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

Ignacio Moya[*,a], Manuel Chica[a,b], Óscar Cordón[a]

[a] *Andalusian Research Institute DaSCI "Data Science and Computational Intelligence", University of Granada, 18071 Granada, Spain*
[b] *School of Electrical Engineering and Computing, The University of Newcastle, Callaghan, NSW 2308, Australia*

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. Nevertheless, 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. On the one hand, evolutionary multiobjective optimization provides, in a single run, different sets of calibration solutions (i.e., parameters' values) with different trade-offs for the considered objectives. 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 present the calibration of two 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).

Keywords— Agent-based modeling, model calibration, evolutionary multiobjective optimization, information visualization.

[*] Corresponding author
Email addresses: imoya@ugr.es (Ignacio Moya), manuelchica@ugr.es (Manuel Chica), ocordon@decsai.ugr.es (Óscar Cordón)

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 [13, 40]. The ABM methodology [5, 12, 21] 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 allow us to represent complex and emerging dynamics as well as to define what-if scenarios and to forecast hypothetical scenarios [17]. 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, 28].

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 [28, 31]. Automated calibration requires a set of historical data, an error measure, and an optimization method for modifying the parameters in a systematic way and 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 [35] that can search across a large span of the model parameter space [7, 23, 34]. 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 [35]. In addition, often modelers choose to build their models considering two or more conflicting criteria or key performance indicators (KPI). Calibrating those models involve a multicriteria decision making process, since there are multiple configurations that are valid and satisfy at least one model's output.

In this multicriteria scenario, the validation of the model 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 when they have more than one KPI by introducing a multicriteria integral framework. This framework combines two elements: evolutionary multiobjective optimization (EMO) [9] algorithms for automatically calibrating the models' parameters from historical data, and an advanced visualization method that enhances the understanding of the calibration process and results. We show with our integrated approach 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 [28, 31]. 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 [25, 29], 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* [37] for the visualization method that represents 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).

We show how our integral approach improves the validation process by presenting the calibration of two instances of an ABM for marketing with different dimensionality (i.e., number of decision variables). These two examples are a specific case of our general proposal, which deals with the problem of calibration and validation of models regardless of its application domain. Since our ABM for marketing scenarios considers two conflicting objectives, we have selected the most extended EMO algorithm, the NSGA-II [11], to be used in our experimentation. Finally, the behavior of the resulting solutions is analyzed by visualizing their parameter settings using moGrams to generate insights about the decision space of the problem and help with the validation process.

The structure of this paper is as follows. We introduce our approach for multiobjective model

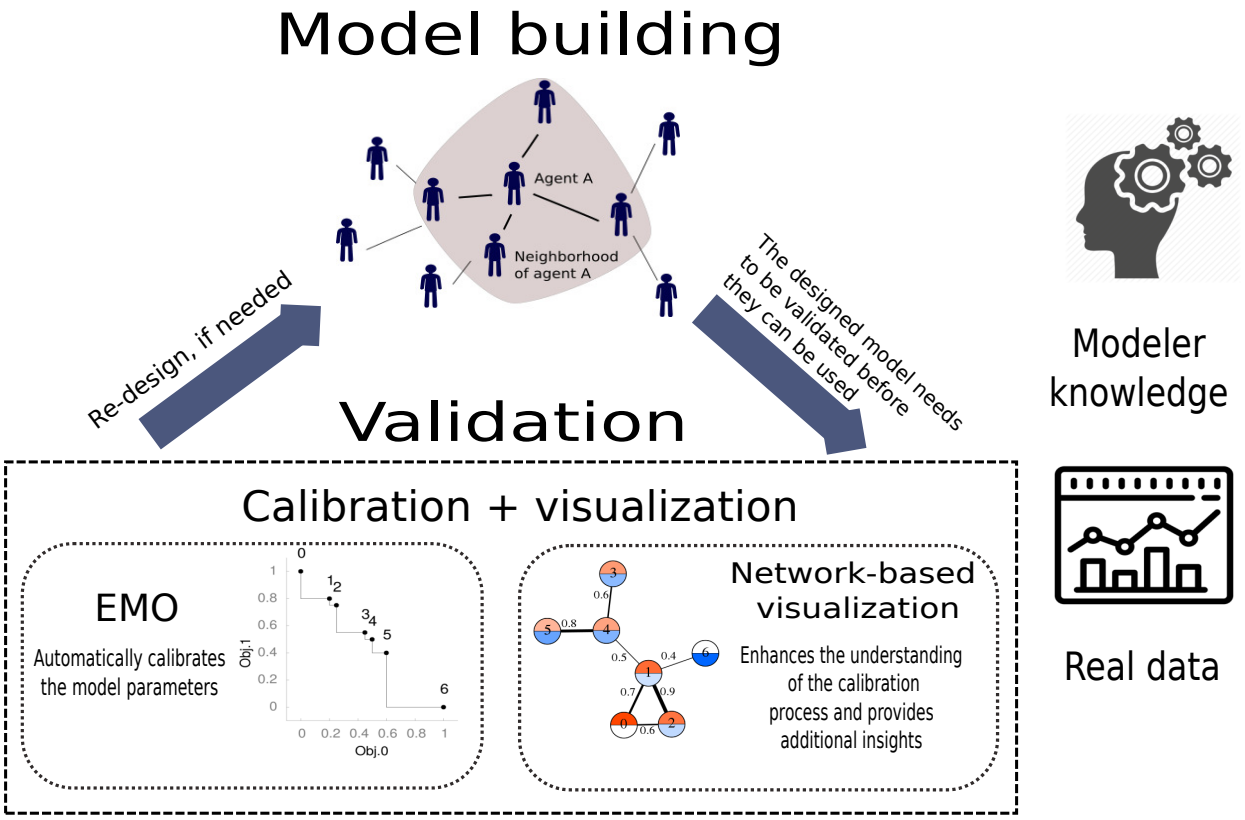


Figure 1: Diagram illustrating the components of our multicriteria integral framework for model calibration.

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 Figure 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 $X = (x_1, ..., x_n)$ as a vector of n decision variables. The model parameters selected to be estimated by automated calibration should be the most uncertain and the hardest to define by the modeler, since the difficulty of validating the calibrated configurations increases with the number of calibrated parameters. 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 KPI. This fitting is computed using a deviation measure ϵ 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 [16], for instance) for the defined objectives. This way, the goal of the optimization algorithm is to minimize $F(X) = (f_1(X), ..., f_M(X))$, where M represents the number of objectives. Each fitness function $f_j(X)$ computes the error associated to objective j, and is defined by Equation 1, where $\widetilde{o}^j$ 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.

$$\min F(X) = \min (f_1(X), ..., f_j(X), ..., f_M(X)), \text{ where } f_j(X) = \epsilon^j(o^j(X), \widetilde{o}^j). \tag{1}$$

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 such as NSGA-II [11], SPEA2 [50], or MOEA/D [19] can be considered. Otherwise, the calibration problem should be treated as a many-objective optimization problem and the selected EMO should perform properly in this environment. Some examples of EMO algorithms for many-objective optimization would be NSGA-III [10], HypE [2], GrEA [45], or KnEA [48].

2.3. Visualization for model calibration

The effective use of 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 recreates a given behavior, since the simulation of ABMs is frequently used for defining *what-if* scenarios and forecast hypothetical scenarios [17, 39]. This can be achieved from a white-box perspective [30], where both modelers and stakeholders are provided and 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 [30]. 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 during model calibration 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, 18]. Thus, visually showing the underlying relationships between an input configuration and its corresponding model's output becomes a critical component of the validation process.

