

Received March 15, 2021, accepted March 27, 2021, date of publication April 2, 2021, date of current version April 15, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3070071

# Evolutionary Multiobjective Optimization for Automatic Agent-Based Model Calibration: A Comparative Study

IGNACIO MOYA<sup>1</sup>, MANUEL CHICA<sup>1,2</sup>, AND ÓSCAR CORDÓN<sup>1</sup>, (Fellow, IEEE)

<sup>1</sup>Andalusian Research Institute DaSCI “Data Science and Computational Intelligence,” University of Granada, 18071 Granada, Spain

<sup>2</sup>School of Electrical Engineering and Computing, The University of Newcastle, Callaghan, NSW 2308, Australia

Corresponding author: Ignacio Moya (imoya@ugr.es)

This work was supported by the Spanish Agencia Estatal de Investigación, the Andalusian Government, the University of Granada, and European Regional Development Funds (ERDF) under Grants EXASOCO (PGC2018-101216-B-I00), SIMARK (P18-TP-4475), and AIMAR (A-TIC-284-UGR18). Manuel Chica was also supported by the Ramón y Cajal program (RYC-2016-19800).

**ABSTRACT** Complex problems can be analyzed by using model simulation but its use is not straight-forward since modelers must carefully calibrate and validate their models before using them. This is specially relevant for models considering multiple outputs as its calibration requires handling different criteria jointly. This can be achieved using automated calibration and evolutionary multiobjective optimization methods which are the state of the art in multiobjective optimization as they can find a set of representative Pareto solutions under these restrictions and in a single run. However, selecting the best algorithm for performing automated calibration can be overwhelming. We propose to deal with this issue by conducting an exhaustive analysis of the performance of several evolutionary multiobjective optimization algorithms when calibrating several instances of an agent-based model for marketing with multiple outputs. We analyze the calibration results using multiobjective performance indicators and attainment surfaces, including a statistical test for studying the significance of the indicator values, and benchmarking their performance with respect to a classical mathematical method. The results of our experimentation reflect that those algorithms based on decomposition perform significantly better than the remaining methods in most instances. Besides, we also identify how different properties of the problem instances (i.e., the shape of the feasible region, the shape of the Pareto front, and the increased dimensionality) erode the behavior of the algorithms to different degrees.

**INDEX TERMS** Model calibration, agent-based modeling, evolutionary multiobjective optimization.

## I. INTRODUCTION

Model simulation is a common approach to the analysis of complex phenomena. It allows users and stakeholders to recreate the desired dynamics so these phenomena could be studied in a controlled environment, where different policies or strategies can be tested. The agent-based model (ABM) methodology [1]–[3] is a well-known model simulation technique that relies in the behaviour of artificial agents, which are autonomous entities that act following simple rules and interacting with other agents. The aggregation of the agent’s behavior and the social interactions allow the modeler to simulate complex emergent dynamics using a bottom-up approach [2]. This approach has proven useful in both the

forecasting of hypothetical scenarios and the definition of what-if scenarios [4], which has increased the visibility of ABM in the latter years [5], [6].

However, the use of these models is not straight-forward as they require the modelers to deal with several issues. On one hand, ABMs must be designed following certain guidelines and methodologies for ensuring their rigor [7]. On the other hand, they require to be properly calibrated and validated before being used [8], [9]. The calibration of a model refers to the process of adjusting its parameters so it can correctly simulate the desired dynamics. This stage in model development can be carried out manually, since many parameters are usually set based on data, but it can be impracticable for models with high dimensionality.

This limitation can be overcome using automated calibration [10], a procedure that relies on two main components

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Liu.

for adjusting the model's parameters: a given error measure and an optimization method. With this approach, the error measure is minimized using the optimization method by adjusting the model's parameters so the simulated output can match the provided real data [11], [12]. Nevertheless, the success of an automated calibration process depends on the capability of its optimization method for exploring the model's parameters search space. This is specially relevant if the model being calibrated considers multiple conflicting criteria since the optimization method requires to handle these criteria jointly [13].

Multiobjective optimization methods are specially tailored for working under these restrictions [13]. In particular, evolutionary multiobjective optimization (EMO) algorithms can be considered the best approach to multiobjective optimization, as they obtain a set of representative Pareto solutions in a single run. In addition, EMO algorithms obtain Pareto-optimal solutions in a reasonable time and can perform successfully without requiring specific properties of the optimized function [14] and have proven to be successful when dealing with dynamic multiobjective optimization problems which are common in real-world applications [15]–[19]. However, there is a large number of EMO algorithms available in the specialized literature and finding the best algorithm for conducting the automated calibration process can be overwhelming.

We propose to delve into this issue by conducting an exhaustive analysis of the performance of the most prominent and recent EMO algorithms when calibrating multiple instances of an ABM jointly optimizing different key performance indicators. The study considers well-known EMO algorithms from the main EMO categories: based on Pareto dominance, indicators, and decomposition. The selected methods include the non-dominated sorting genetic algorithm II (NSGA-II) [20], the improved strength Pareto evolutionary algorithm (SPEA2) [21], the general indicator-based evolutionary algorithm (IBEA) [22], the  $S$  metric selection multiobjective optimization algorithm (SMS-EMOA) [23], and the multiobjective evolutionary algorithm based on decomposition (MOEA/D) [24], [25]. Additionally, we also make use of two recent EMO developments that have shown competitive results when dealing with real problems [26]–[28]. Specifically, we incorporate to the study the many-objective metaheuristic based on the R2 indicator II (MOMB2) [29] and the global weighting achievement scalarizing function genetic algorithm (GWASF-GA) [30]. Finally, we included a classical optimization method such as the Nelder-Mead's simplex method [31] as the baseline for the EMO algorithms, that we adapted to our multiobjective problem by using the adaptive  $\epsilon$ -constraint method, which is a common approach [32], [33]. We analyze the results of the selected algorithms using multiobjective performance indicators and attainment surfaces [34]. In addition, we perform a statistical test for studying the significance of the performance indicator values.

Our study considers a benchmark of 15 instances of an ABM for marketing, which is the selected computational

model for our experiments. This ABM tackles marketing scenarios involving two conflicting outputs or key performance indicators: the global awareness of the consumers regarding the brands available in the market and the number of word-of-mouth consumer interactions for those brands. Both the instances and historical data for our calibration benchmark are taken from a real banking marketing scenario in Spain. Although there is previous work using EMO for multicriteria calibration of ABMs [35]–[37], none of these contributions considers a rigorous and exhaustive comparison of several EMO algorithms for calibrating multiple model instances and the subsequent analysis of the algorithms' performance according to the problem characteristics.

Hence, the main contributions of the current manuscript are:

- An exhaustive analysis of the performance of relevant EMO algorithms when calibrating multiple instances of an ABM for marketing considering different outputs.
- The design of an appropriate experimental setup for the study, which is based on a benchmark comprising 15 instances considering two key performance indicators and up to 175 decision variables.
- A comprehensive analysis of the results which considers both unary and binary performance indicators, attainment surfaces, statistical significance/tests, and two discussion sections addressing both the influence of the instances' properties on the algorithms' performance and the drawbacks of the methods.

The structure of this paper is as follows. Section II addresses the related work on the use of EMO algorithms for multicriteria calibration of computational models. Then, our approach to the multicriteria calibration of ABMs using EMO algorithms is depicted in Section III, which reviews several concepts of EMO-based model calibration and the algorithms selected for our study. The ABM to be calibrated in our experiments is described in Section IV. The analysis of the results is thoroughly reviewed in Section V. Finally, Section VI discusses our final remarks and Section VII reviews the practical implications and future directions of our work.

## II. RELATED WORK

There are some examples of the use of EMO algorithms for multicriteria calibration of computational models [38]–[44]. Many of them are focused in the calibration of hydrological models, such as the soil and water assessment tool [45]–[48], the rainfall–runoff models [41], empirical hydrological models for streamflow forecasting [39], and an integrated water system model [44]. The thorough review of these contributions reveals that their usual approach relies on employing the NSGA-II for running the calibration process, probably because it is the most popular EMO algorithm. Apart from NSGA-II, we can also find some studies using SPEA2 [39], [40], [47], [48].

The application of EMO for multicriteria calibration of ABMs is not frequent although there are few examples

tackling this issue [35]–[37]. Farhadi *et al.* [35] present a framework for sustainable groundwater management including a Nash bargaining model, which is implemented as an ABM incorporating cooperative and non-cooperative agents that consume the water of the modeled scenario at different ratios. The parameters of the model are calibrated using NSGA-II and considering three objectives and a single calibration scenario. Narzisi *et al.* [36] deal with the calibration of an ABM for emergency response planning using NSGA-II. Their model is calibrated for minimizing the percentage of fatalities and the average waiting time of the population before receiving attention at the hospitals. This calibration process is applied to a single scenario considering ten real-coded model parameters with several restrictions. Finally, Read *et al.* [37] introduce the calibration of artificial murine multiple sclerosis simulation, an established immunological ABM for computational biology. They use NSGA-II in the calibration of 16 integer and real parameters with respect to four objectives. The authors consider a single scenario for the model and run three independent calibration executions for its analysis.

Therefore, NSGA-II is the recurrent EMO algorithm for multicriteria calibration of ABMs and there is not any comparative study on different EMO algorithms for this problem. In addition, it can also be recognized that the methodology of these contributions generally limit their experimentation to a single run of the EMO algorithm, which can be explained by the high computational cost of simulating multiple times for every evaluation of a single model configuration. However, this approach is not taking into account that EMO algorithms are stochastic, thus requiring multiple runs using different seeds. Analyzing the results of a single algorithmic execution reduces the amount of information provided by the calibration process because valuable model configurations may be skipped in the initial run, specially if the EMO algorithm is not properly tuned.

### III. MULTICRITERIA CALIBRATION OF ABMs USING EMO ALGORITHMS

This section describes the key features of using EMO algorithms for multicriteria calibration of ABMs. First, Section III-A reviews some basics on multiobjective optimization and Section III-B presents the common design of the EMO algorithms. Then, Section III-C includes several subsections for introducing the selected EMO algorithms according to their category, depending on their operation mode.

#### A. MULTI-OBJECTIVE OPTIMIZATION

The multicriteria calibration of ABMs can be approached as a multiobjective optimization problem since there is usually a need for calibrating the model according to different outputs. In these kinds of problems, the quality of a setting is evaluated regarding multiple conflicting criteria instead of considering a single error measure. Thus, the optimization algorithm aims to minimize  $F(x) = f_1(x), \dots, f_m(x)$ , where  $m$  represents the

number of objectives and  $x$  is the set of decision variables for the optimization problem (i.e., decision space). Each function  $f_i$  computes the quality of the parameter setting to a calibration objective using a deviation error measure  $\epsilon_i$ . In a calibration problem, each objective is associated to one simulated output  $o_i$ , resulting in  $f_i(x) = \epsilon_i(o_i(x), \tilde{o}_i)$ , with  $\tilde{o}_i$  being the historical target values for the  $i$ -th output. Any of the well-known deviation measures such as MAPE, RMSE, or MARE [49] can be chosen for computing this deviation error.

Multicriteria model configurations thus need to be analyzed using multiobjective semantics like the Pareto dominance concept [13]. Given two feasible configurations  $u$  and  $v$  from the decision space with  $u \neq v$ ,  $u$  dominates  $v$  if  $u_i \leq v_i, \forall i : 1 \leq i \leq m$  and  $\exists j : 1 \leq j \leq m : u_j < v_j$ , i.e., if  $u$  is equal or better than  $v$  for every objective and strictly better for at least one objective. However, these inequalities should be reversed for any objective that is being maximized (to dominate means to be better). Using the dominance concept, the global Pareto-optimal configurations are those vectors  $u$  such that there is no feasible vector  $v$  that dominates  $u$ . A set of  $u$  configurations where there is no  $v$  that dominates any of the other solutions is called a Pareto-optimal set. In addition, the representation of the solutions in the Pareto set as points from the objective space is called a Pareto-optimal front [13].

