# Combined Defuzzification Under Shared Constraint

Jörg Verstraete, Weronika Radziszewska, Katarzyna Kaczmarek-Majer, and Sebastian Bykuć

***Abstract*—Defuzzification of fuzzy sets is an important aspect of fuzzy processing, as it determines how the fuzziness is dealt with when performing the final step to obtain crisp solutions. In this contribution, we consider the possibility of defuzzifying multiple general type-1 fuzzy sets, given that their defuzzified values are bound by a single, known constraint. This stems from an application in spatial data processing, where we obtained a number of fuzzy sets whose defuzzified values should sum up to a crisp and known value. It is possible to defuzzify each fuzzy set individually and rescale the outcomes to meet the constraint. However, considering that the fuzzy set contains information on what constitutes good values, accounting for the constraint in the defuzzification process has the potential to yield better results. We considered this as an optimization problem and investigated appropriate goal functions. The approach and goal functions are discussed with respect to typical properties of defuzzifiers.**

***Index Terms*—Combined defuzzification, spatial reasoning.**

## I. Introduction

DEFUZZIFICATION is an important aspect when interpreting the results of any calculation that yields fuzzy sets, as this step permits to continue crisp calculations or to draw conclusions. The goal of defuzzification is to determine a single crisp value that is representative for the fuzzy set; various operations for defuzzification exist, each with a justification as to why the determined value is representative but also with criticism and cases in which the value appears less intuitive and potentially less suitable. Here, a variation of the defuzzification problem is considered. The problem at hand stems from an application of type-1 fuzzy inference systems in a spatial context, where the inference system is used to determine the values associated with geographic features [1]. These resulting values are bound by a shared constraint, i.e., these values should sum up to a known *crisp* value. While it is possible to defuzzify each set individually and rescale values to meet the constraint, one can consider that—as the fuzzy sets represent suitable values for the solutions—a global solution can be found that meets the constraint while selecting *better* values using the definition of fuzzy sets. *Better* in this context can be evaluated by means of a goal (or objective) function in combination with the membership functions. In this situation, the defuzzification of the multiple type-1 fuzzy sets effectively becomes an optimization problem. In this article, we will present an approach that determines translations from known defuzzifiers using different goal functions to define what constitutes *better* values and discuss the performance and behaviour of the results with respect to properties of defuzzifiers.

## II. Problem

### A. Origin: Spatial Data Processing

In spatial data modeling, data are often represented in a gridded format: The region of interest is partitioned in smaller regions and each of these has a value associated. The definition/geometry of the partitions are chosen to suit the data-gathering method, the needs of the application or other reasons. This partitioning can be regular (e.g., a grid of rectangular cells) or irregular (e.g., using administrative borders). The grid or raster provides for a discrete approximation of the spatial distribution of a value that may be continuous in the 2-D space. Examples are temperatures, air pollution concentrations, or demographic data. In order to perform spatial calculations on raster data (typically an application of Tomlin's map algebra, [2]) or compare different datasets, a remapping or regridding of the data is often necessary as data may be defined using different partitionings. When this is the case, the data are considered *incompatible*, the problem of this remapping is called the *map overlay problem*. Different approaches to the map overlay problem exist, all in some way trying to determine an assumed underlying distribution. In [1], the authors presented an approach that employs a type-1 fuzzy inference system to perform spatial disaggregation, mapping the data to a more refined partitioning, using additionally available information. This approach was later extended to perform a general regridding [3]. The inference system is applied once for each new partition and yields a fuzzy set which contains the possible

Manuscript received 15 May 2023; revised 27 October 2023; accepted 13 February 2024. Date of publication 19 February 2024; date of current version 3 May 2024. The work of Jörg Verstraete was supported by the European Union's Horizon 2020 research and innovation programme through project LOCALISED–localised decarbonisation pathways for citizens, local administrations and businesses to inform for mitigation and adaptation action–under Grant 101036458. The work of Katarzyna Kaczmarek-Majer was supported by the Foundation for Polish Science through the "NeuroSmog: Determining the impact of air pollution on the developing brain" under Grant POIR.04.04.00-1763/18-00 which is implemented as part of the TEAM-NET programme of the Foundation for Polish Science, co-financed from EU resources, obtained from the European Regional Development Fund under the Smart Growth Operational Programme. The work of Sebastian Bykuć was supported by the project Greenheat. Publication funded with the GREENHEAT project ("GREEN HEAT - towards collaborative local decarbonization"), which benefits from a € 1.3 m grant from Iceland, Liechtenstein and Norway through the EEA Grants. The article reflects the author's views. The European Commission is not responsible for any use that may be made of the information it contains. Recommended by Associate Editor F. Li. *(Corresponding author: Jörg Verstraete.)*

Jörg Verstraete and Sebastian Bykuć are with the Szewalski Institute of Fluid-Flow Machinery, Polish Academy of Sciences, 80-231 Gdańsk, Poland (e-mail: jorg.verstraete@imp.gda.pl; sbykuc@imp.gda.pl).

Weronika Radziszewska and Katarzyna Kaczmarek-Majer are with Systems Research Institute, Polish Academy of Sciences, 01-447 Warsaw, Poland (e-mail: weronika.radziszewska@ibspan.waw.pl; katarzyna.kaczmarek@ibspan.waw.pl).

This article has supplementary downloadable material available at https://doi.org/10.1109/TFUZZ.2024.3367008, provided by the authors.

Digital Object Identifier 10.1109/TFUZZ.2024.3367008

values for the result of the disaggregation, a disaggregation thus requires a rulebase application for each partition. The catch is that the data modeled should not change: If the data concerns the number of people living in a region, the number of people in the partitions should match the number in the region prior to disaggregation. This example yields the constraint that the results after defuzzification should sum up to a known value. Other data may have differently defined constraints, as is the case, e.g., for temperatures or concentrations.

The problem of disaggregation therefor translates to a problem of extracting elements of a fuzzy sets that meet a given constraint. With the fuzzy sets representing possible values, this can be considered a defuzzification problem in which the defuzzified values are bound by a shared constraint.

## B. Applications

The aforementioned spatial problem is common was encountered by the authors in several projects. The regridding solution was developed to match air pollution data with population data to estimate exposure to airborne pollutants, and was recently applied to determine buildings' heating requirements based on spatial data in the context of the project GREEN HEAT. In NeuroSmog, which aims to assess the correlation between the historical air pollution exposure and neurological diseases in children, the problem occurred when preprocessing data about explanatory variables related to land-use changes, roads, emissions, and atmospheric dispersion. The project LOCALISED heavily depends on disaggregation for gathering data, and as such also encounters the spatial data issue. Apart from spatial applications, there are other situations where a simultaneous constrained defuzzification can occur: The authors in [4] presented a problem of counting traffic, where the inaccuracy of the hardware combined with knowledge on the road network creates a similar need for a constrained defuzzification.

## C. Example

An example of an application of spatial disaggregation is the disaggregation of a population density grid with $2 \times 2$ km cells into $1 \times 1$ km cells. Using the rulebase approach, each $1 \times 1$ km cell will receive a fuzzy set representing the possible number of residents based on the output of the rulebase. As the total number of residents should not change, a correction would follow the defuzzification in order to meet this constraint. In this article, we consider a smaller artificial example consisting of three fuzzy sets $A$, $B$, and $C$, defined, respectively, using the control points $\{(0.0, 1.0), (0.4, 0.0)\}$, $\{(0.5, 0.0), (0.7, 1.0), (0.9, 1.0), (1.0, 0.4)\}$, $\{(0.0, 0.0), (0.1, 0.8), (0.2, 0.7), (0.5, 1.0), (0.5, 0.0)\}$. The fuzzy sets are deliberately constructed in order to highlight issues in the defuzzification. They are shown on Fig. 1; Table I lists their defuzzified values using COG and MeOM.

