided the minimal *MGE* and a slight improvement of the specificity *Spa*(γ) at the price of two additional rules, $c = 13$. An almost exact fuzzy granular model of the Mackey-Glass dynamical system and a narrow enclosure of the time series were achieved. Adjustments can be considered concluded with the choice of a model such as those produced using $\alpha = 0.1, 0.4$ or 0.7 . We choose $\alpha = 0.4$.

The merging related parameter, ω , can be increased to reduce the number of rules c and processing time. As merged granules are less specific than its precursors, the price of a more compact rule base is generally paid with a worse granular estimation and granule specificity.

Remark 1. Reducing the length of the time window, ν , to delete granules more often and obtain a more compact rule base should be avoided as long and medium-term memory may be lost. Larger time windows, ν , can be attempted for a slightly superior *RMSE* and *MGE*.

Table III shows the eOGS results for different values of ω .

TABLE III: eOGS results for the Mackey-Glass time series: increasing ω for a more compact rule base

ω	<i>RMSE</i>	<i>MGE</i>	<i>c</i>	<i>Spa</i> (γ)	Time(s)
0.01	0.0032	0.3080	13	0.8167	8.4
0.02	0.0032	0.3488	10	0.8006	7.5
0.03	0.0032	0.3523	9	0.7783	6.7

In Table III, the *RMSE* is maintained with the increase of ω . Naturally, as a smaller number of rules for higher values of ω are being used, the maintenance of such error index is only possible because the granules are focused on recent occurrences and being slide to track the time series. Therefore, eOGS memory of past information is kept in the model, but in a more abstract way, which may or may not be interesting.

We opt for the model with the smaller *MGE* and 13 rules as solution. Such model is obtained using the default parameters, except for $\bar{U} = 30$ and $\alpha = 0.4$. The results are shown in the first row of Table III.

2) *Autonomous operation:* Let the numerical prediction error be the primary objective and consider constraints for the remaining objectives. The optimization problem is given as

$$\begin{aligned}
 F(\mathbb{P}) &= \min \quad RMSE \\
 s.t. \quad &MGE \leq 0.35 \\
 &c \leq 10 \\
 &Spa(\gamma) \geq 0.8
 \end{aligned} \tag{36}$$

Using the default parameters \mathbb{P} initially, that is, $\alpha = 0.1, \Psi = 2, \nu = 500, \omega = 0.01$, and Minimal initial dispersions, the autonomous eOGS algorithm searches for appropriate parameters \mathbb{P} to satisfy the constraints and minimize the *RMSE* over the iterations. Figure 3 shows the numerical and granular predictions and the evolution of the number of rules, specificity, error indices and α level.

