An evolutionary technique for supporting the consensus process of group decision making

Miguel Benito-Parejo, Mercedes G. Merayo and Manuel Núñez

Abstract—**Discrepancies arise when experts have to decide the preference on alternatives for a problem. Henceforth, it is necessary to carry out a process during which they adjust their opinions in order to achieve an acceptable consensus. Usually, this process is coordinated by a moderator that makes suggestions to the participants regarding the most adequate changes. In order to simplify this process, we propose an evolutionary technique based on the search of the changes of the preferences that improve the consensus degree. We also consider that the opinions of the experts should stay closer to the original ones. Taking into account these two factors, we are able to provide useful feedback for the experts willing to get a consensus and measure how much improvement can be achieved.**

Index Terms—**Group decision making, Genetic algorithms, Consensus, Consensus moderator support, Fuzzy preferences**

I. Introduction

Collective Intelligence explores how the collaboration of groups of people and computers can be more productive than an individual person or group can be. Among other applications [5], [27], [31]–[34], Collective Intelligence appears in *group decision making* [11], [16], which is based on the integration of the individual knowledge of the members of large collectives [9], [10], [35]. In group decision making, the goal is to reach a high level of consensus among experts that participate in the search of a solution. Basically, the experts express their preferences on a set of alternatives and, using similarity functions, the level of agreement of the experts is estimated. Although consensus should ideally show an unanimous agreement, due to the broad diversity of participants such state may be difficult to be reached in real world situations. This leads to the necessity of carrying out a process that guides the experts toward consensus, bringing their positions closer together. Usually, the *consensus process* is managed by a moderator that try to persuade the experts to modify their opinions. Nevertheless, it does not mean that a full agreement must be reached and this has conducted to *relaxed* definitions of consensus. Among them, it is worth to mention those approaches that introduce a *soft* notion of consensus [19]. This notion captures the concept of *fuzzy majority* [24] in which imprecise measures like *most* or *more than a half* can be considered.

This work has been supported by the Spanish MINECO-FEDER grant FAME (RTI2018-093608-B-C31), the Region of Madrid grant FORTE-CM (S2018/TCS-4314) co-funded by EIE Funds of the European Union and the Region of Madrid - Complutense University of Madrid (grant number PR65/19-22452).

Design and Testing of Reliable Systems research group, Universidad Complutense de Madrid, Spain, {mibeni01,mgmerayo,manuelnu}@ucm.es

Several approaches, based on the concept of fuzzy majority led by a moderator, have been proposed with the objective of modelling the consensus process [6]. Unfortunately, there does not exist a feasible way for determining the *best solution*, that is, a solution that completely satisfies all the experts and reach a high level of consensus. In that sense, metaheuristics excel when we need to find a *good enough approximation* of the best possible solution for a given problem. In this work, we propose an evolutive technique, *genetic algorithms*, to find the *best* proposal to modify the preferences provided by the experts with the aim of improving the consensus degree at each step of the consensus process. Beginning with a random population (set of candidate solutions), this technique evolves into an optimal solution, according to the conditions reflected in the objective function. The algorithm that we have designed takes into account both the distance between the preferences provided by the experts and the potential solution, and the level of consensus. In addition, we present the results of the experiments carried out with the goal of evaluating the performance of our proposal.

The rest of the paper is structured as follows. In Section II we introduce the main concepts related to group decision making. Section III presents the application of genetic algorithms based techniques. Next, Section IV explains in detail our proposal and in Section V we report on the experiments performed to evaluate our proposal. Finally, Section VI presents our conclusions and some lines for future work

II. Group Decision Making

Group decision making (GDM) consists in reaching the consensus of multiple experts, according to their preferences and considering only feasible alternatives, to solve a problem [23]. The motivations, opinions and viewpoints of the experts can be very different, but all of them share a common goal: reach an agreement on the solution [39]. There are two phases that must be performed for obtaining the final solution [17]. The first phase is the *consensus process*, which aims at obtaining the maximum level of agreement among the participants. This is an iterative process, usually coordinated by a moderator [40]. In each iteration, the experts should adjust their preferences, based on the suggestions of the moderator, to make them closer. During this process, the moderator decides whether the level of consensus of the experts is acceptable before carrying out the next phase. The second phase is the *selection process*, whose main goal is to obtain the solution based on the preferences of the experts. Obviously, the greater the level