However, if the model considers two or more conflicting outputs, multiobjective 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

[38, 41]. 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 even less proposals derive joint visualizations for both the objective and design spaces [37]. One of the few existing approaches is the *moGrams* methodology [37], which mutually analyzes and visualizes the solutions delivered by EMO algorithms in the decision and objective spaces.

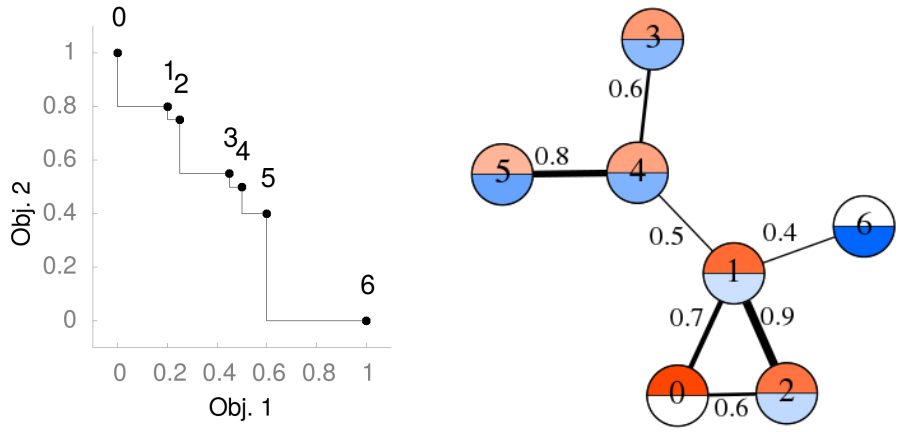
2.4. moGrams

moGrams [37] represents non-dominated solutions 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 [33] 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.

Figure 2 shows an example of moGrams generated for a Pareto set of a problem with two objectives. Additionally, we also show the original Pareto set for the generated moGrams network, 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, being the latter 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 set. 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 networks enables 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.



(a) Pareto set. (b) moGram representation.

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

3. Description of the agent-based model for marketing

This section describes the main features of the 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 we model the mass media channels. Finally, in Section 3.4 we summarize the parameters of the model selected for calibration and we describe the conflict of adjusting both KPIs of the model with their corresponding fitting functions in Section 3.5.

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

Our proposed model simulates a given number of weeks (T) 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 key performance indicators (KPIs): brand awareness and WOM volume (i.e., the

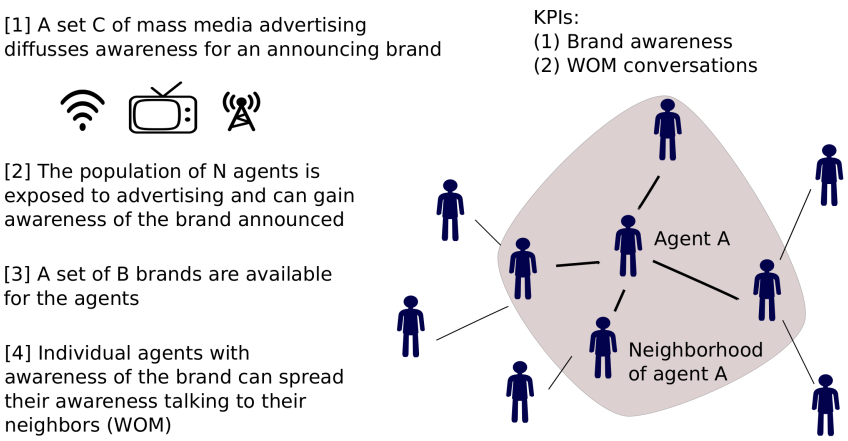


Figure 3: General scheme and structure of the ABM with an example of a brand advertised using mass media. The agents exposed to advertising can gain awareness of the brand announced and talk about it to their neighbors.

number of WOM interactions among the consumers). We select these KPIs because they have an important role in market expansion [20, 22]. A general scheme of our model is presented in Figure 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.

The awareness values of the agents do not remain static and change during the simulation: agents may loose 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 decay process [43, 44].

We model these losing/gaining effects with additional parameters. The parameter regulating the rate at which awareness is lost over time is called awareness decay ($d \in [0, 1]$). This parameter is modeled as follows: at the start of each step t, the agent i checks each brand b it is aware of ($a_i^b(t) = 1$). Each of these awareness values will be deactivated with a probability d by setting $a_i^b(t) = 0$. 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, 42]. We model this social network using an artificial scale-free network [3] because many real networks match with this network model [3, 27]. In these kinds of networks, the degree distribution follows a power law [3]. It 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's 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 [27, 32, 46, 47]. 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 [15] using a similar modeling approach to the one applied in the social network. The external influences are parame-

terized 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 [14], 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 scheduling algorithm for modeling impacts of the media channels over the population.

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

```
1  begin
2      reach_step = 0;
3      total_hits = χ_c^b(t) · 0.01 · 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 ≤ r_c then
12             impact agent;
13             i++;
14             reach_step += reach_increment;
```

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 α_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 [24]. 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 (τ_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 (σ_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 Equation 2.

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

3.4. Parameters selected for calibration

A summary of the complete set of model parameters is listed in Table 1. From those, we select the parameters that either modify the agent's 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 decay (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 $\{2,...,8\}$ 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

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 decay probability in the model
m	Parameter for social network generator
$p_i^b(0)$	Initial talking probability, same value for each brand b
α^{WOM}	Awareness impact for social interactions
χ_c^b	GRP units invested by brand b in channel c
r_c	Reach for mass media channel c
α_c	Awareness impact for mass media channel c
τ_c	Buzz increment for mass media channel c
$d\tau_c$	Buzz decay for mass media channel c

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

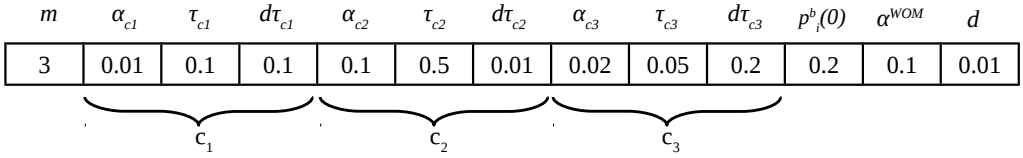


Figure 4: Coding scheme for a model instance with three mass media channels. The first gene (m parameter) is an integer value bounded to $\{2, ..., 8\}$. The rest of genes in the chromosome represent the real-coded parameters and are defined in $[0, 1]$.

social parameters. Briefly, those parameters are the following:

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

Therefore, the number of calibration parameters is $3 \cdot |C| + 4$. Figure 4 shows the coding scheme for a model instance using three mass media channels.

3.5. KPI fitting functions

Equations 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 $\widetilde{a}$ and $\widetilde{\omega}$ 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.

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

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

We can observe that both outputs of the model cannot be adjusted at the same time. 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 that jointly adjusts both KPIs to the given target data.

Finally, a sampling of the decision space is shown in Figure 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. We can observe how both objectives are in conflict since the individuals that have the lower deviation for awareness (f_1) also have a high deviation for WOM volume (f_2). Thus, both objectives are clearly not correlated and most sampled values are gathered at extreme locations of the decision space.

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 show the application of the visualization method to the calibration results and analyzes the composition of the non-dominated solutions from the decision maker's point of view.

