

# A multicriteria integral framework for agent-based model calibration using evolutionary multiobjective optimization and network-based visualization

Ignacio Moya<sup>a,\*</sup>, Manuel Chica<sup>a,b</sup>, Óscar Cordon<sup>a</sup>

<sup>a</sup> Andalusian Research Institute DaSCI "Data Science and Computational Intelligence", University of Granada, Granada 18071, Spain

<sup>b</sup> School of Electrical Engineering and Computing, The University of Newcastle, Callaghan, NSW 2308, Australia

## ARTICLE INFO

### Keywords:

Agent-based modeling  
Model calibration  
Evolutionary multiobjective optimization  
Information visualization

## ABSTRACT

Automated calibration methods are a common approach to agent-based model calibration as they can estimate those parameters which cannot be set because of the lack of information. The modeler requires to validate the model by checking the parameter values before the model can be used and this task is very challenging when the model considers two or more conflicting outputs. We propose a multicriteria integral framework to assist the modeler in the calibration and validation of agent-based models that combines evolutionary multiobjective optimization with network-based visualization, which we believe is the first integral approach to model calibration. On the one hand, evolutionary multiobjective optimization provides several sets of calibration solutions (i.e., parameter values) with different trade-offs for the considered objectives in a single run. On the other hand, network-based visualization is used to better understand the decision space and the set of solutions from the obtained Pareto set approximation. To illustrate our proposal, we face the calibration of three agent-based model examples for marketing which consider two conflicting criteria: the awareness of the brand and its word-of-mouth volume. The final analysis of the calibrated solutions shows how our proposed framework eases the analysis of Pareto sets with high cardinality and helps with the identification of flexible solutions (i.e., those having close values in the design space).

## 1. Introduction

Model simulation is useful for representing and analyzing complex systems, but validating a model is not straightforward, specially if the modeling technique involves the definition and setting of many parameters. This is the case of agent-based modeling (ABM), a model simulation technique that has become highly relevant in the recent years [14, 43]. The ABM methodology [5, 13, 24] relies on a population of autonomous entities called agents which behave according to simple rules and by social interactions with other agents. The aggregation of these simple rules and interactions allows us to represent complex and emerging dynamics as well as to define what-if scenarios and to forecast hypothetical scenarios [19]. However, creating and configuring a model for a specific problem from scratch can be difficult for designers and decision makers. If some of the model parameter values cannot be specified using the available information and knowledge, the modeler needs to manually estimate them for properly simulating the desired dynamics. The process of adjusting the values is known as the calibration of the model and it is a crucial step during the model validation [7, 8, 31].

A common calibration approach is automated calibration, a data-rich and computationally intensive process that compares real-world data to model outputs and tunes a set of model's parameters to match the data [31, 34]. Automated calibration requires a set of historical data, an error measure, and an optimization method for modifying the parameters in a systematic way by minimizing the error measure. However, after the application of the optimization method, the resulting parameter values need to be carefully reviewed and validated, since a good fitting of the historical data does not ensure the validity of the model. Additionally, typical parameters of computational models exhibit non-linear interactions and usually the best approach is to use a non-linear optimization algorithm such as metaheuristics [38] that can search across a large span of the model parameter space [7, 26, 37]. Metaheuristics are a family of approximate non-linear optimization techniques that provide high quality solutions in a reasonable time for solving complex problems in science and engineering [38]. In addition, often modelers design their models considering two or more conflicting criteria or key performance indicators (KPIs). In this scenario, the existence of an optimal solution is replaced by a set of Pareto-optimal solutions with the best trade-off between the different criteria.

\* Corresponding author.

E-mail addresses: [imoya@ugr.es](mailto:imoya@ugr.es) (I. Moya), [manuelchica@ugr.es](mailto:manuelchica@ugr.es) (M. Chica), [ocordon@decsai.ugr.es](mailto:ocordon@decsai.ugr.es) (Ó. Cordon).

<https://doi.org/10.1016/j.dss.2019.113111>

Received 18 December 2018; Received in revised form 11 July 2019; Accepted 11 July 2019

Available online 19 July 2019

0167-9236/ © 2019 Elsevier B.V. All rights reserved.

Therefore, calibrating those models involve a multicriteria decision making process since the modeler needs to select a model configuration between different solutions that satisfy the multiple criteria at different levels.

The validation of these models becomes even more difficult as there are multiple sets of parameter values that could be considered valid. In this paper, we propose to improve the validation of ABM systems and other discrete-event simulation models with multiple conflicting KPIs by introducing a multicriteria integral framework. This framework combines two elements: evolutionary multiobjective optimization (EMO) [10] algorithms for automatically calibrating the model parameters from historical data and an advanced visualization method that enhances the understanding of the calibration process and its results. To the best of our knowledge, ours is the first integral approach dealing with the calibration of simulation models from a multicriteria perspective with the aid of visualization tools. Our integrated approach shows that visualization is a key issue for automated calibration as it increases the understanding of the calibrated model, assisting the designer on the model validation [31, 34]. This way the modeler can adapt the automatic calibration process to consider the mentioned conflicting outputs replacing the underlying optimization algorithm by a multiobjective metaheuristic. Although there are previous efforts using EMO for calibrating ABMs [28, 32], none of them considered an integral framework incorporating novel visualization methods for easing the validation of the calibrated models.

We propose the use of *moGrams* [40] to represent the set of calibration results. It is a methodology that combines the visualization of non-dominated solutions in both the design and the objective spaces. A *moGram* is a weighted network where the nodes represent the solutions of a Pareto set approximation and each edge represents a similarity relationship between two solutions in the design space. By using *moGrams*, the modeler will improve her/his understanding of the calibration problem by being able to identify clusters of solutions, to detect the most flexible ones (i.e., those that can be exchanged with another solution with minimum changes in their decision variables), and to conveniently validate their selection of parameter values based on the relationships between solutions (i.e., model parameter configurations).

Two different experimental setups have been designed to validate our proposal. First, we introduce a controlled experiment where we show how the framework properly identifies the optimal values in a design environment. Second, we present its application to a practical problem calibrating two instances of a real banking marketing scenario which considers real data and different dimensionality (i.e., number of decision variables). In both cases we consider an ABM for marketing modeling two conflicting criteria: brand awareness and word-of-mouth volume. Marketing and word-of-mouth programs are one of the fields with the greatest number of ABM applications [9], although our framework is not restricted to this area nor any specific modeling technique.

NSGA-II [12], the most extended EMO algorithm, is used in our experimentation although the framework allows us to consider any EMO algorithm. Finally, the behavior of the resulting solutions for the practical problem is analyzed by visualizing their parameter settings using *moGrams*, which allows us to uncover additional insights about the problem decision space and help with the validation process.

The structure of this paper is as follows. We introduce our approach for multiobjective model calibration in Section 2. Then, we present the used ABM for marketing in Section 3. We describe our experimentation and its results in Section 4, including the analysis and validation of the calibrated solutions. Finally, in Section 5 we discuss our conclusions and final remarks.

## 2. Multicriteria integral framework for model calibration

In this section we describe our integral multicriteria framework for model calibration using EMO algorithms and network-based visualization. A diagram illustrating the components of our framework is shown in Fig. 1. We start by presenting how we handle the problem objectives and the parameters considered during calibration in Section 2.1. Then, Section 2.2 explains the role of the EMO algorithm, which is a core component of the framework. Section 2.3 elaborates the importance of visualization for model calibration and validation. Finally, Section 2.4 introduces *moGrams*, the selected visualization methodology to analyze and better understand the calibrated solutions, which is the other core component of the proposed framework.

### 2.1. Calibration objectives and parameters

In an automated calibration process, the values of the model parameters are adjusted to match the model outputs with the data reality of the modeled scenario. We define each parameter configuration  $X = (x_1, \dots, x_n)$  as a vector of  $n$  decision variables. The modeler should carefully select the model parameters that will be estimated by automated calibration, since the difficulty of validating the calibrated configurations increases with the number of calibrated parameters. On the one hand, the modeler should consider those parameters being the most uncertain and the hardest to define by her/him according to the available information. On the other hand, sensitive parameters should also be considered for calibration since small changes in their values can significantly affect the model's response and the global output. Our approach considers the calibration of parameters using either integer or real values.

In our multiobjective approach, we assess the quality of a given model configuration regarding two or more conflicting criteria which are considered as calibration objectives. We evaluate the quality of the model regarding the fitting of its output to the provided real, historical data for the multiple defined KPIs. In order to avoid over-fitting in the resulting calibrated parameters, the historical data can be split into training and hold-out. Hold-out sets are encouraged to be considered when having real data for calibration purposes in order to check if the model is general enough and to avoid over-fitting (see guidelines when building agent-based decision support systems in [9]). These hold-out sets are then useful for testing the results of the calibration procedure and improving the model confidence. However, since normally ABM implementations are employed as explanatory models instead of forecasting ones [17], the use of hold-out sets can be unnecessary in some cases. The fitting over the training historical data is computed using a deviation measure  $e$  that guides the optimization algorithm calculating the error between the provided ground-truth data and the model output for a given objective. This distance can be computed using any of the standard deviation measures (RMSE, MAPE, or MARE [18], for instance). The modeler should choose the most appropriate for each objective. The goal of the optimization algorithm is to minimize  $F(X) = (f_1(X), \dots, f_M(X))$ , where  $M$  represents the number of objectives. Each fitness function  $f_j(X)$  computes the error associated to objective  $j$ , and is defined by Eq. (1), where  $\delta^j$  represents the target ground-truth values for the  $j$ -th output and  $o^j(X)$  represents the simulated values of the model using the parameter configuration  $X$ . Note that  $e^j$  is independent for each objective  $j \in [1, M]$  and different deviation measures can be used. Our framework does not impose any kind of restriction on that issue.

$$\min F(X) = \min(f_1(X), \dots, f_j(X), \dots, f_M(X)),$$

$$\text{where } f_j(X) = e^j(o^j(X), \delta^j). \quad (1)$$

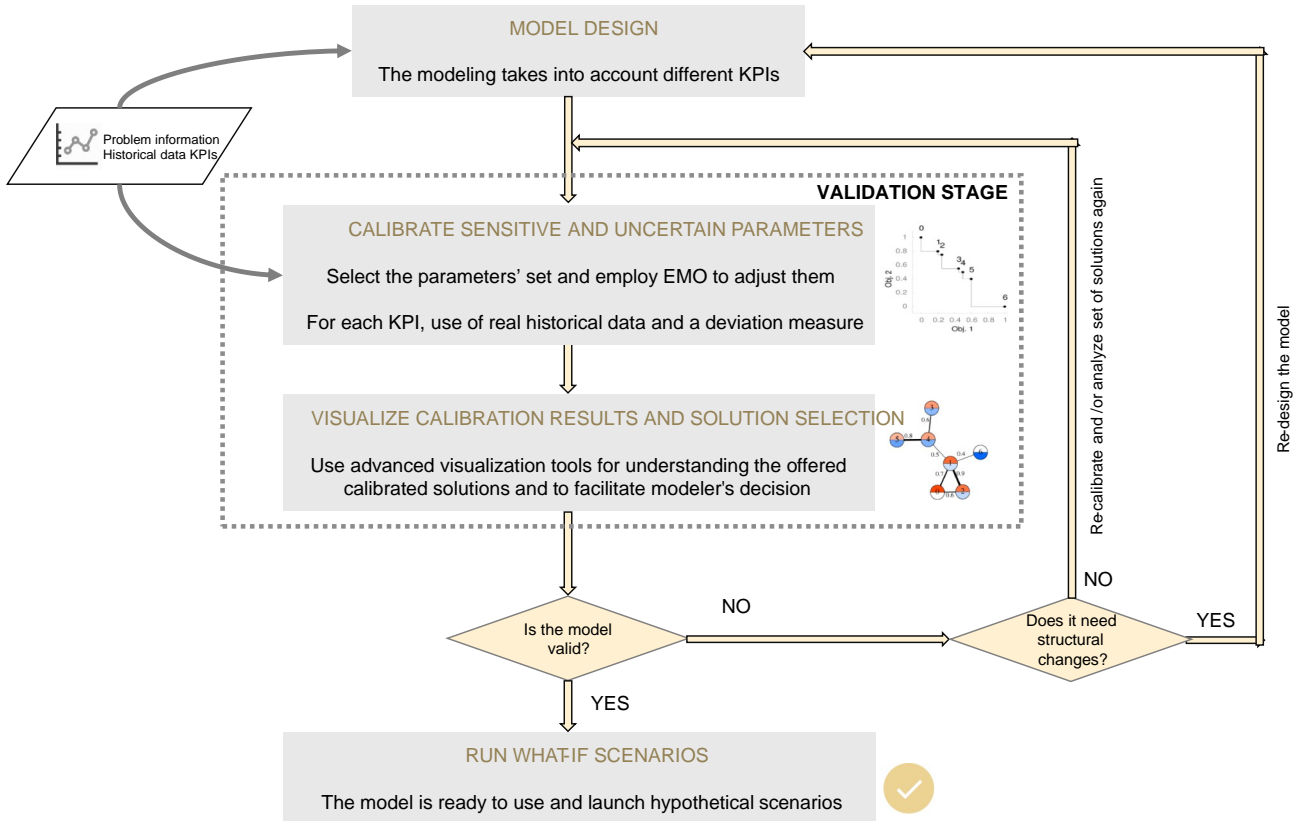


Fig. 1. Diagram illustrating the components and the flow of our multicriteria integral framework for model calibration.