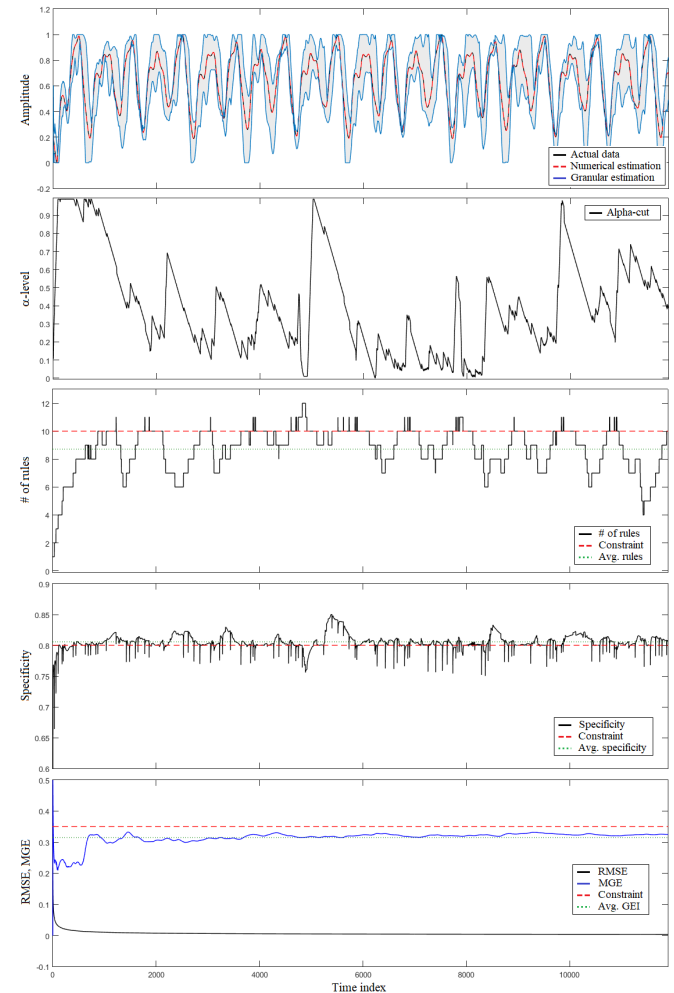


Fig. 3: Numerical and granular eOGS prediction, and evolution of α , number of rules, specificity, and error indices

From Fig. 3 we observe a quite low $RMSE$ index, 0.0032, and that the numerical prediction almost coincides with the actual data. The granular enclosure ($MGE = 0.3253$) gives a path for the time series and a range of possibilities that may be useful to support decision making in different ways depending on the purpose of the model. For example, in case f represents the production of white blood cells to defend the human body against pathogens as in Glass and Mackey [50], eOGS granular prediction may represent the range of normal activity of hematopoietic stem cells. Regarding the number of rules ($c = 10$) and specificity ($Spa(\gamma) = 0.8057$), the online incremental algorithm responded as soon as possible when their values overcame the allowable values.

Lastly, in Table IV we compare the eOGS results shown in Fig. 3 with the performance of other evolving models in relation to the $RMSE$ and number of local models c . The effectiveness of the eOGS approach in predicting chaotic time series without prior knowledge about the data is verified. Different from other models, eOGS guarantees a level of specificity of the granules and a level of compactness of the rule base. In addition, it gives upper and lower boundaries; an envelope of the actual values.

TABLE IV: Mackey-Glass time series: comparative results

Model	Reference	c	$RMSE$
EFuNN	[51]	193	0.0822
RAN	[52]	113	0.0802
eTS	[53]	9	0.0799
xTS	[53]	10	0.0711
DENFIS	[51]	58	0.0593
Neural Gas	[54]	1000	0.0133
IBeM	[55]	98	0.0126
FBeM	[29]	33	0.0122
eOGS	This paper	10	0.0032

B. Multivariate data stream

As an example of application in which samples are not correlated in time we choose the Concrete Compressive Strength data set, available at the UCI Machine Learning Repository. It consists of 1030 samples, 8 inputs and an output. Concrete ingredients and age of the mixture are the independent variables of the compression function. Ingredients include cement, blast furnace slag, fly ash, water, superplasticizer, coarse and fine aggregate [56]. Compressive strength is the capacity of a material to withstand axially-directed pushing forces. Compressive tests measure how well concrete holds up to the compressive pressures around it. When the limit of compressive strength is reached, materials are crushed. When building with concrete, it is important to know whether it can bear the compressive forces [57].

Samples are disposed sequentially to the eOGS algorithm. The model is built from an empty rule base. At each iteration: (i) a prediction is given; (ii) the actual output becomes available; and (iii) adaptation of model parameters and structure are done if necessary.

1) *Interactive Design*: Default parameters $\alpha = 0.1$, $\Psi = 2$, $\nu = 500$, $\omega = 0.01$ and the Stiegler approach to initialize dispersions are employed. Smoothness on the predictions is

needless as the samples are uncorrelated, i.e., as the independent data may be completely different in consecutive iterations, jumps on the predictions are expected, therefore $\bar{U} = 1$.

First, we change the α -level with focus on the minimal numerical error. The results obtained for different eOGS constructs considering the numerical, $RMSE$, and granular, MGE , indices; number of rules c ; and specificity of the granular mapping $Spa(\gamma)$ are shown in Table V. The total CPU time to process 1030 samples considers a i7-8550U CPU dual-core 1.88-1.99GHz processor with 8GB of RAM.