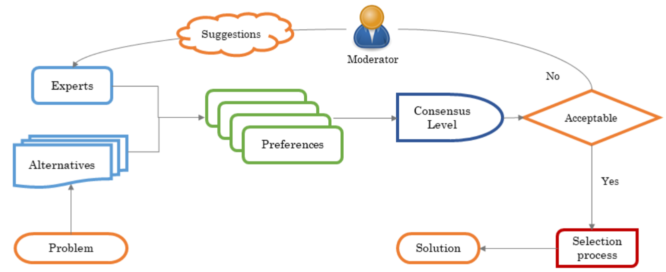

Fig. 1. GDM flowchart

of consensus, the easier it is to select the final solution. Fig. 1 depicts the flow of the GDM process.

The resolution of a GDM problem P consists in the selection of a solution within a set of at least two alternatives. The search for the best solution considers the preferences about the alternatives provided by a set of at least two experts. Although different formats can be used by the experts for representing their assessments, the most usual is the *fuzzy preference relation* [37] due to the fact that it facilitates the aggregation of the evaluations of the experts.

Definition 1: Let $X = \{x_1, \ldots, x_n\}$, with $n \geq 2$, be a set of alternatives. A *fuzzy preference relation* P on X is given by a function $\mu_P : X \times X \to [0,1]$. A fuzzy preference relation can be represented by a matrix $p \in [0,1]^{n\times n}$, such that for all $1 \leq i, j \leq n$ we have $p_{ij} = \mu_P(x_i, x_j)$ and $p_{ij} + p_{ji} = 1$. □

The values stored in the matrix indicate the preference for the alternative x_i over x_j. If $p_{ij} > 0,5$ then x_i is preferred. If $p_{ij} = 1$ then x_i is completely preferred. Finally, if $p_{ij} = 0,5$ then there does not exist preference for any of the alternatives.

Although consensus refers to an unanimous decision, the opinions of the experts tend to differ and, therefore, a general agreement is difficult to reach, specially in real world scenarios with large groups of participants. This fact causes the necessity of defining a consensus process with the goal of reaching a solution. In this paper, we consider that this process is being led by a moderator. In addition, we consider a notion of *soft consensus* concerning the concept of *fuzzy majority*, which is expressed by means of fuzzy linguistic quantifiers such as *at least half*, *almost all* and *as many as possible* [43]. The fuzzy majority allows us to define the consensus degree in a wider range of possible agreements. These quantifiers will be used by the aggregation operators during the aggregation phase. For instance, we will consider the Ordered Weighted Average (OWA) operator [42] that will allow us to establish a consensus degree that helps to determine the level of agreement of the experts and guide the consensus process. Next, we give the formal definition of these concepts.

Definition 2: A function $Q : [0,1] \to [0,1]$ is a *regular monotonically non-decreasing quantifier* if it holds:

- $Q(0) = 0$,
- $Q(1) = 1$, and
- $Q(r_1) \geq Q(r_2)$ when $r_1 \geq r_2$

An *OWA operator* for a quantifier Q is a function $\Phi_Q : \mathbb{R}^n \to \mathbb{R}$ defined as:

$$\Phi_Q(x_1, \ldots, x_n) = \sum_{i=1}^{n} w_i \cdot x_{\rho(i)}$$

and such that for all $1 \leq i \leq n$ we have $w_i \in [0,1]$, $\sum_{i=1}^{n} w_i = 1$ and

$$w_i = Q\left(\frac{i}{n}\right) - Q\left(\frac{i-1}{n}\right)$$

Finally, the function $\rho : \{1, \ldots, n\} \to \{1, \ldots, n\}$ assigns to each $1 \leq i \leq n$ the position of the ith largest value in $(x_1, \ldots, x_n)$. □

In the literature we can find different proposals of consensus models based on soft consensus measures [3], [6], [14], [18], [25], where the level of consensus must be determined during the consensus process and requires to establish consensus measures. These measures must reflect the similarity among the preferences of the experts and the consensus degree at each stage of the process. In our proposal we adopt a consensus model [17] that considers three different consensus degrees related to each of the levels of the fuzzy preference relations: pairs of alternatives, alternatives and the relation. The next definitions formally provide the notions of distance and similarity functions required for computing the consensus degree used in our approach. We also give the definition of the similarity matrix associated to each expert, which reflects the proximity of his/her preferences and the ones corresponding to the rest of participants.

