

Modelling uncertainty in production processes using non-singleton fuzzification and fuzzy cognitive maps - a virgin olive oil case study.

P. Cano Marchal*, C. Wagner[†], J. Gmez Garca* and J. Gmez Ortega*

*Robotics, Automation and Computer Vision Group

University of Jan. Agrifood Campus of International Excellence (ceiA3). Jan, Spain
{pcano,jggarcia}@ujaen.es

[†]Lab for Uncertainty in Data and Decision Making (LUCID), School of Computer Science
University of Nottingham

Abstract—Decision support systems (DSSs) are a convenient tool to aid plant operators in the selection of process set points. Inputs to these systems for variables that are not easily measured online often come from assessments made by experts, with an associated degree of uncertainty. The application of fuzzy sets and systems as part of DSSs provides a systematic approach to addressing the uncertainty in its variables. This paper builds on prior work on DSSs utilising fuzzy cognitive maps and introduces a non-singleton fuzzification stage which directly addresses uncertainty in system inputs. The motivation of the proposed system is grounded in the real world challenges of producing high-quality olive oil and the paper provides promising application and analysis results as part of the Virgin Olive Oil Production Process.

Index Terms—Nonsingleton Fuzzification, Fuzzy Cognitive Map.

I. INTRODUCTION

The virgin olive oil production process (VOOPP) is an industrial process with many relevant process variables and conflicting production objectives. A fundamental tradeoff between the *quality* of the produced virgin olive oil (VOO) and the obtained quantity – extraction yield – confers a key relevance to the choice between production objectives, as it greatly influences the economic revenue of the activity.

The selection of the set points of the process variables is a major step in the VOOPP, as these values determine the operation of the plant, and, consequently, the achievement of the production objective. This selection requires addressing the process from a global point of view, taking into account the constraints imposed by the properties of the incoming olives and the influence of the process variables on the production process and thus the potential of meeting the objectives.

In the VOOPP, there are many relevant variables of the process that are difficult to measure, which are thus typically assessed by expert operators. The properties of the incoming olives and the organoleptic quality of the produced olive oil are some examples of this type of variables. This difficulty has implications in two different facets: on the one hand, it hinders the construction of process models suitable to be used for the design of high-level controllers capable of providing the most

appropriate value for the process set points, as experimental data is scarce and expensive to obtain. On the other hand, these estimations naturally convey a certain degree of uncertainty about the true values of the estimated parameters, inherent to the approximate and, to some extent, subjective nature of the assessments (e.g. flavour). This uncertainty is often apparent when more than one expert provides an assessment for the same property, as their assessments usually do not exactly match.

In previous works [1] we have addressed the first challenge, by proposing a decision support system (DSS) capable of suggesting adequate set points for the VOOPP. The lack of experimental data and the high number of relevant process variables and production scenarios led to the design and construction of a model of the process based on Fuzzy Cognitive Maps (FCMs) [2], [3]. FCMs are a convenient tool for modeling complex systems with many variables. They provide an intuitive representation of the relationships between the variables, allow to easily decompose the model into simpler parts and facilitate an incremental approach in the construction of the model.

In [1], a set of set points of the process variables were provided by the solution of an optimization problem that used the FCM model a constraint. The inclusion of an observer for the process outputs enabled the inclusion of feedback to the system, as well as the updating of specific set points if the production objective were not met using the initially suggested values.

In this work we address the second facet of the problem, and analyze the modelling and the effect of the uncertainty on the decision making process. The uncertainty affecting an actual value of a given parameter can be modeled using a fuzzy set [4], that can in turn be used as input to the decision support system (DSS), instead of feeding a singleton fuzzy set that would disregard the available information about the uncertainty. This approach follows the same principle as applied in non-singleton fuzzy logic systems, where a non-singleton fuzzy set is used to model input uncertainty as part of a traditional fuzzy logic inference model. Non-singleton fuzzy logic systems have

been explored and shown to have substantial potential in a variety of contexts [5], [6], [7].

In particular, the objective of this paper is to explore the implications of using nonsingleton fuzzification with the FCM modeling approach and to analyze its application to a simplified DSS for the VOOPP. The rest of the paper is organized as follows: Section II presents some background of the problem, briefly addressing FCM, the VOOPP and nonsingleton fuzzification. In turn, Section III deals with the analysis of nonsingleton FCM, while Section IV addresses their application to the VOOPP decision support system. Finally, Section V includes the conclusions of the work.

II. BACKGROUND

A. Fuzzy Cognitive Maps in the context of Fuzzy Logic Systems

Fuzzy Cognitive Maps (FCM) are a convenient modeling approach for systems with a high number of variables. This methodology eases the elicitation of knowledge from experts and allows to use a systematic approach in this elicitation of the knowledge.

The proposed modeling method is based on Simplified Dynamic Cognitive Networks (sDCN) [8]. Formally, the model is defined as a tuple:

$$\mathbf{M} = \langle \mathbf{V}, \mathbf{A} \rangle, \quad (1)$$

where \mathbf{V} designates the set of nodes – representing variables –, and \mathbf{A} being the set of arcs – representing relationships between the variables.

The nodes of the network can act as predecessor or successor nodes, according to their role in the computation process. Predecessor nodes are similar to variables included in the antecedent part of a fuzzy rule, while successor nodes can be identified with the variables in the consequent part of the rule. Figure (1) shows a generic multiple-input single-output one-layer model for traditional singleton FCM. For multilayer models, a node can play both roles, if there are arcs both incoming and departing from the node.