TABLE V: eOGS results for the Compressive Strength: changing the α level

α	$RMSE$	MGE	c	$Spa(\gamma)$	Time(s)
0.9	0.1526	0.5346	79	0.5518	3.9
0.8	0.1426	0.5984	47	0.4820	2.7
0.7	0.1337	0.6545	39	0.4460	2.3
0.6	0.1295	0.7052	39	0.4290	2.1
0.5	0.1327	0.7247	30	0.3938	1.8
0.4	0.1334	0.7408	27	0.3767	1.7
0.3	0.1282	0.7576	21	0.3507	1.4
0.2	0.1220	0.7878	23	0.3345	1.4
0.1	0.1256	0.8063	20	0.3042	1.2
0.05	0.1242	0.8151	20	0.2941	1.2
0.01	0.1198	0.8386	17	0.2266	1.1

Notice that the numerical and granular outcomes are conflicting. We can accept an intermediate condition or search for the best set of parameters with focus on the $RMSE$ or MGE . In case we opt for a smaller granular error, we can proceed with $\alpha = 0.9$ as higher values of α will produce a substantial number of rules and granules c .

Next, using $\alpha = 0.01$, we change ν and ω for fine tuning the model. Reducing ν may cause some ‘somewhat’ inactive rules to be deleted and a better $RMSE$, but the memory of the resulting model becomes smaller. Increasing ω may also cause excessive rule merging and therefore lost of model memory capacity. Care should be taken to avoid a great reduction of the number of rules c . We search for a proper ν and ω with focus on the $RMSE$. Table VI shows the results.

TABLE VI: eOGS results for the Compressive Strength: changing ν , ω and Ψ for a smaller numerical error

$\omega \uparrow$	ν	Ψ	$RMSE$	MGE	c	$Spa(\gamma)$	Time(s)
0.01	500	2	0.1198	0.8386	17	0.2266	1.1
0.015	500	2	0.1180	0.8434	10	0.1907	0.8
0.02	500	2	0.1173	0.8815	6	0.1561	0.8
0.025	500	2	0.1157	0.9040	6	0.1350	0.8
0.03	500	2	0.1122	0.8945	4	0.0757	0.7
ω	$\nu \uparrow$	Ψ	$RMSE$	MGE	c	$Spa(\gamma)$	Time(s)
0.025	600	2	0.1157	0.9162	7	0.1428	0.8
0.025	700	2	0.1158	0.9192	7	0.1436	0.9
0.025	800	2	0.1152	0.9194	7	0.1433	0.8
ω	ν	$\Psi \uparrow$	$RMSE$	MGE	c	$Spa(\gamma)$	Time(s)
0.025	800	2.1	0.1136	0.9003	6	0.1334	0.8
0.025	800	2.2	0.1113	0.8813	5	0.0881	0.8
0.025	800	2.3	0.1096	0.9035	6	0.0929	0.8
0.025	800	2.4	0.1095	0.9142	6	0.0764	0.8
0.025	800	2.5	0.1101	0.9262	5	0.0586	0.8

Notice that the parameters that produced the most accurate eOGS model according to changes of ω ($RMSE = 0.1122$) keeps only 4 rules in its rule base. Important rules representing past occurrences were merged. Frequently, we want

the model to maintain information from the past. Therefore, we proceeded with $\omega = 0.0025$. Increasing ν to 800 yielded the best $RMSE$. Next, increasing Ψ to enlarge the Gaussian distributions generated an interesting 6-rule eOGS model with $RMSE = 0.1095$, see the second last row of Table VI. This model was chosen as solution of the interactive design.