Definition 3: Let $\alpha = (\alpha_1, \ldots, \alpha_\nu), \beta = (\beta_1, \ldots, \beta_\nu) \in \mathbb{R}^\nu$ be two vectors. The *Manhattan measure function* $d : \mathbb{R}^\nu \times \mathbb{R}^\nu \to [0,1]$ is defined as:

$$d(\alpha, \beta) = \sum_{i=1}^{\nu} |\alpha_i - \beta_i|$$

The *similarity function* $s : \mathbb{R}^\nu \times \mathbb{R}^\nu \to [0,1]$ is defined as:

$$s(\alpha, \beta) = 1 - d(\alpha, \beta)$$

Let $E = \{e_1, \ldots, e_m\}$, with $m \geq 2$, be a set of experts and $p^1, \ldots, p^m$ be the corresponding fuzzy preference relations of each of them for a set of alternatives $X = \{x_1, \ldots, x_n\}$. We define the *similarity matrix* $Sm^k \in [0,1]^{n\times n}$ such that Sm^k_{ij} is equal to:

$$\begin{cases} s((p^1_{ij}, \ldots, p^1_{ij}), (p^2_{ij}, \ldots, p^m_{ij})) & \text{if } k = 1 \\ s((p^k_{ij}, \ldots, p^k_{ij}), (p^1_{ij}, \ldots, p^{k-1}_{ij}, p^{k+1}_{ij}, \ldots, p^m_{ij})) & \text{if } 1 < k < m \\ s((p^m_{ij}, \ldots, p^m_{ij}), (p^1_{ij}, \ldots, p^{m-1}_{ij})) & \text{if } k = m \end{cases}$$

for all $1 \leq k \leq m$ and $1 \leq i, j \leq n$. □

On the basis of these consensus measures, we explain how the consensus degree is computed in our framework.

- The OWA operator for a quantifier of interest is applied for aggregating the similarity matrices of all the experts and obtain the *consensus matrix* $cons$.

$$\forall\, 1 \leq i, j \leq n : c_{ij} = \Phi_Q(Sm^1_{ij}, \ldots, Sm^m_{ij})$$

This matrix reflects the agreement of the group on each pair of alternatives.

- The OWA operator is applied again for generating the *consensus on alternatives*.

$$\forall\, 1 \leq i \leq n : calt_i = \Phi_Q(\gamma_1, \gamma_2, \ldots, \gamma_{2\cdot(n-1)})$$

where $\{\gamma_1, \ldots, \gamma_{2\cdot(n-1)}\} = \{c_{ij} | 1 \leq j \leq n \wedge j \neq i\} \cup \{c_{ji} | 1 \leq j \leq n \wedge j \neq i\}$. This measure captures the agreement on each alternative among all the participants. It is obtained from the consensus matrix, aggregating the consensus degrees of the pairs of alternatives related to the corresponding alternative.

- Finally, the consensus on the relation, $consRel$, is computed by applying the OWA operator to the consensus degrees at the level of alternatives:

$$consRel = \Phi_Q(calt_1, \ldots, calt_n)$$

This measure corresponds to the general agreement of the experts that participate in the GDM problem.

III. Genetic Algorithms

Our proposal considers *Genetic Algorithms* (GA) [15], a heuristic optimization technique whose behaviour is based on the processes of evolution in nature. GAs and other optimization algorithms have been used in search optimization problems, where finding the optimal solution is, in practice, unfeasible. Generally, a GA consists of a group of individuals (population), each representing a potential solution to the problem in hand. Therefore, the first issue we need to address is how to represent the potential solutions. An initial population with such individuals is usually selected randomly. Then, a parent selection process is used to pick a few of these individuals. There exist alternative selection models, such as linear rank, remains, roulette wheel, tournament, truncation or stochastic universal. New offspring individuals are produced using *crossover*, keeping some of the features of their parents. The simplest one is known as single point crossover [30]. Then, *mutation*, which injects some new genetic material, is applied. The quality of each individual, that is, how close the individual is to being a good solution, is measured by a *fitness function*, defined for the particular search problem. When ranking is used the population is sorted according to the fitness value of the individuals. Then, each individual is ranked irrespective of the size of it and its predecessors fitness. This is known as linear ranking. It has been shown that linear rankings reduce the chance of a very few fit individuals dominating the search leading to a premature convergence [30]. The population is iteratively recombined and mutated to evolve successive populations, known as *generations*. When the specified termination criterion is satisfied, the algorithm terminates. Depending on the fitness function, there can be different termination criteria for a GA. If the fitness function is such that a solution would produce a known fitness value, then the GA can terminate when an individual with such fitness is generated. In many cases it is not possible to establish this value in advance. Therefore, the GA must be given other termination criteria. For example, we can set a maximum number of generations after which the GA will terminate irrespective of whether a solution has been generated. A flowchart for a simple GA is presented in Fig. 2.