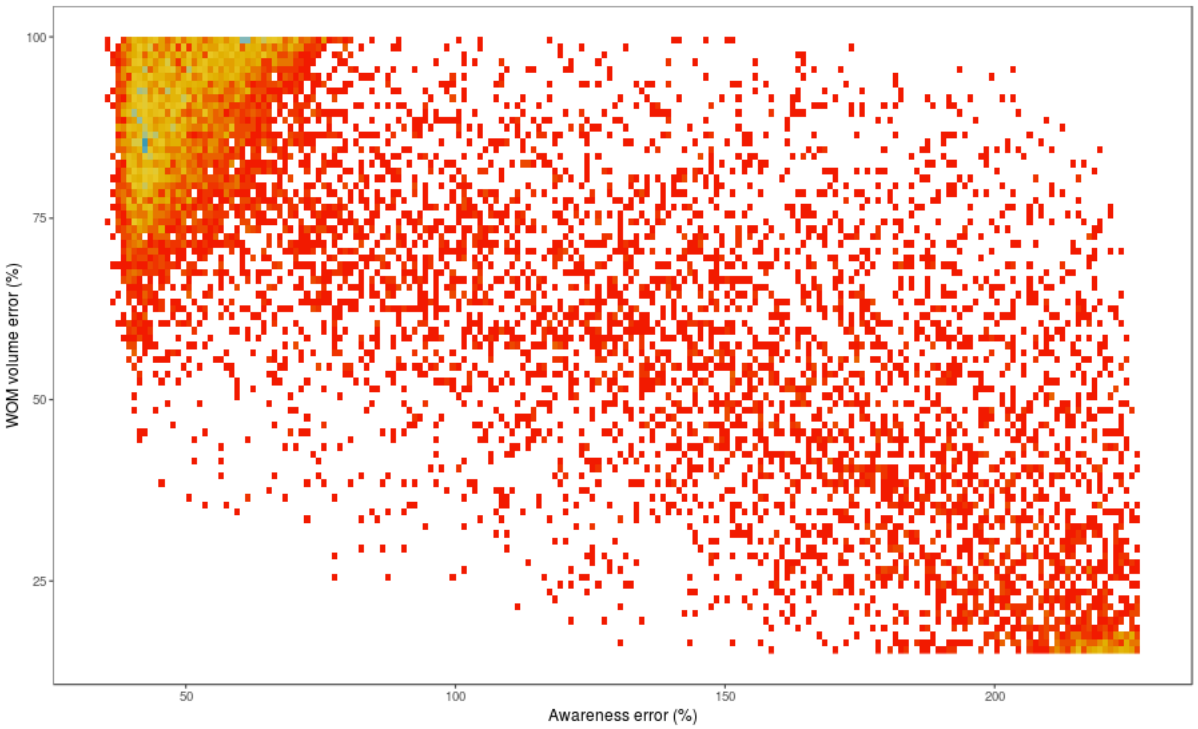


Figure 5: Sampling of the decision space for the two objectives. The values are shown as a scatter plot, where locations gathering multiple individuals have more intense colors.

4.1. Experimental setup

Our experimentation considers two different instances of the ABM model. These instances are generated starting form 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 an additional instance that include variations of the initial instance. This model variation includes ten 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 previous one by adding new decision variables to enable a more complete analysis of the algorithms' performance.

On the one hand, the perturbations to build P-55 consist of multiplying the investment of each brand at each time step by a given factor. We consider reductions in the original investment by 15%, 30%, 45%, and 60%. In addition, we increase the original investment by 100%, 200%, 300%, and 400%. The decision of whenever increasing or decreasing a brand investment is made at random and remains constant for each step. On the other hand, the modifications on the target

Baseline instance P-25							
Name	**Value**	**Name**	**Value**	**Name**	**Value**	**Name**	**Value**
N	1000	$\|B\|$	8	$\|C\|$	7	T	52
$a^{b_1}(0)$	0.709	$a^{b_2}(0)$	0.757	$a^{b_3}(0)$	0.589	$a^{b_4}(0)$	0.2559
$a^{b_5}(0)$	0.081	$a^{b_6}(0)$	0.429	$a^{b_7}(0)$	0.395	$a^{b_8}(0)$	0.34
r_{c_1}	0.928	r_{c_2}	0.579	r_{c_3}	0.548	r_{c_4}	0.035
r_{c_5}	0.432	r_{c_6}	0.382	r_{c_7}	0.696	$p_i^b(0)$	0.1

Table 2: Summary of the configuration of the baseline model instance P-25 for parameters that are not modified during calibration. The parameters are presented as name-value pairs.

historical values for both objectives involve directly adding or subtracting a different value for each brand to each of its time steps. In order to avoid unrealistic values we truncate the resulting historical values: the awareness values cannot exceed the $[0, 1]$ interval and WOM volume values have a minimum value of 0. Similarly to mass media investment, the decision of whether increasing or decreasing the target values is made at random and remains constant for each step.

Table 2 shows a summary of the configuration for the baseline instance P-25 showing the parameter values that are not modified during calibration. Notice that, there are $T = 52$ time steps corresponding to the weeks of a year of simulation. The generated instance P-55 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 $r_{c_9} = r_{c_5}$.

4.2. NSGA-II as the EMO algorithm

We select NSGA-II as our EMO algorithm during our experiments. NSGA-II [11] has become one of the most well-known EMO algorithms and there are several applications in model calibration [1, 29, 49]. 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 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 polynomial mutation [9] as our mutation strategy. It modifies the values of an individual's 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). Regarding the crossover strategy, we choose simulated binary crossover (SBX) [9] with a crossover 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 $P_1 = (p_{11}, ..., p_{1n})$ and $P_2 = (p_{21}, ..., p_{2n})$, SBX generates two springs $C_1 = (c_{11}, ..., c_{1n})$ and $C_2 = (c_{21}, ..., c_{2n})$ 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.

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. During each evaluation of an individual, the model is evaluated by 15 Monte-Carlo runs. 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 [26].

4.3. Analysis of the multicriteria calibration results

We begin the analysis and validation of the calibrated model instances by visualizing the solutions of the Pareto front approximations with respect to the two conflicting objectives (see Figure 6 with the two Pareto front approximations). We have selected three of the most representative solutions of the Pareto sets for the two instances: a) the solution with lowest awareness, 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 1,000 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 δ 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 $\overline{F}(s_i) = \frac{1}{1000} \sum_{j=1}^{1000} \delta \cdot w_j \cdot f_1(s_i) + (1 - w_j) \cdot f_2(s_i)$. The