## 2.2. EMO algorithm

The EMO algorithm is a core component of the framework. EMO algorithms are population-based metaheuristics that represent the solutions of the problem as individuals of a population. They can provide different model configurations (i.e., sets of parameter values) in a single run. In our design for the calibration problem, each individual of the population has  $n$  genes that corresponds with the  $n$  decision variables that represent each model configuration, with these genes being either real-coded or integer-coded values. The model configurations obtained by EMO algorithms have different values in the objective space and comprise a Pareto-optimal set approximation.

Any EMO algorithm can be selected for performing this process and should be chosen depending on the characteristics of the model being calibrated. For instance, if the calibration problem considers less than four objectives, any well-known EMO algorithm such as NSGA-II [12], SPEA2 [53], or MOEA/D [21] can be considered. Otherwise, the calibration problem should be treated as a many-objective optimization problem and the selected EMO algorithm should perform properly in this environment. Some examples of EMO algorithms for many-objective optimization would be NSGA-III [11], HypE [2], GrEA [48], or KnEA [51]. In addition, the modeler should consider the use of additional tools and methods for ensuring the good performance of the selected EMO algorithm. For instance, the modeler could use approaches such as irace [23] for automatically adjusting the EMO parameters. However, this decision should be addressed by the modeler since it depends on the specific search space of the defined calibration problem.

## 2.3. Visualization for model calibration

The effective use of an ABM simulation for representing a complex

system heavily relies on the transparency of the underlying model. ABM modelers and stakeholders require to understand how the model re-creates a given behavior, since the simulation of ABMs is frequently used for defining *what-if* scenarios and forecast hypothetical scenarios [19, 42]. This can be achieved from a white-box perspective [33], where both modelers and stakeholders can make use of visualization tools for increasing the *explainability* of the model.

Improving the understanding of artificial intelligence-based models is one of the goals of the emerging area of explainable artificial intelligence [33]. It encourages modelers and researchers to *open* black-box models so their behavior can be easily understood and their output can be better explained. Explainable artificial intelligence also empowers the solutions delivered by white-box models, since boosting the transparency of the delivered solutions should increase the trust in the behavior and performance of these solutions.

This highlights the role of visualization methods for model calibration, since they are powerful tools that increase the understanding of the modeler on the calibrated model and its parameter settings [7]. The use of visualization increases the transparency of the quality indicators (mostly focused on the fitting of the model to real data) for the validation of the calibrated model [4, 7, 20]. Thus, visually showing the underlying relationships between an input configuration and its corresponding model output becomes a critical component of the validation process.

When the model considers two or more conflicting outputs, multi-objective visualization methods are specifically required for the validation of the model. Most of the available contributions in the literature focus on the visualization of the non-dominated solutions in the objective space [41, 44]. In contrast, only a few contributions tackle the visualization of the solutions in the design space, which is the most interesting for discovering knowledge about the parameter values, and

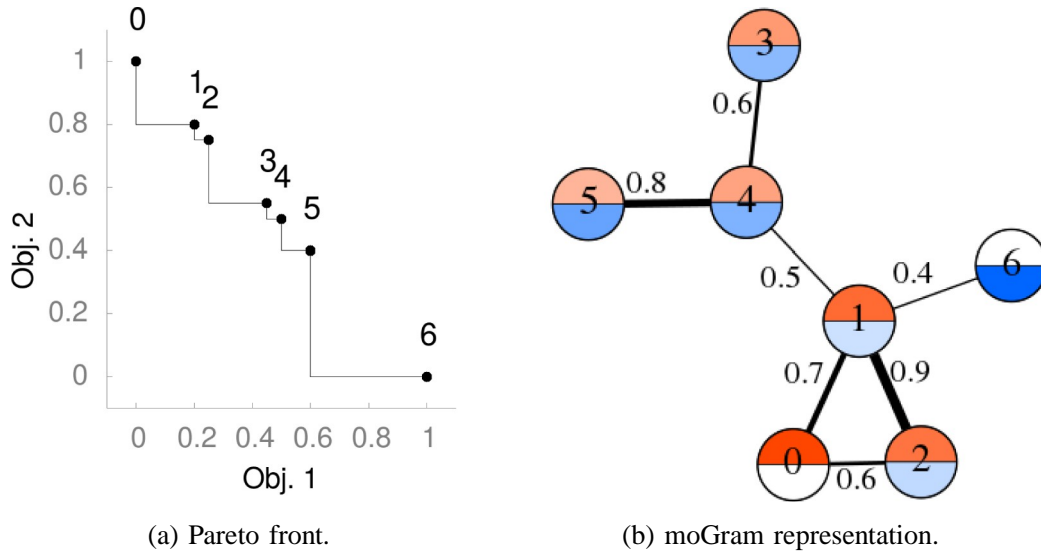


Fig. 2. Generated moGram network example for a given Pareto set approximation corresponding to a problem with two objectives.

even less proposals derive joint visualizations for both the objective and design spaces [40]. One of the few existing approaches of such kind is the *moGrams* methodology [40], which mutually analyzes and visualizes the solutions obtained by EMO algorithms in the decision and objective spaces.

#### 2.4. moGrams

moGrams [40] represents the non-dominated solutions in a Pareto set approximation as nodes in a weighted network where each edge stands for a relationship between the connected solutions in the design space. The weights of the edges are computed using a similarity metric specifically defined for each problem by the designer. In order to improve the readability of the network, moGrams employs the Pathfinder network pruning algorithm [36] for reducing the edges of the network leaving only the most relevant ones from a global viewpoint. In addition, each node is divided into sectors of the same size, each of them associated to a different objective, which are colored differently with its opacity proportional to the quality of the solution for the respective objective. For example, if a problem considers four objectives, the node is divided into four sectors with different colors. This way, the modeler has access to the whole information of both the objective and the design spaces at the same time. For our problem (i.e., the validation of a model) it provides additional information regarding the parameter settings of the different calibrated model configurations, highlighting similarities between them.

Fig. 2 shows an example of the moGram generated for the Pareto set approximation of a problem with two objectives. The Pareto front approximation obtained for the generated moGrams network is also shown, since the joint visualization of both elements is suggested for better understanding the relationships in the design space. In this network example, we can identify two separate clusters connected by the edge between Solution 4 and Solution 1, with the latter being the most connected solution. From the neighbors of Solution 1, we can observe that Solution 2 has the highest similarity relationship. This means that the parameter configuration of Solution 2 is highly similar to the one of Solution 1.

However, the high similarity between Solutions 1 and 2 could lead us to think that both solutions are close in the objective space. In this regard, moGrams provides the user with other relationships that are not intuitive, such as the relationship between Solution 1 and Solution 6.

This relationship reveals that the closer configuration to Solution 6 is the one defined by Solution 1, which is located at the other end of the Pareto front. Thus, the decision maker can detect parameters that drastically change the behavior of the solutions using this relationship. In addition, the topology of the moGrams network allows the modeler to identify Solution 1 as the most flexible solution, which can be swapped with other solutions with minimum changes on its decision variables.

### 3. Description of the agent-based model for marketing

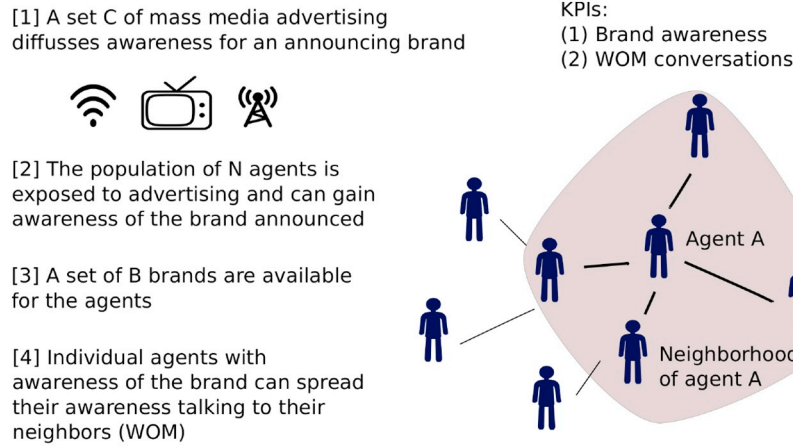
This section describes the main features of the considered ABM for marketing scenarios. First, the general structure of the model and behavior of the agents are presented in Section 3.1. Then, Section 3.2 introduces the artificial social network and its features. Section 3.3 presents how the mass media channels are modeled. Finally, in Section 3.4 the parameters of the model selected for calibration are summarized and the conflict of adjusting both KPIs of the model with their corresponding fitting functions is described in Section 3.5.

#### 3.1. ABM general structure and agent's state and update rule

Our proposed model considers a terminating simulation with  $T$  weeks of a market that comprises a set of brands  $B$ . Using a time-step of a week, the model simulates the behavior of  $N$  agents and their reaction to social influence through a social network in a word-of-mouth (WOM) process; and external influences (advertisement) through a set of  $C$  mass media channels. The model has two main outputs or KPIs: brand awareness and WOM volume (i.e., the number of WOM interactions among the consumers). We select these KPIs because they have an important role in market expansion [22, 25]. A general scheme of our model is presented in Fig. 3.

The awareness values of the agents are modeled using a state variable called  $a_i^b \in \{0, 1\}$ . If  $a_i^b(t) = 1$ , then the agent  $i$  is aware of brand  $b$  at time step  $t$ . Otherwise, the agent does not have awareness of brand  $b$ . This state variable is initialized using an initial awareness parameter set for each brand ( $a^b(0) \in [0, 1]$ ) which is the global awareness of the population and fulfills  $a^b(0) = \frac{1}{N} \sum_{i=1}^N a_i^b(0)$ . Therefore, the initial awareness parameter for each brand specifies the percentage of agents that have awareness of that brand at the beginning of the simulation. This process is carried out during the initialization of





**Fig. 3.** General scheme and structure of the ABM with an example of a brand advertised using mass media. The agents exposed to advertising can gain awareness of the brand announced and talk about it to their neighbors.

the agents, where they activate their awareness of each brand  $b$  with probability  $a^b(0)$ .

The awareness values of the agents do not remain static but change during the simulation: agents may lose or gain awareness of any brand at each step of the simulation. On the one hand, agents may gain awareness of a brand due to advertising or due to interacting with other agents through a WOM diffusion process. On the other hand, if the awareness of a brand is not reinforced, it may be lost over time because of a deactivation process [46, 47].

We model these losing/gaining effects with additional parameters. The parameter regulating the rate at which awareness is lost over time is called awareness deactivation probability ( $d \in [0,1]$ ). This parameter is modeled as follows. At the beginning of each step  $t$ , the agent  $i$  checks all the brands the agent is aware of ( $a_i^b(t) = 1, \forall b \in B$ ). Each of these awareness values will be deactivated with a probability  $d$  by setting  $a_i^b(t) = 0$ . If the deactivation takes effect, the agent could still re-gain awareness due to the WOM diffusion and/or the mass media channels during the same simulation step, but it will not check for deactivation until the next simulation step. The modeling processes of the awareness obtained by WOM diffusion and mass media channels are explained in detail in Sections 3.2 and 3.3, respectively.

In addition, each agent stores the number of conversations produced during its diffusion process (depicted in Section 3.2) in order to compute the WOM volume for each brand ( $\omega_i^b(t)$ ). This way, every time an agent starts a diffusion process and talks with its neighborhood, the variable  $\omega_i^b(t)$  will be updated by incrementing it with the number of agents' neighbors (i.e., conversations). Finally, it will update the global  $\omega^b(t)$  variable for the respective brand and time step.

### 3.2. Social network of agents and their word-of-mouth interactions

Our agents populate an artificial social network [3, 45]. We model this social network using an artificial scale-free network [3] because the most real networks match with this network model [3, 30]. In these kinds of networks, the degree distribution follows a power law [3]. This means that few nodes have a significantly large number of connections (hubs of the social network) and most nodes have a very low number of connections. We generate our scale-free network using the Barabasi-Albert preferential attachment algorithm [3]. This algorithm has a main parameter  $m$  which regulates the network growth rate and its final density. The generation process starts with a small clique (a completely connected network) with  $m_0$  nodes. At each generation step, a new node is added and connected to  $m$  different existing nodes. When a new node is included, the probability of choosing an existing node is proportional

to its degree (preferential attachment). After  $t$  iterations, the Barabasi-Albert algorithm results in a social network with  $m \cdot t$  edges. Finally, the average degree of the social network is  $\langle k \rangle = 2 \cdot m$ .