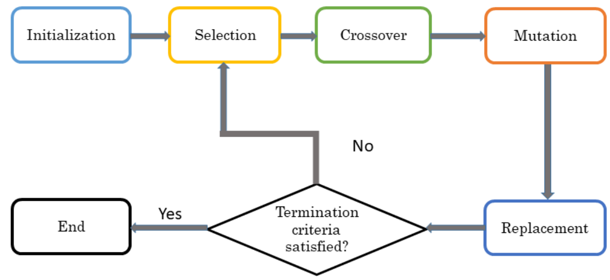

Fig. 2. GA flowchart

IV. Our heuristic proposal

Our approach aims at providing a consensus support tool that generates, during the consensus process, a proposal of modifications to be performed by each expert in their preferences. This approach tries to increase the consensus degree while keeping a close distance between the original opinions and the proposed ones. The main contribution of our approach is the design and application of a GA that, in each step of the consensus process, finds a set of fuzzy preference relations that improve the consensus degree. Next, we explain the specifics that we have used to design our algorithm. We assume that a set of m experts have to express their preferences on a set of n alternatives for solving a GDM problem.

The Population. In our design each chromosome represents a fuzzy preference relation corresponding to an expert that participates in finding the solution of the problem. An individual is composed of a collection of m chromosomes corresponding to a group of experts. Our GA has a population of initial sets of preference relations that will evolve to generate better proposals to be suggested by the moderator. This process stops either when an established threshold of acceptable consensus level is reached or when an established maximum number of iterations is reached.

Initialization Method. We have designed an *incremental initialization*. This provides a variety of individuals, each of them with n fuzzy preference relations, that generates a high level of diversity.

Selection of Population Methods. During the selection phase, the individuals with the highest fitness are selected for reproduction with the goal of transferring their genes to the next generation. The transition from a generation to the next one has to ensure that the representatives of the foremost individuals are selected. The idea is to reward the best ones with more appearances and penalise the worst ones even with no appearances at all. There exist several selection methods to be applied in GAs. In our proposal, we use roulette wheel, truncation and stochastic universal.

CROSSOVER METHODS. The crossover phase recombines the information of the parents to generate new offspring. In our case, we will apply *continuous crossover* (see Fig. 3) that takes two individuals of the population and exchanges the chromosome (fuzzy preference relations) corresponding to a randomly selected expert.

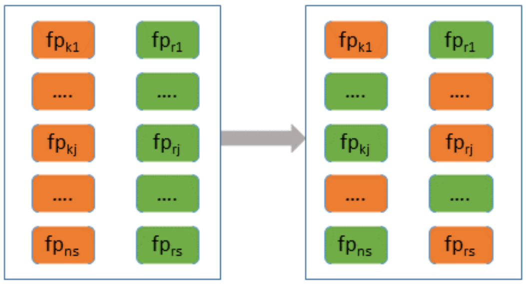

Fig. 3. Continuous crossover

MUTATION OF POPULATION METHODS. The mutation phase is used to keep a diversity from one generation to the next one. As the initial population might not be very diverse, it is sensible to refresh the population with some slight changes that could renew some stale state. In our case, we consider a technique called *replacing mutation* (see Fig. 4). This technique replaces a random number of positions of some of the fuzzy preference relations of the generated individuals. In order to maintain the consistency of the relations, if the value of a position ij is modified then the position ji is changed accordingly.