For each node $v_i \in \mathbf{V}$ of the network, the following properties are defined:

- U_{v_i} : the universe of discourse of the node. It is the set that contains all the possible crisp values of the variable represented by v_i .
- H_{v_i} : the collection of terms (fuzzy sets) $L_{v_i}^k$ defined in U_{v_i} :

$$L_{v_i}^k = \{ \langle x, \mu_{L_{v_i}^k}(x) \rangle : x \in U_{v_i} \}, \quad (2)$$

$$H_{v_i} = \{ L_{v_i}^k, k = 1, 2, \dots, K_i \}. \quad (3)$$

Here, K_i represents the total number of sets defined in the node.

- $S_f(v_i)$: an array containing the degree of membership of for each fuzzy set $L_{v_i}^k$ of a given v_i for one crisp input (see Fig. 1).

$$S_f(v_i) = [\mu_{L_{v_i}^1}, \dots, \mu_{L_{v_i}^{K_i}}]^T. \quad (4)$$

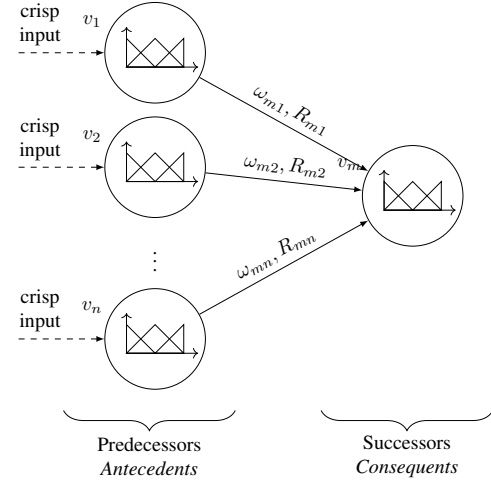


Figure 1. Generic multi-input single output model for traditional, singleton FCM. A more complex model can be built by composition of this elementary structure.

- $S_c(v_i)$: the crisp value assigned to the node. If v_i acts as a successor node, this value is computed using the method explained below in Equations (6)–(8). If v_i acts as a predecessor node, this value is considered already available, either provided by an external input to the node – if the node does not have any predecessors – or previously computed.

In turn, for each arc a_{ij} the following properties are defined:

- ω_{ij} : intensity of the relation between the nodes v_i and v_j .
- R_{ij} : causal relationship matrix. The entries of this matrix are required to be non-negative, and they define the relationship between the fuzzy sets of the predecessor and the successor nodes connected by the arc. The size of the matrix is $K_i \times K_j$, with K_i and K_j being the number of labels in H_{v_i} and H_{v_j} respectively.

These properties of the arcs ω_{ij} and R_{ij} play a similar role to the rules in a rule-based FLS, as these elements encode the relationships between the different variables of the system. In the R_{ij} matrix, each row is associated with a fuzzy set defined in U_{v_i} , and each column is associated with a fuzzy set defined in U_{v_j} . Each entry of R_{ij} can be thought of as a rule relating the value of the predecessor (node v_j) with the successor (v_i), where the rule weight is given by the value of the entry times the weight associated with the arc. For example, let $L_{v_i}^k$ be triangular fuzzy sets and let $\mathbf{q}_i = [q_i^1 \ q_i^2 \ \dots \ q_i^{K_i}]^T$ denote the peaks of these fuzzy sets. Then, each matrix entry is associated to a rule of the type:

If v_j is $L_{v_j}^b$ Then v_i is q_i^a , with weight $\omega_{ij} R_{ij}^{ab}$.
As an example, consider the following matrix:

$$R_{ij} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}. \quad (5)$$

The shape of the matrix denotes that three fuzzy sets have been defined in both U_{v_i} and U_{v_j} , and the associated rules are:

If v_j is $L_{v_j}^1$ Then v_i is q_i^3 , with weight ω_{ij} .

If v_j is $L_{v_j}^2$ Then v_i is q_i^2 , with weight ω_{ij} .

If v_j is $L_{v_j}^3$ Then v_i is q_i^1 , with weight ω_{ij} .

The computation of the value of a successor node, given the value of its predecessors, is computed as follows:

- 1) The impact received by the node i is defined as:

$$\mathbf{w}_i = \sum_{j=1}^{n_i} \omega_{ij} R_{ij} S_f(v_j) = [w_i^1 w_i^2 \cdots w_i^{K_i}]^T \quad (6)$$

- 2) The computation of the crisp value $S_c(v_i)$ of the node is performed using a weighted average combination of the value of the peak of each fuzzy set using the impact received by the node:

$$S_c(v_i) = \frac{\sum_{k=1}^{K_i} w_i^k q_i^k}{\sum_{k=1}^{K_i} w_i^k}. \quad (7)$$

- 3) Finally, the fuzzy state vector of the node $S_f(v_i)$ captures the membership values for each label as follows:

$$S_f(v_i) = [\mu_{L_{v_i}^1}(S_c(v_i)) \mu_{L_{v_i}^2}(S_c(v_i)) \cdots \mu_{L_{v_i}^{K_i}}(S_c(v_i))]^T. \quad (8)$$

Equation (7) perform the inference and defuzzification steps simultaneously, and is very similar to the zero-order Takagi-Sugeno-Kang (TSK) model computation of the crisp value of the output of a set of rules.

As pointed out in [9], for general FCM, the main constraint of the methodology is the fact that rules can only have a single variable in the premise clause. Part of the appeal of FCMs is that the selection of different matrices enables the direct capture of potential intuitive behaviours that may be perceived by experts, such as the ones presented in [1]. FCMs traditionally allow having cycles in the graph, however, the FCM framework introduced in [1] currently does not contemplate having them in the models. More details on the reasons for this decision are found in [1].

B. Virgin Olive Oil Production Process

The VOOPP is usually considered to begin with the reception of olives at the factory. However, aspects such as the harvesting date, the harvesting method and the time elapsed from grove to factory are also important aspects to consider, since they affect properties of the olives that influence the final characteristics of the oil.