2) *Autonomous operation:* Let the numerical error be the primary objective and admit the remaining objectives as constraints. The optimization problem is

$$\begin{aligned} F(\mathbb{P}) = \min \quad & RMSE \\ \text{s.t.} \quad & MGE \leq 0.7 \\ & c \leq 15 \\ & Spa(\gamma) \geq 0.4 \end{aligned} \quad (37)$$

Compared to the interactive design, we are now forcing the granular error, MGE , and granular specificity, $Spa(\gamma)$, to more audacious values. $MGE = 0.9142$ and $Spa(\gamma) = 0.0764$ should now be less than 0.7 and more than 0.4, respectively. A larger number of rules and a higher $RMSE$ are therefore expected for such problem formulation.

Figure 4 shows the numerical and granular approximation of the concrete compressive strength function and the evolution of the number of rules, specificity, error indices and α level. The figure also shows the final Gaussians covering the output space. The upper right plot expands the approximation of the upper left plot in the range [935, 1030] for clarity.

The numerical and granular approximations give a value of compressive strength and a range of values in its neighborhood. Granular approximation may enhance model acceptability and be made tighter if we accept a larger number of rules. Different from the Mackey-Glass problem, the granular enclosure of the compression function approaches the actual data more carefully ($MGE = 0.6894$) due to lack of smoothness and time correlation between consecutive samples. Although the $RMSE$ is 0.1310, notice that the instantaneous square error reduces over time. This means that the model is achieving maturity; the system is effectively learning on the fly, and the accumulated $RMSE$ index tends to become smaller if more data samples are shown.

The evolution of the number of rules also demonstrates a level of stability of the eOGS model structure, especially after 420 iterations. If the algorithm creates rules beyond the allowable value, parameters \mathbb{P} are changed – see, e.g., the variation of the α -cut over time. The final number of rules was $c = 14$. Similarly, the average granular mapping specificity ($Spa(\gamma) = 0.4217$) is kept above the allowable value. The final output Gaussians cover the output space as illustrated. Highly overlapped output membership functions are common as the merging procedure depends also on the similarity of Gaussians in the universe of the input variables. For example, two granules that are not significantly overlapped in the input space can be mapped to close regions in the output space. Their local linear functions may have similar slope or not. The total CPU time to process 1030 samples was 0.96 seconds, which indicates that the algorithm can operate in real-time considering high-frequency data streams. In general, as

a decision making support system, results may recommend changing ingredients mix ratio or adding special hardeners to the concrete compound.

Comparisons of eOGS with alternative evolving methods are reported in Table VII. The results for eOGS are those obtained in the interactive design with focus on the $RMSE$, similar to the remaining algorithms. The fully autonomous eOGS algorithm described in this section equilibrates multiple numerical and granular objectives and is therefore dependent on what one expects about the granular error and specificity.

TABLE VII: Concrete compressive strength prediction: comparative results

Model	Reference	c	$RMSE$
DENFIS	[51]	4	0.1240
IBeM	[55]	4	0.1178
xTS	[53]	17	0.1148
eGNN	[7]	4	0.1133
eOGS	This paper	6	0.1095
eTS	[53]	21	0.1078

Table VII shows that eOGS is very competitive in this application based on the relation numerical accuracy/compactness. It reached a 0.1095 $RMSE$ using an average of 6 rules. eTS (evolving Takagi-Sugeno) has been the most accurate method, but it requires a larger rule base and gives no prediction boundaries and no guarantees on the specificity of the local models. When using only 4 local models – similar to DENFIS (Dynamic evolving neuro-fuzzy inference system), IBeM (Interval-based evolving modeling) and eGNN (evolving Granular Neural Network) – eOGS provided a 0.1122 $RMSE$, as shown in the fifth row of Table VI. This result is also marginally superior to those of the other 4-rule evolving models. Finally, we opt for the 6-rule eOGS model aiming at a greater memory of past information. The advantages of eOGS over the other approaches reside in its granular enclosure provided as additional information to assist decision making, the possibility of trading-off multiple objectives, and a slightly superior accuracy/compactness relation in the underlying multivariate data stream application.