In this example, we consider that the sum of the defuzzified values should be equal to 1.5 (this is an arbitrarily chosen value reachable as the sum of values contained in the supports of all three sets). As a simple way of meeting the constraint, we scale the values. This can be done by defining a translation that uses

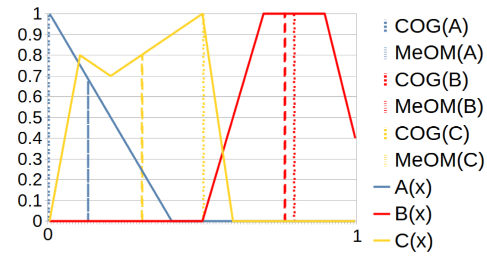



Fig. 1. Three fuzzy sets used in the example. The defuzzifed values using center of gravity and mean of max are indicated with the dashed and dotted lines, respectively.

TABLE I
DEFUZZIFIED VALUES (COG AND MEOM) OF THE FUZZY SETS IN THE EXAMPLE

| | | | $\Sigma = 1.5$ | |
|---|---|---|---|---|
| | COG | MeOM | COG | MeOM |
| A | 0.13 | 0.00 | 0.23 | 0.11 |
| B | 0.77 | 0.80 | 0.86 | 0.86 |
| C | 0.30 | 0.50 | 0.41 | 0.51 |
| Sum | 1.21 | 1.30 | 1.50 | 1.50 |

Translated values to meet the constraint are shown in the last two columns.

the difference between the defuzzified value and the maximum value of the support $x_{\sup}$ as weights, i.e., $\forall j = 1..n : D(A'_j) = D(A_j) + \frac{x_{\sup_j} - D(A_j)}{\sum_{i=1}^{n} |x_{\sup_i} - D(A_i)|}$ (if the constraining value would be less than the sum of defuzzifiers, we replace $x_{\sup}$ with the minimum of the support $x_{\inf}$). This approach guarantees that the translated values are within the support for a constraining value that can be reached with values from the support (if the support is an interval). In this example, the constraining value is larger than the sum of the defuzzified values (both for CoG and MeOM), which means that all defuzzified values are translated to greater values; as listed on Table I.

The total translation of the defuzzifiers to meet the constraint is 0.29 in case of CoG and 0.20 in case of MeOM; the question arises if the presented outcome is good – whatever good means. The choice of the translation algorithm is quite arbitrary and while it is possible to devise different methods, such approaches completely ignore the information contained in fuzzy sets.

The rest of this article is organized as follows. Section III lists work related to the problem from the defuzzification point of view, including a short introduction on defuzzifers. Our proposed solution is elaborated on in Section IV. Section V considers properties specific to this solution, whereas Section VI evaluates the behaviour using common properties of defuzzifiers. Finally, Section VIII concludes this article

# III. RELATED WORK

## A. Defuzzifiers

A fuzzy set $A$ over the domain $\mathbb{R}$ is a function that maps values from $\mathbb{R}$ to the interval [0,1]: $A(x) : \mathbb{R} \to [0, 1], x \mapsto A(x)$. The defuzzifier returns a single element of the domain that best represents it; a defuzzifier $D$ is defined as $D : F(X) \to X, A \mapsto D(A)$, with $F(X)$ the notation for fuzzy sets over the domain $X$. In this contribution, only fuzzy sets over $X = \mathbb{R}$ are considered.

Common examples of defuzzification operators for this domain are Mean Of Max (MeOM) and Center Of Gravity (COG), respectively, returning the mean of the values of the core and the center of gravity of the area under the membership function

$$\text{MeOM}(A) = \frac{\sum_{x\in\text{core}(A)} x}{|\text{core}(A)|} \tag{1}$$

$$\text{COG}(A) = \frac{\sum_{x=x_{\text{inf}}}^{x_{\text{sup}}} xA(x)}{\sum_{x=x_{\text{inf}}}^{x_{\text{max}}} A(x)}. \tag{2}$$

Various other defuzzifiers and classes of defuzzifiers (e.g., Center Of Area, BADD, SLIDE, FuzzyMeans, etc.) exist and we refer to [5] for further overview and discussion.

### B. Constrained Defuzzification

The authors in [6] provided a survey of defuzzification strategies and only refer to [7] in the context of constraint defuzzification. The authors in [7] and [8] considered constrained defuzzification, which is defined as a defuzzification where a constraint imposes limits on the domain of possible defuzzified values. In [9], defuzzification was approached as a clustering problem, in order to provide an alternative to the aforementioned defuzzification. The constraints in these context modify the defuzzification of a single fuzzy set and are independent of other fuzzy sets or their defuzzification and, to some extent, comparable to the *permissible zones* described in [10].

The defuzzification of a single fuzzy set was considered as a constrained optimization problem in [11], while limitations on defuzzifications in applications were elaborated upon in [12]. Neither refers to constraints as an aspect of the defuzzification process itself.

Closer to the specific defuzzification in this article, the authors in [4] solved a very specific problem of mitigating the inaccuracies of the hardware used to count traffic. This is done by defining triangular fuzzy sets around the obtained values and requiring defuzzified values to meet a constraint (values of different counters should sum up). As a problem, this last requirement somewhat resembles our constraint defuzzification, however the input is stricter defined and the solution method—while sufficient for their application—is not easily generalized and has not been verified theoretically.

In [13], an attempt at algorithmically performing a constraint defuzzification of multiple fuzzy sets under a shared constraint was made. The algorithm was limited to convex fuzzy sets and shifted the values of the MeOM defuzzifers, maximizing the lowest membership grade. The restrictions to both the defuzzifier and the goal function are rather limiting. In this contribution, an entirely different approach to the problem allows not only for nonconvex fuzzy sets, but also for defuzzifiers other than mean of max and for goals different from maximizing lowest membership.

The key difference between the constraints on defuzzification in literature and the presented method is that the presented method relates to a constraint that affects the defuzzification of multiple fuzzy sets at once. The methods in [4] and [13] shared this problem but the proposed methods were not generally applicable and not suitable for a comparison that involved more general fuzzy sets.

## IV. Simultaneous Defuzzification as an Optimization Problem

### A. Formalization of the Problem

We consider $n$ fuzzy sets $A_i, i = 1..n$ over the domain $\mathbb{R}$. The simultaneous constrained defuzzification $D'$ is then defined as $D' : \wp(\mathbb{R})^n \to \mathbb{R}^n, (A_1,\ldots A_n) \mapsto (a_1,\ldots a_n)$, with $\wp(\mathbb{R})$ the notation for the fuzzy powerset of $\mathbb{R}$ (set of fuzzy sets on the domain $\mathbb{R}$). In line with the notation of traditional defuzzifiers, we will denote the obtained value for each $A_i$ as $D'(A_i)$. The constraint in the simultaneous defuzzification $D'$ means that the values $D'(A_i)$ satisfy

$$\sum_{i=1}^{n} D'(A_i) - c = 0 \tag{3}$$

with $c$ the known constraining value. There is a benefit to consider the problem as a translation from an existing defuzzified value $D$: Having the value of an underlying defuzzifier allows to also consider the size of the translation as an additional criterion besides the membership grades, this offers more freedom in defining what constitutes *better* solutions. A second benefit is that, for numerical algorithms, the underlying defuzzified values also provide suitable initial values. Therefore, the constraint is rewritten using translations from this known defuzzified value