The agents of the model can spread their awareness values during the simulation through the artificial social network. We model this social interaction as a contagion process which allows information diffusion through the nodes of the social network depending on their connectivity [30, 35, 49, 50]. Every agent  $i$  has a talking probability ( $p(t)_i^b \in [0, 1]$ ) to spread the brands it is aware of at time step  $t$  (i.e., for every brand  $b$  where  $a_i^b = 1$ ). This probability  $p_i^b$  specifies when the agent  $i$  talks with all of its neighbors in the artificial social network, having the chance of transferring its awareness (i.e., a contagion process). We model this contagion effect using a parameter called WOM awareness impact ( $\alpha^{WOM} \in [0,1]$ ), which represents the probability for an agent in the neighborhood to be aware of a brand after having a conversation about it.

### 3.3. Modeling mass media channels

We model external influences like brand advertising as global mass media [16] using a similar approach to the one applied in the social network. The external influences are parameterized to define the differences between the channels (i.e., press, radio, and television). The set of  $C$  mass media channels can influence any number of agents at random depending on the channel potential for reaching the population and the investment of each brand.

The maximum population percentage that can be reached by a mass media channel is bounded by the nature of the channel itself. In this sense, some media are able to reach more people than others. For example, the maximum population percentage that can be reached by a campaign scheduled in the radio is bounded by the maximum population percentage that listens to the radio. We model this behavior with a reach parameter ( $r_c \in [0,1], \forall c \in C$ ), which defines the maximum number of people a channel  $c$  is able to hit during a single step.

The advertising campaigns of the mass media channels are modeled using gross rating points (GRPs). In advertising [15], a GRP is a measure of the magnitude of the impressions scheduled for a mass media channel. Specifically, we use the convention that one GRP means reaching 1% of the target population. The investment units in GRPs for channel  $c$  by brand  $b$  and time step  $t$  is modeled by the variable  $\chi_c^b(t)$ . Each channel has different costs for GRP and the brands need to carefully choose their investment since increasing the population awareness using mass media channels implies a monetary cost. Using both the supplied GRPs for a given brand and the reach values for a mass media channel, we are able to model brand advertising. Algorithm 1 shows the

**Table 1**  
List of parameters of our proposed marketing model.

| Name           | Description  |
|----------------|--|
| $N$            | Number of agents running in the model                      |
| $ B $          | Number of brands contained in the model                    |
| $ C $          | Number of mass media channels in the model                 |
| $T$            | Number of steps of the model                               |
| $a^b(0)$       | Initial awareness for brand $b$                            |
| $d$            | Awareness deactivation probability in the model            |
| $m$            | Parameter for social network generator                     |
| $p_i^b(0)$     | Initial talking probability, same value for each brand $b$ |
| $\alpha^{WOM}$ | Awareness impact for social interactions                   |
| $\chi_c^b$     | GRP units invested by brand $b$ in channel $c$             |
| $r_c$          | Reach for mass media channel $c$                           |
| $\alpha_c$     | Awareness impact for mass media channel $c$                |
| $\tau_c$       | Buzz increment for mass media channel $c$                  |
| $d\tau_c$      | Buzz decay for mass media channel $c$                      |

scheduling algorithm for modeling impacts of the media channels over the population.

```

1 begin
2   reach_step = 0;
3   total_hits =  $\chi_c^b(t) \cdot 0.01 \cdot N$ ;
4   reach_increment = 1 /  $N$ ;
5   i = 0;
6   while  $i < total\_hits$  do
7     select agent randomly;
8     if selected agent was already hit then
9       impact agent;
10      i++;
11    else if  $reach\_step + reach\_increment \leq r_c$  then
12      impact agent;
13      i++;
14      reach_step += reach_increment;
```

**Algorithm 1.** Pseudo-code of the advertising scheduling of the model for a given brand, time step and channel.

Each mass media channel has an awareness impact parameter ( $\alpha_c \in [0,1], \forall c \in C$ ) that defines the probability of the agent becoming aware of the brand after one impact. If the agent is not aware of the brand at a given time step  $t$  ( $a_i^b(t) = 0$ ), this probability  $\alpha_c$  will activate the awareness of the agent for brand  $b$ .

Moreover, the advertising transmitted by mass media channels can produce a viral buzz effect in the reached agent, as done in [27]. This buzz effect increases the number of conversations about the announced brand, modifying the talking probability ( $p_i^b$ ) of the reached agents. We model this effect through a variable called buzz increment ( $\tau_c$ ) for each channel  $c \in C$ . This increment of the agents talking probability is computed as a percentage increment over the initial talking probability ( $p_i^b(0)$ ) of the agent. However, if the generated buzz is not reinforced, its effect could decay over time as previous interactions are forgotten. We model this effect with a variable called buzz decay ( $d\tau_c$ ). The action of buzz decay reduces the previous increment of talking probability ( $\sigma_c$ ) applied to the agent through channel  $c$ . The update process for the talking probability value of agent  $i$  for brand  $b$  due to both buzz

increment and decay effects of channel  $c$  is shown in Eq. (2).

$$p_i^b(t+1) = p_i^b(t) - \sigma_i^b(t) \cdot d\tau_c + p_i^b(0) \cdot \tau_c, \text{ where } \sigma_i^b(t) = \sum_{i=1}^t (p_i^b(t) - p_i^b(0) \cdot \tau_c). \quad (2)$$

### 3.4. Parameters selected for calibration

A summary of the complete set of model parameters is listed in Table 1. From those, we select for calibration the parameters that either modify the agent awareness values or their number of conversations for the automated calibration process since they are the most uncertain and the hardest to estimate. These parameters regulate the awareness and talking probability gained by mass media and social interactions with the addition of the awareness deactivation probability ( $d$ ) and the social network generation parameter ( $m$ ). The range of possible parameter values during the calibration process is limited to  $[0,1]$  for the real-coded parameters and to  $\{2, \dots, 8\}$  for the social network generation parameter ( $m$ ), which is the only integer-coded parameter of the model. Notice that, the density of the agents social network is a parameter that is always

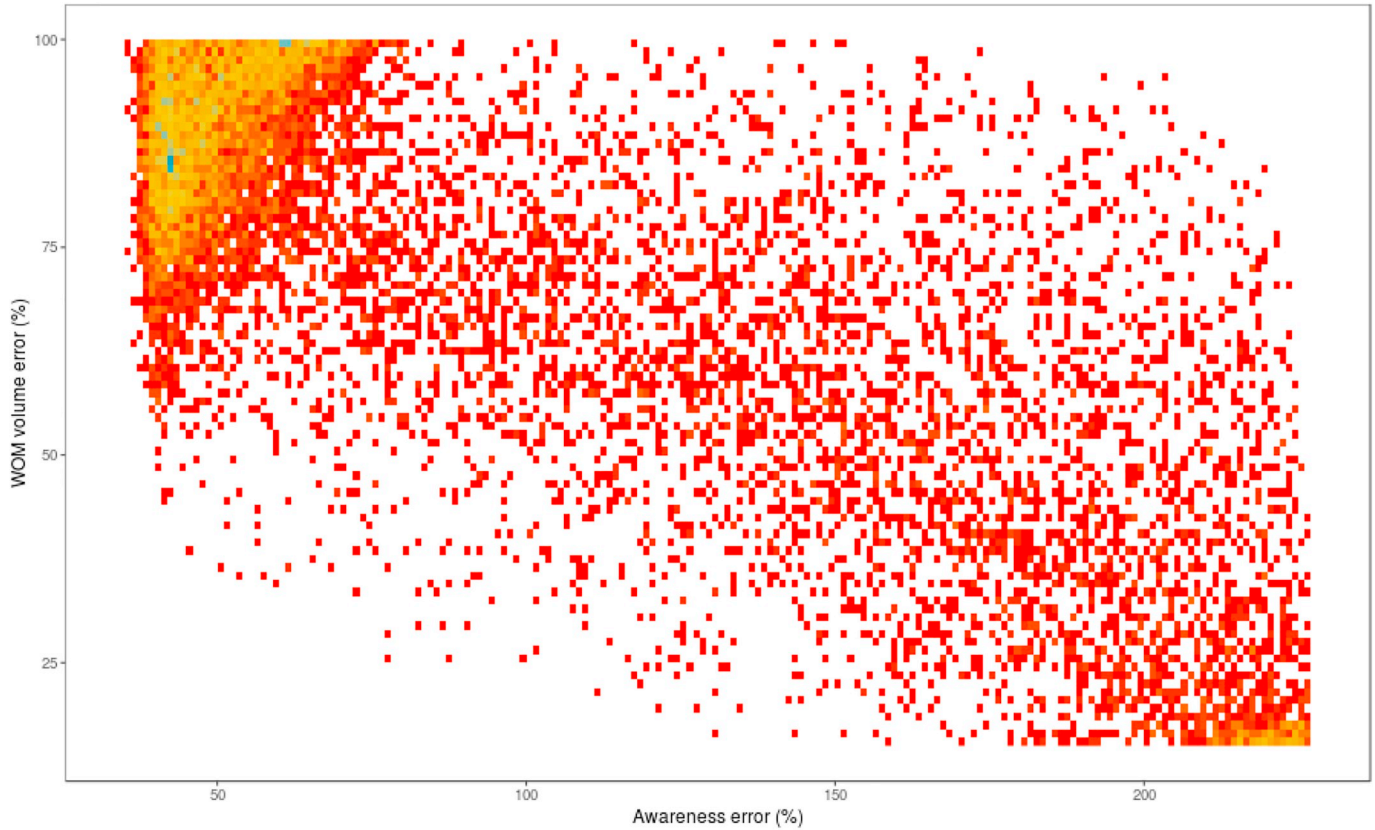
difficult to identify. This way, each of the selected model calibration parameters corresponds with one decision variable in the coding scheme of the EMO algorithm. The final set of parameters to be calibrated for each model instance is determined by the size of the modeled scenario: three parameters for each mass media channel plus four fixed social parameters. Briefly, those parameters are the following:

- Social network parameters. We calibrate the initial talking probability ( $p_i^b(0)$ ), social awareness impact ( $\alpha^{WOM}$ ), awareness deactivation probability ( $d$ ), and social network generation parameter ( $m$ ).
- Mass media parameters. For each defined mass media channel  $c \in C$ , we calibrate its awareness impact ( $\alpha_c$ ), buzz increment ( $\tau_c$ ), and buzz decay ( $d\tau_c$ ).

Therefore, the number of calibrated parameters is  $3 \cdot |C| + 4$ . Fig. 4 shows the coding scheme for a model instance using three mass media channels, that is, composed of 13 genes.

|       |               |             |              |               |             |              |               |             |              |            |                |      |
|-------|---------------|-------------|--------------|---------------|-------------|--------------|---------------|-------------|--------------|------------|----------------|------|
| $m$   | $\alpha_{c1}$ | $\tau_{c1}$ | $d\tau_{c1}$ | $\alpha_{c2}$ | $\tau_{c2}$ | $d\tau_{c2}$ | $\alpha_{c3}$ | $\tau_{c3}$ | $d\tau_{c3}$ | $p_i^b(0)$ | $\alpha^{WOM}$ | $d$  |
| 3     | 0.01          | 0.1         | 0.1          | 0.1           | 0.5         | 0.01         | 0.02          | 0.05        | 0.2          | 0.2        | 0.1            | 0.01 |
| $c_1$ |               |             | $c_2$        |               |             | $c_3$        |               |             |              |            |                |      |

**Fig. 4.** Coding scheme for a model instance with three mass media channels. The first gene ( $m$  parameter) is an integer value bounded to  $\{2, \dots, 8\}$ . The rest of genes in the chromosome represent the real-coded parameters and are defined in  $[0,1]$ .



**Fig. 5.** Sampling of the decision space for the two objectives. The values are shown as a scatter plot, where locations gathering multiple individuals have more intense colors. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

### 3.5. KPI fitting functions

Eqs. (3) and (4) define the objective fitting functions for the two KPIs,  $f_1$  (awareness deviation error) and  $f_2$  (WOM volume deviation error), respectively. These functions compute the deviation error between the provided series of target data and the model outputs for each objective using the standard mean absolute percentage error (MAPE) function, where  $\tilde{a}$  and  $\tilde{\omega}$  represent the ground-truth target values of awareness and WOM volume of the whole population, respectively. The simulated values are generated using Monte-Carlo simulations by computing the average of those independent runs.

$$f_1 = \frac{100}{T \cdot |B|} \sum_{b=1}^{|B|} \sum_{t=1}^T \left| \frac{a^b(t) - \tilde{a}^b(t)}{\tilde{a}^b(t)} \right|, \quad (3)$$

$$f_2 = \frac{100}{T \cdot |B|} \sum_{b=1}^{|B|} \sum_{t=1}^T \left| \frac{\omega^b(t) - \tilde{\omega}^b(t)}{\tilde{\omega}^b(t)} \right|. \quad (4)$$

We can observe that both outputs of the model are correlated. Thus, it is not advisable to calibrate the outputs independently because modifying the parameters of the model implies changes for both outputs. On the one hand, mass media channels activate brand awareness using advertisement but this effect also modifies the WOM volume generated via the buzz effect of the campaign, which can deviate its value beyond the target data. Additionally, as WOM spreads brand awareness via the agent interactions through the social network, adjusting the number of conversations will modify the awareness of the population, deviating its value beyond the target data. Therefore, for most model instances there will be no model configuration able to satisfy the fitting of both KPIs to the historical data at the same time.