#### B. COMMON DESIGN

Before describing the selected EMO algorithms, we introduce their common characteristics. Each candidate solution has  $n$  decision variables corresponding to the model parameters being calibrated, which can either be integer-coded or real-coded values. The considered algorithms include polynomial mutation [14] as their mutation strategy. It modifies the values of a solution's variables with a probability  $p_m \in [0, 1]$  using a polynomial distribution. This mutation strategy uses a distribution index parameter that regulates the strength of the mutation. Unless stated otherwise, the proposed algorithms use simulated binary crossover (SBX) [14] with a crossover probability  $p_c \in [0, 1]$  as their crossover strategy. SBX emulates the operation of a single-point crossover from binary-encoding when performing crossover into real-coding decision variables. SBX operates 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  $c_{1i} = \bar{X} - \beta/2 \cdot (p_{2i} - p_{1i})$  and  $c_{2i} = \bar{X} + \beta/2 \cdot (p_{2i} - p_{1i})$ , where  $\bar{X} = 1/2 \cdot (p_{1i} + p_{2i})$ .  $\beta$  is a random value fetched from a random distribution initialized by setting a distribution index that acts as the spread factor of the operation.

#### C. CONSIDERED EMO ALGORITHMS

##### 1) ALGORITHMS BASED ON PARETO DOMINANCE

Pareto dominance-based algorithms assign the quality of the solutions (thus guiding the selection mechanism) according to their dominance of other solutions in the population.

The selected Pareto dominance-based algorithms are NSGA-II and SPEA2.

- NSGA-II [20] can be identified as the most popular and well-known EMO algorithm. NSGA-II's approach relies on non-dominated sorting, which allows it to combine elitism with good levels of diversity in a single population while being computationally fast, specially for problems with two or three objectives. NSGA-II produces an offspring set  $Q_t$  at each generation using the solutions of the previous set  $P_t$ . Then, both sets are merged into the temporary set  $R_t$  where previous and newly generated solutions are ranked according to its non-dominance level. The non-dominance level of a solution corresponds with the number of solutions that dominate it. The next set  $P_{t+1}$  is generated by selecting the solutions with the best ranking, which are the solutions not dominated by other solutions in the previous set. This process is iterated for the next ranks until a population size  $|P|$  is reached. This strategy guides the algorithm to non-dominated regions while a set of non-dominated solutions are maintained in the population. The first solution set that does not fit  $P_{t+1}$  is filtered using a crowding mechanism for boosting the diversity of the new population.
- SPEA2 [21] is a well-known EMO algorithm that computes the fitness of its individuals calculating a "strength" value that represents how many solutions it dominates. Then, the fitness value for each solution is computed by summing the "strength" values of the solutions that dominate it. SPEA2 considers a separate population, named the "archive" ( $\bar{P}_t$ ), designed to store non-dominated solutions. At each step, non-dominated solutions in  $P_t$  and  $\bar{P}_t$  are copied to  $\bar{P}_{t+1}$ . If  $\bar{P}_{t+1}$  exceeds the size of  $P$ , then its solutions are filtered using a truncation operator inspired in the  $k$ -th nearest neighbor method that selects the solutions with the minimum distance. If there are not enough solutions for filling  $\bar{P}_{t+1}$  then the dominated solutions with the minimum fitness are included until  $|\bar{P}_{t+1}| = P$ . Then a mating pool is set using binary tournament on  $\bar{P}_{t+1}$ . Finally  $P_{t+1}$  is the result of applying the crossover and mutation operators to the mating pool.

## 2) ALGORITHMS BASED ON INDICATORS

Indicator-based algorithms assign the fitness of the solutions using indicator values. The selected indicator-based EMO algorithms are IBEA, SMS-EMOA, and MOMBI2.

- IBEA [22] is a classic EMO algorithm that qualifies solutions regarding their relative contribution to a given performance indicator with respect to the rest of solutions of the population. Therefore, IBEA computes the loss of quality of removing a solution from the population using dominance preserving binary indicators. In order to carry out this task, some suitable indicators would be the additive  $I_\epsilon$  or the  $I_{HD}$  indicator, that is

based on the concept of hypervolume [50]. Using these concepts, IBEA's fitness evaluation for solution  $x$  using a binary indicator  $I$  and a scaling parameter  $\kappa$  is computed as  $F(x) = \sum_{y \in P \setminus x} -\exp[-I(y, x)/\kappa]$ . Finally, IBEA performs elitism and only the worst solutions of the population are removed, although this implies that the fitness of the remaining solutions need to be updated each time a solution is removed from the population.

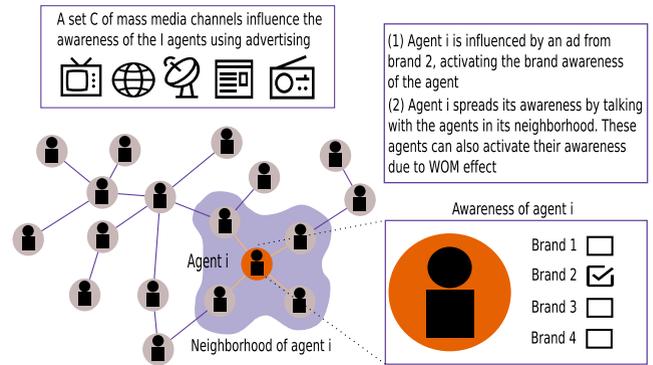
- SMS-EMOA [23] introduces the maximization of the dominated hypervolume into the search process for approximating the true Pareto front. SMS-EMOA borrows NSGA-II's non-dominated sorting mechanism for merging the current population  $P_t$  with the offspring population  $Q_t$  into  $P_{t+1}$ . However, SMS-EMOA considers a replacement strategy that targets the solutions from the worst front with the lesser contribution to the hypervolume of their respective front. This process maximizes the quality of the population regarding their hypervolume [50]. In addition, as the repeated calculation of hypervolume values is computationally expensive, SMS-EMOA follows a steady-state scheme for easing the replacement mechanism and allowing an easy parallelization of the fitness evaluation. Unlike other EMO algorithms like SPEA2, SMS-EMOA does not consider a separate archive for storing non-dominated solutions. Instead, it maintains a population of constant size that includes dominated and non-dominated solutions (as NSGA-II does). SMS-EMOA also preserves the extreme solutions (i.e., the ones with best fitness for one objective and worst fitness for the other) into the population for biobjective problems such as our ABM calibration problem instead of requiring a reference point for computing hypervolume. For problems with more objectives, a reference point is calculated dynamically at each generation.
- MOMBI2 [29] relies in the R2 quality indicator for ranking the solutions, a Pareto compliant indicator with a reduced computational cost. This quality indicator uses a utility function for mapping each objective into a single value. A common MOMBI2 configuration employs the achievement scalarizing function (also used by GWASF-GA) since it allows the algorithm to obtain weekly Pareto optimal solutions, although there are several candidate utility functions for the algorithm. In addition, instead of updating the nadir point at each generation, MOMBI2 updates this reference point taking into account its historic values during previous generations. This update takes two parameters  $\alpha$  and  $\epsilon$  as the threshold and the tolerance threshold, respectively. These historic values are used for estimating how far current solutions are from the true Pareto front: high variance suggests that the solutions are far from it and a low variance suggests that the solutions are close. The solution ranking using R2 proceeds as follows: first, the solutions with the best rank (i.e., those that optimize the weight vectors) are selected, removed from  $P_t$  and

introduced into  $P_{t+1}$ ; then, this process goes on ranking solutions until every solution has been ranked and  $|P|$  solutions are selected. In case two solutions provide the same utility value, the solution with the lowest Euclidean distance is selected.

### 3) ALGORITHMS BASED ON DECOMPOSITION

Decomposition-based algorithms transform a given multi-objective problem into several subproblems. The selected decomposition-based EMO algorithms are MOEA/D and GWASF-GA.

- MOEA/D [24], [25] is an evolutionary algorithm that has received great attention in the evolutionary computation literature in the last few years. It employs decomposition techniques for reducing the multiobjective problem into as many subproblems as individuals ( $|P|$ ). Then, MOEA/D solves every subproblem jointly by evolving its solution population ( $P_t$ ), which contains the best solution found for each subproblem. The optimization of each subproblem is performed by only using information from its neighboring subproblems. Although MOEA/D is compatible with any decomposition approach that transforms the Pareto approximation problem into several scalar optimization problems, we choose the Tchebycheff approach in this paper, as recommended by the authors [24], [25]. In addition, MOEA/D uses an external population for storing the non dominated solutions found during the execution of the algorithm, similarly to SPEA2. Finally, we select a differential evolution operator as the crossover strategy instead of the SBX employed in the other algorithms, also following authors' recommendation [24]. This operator generates each offspring  $C = (c_1, \dots, c_n)$  as  $c_i = P_1(i) + F \cdot (P_2(i) - P_3(i))$  with probability  $CR$  and  $c_i = P_1(i)$  with probability  $1 - CR$ , where  $P_1$ ,  $P_2$ , and  $P_3$  are the donor individuals and  $CR$  and  $F$  are the control parameters.
- GWASF-GA [30] is a recent aggregation-based evolutionary algorithm. GWASF-GA approximates the true Pareto front transforming the original problem into a set of scalar subproblems that are minimized using the achievement scalarizing function, based on the Tchebychev distance. This scalarizing function uses two reference points: the nadir point and the utopian point. The former is a point containing the worst objective values of the solutions of the entire Pareto-optimal set. The latter is a point that is chosen for dominating the ideal point and that will not be obtainable for any solution. During each algorithm iteration, every solution in the population is classified into different fronts by computing their achievement scalarizing function values using the two mentioned reference points and a set of weight vectors. Each of these fronts contains the solutions with the lowest scalarizing function value for the weight vectors in the set. The set of weight vectors is predefined for



**FIGURE 1. General scheme of the ABM showing an example of the effects of an ad to an agent. The agents reached by advertising can activate its awareness value for the ad brand and spread it to other agents in the social network.**

ensuring that its inverse is well distributed, ensuring that the algorithm maintains diversity. Then, the fronts with the lowest function values are introduced into the next population until  $|P|$  solutions are selected.

## IV. ABM DESCRIPTION

The current section summarizes the main characteristics of the selected ABM [51]. Section IV-A presents the general structure of the model and the mechanics of the agents. Then, Section IV-B summarizes the parameters selected for calibration.

### A. ABM GENERAL STRUCTURE

The model performs a terminating simulation during a given number of steps  $T$  of a market with a set of  $B$  competing brands, where the time-step represents a natural week. The model simulates a set of agents  $I$  and their behavior when exposed to social interactions and the advertisement of  $C$  mass media channels. It considers two outputs relevant to market expansion [52], [53]: the word-of-mouth interactions between consumers (referred as WOM volume) and the awareness of the brands. During the simulation, the  $I$  consumer agents are exposed to the information spread by mass media channels and the WOM process generated through their social network. These two processes are connected because the activation of brand awareness using advertising also increases WOM volume due to the buzz effect produced by the campaign. Additionally, WOM interactions spread the agents' brand awareness through the social network. Therefore, we can see that both outputs cannot be adjusted separately, since improving the fitting of one output decreases the fitting of the other. As a consequence, there is no configuration that jointly satisfies the fitting of both outputs. We present a general scheme of our marketing model in Figure 1.

We model agent's awareness with respect to the different brands ( $b \in B$ ) using a binary state variable  $a_i^b \in \{0, 1\}$ . If  $a_i^b(t) = 1$  at a given time step  $t \in [1, T]$ , the awareness of agent  $i \in I$  is active for brand  $b$  (i.e., it is aware of the brand). In contrast, a value of 0 represents that the awareness is not active at this time step. This variable is initialized by using a parameter called initial awareness ( $a^b(0) \in [0, 1]$ )

$m$	$\alpha_1$	$\alpha_2$	$\alpha_3$	$\tau_1$	$\tau_2$	$\tau_3$	$d\tau_1$	$d\tau_2$	$d\tau_3$	$p_i^b(0)$	$\alpha^{WOM}$	$d$
4	0.02	0.015	0.05	0.1	0.15	0.22	0.12	0.05	0.23	0.2	0.1	0.01

**FIGURE 2.** Example chromosome for a model instance with  $|C| = 3$  where the first decision variable matches the network’s generation parameter  $m$ , an integer parameter bounded to  $\{2, \dots, 8\}$ . The remaining decision variables match the real-coded parameters of the model, which are limited to  $[0, 1]$ .

that represents the percentage of the agent population whose awareness is active at the start of the simulation for each brand ( $a^b(0) = 1/|I| \cdot \sum_{i=1}^{|I|} a_i^b(0)$ ).