$$\sum_{i=1}^{n}(D(A_i) + x_i) - c = 0 \tag{4}$$

with $D(A)$ denoting the defuzzified value of $A$ using a known defuzzifier. For different datasets, the constraint may be different (e.g., for temperatures or concentration of pollutants this could be an average or weighted average instead of a sum); in general, it constitutes a function $f$ on the defuzzified values, which are the sum of an underlying defuzzifier and a translation. As such, $f : \wp(\mathbb{R})^n \times \mathbb{R}^n \to \mathbb{R} : (A_1,\ldots,A_n,x_1,..,x_n) \mapsto f(A_1,\ldots,A_n,x_1,..,x_n)$. The constraint is defined as

$$f(A_1,\ldots,A_i,\ldots,A_n,x_1,\ldots,x_n) = 0. \tag{5}$$

As the constraining function can have quite an impact on possible properties, from now on assume that the contribution of each $x_i$ is equal

$$\begin{aligned}\forall i,j,\forall\epsilon\in\mathbb{R} : f(A_1,\ldots,A_i,\ldots,A_n,x_1,\ldots x_i+\epsilon,\ldots,x_n)\\ = f(A_1,\ldots,A_i,\ldots,A_n,x_1,\ldots x_j+\epsilon,\ldots,x_n).\end{aligned} \tag{6}$$

This is true for the sum-constraint (4), on which the discussion will mainly be focussed.

### B. Goal Functions

The problem of simultaneous defuzzification was translated to a problem of finding $n$ values of $x_i, i = 1..n$ that satisfy a constraint of the form (5). In the example, $f$ sums translations of the underlying defuzzifications of the $n$ fuzzy sets $A_i, i = 1..n$.

In general, there will be many combinations of $x_i$ that will meet the constraint. To define what comprises a *better* solution for the problem, a goal function or objective function is introduced: This function allows to compare and rank different solutions by calculating a quantitative measure for the solution. With this goal function the problem is approached as a minimization (or maximization) problem.

A goal function can make use of the fuzzy sets involved as well the known defuzzified values. In this contribution, we will introduce four goal functions (or objective functions), that use different criteria to evaluate and rank solutions. This ranges from the goal function G.1 which aims to maximize the membership grades of the solutions, G.2 which minimizes the size of the greatest translation and G.3 and G.4 which use an integral to combine the first two criteria. The following goal functions $g : \wp(\mathbb{R})^n \times \mathbb{R}^n \to \mathbb{R} : (A_1,\ldots, A_n, x_1, .., x_n) \mapsto g(A_1,\ldots, A_n, x_1, .., x_n)$ are considered:

**G.1** $g(A_1,\ldots, A_n, x_1, .., x_n) = \min_i(A_i(D(A_i) + x_i))$;

**G.2** $g(A_1,\ldots, A_n, x_1, .., x_n) = \max_i(|x_i|)$;

**G.3** $g(A_1,\ldots, A_n, x_1, .., x_n) = \sum_i |\int_{D(A_i)}^{D(A_i)+x_i} b - A_i(x)dx|$;

**G.4** $g(A_1,\ldots, A_n, x_1, .., x_n) =$

$$\sum_i \left| \int_{D(A_i)}^{D(A_i)+x_i} h(x) - A_i(x)dx \right|, \text{ with } h(x)$$

$$= \begin{cases} |x - D(A_i)|/(D(A_i) - x^i_{\text{inf}}) + 1 & \text{if } x < D(A_i) \\ |x - D(A_i)|/(x^i_{\text{sup}} - D(A_i)) + 1 & \text{if } x > D(A_i) \\ 0 & \text{if } x = D(A_i). \end{cases}$$

The goal function G.1 is the only one that creates a maximization problem (easily rewritten as a minimization problem of complements), whereas the others constitute minimization problems. The goal function G.1 provides for finding a solution that maximizes the lowest membership (as in [13]), reflecting the notion that values with higher membership are *better* values. The underlying defuzzifier $D$ may however not share this notion: The defuzzified value using COG, e.g., is not guaranteed to have the highest membership value. This creates an argument for another goal function that looks solely at the value of the defuzzifier: Goal function G.2 minimizes the largest $|x_i|$, reflecting that *better* means *as close as possible to the values of the underlying defuzzifier*, independent of the membership function. This goal function has the effect that all determined $x_i$ will be equal.

The goal function G.3 attempts to combine aspects of G.1 with G.2 by minimizing the total area between the membership functions and a constant function $x = b$ with $b > 1$. [1]. The value of $b$ was arbitrarily set to $b = 2$.

The goal function G.4 is a generalization of G.3, using the area under functions other than the constant function. This was initially considered with $h(x) = |x - D(A_i)| + 1$ to improve the behavior of G.3 in practical situations (see further in Section V-B), but it made $h(x)$, and thus $g(x)$ dependent on the values of the domain (the impact of a same deviation from $D$ differed with the domain values). To overcome this, an additional scaling was added. The properties of this goal function can widely vary with the definition of $h(x)$, in this article we only consider $h(x)$ as defined in G.4.

The goal functions G.2, G.3, and G.4 (for the presented $h$) are—independent of the defuzzifier—strictly decreasing in the arguments $x_i : x_i < 0$ and strictly increasing in $x_i : x_i > 0$. This does not hold for G.1.

Fig. 2, the three fuzzy sets from Section II-C, along with the values of the four goal functions, referenced to CoG and MeOM. The same pattern is observed for all fuzzy sets: G.1 appears as the complement of the membership function, G.2 increases linearly away from the reference (CoG respectively, MeOM), G.3 and G.4 both increase progressively away further from the reference point, with the latter increasing slower initially, but surpassing the former at some point.

First, some specific aspects of the simultaneous constraint defuzzification will be considered in Section V, before moving to general defuzzification properties in Section VI.

## V. Properties of the Simultaneous Defuzzifier and Goal Functions

In this Section, some properties of interest of the defuzzifier $D'$ and its solution $(D'(A_1),\ldots, D'(A_n)) = (a_1,\ldots, a_n)$ are considered; the properties relate to the defuzzified values and are verified for the underlying defuzzifiers MeOM and CoG in combination with the goal functions described above. The arguments of the defuzzification are assumed to be continuous, but not necessarily convex fuzzy sets. To improve the structure and readability of the article, the proofs were delegated to the appendix.

### A. Unidirectionality of the Solution

This property states that none of the defuzzified values shift in opposite directions in order to meet the constraint: for a solution $(a_1,\ldots, a_n) : \forall i : a_i \geq 0 \cup \forall i : a_i \leq 0$.

For compliance with unidirectionality to be possible, the constraining function $f(A_1,\ldots, A_n, x_1,\ldots, x_n)$ has to be either increasing or decreasing in all its arguments $x_i$; this is covered by the assumption 6. Simple counterexamples suffice to show that unidirectionality is not fulfilled for G.1 with either CoG or MeOM. The property is fulfilled for Sections G.2, G.3, and G.4, both for MeOM and CoG (Appendix III-A).

### B. Uniqueness of the Solution

This property states that the solution that optimizes the goal function, i.e., the *best* solution, is unique. The uniqueness property consequently depends on whether or not the goal function has a *single* minimum (respectively, maximum) in case of a minimization (respectively, maximization). For the goal functions G.1, G.3, and G.4, counterexamples can be designed to show that the property is not met (Appendix, Section III-B). To prove

[1]If $b = 1$ and there is a nondegenerate interval in $\text{core}(A_i)$, then all $x_i$ in this interval will result in the same area for $A_i$ and the goal function will not be able to distinguish these cases; values $b > 1$ allow for them to be distinguished.