Finally, a sampling of the decision space is shown in Fig. 5. These values were obtained generating 10,000 random calibration solutions for a given instance of the model and plotting their fitting for both KPIs.

**Table 2**

Summary of the configuration of the baseline model instance P-25 for the parameters that are manually set. These values are set from the real available data.

| Baseline instance P-25 |       |             |       |             |        |             |       |             |       |
|------------------------|-------|-------------|-------|-------------|--------|-------------|-------|-------------|-------|
| Name                   | Value | Name        | Value | Name        | Value  | Name        | Value | Name        | Value |
| $N$                    | 1000  | $ B $       | 8     | $ C $       | 7      | $T$         | 52    | $a^{b1}(0)$ | 0.709 |
| $a^{b2}(0)$            | 0.757 | $a^{b3}(0)$ | 0.589 | $a^{b4}(0)$ | 0.2559 | $a^{b5}(0)$ | 0.081 | $a^{b6}(0)$ | 0.429 |
| $a^{b7}(0)$            | 0.395 | $a^{b8}(0)$ | 0.34  | $r_{c1}$    | 0.928  | $r_{c2}$    | 0.579 | $r_{c3}$    | 0.548 |
| $r_{c4}$               | 0.035 | $r_{c5}$    | 0.432 | $r_{c6}$    | 0.382  | $r_{c7}$    | 0.696 |             |       |

In this figure, we can visually confirm the fact that both optimization objectives are in conflict, since the solutions that have the lowest deviations for awareness ( $f_1$ ) have the highest deviations for WOM volume ( $f_2$ ) and *vice versa*. This can be observed in the extreme locations of the decision space, where the greater concentration of sampled values are located. In addition, several sampled solutions scatter diagonally from one concentration of solutions to the other, showing the difficulty of balancing the optimization of these two criteria.

## 4. Experimentation

This section presents the experimentation developed and the analysis of results. Sections 4.1 and 4.2 explain the experimental setup, describing the considered problem instances and the algorithmic configuration. Then, Sections 4.3 and 4.4 present the application of our framework both to a designed environment and to real calibration scenarios. Finally, Section 4.5 illustrates the use of the visualization method on the calibration results and analyzes the composition of the non-dominated solutions from the decision maker's point of view.

#### 4.1. Experimental setup

Our experimentation considers three different model instances. These instances are generated starting from an initial *baseline* instance (called P-25 because of its 25 decision variables), which is based on real data from a banking marketing scenario. Then, we synthetically generate two additional instances: a simplified instance for conducting a controlled experiment and an additional instance with increased dimensionality. The simplified instance considers a reduced set of five model parameters, which are selected because of their sensitivity on the model's behavior. The instance with increased dimensionality includes 10 additional mass media channels (which are generated perturbing the GRP investment of the existing ones) and modifications of the target series of historical values for both objectives: awareness and WOM volume. Thus, P-55 will increase the dimensionality of the baseline real-world instance by adding 30 new decision variables to enable a more complete analysis of the algorithms' performance.

Table 2 shows the parameters' values that are manually set using data for the baseline instance P-25. This instance belongs to a real marketing scenario of banking in Spain. These parameters define the initial conditions of our terminating simulation, such as the delimitation of the market (number of brands  $B$ , number of channels  $C$ , and total number of agents  $N$ ) and the initial awareness conditions. Each of these agents initially activate their awareness for each brand  $b$  with probability  $a^b(0)$  which is an input parameter of the model. These awareness values are taken from a brand tracking study used by one of the competing brands. In addition, all the media channels are initialized using their input parameters (i.e., mass media parameters either manually set or automatically calibrated). For instance, we set the values for the reach parameters from a mass media study in Spain<sup>1</sup>. Notice that our terminating simulation takes  $T = 52$  time steps, matching the weeks of a year of simulation. This period is selected because of the historical data, which considers the behavior of the model KPIs during one year.

The generated P-55 instance shares the previous parameter configuration with the addition of the corresponding reach parameter  $r_c$  for the additional mass media channels. The reach parameter values of these new mass media channels take the value of the original ones employed for its generation. For example, if the additional mass media channel  $c_9$  was generated from the original channel  $c_5$ , then  $r_{c_9} = r_{c_5}$ . The generated simplified instance shares the reach parameter configuration with P-25. Those parameters that were not selected for calibration are set to 0 to avoid adding more complexity in the controlled experiment. Finally, the model's historical target values are generated using the output of two known parameter configurations for each objective, which are identified as the best values for the designed environment.

#### 4.2. NSGA-II as the EMO algorithm

We select NSGA-II as our EMO algorithm during our experiments. NSGA-II [12] has become one of the most well-known EMO algorithms and there are several applications in model calibration [1, 32, 52]. It has been proven computationally fast while maintaining good levels of diversity by using a search strategy based on non-dominated sorting in problems with 2 or 3 objectives. During each generation, NSGA-II creates an offspring population  $Q_t$  from the previous parents population  $P_t$ . These two sets are joined into a temporary population  $R_t$  of size  $2 \cdot |P|$  ranking every solution according to its non-dominance level, that is, how many solutions it is dominated by. The new population  $P_{t+1}$  is created including the solutions with the best rank, which belongs to the best non-dominated front. Then, the solutions from the following ranks are included iteratively until  $|P|$  individuals are selected. This way the algorithm is guided to non-dominated regions and the solutions from

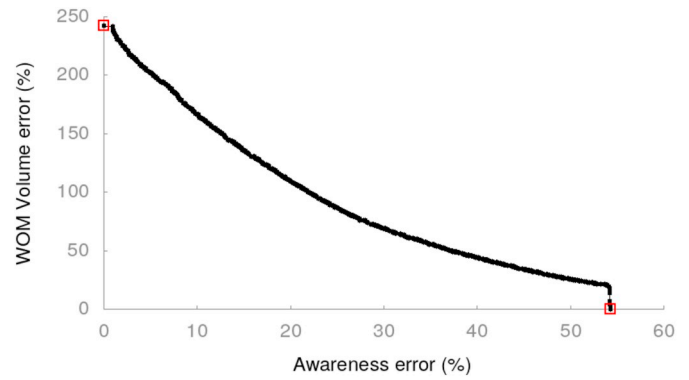


Fig. 6. Attainment surface for the controlled experiment. Best solutions for each objective are located at the left end and the right end of the front respectively. These solutions are highlighted in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

the best non-dominated front are always kept in the population. In order to enhance diversity, the first non-dominated front that cannot be fully included in  $P_{t+1}$  is filtered using a crowding distance calculated for only including the most diverse individuals.

We include simulated binary crossover (SBX) [10] as our crossover strategy, which is applied with probability  $p_c = 1.0$ . SBX performs the crossover operation on real-coded decision variables emulating the behavior of a single-point crossover from binary-encoding. This operator works as follows: given two parents  $P_1 = (p_{11}, \dots, p_{1n})$  and  $P_2 = (p_{21}, \dots, p_{2n})$ , SBX generates two springs  $C_1 = (c_{11}, \dots, c_{1n})$  and  $C_2 = (c_{21}, \dots, c_{2n})$  as  $c_{1i} = \bar{X} - \frac{1}{2} \cdot \bar{\beta} \cdot (p_{2i} - p_{1i})$  and  $c_{2i} = \bar{X} + \frac{1}{2} \cdot \bar{\beta} \cdot (p_{2i} - p_{1i})$ , where  $\bar{X} = \frac{1}{2}(p_{1i} + p_{2i})$  and  $\bar{\beta}$  is the spread factor, a random number fetched from a probability distribution generated using a given distribution index. Regarding the mutation strategy, we choose polynomial mutation [10]. It modifies the values of an individual genes using a polynomial distribution. It is applied with probability  $p_m = 1/n$ , where  $n$  is the number of decision variables (i.e., parameters of the model to be calibrated).

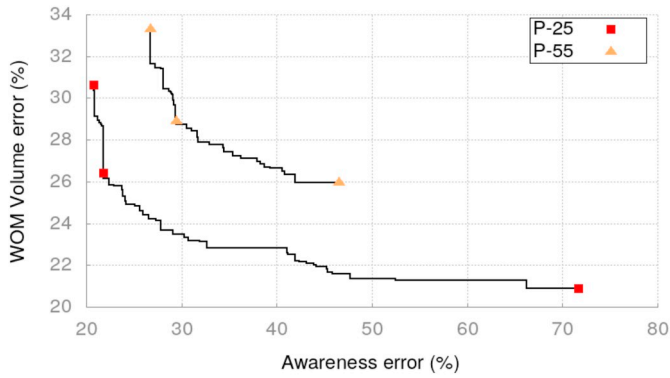
Since EMO algorithms are stochastic algorithms, they must be ran several times for ensuring a good performance for any optimization problem they are applied to. Therefore, we run NSGA-II 20 times using different seeds for each run. The NSGA-II has a population of 100 individuals ( $|P| = 100$ ) and evolves during 100 generations using 10,000 evaluations as stopping criteria. In addition, the first 39 steps of the historical data (i.e., 75% of total) are used for training, leaving the remaining values as hold-out. Each evaluation of an individual is a Monte-Carlo simulation of 15 runs. This value showed an acceptable trade-off between the stability of the model's output and the required runtime. The mutation operator uses a distribution index value of 10 and a different mutation probability value depending on the number of parameters of each model instance. We implemented NSGA-II using the jMetal framework [29].

#### 4.3. Controlled experiment in a designed environment

We define a controlled experiment by employing an instance with a reduced set of sensitive parameters. Therefore, this calibration scenario defines a controlled environment where the optimal parameters values for the defined historical data for each objective are known. The selected parameters are  $\alpha_{c_1}$ ,  $\alpha_{c_2}$ ,  $\alpha_{c_3}$ ,  $\alpha_{c_5}$ , and  $\alpha_{c_7}$ . The latter channels are those with the highest investment in the model and can provide changes into the model's response without being affected by other parameter values (e.g.,  $\alpha_{c_4}$  and  $\alpha_{c_6}$  belong to mass media channels with lower investment and thus are excluded from the experiment). By

<sup>1</sup> <https://es.slideshare.net/ZenithES/libro-de-medios-2015>





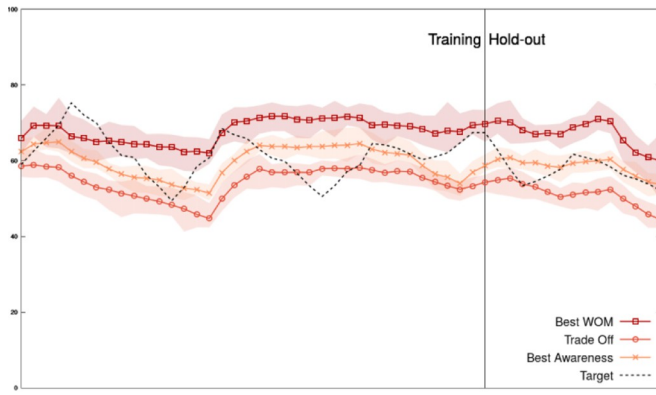
**Fig. 7.** Selected solutions for the two model instances (P-25 and P-55): lowest awareness error, lowest WOM volume error, and best trade-off. Solutions from P-25 are represented using squares and solutions from P-55 are displayed using triangles.

selecting these parameters we see if the calibration algorithm can find solutions to perfectly match each KPI in isolation or a trade-off combination: when the parameters are set to 1, the model perfectly matches the historical awareness values; and when the parameters are set to 0, the model perfectly matches the historical WOM volume values.