VI. CONCLUSION

We described an online incremental learning method for the optimal design of granular rule-based models from data streams. We presented a multiobjective formulation to tradeoff information specificity, model compactness, and numerical and granular estimations. We provided assurance that certain conditions can be expressed and fulfilled. Adaptive α -level sets over Gaussian fuzzy membership functions was used to set model granularity and operate with hyper-rectangular forms of granules in nonstationary environment.

The resulting models are formed in a formal and systematic fashion. In general, models can be applied to time series prediction, nonlinear function approximation or dynamic systems identification. Accurate estimates and enclosures are given by linear piecewise and inclusion functions related to optimal granular mappings. This was demonstrated empirically using data from a chaotic nonlinear system, and from a multivariable system whose samples are time independent. We also gave an

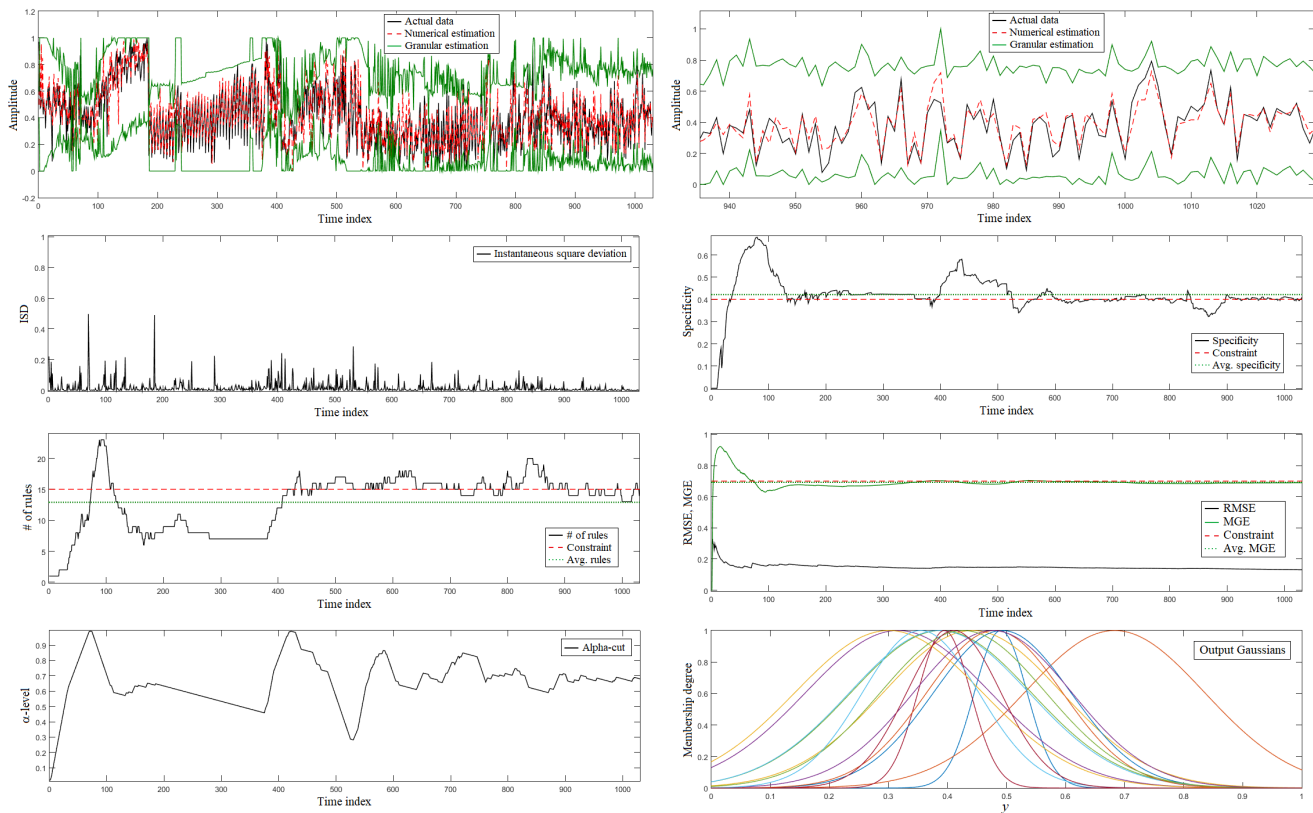


Fig. 4: Numerical and granular eOGS prediction; evolution of α , number of rules, granular mapping specificity, and error indices; and final output Gaussians for the Concrete Compressive Strength problem

interactive guideline and a fully-autonomous way to develop granular systems.