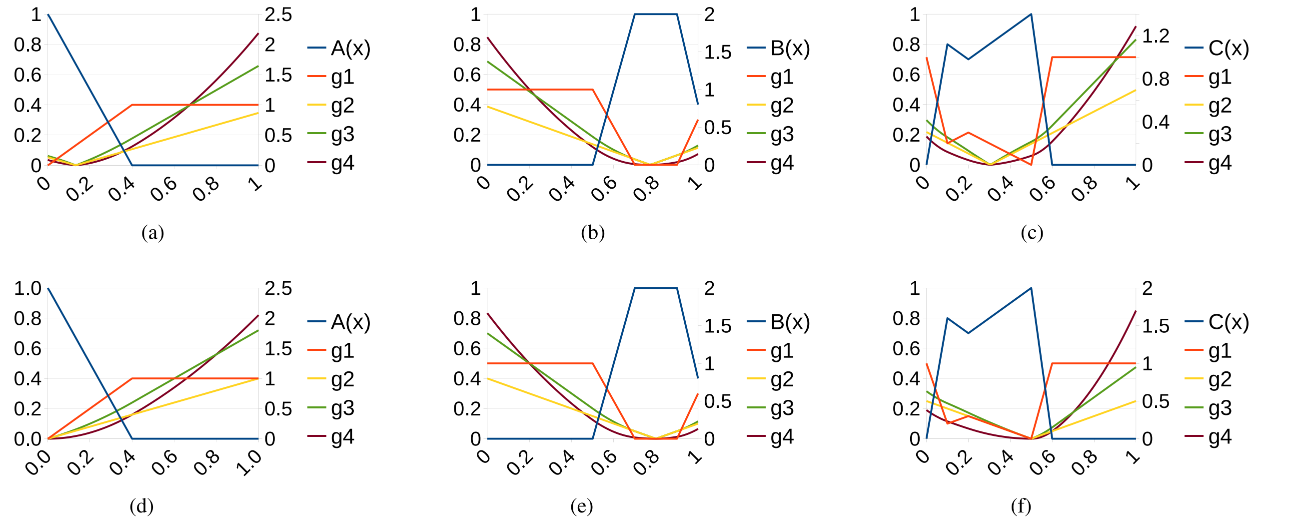



Fig. 2. Fuzzy sets with goal value contributions, in function of CoG and MeOM; membership grades along the left axes, goal values along the right axes. (a) A, CoG. (b) B, CoG. (c) C, CoG. (d)A, MeOM. (e) B, MeOM. (f) C, MeOM.

the uniqueness with the goal function G.2, it is sufficient to show that the goal of minimizing the maximum of $(|x_1|, \ldots, |x_n|)$ implies that all values $x_i$ are equal (Appendix III-A) and that there cannot be two different solutions.

While most of the goal functions do not satisfy the criterion, their behavior in practice is different. When we consider the counterexample with a sum-constraint equal to 10 and trapezoidal fuzzy sets $A_1$ and $A_2$ both defined by the points $\{(0,0),(4,1),(6,1),(10,0)\}$, there are infinite solutions $a_1$ and $a_2$ for which $\sum_{i=1}^{2}(D(A_i)+a_i)=10$, and for which the goal function G.1 is maximized (equal to 1). This is for both CoG or MeOM as underlying defuzzifier. The same example can be used to show that G.3 also does not guarantee a unique solution, as there is a range in which translations $x_1$ and $x_2$ yield the same result for the goal function. Noncompliance with this property was the main argument to introduce G.4, which for this example does yield a unique solution. It still is possible however to construct an example where the fuzzy sets have the same change of area for the same translation of $x_i$ (e.g., $\{(0,0),(4.5,1),(5,0.5),(5.5,1),(10,0)\}$), so uniqueness is in general also not guaranteed with G.4. However, these counterexamples are more artificial and less likely to occur in normal usage than, e.g., fuzzy sets with a constant core or part. As such, while mathematically also not guaranteeing a unique solution, in practice G.4 has less issues with the occurrence of multiple solutions. Other definitions of $h$ could be considered to further decrease the likelihood of multiple solutions in specific applications. If if uniqueness in a general sense is not satisfied, it can still be valuable to consider the property for a given input (set of fuzzy sets) and a selected goal function.

### C. Neutrality of the Operation

The simultaneous defuzzification is considered neutral if a zero solution is returned provided such a solution exists. This can be expressed as $f(A_1,\ldots,A_n,0,\ldots,0))=0 \Rightarrow D'(A_1,\ldots,A_n)=(0,\ldots,0)$.

Compliance with neutrality is not dependent on the constraining function $f$, but on the goal function reaching its minimum (respectively, maximum) when $\forall i: x_i=0$ if the goal is to minimize (respectively, maximize). Neutrality is not guaranteed for G.1 in combination with either defuzzifer, but can be fulfilled for both defuzzifiers in combination with the goal functions G.2, G.3, and G.4, provided the uniqueness property is fulfilled (thus, $(0,\ldots,0)$ is the only minimum; Appendix: III-C).

### D. Guaranteed Support-Selection

Support-selection states that the algorithm will return solutions from the support of each fuzzy set, provided such a solution exist. Formally, $\exists x_{i,i=1..n}: f(A_1,\ldots,A_n,x_1,\ldots,x_n)=0 \wedge \forall i: A_i(D(A_i)+x_i)>0 \Rightarrow (a_1,\ldots,a_n): \forall i: A_i(D(A_i)+a_i)>0$, with $(a_1,\ldots,a_n)$ the notation for the solution that minimizes the goal function. Support-selection is easy to prove for G.1 and easy to disprove for G.2. For G.3 and G.4, counterexamples show the noncompliance; restricting the arguments to fuzzy sets whose supports are intervals (and a solution exists within these intervals) is sufficient for satisfying support selection with both these goal functions.

## VI. General Properties of Defuzzifiers

In this section, the combined defuzzification will be tested against axioms for defuzzification strategies, as presented in [10]. These axioms were described with normal defuzzifiers in mind, and as such variations to make them applicable on the constraint defuzzification are introduced where necessary.

### *A. Prerequisites for Considering the Evaluating the Constrained Defuzzification*

The authors in [10] presented a set of axioms for defuzzification strategies, which serve to objectively evaluate defuzzifiers under different changes to the fuzzy sets. In order to verify these axioms for the simultaneous defuzzification problem where multiple sets are defuzzified, we will consider the validity of an axiom for each participating individual fuzzy set or—if necessary—consider appropriate variations of the axiom. Key aspects that may influence the behavior are the defuzzifier used, the constraining condition, and the goal function. In this section, the combined defuzzification will be tested against these axioms or variations thereof. The defuzzifiers MeOM and COG are selected as two typical examples of defuzzifiers with different properties (Section III-A). They are combined with the four goal functions as defined in Section IV-B; no constraints apart from those imposed by the problem are added.

### *B. Axioms for Defuzzification Strategies*

The axioms from [10] are briefly repeated, as we intend to use them as basis to evaluate the simultaneous defuzzification of multiple fuzzy sets under a shared constraint. The notations as before are used, the support interval is denoted $[x_{\inf}, x_{\sup}]$.

**A.1** Basic constraints

**A.1.1** Zero element: For a membership function that assigns a constant degree $\alpha$ to all elements of its support interval, $D(A) = \frac{x_{\inf}+x_{\sup}}{2}$.