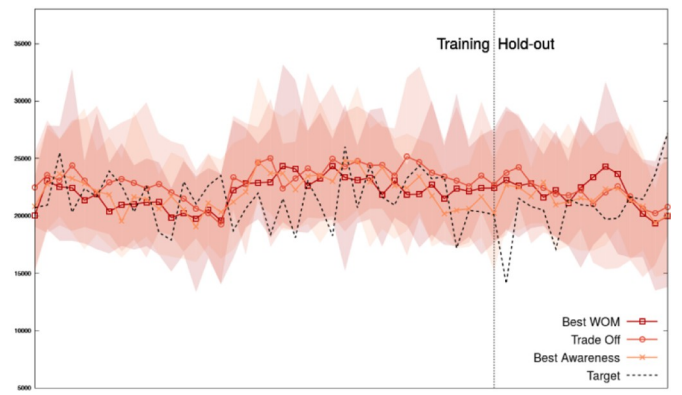
Therefore, this controlled experiment will test if the calibration process is able to search across the whole solution space by setting the values of the parameters to their limits.

Fig. 6 shows the resulting attainment surface [54] of applying our framework to the designed instance. Attainment surfaces are surfaces uniquely determined by a set of non-dominated points that separates the region of the objective space dominated by the set from the region not dominated by it [54]. The surface of Fig. 6 is obtained by merging all the obtained Pareto set approximation from every execution of NSGA-II and by removing the dominated solutions. It shows how the framework identifies the best result for each of the objectives (i.e., the parameter configuration obtaining a perfect match), which are represented by red boxes in Fig. 6. Additionally, the resulting attainment surface provides interesting insights regarding the behavior of the WOM volume dynamics in the model. For example, there are several solutions at the right end of the attainment surface that achieve similar/equal awareness error while decreasing WOM error. This can be explained by the sensitivity of WOM volume dynamics to Monte-Carlo variability.

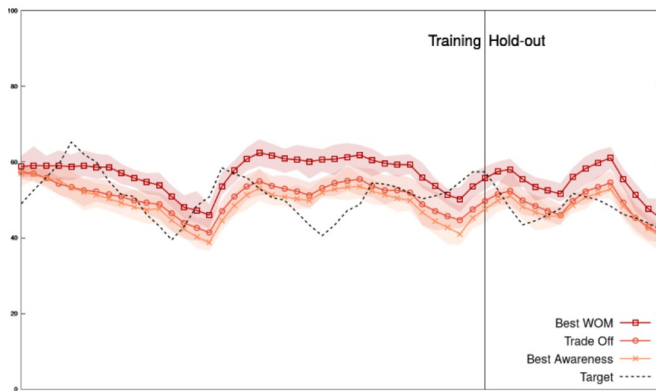
This simple experiment allows us to properly validate the performance of the first stage of our framework. On the one hand, it shows how the EMO algorithm is actually able to obtain the optimal parameter values for each considered KPI in the specific model calibration instance. On the other hand, it also illustrates the capability of the EMO algorithm to generate a Pareto set composed of a large number of parameter configurations showing different trade-offs between the two KPIs being optimized.



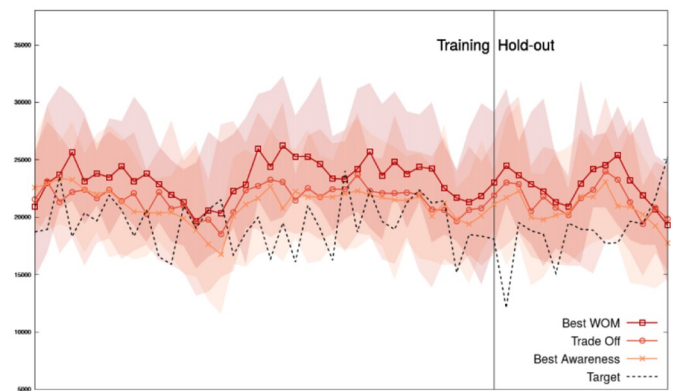
(a) P-25 : awareness fitting



(b) P-25 : WOM volume fitting

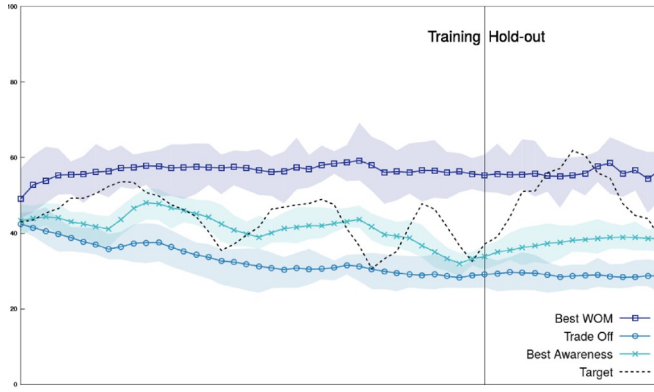


(c) P-55 : awareness fitting

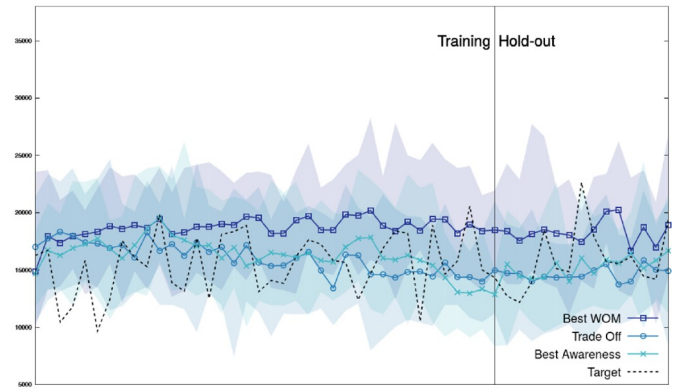


(d) P-55 : WOM volume fitting

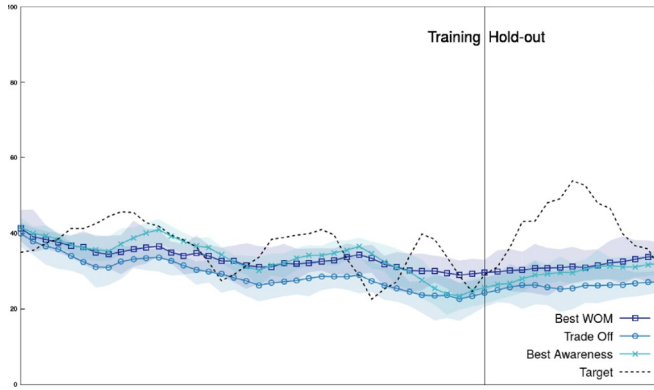
**Fig. 8.** Awareness output and WOM volume over time for P-25 and P-55 regarding brand 3. In these charts, the central line represents the average of the Monte-Carlo simulation and the blurred areas represent the minimum and maximum values obtained for all the Monte-Carlo 15 independent runs. In addition, the dashed lines represent target values. Best WOM and best awareness (lowest error) solutions are represented with pointed lines containing squares and crosses respectively. Trade-off solutions are represented using lines with circles.



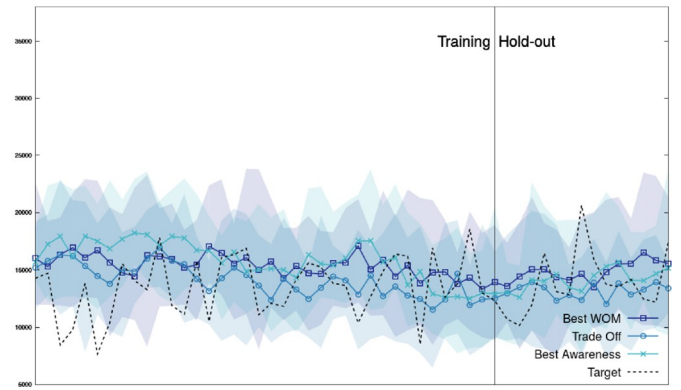
(a) P-25 : awareness fitting



(b) P-25 : WOM volume fitting



(c) P-55 : awareness fitting



(d) P-55 : WOM volume fitting

**Fig. 9.** Awareness output and WOM volume over time for P-25 and P-55 regarding brand 6. In these charts, the central line represents the average of the Monte-Carlo simulation and the blurred areas represent the minimum and maximum values obtained for all the Monte-Carlo 15 independent runs. In addition, the dashed lines represent target values. Best WOM and best awareness (lowest errors) solutions are represented with pointed lines containing squares and crosses respectively. Trade-off solutions are represented using lines with circles.

**Table 3**

Sample of the parameter values for the selected solutions for P-25 and P-55 instances. This sample includes a selection of parameters for the social network and different mass media channels.

| P-25      | $\tau_{c1}$ | $\alpha_{c3}$ | $\tau_{c4}$ | $d\tau_{c4}$ | $\alpha_{c7}$ | $d\tau_{c7}$ | $p_i^b(0)$ | $\alpha^{WOM}$ |
|-----------|-------------|---------------|-------------|--------------|---------------|--------------|------------|----------------|
| Best Aw.  | 0.159       | 0.212         | 0.207       | 0.728        | 0.2           | 0.09         | 0.29       | 0.152          |
| Trade-Off | 0.579       | 0.492         | 0.273       | 0.672        | 0.25          | 0.679        | 0.326      | 0.138          |
| Best WOM  | 0.935       | 0.625         | 0.655       | 0.762        | 0.709         | 0.023        | 0.289      | 0.833          |

| P-55      | $\tau_{c1}$ | $d\tau_{c2}$ | $\alpha_{c34}$ | $d\tau_{c4}$ | $\alpha_{c15}$ | $\tau_{c16}$ | $\alpha^{WOM}$ | $d$   |
|-----------|-------------|--------------|----------------|--------------|----------------|--------------|----------------|-------|
| Best Aw.  | 0.6         | 0.187        | 0.127          | 0.128        | 0.003          | 0.904        | 0.15           | 0.201 |
| Trade-Off | 0.629       | 0.043        | 0.404          | 0.004        | 0.02           | 0.988        | 0.175          | 0.2   |
| Best WOM  | 0.542       | 0.078        | 0.976          | 0.514        | 0.245          | 0.997        | 0.243          | 0.207 |

#### 4.4. Application to a practical problem

We begin the analysis and validation of the model instances P-25 and P-55 by visualizing the solutions of the generated Pareto set approximations with respect to the two conflicting objectives (see Fig. 7 with the two Pareto front approximations). We have selected three of the most representative solutions for the two instances: a) the solution with lowest awareness error, b) the solution with lowest WOM volume error, and c) the solution with the best trade-off for both objectives. In order to select the best trade-off solution we use the procedure followed in [6]. Briefly, we generate 1000 random weights  $w \in [0,1]$  and compute the average value of the aggregation function of both objectives  $f_1$  and  $f_2$ . Since the values of  $f_1$  are much bigger than the values of  $f_2$ , we

apply a normalization factor  $\delta$  in order to scale them  $\delta = \frac{1}{|s|} \sum_{i=1}^{|s|} \frac{f_2(s_i)}{f_1(s_i)}$ , where  $s$  is the set of solutions in the Pareto front. We formulate this process as  $F(s_i) = \frac{1}{1000} \sum_{j=1}^{1000} \delta \cdot w_j \cdot f_1(s_i) + (1 - w_j) \cdot f_2(s_i)$ . The selected solutions are highlighted in their respective Pareto front approximations in Fig. 7.

We visualize the outputs of the model using the calibrated configurations setting the focus on some specific brands in order to carry out an understandable analysis of the behavior of the selected solutions. The KPI evolution along the simulation steps for brands 3 and 6 is shown in Figs. 8 and 9 respectively. These brands were chosen because their behavior is a good resemblance of the rest of the brands for both objectives. These charts show the model output for both the training and the hold-out sets, where we can note that the latter obtain similar fitting than the former and therefore, the model can generalize well. Both figures show that adjusting the behavior the dynamics of the awareness evolution over time is harder than the WOM volume dynamics. In contrast, as already identified in the controlled experiment, WOM volume dynamics are more sensible to Monte-Carlo variability for both model instances (as seen in the blurred areas in Figs. 8b, d and 9b, d).

This is specially relevant for the P-25 instance, as shown in Figs. 8a and 9a, since the best calibrated solutions only capture the trend of the target values. In the case of the P-55 instance, the awareness output of the solutions is wavier but the resulting values are far from the target data and the final awareness error for this model instance ends up being greater (as seen in Fig. 7). However, this could be a consequence of the synthetically generated target values, that could be too difficult to match. We extract similar conclusions for the WOM volume objective.