Olives usually arrive with some dust, leaves and small pebbles that need to be removed before their processing. Once the olives are clean, they are fed into hoppers where they are stored for a certain period of time, and then crushed to form the so called *olive paste*. The state of the paste as it comes out of this operation is not fit for the separation of the oil, as it would not allow a good separation yield. In order to improve this yield, the paste is fed into a thermomixer where it is gently stirred and heated to promote the coalescence of the oil drops. After this step, the paste is fed into a solid-bowl horizontal centrifuge where oil is separated from the *pomace* (solids and water). The moisture and impurity content of the oil is not yet adequate for its storage, so a further separation step is

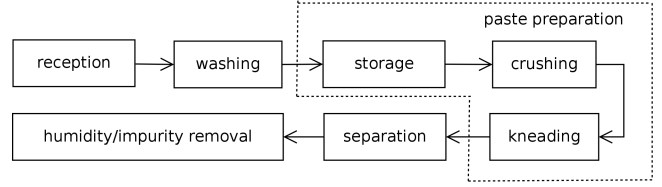


Figure 2. Block diagram of the olive oil production process.

Table I
VARIABLES INVOLVED IN EACH OPERATION CAPTURED IN FIG. (2)

Operation	Symbol	Variable
Paste preparation	t_s	Storage time
	C	Size of the sieve hole
	t_b	Kneading time
	T_b	Kneading temperature
	A_t	Microcalc addition
Separation	F_p	Paste flow into decanter
	F_w	Addition water flow into decanter
	r_1	Oil outcome weirs position
	$\Delta\omega$	Screw-bowl differential velocity
Humidity removal	t_r	Residence time in settling tank
	f_p	Purge frequency of settling tank
	$F_{w,vc}$	Addition water flow to vertical centrifuge

performed in vertical centrifuges or static settling tanks. After this step, the oil is prepared for its storage. Fig. 2 shows a block diagram of the process.

Overall, three different major operations make up this process: paste preparation, oil-pomace separation and impurity and humidity removal. The paste preparation includes all the operations until feeding the paste into the horizontal centrifuge, where the oil-pomace separation operation is performed. The impurity and humidity removal comprises the remaining operations.

The main high level output parameters of the process are the quality of the oil and the extraction yield. Olive characteristics impose an upper bound on these two variables, while the different technical variables involved in the process further determine their final values within those bounds. The paste preparation step exerts a key influence on the quality of the oil, and sets an upper bound on the yield. In turn, the separation step does not play a significant role for the quality, but greatly determines the yield, as sub-optimal values may be obtained if the operation is not carried out properly. The humidity and impurity removal step, although important for the proper storage of the oil, plays a more subtle role in the process [10], and is usually not considered, as its influence on the outputs is lower compared to the other two steps. The major technical variables of each phase are presented in Table I.

During regular VOOPP operation, many of the relevant process variables are assessed and set by expert operators. Some variables are inherently uncertain and only assessed through the experience and expertise of the expert operators of the process. An example of this is *paste preparation*, which is an assessment of the expert based on some features of the

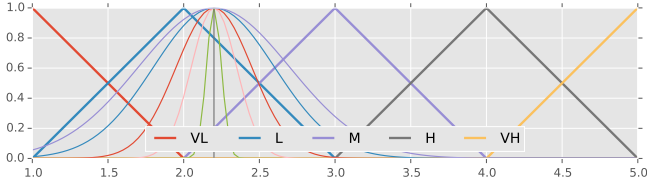


Figure 3. Intersection of Gaussians with increasing values of σ with triangular membership functions. Gaussians with higher σ intersect more labels, and at higher membership values.

paste inside the thermomixer – granularity, cleanness of the blades and color of the paste, mainly. This variable provides information on the expected yield before the separation process is carried out.

Some other variables, such as the maturity and the moisture content of the olives, do have definite crisp values; however, given the conditions of normal operation of the factory (e.g., where multiple batches or olives with different properties are mixed), the only information about their values during production is an assessment provided by one or more expert operators, and decisions are made based on this input.

Given the nature of these assessments and the heterogeneity of the batches of olives being processed, a considerable degree of uncertainty is commonplace even where some numeric properties of specific olive batches are known. It is interesting to consider the effect of these uncertainties on the decision-making process of a decision support system in order to enable a robust manufacturing process which can reliably achieve *optimal* production.

C. Nonsingleton Fuzzification

As commented in the previous Section, the values of input variables in the VOOPP usually have significant associated uncertainty. This uncertainty can be modeled using fuzzy sets to describe the values of the variables assessed by the experts, instead of just using a crisp value to represent them.

The use of fuzzy sets as inputs to RBFLSs requires defining a method to compute the firing degree of the rules based on the interaction of the fuzzy set representing the input and the fuzzy sets of the antecedent clause of each rule. A common approach is the sup-min interaction of the fuzzy sets. Let X denote the fuzzy set representing the value of the input, and L denote the fuzzy set in the antecedent clause of a rule, then:

$$\mu_{X \circ L} = \sup_{u \in U_X} \min[\mu_X(u), \mu_L(u)]. \quad (9)$$

Figure (3) represents the intersection of Gaussian fuzzy sets with increasing values of their standard deviation (σ) – representing increasing levels of uncertainty about the input – with a collection of triangular fuzzy sets. As can be seen in the Figure, Gaussian sets with a higher degree of uncertainty intersect more triangular fuzzy sets – meaning that more rules are to be triggered – and with higher firing degree. This reflects the intuitive interpretation that as the input is uncertain, different rules would potentially be fired to higher degrees.