The advantages of the evolving optimal granular approach over any other evolving approach in the literature stands for its granular enclosure, given as additional information to assist decision making, and the possibility of trading-off multiple objectives. Particularly, the proposed method outperformed other evolving methods in the sense of the accuracy/compactness relation in the chaotic and multivariate problems studied.

REFERENCES

- [1] Nguyen, H.-L.; Woon, Y.-K.; Ng, W.-K. "A survey on data stream clustering and classification." *Knowl Inf Syst*, 45(3), pp. 535-569, 2015.
- [2] Leite, D. *Evolving Granular Systems*. PhD thesis, University of Campinas, School of Electrical and Computer Engineering, 2012.
- [3] Dovžan, D.; Logar, V.; Škrjanc, I. "Implementation of an evolving fuzzy model (eFuMo) in a monitoring system for a waste-water treatment process." *IEEE Trans Fuzzy Syst*, 23(5), pp. 1761-1776, 2015.
- [4] Sayed-Mouchaweh, M.; Lughofer, E. (Eds.) *Learning in Non-Stationary Environments: Methods and Applications*. Springer: New York, 2012.
- [5] Silva, S.; Costa, P.; Gouvea, M.; Lacerda, A.; Alves, F.; Leite, D. "High impedance fault detection in power distribution systems using wavelet transform and evolving neural network." *Electr Pow Syst Res*, 154, pp. 474-483, 2018.
- [6] Maciel, L.; Ballini, R.; Gomide, F. "Evolving granular analytics for interval time series forecasting." *Granular Comput*, 1(4), pp. 213-224, 2016.
- [7] Leite, D.; Costa, P.; Gomide, F. "Evolving granular neural networks from fuzzy data streams." *Neural Networks*, 38, pp. 1-16, 2013.
- [8] Soares, E.; Costa, P.; Costa, B.; Leite, D. "Ensemble of evolving data clouds and fuzzy models for weather time series prediction." *Appl Soft Comput*, 64, pp. 445-453, 2018.
- [9] Tomažič, S.; Dovžan, D.; Škrjanc, I. "Confidence-interval fuzzy model-based indoor localization." *IEEE Trans Ind Electron*, 66(3), pp. 2015-2024, 2019.
- [10] Domingos, D.; Camargo, G.; Gomide, F. "Autonomous fuzzy control and navigation of quadcopters." *IFAC-PapersOnLine*, 49(5), pp. 73-78, 2016.
- [11] Zdešar, A.; Dovžan, D.; Škrjanc, I. "Self-tuning of 2DOF control based on evolving fuzzy model." *Appl Soft Comput*, 19, pp. 403-418, 2014.
- [12] Leite, D.; Palhares, R.; Campos, V.; Gomide, F. "Evolving granular fuzzy model-based control of nonlinear dynamic systems." *IEEE Trans Fuzzy Syst*, 23(4), pp. 923-938, 2015.
- [13] Zadeh, L. "Some reflections on soft computing, granular computing, and their role in the conception, design, and utilization of information/intelligent systems." *Soft Computing*, 2(1), pp. 23-25, 1998.
- [14] Yao, J. T.; Vasilakos, A. V.; Pedrycz, W. "Granular computing: Perspectives and challenges." *IEEE Trans Cybern*, 43(6), pp. 1977-1989, 2013.
- [15] Yao, Y. "A triarchic theory of granular computing." *Granular Computing*, 1(2), pp. 145-157, 2016.
- [16] Zadeh, L. A. "Towards a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic." *Fuzzy Sets Syst*, 90(2), pp. 111-127, 1997.
- [17] Bargiela, A.; Pedrycz, W. "Granular mappings." *IEEE Trans Syst Man Cybern - Part A*, 35(2), pp. 292-297, 2005.
- [18] Yager, R. "Default knowledge and measures of specificity." *Info Sci*, 61(1-2), pp. 1-44, 1992.
- [19] Yager, R. "Measures of specificity over continuous spaces under similarity relations." *Fuzzy Sets Syst*, 159(17), pp. 2193-2210, 2008.
- [20] Pedrycz, W.; Homenda, W. "Building the fundamentals of granular computing: A principle of justifiable granularity." *Appl Soft Comput*, 13(10), pp. 4209-4218, 2013.
- [21] Zadeh, L. A. "Generalized theory of uncertainty (GTU) - principal concepts and ideas." *Comput Stat Data An*, 51(1), pp. 15-46, 2006.
- [22] Škrjanc, I.; Iglesias, J.; Sanchis, A.; Leite, D.; Lughofer, E.; Gomide, F. "Evolving fuzzy and neuro-fuzzy approaches in clustering, regression, identification, and classification: A survey." *Info Sci*, 490, pp. 344-368, 2019.