**A.1.2** One element: For a membership function that has a single nonzero element $x$, $D(A) = x$.

**A.1.3** Monotony: defined by four subproperties (*removing* means lowering membership grades; *adding* means increasing membership grades).

**1)** *removing* a part of $A$ right from $D(A)$ does not move $D(A)$ to the right.

**2)** *adding* a part to $A$ right from $D(A)$ does not move $D(A)$ to the left.

**3)** *removing* a part of $A$ left from $D(A)$ does not move $D(A)$ to the right.

**4)** *adding* a part to $A$ left from $D(A)$ does not move $D(A)$ to the left.

**A.2** Graphically motivated constraints

**A.2.1** Symmetry: the relative position of $D(A)$ does not vary if the orientation of the support interval changes.

**A.2.2** x-Translation: the relative position of $D(A)$ stays constant when $\mu_A$ is moved to the left or right.

**A.2.3** x-Scaling: the relative position of $D(A)$ stays constant when scaling the support interval by a constant value.

**A.2.4** $\mu$-Translation (offset): adding a constant offset to the membership function yields two possibilities.

**1)** Strong: $D(A)$ stays constant.

**2)** Weak: $D(A)$ moves towards the mean of the support interval.

**A.2.5** $\mu$-Scaling: $D(A)$ stays constant when multiplying all membership values of $A$ with a constant value.

**A.3** Constraints motivated by fuzzy operations

**A.3.1** T-norm: $D(A) \leq D(B) \Rightarrow D(A) \leq D(T(A, B)) \leq D(B)$.

**A.3.2** T-conorm: $D(A) \leq D(B) \Rightarrow D(A) \leq D(S(A, B)) \leq D(B)$.

**A.3.3** *The axioms relating to hedges are only considered for monotonous functions; as general fuzzy sets are considered here, these axioms are not applicable.*

**A.4** Constraints related to specific applications

**A.4.1** For fuzzy numbers ([14], [15]) $A$: $A(D(A)) = 1$.

**A.4.2** Permissible zones to exclude solutions from nonpermissible zones: $A(D(A)) \geq \alpha$, for a given $\alpha \in ]0, 1]$.

The authors in [5] have investigated properties of various defuzzifiers in general domains, revisited several axioms, and added new properties that also have meaning on the real domain, such as the core selection ($D(A) \in \text{core}(A)$) and continuity (small variation in membership implies small change in defuzzification). For example, for the defuzzifiers MeOM and COG, it can be shown that both satisfy all basic axioms (A.1), the core selection criteria is not guaranteed for COG, but is for MeOM on convex sets, continuity on the other hand is fulfilled by COG but not by MeOM.

### *C. Extending the Axioms for Simultaneous Defuzzificiation*

To consider the axioms for simultaneous defuzzification, we verify the axioms with respect to the defuzzification as determined by the optimization, which for each fuzzy set $A_i$ is $D(A_i) + a_i$, denoted $D'(A_i)$. An axiom is considered fulfilled if it is satisfied for all fuzzy sets in the argument as follows:

1) without changing the constraint, or;
2) when it is possible to modify in accordance with the modification of argument(s).

The axioms A.3.1 and A.3.2 relate the defuzzification of a fuzzy norm/conorm applied on fuzzy sets with the individual defuzzification of the fuzzy sets. The simultaneous defuzzification takes sets of fuzzy sets as arguments, and while it is possible to, e.g., perform a pair-wise combination, there would be a necessary change on the constraint which is impossible to assess in general; this renders these axioms not applicable on the arguments of the simultaneous defuzzification.

As the problem is considered as an optimization with constraints, it is possible to include additional constraints to allow for searching only those solutions that satisfy the Axiom A.4.2 on nonpermissible zones even if the underlying defuzzifiers do not satisfy this axiom. For the subsequent discussion on compliance with the axioms, no additional conditions are added, even allowing solutions with membership grade zero and explicitly foregoing compliance with Axiom A.4.2.

*1) Zero-Element and One-Element:* Given the nature of the constraint defuzzification (translating underlying defuzzifier result), the possibilities for compliance with these axioms is limited. Looking at any single fuzzy set $A_i$ in a simultaneous defuzzification, it is clear that axiom A.1.1 is guaranteed when the underlying defuzzifier satisfies the axiom A.1.1 and the solution $(a_1, \ldots, a_n)$ is unique and such that $a_i = 0, \forall i$. This property can be weakened: As long as the solution involves no translation for the defuzzified value of constant fuzzy sets in a specific problem, the axiom is satisfied. If there are $m$ fuzzy sets, e.g., $A_i, i = 1..m \leq n$, for which $A_i(x) : [x_{\inf_i}, x_{\sup_i}] \rightarrow [0, 1], x \mapsto \alpha_i$, and the defuzzifier $D$ satisfies the axiom A.1.1, the zero-element axiom will be met as long as the goal function reaches its single minimum (respectively, maximum) in $(0, \ldots, 0, x_{m+1}.., x_n)$ in case of a minimization (respectively, maximization). This formulation makes compliance with zero-element dependent on the arguments, but this provides the necessary condition in which zero-element can be considered.

For both G.1 independent of the underlying defuzzifier, zero-element as defined above is only fulfilled if the underlying defuzzifier satisfies zero-element, all $a_i = 0$ and this solution is unique–which is dependent on the arguments. The solution for G.2 is unique, so here it is sufficient that the underlying defuzzifier satisfies the property and all $a_i = 0$.

For the goal functions G.3 and G.4, the property is fulfilled if the underlying defuzzifier satisfies the criterion and there exists a unique solution $(0.0, \ldots, x_m, x_{m+1}, \ldots, x_n)$ such that $\forall i : m+1..n, \forall x \in [D(A_i), D(A_i) + x_i] : D(A_i) + x < \max_{j:1..m}(A_j(x))$ (Appendix Section IV-A).

Compliance with axiom A.1.2 can be considered in a similar way.

*2) Monotony:* This property for a fuzzy set $A$ states that if elements are removed from the right/left of $D(A)$, the defuzzified value does not shift to the right/left (similar for addition of elements). We introduce the notation $A^*$ for the modification of set a $A$ to which elements were added or removed. Note that $D'(A_i) = D(A_i) + a_i$ was introduced earlier as the notation for the defuzzified value of $A_i$ in the simultaneous defuzzification $D'$, with $a_i$ the outcome of the optimization and $D$ the underlying defuzzifier. Monotony for constraint defuzzification considers the modification of *any single* selected fuzzy set $A_k, 1 \leq k\, leqn$ without changing the other fuzzy sets nor the constraining function $f$ and is described by four similar subproperties as follows.

a) *Removing* a part of $A_k$ right from $D'(A_k)$ does not move $D'(A_k)$ to the right.
b) *Adding* a part to $A_k$ right from $D'(A_k)$ does not move $D'(A_k)$ to the left.
c) *Removing* a part of $A_k$ left from $D'(A_k)$ does not move $D'(A_k)$ to the right.
d) *Adding* a part to $A_k$ left from $D(A_k)$ does not move $D(A_k)$ to the left.

This property is fulfilled for the goal function G.2, with both MeOM and COG (Appendix Section IV-B). For the goal functions G.1, G.3, and G.4), the property is only fulfilled if the considered problem has a single solution (satisfies uniqueness, Section V-B), in which case it is fulfilled both with MeOM and COG (Appendix Section IV-B).

*3) Symmetry, X-Translation, X-Scaling:* These three axioms relate to changes to *x*-axis of a fuzzy set and comment the relative position of the defuzzification. For each, we consider the possibility of changing a single set, or applying the change on all fuzzy sets simultaneously.