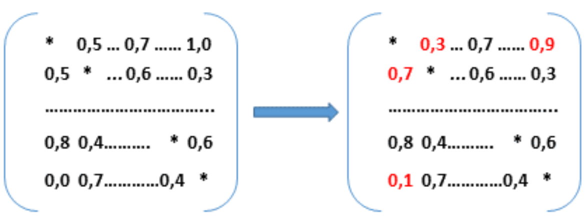

Fig. 4. Replacing Mutation

REPLACEMENT OF POPULATION METHODS. The last step of the GA consists in replacing the current population by the offspring. In this case, we have two options. On the one hand, we have the trivial option that substitutes the last population by the new one, even if it is worse. In this way, less operations are performed at this stage and, as a result, the execution will be faster. On the other hand, a high percentage of the new generation can replace the previous one. This approach is called *elitist replacement* (see Fig. 5) and allows the population to keep the best solutions of the current generation. As a counterpart, more calculations have to be made and that cost might decelerate the execution.

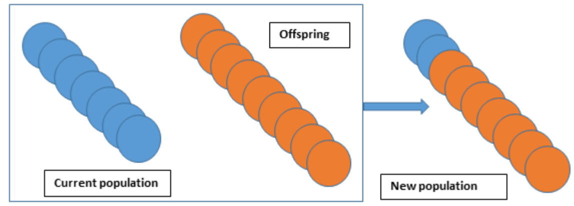

Fig. 5. Elitist Replacement

The main goal of our approach is to find the *best* proposal to modify the preferences provided by the experts with the aim of improving the consensus degree. Therefore, we need to establish criteria that allow us to determine the *quality* of the proposed changes. This level of quality is captured by a *fitness function*. This function evaluates how close an individual is to the optimum solution of the considered problem by means of a score. This score depends on how well that individual solves the problem. In our case, the fitness function is defined with respect to two parameters. On the one hand, it takes into account the distance between the preferences provided by the experts and the potential solution. The shorter the distance is, the better the solution is. In this case, the chance of experts accepting the changes will be higher because the function penalizes the solutions with the lower consensus degree. On the other hand, we are looking for an improvement of the consensus degree. Therefore, the fitness function should award those solutions that have associated a consensus degree higher than the one obtained from the original preferences of the experts. These factors lead to achieve the following two objectives: maximum consensus degree and minimum changes. The lower the fitness score is, the better the individual is. We define our fitness fitness function below.

Definition 4: Let $E = \{e_1, \ldots, e_m\}$, with $m \geq 2$, be a set of experts, $X = \{x_1, \ldots, x_n\}$, with $n \geq 2$, be a set of alternatives and $p = (p^1, \ldots, p^s)$ be the corresponding fuzzy preference relations. For all $q = (q^1, \ldots, q^s)$ fuzzy preference relations associated to an individual, the fitness function $\mathcal{F} : (\mathbb{R}^n)^s \to \mathbb{R}$ is defined as:

$$\mathcal{F}(q) = \frac{\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}(p_{ij} - q_{ij})^2}}{consRel(q)}$$

where $consRel(q)$ corresponds to the consensus on the relation q as defined in section II. □

V. EXPERIMENTS

In this section we report on the experiments that we have performed to evaluate the efficiency of our proposal and the obtained results. We analyze the increment of the consensus degree of the proposed solutions with respect to the original preference relations.

The workbench that we have used for performing the experiments has an Intel i5-8250U processor of 3.4 GHz and 8GB of RAM. In our experiments we have considered the combinations of different number of experts $(8, 12, 16)$, alternatives $(6, 10)$, linguistic quantifiers (*most of* and *At least half*) and tree different methods for the population phase of the proposed GA (roulette wheel, truncation and universal stochastic). The initial fuzzy preference relations were randomly generated and the measure consensus at the three possible levels (pairs of alternatives, alternatives and relation) were calculated by means of the OWA operator presented in Section II. We finally measured the improvement of the level of consensus obtained as a percentage of the original consensus degree. We have applied the GA 10 times per each experiment, considering as initial state of the preferences of the experts the solution proposed in the previous execution.

Fig. 6 shows the results corresponding to the experiments with 8 experts, 6 and 10 alternatives and the different types of selection methods and qualifiers. We can observe a high performance of the truncation method, achieving almost an increase of 40% on the consensus with respect to the initial one. However, other methods produced no improvement, despite the run of several steps, showing how hard the problem of improving the consensus gets to be. Also, we found that a big number of alternatives make lower improvements overall, as it is harder to agree on more possibilities. Fig. 7 presents similar results for 12 experts. Finally, Fig. 8 reflects also the trend of a higher performance of the truncation method over the other ones when considering 16 experts. Moreover, the difference between this method and the other ones increases, suggesting that there is a high (co-)relation between the selection method applied and the results obtained. In general, the results obtained in the experiments are similar independently of the applied quantifier. In that sense, we cannot assure that the quantifier affects the improvement obtained.