III. NONSINGLETON FUZZY COGNITIVE MAPS

The use of nonsingleton fuzzification in FCM implies considering that the inputs to the nodes in the first layer (v_j) of the model are no longer crisp values (x_j), but fuzzy sets (X_j). Consequently, the computation of $S_f(v_j)$ needs to be performed taking into account the interaction of X_j and the fuzzy set associated with each of its entries $L_{v_j}^k$. If we choose sup-min interaction of the sets, then the elements of $S_{f,ns}(v_j)$ can be computed as:

$$S_{f,ns}^k(v_j) = \sup_{u \in U_{v_j}} \min[\mu_{X_j}(u), \mu_{L_{v_j}^k}(u)] \quad (10)$$

The left column of Fig. 4 shows plots of the value of the elements of $S_f(v_j)$ when X_k is a Gaussian fuzzy set whose mean corresponds to the value of the x -axis, for three different values of σ . As depicted in the Figure, for plots where $\sigma > 0$ the elements of $S_f(v_j)$ do not add up to one (i.e., for some vertical slices, the sum of the memberships in the different MFs is higher than one); in fact, the higher σ , the higher the value of this sum. As commented in the previous section, this is a known characteristic of nonsingleton fuzzification, as sets with higher uncertainty usually trigger more fuzzy rules and at higher firing values.

In the singleton-FCM, the fuzzy labels defined in the universe of discourse are chosen such that they constitute a fuzzy partition, so it was guaranteed that the sum of all the elements of S_f add up to one. With the nonsingleton fuzzification, that property is no longer guaranteed. However, to re-establish the property, we can *re-normalise* the membership functions as follows: $S_{f,ns}(v_j)$ as:

$$S_{f,ns}(v_j) = c_j \hat{S}_{f,ns}(v_j) \quad (11)$$

with $c_j = \sum_k S_{f,ns}^k(v_j)$ and $\sum_i \hat{S}_{f,ns}^k = 1$. Then, the impact exerted by node v_j on the node v_i can be expressed as:

$$\omega_{ij} R_{ij} S_{f,ns}(v_j) = \omega_{ij} R_{ij} c_j \hat{S}_{f,ns}(v_j) = \hat{\omega}_{ij} R_{ij} \hat{S}_{f,ns}(v_j) \quad (12)$$

This Equation can be interpreted according to the meaning of the different elements of the FCM model. In the FCM methodology, the influence of a node v_j on its successor v_i can be separated into two aspects:

- First, the relation matrix connecting both nodes (R_{ij}) defines which rules should be triggered based on the value of $S_f(v_i)$. This can be seen as a definition of what the value of the sucesor node should be, based exclusively to the value of this particular input node.
- Second, the relation weight ω_{ij} accounts for the intensity of the influence of v_j relative to all the other nodes that affect the successor. That is, it defines the impact in the final computation of v_i of the value advocated for by v_j .

The substitution of $S_f(v_j)$ with $\hat{S}_{f,ns}(v_j)$ can be intuitively interpreted as changes in the consideration of the influence of node v_j due to uncertainty in its value. The right column of Fig. 4 shows plots of the value of the elements of $\hat{S}_{f,ns}(v_j)$ for the corresponding $S_{f,ns}(v_j)$ plotted on the left column. As can be seen in the plot, the increase in the uncertainty produces a

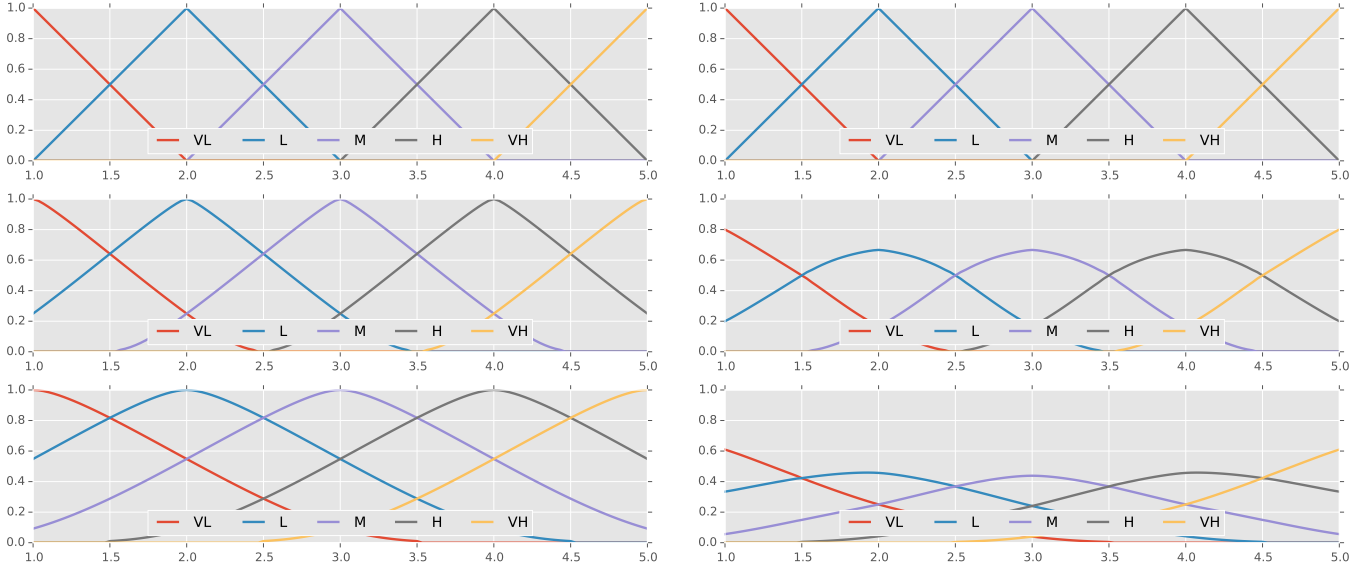


Figure 4. Comparison of $S_{f,ns}(u_k)$ (left) and $\hat{S}_{f,ns}(u_k)$ (right) for $S_{f,ns}^k(v_j) = \sup_{u \in U_{v_j}} \min[\mu_{X_j}(u), \mu_{L_{v_j}^k}(u)]$ and $\sigma = [0, 0.25, 0.5]$ (top to bottom).

decrease in the value of each of the components of $\hat{S}_{f,ns}(v_j)$, with more elements having nonzero values. This means that more rules get activated, but at a lower level of certainty each of them.