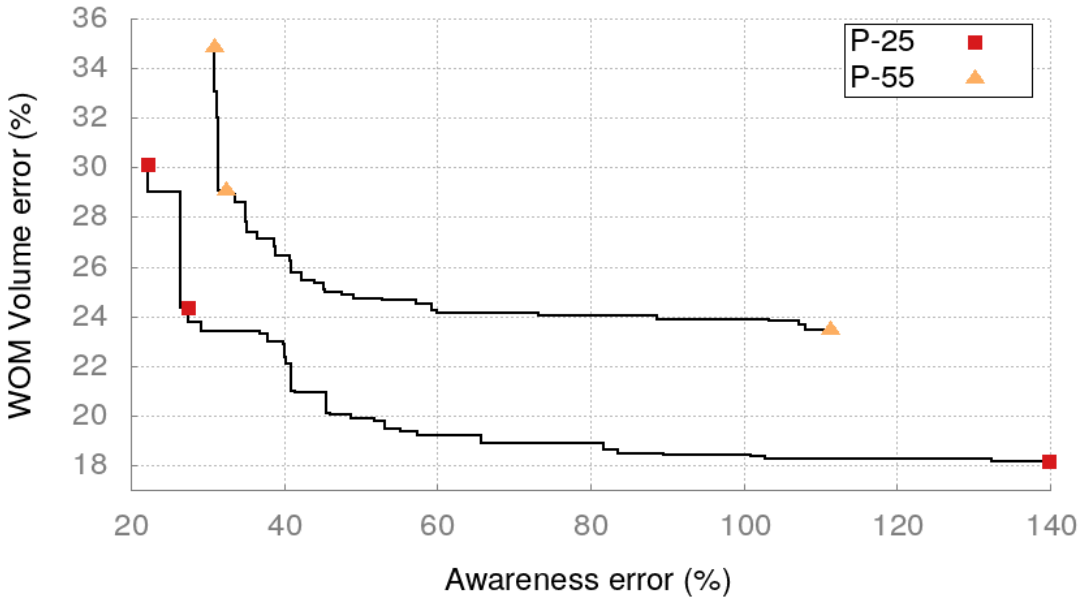


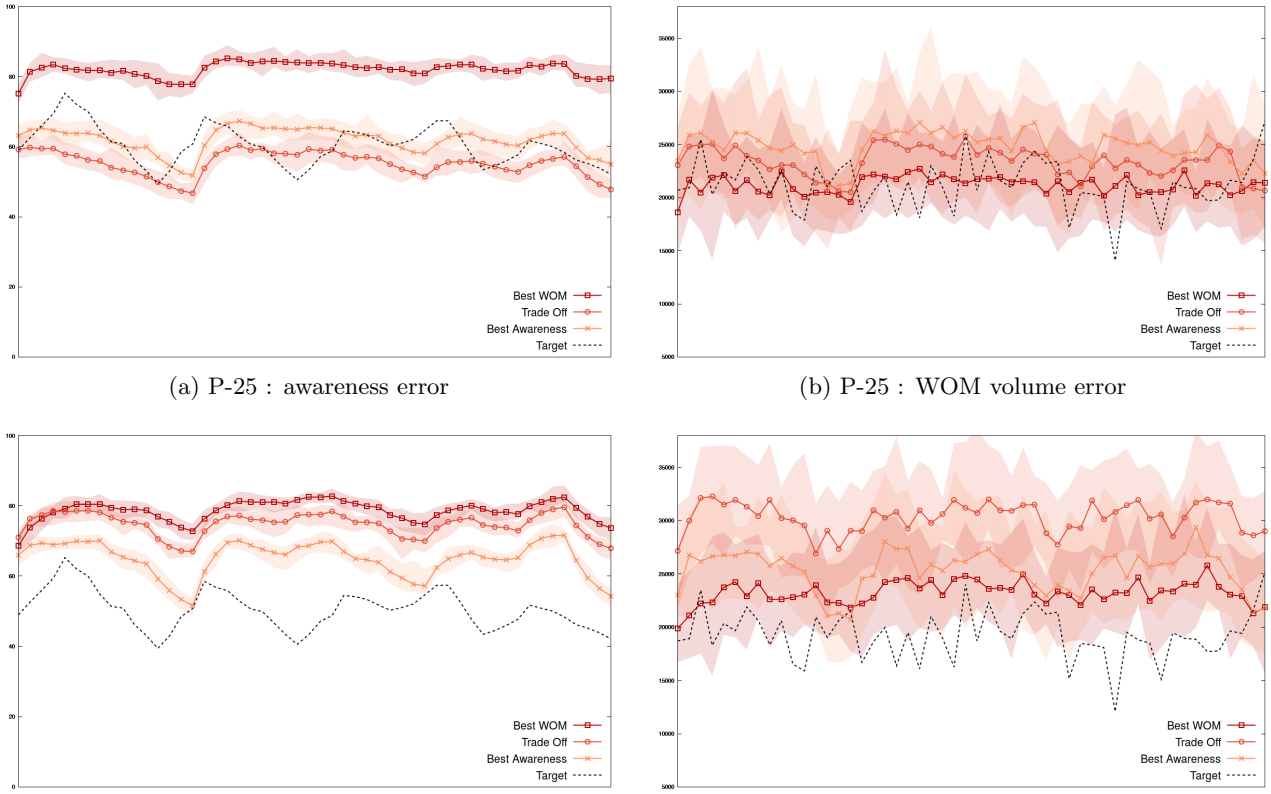
Figure 6: Selected solutions for the two model instances (P-25 and P-55): lowest awareness error, lowest WOM volume error, and best trade-off. Solutions from P-25 are represented using squares and solutions from P-55 are displayed using triangles.

selected solutions using this strategy are shown with their respective Pareto front approximations in Figure 6.

We visualize the outputs of 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. We show the behavior of the selected solutions for brands 3 and 6 in Figures 7 and 8 respectively. These brands were chosen because their behavior is a good resemblance of the rest of the brands for both objectives. These charts display the model output for the two instances regarding the two objectives. Both figures show that adjusting the behavior the dynamics of the awareness evolution over time are harder to mimic than the WOM volume dynamics. In contrast, WOM volume dynamics are more sensible to Monte-Carlo variability for both model instances (as seen in the blurred areas in Figures 7b, 7d, 8b, and 8d).

This is specially relevant for the P-25 instance, as shown in Figures 7a and 8a, 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 it can be observed in Figure 6). 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. Although trade-off and best awareness solutions achieve decent WOM volume outputs for the P-25 instance (Figures 7b and 8b), final WOM volume errors are higher for the P-55 instance even when considering the fittest solutions (as shown in Figures 7d and 8d).