To sum up, the GA with the truncation selection method was able to highly increase the consensus of the experts while performing small changes. There is not a decisive difference in the results for the different qualifiers used in the OWA operator. Therefore, we cannot consider one to be better than the other. Last, we should expect a smaller improvement when more alternatives are available.

VI. Conclusions and Future Work

We have proposed a genetic algorithm to find the best proposal to modify the preferences provided by the experts in a group decision making for a problem with the aim of improving the consensus degree at each step of the consensus process. This technique evolves into an optimal solution, according to the conditions reflected in the fitness function. The algorithm that we have designed takes into account both the distance between the preferences provided by the experts and the potential solution, and the increment in the consensus level. The results obtained in the performed experiments are very promising, showing an increase around 40%, and encourage us to continue working in this line of research.

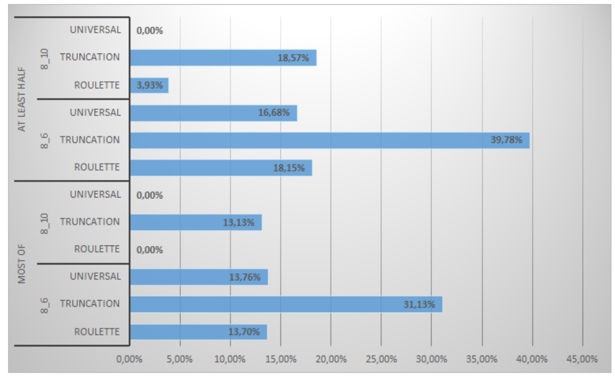

Fig. 6. Results of the experiments with 8 experts

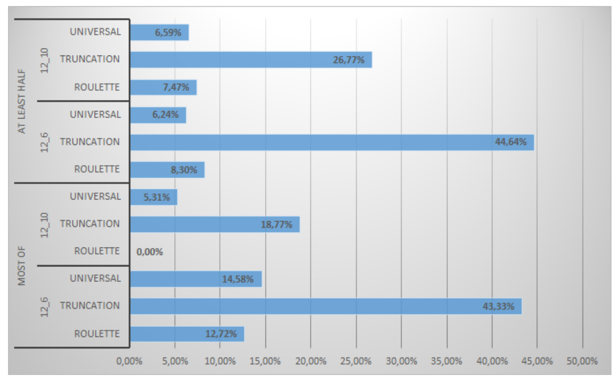

Fig. 7. Results of the experiments with 12 experts

This is only the very first step in our work concerning the application of GAs to GDM. We are already working on the comparison between our approach and state-of-the-art proposals [10], [41]. The preliminary results are promising but we need to perform more experiments to achieve statistical significance. In order to improve the performance of our framework, we would like to consider different distance measure functions and different methods for the selection phase of the population. Another line of future work will consider the application of our proposal to incomplete group decision making problems and different formats of representations of the preferences. An orthogonal line of work consists in applying our framework to our main area of research: testing. Specifically, we would like to to choose among different test suites the one with a higher consensus among the testers. We will consider different types of systems to evaluate the versatility of our framework: asynchronous and distributed [20]–[22], [26], [28], [29], CEP [2], [12], [13], IoT [8], [38] and cloud [1], [7], [36] systems. Finally, we would like to improve the usability of our prototype by integrating it into our framework to represent and analyze fuzzy systems [4].

References

[1] A. Bernal, M. E. Cambronero, V. Valero, A. Núñez, and P. C. Cañizares. A framework for modeling cloud infrastructures and user interactions. *IEEE Access*, 7:43269–43285, 2019.

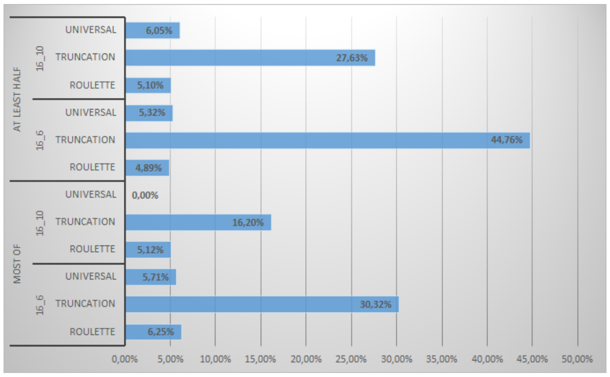

Fig. 8. Results of the experiments with 16 experts

[2] J. Boubeta-Puig, G. Díaz, H. Macià, V. Valero, and G. Ortiz. MEdit4CEP-CPN: An approach for complex event processing modeling by prioritized colored Petri nets. *Information Systems*, 81:267–289, 2019.