In turn, the conversion of ω_{ij} into $\hat{\omega}_{ij}$ can be seen as a change in the relative weights of the system, that is, a change of the relative strength of the influence of all the nodes that influence v_i . As commented before, higher uncertainty normally corresponds to higher c_j , and, consequently, higher $\hat{\omega}_{ij}$. This means that variables whose value is more uncertain are assigned greater influence in the model, which is not desirable. This can be corrected by simply ignoring c_j in the model. This way, the computation of the impact of node v_i on v_j is defined to be:

$$\mathbf{w}_{ij} = \omega_{ij} R_{ij} \hat{S}_{f,ns}(v_j). \quad (13)$$

This equation benefits from the smoothing effect of higher spread of $\hat{S}_{f,ns}(v_j)$ (inherent to non-singleton fuzzification) without artificially altering the weights of the models. The next Section presents and discusses the different results obtained considering Eq. (12) and (13) in the computations.

IV. APPLICATION OF NONSINGLETON FCM TO THE VOOPP

Previous works have dealt the construction of a DSS for the VOOPP capable of suggesting the values of the set points of the process variables that maximize some given objective function, subject to a certain set of characteristics of the input olives [1]. The DSS consists of a FCM model of the VOOPP that acts as a set of constraints for an optimization problem. Figure 5 shows the graph representing the simplified model of the VOOPP used for the DSS.

Although the main purpose of the DSS is to provide the set points of the process, which are given by the solution of an optimization problem, it is interesting to analyse the influence

on the nonsingleton approach on the output variables of the model.

Figure 6 depicts *Kneading State* as a function of *Incoming Olive Moisture* and *Coadjuvant Addition* for three levels of uncertainty in *Incoming Olive Moisture* – left to right – and using both the normalized $\hat{S}_{f,ns}$ and non-normalized approach – top and bottom, respectively. In turn, *Storage Time in Hopper*, *Kneading Time* and *Kneading Temperature* are fixed. The inclusion of the uncertainty can be seen to moderately reshape the colormap, particularly for high values of *Incoming Olive Moisture*. There is, however, no visible difference between the normalized and non-normalized approach because of the chosen value of *Storage Time in Hopper*, which has very similar effects on the output in both approaches.

Figure 7 shows the dependence of *Kneading State* in *Incoming Olive Moisture* and *Storage Time in Hopper*. In this plot, the different implications of using the normalized vs. non-normalized approach become apparent, as increasing levels of uncertainty produce opposite effects in the area of high values of *Incoming Olive Moisture*. We can also see the similarity of both approaches in the very low range of values of *Storage Time in Hopper*.

We now focus on the complete DSS. The variables of the model can be grouped according to their role in the optimization problem as:

- Parameters (\mathbf{p}): variables whose values are fixed, i.e., constants. These include both the properties of the input olives and other process parameters that can be regarded as constant. They specify the considered production scenario.
- Outputs (\mathbf{y}): variables that constitute the output of the process. In the model presented in Figure 5, these are *Fruity*, *Paste Preparation* and *Defect*.
- Decision nodes (\mathbf{u}): these are process variables whose value comprises the output of the DSS, that is, the set

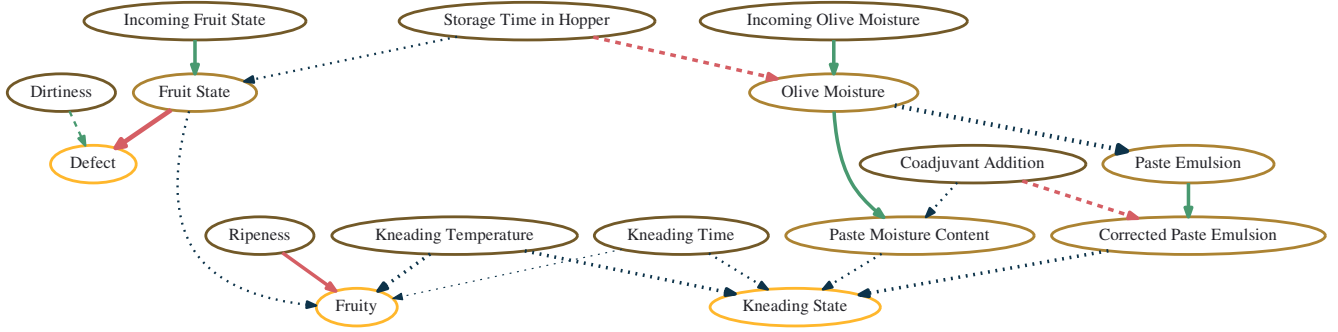


Figure 5. Model of the VOOPP.