(a) P-25 : awareness error

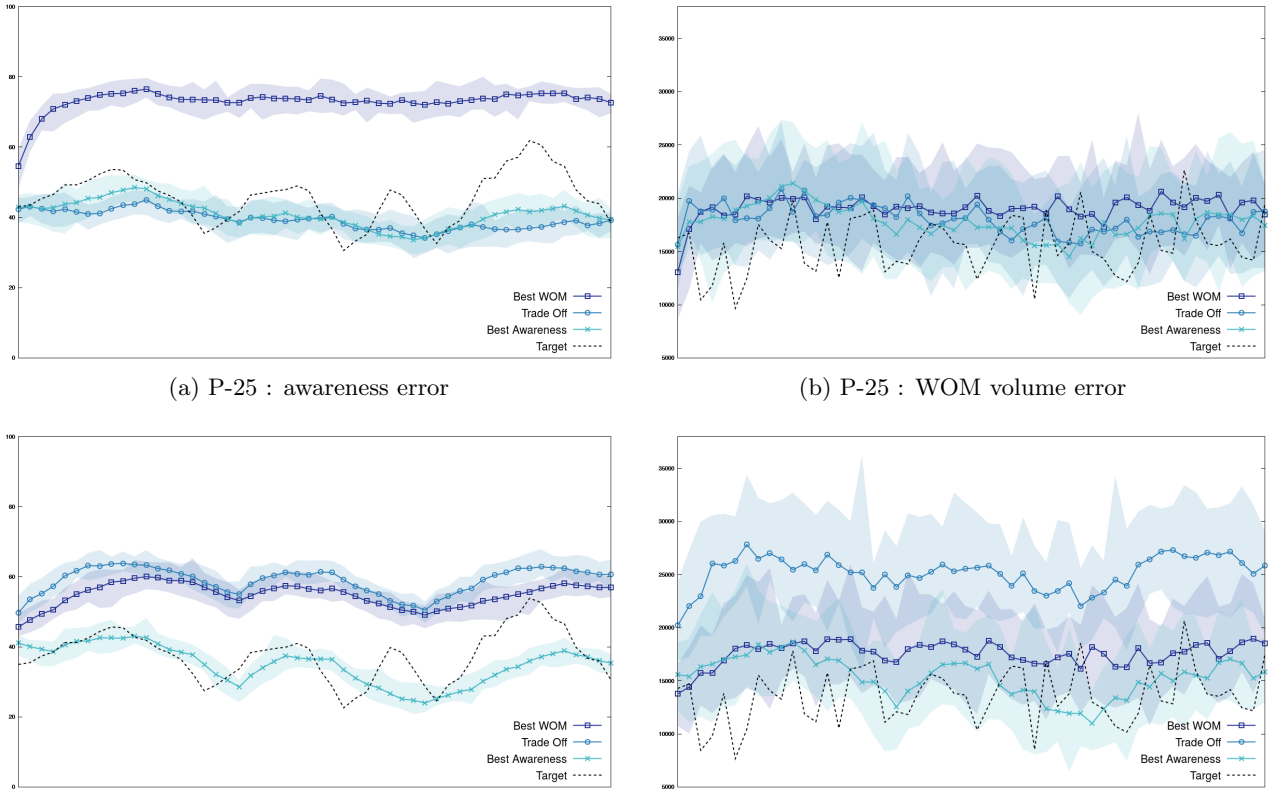
(b) P-25 : WOM volume error

(c) P-55 : awareness error

(d) P-55 : WOM volume error

Figure 7: Awareness output and WOM volume over time for P-25 and P-55 regarding brand 3. In these charts, blurred areas represent the Monte-Carlo variability and 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.

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 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 Figures 7 and 8. 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



(a) P-25 : awareness error

(b) P-25 : WOM volume error

(c) P-55 : awareness error

(d) P-55 : WOM volume error

Figure 8: Awareness output and WOM volume over time for P-25 and P-55 regarding brand 6. In these charts, blurred areas represent the Monte-Carlo variability and 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.

have higher buzz increment values (τ_{c_1} and τ_{c_4}) and higher social awareness impact (α^{WOM}). In contrast, the behavior of the selected solutions for the P-55 instance show high awareness impact parameter values for the solution with best trade-off (α_{c_4}, $\alpha_{c_{12}}$, and α^{WOM}). We can also identify that the *Best WOM* solution has a really low value for the awareness decay parameter (d). This also shows how configurations with high awareness values reduces WOM error (improves WOM adjustment) but reduces awareness fitting.

4.4. Visual and qualitative multicriteria analysis using moGrams

We continue the analysis of the calibrated model instances using moGrams. 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 [37]. Our approach is similar to the one followed during the behavior analysis: we apply moGrams to the two Pareto

P-25	τ_{c_1}	α_{c_3}	τ_{c_4}	$\mathrm{d}\tau_{c_4}$	α_{c_5}	$d\tau_{c_7}$	$p_i^b(0)$	α^{WOM}
Best Aw.	0.241	0.798	0.012	0.154	0.674	0.801	0.329	0.288
Trade-Off	0.584	0.795	0.015	0.399	0.049	0.423	0.338	0.268
Best WOM	0.246	0.537	0.676	0.059	0.577	0.800	0.222	0.955
P-55	τ_{c_1}	$d\tau_{c_2}$	α_{c_4}	$d\tau_{c_4}$	$\alpha_{c_{12}}$	$\tau_{c_{16}}$	α^{WOM}	d
Best Aw.	0.392	0.170	0.278	0.196	0.182	0.859	0.150	0.192
Trade-Off	0.866	0.583	0.618	0.775	0.457	0.593	0.280	0.171
Best WOM	0.488	0.104	0.222	0.262	0.323	0.610	0.159	0.085

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.

fronts 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 Equation 5.

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

The generated moGram for the P-25 model instance is shown in Figure 9 and its associated Pareto front approximation is displayed in Figure 10. We can see that, given the relatively high cardinality of the Pareto front approximation (29 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., 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 and 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 for making their relation clearer (see Figures 9 and 10). We provide several observations from the moGrams visualization:

- Regarding the structure of the network, we can identify four clusters of solutions (i.e., groups of solutions) in the design space. Two of these subsets of solutions, located in the left side

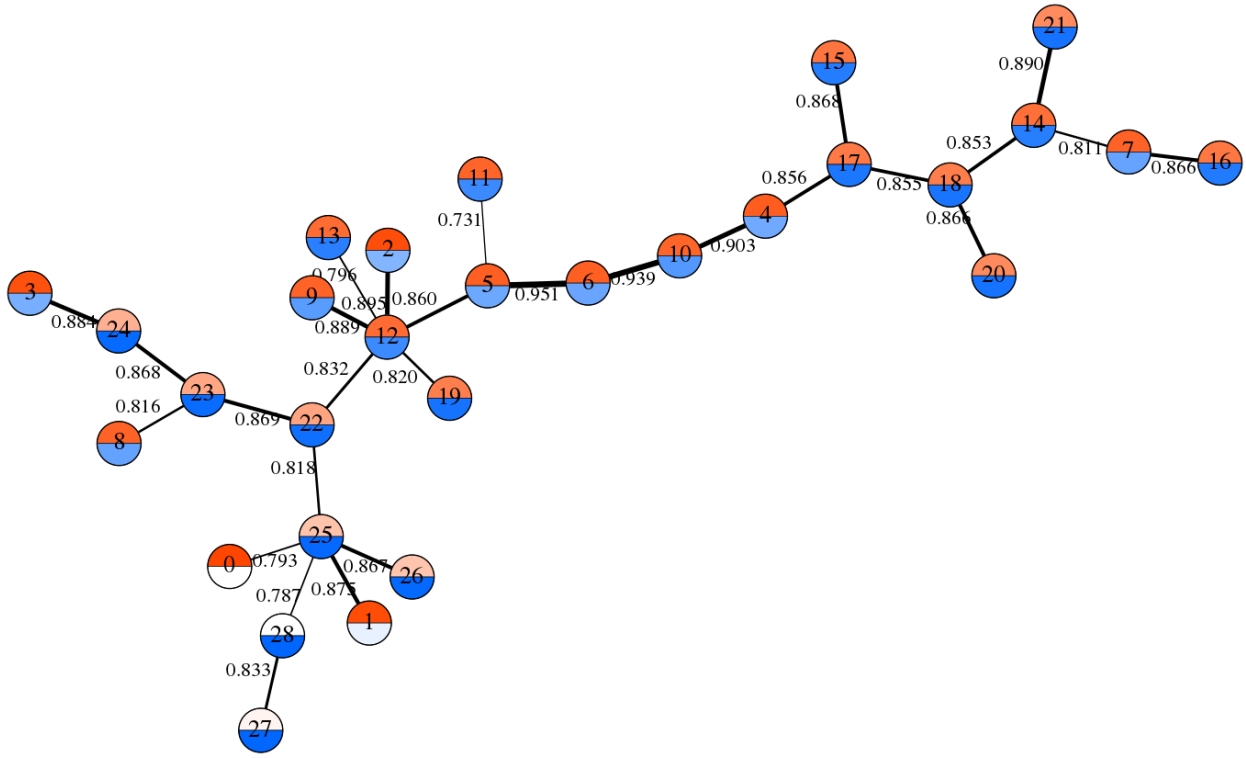


Figure 9: moGrams network representing the non-dominated calibration solutions for P-25 model instance.