Agents’ awareness values are dynamic because the awareness of any brand can be activated or deactivated at every simulation step. On the one hand, brand awareness can be activated by WOM interactions within the social network or through brand advertising. On the other hand, agents’ brand awareness can be deactivated [54], [55] if it is not reinforced during the simulation. This activation/deactivation effect is modeled with additional parameters. First, the awareness deactivation parameter ( $d \in [0, 1]$ ) models the probability of an agent deactivating its awareness of a given brand. This parameter takes effect during the start of each step  $t$ , when every agent  $i$  checks each brand  $b$  that it is aware of ( $a_i^b(t) = 1$ ). Then, each brand awareness value is deactivated with probability  $d$  by switching  $a_i^b(t)$  to 0.

The agents in  $I$  compose an artificial scale-free social network [56] where their awareness values flow due to their social interactions. This effect is modeled as a contagion process [57] by the probability of talking of the agents ( $p(t)_i^b \in [0, 1]$ ). Thus, each agent  $i$  can spread their brand awareness for every brand  $b \in B$  where  $a_i^b = 1$  at time step  $t$ . The WOM awareness impact parameter ( $\alpha^{WOM} \in [0, 1]$ ) regulates the probability for each neighboring agent to activate its awareness of a brand after one of its neighbors talked about it. Additionally, the scale-free network is generated by Barabasi-Albert’s preferential attachment algorithm [56], which considers a main parameter  $m$  that regulates the network’s growth rate and its final density. Using  $m$ , the average degree of the social network can be computed as  $\langle k \rangle = 2 \cdot m$ . Finally, variable  $\omega_i^b(t)$  stores the number of new conversations of the agent  $i$  about brand  $b$  in the WOM process.

The mass media channels in  $C$  influence agents randomly depending of the capability of the channel to reach high percentages of the population and the amount of investment employed. The reach parameter ( $r_c \in [0, 1], \forall c \in C$ ) regulates the maximum percentage of the population that can be reached by channel  $c$  during a single step. We can schedule the resulting media impressions over the agent population using the given investment by assigning them at random between the agents without violating the reach constraints for the media channel [51], [58].

All the channels  $c \in C$  consider an awareness impact parameter ( $\alpha_c \in [0, 1]$ ) that regulates the likelihood for an impacted agent to activate its awareness of the announcing brand after a single media impact. In addition, the advertising scheduled in mass media may generate a viral buzz effect [58], increasing the talking probability ( $p_i^b$ )

of the reached agent for the announcing brand. This effect is modeled by the buzz increment parameter ( $\tau_c$ ), which increases the current talking probability by a percentage of its initial value ( $p_i^b(0)$ ). Nevertheless, the buzz created by advertising is dynamic and decays over time by buzz decay parameter ( $d\tau_c$ ) if it is not reinforced.

**B. PARAMETERS SELECTED FOR CALIBRATION AND OBJECTIVE FITTING FUNCTIONS**

The parameters selected for automated calibration are those involved in modifying the agents’ volume of conversations and their awareness values as those are the most uncertain and the hardest to set manually. From the parameters controlling the WOM behavior, we calibrate the  $m$  parameter for the network’s generation, the initial talking probability ( $p_i^b(0)$ ), the WOM’s awareness impact ( $\alpha^{WOM}$ ), and the awareness deactivation probability ( $d$ ). From the parameters which control the behavior of the mass media channels, we include the awareness impact ( $\alpha_c$ ), buzz increment ( $\tau_c$ ), and buzz decay ( $d\tau_c$ ).

Each parameter is represented by a real-coded decision variable in  $[0, 1]$ , with the exception of  $m$  that is limited to  $\{2, \dots, 8\}$  because it requires integer values. The overall number of parameters to be calibrated depends on the number of channels  $|C|$  in the market. Hence, the final dimensionality of each instance is  $|C| \cdot 3 + 4$ . Figure 2 shows an example of a chromosome encoding a candidate solution for a market instance with  $|C| = 3$ .

The objective fitting functions for our model are defined by Equations 1 and 2. These functions are a specification of the generic multicriteria calibration approach given in Section III-A:  $f_1$  computes the awareness deviation error and  $f_2$  computes the error for the number of conversations. Both of these fitting functions are similar to a standard MAPE function. The series of target data are represented by  $\tilde{a}$  and  $\tilde{\omega}$ , with the former being the target awareness values and the latter being the target WOM volume values. The simulated outputs are the result of running multiple Monte-Carlo simulations for each set of calibrated model’s parameters encoded in a chromosome and averaging the values of these independent runs. As introduced in Section IV-A, both model’s outputs are in conflict and should not be adjusted separately since improving the fitting of one output decreases the fitting of the other [51].

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

$$f_2 = \frac{100}{T \cdot |B|} \sum_{t=1}^T \sum_{b \in B} \left| \frac{\omega^b(t) - \tilde{\omega}^b(t)}{\tilde{\omega}^b(t)} \right| \tag{2}$$

## V. EXPERIMENTATION

Section V-A explains the setup of our experiments, which includes the description of the different benchmark instances and algorithms' configuration. Section V-B discusses the calibration results using multiobjective performance indicators, attainment surfaces, and statistical tests. Finally, Section V-C reviews the drawbacks observed in the performance of the EMO algorithms and Section V-D analyzes the influence of the instances' properties on the behavior of the algorithms.

### A. SETUP

We consider a benchmark with 15 instances of the model corresponding to different market configurations with a variable number of mass media channels. These instances are the result of synthetically generating 14 additional instances from an initial real-world, baseline instance, referred as P1(25) [51]. Notice that, this original instance corresponds to a market with 7 channels, thus resulting in 25 parameters to be calibrated, the number enclosed in brackets. The additional instances are generated applying variations on the initial baseline instance. Each model variation incorporates new mass media channels that are generated from the existing ones by perturbing their investment values. The new instances also include modifications on the target data for the fitting of both outputs (i.e., WOM values and awareness values). Each new instance increases the dimensionality of the previous one as the parameters of the new channels are added as new decision variables, enabling a deeper analysis of the algorithms' performance.

On the one hand, the perturbations on the existing mass media channels  $C$  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 whether 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 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 awareness values to a maximum of 100% and a minimum of 1%. Each addition/subtraction on the awareness target values will be by 2%, 5%, 8%, or 10%. In the case of WOM volume each addition/subtraction will be by 1,000, 2,000, 4,000, or 6,000 conversations, with 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.

The new generated instances are labeled according to their dimensionality: P2(40), P3(46), P4(55), P5(61), P6(70), P7(76), P8(85), P9(91), P10(100), P11(115), P12(130), P13(145), P14(160), and P15(175). The parameter configuration of baseline P1(25) considers  $|I| = 1000$ ,  $|B| = 8$ ,  $|C| = 7$ , and  $T = 52$ . Awareness is initialized to  $a(0) = (0.7, 0.75, 0.58, 0.25, 0.08, 0.42, 0.39, 0.34)$  and mass media

TABLE 1. Parameters settings for the EMO algorithms.

Parameter	Value
<b>Shared by all the EMO</b>	
Population	100
Maximum evaluations	10,000
Mutation probability	$1/n$
Mutation distribution index	10
SBX $p_c$	1
SBX distribution index	5
<b>MOEA/D</b>	
Crossover rate	0.5
F	0.5
Neighborhood size	20
<b>MOMBI2</b>	
$\alpha$	0.5
$\epsilon$	0.001

channels consider  $r = (0.92, 0.57, 0.54, 0.03, 0.43, 0.38, 0.69)$ . The generated instances share this baseline configuration, including the reach parameters values  $r_c$  of the new channels, that take the value of the channel used for its generation. For example, if a new channel 9 is generated from the original channel 5, the reach value of the former channel is set to the value of the latter (i.e.,  $r_9 = r_5$ ).

Each EMO algorithm is run 30 times using different seeds for each run to account for the probabilistic nature of the calibration algorithms considered. Every algorithm considers a population of 100 individuals ( $P = 100$ ) and evolves for 100 generations with a stopping criteria of 10,000 evaluations. Due to the highly time-consuming task of simulating multiple times for every parameter configuration, each evaluation of a candidate solution involves 15 Monte-Carlo runs. The distribution index of the mutation operator is set to 10 and the mutation probability value is set to  $p_m = 1/n$  where  $n$  is the number of parameters being calibrated for the model instance (i.e., decision variables). The SBX crossover operator considers a crossover probability of  $p_c = 1.0$  and sets its distribution index value to 5. In addition, the EMO algorithms designed to use a set of weights, such as MOEA/D, MOMBI2, and GWASF-GA, initialize their values by generating a uniform set of 100 vectors. That is the usual setup when dealing with two objectives and only 100 individuals. In addition, MOMBI2 is set to  $\epsilon = 0.001$  and  $\alpha = 0.5$ . Finally, MOEA/D uses a neighborhood size of 20 and its differential evolution operator considers  $CR = 0.5$  and  $F = 0.5$ . We have implemented all the EMO algorithms in Java using the jMetal framework [59]. Table 1 shows a brief summary of the EMO's parameters.

In addition, we have included a classical mathematical optimization method in our experiments, the Nelder-Mead simplex method [31]. This classical method allows us to benchmark the performance obtained by the different EMO algorithms when compared with traditional approaches. In order to adapt the Nelder-Mead algorithm to our multiobjective problem, we employ the adaptive  $\epsilon$ -constraint method [32], [33], which allows single-objective optimization methods to deal with multiple objectives. The Nelder-Mead's approach also involves starting from different

**TABLE 2.** Average HVR values for every algorithm and model instance. The best value for each model instance is shown in bold font. Additionally, the average HVR values across the multiple instances is shown along with the standard deviation.

HVR	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	Avg.	Std. dev.
MOEA/D	<b>0.946</b>	0.956	<b>0.917</b>	<b>0.885</b>	<b>0.906</b>	<b>0.936</b>	<b>0.912</b>	<b>0.93</b>	0.857	0.892	<b>0.902</b>	0.812	<b>0.794</b>	<b>0.895</b>	<b>0.884</b>	<b>0.895</b>	0.045
SPEA2	0.904	0.969	0.817	0.851	0.859	0.914	0.878	0.803	0.757	0.901	0.825	0.748	0.688	0.794	0.803	0.834	0.073
SMS-EMOA	0.901	0.969	0.795	0.857	0.85	0.93	0.87	0.879	0.836	0.908	0.863	0.802	0.779	0.868	0.843	0.863	0.051
IBEA	0.859	0.95	0.758	0.832	0.778	0.919	0.851	0.876	0.851	0.878	0.863	0.809	0.737	0.84	0.845	0.843	0.056
NSGA-II	0.902	<b>0.972</b>	0.821	0.856	0.865	0.926	0.888	0.821	0.757	<b>0.909</b>	0.839	0.764	0.746	0.822	0.822	0.847	0.065
GWASFGA	0.881	0.959	0.769	0.816	0.808	0.899	0.825	0.853	0.827	0.863	0.861	0.742	0.69	0.855	0.841	0.833	0.065
MOMB12	0.873	0.921	0.799	0.846	0.831	0.894	0.838	0.857	<b>0.863</b>	0.843	0.878	<b>0.824</b>	0.758	0.839	0.857	0.848	0.039
Nelder-Mead	0.161	0.169	0.074	0.177	0.17	0.284	0.334	0.353	0.608	0.549	0.383	0.501	0.288	0.348	0.411	0.321	0.155

solutions for obtaining a Pareto set approximation. Therefore, each run generates 50 random solutions that are optimized until reaching 200 evaluations. This setup slightly modifies the one employed by the EMO algorithms because the Nelder-Mead simplex method requires a model evaluation for every single modification in the decision variables.

### B. ANALYSIS OF THE EMO ALGORITHMS' PERFORMANCE

We evaluate the performance of the selected EMO algorithms using widely-used unary and binary multiobjective performance indicators. First, the calibration results are analyzed using a unary performance indicator and attainment surfaces in Section V-B1. Then, we continue the analysis by means of a binary performance indicator and a statistical test in Section V-B2.