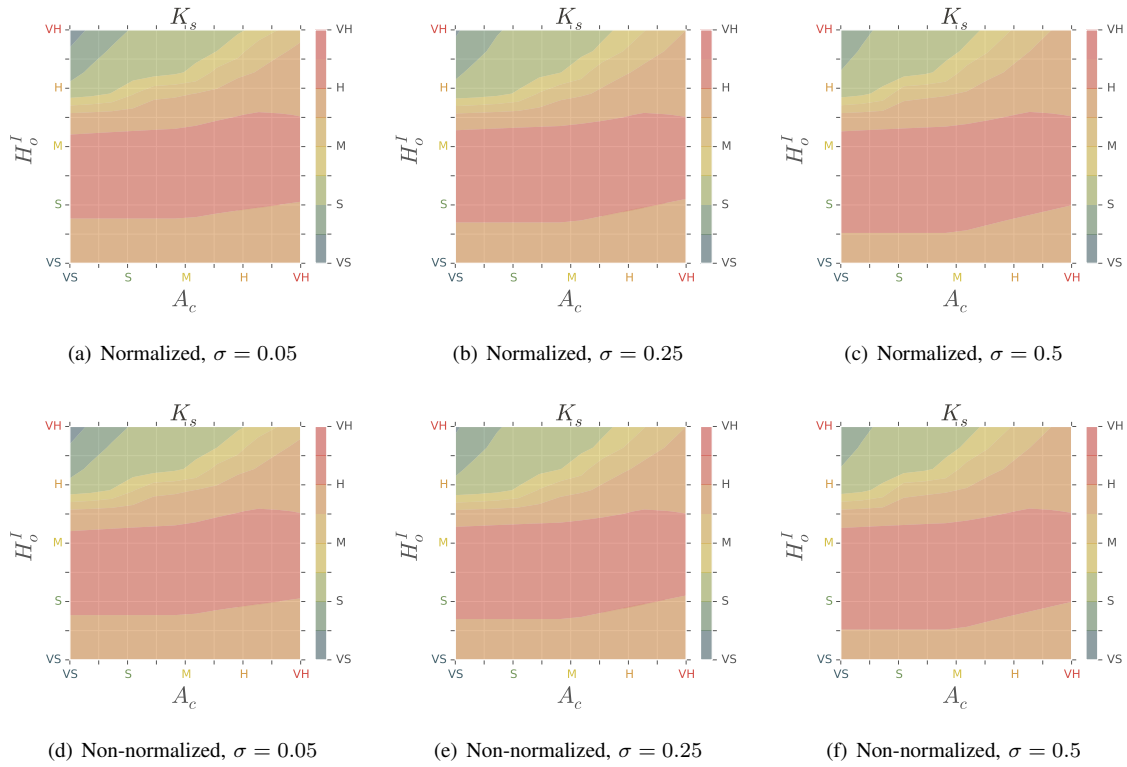


Figure 6. *Kneading State* as of function of *Coadjuvant Addition* (x -axis) and *Incoming Olive Moisture* (y -axis) for $\sigma = [0.05, 0.25, 0.5]$ (left to right). Upper and lower rows represent the normalized and non-normalized approaches, respectively.

points suggested to the operator.

Let's consider the problem of finding the optimal values of the decision nodes \mathbf{u} that allow obtaining a certain prescribed value of the outputs \mathbf{y} for a given production scenario specified by the parameters \mathbf{p} . This problem can be formalized as:

$$\begin{aligned}
 & \underset{\mathbf{u}_k}{\text{minimize}} && J = (\mathbf{y} - \mathbf{T})^T Q (\mathbf{y} - \mathbf{T}) + \mathbf{u}^T R \mathbf{u} \\
 & \text{subject to} && \mathbf{y} = f(\mathbf{u}, \mathbf{p}) \\
 & && \mathbf{p} = \mathbf{p}^0 \\
 & && \mathbf{u}_{min} \leq \mathbf{u} \leq \mathbf{u}_{max}.
 \end{aligned}$$

Here \mathbf{p}^0 stands for the values of \mathbf{p} for the situation at hand, \mathbf{T} represent the prescribed values of \mathbf{y} and $f(\cdot)$ denotes the fuzzy model. The relative weight for the different outputs and

the cost of the decision variables is defined using the matrices Q and R respectively, as is usual in Model Predictive Control [11].

Let's consider a production scenario where *Ripeness* is very low, but with a varying degree of certainty about its true value. Let's further suppose that the production objective is attaining a value of *Fruity* as close to 4 as possible, and the highest possible *Kneading State*, subject to fulfilling the *Fruity* objective.

Tables II and III show the prescribed set points for *Kneading Time* and *Kneading Temperature* – the variables whose set point is to be decided, the value of the rest of the variables is considered to be fixed – and the values of the output variables predicted by the model for the case when normalization of