- [23] Angelov, P. *Autonomous Learning Systems: From Data Streams to Knowledge in Real-time*. Wiley, 2013.
- [24] Lughofer, E. *Evolving Fuzzy Systems: Methodologies, Advanced Concepts and Applications*. Springer: Verlag Berlin Heidelberg, 2011.
- [25] Kasabov, N. *Evolving Connectionist Systems: The Knowledge Engineering Approach*. Springer, 2 ed., 2007.
- [26] Dovzan, D.; Skrjanc, I. "Recursive fuzzy c-means clustering for recursive fuzzy identification of time-varying processes." *ISA Transactions*, 50(2), pp. 159-169, 2011.
- [27] Rubio, J. J.; Bouchachia, A. "MSAFIS: an evolving fuzzy inference system." *Soft Computing*, 21(9), pp. 2357-2366, 2017.
- [28] Lughofer, E.; Pratama, M.; Skrjanc, I. "Incremental rule splitting in generalized evolving fuzzy systems for autonomous drift compensation." *IEEE Trans Fuzzy Syst*, 26(4), pp. 1854-1865, 2018.
- [29] Leite, D.; Ballini, R.; Costa, P.; Gomide, F. "Evolving fuzzy granular modeling from nonstationary fuzzy data streams." *Evolving Systems*, 3(2), pp. 65-79, 2012.
- [30] Angelov, P.; Sadeghi-Tehran, P.; Ramezani, R. "An approach to automatic real-time novelty detection, object identification, and tracking in video streams based on recursive density estimation and evolving Takagi-Sugeno fuzzy systems." *Int J Intell Syst*, 26(3), pp. 189-205, 2011.
- [31] Precup, R.-E.; Teban, T.-A.; Albu, A.; Szedlak-Stinean, A.-I.; Bojan-Dragos, C.-A. "Experiments in incremental online identification of fuzzy models of finger dynamics." *Romanian J of Information Science and Technology*, 21(4), pp. 358-376, 2018.
- [32] Castillo, O.; Melin, P. "Hybrid intelligent systems for time series prediction using neural networks, fuzzy logic, and fractal theory." *IEEE Trans Neural Netw*, 13(6), pp. 1395-1408, 2002.
- [33] Pedrycz, W.; Al-Hmouz, R.; Morfeq, A.; Balamash, A. "The design of free structure granular mappings: the use of the principle of justifiable granularity." *IEEE Trans Cybern*, 43(6), pp. 2105-2113, 2013.
- [34] Castillo, O.; Melin, P.; Pedrycz, W. "Design of interval type-2 fuzzy models through optimal granularity allocation." *Appl Soft Comput*, 11(8), pp. 5590-5601, 2011.
- [35] Škrjanc, I. "Fuzzy confidence interval for pH titration curve". *Applied Mathematical Modelling*, 35(8), pp. 4083-4090, 2011.
- [36] Liu, S.; Pedrycz, W.; Gacek, A.; Dai, Y. "Development of information granules of higher type and their applications to granular models of time series." *Eng Appl of AI*, 71, pp. 60-72, 2018.
- [37] Halmos, P. R. *Measure Theory*. Springer-Verlag, Berlin, 1974.
- [38] Moore, R. E.; Kearfott, R. B.; Cloud, M. J. *Introduction to Interval Analysis*. SIAM, Philadelphia, 2009.