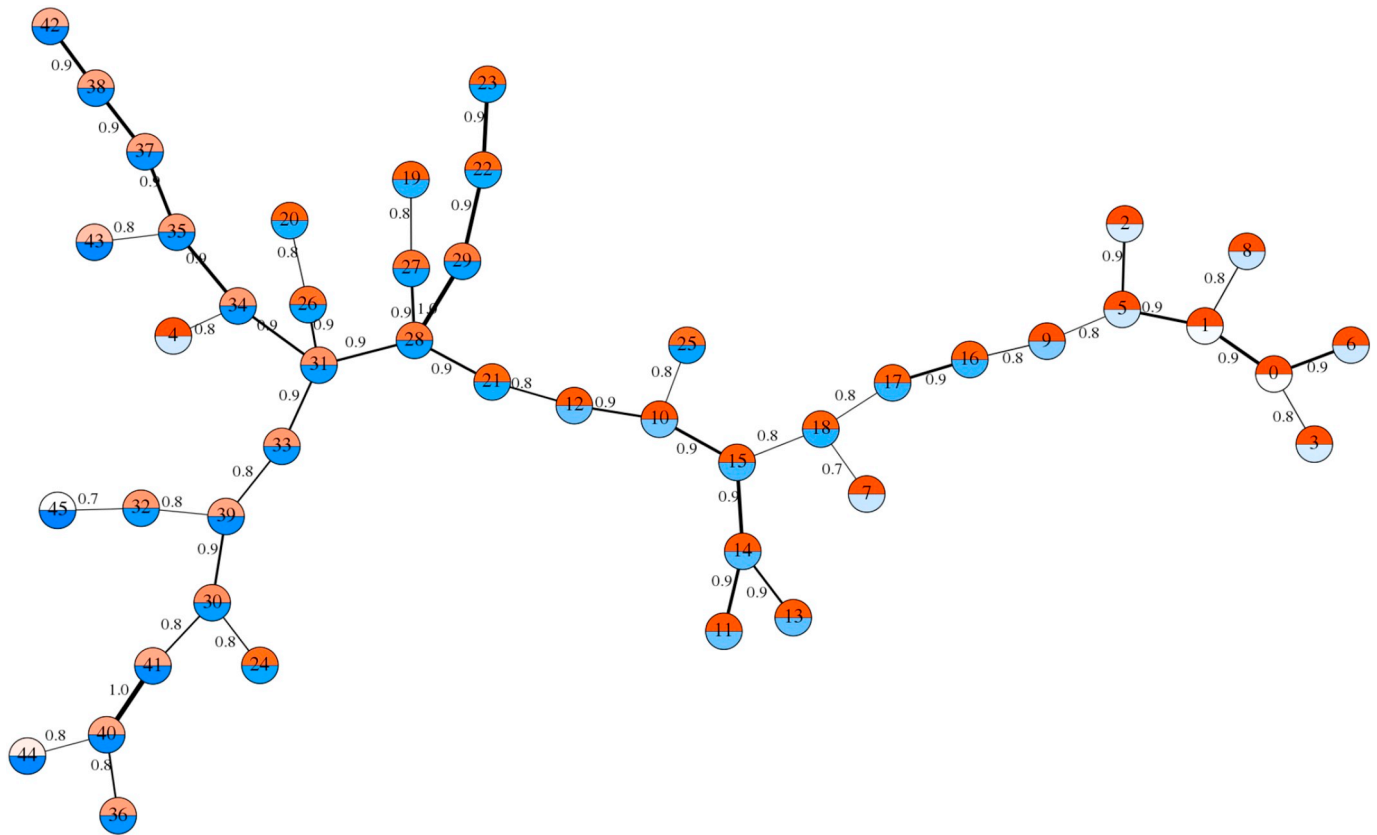


Fig. 10. moGrams network representing the non-dominated calibration solutions for P-25 model instance.

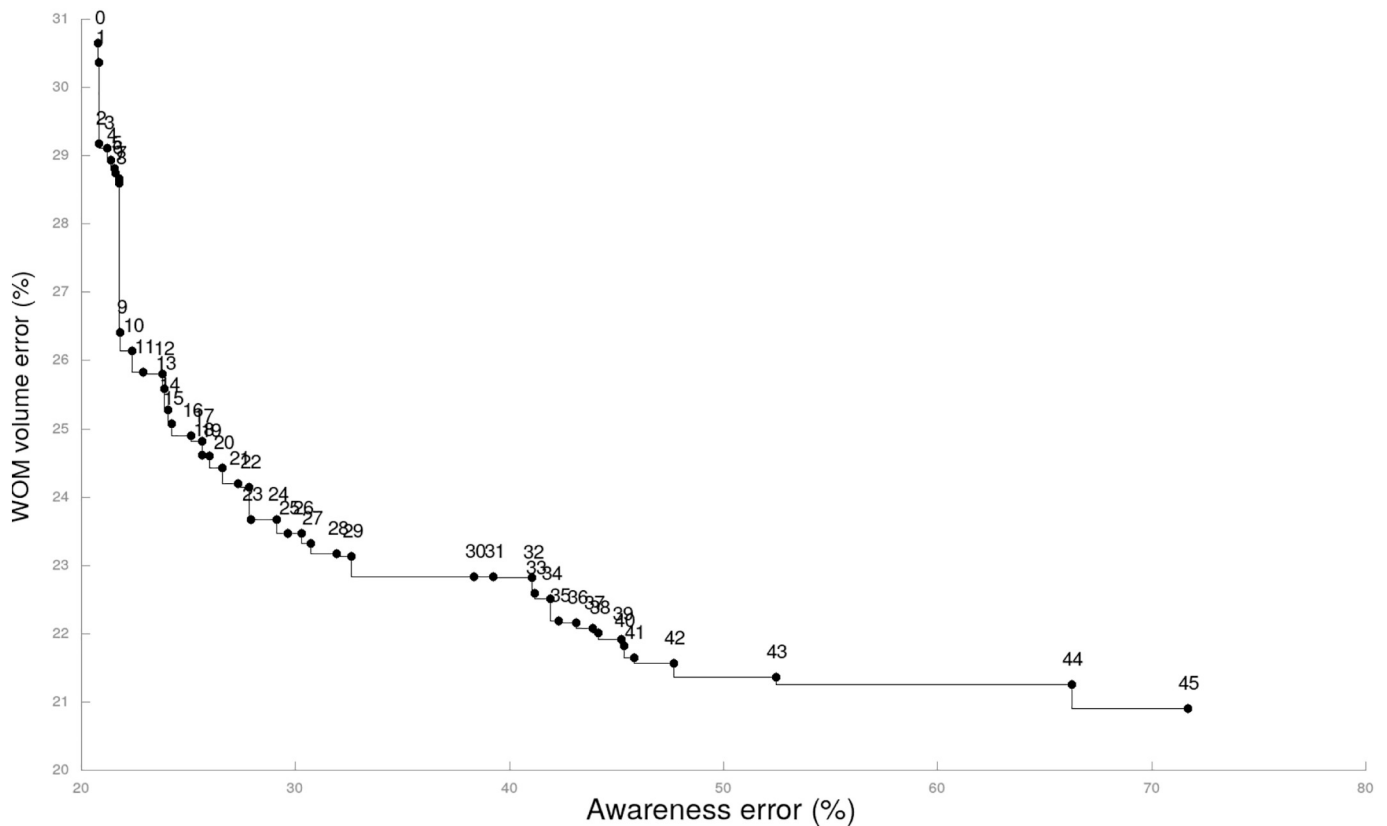


Fig. 11. Pareto front approximation for P-25 model instance associated to the moGram of Fig. 10.

Although trade-off and best awareness solutions achieve reasonable WOM volume outputs for the P-25 instance (Figs. 8b and 9b), final WOM volume errors are higher for the P-55 instance even when considering the fittest solutions (as shown in Figs. 8d and 9d).

Finally, Table 3 shows a sample of the parameter values of the selected solutions for P-25 and P-55 instances. This sample includes the values of the social network parameters and the different mass media channels. In these solutions we can observe how higher values for the awareness parameters benefit the fitting of WOM dynamics, since more awareness involves more conversations in the social network. However, these values also reduce awareness fitting. We can see that these values are consistent with the behavior shown in Figs. 8 and 9. For example, the solution with best trade-off and the solution with lowest WOM volume error (Best WOM in Table 3) for the P-25 instance have higher buzz increment values ( $\tau_{c1}$  and  $\tau_{c4}$ ) and higher social awareness impact ( $\alpha^{WOM}$ ). In contrast, the selected solutions for the P-55 instance show similar buzz increment values for all the selected solutions. In addition, these solutions also have similar values for the awareness deactivation probability ( $d$ ). This also shows how configurations with high awareness values reduce WOM error (improves WOM adjustment) while reducing awareness fitting.

#### 4.5. Visual and qualitative multicriteria analysis using moGrams

We continue the analysis of the calibrated model instances using moGrams, which composes the second stage of our framework. As said, moGrams is a visualization methodology that combines the visualization of both the design and the objective spaces that aids the decision maker enhancing her understanding of the problem [40]. Our approach is similar to the one followed during the behavior analysis: we apply moGrams to two Pareto set approximations obtained by NSGA-II for both P-25 and P-55 model instances. In order to perform moGrams generation, we need to define a similarity metric for our calibration problem. Our similarity metric  $Sim(X_i, X_j) \in [0,1]$  compares two solutions (i.e., set of calibrated parameters)  $X_i$  and  $X_j$  using the normalized Euclidean distance, since our calibration problem considers many independent decision variables. The similarity metric is defined in Eq. (5).

$$Sim(X_i, X_j) = 1 - \sqrt{\frac{(x_{i1} - x_{j1})^2 + \dots + (x_{in} - x_{jn})^2}{n}}. \quad (5)$$

The generated moGram for the P-25 model instance is shown in Fig. 10 and its associated Pareto front approximation is displayed in Fig. 11. We can see that, given the relatively high cardinality of the Pareto front approximation (46 solutions), the decision making process for this model instance seems too complex to deal without a visualization method. Following the moGrams methodology, each node in the generated network is associated to an individual solution (i.e., model parameter setting) from the Pareto set approximation. We draw each node as a pie where the upper pie segment represents the awareness error objective using degradation between orange and white while the lower pie segment represents the WOM volume error objective using degradation between blue and white. For both objectives, a more intense color means a better value with a white color being the worst possible value. In addition, we have included indexes for the solutions in both the network and the Pareto front approximation for making their relation clearer (see Figs. 10 and 11). We provide several observations from the moGrams visualization:

- Regarding the structure of the network, we can identify multiple clusters of solutions (i.e., groups of solutions) in the design space. Two of these subsets of solutions, located in the left side of the network, are connected to the general network by Solution 31, which bridges with another subset through Solution 28. In addition, Solution 15 can be identified as another hub that connects to another subset of solutions located in the right side of the network.

- From those clusters we can identify Solution 31 and Solution 28 as the most connected ones, since they are the only solutions with degree 4. Due to their connectivity and the additional information provided by moGrams, these solutions could be interesting configurations for the modeler. In terms of similarity, both solutions have values of 0.9, which suggests they have good flexibility and can be swapped by other solutions with minimum parameter changes.
- The moGrams visualization methodology assists us in validating the best trade-off solution (Solution 9, located in the right side of the map), which could be a suitable model configuration due to its good balance for both objectives. This solution is connected with Solutions 5 and 16 with similarity values of 0.8. However, these solutions are located in the same region of the Pareto front approximation, reducing the interest of swapping Solution 9 with any of its neighbors.
- With respect to the best solutions for each objective (Solutions 0 and 45), both of them are located in opposite regions of the network. Solution 0 has three neighbors with similarity values beyond 0.8, meanwhile Solution 45 has a single connection to Solution 32 with a low similarity value (0.7). However, the neighborhood of Solution 0 may not be really interesting, since all the solutions are close in the Pareto front approximation.

Fig. 12 shows the generated moGram for the P-55 model instance while its associated Pareto front approximation is displayed in Fig. 13. Similarly to the previous moGram, its associated Pareto front approximation has a relatively high cardinality (32 solutions). Again, we can provide the following observations and interesting insights for the modeler from a validation point of view:

- Due to the star topology of the network, three main subsets of solutions in the design space arise, which grow as tree-shaped sub-networks from Solution 13. We can identify Solution 16 as the most connected, since it is the only with four neighbors. This solution could be interesting for the modeler since its connections include solutions from opposite regions of the Pareto front approximations (i.e., Solution 31, the solution with the lowest WOM error and Solution 2, which is close to the solution with the lowest awareness error).
- In addition, several other solutions have three connections. From those, we could highlight Solution 2 and Solution 12 since their connections are more diverse and include solutions from other regions of the Pareto front approximation. Solution 12 can also be identified as the trade-off solution, which could be a plus for the modeler. The most interesting neighbor of Solution 12 is Solution 30 with a similarity value of 0.8.
- Finally, the best solutions for each objective are located in the same subset of solutions. As already pointed out, Solution 31 is connected to the hub defined by Solution 16. However, Solution 0 is isolated with a single connection to Solution 2, which belongs to the same region of the Pareto front approximation. Additionally, we can observe that the proximity in the network for both solutions could be relevant for the modeler, since it suggests that modifying the value of some sensible parameters can drastically change the behavior of the model.