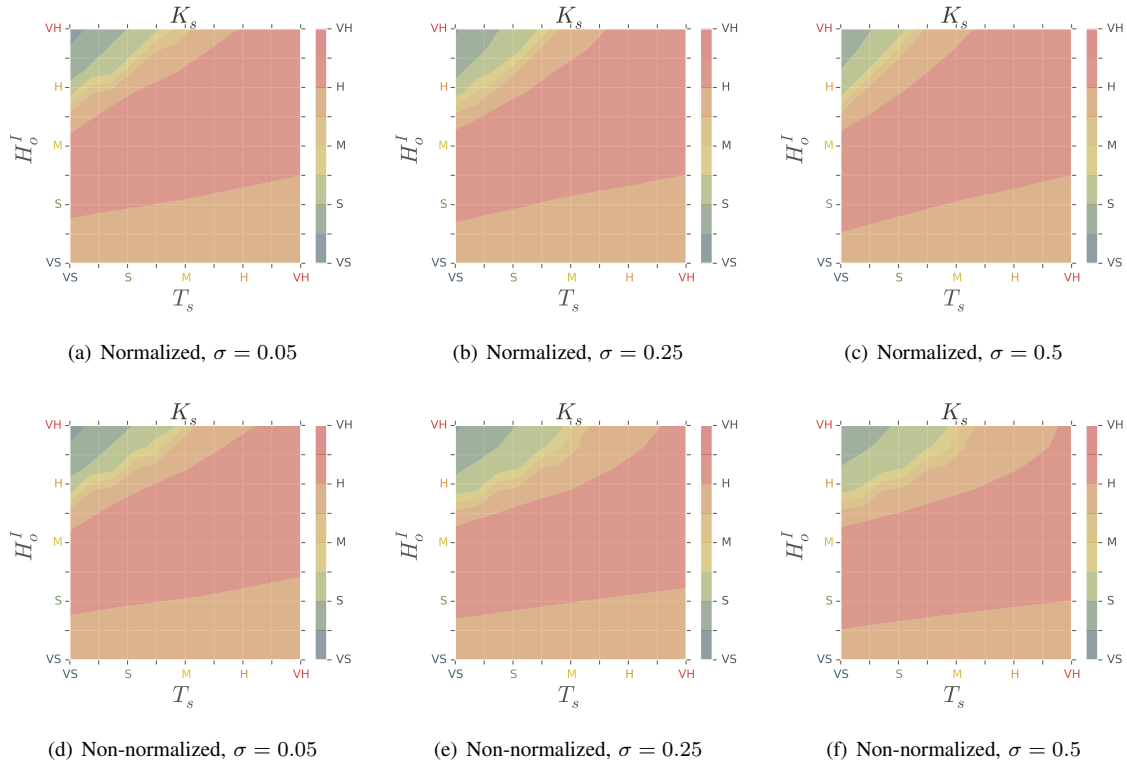


Figure 7. *Kneading State* as of function of *Storage Time in Hopper* (x -axis) and *Incoming Olive Moisture* (y -axis) for $\sigma = [0.05, 0.25, 0.5]$ (left to right). Upper and lower rows represent the normalized and non-normalized approaches, respectively.

Table II

SET POINTS AND EXPECTED OUTPUT VALUES FOR NONSINGLETON MODEL WITH $S_f^{m,f}$ NORMALIZATION FOR GIVEN VALUES OF σ

σ	0	0.05	0.25	0.5
<i>Ripeness</i>	4.00	4.00	4.00	4.00
<i>Fruity</i>	3.95	3.95	3.91	3.93
<i>Kneading State</i>	3.06	2.98	2.85	2.61
<i>Defect</i>	1.00	1.00	1.00	1.00
<i>Kneading Time</i>	3.87	3.26	3.00	3.00
<i>Kneading Temperature</i>	2.00	2.00	1.90	1.62

Table III

SET POINTS AND EXPECTED OUTPUT VALUES FOR NONSINGLETON MODEL WITHOUT $S_f^{m,f}$ NORMALIZATION FOR GIVEN VALUES OF σ

σ	0	0.05	0.25	0.5
<i>Ripeness</i>	4.00	4.00	4.00	4.00
<i>Fruity</i>	3.95	3.94	3.95	3.93
<i>Kneading State</i>	3.06	3.06	3.07	3.04
<i>Defect</i>	1.00	1.00	1.00	1.00
<i>Kneading Time</i>	3.87	3.94	4.00	3.75
<i>Kneading Temperature</i>	2.00	2.00	2.00	2.00

$S_f^{m,f}$ is and is not implemented, respectively.

As is clear from Table II, as σ increases, the normalized $S_f^{m,f}$ approach provides more conservative values for assuring that the *Fruity* objective is met, that is, decrease *Kneading Time* and *Kneading Temperature*, which makes intuitive sense. If there is uncertainty about the value of the input, and the production objective definitely encourages *Fruity* to reach a certain threshold, the rational choice is to be conservative in the selection the set points to assure achieving the primary

objective, at the expense of the secondary objective.

On the other hand, the non-normalized approach (Table III) initially increases the value of *Kneading Time*, which is counter intuitive and definitely not the action an expert operator of the process would take. This is caused by the modified weight assigned to the arc relating *Ripeness* and *Fruity* due to the triggering of more rules for the node and at a higher firing strength. The effect is equivalent to considering that the influence of having a certain value of *Ripeness* exerts a higher influence on *Fruity*. Since the value is favorable for obtaining a good value of *Fruity*, there is more room to pursue the secondary objective, thus the increase of the *Kneading Time*. This effect can be visualized in Figure 8, which depicts *Fruity* as a function of *Kneading Temperature* and *Ripeness* for different values of uncertainty in this last variable. The plots clearly show the increased weight assigned to *Ripeness* in the non-normalized approach, apparent in the higher values of *Fruity* assigned to low values of *Ripeness* for high values of *Kneading Temperature* (marked with an X on the plot).

V. CONCLUSIONS

In this paper we have analyzed the implications of using nonsingleton fuzzification for FCM models and its application to a DSS for the VOOPP. We have seen that normalizing $S_{f,ns}$ provides results that correspond better to the decisions an expert operator would make, and an interpretation of the cause has been discussed.