*a) Symmetry:* A simple example for symmetry shows that a change of a single fuzzy set has the potential to completely change the problem. Consider two triangular fuzzy sets $A_1$ and $A_2$, both defined by the control points $\{(0.0, 0.0), (4.0, 1.0), (10.0, 0.0)\}$, with the constraining function $f = \sum_{i=1}^{n}(D(A_i) + x_i) - 10 = 0$. MeOM of both fuzzy sets equals 4.0. For all considered goal functions, the solution would be $x_1 = x_2 = 1.0$. When replacing $A_1$ by the symmetrical fuzzy set $A_1^*$ defined by $\{(0.0, 0.0), (6.0, 1.0), (10.0, 0.0)\}$, under the same constraint, the solution with MeOM would be $x_1 = x_2 = 0.0$, yielding 4.0 as defuzzified value for both. To satisfy the symmetry, a modified constraint would have to be such that $x_1 = -1.0, x_2 = 1.0$, yielding 3.0 and 5.0 as defuzzified values. However, this solution does not maximize the goal function G.1, nor does it minimize the goal functions G.2, G.3, and G.4. The reasoning with COG is similar. As such, we will only consider symmetry in the context of constraint defuzzification for reversing *all* sets at the same time and with an appropriate modification of the constraining function.

Therefore, symmetry for the presented constraint defuzzification is described as follows: If all fuzzy sets $A_i(x)$ are replaced by their symmetrical counterpart $A_i^*(x) = A_i(\inf + \sup - x)$, the sum constraint is modified in a similar way to $f(A_1^*, \ldots, A_i^*, \ldots, A_n^*, x_1^*, \ldots, x_i^*, \ldots, x_n^*)$ with $x_i^* = (D(A_i) - D(A_i^*) - x_i)$ and the underlying defuzzifier satisfies symmetry (axiom A.2.1), then symmetry is fulfilled. One caveat is that G.1, G.3, and G.4 do not satisfy unicity in general (Section V-B): even though the problem is equivalent, there is no certainty that the optimization will converge to the symmetrical solution. As such, the given problem will be required to satisfy unicity (Section V-B) with the presented arguments, in order to guarantee the modified problem will converge to the symmetrical solution. Under these conditions, all considered goal functions satisfy symmetry, both with MeOM and COG (Appendix Section IV-C).

*b) x-translation:* *X*-translation can be considered for a single fuzzy set within a constrained defuzzification, if the underlying defuzzifier satisfies *x*-translation and the constraining function allows the translation of arguments, i.e., $\forall i : f(A_1, \ldots, A_n, x_1, \ldots, x_{k-1}, x_k + \delta, x_{k+1}, \ldots, x_n) = f(A_1, \ldots, A_n, x_1, \ldots, x_{k-1}, x_k, x_{k+1}, \ldots, x_n) + \delta$. The four goal functions considered in the article (Section IV-B) in combination with MeOM and COG as underlying defuzzifier and the sum constraint satisfy *x*-translation. This means that if one fuzzy set $A_k$ is replaced by $A_k^* = A_k + \delta$ and the constraining function is defined as $f(A_1, \ldots, A_{k-1}, A_k^*, A_{k+1}, \ldots, A_n, x_1, \ldots, x_{k-1}, x_k^*, x_{k+1}, \ldots, x_n)$ with $A_k^* = A_k + \delta$ and $x_k^* = x_k - \delta$, the relative position of the defuzzifiers is the same (Appendix Section IV-D).

The same caveat for G.1, G.3, and G.4 caused by noncompliance with uniqueness (Section V-B) and present in symmetry is also is present here, thus there is no guarantee that the optimization will converge to the solution with the same relative position of defuzzifier unless the problems is such that the solution is unique.

*c) x-scaling:* It is possible to consider *x*-scaling for a single fuzzy set within a constrained defuzzification, similar to *x*-translation. In the example, if the underlying defuzzifier satisfies *x*-scaling and the argument to the constraining function is scaled with the inverse of the scaling factor of the fuzzy set, the property

TABLE II
SUMMARY OF THE COMPLIANCE WITH THE PRESENTED PROPERTIES

| | G.1 | | G.2 | | G.3 | | G.4 | |
|---|---|---|---|---|---|---|---|---|
| Axiom | MeOM | COG | MeOM | COG | MeOM | COG | MeOM | COG |
| Unidirectionality | no | no | yes | yes | yes | yes | yes | yes |
| Uniqueness | no | no | yes | yes | partial[1] | partial[1] | partial[1] | partial[1] |
| Neutrality | yes[2,3] | no | yes[3] | yes[3] | yes[3] | yes[3] | yes[3] | yes[3] |
| Support-selection | yes | yes | no | no | yes[4] | yes[4] | yes[4] | yes[4] |
| Zero-Element | no[3,5] | no[3,5] | no[3,5] | no[3,5] | no[5] | no[5] | no[5] | no[5] |
| One-Element | no[3,5] | no[3,5] | no[3,5] | no[3,5] | no[5] | no[5] | no[5] | no[5] |
| Monotony | —[6] | | —[6] | | —[6] | | —[6] | |
| Symmetry | yes[3,7,8] | yes[3,7,8] | yes[3,7,8] | yes[3,7,8] | yes[3,7,8] | yes[3,7,8] | yes[3,7,8] | yes[3,7,8] |
| $x$-translation | yes[3,7,9] | yes[3,7,9] | yes[3,7,9] | yes[3,7,9] | yes[3,7,9] | yes[3,7,9] | yes[3,7,9] | yes[3,7,9] |
| $x$-scaling | yes[3,7,8] | yes[3,7,8] | yes[3,7,9] | yes[3,7,9] | yes[3,7,9] | yes[3,7,9] | yes[3,7,9] | yes[3,7,9] |
| $\mu$-translation | yes[3,8] | no | yes[3,9] | no | yes[3,8] | no | yes[3,8] | no |
| $\mu$-scaling | yes[3,8] | no | yes[3,9] | no | yes[3,8] | no | yes[3,8] | no |
| Note | [1] Uniqueness has to be considered in the context of a given input.<br>G.4 is more likely to support unicity than G.3 in practical applications.<br>[2] For convex fuzzy sets<br>[3] If uniqueness is fulfilled<br>[4] If the support is an interval<br>[5] Yes only under very strict conditions and not generally<br>[6] Not applicable<br>[7] Requires an appropriate change of the constraining function<br>[8] Only when changing all sets in the same way<br>[9] It is possible to change a single set | | | | | | | |

can be considered. It is however only fulfilled for G.1, as this is the only goal function that is not impacted by a change in $x$-scale, but only if it satisfies uniqueness (Section V-B). In the other goal functions, the changed $x$-values of a single set changes the evaluation of the goal function and makes the solver converge to a different solution.

Similarly to symmetry, we can also consider the scaling of all fuzzy sets at the same time, using the same scaling factor. If all fuzzy sets $A_i(x)$ are replaced by their scaled counterpart $A_i^*(x) = A_i(\delta x)$, and the sum constraint is modified in a similar way to $f^*(x_1^*, \ldots, x_i^*, \ldots, x_n^*)$ with $x_k^* = D(A_k)(1-\delta) + x_k$, then $x$-scaling is met for the four goal functions, both with MeOM and CoG (Appendix Section IV-E). The same remark regarding unicity for G.1, G.3, and G.4 as for symmetry and $x$-translation holds.