The previous statement can be confirmed by sampling the parameters of Solutions 0 and 16, shown in Table 4. Additionally, we also included the parameters of the solutions with best trade-off and the solution with lowest WOM error. These parameters were selected because they show the largest differences among the selected configurations. It can be observed that most of these parameters are related with buzz increment and buzz decay. Solution 0 has lower increment and decay values of several mass media channels, suggesting that these parameters are really important to control the behavior of the model. Finally, we should note that the awareness impact values of Solution 31



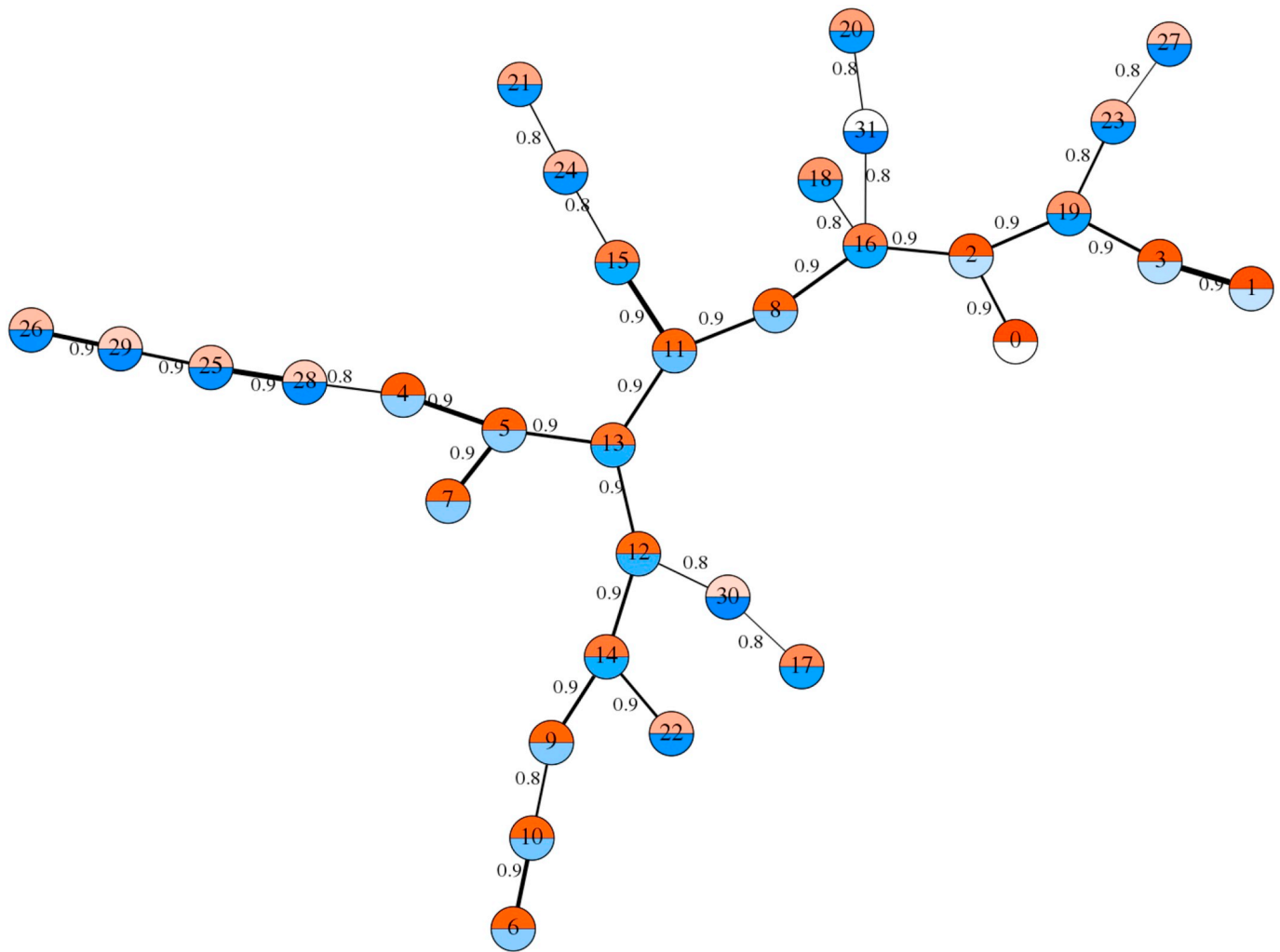


Fig. 12. moGrams network representing the non-dominated calibration solutions for P-55 model instance.

are higher than the values the other solutions, supporting our previous insight that higher awareness can benefit WOM fitting.

## 5. Final remarks

In this paper we have introduced a multicriteria integral framework for the calibration and validation of ABMs considering multiple objectives. The framework comprises an EMO method to search for the best set of configuration parameters and a visualization method to help the modeler in the decision making of the best model parameter setting. We have applied the novel multicriteria framework to an ABM for marketing scenarios, driven by awareness and WOM volume as KPIs.

We have designed and implemented our calibration approach using NSGA-II. The proposed framework has been validated in two different experiments: through a controlled experiment where we show that our approach identifies the optimal model configurations for each objective, and through its application to two model instances using real data. The first experiment has allowed us to demonstrate that the multi-objective optimization considered in the first stage is actually able to obtain high-quality calibrated models. In the second experiment, we have analyzed the latter resulting Pareto front approximations by selecting three solutions: the solution with best awareness error ( $f_1$ ), the solution with best WOM volume error ( $f_2$ ), and the solution with best trade-off for both objectives. Our analysis suggested that awareness dynamics were more difficult to adjust than the WOM volume for the calibrated instances, specially for P-55, the instance with highest

dimensionality. Due to the reasonable fitting behavior for the baseline model, we can conclude that the increasing dimensionality of the problem influences the fitting of the resulting models.

Finally, we have applied the second stage of our framework by analyzing the design space for our calibration problem using moGrams on individual Pareto front approximations from P-25 and P-55 instances. This analysis has shown the usefulness of our framework when validating relevant solutions and assessing their flexibility (i.e., the solution with best trade-off for both objectives) from the Pareto front approximation. We have concluded that the solutions with the best trade-off had good flexibility but they did not have interesting neighboring solutions in the decision space. In contrast, other solutions with higher degree had the potential of being more relevant for the modeler. We could also notice that analyzing a Pareto set approximation with this high cardinality (31 and 46 solutions, respectively) without a visualization methodology such as moGrams can be difficult for modelers and stakeholders. For example, it would have been tricky to identify and validate alternative solutions (Solution 16 from Fig. 12 would be an example). Thus, we have shown moGrams is a powerful resource for aiding modelers when dealing with multiobjective model calibration problems.

As future works, we consider extending our designed consumer model for including brand sales as an additional KPI. In addition, including new objectives could involve replacing NSGA-II with another EMO algorithm that could be able to successfully deal with many-objective optimization. Due to the high cardinality of the Pareto set

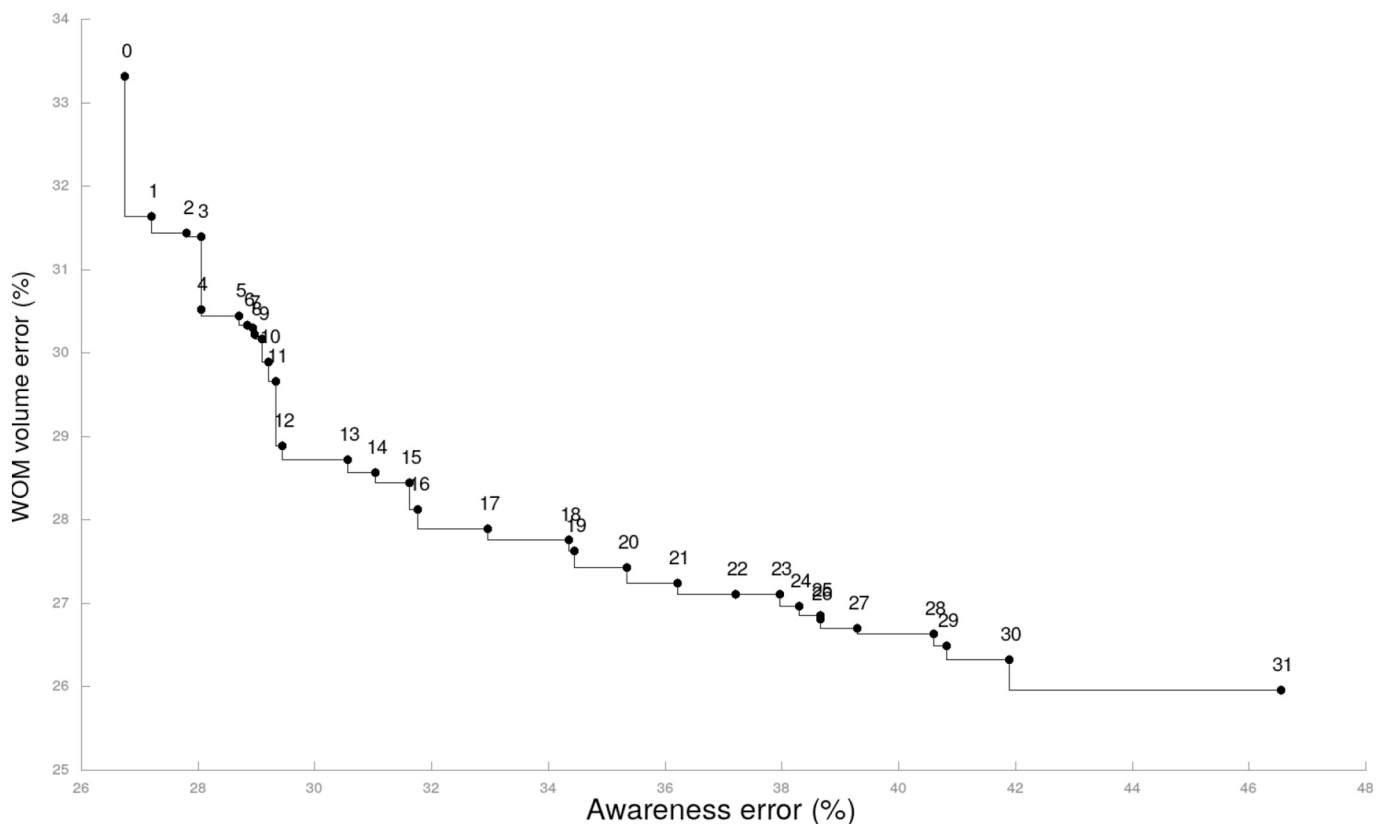


Fig. 13. Pareto front approximation for P-55 model instance associated to the moGram of Fig. 12.

Table 4

Sample of the parameter values for the identified solutions for P-55 using its moGram at Fig. 12.

|             | $d\tau_{c1}$ | $\alpha_{c3}$ | $\tau_{c3}$ | $\tau_{c4}$ | $d\tau_{c4}$ | $d\tau_{c6}$ | $\tau_{c9}$ | $\alpha_{c10}$ |
|-------------|--------------|---------------|-------------|-------------|--------------|--------------|-------------|----------------|
| Solution 0  | 0.041        | 0.127         | 0.172       | 0.407       | 0.128        | 0.370        | 0.954       | 0.045          |
| Solution 2  | 0.626        | 0.563         | 0.718       | 0.468       | 0.112        | 0.440        | 0.794       | 0.071          |
| Solution 12 | 0.133        | 0.404         | 0.852       | 0.855       | 0.004        | 0.456        | 0.492       | 0.079          |
| Solution 16 | 0.602        | 0.560         | 0.199       | 0.873       | 0.010        | 0.473        | 0.958       | 0.095          |
| Solution 31 | 0.555        | 0.976         | 0.221       | 0.922       | 0.514        | 0.918        | 0.417       | 0.469          |

approximations delivered during our experiments, we also consider interesting to extend our calibration approach to evaluate the impact of including the modeler's preferences during the calibration process [39].

## Acknowledgments

This work is supported by Spanish Ministerio de Economía y Competitividad under the EXASOCO project (ref. PGC2018-101216-B-I00), including European Regional Development Fund (ERDF). M. Chica is also supported through the Ramón y Cajal program (RYC-2016-19800).