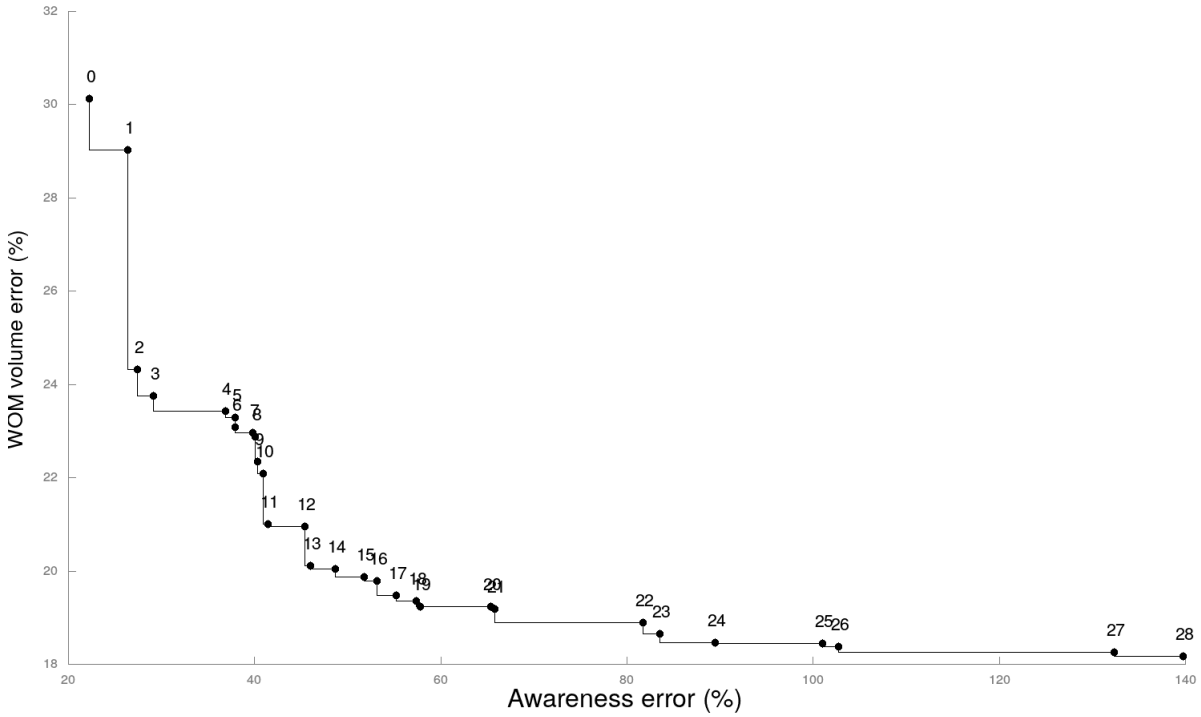


Figure 10: Pareto front approximation for P-25 model instance associated to the moGram of Figure 9.

of the network, are connected to the central cluster by Solution 22. In addition, Solution 4 is connected to another subset of solutions, located in the right side of the network.

- From those clusters we can identify Solution 12 as the most connected one since it is the only solution with degree 6. Due to its connectivity and the additional information provided by moGrams, Solution 12 could be an interesting configuration for the modeler. In this regard, many of the connections of Solution 12 have similarity values beyond 0.86, which suggests that this solution has good flexibility and can be swap by other solutions with minimum parameter changes. In addition, the flexibility of Solution 12 is specially interesting since its connections belong to different regions of the Pareto front approximation.

- The moGrams visualization methodology assists us in validating the best trade-off solution (Solution 6, located in the center 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 10 with elevated similarity values (beyond 0.93). However, these solutions are located in the same region of the Pareto front approximation, reducing the interest of swapping Solution 6 with any of its neighbors.

- With respect to the best solutions for each objective (Solutions 0 and 28), both of them are located in the same cluster of solutions, which suggests that there could be some highly sensible parameters that can drastically change the behavior of the model. In this regard, we can observe that all the solutions of this cluster come from extreme regions of the Pareto front approximation. Additionally, both of these solutions (0 and 28) are connected to Solution 25, which is the hub of its cluster, but their similarities are not specially high (~ 0.79).

Figure 11 shows the generated moGram for the P-55 model instance while its associated Pareto front approximation is displayed in Figure 12. Similarly to the previous moGram, its associated Pareto front approximation has a relatively high cardinality (also 29 solutions). Again, we can provide the following observations and interesting insights for the modeler from a validation point of view:

- Due to the topology of the network (mixed between star and tree), we can identify three subsets of solutions in the design space. Those would be the three-shaped subnetwork growing

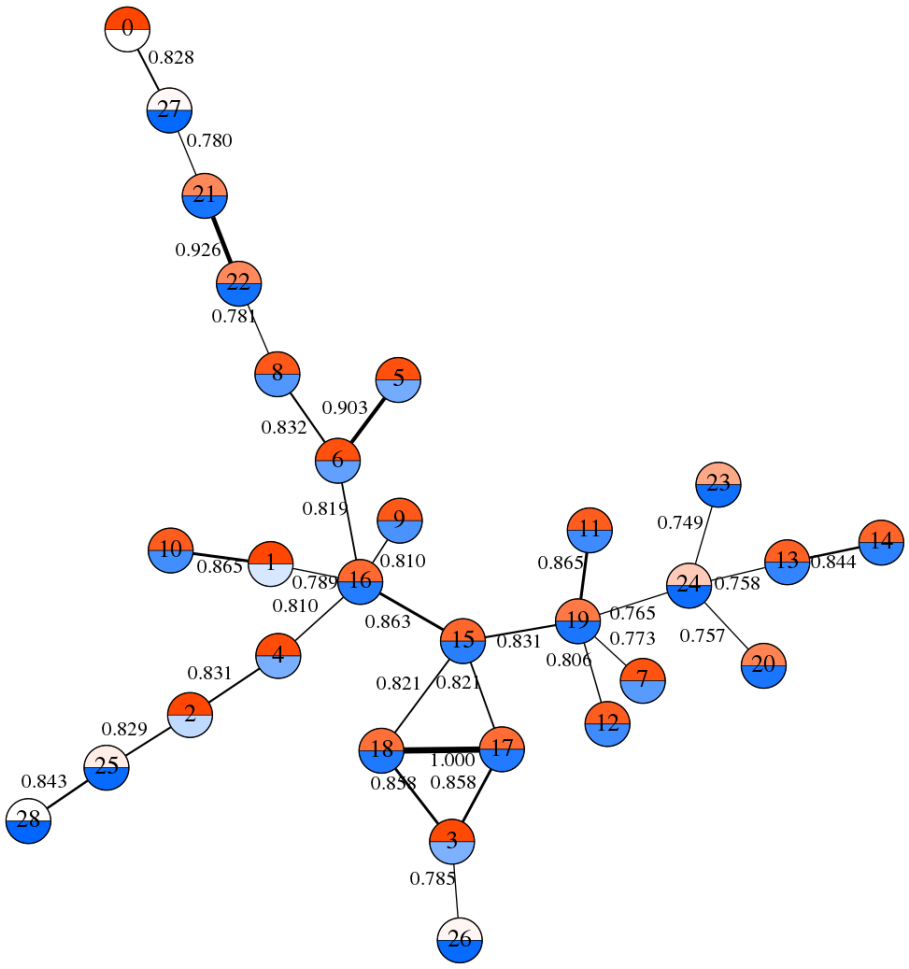


Figure 11: moGrams network representing the non-dominated calibration solutions for P-55 model instance.