- [39] Mermillod, M.; Bugaiska, A.; Bonin, P. "The stability-plasticity dilemma: investigating the continuum from catastrophic forgetting to age-limited learning effects." *Front Psychol*, 4(504), pp. 1-3, 2013.
- [40] Haimes, Y. Y.; Lasdon, L. S.; Wismer, D. A. "On a bicriterion formulation of the problems of integrated systems identification and system optimization." *IEEE Trans Syst Man Cybern - SMC-1*, pp. 296-297, 1971.
- [41] Miettinen, K. M. *Nonlinear Multiobjective Optimization*. Kluwer Academic Pub., Boston, 1999.
- [42] Haimes, Y. Y.; Chankong, V. "Kuhn-Tucker multipliers as trade-offs in multiobjective decision-making analysis." *Automatica*, 15(1), pp. 59-72, 1979.
- [43] Bazaraa, M. S.; Sherali, H. D.; Shetty, C. M. *Nonlinear Programming: Theory and Algorithms*. Wiley, 3rd ed. 2006.
- [44] Leite, D.; Costa Jr., P.; Gomide, F. "Granular approach for evolving system modeling." In: E. Hillermeier, et al. *Lecture Notes in Artificial Intelligence*, 6178, pp. 340-349, Springer, Heidelberg, 2010.
- [45] Stiegler, S. M. "A modest proposal: A new standard for the normal." *The American Statistician*, 36-2, JSTOR, 1982.
- [46] Jaulin, L.; Keiffer, M.; Didrit, O.; Walter, E. *Applied Interval Analysis*. Springer-Verlag, London, 2001.
- [47] Hansen, E. R.; Walster, G. W. *Global Optimization using Interval Analysis*. Marcel Dekker, New York - Basel, 2nd edition, 2004.
- [48] Hyndman, R. J.; Koehler, A. B. "Another look at measures of forecast accuracy." *International Journal of Forecasting*, 22(4), pp. 679-688, 2006.
- [49] Devries, P. L.; Hasbun, J. E. *A First Course in Computational Physics*. 2nd edition. Jones and Bartlett Publishers, 2011.
- [50] Mackey, M.; Glass, L. "Oscillation and chaos in physiological control systems." *Science*, 197(4300), pp. 287-289, 1977.
- [51] Kasabov, N. *Evolving Connectionist Systems: The Knowledge Engineering Approach*. Springer-Verlag London, 2nd edition, 2007.
- [52] J. Platt, "A resource-allocating network for function interpolation." *Neural Computation*, 3, pp. 213-225, 1991.
- [53] Angelov, P.; Zhou, X. "Evolving fuzzy systems from data streams in real-time." *Int. Symp. on Evolving Fuzzy Systems*, pp. 29-35, 2006.
- [54] Fritzsche, B. "A growing neural gas network learns topologies." *Advances in Neural Information Processing Systems*, 7, pp. 625-632, 1995.
- [55] Leite, D.; Costa, P.; Gomide, F. "Interval approach for evolving granular system modeling." In: Sayed-Mouchaweh M., Lughofer E. (eds), *Learning in Non-Stationary Environments*. Springer: NY, pp. 271-300, 2012.
- [56] Yeh, I.-C. "Modeling of strength of high performance concrete using artificial neural networks." *Cement and Concrete Research*, 28(12), 1797-1808, 1998.
- [57] Kausay, T.; Simon, T. K. "Acceptance of concrete compressive strength". *Concrete Structures*, 8, 54-63, 2007.