#### 1) UNARY PERFORMANCE INDICATOR AND ATTAINMENT SURFACES

Unary performance indicators evaluate a single Pareto front approximation individually. We have selected hyper-volume ratio (HVR) [13] as our unary performance indicator. HVR measures the distribution and convergence of a given Pareto front approximation. It is defined as  $HVR = HV(P)/HV(P^*)$ , with  $HV(P)$  being the volume of the given Pareto front approximation and  $HV(P^*)$  the volume of the true Pareto front. However, we do not know the true Pareto front for any of the model instances, so we use a pseudo-optimal Pareto front for computing the HVR values instead. The pseudo-optimal Pareto front is an approximation obtained by merging all the Pareto front approximations generated by every algorithm for that instance in every independent execution and removing the dominated solutions.

Table 2 shows the computed values of HVR for the resulting Pareto front approximations of each algorithm for every model instance. These values are presented using the average of the individual HVR values computed for the individual Pareto front approximations resulting in each of the 30 algorithm executions. The average HVR values show that MOEA/D consistently achieves better values than the other algorithms for most model instances, obtaining the best average HVR in all but four instances. However, for these four instances MOEA/D obtains values close to the best ones. For example, in P9 and P12 the best HVR values are obtained by MOMB12 (0.863 and 0.824), closely followed by MOEA/D (0.857 and 0.812). In addition, MOEA/D obtains the best average value across the 15 instances with the second lowest standard deviation. These results also highlight the

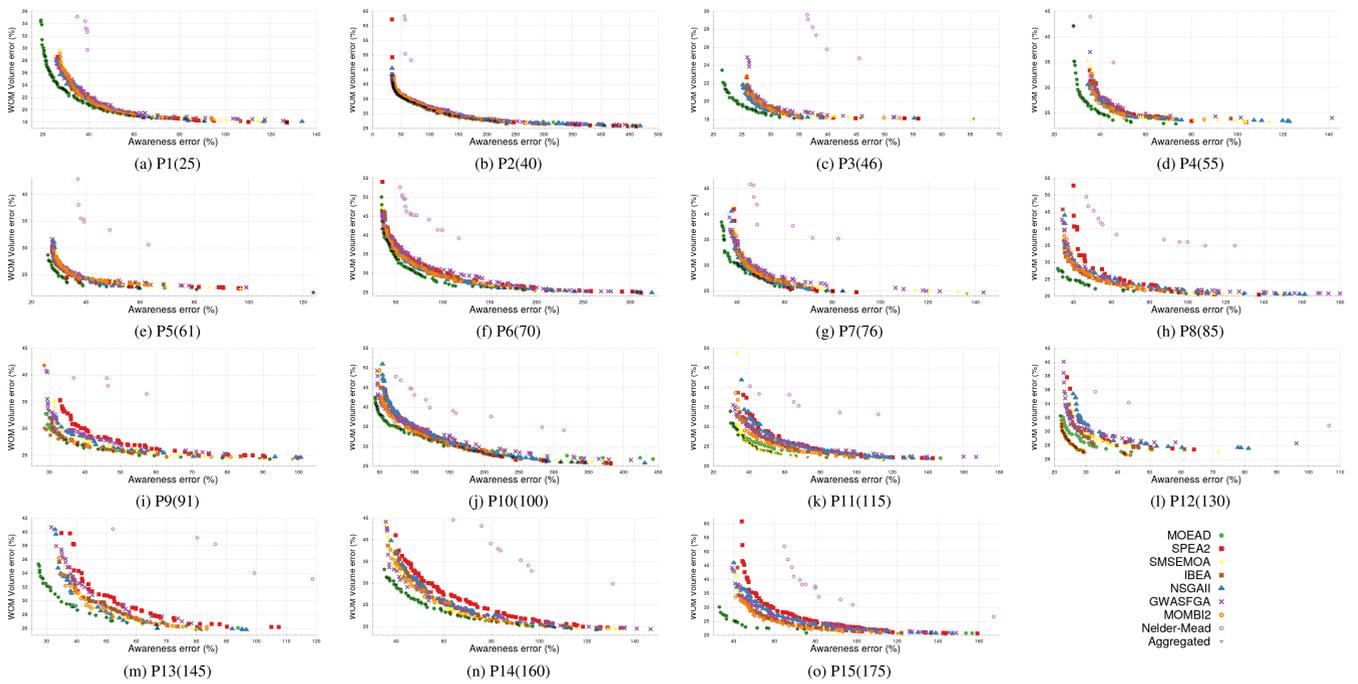
poor performance of the Nelder-Mead simplex method when calibrating our problem instances. It obtains the lowest HVR value for every problem instance with a very significant difference with respect to the EMO approaches (an average value of 0.321 while the worst performing EMO algorithm is over 0.83).

The results of HVR can be visually corroborated using attainment surfaces [60] for each model calibration instance, reported in Figure 3. These attainment surfaces exhibit that the surface obtained by MOEA/D (represented using green filled circles) outperforms the other algorithms for most instances. These surfaces are coherent with the HVR values and P12 is the only problem instance where MOEA/D is visually dominated by other algorithm (MOMB12, represented by orange circles). Nevertheless, these surfaces also reflect how every EMO algorithm performs competitively for certain instances such as P2 where every attainment surface converges to the aggregated Pareto front approximation. By using these surfaces we can also observe the visual difference between Nelder-Mead and the EMO algorithms, where Nelder-Mead's surface is outperformed by every EMO algorithm.

#### 2) BINARY PERFORMANCE INDICATORS AND STATISTICAL SIGNIFICANCE/TESTS

A binary performance indicator compares two given Pareto front approximations generated for the same model instance. Our selected binary performance indicator is the multiplicative  $I_\epsilon$  measure [34]. The calculation of  $I_\epsilon(P, Q)$  for Pareto front approximations  $P$  and  $Q$  is shown in the following equation:  $I_\epsilon(P, Q) = \inf_{\epsilon \in \mathbb{R}} \{ \forall z^2 \in B \exists z^1 \in A : z^1 \succeq_\epsilon z^2 \}$ . The value computed by  $I_\epsilon(P, Q)$  represents the minimum factor required to multiply every element in  $P$  in order to weakly dominate  $Q$ . That is, the minimum  $\epsilon$  so  $P \in$ -dominates  $Q$ . As our calibration problem constitutes a minimization problem, if  $I_\epsilon(P, Q) < I_\epsilon(Q, P)$  then we can assume that  $P$  is better than  $Q$ .

Tables 3 and 4 present the multiplicative  $I_\epsilon$  values computed for the resulting Pareto front approximations of each algorithm for every model instance. These values are the average of each possible  $I_\epsilon(P, Q)$ , with  $P$  and  $Q$  being any pair of Pareto front approximations of different algorithms, resulting from any of the 30 independent executions (i.e., a pair-wise comparison of every run). The values for Tables 3 and 4 support the previous conclusions drawn from the HVR indicator: MOEA/D outperforms the remaining



**FIGURE 3.** Attainment surfaces for the different problem instances. Each attainment surface represents the aggregated Pareto front approximation obtained by each algorithm: MOEA/D (green filled circles), SPEA2 (red squares), SMS-EMOA (yellow crosses), IBEA (brown asterisks), NSGA-II (blue triangles), GWASF-GA (purple blades), MOMB12 (empty orange circles), and Nelder-Mead (light purple empty circles). Finally, inverted triangles represent the aggregated Pareto front.

**TABLE 3.** Average  $I_e$  values for every pair of algorithms for P1, P2, P3, P4, P5, P6, P7, and P8 instances. Best values for each pair of algorithms are shown using bold font.