[3] N. Bryson. Group decision-making and the analytic hierarchy process: Exploring the consensus-relevant information content. *Computers & Operations Research*, 23(1):27–35, 1996.

[4] I. Calvo, M. G. Merayo, and M. Núñez. A methodology to analyze heart data using fuzzy automata. *Journal of Intelligent & Fuzzy Systems*, 37(6):7389–7399, 2019.

[5] A. Camacho, M. G. Merayo, and M. Núñez. Collective intelligence and databases in eHealth: A survey. *Journal of Intelligent & Fuzzy Systems*, 32(2):1485–1496, 2017.

[6] C. Carlsson, D. Ehrenberg, P. Eklund, M. Fedrizzi, P. Gustafsson, P. Lindholm, G. Merkuryeva, T. Riissanen, and A. G. S. Ventre. Consensus in distributed soft environments. *European Journal of Operational Research*, 61(1):165–185, 1992.

[7] P. C. Cañizares, A. Núñez, J. de Lara, and L. Llana. MT-EA4Cloud: A methodology for testing and optimising energy-aware cloud systems. *Journal of Systems and Software*, 163:110522:1–110522:25, 2020.

[8] D. Corral-Plaza, I. Medina-Bulo, G. Ortiz, and J. Boubeta-Puig. A stream processing architecture for heterogeneous data sources in the internet of things. *Computer Standards & Interfaces*, 70:103426:1–103426:13, 2020.

[9] D. T. Dang, N. T. Nguyen, and D. Hwang. Multi-step consensus: An effective approach for determining consensus in large collectives. *Cybernetics and Systems*, 50(2):208–229, 2019.

[10] D. T. Dang, N. T. Nguyen, and D. Hwang. A new heuristic algorithm for 2-optimality consensus determination. In *2019 IEEE Int. Conf. on Systems, Man and Cybernetics, SMC'19*, pages 70–75. IEEE Press, 2019.

[11] M. H. Degroot. Reaching a consensus. *Journal of the American Statistical Association*, 69(345):118–121, 1974.

[12] G. Díaz, H. Macià, V. Valero, J. Boubeta-Puig, and F. Cuartero. An intelligent transportation system to control air pollution and road traffic in cities integrating CEP and colored petri nets. *Neural Computing and Applications*, 32(2):405–426, 2020.

[13] G. Díaz, H. Macià, V. Valero, J. Boubeta-Puig, and G. Ortiz. Facilitating the quantitative analysis of complex events through a computational intelligence model-driven tool. *Scientific Programming*, 2019:2604148:1–2604148:17, 2019.

[14] M. Fedrizzi, J. Kacprzyk, and H. Nurmi. Consensus degrees under fuzzy majorities and fuzzy preferences using OWA (ordered weighted average) operators. *Control and Cybernetics*, 22(2):71–80, 1993.

[15] D. E. Goldberg. *Genetic Algorithms in Search, Optimisation and Machine Learning*. Addison-Wesley, 1989.

[16] F. Harary. On the measurement of structural balance. *Behavioral Science*, 4(4):316–323, 1959.

[17] F. Herrera, E. Herrera-Viedma, and J. L. verdegay. A model of consensus in group decision making under linguistic assessments. *Fuzzy Sets and Systems*, 78(1):73–87, February 1996.

[18] E. Herrera-Viedma, S. Alonso, F. Chiclana, and F. Herrera. A consensus model for group decision making with incomplete fuzzy preference relations. *IEEE Transactions on Fuzzy Systems*, 15:863–877, 2007.

[19] E. Herrera-Viedma, F. J. Cabrerizo, J. Kacprzyk, and W. Pedrycz. A review of soft consensus models in a fuzzy environment. *Information Fusion*, 17:4–13, 2014.

[20] R. M. Hierons, M. G. Merayo, and M. Núñez. An extended framework for passive asynchronous testing. *Journal of Logical and Algebraic Methods in Programming*, 86(1):408–424, 2017.