*4) $\mu$-Translation, $\mu$-Scaling:* These axioms describe how the defuzzifier behaves with global modifications of the membership grades; in the weak variants, a shift of the defuzzified value towards the mean is considered. Performing a $\mu$-translation or $\mu$-scaling on a single fuzzy set changes the constraint defuzzification problem in an unpredictable way: the underlying defuzzifier can shift (which in turn affects the solution to meet the constraint as well as the behavior of the goal function), but in addition also the membership grades change (potentially impacting the behavior of the goal function). The goal function G.2 does not make use of the membership grades at all, and as such its evaluation is not affected by the change of a single fuzzy set.

The constraint defuzzification using the goal function G.2 in combination with an underlying defuzzifier that satisfies $\mu$-translation (respectively, $\mu$-scaling), satisfies $\mu$-translation (respectively, $\mu$-scaling), even when applied on a single fuzzy set. Both $\mu$-translation and $\mu$-scaling are satisfied for MeOM; neither for COG (which does not satisfy either criterion).

A variation of $\mu$-translation and $\mu$-scaling in which all fuzzy sets undergo the same transformation (translation or scaling) can be also considered if the underlying defuzzifier satisfies those axioms–this is the case for MeOM but not for COG. Translating or scaling all fuzzy sets equally when using the goal function G.1 in combination with MeOM, can still yield the same solution, however the solver is not guaranteed to converge to this solution if the problem does not have a unique solution. The remaining goal functions G.3 and G.4 in combination with MeOM satisfy $\mu$-translation (if the problem has a unique solution), but not $\mu$-scaling (Appendix Section IV-F, IV-G).

### D. Summary

Compliance of the different combinations of goal functions and defuzzifiers with the different presented properties is summarized in Table II. For some axioms, the proofs were based on specific properties of the goal functions. Similarly, some proofs are independent of the underlying defuzzifiers. As such, some generalizations are possible: Table III lists an overview of necessary and sufficient conditions.

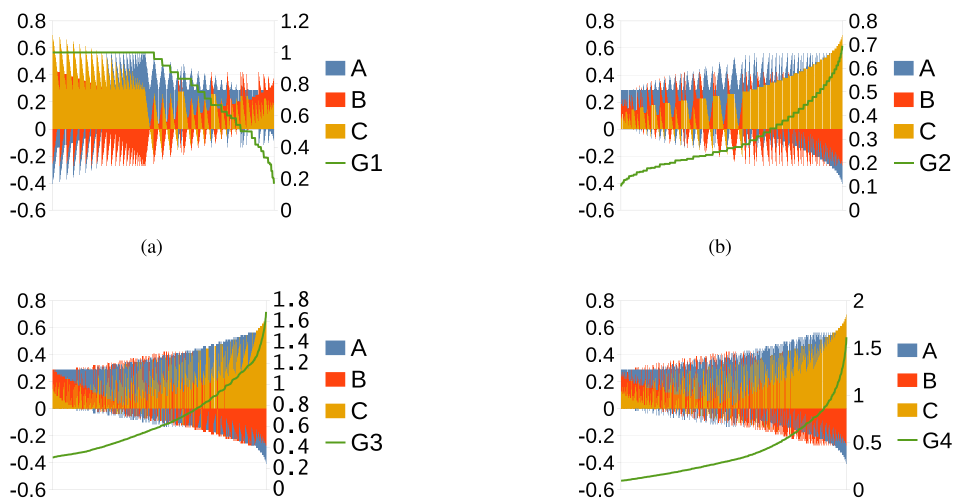
## VII. EXAMPLE

We will now revisit the example from Section II-C using the constrained defuzzification. The values of the different goal functions for the fuzzy sets are plotted on Fig. 2.

To illustrate the behavior and impact of the goal functions, we implemented a brute force approach that considers all possible combinations of 60 evenly spread samplepoints in the domain [0,1] of each fuzzy set and only retains those combinations for which the values are in the support of their respective fuzzy sets, and the constraint is met. This yielded 774 solutions, characterized by the translations from the underlying

TABLE III
MINIMAL AND SUFFICIENT CONDITIONS FOR COMPLIANCE WITH THE AXIOMS

| Axiom | Generalizations |
|---|---|
| Unidirectionality | Fulfilled if the goal function is decreasing (respectively, increasing) left (respectively, right) of the underlying defuzzified value, compliance is independent of the underlying defuzzifier |
| Uniqueness | Compliance is independent of the underlying defuzzifier |
| Neutrality | Cannot be fulfilled if uniqueness is not satisfied (for the goal function, defuzzifier and fuzzy sets) fulfilled if the goal function reaches its optimum at the values for the underlying defuzzifier (minimum in case of a minimization problem, maximum in case of a maximization problem) |
| Monotony | Fulfilled for G.3,G.4 if the underlying defuzzifier satisfies monotony *and* has a single solution |
| Symmetry | Cannot be fulfilled if uniqueness is not satisfied *and* the underlying defuzzifier does not satisfy symmetry |
| $x$-translation | Fulfilled if the goal function is nonnegative *and* continuous |
| Note | All these properties are under the assumption that the contribution to the constraint is equal, Section IV-A, (6) |




(a) (b) (c) (d)

Fig. 3. $x$-axis lists 774 solutions that meet the constraint, ordered by best goal value using CoG as underlying defuzzifier. The left $y$-axis shows the individual and cumulative translation (which equals 0.2, see Section II-C, Table I) of each center of gravity of A,B, and C. The goal value is referenced to the right $y$-axis. (a) CoG, G.1. (b) CoG, G.2. (c) CoG, G.3. (d) CoG, G.4.

defuzzifiers. These results where subsequently ordered according to increasing goal value (except for G.1, which is by decreasing goal value). Solutions with the same goalvalue were further ordered by increasing size of translation of $D(A)$, then increasing size of translation of $D(B)$ and lastly increasing size of translation of $D(C)$. This approach results in the plots on Fig. 3 when using CoG as underlying defuzzifier; with the $x$-axis listing the solutions from better to worse.

The figures clearly illustrate the uniqueness property (Section V-B): For G.1 there are many solutions that have the maximum goal value, whereas the others have a single solution with best goal value for this example. Similarly, noncompliance with unidirectionality (Section V-A) of G.1 is visible in solutions with maximum goal value where the contribution of some fuzzy sets is positive and that of others is negative. The plots for MeOM reveal a similar behaviour and are shown on Fig. 4– here the cumulative translation is equal to 0.29 (see Section II-C, Table I). Apart from the change in values, the general trend of the goal functions is similar to the previous case.

An overview of the best solutions with each of the different goal functions and underlying defuzzifiers is provided on Table IV. The reported solutions are the first solutions returned by the defuzzifiers; in case of G.1, this is just one of the many solutions as illustrated on Figs. 3 and 4. The table further confirms

TABLE IV
DEFUZZIFIED VALUES (COG AND MEOM) OF THE FUZZY SETS IN THE EXAMPLE

| | COG | | | | MeOM | | | |
|---|---|---|---|---|---|---|---|---|
| | g1* | g2 | g3 | g4 | g1* | g2 | g3 | g4 |
| $D'(A)$ | 0.06 | 0.23 | 0.13 | 0.16 | 0.06 | 0.07 | 0.06 | 0.09 |
| $D'(B)$ | 0.92 | 0.87 | 0.91 | 0.89 | 0.92 | 0.87 | 0.92 | 0.89 |
| $D'(C)$ | 0.51 | 0.40 | 0.46 | 0.44 | 0.51 | 0.57 | 0.51 | 0.52 |
| Sum | 1.50 | 1.50 | 1.50 | 1.50 | 1.50 | 1.50 | 1.50 | 1.50 |
| $\Delta D(A)$ | -0.07 | 0.10 | 0.00 | 0.03 | 0.06 | 0.07 | 0.06 | 0.09 |
| $\Delta D(B)$ | 0.15 | 0.10 | 0.13 | 0.12 | 0.12 | 0.07 | 0.12 | 0.09 |
| $\Delta D(C)$ | 0.21 | 0.10 | 0.16 | 0.14 | 0.01 | 0.07 | 0.01 | 0.02 |
| Note | * The solution is not unique; the column lists the first solution found by the brute force algorithm. | | | | | | | |