P1(25)									P2(40)								
	MOEAD	SPEA2	SMS-EMOA	IBEA	NSGA-II	GWASF-GA	MOMB12	Nelder-Mead		MOEAD	SPEA2	SMS-EMOA	IBEA	NSGA-II	GWASF-GA	MOMB12	Nelder-Mead
MOEAD	-	<b>1.0398</b>	<b>1.0347</b>	<b>1.0026</b>	<b>1.0383</b>	<b>1.0184</b>	<b>1.0118</b>	<b>0.7244</b>	MOEAD	-	<b>1.0736</b>	<b>1.0681</b>	<b>1.0518</b>	<b>1.0725</b>	1.0527	<b>1.0301</b>	<b>0.5817</b>
SPEA2	1.3562	-	<b>1.043</b>	<b>1.0423</b>	1.0525	1.0798	<b>1.0399</b>	<b>0.8211</b>	SPEA2	1.1193	-	<b>1.0503</b>	<b>1.0479</b>	1.0619	1.0991	<b>1.0418</b>	<b>0.6305</b>
SMS-EMOA	1.3804	1.0601	-	<b>1.0465</b>	1.0599	1.0934	<b>1.0395</b>	<b>0.8297</b>	SMS-EMOA	1.1576	1.0794	-	<b>1.049</b>	1.0831	1.1331	<b>1.0454</b>	<b>0.6542</b>
IBEA	1.3821	1.0899	1.0793	-	1.0894	1.1004	<b>1.0494</b>	<b>0.8301</b>	IBEA	1.1934	1.122	1.0903	-	1.1221	1.1677	<b>1.0559</b>	<b>0.6747</b>
NSGA-II	1.3571	<b>1.0507</b>	<b>1.0428</b>	<b>1.0403</b>	-	1.0737	<b>1.0394</b>	<b>0.8192</b>	NSGA-II	1.1091	<b>1.0488</b>	<b>1.0419</b>	<b>1.0397</b>	-	1.0878	<b>1.0328</b>	<b>0.6275</b>
GWASF-GA	1.2867	<b>1.0596</b>	<b>1.0596</b>	<b>1.0475</b>	<b>1.0595</b>	-	<b>1.0545</b>	<b>0.7947</b>	GWASF-GA	<b>1.0485</b>	<b>1.0557</b>	<b>1.0625</b>	<b>1.0632</b>	<b>1.058</b>	-	<b>1.0472</b>	<b>0.5933</b>
MOMB12	1.4394	1.0991	1.085	1.0717	1.0971	1.1315	-	<b>0.8543</b>	MOMB12	1.2211	1.177	1.1495	1.112	1.1758	1.1985	-	<b>0.6889</b>
Nelder-Mead	1.8792	1.8375	1.8273	1.7152	1.8361	1.8014	1.7512	-	Nelder-Mead	4.9663	5.1355	5.0527	4.7961	5.1252	4.9992	4.6028	-
P3(46)									P4(55)								
	MOEAD	SPEA2	SMS-EMOA	IBEA	NSGA-II	GWASF-GA	MOMB12	Nelder-Mead		MOEAD	SPEA2	SMS-EMOA	IBEA	NSGA-II	GWASF-GA	MOMB12	Nelder-Mead
MOEAD	-	<b>1.0035</b>	<b>0.9973</b>	<b>0.9812</b>	<b>1.0026</b>	<b>0.9891</b>	<b>0.9937</b>	<b>0.728</b>	MOEAD	-	<b>1.023</b>	<b>1.0289</b>	<b>1.0094</b>	<b>1.0218</b>	<b>0.9975</b>	<b>1.0206</b>	<b>0.7981</b>
SPEA2	1.181	-	<b>1.0256</b>	<b>1.0202</b>	1.0351	<b>1.0156</b>	<b>1.0276</b>	<b>0.8331</b>	SPEA2	1.1484	-	1.0441	<b>1.0385</b>	1.047	<b>1.0359</b>	<b>1.0473</b>	<b>0.8805</b>
SMS-EMOA	1.1927	1.0428	-	<b>1.0292</b>	1.0463	<b>1.0267</b>	1.038	<b>0.8433</b>	SMS-EMOA	1.1563	<b>1.0386</b>	-	<b>1.0305</b>	1.046	<b>1.0395</b>	<b>1.0401</b>	<b>0.8882</b>
IBEA	1.2139	1.0547	1.0481	-	1.057	1.038	<b>1.0445</b>	<b>0.8599</b>	IBEA	1.1619	1.0464	1.0472	-	1.0487	<b>1.0412</b>	<b>1.0385</b>	<b>0.8948</b>
NSGA-II	1.1751	<b>1.0282</b>	<b>1.0231</b>	<b>1.0157</b>	-	<b>1.0126</b>	<b>1.0238</b>	<b>0.8316</b>	NSGA-II	1.135	<b>1.0308</b>	<b>1.0361</b>	<b>1.0264</b>	-	<b>1.0236</b>	<b>1.0361</b>	<b>0.871</b>
GWASF-GA	1.1997	1.0464	1.0414	<b>1.0343</b>	1.0504	-	<b>1.0419</b>	<b>0.847</b>	GWASF-GA	1.1347	1.055	1.0639	1.0534	1.0585	-	1.0641	<b>0.867</b>
MOMB12	1.1917	1.0367	<b>1.0315</b>	<b>1.0211</b>	1.0401	<b>1.0207</b>	-	<b>0.8428</b>	MOMB12	1.1695	1.0496	1.049	<b>1.0325</b>	1.0545	<b>1.0492</b>	-	<b>0.9001</b>
Nelder-Mead	2.5998	2.5702	2.5519	2.5005	2.568	2.5343	2.542	-	Nelder-Mead	1.6515	1.663	1.6692	1.6161	1.654	1.6141	1.6392	-
P5(61)									P6(70)								
	MOEAD	SPEA2	SMS-EMOA	IBEA	NSGA-II	GWASF-GA	MOMB12	Nelder-Mead		MOEAD	SPEA2	SMS-EMOA	IBEA	NSGA-II	GWASF-GA	MOMB12	Nelder-Mead
MOEAD	-	<b>1.0121</b>	<b>1.0043</b>	<b>0.9888</b>	<b>1.0093</b>	<b>0.9941</b>	<b>0.9958</b>	<b>0.8466</b>	MOEAD	-	<b>1.0505</b>	<b>1.0557</b>	<b>1.0344</b>	<b>1.0545</b>	<b>1.0515</b>	<b>1.0206</b>	<b>0.7621</b>
SPEA2	1.0986	-	1.0376	<b>1.0425</b>	1.0415	1.0426	1.0408	<b>0.9117</b>	SPEA2	1.0932	-	<b>1.0582</b>	<b>1.0547</b>	1.0607	1.0632	<b>1.0689</b>	<b>0.8034</b>
SMS-EMOA	1.088	<b>1.0314</b>	-	<b>1.03</b>	1.0349	<b>1.0339</b>	<b>1.0282</b>	<b>0.9043</b>	SMS-EMOA	1.0972	1.0597	-	<b>1.0464</b>	1.0656	1.0799	<b>1.0619</b>	<b>0.7961</b>
IBEA	1.0901	1.0591	1.0499	-	1.0565	1.0468	1.0394	<b>0.9013</b>	IBEA	1.1314	1.1011	1.0908	-	1.1031	1.1293	1.0735	<b>0.7985</b>
NSGA-II	1.0865	<b>1.026</b>	<b>1.0278</b>	<b>1.0316</b>	-	<b>1.0314</b>	<b>1.03</b>	<b>0.9011</b>	NSGA-II	1.0791	<b>1.0413</b>	<b>1.046</b>	<b>1.0424</b>	-	<b>1.0538</b>	<b>1.0545</b>	<b>0.7934</b>
GWASF-GA	1.0841	1.0459	1.042	<b>1.0365</b>	1.0464	-	1.0398	<b>0.8908</b>	GWASF-GA	1.1005	<b>1.0615</b>	<b>1.0786</b>	<b>1.0836</b>	1.0759	-	<b>1.0875</b>	<b>0.8036</b>
MOMB12	1.0899	<b>1.0369</b>	1.033	<b>1.0305</b>	1.038	<b>1.0371</b>	-	<b>0.9092</b>	MOMB12	1.106	1.1035	1.1009	<b>1.0672</b>	1.1036	1.1072	-	<b>0.7821</b>
Nelder-Mead	1.7265	1.7202	1.7024	1.6418	1.7149	1.6832	1.6804	-	Nelder-Mead	1.7925	1.803	1.8023	1.7494	1.8046	1.793	1.6763	-
P7(76)									P8(85)								
	MOEAD	SPEA2	SMS-EMOA	IBEA	NSGA-II	GWASF-GA	MOMB12	Nelder-Mead		MOEAD	SPEA2	SMS-EMOA	IBEA	NSGA-II	GWASF-GA	MOMB12	Nelder-Mead
MOEAD	-	<b>1.0208</b>	<b>1.0117</b>	<b>1.002</b>	<b>1.0196</b>	<b>0.9931</b>	<b>1.0003</b>	<b>0.7977</b>	MOEAD	-	<b>1.0283</b>	<b>1.0413</b>	<b>1.0222</b>	<b>1.0284</b>	<b>1.0183</b>	<b>1.0192</b>	<b>0.7381</b>
SPEA2	1.1284	-	1.0396	<b>1.0408</b>	1.0489	1.0414	1.0422	<b>0.8819</b>	SPEA2	1.3567	-	1.1705	1.2018	1.0869	1.1694	1.1974	<b>0.9229</b>
SMS-EMOA	1.1225	<b>1.0345</b>	-	<b>1.0338</b>	1.0451	<b>1.0368</b>	<b>1.0345</b>	<b>0.8756</b>	SMS-EMOA	1.2166	<b>1.0091</b>	-	1.0841	<b>1.0165</b>	<b>1.0596</b>	<b>1.0829</b>	<b>0.843</b>
IBEA	1.1278	1.0464	1.0438	-	1.0531	<b>1.044</b>	<b>1.0364</b>	<b>0.8832</b>	IBEA	1.1851	<b>1.0543</b>	<b>1.073</b>	-	<b>1.0565</b>	<b>1.0655</b>	<b>1.0671</b>	<b>0.8257</b>
NSGA-II	1.114	<b>1.0258</b>	<b>1.0282</b>	<b>1.0276</b>	-	<b>1.0292</b>	<b>1.0283</b>	<b>0.8698</b>	NSGA-II	1.3252	<b>1.0521</b>	1.146	1.1755	-	1.1457	1.1722	<b>0.9068</b>
GWASF-GA	1.1176	1.063	1.0571	1.0539	1.0666	-	<b>1.0523</b>	<b>0.8629</b>	GWASF-GA	1.2102	<b>1.0517</b>	1.0944	1.0999	<b>1.0587</b>	-	<b>1.1006</b>	<b>0.8303</b>
MOMB12	1.1351	1.0567	1.0539	1.0471	1.0634	1.0528	-	<b>0.8876</b>	MOMB12	1.2585	<b>1.1061</b>	1.1428	1.1413	<b>1.1122</b>	1.1366	-	<b>0.8745</b>
Nelder-Mead	1.6197	1.6104	1.5963	1.5701	1.6085	1.5667	1.5616	-	Nelder-Mead	2.0949	2.0418	2.0544	1.9747	2.0387	2.0213	1.9565	-

algorithms for most model instances. MOEA/D obtains a lower average  $I_e$  value for every comparison with the other algorithms, with the exception of P2, where GWASF-GA obtains a better indicator value. The Nelder-Mead's simplex method is again outperformed by every EMO algorithm in every model instance.

In addition, we develop a statistical test and study the significance of the  $I_e$  values to avoid that isolated results could bias our former analysis. We perform this test following the methodology described in [61], [62]: let  $N$  be the number of repetitions of two algorithms  $A$  and  $B$ ; then let  $A_i$  and  $B_j$  be two arbitrary resulting Pareto front approximations with

**TABLE 4.** Average  $I-\epsilon$  values for every pair of algorithms for P9, P10, P11, P12, P13, P14, and P15 instances. Best values for each pair of algorithms are shown using bold font.

P9(P1)									P10(P10)								
MOEAD	SPEA2	SMS-EMOA	IBEA	NSGA-II	GWASF-GA	MOMBI2	Nelder-Mead		MOEAD	SPEA2	SMS-EMOA	IBEA	NSGA-II	GWASF-GA	MOMBI2	Nelder-Mead	
MOEAD	-	<b>1.0308</b>	<b>1.0501</b>	<b>1.0492</b>	<b>1.031</b>	<b>1.04</b>	<b>1.0588</b>	<b>0.8305</b>	MOEAD	-	<b>1.1088</b>	<b>1.1079</b>	<b>1.0623</b>	<b>1.1138</b>	<b>1.0753</b>	<b>1.0435</b>	<b>0.7763</b>
SPEA2	1.1565	-	1.1079	1.1262	1.056	1.1389	1.1444	<b>0.9295</b>	SPEA2	1.2404	-	<b>1.0534</b>	<b>1.0492</b>	1.0572	1.1913	<b>1.0814</b>	<b>0.8328</b>
SMS-EMOA	1.0859	<b>1.011</b>	-	1.062	<b>1.0135</b>	1.0656	1.0785	<b>0.8673</b>	SMS-EMOA	1.2665	1.0685	-	<b>1.0494</b>	1.0669	1.2168	<b>1.0897</b>	<b>0.8389</b>
IBEA	1.0682	<b>1.0186</b>	<b>1.0408</b>	-	<b>1.0201</b>	<b>1.0495</b>	1.0614	<b>0.853</b>	IBEA	1.3202	1.116	1.1065	-	1.1145	1.2743	1.1055	<b>0.8585</b>
NSGA-II	1.1536	<b>1.0548</b>	1.1069	1.1247	-	1.1336	1.1426	<b>0.9259</b>	NSGA-II	1.2482	<b>1.0566</b>	<b>1.0501</b>	<b>1.0456</b>	-	1.1981	<b>1.0814</b>	<b>0.8309</b>
GWASF-GA	1.0693	<b>1.0402</b>	<b>1.0598</b>	1.061	<b>1.0398</b>	-	1.0732	<b>0.8345</b>	GWASF-GA	1.0917	<b>1.0929</b>	<b>1.0998</b>	<b>1.0877</b>	<b>1.0988</b>	-	<b>1.0787</b>	<b>0.8035</b>
MOMBI2	1.0674	<b>1.0277</b>	<b>1.0479</b>	<b>1.0478</b>	<b>1.0274</b>	<b>1.0492</b>	-	<b>0.8446</b>	MOMBI2	1.2741	1.1544	1.1499	<b>1.0901</b>	1.1566	1.2297	-	<b>0.8399</b>
Nelder-Mead	1.8012	1.7617	1.7964	1.7787	1.7593	1.7794	1.7872	-	Nelder-Mead	2.0886	1.8768	1.8639	1.8167	1.8787	2.0105	1.8482	-
P11(P15)									P12(P13)								
MOEAD	-	<b>1.0383</b>	<b>1.0484</b>	<b>1.0292</b>	<b>1.04</b>	<b>1.0362</b>	<b>1.0425</b>	<b>0.7775</b>	MOEAD	-	<b>1.022</b>	<b>1.0346</b>	<b>1.0303</b>	<b>1.0213</b>	<b>1.0108</b>	<b>1.0361</b>	<b>0.7534</b>
SPEA2	1.2678	-	1.0932	1.1233	1.0661	1.1961	1.1572	<b>0.9519</b>	SPEA2	1.171	-	1.0747	1.0919	1.0513	1.1252	1.1105	<b>0.842</b>
SMS-EMOA	1.2128	<b>1.0201</b>	-	1.0771	<b>1.0308</b>	1.1463	1.1084	<b>0.9136</b>	SMS-EMOA	1.1379	<b>1.0293</b>	-	1.0585	<b>1.021</b>	1.0953	1.0775	<b>0.8222</b>
IBEA	1.168	<b>1.0498</b>	<b>1.0614</b>	-	<b>1.0516</b>	1.105	1.0747	<b>0.8769</b>	IBEA	1.1081	<b>1.0215</b>	<b>1.0317</b>	-	<b>1.0132</b>	1.0691	1.052	<b>0.8051</b>
NSGA-II	1.2499	<b>1.0391</b>	1.0796	1.1074	-	1.1802	1.1403	<b>0.9395</b>	NSGA-II	1.1676	<b>1.048</b>	1.064	1.0816	-	1.1239	1.103	<b>0.8438</b>
GWASF-GA	1.1132	<b>1.0602</b>	<b>1.0769</b>	<b>1.0613</b>	<b>1.0641</b>	-	<b>1.0766</b>	<b>0.814</b>	GWASF-GA	1.0839	<b>1.0475</b>	<b>1.0615</b>	<b>1.0621</b>	<b>1.0468</b>	-	1.0692	<b>0.7818</b>
MOMBI2	1.1473	<b>1.0463</b>	<b>1.0554</b>	<b>1.0365</b>	<b>1.0471</b>	1.0872	-	<b>0.8598</b>	MOMBI2	1.0837	<b>1.0107</b>	<b>1.0211</b>	<b>1.0265</b>	<b>1.0048</b>	<b>1.0441</b>	-	<b>0.7858</b>
Nelder-Mead	1.887	1.8816	1.8895	1.8076	1.8823	1.8476	1.8152	-	Nelder-Mead	1.7398	1.6749	1.6962	1.6943	1.6719	1.6915	1.7104	-
P13(P145)									P14(P16)								
MOEAD	-	<b>1.0256</b>	<b>1.051</b>	<b>1.0349</b>	<b>1.0384</b>	<b>1.0104</b>	<b>1.0511</b>	<b>0.744</b>	MOEAD	-	<b>1.0142</b>	<b>1.0463</b>	<b>1.0264</b>	<b>1.0259</b>	<b>1.0372</b>	<b>1.0274</b>	<b>0.6615</b>
SPEA2	1.2326	-	1.0922	1.072	1.0866	1.1095	1.1032	<b>0.862</b>	SPEA2	1.1558	-	1.1044	1.0754	1.0662	1.1316	1.0993	<b>0.7218</b>
SMS-EMOA	1.2015	<b>1.018</b>	-	<b>1.0286</b>	<b>1.0488</b>	1.0832	<b>1.057</b>	<b>0.8441</b>	SMS-EMOA	1.0959	<b>1.0029</b>	-	<b>1.0235</b>	<b>1.0194</b>	<b>1.0679</b>	<b>1.0415</b>	<b>0.6843</b>
IBEA	1.2382	<b>1.0474</b>	1.0724	-	1.0812	1.1179	<b>1.0823</b>	<b>0.869</b>	IBEA	1.1351	<b>1.0331</b>	1.0857	-	<b>1.0519</b>	1.12	1.0707	<b>0.6961</b>
NSGA-II	1.1809	<b>1.0192</b>	1.0542	<b>1.0384</b>	-	<b>1.064</b>	<b>1.0643</b>	<b>0.8295</b>	NSGA-II	1.1366	<b>1.0286</b>	1.0846	1.0557	-	1.1111	<b>1.0793</b>	<b>0.707</b>
GWASF-GA	1.1481	<b>1.0468</b>	<b>1.0803</b>	<b>1.0649</b>	1.0648	-	<b>1.0871</b>	<b>0.8042</b>	GWASF-GA	1.0794	<b>1.0332</b>	1.069	<b>1.0531</b>	<b>1.0462</b>	-	<b>1.0567</b>	<b>0.6837</b>
MOMBI2	1.2741	<b>1.0877</b>	1.1095	1.088	1.1179	1.1533	-	<b>0.8997</b>	MOMBI2	1.137	<b>1.0657</b>	1.1033	<b>1.0674</b>	1.0803	1.1184	-	<b>0.6915</b>
Nelder-Mead	2.0712	1.9167	1.9591	1.9175	1.9574	1.9523	1.9364	-	Nelder-Mead	2.2442	2.1266	2.1995	2.1276	2.155	2.2156	2.0988	-
P15(P175)																	
MOEAD	-	<b>1.0294</b>	<b>1.0428</b>	<b>1.0306</b>	<b>1.0331</b>	<b>1.0278</b>	<b>1.0471</b>	<b>0.7095</b>									
SPEA2	1.2198	-	1.1024	1.1218	1.0788	1.1289	1.1481	<b>0.7989</b>									
SMS-EMOA	1.1603	<b>1.0186</b>	-	1.062	<b>1.0373</b>	1.0869	1.0833	<b>0.7549</b>									
IBEA	1.147	<b>1.0476</b>	<b>1.0615</b>	-	<b>1.0546</b>	1.0835	1.0665	<b>0.7434</b>									
NSGA-II	1.1874	<b>1.0308</b>	1.0764	1.0922	-	1.1041	1.1166	<b>0.7783</b>									
GWASF-GA	1.1287	<b>1.0477</b>	<b>1.0701</b>	<b>1.0688</b>	<b>1.0548</b>	-	1.0892	<b>0.7536</b>									
MOMBI2	1.1328	<b>1.0455</b>	<b>1.0557</b>	<b>1.0369</b>	<b>1.051</b>	<b>1.076</b>	-	<b>0.7284</b>									
Nelder-Mead	1.9677	1.9101	1.9207	1.8513	1.9125	1.9083	1.8518	-									