[21] R. M. Hierons, M. G. Merayo, and M. Núñez. Bounded reordering in the distributed test architecture. *IEEE Transactions on Reliability*, 67(2):522–537, 2018.

[22] R. M. Hierons and M. Núñez. Implementation relations and probabilistic schedulers in the distributed test architecture. *Journal of Systems and Software*, 132:319–335, 2017.

[23] J. Kacprzyk. Group decision making with a fuzzy linguistic majority. *Fuzzy Sets and Systems*, 18(2):105–118, 1986.

[24] J. Kacprzyk and M. Fedrizzi. 'soft' consensus measures for monitoring real consensus reaching processes under fuzzy preferences. *Control and Cybernetics*, 15(3-4):309–323, 1986.

[25] J. Kacprzyk and M. Fedrizzi. A 'human-consistent' degree of consensus based on fuzzy login with linguistic quantifiers. *Mathematical Social Sciences*, 18(3):275–290, 1989.

[26] R. Lefticaru, R. M. Hierons, and M. Núñez. Implementation relations and testing for cyclic systems with refusals and discrete time. *Journal of Systems and Software*, 170:110738:1–110738:20, 2020.

[27] M. Maleszka. A vector-fuzzy model of a decentralized collective. *Journal of Intelligent & Fuzzy Systems*, 37(6):7313–7323, 2019.

[28] M. G. Merayo, R. M. Hierons, and M. Núñez. Passive testing with asynchronous communications and timestamps. *Distributed Computing*, 31(5):327–342, 2018.

[29] M. G. Merayo, R. M. Hierons, and M. Núñez. A tool supported methodology to passively test asynchronous systems with multiple users. *Information & Software Technology*, 104:162–178, 2018.

[30] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer, 3rd, revised and extended edition, 1996.

[31] N. T. Nguyen, D. Hwang, and E. Szczerbicki. Computational collective intelligence for enterprise information systems. *Enterprise IS*, 13(7-8):933–934, 2019.

[32] N. T. Nguyen, M. Núñez, and B. Trawiński. Collective intelligent information and database systems. *Journal of Intelligent & Fuzzy Systems*, 32(2):1157–1160, 2017.

[33] N. T. Nguyen, E. Szczerbicki, B. Trawinski, and V. D. Nguyen. Collective intelligence in information systems. *Journal of Intelligent and Fuzzy Systems*, 37(6):7113–7115, 2019.

[34] V. D. Nguyen and N. T. Nguyen. Intelligent collectives: Theory, applications, and research challenges. *Cybernetics and Systems*, 49(5-6):261–279, 2018.

[35] V. D. Nguyen, H. B. Truong, M. G. Merayo, and N. T. Nguyen. An overview on consensus-based approaches to processing collective inconsistency and knowledge integration. *WIREs Data Mining and Knowledge Discovery*, 9(4):1311:1–1311:9, 2019.

[36] A. Núñez, P. C. Cañizares, M. Núñez, and R. M. Hierons. TEA-Cloud: A formal framework for testing cloud computing systems. *IEEE Transactions on Reliability (in press)*, 2020.

[37] S. A. Orlovsky. Decision-making with a fuzzy preference relation. *Fuzzy Sets and Systems*, 1(3):155–167, 1978.

[38] J. Roldán, J. Boubeta-Puig, J. L. Martínez, and G. Ortiz. Integrating complex event processing and machine learning: An intelligent architecture for detecting iot security attacks. *Expert Systems with Applications*, 149:113251:1–113251:22, 2020.

[39] M. Roubens. Fuzzy sets and decision analysis. *Fuzzy Sets and Systems*, 90(2):199–206, 1997.

[40] S. Saint and J. R. Lawson. *Rules for Reaching Consensus: A Modern Approach to Decision Making*. Wiley, 1994.

[41] Z. Wu and J. Xu. A consensus model for large-scale group decision making with hesitant fuzzy information and changeable clusters. *Information Fusion*, 41:217–231, 2018.

[42] R. R. Yager. On ordered weighted averaging aggregation operators in multicriteria decisionmaking. *IEEE Transactions on Systems, Man, and Cybernetics*, 18(1):183–190, 1988.

[43] L. A. Zadeh. A computational approach to fuzzy quantifiers in natural languages. In *Computational Linguistics*, Int. Series in Modern Applied Mathematics and Computer Science, pages 149–184. Pergamon, 1983.