The translated values to meet the constraint are also shown.

that G.2 equalizes all the translations. The values obtained by simply scaling the underlying defuzzified values (Table I) resemble the values obtained with G.2; this is because the supports of the fuzzy sets are quite similar in length, resulting in a relatively equal scaling. On this small example, when compared to G.3, G.4 has the tendency to lower the difference between the smallest and largest translations of the different fuzzy sets. This makes sense as G.4 penalizes progressively more as the translation increases. As with all defuzzification methods, different methods yield

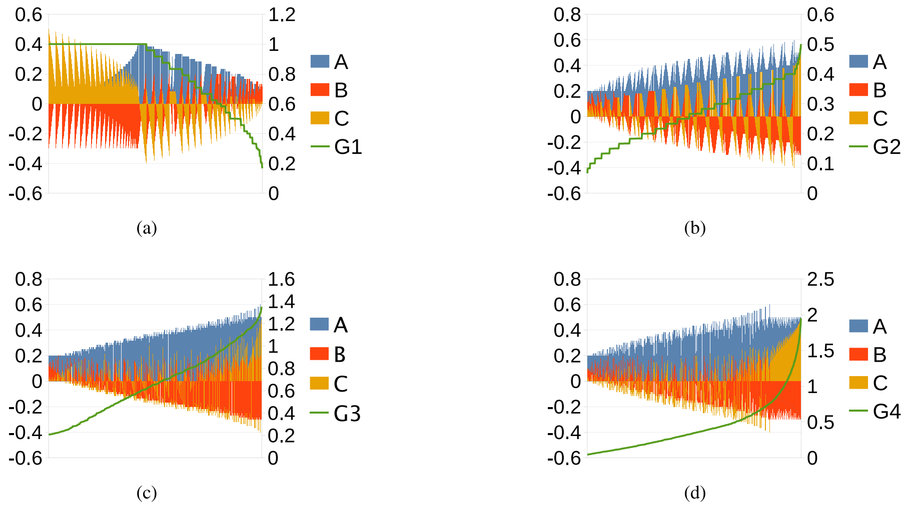



Fig. 4. *x*-axis lists 774 solutions that meet the constraint, ordered by best goal value using MeOM as underlying defuzzifier. The left *y*-axis shows the individual and cumulative translation (which equals 0.2, see Section II-C, Table I) of each center of gravity of A,B, and C. The goal value is referenced to the right *y*-axis. (a) MeOM, G.1. (b) MeOM, G.2. (c) MeOM, G.3. (d) MeOM, G.4.

different solutions and an appropriate method has to be chosen to fit the interpretation.

## VIII. Conclusion

A specific problem of finding crisp values in a number of connected general type-1 fuzzy sets was presented. This problem effectively constitutes a defuzzification of multiple fuzzy sets under a constraint, and was viewed as an optimization problem. A solution was determined by a translation from an exisiting defuzzifier; with a goal function to allow the evaluation and comparison of solutions. Four goal functions were proposed, each combined with two underlying defuzzifiers (CoG and MeOM). The behavior as a defuzzifier was studied and common properties were verified and (dis)proven. A simple implementation supported the evaluation of a small example to illustrate the findings. The quality of the defuzzified values is application dependent and depends on the underlying defuzzifier as well as on the extent each of the defuzzified values is shifted away from this underlying value. This later element is steered using the choice of the goal function. As such, the article presents a range of combinations (goal functions and underlying defuzzifiers) that allow one to select the most appropriate combination for a given problem. The knowledge on compliance/noncompliance with the axioms should help with this choice; the generalizations further support the creation of other goal functions and the adoption of other defuzzifiers. Future work is aimed at implementation aspects, in particular at studying the various aspects related to the application of numerical solvers (convergence, setting starting parameters, etc.) to improve on the brute force approach (which suffers drawbacks such as discretization of the search-space, slow calculation). Exploring type-2 fuzzy sets is also considered as a future generalization with a particular focus on exploiting the additional uncertainty.

## References


[1] J. Verstraete, "The spatial disaggregation problem: Simulating reasoning using a fuzzy inference system," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 3, pp. 627–641, Jun. 2017.

[2] C. Tomlin, "Special issue landscape planning: Expanding the tool kit map algebra: One perspective," *Landscape Urban Plan.*, vol. 30, no. 1, pp. 3–12, 1994.

[3] J. Verstraete, "Solving the general map overlay problem using a fuzzy inference system designed for spatial disaggregation," *Mitigation Adapt. Strategies Glob. Change*, vol. 24, pp. 1573–1596, Jul. 2019.

[4] S. Kikuchi, "A method to defuzzify the fuzzy number: Transportation problem application," *Fuzzy Sets Syst.*, vol. 116, no. 1, pp. 3–9, 2000. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0165011499000330

[5] W. V. Leekwijck and E. E. Kerre, "Defuzzification: Criteria and classification," *Fuzzy Sets Syst.*, vol. 108, no. 2, pp. 159–178, 1999. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0165011497003370

[6] S. Roychowdhury and W. Pedrycz, "A survey of defuzzification strategies," *Int. J. Intell. Syst.*, vol. 16, no. 6, pp. 679–695, 2001. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/int.1030

[7] R. R. Yager and D. P. Filev, "Constrained defuzzification," in *Proc. 5th IFSA Congres - Seoul. IFSA*, 1993, pp. 1167–1170.

[8] R. R. Yager and D. P. Filev, "Defuzzification with constraints," *Fuzzy Log. Appl. Eng., Inf. Sci., Intell. Syst. - Theory Dec. Library*, vol. 16, pp. 157–166, 1996.

[9] A. Asgari Safdar, "Evaluation numbers based on RMS," *Indian J. Fundam. Appl. Life Sci.*, vol. 4, no. S4, pp. 250–254, 2014.

[10] T. A. Runkler and M. Glesner, "A set of axioms for defuzzification strategies towards a theory of rational defuzzification operators," in *Proc. IEEE 2nd Int. Conf. Fuzzy Syst.*, 1993, pp. 1161–1166.

[11] A. Mert, "Shannon entropy-based approach for calculating values of wabl parameters," *J. Taibah Univ. Sci.*, vol. 14, no. 1, pp. 1100–1109, 2020. [Online]. Available: https://doi.org/10.1080/16583655.2020.1804157

[12] D. Z. Saletic and U. Popovic, "On possible constraints in applications of basic defuzzification techniques," in *Proc. 8th Seminar Neural Netw. Appl. Elect. Eng.*, 2006, pp. 225–230.

[13] J. Verstraete, "Algorithm for simultaneous defuzzification under constraints: Shifted mean-max," in *Proc. Conf. Int. Fuzzy Syst. Assoc. Eur. Soc. Fuzzy Log. Technol.*, Atlantis Press, 2015, pp. 326–332.

[14] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning I, II, III," *Inf. Sci.*, vol. 8, no. 3, pp. 199–251, 1975.

[15] J. Dijkman, H. van Haeringen, and S. de Lange, "Fuzzy numbers," *J. Math. Anal. Appl.*, vol. 92, no. 2, pp. 301–341, 1983. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0022247X83902536