$1 \leq i \leq N$  and  $1 \leq j \leq N$ ; finally let  $p_{A_i}(B_j)$  be 1 if  $A_i$  dominates  $B_j$  based on the computed  $I_\epsilon$  value (i.e.,  $I_\epsilon(A_i, B_j) \leq 1$  and  $I_\epsilon(B_j, A_i) > 1$ ) and 0 otherwise. Using  $p_{A_i}(B_j)$ , we can define  $P$  by the equation  $P_{A_i}(B) = 1/N \cdot \sum_{j=1}^N p_{A_i}(B_j)$  as the percentage or resulting Pareto front approximations obtained by algorithm  $B$  that are dominated by  $A_i$ .

We have included boxplots representing the resulting  $\epsilon$  dominance percentage values in Figure 4, which contains the computed  $P_A(B)$  for every pair of algorithms and each model instance with the exception of Nelder-Mead's simplex method, which has been widely outperformed at this stage of the analysis. In these charts we can observe how MOEA/D generally obtains bigger dominance percentages than the remaining algorithms. On the one hand, the boxplots on the right of the MOEA/D label contain the values of  $P_B(MOEA/D)$ . In these charts we can observe how both their boxes and whiskers cover a considerable percentage of the interval, implying a big dominance probability. In contrast, boxplots below the MOEA/D label contain the values of  $P_{MOEA/D}(A)$ , where we can notice that, in general, the values for all instances but P9 are small. This is coherent with the previous conclusions of our analysis as MOEA/D was already outperformed by other algorithms in instances like P9.

Finally, we can consider vector  $P_A(B) = P_{A_1}(B), P_{A_2}(B), \dots, P_{A_N}(B)$  as a random variable representing the percentage of times that algorithm  $A$  outperforms algorithm  $B$ , since it is the proportion of resulting Pareto approximations of algorithm  $A$  dominating the Pareto approximations delivered by algorithm  $B$ . Therefore, if the expectation of  $P_A(B)$  is greater than the expectation of  $P_B(A)$  we can claim that  $A$  is better than  $B$  because it is more likely that the resulting Pareto

approximations of  $A$  dominate those obtained by  $B$ . Our selected test is the Wilcoxon ranksum test (null hypothesis  $E(P_A(B)) = E(P_B(A))$ , alternate hypothesis  $E(P_A(B)) > E(P_B(A))$ ), seeing that it has proven to be useful when analyzing the performance of evolutionary algorithms [63]. The significance level considered is 0.05.

Table 5 shows the significance for the resulting  $p$ -values of the statistical test. These results are again consistent with the previous indicator values, as MOEA/D shows an outstanding and robust behavior, being able to perform significantly better than the remaining algorithms in most instances. Hence, MOEA/D is the best performing decomposition-based algorithm for our problem, since it almost always outperforms GWASF-GA. Regarding the performance of the rest of the algorithms, we can see how the Pareto dominance-based EMO algorithms (NSGA-II and SPEA2) outperform most of the algorithms for the first seven instances. However, if we compare these two algorithms, we can observe that SPEA2 does not significantly outperform NSGA-II in any instance, suggesting that NSGA-II is the best algorithm from this family when dealing with the ABM calibration problem. SMS-EMOA would be the best performing indicator-based EMO algorithm but we can find some instances like P12 where it is outperformed by MOMBI2.

Nevertheless, the behavior of MOEA/D is eroded when dealing with specific instances like P6 or some of the bigger instances like P9, P12, and P15. Although MOEA/D obtains the best HVR values for some of these instances, the statistical tests revealed that it is dominated by other algorithms. In the case of P6, SMS-EMOA and NSGA-II are the best performing algorithms and significantly outperform MOEA/D.

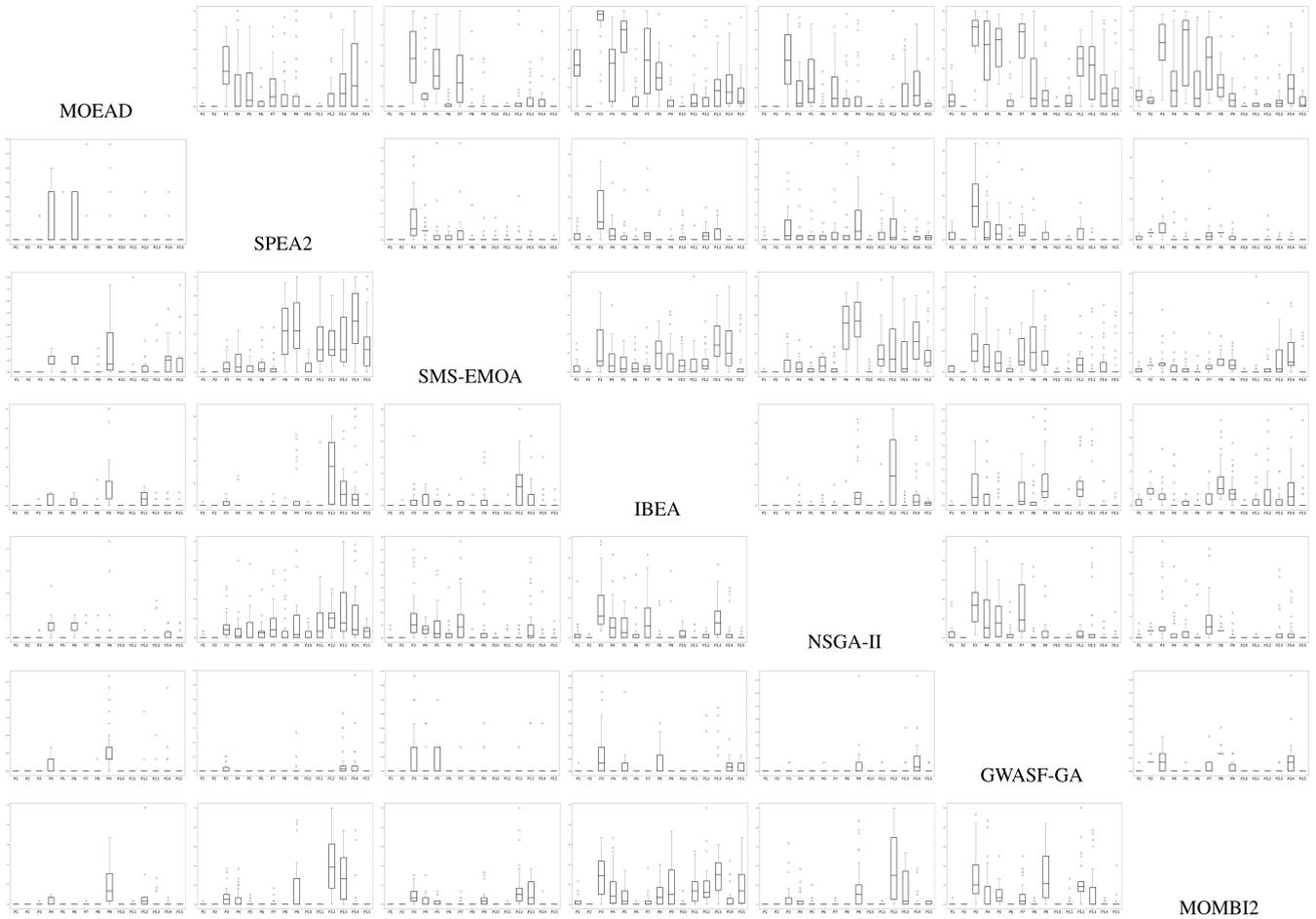


FIGURE 4. Boxplots representing the  $\epsilon$  dominance percentage values for each model instance and every pair of EMOs.

As we pointed out, MOMBI2 arises as the best performing algorithm for the P12 model instance. The p-values for the P9 instance, where SMS-EMOA and IBEA perform significantly better than the other algorithms, with SMS-EMOA finally outperforming IBEA. SMS-EMOA shows better convergence for this instance, which explains the better dominance  $P_B(A)$  values shown at Figure 4 and corroborated by the statistical test. Finally, SMS-EMOA is the best performing EMO for the P15 instance, as it dominates the remaining six algorithms despite obtaining the fourth average HVR value, which shows the great convergence of SMS-EMOA for this instance. Therefore, we can observe how SMS-EMOA’s specific features (such as its combination of hypervolume maximization and its replacement strategy) are effective with optimization problems with these characteristics.