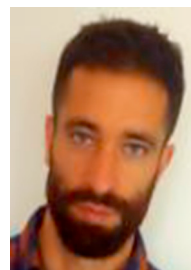
## References

- [1] S. Atamturktur, Z. Liu, S. Cogan, H. Juang, Calibration of imprecise and inaccurate numerical models considering fidelity and robustness: a multi-objective optimization-based approach, *Struct. Multidiscip. Optim.* 51 (3) (2015) 659–671.
- [2] J. Bader, E. Zitzler, HypE: an algorithm for fast hypervolume-based many-objective optimization, *Evol. Comput.* 19 (1) (2011) 45–76.
- [3] A.L. Barabási, R. Albert, Emergence of scaling in random networks, *Science* 286 (5439) (1999) 509–512.
- [4] N.D. Bennett, B.F. Croke, G. Guariso, J.H. Guillaume, S.H. Hamilton, A.J. Jakeman, S. Marsili-Libelli, L.T. Newham, J.P. Norton, C. Perrin, S.A. Pierce, B. Robson, R. Seppelt, A.A. Voinov, B.D. Fath, V. Andreassian, Characterising performance of environmental models, *Environ. Model. Softw.* 40 (2013) 1–20.
- [5] E. Bonabeau, Agent-based modeling: methods and techniques for simulating human systems, *Proc. Natl. Acad. Sci.* 99 (2002) 7280–7287.
- [6] B.R. Campomanes-Álvarez, O. Cerdón, S. Damas, Evolutionary multi-objective optimization for mesh simplification of 3D open models, *Integrated Comput. Aided Eng.* 20 (4) (2013) 375–390.
- [7] M. Chica, J. Barranquero, T. Kajdanowicz, O. Cerdón, S. Damas, Multimodal optimization: an effective framework for model calibration, *Inform. Sci.* 375 (2017) 79–97.
- [8] M. Chica, Óscar Cerdón, S. Damas, V. Iglesias, J. Mingot, Identimod: modeling and managing brand value using soft computing, *Decis. Support. Syst.* 89 (2016) 41–55.
- [9] M. Chica, W. Rand, Building agent-based decision support systems for word-of-mouth programs: a freemium application, *J. Market. Res.* 54 (5) (2017) 752–767.
- [10] K. Deb, Multi-objective Optimization Using Evolutionary Algorithms, 16 John Wiley & Sons, 2001.
- [11] K. Deb, H. Jain, An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints, *IEEE Trans. Evol. Comput.* 18 (4) (2014) 577–601.
- [12] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Trans. Evol. Comput.* 6 (2) (2002) 182–197.
- [13] J.M. Epstein, *Generative Social Science: Studies in Agent-based Computational Modeling*, Princeton University Press, 2006.
- [14] J.D. Farmer, D. Foley, The economy needs agent-based modelling, *Nature* 460 (7256) (2009) 685–686.
- [15] P.W. Farris, N.T. Bendle, P.E. Pfeifer, D.J. Reibstein, *Marketing Metrics: The Definitive Guide to Measuring Marketing Performance*, second, Wharton School Publishing, 2010.
- [16] J.C. González-Avella, M.G. Cosenza, K. Klemm, V.M. Eguíluz, M. San Miguel, Information feedback and mass media effects in cultural dynamics, *J. Artif. Soc. Soc. Simul.* 10 (3) (2007) 9.
- [17] S. Hassan, J. Arroyo, J.M. Galán, L. Antunes, J. Pavón, Asking the oracle: introducing forecasting principles into agent-based modelling, *J. Artif. Soc. Soc. Simul.* 16 (3) (2013) 13.
- [18] R.J. Hyndman, A.B. Koehler, Another look at measures of forecast accuracy, *Int. J. Forecast.* 22 (4) (2006) 679–688.
- [19] M.A. Janssen, E. Ostrom, Empirically based, agent-based models, *Ecol. Soc.* 11 (2) (2006) 37.
- [20] J.-S. Lee, T. Filatova, A. Ligmann-Zielinska, B. Hassani-Mahmoei, F. Stonedahl, I. Lorscheid, A. Voinov, J.G. Polhill, Z. Sun, D.C. Parker, The complexities of agent-based modeling output analysis, *J. Artif. Soc. Soc. Simul.* 18 (4) (2015) 4.
- [21] H. Li, Q. Zhang, Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II, *IEEE Trans. Evol. Comput.* 13 (2) (2009) 284–302.
- [22] B. Libai, E. Muller, R. Peres, Decomposing the value of word-of-mouth seeding programs: acceleration versus expansion, *J. Market. Res.* 50 (2) (2013) 161–176.
- [23] M. López-Ibáñez, J. Dubois-Lacoste, L.P. Cáceres, M. Birattari, T. Stützle, The irace package: iterated racing for automatic algorithm configuration, *Oper. Res. Perspect.* 3 (2016) 43–58.

- [24] C.M. Macal, M.J. North, Tutorial on agent-based modeling and simulation, *Proceedings of the 37th Conference on Winter Simulation*, ACM, 2005, pp. 2–15.
- [25] E.K. Macdonald, B.M. Sharp, Brand awareness effects on consumer decision making for a common, repeat purchase product: a replication, *J. Bus. Res.* 48 (1) (2000) 5–15.
- [26] J.H. Miller, Active nonlinear tests (ANTs) of complex simulation models, *Manag. Sci.* 44 (6) (1998) 820–830.
- [27] I. Moya, M. Chica, J.L. Sáez-Lozano, Ó. Cordón, An agent-based model for understanding the influence of the 11-M terrorist attacks on the 2004 Spanish elections, *Knowl.-Based Syst.* 123 (2017) 200–216.
- [28] G. Narzisi, V. Mysore, B. Mishra, Multi-objective evolutionary optimization of agent-based models: an application to emergency response planning, *Proceedings of the 2nd IASTED International Conference on Computational Intelligence*, CI 2006, 2006, pp. 224–230.
- [29] A.J. Nebro, J.J. Durillo, M. Vergne, Redesigning the jMetal multi-objective optimization framework, *Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation*, ACM, New York, NY, USA, 2015, pp. 1093–1100.
- [30] M. Newman, A.L. Barabási, D.J. Watts, *The Structure and Dynamics of Networks*, Princeton University Press, 2006.
- [31] R. Oliva, Model calibration as a testing strategy for system dynamics models, *Eur. J. Oper. Res.* 151 (3) (2003) 552–568.
- [32] M.N. Read, K. Alden, L.M. Rose, J. Timmis, Automated multi-objective calibration of biological agent-based simulations, *J. R. Soc. Interface* 13 (122) (2016).
- [33] W. Samek, T. Wiegand, K. Müller, Explainable artificial intelligence: understanding, visualizing and interpreting deep learning models, *CoRR abs/1708.08296* (2017).
- [34] R.G. Sargent, Verification and validation of simulation models, *Proceedings of the 37th Conference on Winter Simulation*, 2005, pp. 130–143.
- [35] M.E. Schramm, K.J. Trainor, M. Shanker, M.Y. Hu, An agent-based diffusion model with consumer and brand agents, *Decis. Support. Syst.* 50 (1) (2010) 234–242.
- [36] R.W. Schvaneveldt, F.T. Durso, D.W. Dearholt, *Network Structures in Proximity Data*, Psychology of Learning and Motivation, vol. 24, Academic Press, 1989, pp. 249–284.
- [37] F. Stonedahl, W. Rand, When does simulated data match real data? Comparing model calibration functions using genetic algorithms, *Advances in Computational Social Science, Agent-Based Social Systems*, 11 Springer, Japan, 2014, pp. 297–313.
- [38] E.G. Talbi, *Metaheuristics: From Design to Implementation*, John Wiley & Sons, 2009.
- [39] L. Thiele, K. Miettinen, P.J. Korhonen, J. Molina, A preference-based evolutionary algorithm for multi-objective optimization, *Evol. Comput.* 17 (3) (2009) 411–436.
- [40] K. Trawiński, M. Chica, D.P. Pancho, S. Damas, O. Cordón, moGrams: a network-based methodology for visualizing the set of nondominated solutions in multi-objective optimization, *IEEE Trans. Cybernetics* 48 (2) (2018) 474–485.
- [41] T. Tušar, B. Filipič, Visualization of Pareto front approximations in evolutionary multiobjective optimization: a critical review and the prosecution method, *IEEE Trans. Evol. Comput.* 19 (2) (2015) 225–245.
- [42] A. Voinov, N. Kolagani, M.K. McCall, P.D. Glynn, M.E. Kragt, F.O. Ostermann, S.A. Pierce, P. Ramu, Modelling with stakeholders-next generation, *Environ. Model. Softw.* 77 (2016) 196–220.
- [43] M.M. Waldrop, Free agents, *Science* 360 (6385) (2018) 144–147.
- [44] D.J. Walker, R. Everson, J.E. Fieldsend, Visualizing mutually nondominating solution sets in many-objective optimization, *IEEE Trans. Evol. Comput.* 17 (2) (2013) 165–184.
- [45] D.J. Watts, S.H. Strogatz, Collective dynamics of ‘small-world’ networks, *Nature* 393 (6684) (1998) 440–442.
- [46] F. Wu, B.A. Huberman, Novelty and collective attention, *Proc. Natl. Acad. Sci.* 104 (45) (2007) 17599–17601.
- [47] J. YangLeskovec, J., Modeling information diffusion in implicit networks, 2010 *IEEE International Conference on Data Mining*, IEEE, 2010, pp. 599–608.
- [48] S. Yang, M. Li, X. Liu, J. Zheng, A grid-based evolutionary algorithm for many-objective optimization, *IEEE Trans. Evol. Comput.* 17 (5) (2013) 721–736.
- [49] M.A. Zaffar, R.L. Kumar, K. Zhao, Diffusion dynamics of open source software: an agent-based computational economics (ACE) approach, *Decis. Support. Syst.* 51 (3) (2011) 597–608.
- [50] M.A. Zaffar, R.L. Kumar, K. Zhao, Using agent-based modelling to investigate diffusion of mobile-based branchless banking services in a developing country, *Decis. Support. Syst.* (2018) (in press).
- [51] X. Zhang, Y. Tian, Y. Jin, A knee point-driven evolutionary algorithm for many-objective optimization, *IEEE Trans. Evol. Comput.* 19 (6) (2015) 761–776.
- [52] Y. Zhang, Q. Shao, J.A. Taylor, A balanced calibration of water quantity and quality by multi-objective optimization for integrated water system model, *J. Hydrol.* 538 (2016) 802–816.
- [53] E. Zitzler, M. Laumanns, L. Thiele, SPEA2: improving the strength Pareto evolutionary algorithm, Technical Report, 103 Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH), Zurich, 2001.
- [54] E. Zitzler, L. Thiele, M. Laumanns, C.M. Fonseca, V.G. Da Fonseca, Performance assessment of multiobjective optimizers: an analysis and review, *IEEE Trans. Evol. Comput.* 7 (2) (2003) 117–132.



**Ignacio Moya** received his M.Sc. degree in Computer Science in 2013 from the Complutense University of Madrid. He joined the European Centre for Soft Computing as a research assistant from 2014 to 2016. He is currently a Ph.D. Candidate at University of Granada with a scholarship granted by the Spanish Ministry of Economy. His research is currently focused in agent-based modeling and social simulation, with special interest in model calibration and validation.



**Manuel Chica** B.Sc., M.Sc. in Computer Science, obtained his PhD degree cum laude from the University of Granada in 2011. Currently, he is “Ramón y Cajal” Senior Researcher at the University of Granada, granted by the Spanish government, and Chief A.I. Officer and scientific partner for ZIO, a SME applying computational intelligence and agent-based modeling to marketing. Additionally, he is Conjoint Lecturer at the University of Newcastle, Australia, after being a post-doctoral Endeavour Research Fellowship. He is a co-inventor of an international patent under exploitation. He has published more than 70 peer-reviewed scientific papers, 26 papers in JCR-indexed journals (20 of them as the first author). He has participated in 18 research projects, playing the role of Principal Investigator in 2

European FP7s, 3 National projects, and 4 research contracts, with a budget of more than 3 million euros.



**Oscar Cordón** is Full Professor with the University of Granada. He was the founder and leader of that University's Virtual Learning Center between 2001 and 2005, and is the Vice-President for Digital University since 2015. From 2006 to 2011 he was one of the founding researchers of the European Centre for Soft Computing, being contracted as Distinguished Affiliated Researcher until December 2015. He has been, for 25 years, an internationally recognized contributor to R&D Programs in fundamentals and real-world applications of computational intelligence. He has published more than 360 peer-reviewed scientific publications including a research book on genetic fuzzy systems (with ~1300 citations in Google Scholar) and 101 JCR-SCI-indexed journal papers (58 in Q1), advised 18 Ph.D. dissertations, and coordinated 32 research projects and contracts (with an overall amount of 8.7M€).

By April 2019, his publications had received 4514 citations (h-index = 35), being included in the 1% of most-cited researchers in the world (source: Thomson's Web of Knowledge, 12,885 citations and h-index = 52 in Google Scholar). He also has a granted international patent on an intelligent system for forensic identification commercialized in Mexico and South Africa. He received the UGR Young Researcher Career Award in 2004; the IEEE Computational Intelligence Society (CIS) Outstanding Early Career Award in 2011 (the first such award conferred); the IFSA Award for Outstanding Applications of Fuzzy Technology in 2011; and the National Award on Computer Science ARITMEL by the Spanish Computer Science Scientific Society in 2014. He was elevated to IEEE Fellow in 2018 and received the IFSA Fellowship in 2019. He is currently or was Associate Editor of 18 international journals, and was recognized as IEEE TRANSACTIONS ON FUZZY SYSTEMS Outstanding AE in 2008. Since 2004, he has taken many different representative positions with EUSFLAT and the IEEE CIS.