Further work will address whether the result is specific to FCM due to their particular modeling methodology, or is also

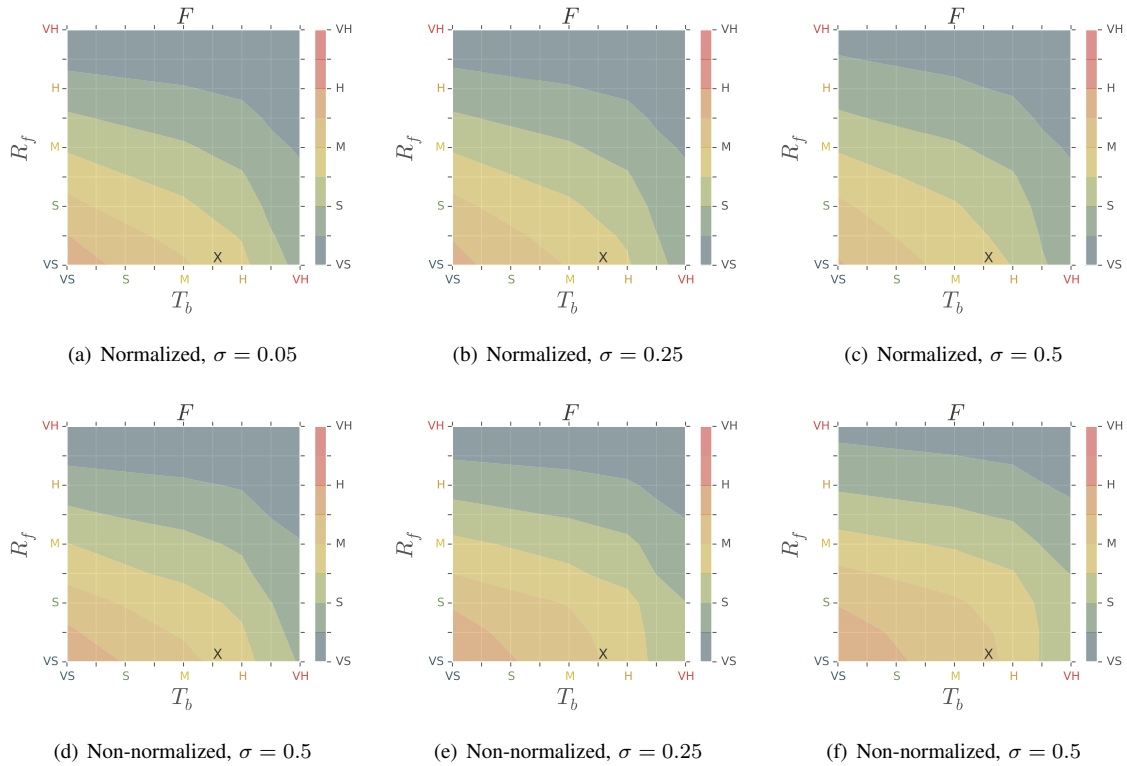


Figure 8. *Fruity* as of function of *Kneading Time* (x -axis) and *Ripeness* (y -axis) for $\sigma = [0.05, 0.25, 0.5]$ (left to right). Upper and lower rows represent the normalized and non-normalized approaches, respectively. The X marks a region where show the increased weight assigned to *Ripeness* in the non-normalized approach is apparent.

applicable to standar rule-based FLSs. Another current line of work includes the analysis of the impact of using different rules for the modelling of the interaction between the system inputs and the antecedent fuzzy sets, in particular for non-symmetric input fuzzy sets. Finally, we are considering the exploration of recently introduced variants of non-singleton FLSs which may provide a better capture of the interaction of input and antecedent FSs [12].

ACKNOWLEDGMENTS

This work was partially supported by the projects DPI2011-27284 and PI10-AGR-6616. P. Cano Marchal has been in receipt of a F.P.U. grant from the Spanish Ministry of Education.

REFERENCES

- [1] P. Cano Marchal, J. Gmez Garcia, and J. Gmez Ortega, "Decision Support System Based on Fuzzy Cognitive Maps and Run-to-Run Control for Global Set-Point Determination," in *2015 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Oct. 2015, pp. 1745–1751.
- [2] B. Kosko, "Fuzzy Cognitive Maps," *International Journal of Man-Machine Studies*, vol. 24, pp. 65–74, 1986.
- [3] C. Stylios and P. P. Groumos, "Modeling complex systems using fuzzy cognitive maps," *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 34, no. 1, pp. 155–162, Jan. 2004.
- [4] G. Mouzouris and J. Mendel, "Nonsingleton fuzzy logic systems: theory and application," *IEEE Transactions on Fuzzy Systems*, vol. 5, no. 1, pp. 56–71, Feb. 1997.
- [5] J. M. Mendel, *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*. Upper Saddle River, NJ: Prentice Hall, 2001.
- [6] C. Wagner and H. Hagra, "Toward General Type-2 Fuzzy Logic Systems Based on zSlices," *IEEE Transactions on Fuzzy Systems*, vol. 18, no. 4, pp. 637–660, Aug. 2010.
- [7] J. Aladi, C. Wagner, and J. Garibaldi, "Type-1 or interval type-2 fuzzy logic systems #x2014; On the relationship of the amount of uncertainty and FOU size," in *2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Jul. 2014, pp. 2360–2367.
- [8] Y. Miao, C. Miao, X. Tao, Z. Shen, and Z. Liu, "Transformation of Cognitive Maps," *IEEE Transactions on Fuzzy Systems*, vol. 18, no. 1, pp. 114–124, Feb. 2010.
- [9] S. W. Khor, M. S. Khan, and K. W. Wong, "Fuzzy inferencing using single-antecedent fuzzy rules," *International Journal of Fuzzy Systems*, vol. 8, no. 2, pp. 65–75, 2006.
- [10] P. Cano Marchal, J. Gmez Ortega, D. Aguilera Puerto, and J. Gmez Garca, "Situacin actual y perspectivas futuras del control del proceso de elaboracin del aceite de oliva virgen," *Revista Iberoamericana de Automtica e Informtica Industrial RIAI*, vol. 8, no. 3, pp. 258–269, Jul. 2011.
- [11] E. F. Camacho and C. Bordons, *Model predictive control*. Springer, 2004.
- [12] A. Pourabdollah, C. Wagner, and J. Aladi, "Changes under the hood - a new type of non-singleton fuzzy logic system," in *2015 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Aug. 2015, pp. 1–8.