C. DRAWBACKS OF THE METHODS

The previous sections have shown how the EMO algorithms, specially MOEA/D and SMS-EMOA, successfully perform in our ABM calibration problem. However, we also acknowledge that these algorithms present certain drawbacks. For example, we can observe how the increased dimensionality of the bigger instances erodes the behavior of most algorithms (we will analyze the impact of the dimensionality in next

Section V-D). The drawbacks of the EMO algorithms are presented next:

- The main drawback of MOEA/D is its sensitivity to the properties of the different problem instances, further discussed in Section V-D. In addition, we can observe how the increased dimensionality reduces its performance. This can be identified even if MOEA/D outperforms other algorithms. For example, the values of Table 2 show that its HVR value for instances bigger than P8 do not surpasses the computed average HVR value for every instance, showing a decay of performance even when outperforming other algorithms.
- SPEA2 and NSGA-II present similar drawbacks. According to their p-values, they perform competitively for the instances below P10, where they significantly outperform several EMO algorithms. However, they reduce their performance, specially for the bigger instances, where they are outperformed by IBEA and GWASF-GA. In addition, a direct comparison between SPEA2 and NSGA-II reveals that NSGA-II is better in most instances.
- SMS-EMOA is the most robust EMO algorithm regarding the dimensionality of our calibration problem. This could be related to its combination of hypervolume

**TABLE 5. Significance of the results with respect of the computed  $p$ -values for every EMO algorithm against the other methods for every model instance. Cells marked as + represent statistical significance.**

MOEA/D	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	SPEA2	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	
SPEA2	-	-	+	-	+	-	+	+	-	-	-	+	+	+	+	MOEA/D	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
SMS-EMOA	-	-	+	+	+	-	+	-	-	-	-	-	+	-	-	SMS-EMOA	-	-	+	-	-	-	-	-	-	-	-	-	-	-	-	-
IBEA	+	-	+	+	+	-	+	+	-	-	+	-	+	+	+	IBEA	+	-	+	+	+	-	+	-	-	+	-	-	-	-	-	-
NSGA-II	-	-	+	-	+	-	+	+	-	-	+	+	+	+	+	NSGA-II	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
GWASF-GA	+	-	+	+	+	+	+	+	-	-	+	+	+	+	+	GWASF-GA	+	-	+	+	+	+	+	-	+	-	-	-	+	-	-	-
MOMBI2	+	+	+	+	+	+	+	+	-	+	+	-	+	+	+	MOMBI2	+	+	+	-	-	-	+	+	-	-	-	-	-	-	-	-

SMS-EMOA	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	IBEA	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	
MOEA/D	-	-	-	-	-	+	-	-	+	-	-	-	-	-	+	MOEA/D	-	-	-	-	-	-	-	-	+	-	-	-	-	-	-	-
SPEA2	-	-	-	-	-	+	-	-	+	+	+	+	+	+	+	SPEA2	-	-	-	-	-	-	-	-	+	-	-	-	-	+	+	+
IBEA	+	-	+	+	+	+	+	+	-	+	+	-	+	+	+	SMS-EMOA	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
NSGA-II	-	-	-	-	-	+	-	+	+	-	+	+	-	+	+	NSGA-II	-	-	-	-	-	-	-	-	+	-	-	-	+	-	+	+
GWASF-GA	+	-	+	+	+	+	+	+	+	-	+	+	-	+	+	GWASF-GA	-	-	-	+	-	-	+	+	-	+	-	-	+	-	-	-
MOMBI2	+	+	-	-	-	+	+	+	+	-	-	-	-	-	+	MOMBI2	-	+	-	-	-	-	+	+	-	-	-	-	-	-	-	+

NSGA-II	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	GWASF-GA	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	
MOEA/D	-	-	-	-	-	+	-	-	-	-	-	-	-	-	-	MOEA/D	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
SPEA2	-	-	+	+	-	+	+	-	-	+	+	+	+	+	+	SPEA2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	+	+
SMS-EMOA	-	-	+	+	+	+	+	-	-	-	-	-	-	-	-	SMS-EMOA	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
IBEA	+	-	+	+	+	+	+	+	-	+	-	-	-	-	-	IBEA	-	-	-	-	-	-	-	-	-	-	-	-	-	-	+	+
GWASF-GA	+	-	+	+	+	+	+	+	+	-	-	+	-	-	-	NSGA-II	-	-	-	-	-	-	-	-	-	-	-	-	-	-	+	-
MOMBI2	+	+	+	-	+	+	+	+	-	-	-	-	-	-	-	MOMBI2	-	+	-	-	-	-	-	+	-	-	-	-	-	-	-	+

MOMBI2	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15
MOEA/D	-	-	-	-	-	-	-	-	-	-	-	+	-	-	-
SPEA2	-	-	-	+	-	-	-	-	-	-	-	+	+	-	-
SMS-EMOA	-	-	-	-	-	-	-	-	-	-	-	-	+	-	-
IBEA	-	-	+	+	+	-	-	-	-	-	+	+	+	-	+
NSGA-II	-	-	-	-	-	-	-	-	+	-	-	+	+	-	+
GWASF-GA	-	-	+	+	+	-	+	-	+	-	+	+	-	-	+

maximization and the replacement strategy that mainly targets the solutions that contribute poorly to the hypervolume of its respective fronts. However, SMS-EMOA is unsuccessful dealing with the smaller instances since it barely outperforms GWASF-GA and IBEA for the instances below P6, where its strategy seems to obtain a lesser impact.

- IBEA and GWASF-GA also present similar drawbacks. In general, they both show a poor performance for our ABM calibration problem, with the only exceptions of P9 for IBEA and P14 for GWASF-GA. However, this can be related with the loss of performance observed in the rest of the algorithms.
- Finally, MOMBI2 stands out by obtaining the best HVR values for two instances and increasing its performance with the number of decision variables of the instances. Nevertheless, its behavior seems unstable, since it outperforms the remaining algorithms for P12 but fails to outperform any of them for P14.

**D. INFLUENCE OF THE INSTANCES' PROPERTIES ON THE ALGORITHMS' PERFORMANCE**

In view of the results obtained by both the unary and binary indicators, we can observe how specific properties of the problem instances are affecting the performance of the evaluated EMO algorithms. These properties are the shape of the feasible region, the shape of the Pareto front, and the dimensionality of the problem instance. Some studies [64], [65] have pointed out a relationship between the performance of decomposition-based EMO algorithms (such as MOEA/D, the best performing algorithm in our study)

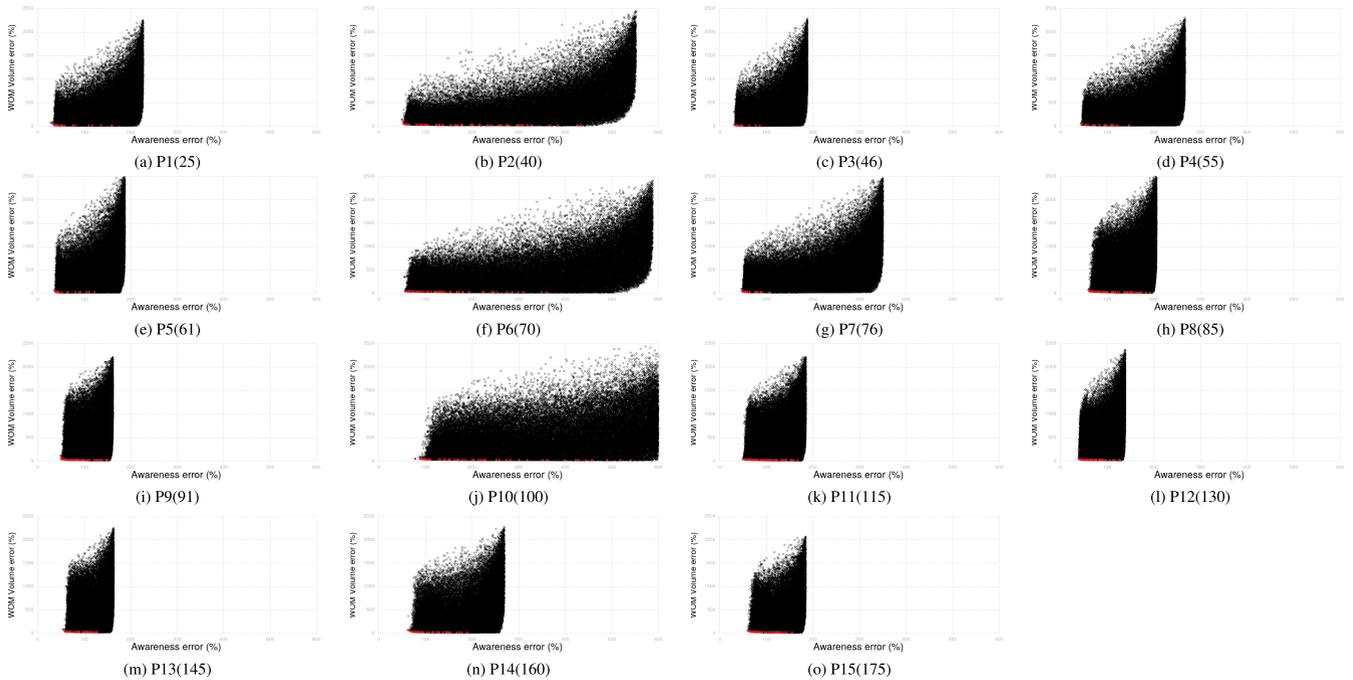
and the shapes of both the feasible region and the Pareto front.

Figure 5 shows an approximation to the search space configuration of the problem instances tackled in the current contribution using scatter-plots. In addition, Figure 6 displays the shape of the Pareto fronts. The shape of the feasible region is approximated by sampling 100,000 random configurations for each problem instance. In the plots in Figure 5, we can observe that the search space extent is considerably bigger for P2, P6, P7, and P10 instances when compared with the rest of the problem instances. Therefore, the shape of the feasible region for these instances can explain their difficulty, specially for the performance of the EMO algorithms that employ reference points [65].

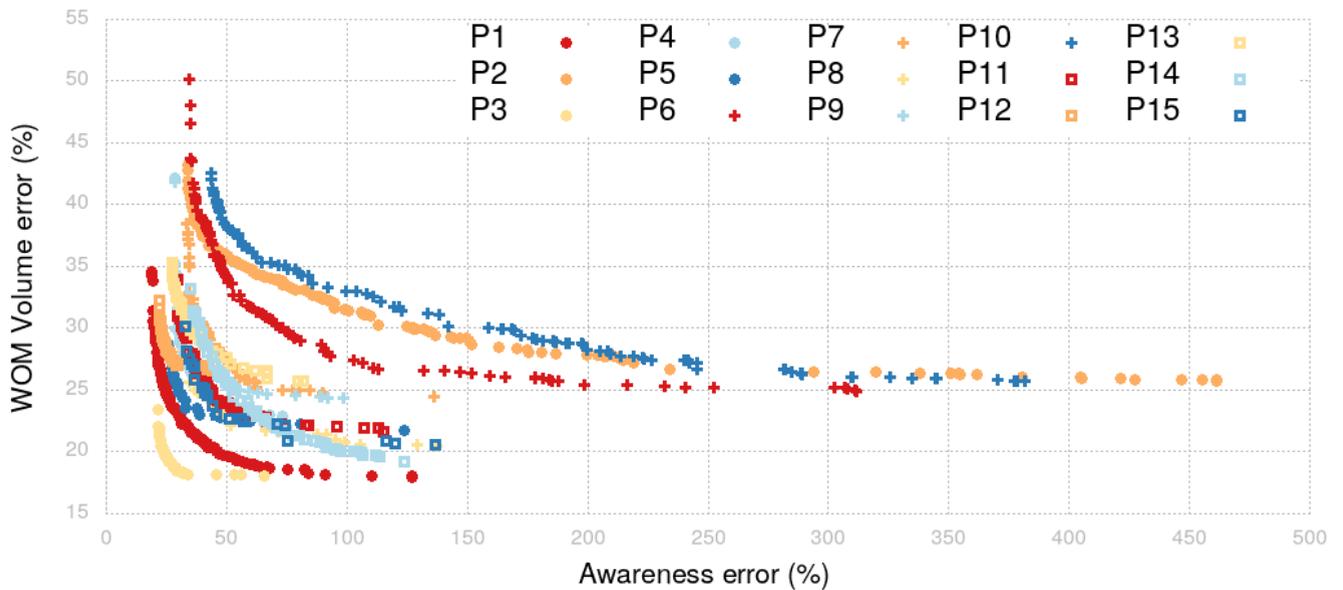
We approximate the shape of the global Pareto fronts by using the aggregated Pareto fronts, which contain the overall non-dominated solutions obtained for each problem instance. In the plot in Figure 6, we can observe that P2, P6, and P10 instances have a long tail shape compared with the rest of instances. These long tail shapes can explain a reduction of performance for the algorithms using weight vectors because these shapes are non-symmetric and mismatch a distribution of uniformly generated weight vectors [64]. Hence, the problem instances with these properties may require a customized set of weight vectors for improving its performance.