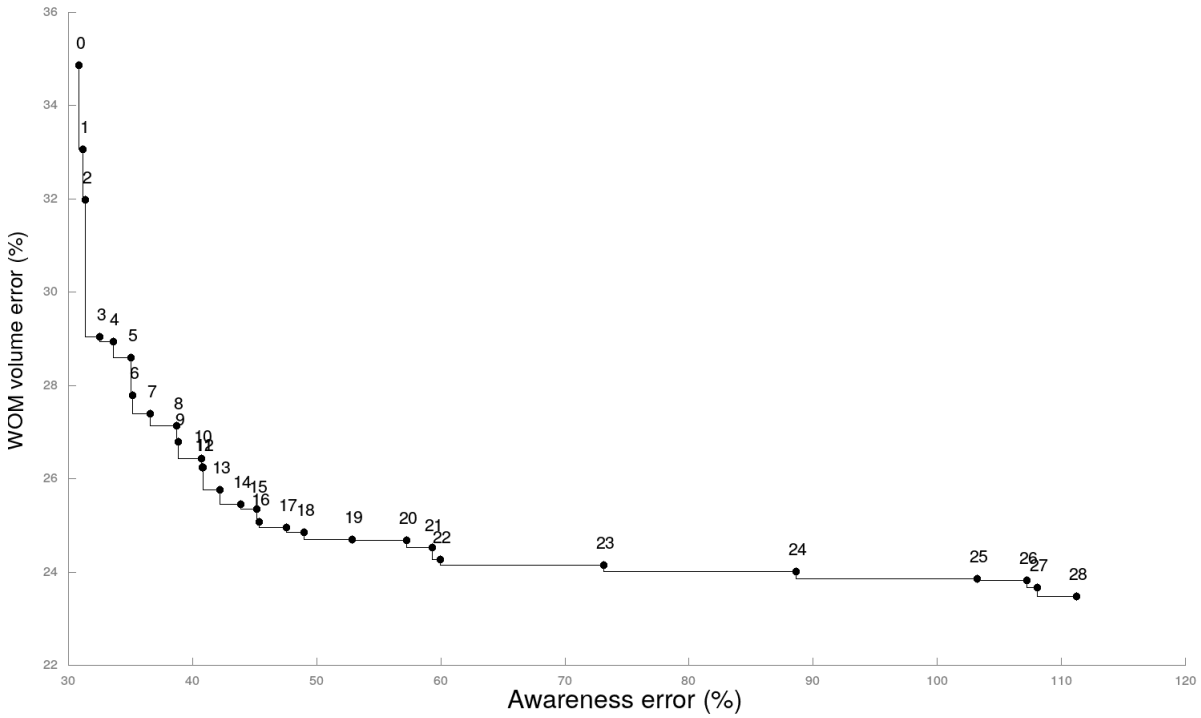


Figure 12: Pareto front approximation for P-55 model instance associated to the moGram of Figure 11.

from Solution 16, another three-shaped subnetwork growing from Solution 19, and a small subnetwork following a mesh topology that connects with the rest of the network through Solution 15.

- Solutions 15, 16, and 19 are the solutions with more connections, with either four or five neighbors. Any of these solutions could be an interesting configuration for the modeler since they have good flexibility. However, there is not enough diversity in the neighbors of Solutions 15 and 16, since most of their neighboring solutions are located in the same region of the Pareto front approximation. In contrast, Solution 19 exhibits more diversity and its neighborhood considers solutions such as 7 or 24, which belong to different regions. With respect to the similarity values of the most connected solutions, due to its higher variety of neighbors, Solution 19 has lower similarity values than Solutions 15 and 16.

- The trade-off solution (Solution 4) provides more interesting neighboring relations to the modeler. It is connected to Solution 2 and it is contained in the neighborhood of the hub defined by Solution 16. Although both of its similarity values are not specially high (0.81

	α_{c_7}	$d\tau_{c_3}$	τ_{c_8}	$d\tau_{c_8}$	$d\tau_{c_{12}}$	$\tau_{c_{14}}$	$\tau_{c_{15}}$	d
Solution 0	0.894	0.488	0.454	0.586	0.918	0.066	0.411	0.192
Solution 4	0.890	0.192	0.216	0.482	0.223	0.511	0.867	0.171
Solution 27	0.564	0.145	0.077	0.181	0.187	0.447	0.754	0.080
Solution 28	0.496	0.907	0.460	0.907	0.299	0.447	0.869	0.085

Table 4: Sample of the parameter values for the identified solutions for P-55 using its moGram at Figure 11.

and 0.83), these solutions are located in different and clearly separated regions of the Pareto front approximation, making these relationships relevant to the modeler.

- Finally, the best solutions for each objective are located in two separated branches of the subset defined by Solution 16. Both of these solutions have a single connection with similar similarity values ~ 0.83. On the one hand, Solution 28 is connected to Solution 25, which belongs to the same region of the Pareto front approximation. On the other hand, Solution 0 is linked to Solution 27, which is located in the other extreme of the Pareto front. This connection is interesting 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 27, 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 those are the ones with bigger differences between the configurations of Solution 0 and Solution 27. It can be observed that most of these parameters are related with buzz increment and buzz decay. Solution 0 has lower increment values and higher decay values of individual mass media channels, suggesting that these parameters are really important to control the behavior of the model. In addition, we can identify that the decay (d) value for both Solution 0 and Solution 4 is higher than the values of Solutions 27 and 28, which are located in the opposite region of the Pareto front approximation.

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. 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 and we have applied it to two different model instances, based on historical real data from a banking model in Spain. We have analyzed the resulting Pareto front approximations of the calibrated instances 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 acceptable 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 analyzed 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. Our analysis 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 with this high cardinality (29 solutions) 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 19 from Figure 11 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 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 [36].

Acknowledgments

This work is supported by Spanish Ministerio de Economía y Competitividad under the NEWSOCO project (ref. TIN2015-67661-P), including European Regional Development Funds (ERDF).

M. Chica is also supported through the Ramón y Cajal program (RYC-2016-19800).

References

[1] Atamturktur, S., Liu, Z., Cogan, S., Juang, H., 2015. Calibration of imprecise and inaccurate numerical models considering fidelity and robustness: a multi-objective optimization-based approach. Structural and Multidisciplinary Optimization 51, 659–671.

[2] Bader, J., Zitzler, E., 2011. Hype: An algorithm for fast hypervolume-based many-objective optimization. Evolutionary Computation 19, 45–76.

[3] Barabási, A.L., Albert, R., 1999. Emergence of scaling in random networks. Science 286, 509–512.

[4] Bennett, N.D., Croke, B.F., Guariso, G., Guillaume, J.H., Hamilton, S.H., Jakeman, A.J., Marsili-Libelli, S., Newham, L.T., Norton, J.P., Perrin, C., Pierce, S.A., Robson, B., Seppelt, R., Voinov, A.A., Fath, B.D., Andreassian, V., 2013. Characterising performance of environmental models. Environmental Modelling & Software 40, 1–20.

[5] Bonabeau, E., 2002. Agent-based modeling: Methods and techniques for simulating human systems. Proceedings of the National Academy of Sciences 99, 7280–7287.

[6] Campomanes-Álvarez, B.R., Cordón, O., Damas, S., 2013. Evolutionary multi-objective optimization for mesh simplification of 3D open models. Integrated Computer-Aided Engineering 20, 375–390.

[7] Chica, M., Barranquero, J., Kajdanowicz, T., Cordón, O., Damas, S., 2017. Multimodal optimization: an effective framework for model calibration. Information Sciences 375, 79–97.

[8] Chica, M., Óscar Cordón, Damas, S., Iglesias, V., Mingot, J., 2016. Identimod: Modeling and managing brand value using soft computing. Decision Support Systems 89, 41–55.

[9] Deb, K., 2001. Multi-objective optimization using evolutionary algorithms. volume 16. John Wiley & Sons.

[10] Deb, K., Jain, H., 2014. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints. IEEE Transactions on Evolutionary Computation 18, 577–601.

[11] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T., 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6, 182–197.

[12] Epstein, J.M., 2006. Generative social science: Studies in agent-based computational modeling. Princeton University Press.

[13] Farmer, J.D., Foley, D., 2009. The economy needs agent-based modelling. Nature 460, 685–686.

[14] Farris, P.W., Bendle, N.T., Pfeifer, P.E., Reibstein, D.J., 2010. Marketing metrics: The definitive guide to measuring marketing performance. Wharton School Publishing. 2nd edition.

[15] González-Avella, J.C., Cosenza, M.G., Klemm, K., Eguíluz, V.M., San Miguel, M., 2007. Information feedback and mass media effects in cultural dynamics. Journal of Artificial Societies and Social Simulation 10, 9.

[16] Hyndman, R.J., Koehler, A.B., 2006. Another look at measures of forecast accuracy. International Journal of Forecasting 22, 679–688.

[17] Janssen, M.A., Ostrom, E., 2006. Empirically based, agent-based models. Ecology and Society 11, 37.

[18] Lee, J.S., Filatova, T., Ligmann-Zielinska, A., Hassani-Mahmooei, B., Stonedahl, F., Lorscheid, I., Voinov, A., Polhill, J.G., Sun, Z., Parker, D.C., 2015. The complexities of agent-based modeling output analysis. Journal of Artificial Societies and Social Simulation 18, 4.

[19] Li, H., Zhang, Q., 2009. Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II. IEEE Transactions on Evolutionary Computation 13, 284–302.

[20] Libai, B., Muller, E., Peres, R., 2013. Decomposing the value of word-of-mouth seeding programs: Acceleration versus expansion. Journal of Marketing Research 50, 161–176.

[21] Macal, C.M., North, M.J., 2005. Tutorial on agent-based modeling and simulation, in: Proceedings of the 37th conference on Winter simulation, ACM. pp. 2–15.

[22] Macdonald, E.K., Sharp, B.M., 2000. Brand awareness effects on consumer decision making for a common, repeat purchase product: A replication. Journal of Business Research 48, 5–15.

[23] Miller, J.H., 1998. Active nonlinear tests (ANTs) of complex simulation models. Management Science 44, 820–830.

[24] Moya, I., Chica, M., Sáez-Lozano, J.L., Cordón, Ó., 2017. An agent-based model for understanding the influence of the 11-M terrorist attacks on the 2004 Spanish elections. Knowledge-Based Systems 123, 200–216.

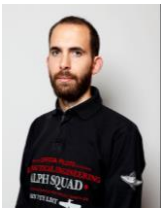
[25] Narzisi, G., Mysore, V., Mishra, B., 2006. Multi-objective evolutionary optimization of agent-based models: An application to emergency response planning, in: Proceedings of the 2nd IASTED International Conference on Computational Intelligence, CI 2006, pp. 224–230.

[26] Nebro, A.J., Durillo, J.J., Vergne, M., 2015. Redesigning the jMetal multi-objective optimization framework, in: Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation, ACM, New York, NY, USA. pp. 1093–1100.
[27] Newman, M., Barabási, A.L., Watts, D.J., 2006. The structure and dynamics of networks. Princeton University Press.
[28] Oliva, R., 2003. Model calibration as a testing strategy for system dynamics models. European Journal of Operational Research 151, 552–568.
[29] Read, M.N., Alden, K., Rose, L.M., Timmis, J., 2016. Automated multi-objective calibration of biological agent-based simulations. Journal of The Royal Society Interface 13.
[30] Samek, W., Wiegand, T., Müller, K., 2017. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. CoRR abs/1708.08296. 1708.08296.
[31] Sargent, R.G., 2005. Verification and validation of simulation models, in: Proceedings of the 37th conference on Winter simulation, pp. 130–143.
[32] Schramm, M.E., Trainor, K.J., Shanker, M., Hu, M.Y., 2010. An agent-based diffusion model with consumer and brand agents. Decision Support Systems 50, 234–242.
[33] Schvaneveldt, R.W., Durso, F.T., Dearholt, D.W., 1989. Network structures in proximity data, Academic Press. volume 24 of *Psychology of Learning and Motivation*, pp. 249–284.
[34] Stonedahl, F., Rand, W., 2014. When does simulated data match real data? Comparing model calibration functions using genetic algorithms, in: Advances in Computational Social Science. Springer, Japan. volume 11 of *Agent-Based Social Systems*, pp. 297–313.
[35] Talbi, E.G., 2009. Metaheuristics: from design to implementation. John Wiley & Sons.
[36] Thiele, L., Miettinen, K., Korhonen, P.J., Molina, J., 2009. A preference-based evolutionary algorithm for multi-objective optimization. Evolutionary computation 17, 411–436.
[37] Trawiński, K., Chica, M., Pancho, D.P., Damas, S., Cordón, O., 2018. moGrams: A network-based methodology for visualizing the set of nondominated solutions in multiobjective optimization. IEEE Transactions on Cybernetics 48, 474–485.
[38] Tušar, T., Filipič, B., 2015. Visualization of Pareto front approximations in evolutionary multiobjective optimization: A critical review and the prosection method. IEEE Transactions on Evolutionary Computation 19, 225–245.
[39] Voinov, A., Kolagani, N., McCall, M.K., Glynn, P.D., Kragt, M.E., Ostermann, F.O., Pierce, S.A., Ramu, P., 2016. Modelling with stakeholders–next generation. Environmental Modelling & Software 77, 196–220.
[40] Waldrop, M.M., 2018. Free agents. Science 360, 144–147.
[41] Walker, D.J., Everson, R., Fieldsend, J.E., 2013. Visualizing mutually nondominating solution sets in many-objective optimization. IEEE Transactions on Evolutionary Computation 17, 165–184.
[42] Watts, D.J., Strogatz, S.H., 1998. Collective dynamics of 'small-world' networks. Nature 393, 440–442.
[43] Wu, F., Huberman, B.A., 2007. Novelty and collective attention. Proceedings of the National Academy of Sciences 104, 17599–17601.
[44] Yang, J., Leskovec, J., 2010. Modeling information diffusion in implicit networks, in: 2010 IEEE International Conference on Data Mining, IEEE. pp. 599–608.
[45] Yang, S., Li, M., Liu, X., Zheng, J., 2013. A grid-based evolutionary algorithm for many-objective optimization. IEEE Transactions on Evolutionary Computation 17, 721–736.
[46] Zaffar, M.A., Kumar, R.L., Zhao, K., 2011. Diffusion dynamics of open source software: An agent-based computational economics (ACE) approach. Decision Support Systems 51, 597–608.
[47] Zaffar, M.A., Kumar, R.L., Zhao, K., 2018. Using agent-based modelling to investigate diffusion of mobile-based branchless banking services in a developing country. Decision Support Systems (in press) .
[48] Zhang, X., Tian, Y., Jin, Y., 2015. A knee point-driven evolutionary algorithm for many-objective optimization. IEEE Transactions on Evolutionary Computation 19, 761–776.
[49] Zhang, Y., Shao, Q., Taylor, J.A., 2016. A balanced calibration of water quantity and quality by multi-objective optimization for integrated water system model. Journal of Hydrology 538, 802–816.
[50] Zitzler, E., Laumanns, M., Thiele, L., 2001. SPEA2: Improving the strength Pareto evolutionary algorithm. Technical Report 103. Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH), Zurich.

*Biographical Note

Author Biographies

Ignacio Moya



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

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, 22 papers in JCR-indexed journals (16 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

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 more than 23 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 >1260 citations in Google Scholar) and 100 JCR-SCI-indexed journal papers (57 in Q1), advised 18 Ph.D. dissertations, and coordinated 32 research projects and contracts (with an overall amount of 8.7M€). By December 2018, his publications had received 4348 citations (h-index=34), being included in the 1% of most-cited researchers in the world (source: Thomson's Web of Knowledge. 12462 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. 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.