We can also observe how these instances' properties produce different effects on the behavior of the EMO algorithms for the identified instances P2, P6, P7, and P10:

- In the case of P2, it was observed how most EMO algorithms obtain HVR values over 0.95 and compete



**FIGURE 5.** Sampled solutions for the different instances using 100,000 random configurations. The non-dominated solutions are coloured in red. The axis of the charts have been fixed for comparing the different shapes of the feasible region for each problem instance.



**FIGURE 6.** Approximation to the shape of the Pareto fronts using the aggregated Pareto fronts for each problem instance. Each front contains the overall non-dominated solutions obtained for each problem instance.

similarly, since MOMB2 is the only EMO that results dominated by the rest of EMO algorithms. Thus, the long tail shape of the P2 instance is not sufficient for eroding the behavior of the EMO algorithms, but this could be explained by reduced the number of variables considered by this instance (only 40).

- With respect to the P6 instance, we have seen that MOEA/D obtains the best HVR values, closely

followed by SMS-EMOA and NSGA-II. Despite that, the results of the statistical test pointed out that MOEA/D is dominated by SMS-EMOA and NSGA-II. A deeper analysis of the Pareto sets obtained by MOEA/D's in its individual runs reveals that for some of these runs MOEA/D performed poorly. This lack of consistency solving P6 explains why is dominated by SMS-EMOA and NSGA-II although

it obtains a better average HVR value across the 30 runs.

- It can be observed that the shape of the feasible region for P7 is not so long-tailed as other instances, but it is still remarkable. However, the results for this instance are comparable with those obtained for other regular instances, as MOEA/D is clearly the best performing algorithm (obtains the best HVR values with some margin and significantly outperforms the remaining EMOs). This suggests that the shape of the feasible region for P7 is not wide enough for eroding MOEA/D's behavior.
- In the case of P10, NSGA-II obtains the best HVR value, closely followed by SMS-EMOA and SPEA2. However, the results of the statistical tests for P10 showed that none of the EMO algorithms is able to significantly dominate more than two of the remaining algorithms. Similarly to the P2 instance, it could be argued that most EMO algorithms are performing similarly, but in this case the HVR values are sensibly lower for P10 than for P2. Because MOEA/D is the best performing algorithm for most of the instances in this study, we can argue that its behavior is more influenced by P10's properties than NSGA-II or SMS-EMOA.

## VI. FINAL REMARKS

In this paper we have conducted an exhaustive analysis of the performance of several EMO algorithms when calibrating multiple instances of an ABM for marketing jointly considering global awareness and WOM volume as its main outputs. Starting from an initial model instance built with real data, we have synthetically generated 14 additional model instances by changing the market characteristics to achieve a progressive dimensionality increase. Using this set of benchmarks, we have tested the calibration performance of seven EMO algorithms from different families and a classical mathematical method. We have analyzed the calibration results using both unary and binary performance indicators along with Wilcoxon ranksum test for assessing the significance of the results. In addition, we have used attainment surfaces for visually supporting the analysis of the performance indicators.

The results of our experimentation allow to provide the following insights:

- MOEA/D shows outstanding and robust behavior for our problem, being able to perform significantly better than the other EMO algorithms in most instances. Therefore, the decomposition-based strategy proposed by MOEA/D is clearly the best performing for the search space of the analyzed problem.
- We could also observe how the performance of MOEA/D was reduced when dealing with certain instances. A deeper analysis of the shape of the feasible region and the shape of the Pareto front of the instances revealed that some of them have certain characteristics

that can affect the performance of decomposition-based algorithms [64], [65].

- The calibration results on the high-dimensional instances have shown that its dimensionality erodes the performance of most of the algorithms. This is not the only characteristic of the instances causing this behavior in some of the algorithms. For instance, SPEA2 and NSGA-II outperform most of the remaining EMO algorithms for instances having less than 90 decision variables. However, their performance decays for the biggest instances and IBEA and GWASF-GA are able to outperform them.
- SMS-EMOA is the most robust EMO algorithm with respect to both the dimensionality of the instances and the shapes of their feasible region and Pareto front. This suggests that the strategy used by the SMS-EMOA, which combines hypervolume maximization and a replacement strategy targeting those solutions that poorly contribute to the hypervolume of its respective fronts, is effective when dealing with optimization problems having these characteristics.

## VII. PRACTICAL IMPLICATIONS AND FUTURE DIRECTIONS

In view of our results, we conclude as final practical implications that the calibration of similar ABM models (i.e., high-dimensional models using a set of historical data values that the model is intended to reproduce) can be improved by using either MOEA/D or SMS-EMOA for tuning their parameters. Although the performance of NSGA-II is competitive for some of our model instances, we encourage practitioners and modelers to go beyond the use of the most popular EMO algorithm.

Future work will be focused on evaluating the possible improvement of including qualitative pattern features, which could be useful for minimizing the loss of information produced by the fitness functions [66], [67]. Since fitness functions like those employed in our study mainly focus on the distance between series of points, the aggregation of these values can potentially loose the shape of the series in the process. This issue can be solved in multiple ways. For example, the current fitness functions could be modified or additional objectives related to each of the model's output could be incorporated. Further research should clarify which alternative produces the best results. Apart from the use of qualitative patterns, other ABM consumer models may require the calibration of additional key performance indicators, such as the calibration of sales. Calibrating more outputs could be approached by including them as additional objectives, which defines a new scenario where the use of many-objective EMO algorithms will be required. Besides we believe that surrogate fitness functions would be useful for future studies due to the high computational costs of simulating multiple times for every evaluation of a single model configuration [68].

## ACKNOWLEDGMENT

The authors would like to thank the “Centro de Servicios de Informática y Redes de Comunicaciones” (CSIRC), University of Granada, for providing the computing resources (Alhambra supercomputer).

## REFERENCES

- [1] E. Bonabeau, “Agent-based modeling: Methods and techniques for simulating human systems,” *Proc. Nat. Acad. Sci. USA*, vol. 99, no. 3, pp. 7280–7287, May 2002.
- [2] J. M. Epstein, *Generative Social Science: Studies in Agent-Based Computational Modeling*. Princeton, NJ, USA: Princeton Univ. Press, 2006.
- [3] C. M. Macal and M. J. North, “Tutorial on agent-based modeling and simulation,” in *Proc. Winter Simulation Conf.*, 2005, pp. 2–15.
- [4] M. A. Janssen and E. Ostrom, “Empirically based, agent-based models,” *Ecol. Soc.*, vol. 11, no. 2, p. 37, 2006.
- [5] J. D. Farmer and D. Foley, “The economy needs agent-based modelling,” *Nature*, vol. 460, no. 7256, pp. 685–686, Aug. 2009.
- [6] M. M. Waldrop, “Free agents,” *Science*, vol. 360, no. 6385, pp. 144–147, 2018.
- [7] W. Rand and R. T. Rust, “Agent-based modeling in marketing: Guidelines for rigor,” *Int. J. Res. Marketing*, vol. 28, no. 3, pp. 181–193, Sep. 2011.
- [8] M. Chica and W. Rand, “Building agent-based decision support systems for Word-of-Mouth programs: A freemium application,” *J. Marketing Res.*, vol. 54, no. 5, pp. 752–767, Oct. 2017.
- [9] R. G. Sargent, “Verification and validation of simulation models,” in *Proc. Winter Simulation Conf.*, 2005, pp. 130–143.
- [10] R. Oliva, “Model calibration as a testing strategy for system dynamics models,” *Eur. J. Oper. Res.*, vol. 151, no. 3, pp. 552–568, Dec. 2003.
- [11] M. Chica, J. Barranquero, T. Kajdanowicz, S. Damas, and Ó. Cordón, “Multimodal optimization: An effective framework for model calibration,” *Inf. Sci.*, vol. 375, pp. 79–97, Sep. 2017.
- [12] F. Stonedahl and W. Rand, “When does simulated data match real data? Comparing model calibration functions using genetic algorithms,” in *Advances in Computational Social Science (Agent-Based Social Systems)*, vol. 11. Tokyo, Japan: Springer, 2014, pp. 297–313.
- [13] C. A. Coello, G. B. Lamont, and D. A. Van Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*, vol. 5. New York, NY, USA: Springer, 2007. [Online]. Available: <https://www.springer.com/gp/book/9780387332543>
- [14] K. Deb, *Multi-Optimization Using Evolutionary Algorithms*. New York, NY, USA: Wiley, 2001.
- [15] M. Rong, D. Gong, Y. Zhang, Y. Jin, and W. Pedrycz, “Multidirectional prediction approach for dynamic multiobjective optimization problems,” *IEEE Trans. Cybern.*, vol. 49, no. 9, pp. 3362–3374, Sep. 2019.
- [16] D. Gong, B. Xu, Y. Zhang, Y. Guo, and S. Yang, “A similarity-based cooperative co-evolutionary algorithm for dynamic interval multiobjective optimization problems,” *IEEE Trans. Evol. Comput.*, vol. 24, no. 1, pp. 142–156, Feb. 2020.
- [17] M. Gong, H. Li, D. Meng, Q. Miao, and J. Liu, “Decomposition-based evolutionary multiobjective optimization to self-paced learning,” *IEEE Trans. Evol. Comput.*, vol. 23, no. 2, pp. 288–302, Apr. 2019.
- [18] S. M. Omran, W. H. El-Behaidy, and A. A. A. Youssif, “Decomposition based multi-objectives evolutionary algorithms challenges and circumvention,” in *Intelligent Computing*, K. Arai, S. Kapoor, and R. Bhatia, Eds. Cham, Switzerland: Springer, 2020, pp. 82–93.
- [19] A. Asilian Bidgoli, H. Ebrahimpour-Komleh, and S. Rahnamayan, “An evolutionary decomposition-based multi-objective feature selection for multi-label classification,” *PeerJ Comput. Sci.*, vol. 6, p. e261, Mar. 2020.
- [20] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
- [21] E. Zitzler, M. Laumanns, and L. Thiele, “SPEA2: Improving the strength Pareto evolutionary algorithm,” Zürich, Switzerland, TIK-Rep. 103, 2001.
- [22] E. Zitzler and S. Künzli, “Indicator-based selection in multiobjective search,” in *Parallel Problem Solving From Nature*. Berlin, Germany: Springer, 2004, pp. 832–842.
- [23] N. Beume, B. Naujoks, and M. Emmerich, “SMS-EMOA: Multiobjective selection based on dominated hypervolume,” *Eur. J. Oper. Res.*, vol. 181, no. 3, pp. 1653–1669, Sep. 2007.
- [24] H. Li and Q. Zhang, “Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II,” *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 284–302, Apr. 2009.
- [25] Q. Zhang and H. Li, “MOEA/D: A multiobjective evolutionary algorithm based on decomposition,” *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.
- [26] J. G. Falcón-Cardona and C. A. C. Coello, “Towards a more general many-objective evolutionary optimizer using multi-indicator density estimation,” in *Proc. Genetic Evol. Comput. Conf. Companion*. New York, NY, USA: Association for Computing Machinery, Jul. 2018, pp. 1890–1893.
- [27] S. S. Meghwani and M. Thakur, “Multi-criteria algorithms for portfolio optimization under practical constraints,” *Swarm Evol. Comput.*, vol. 37, pp. 104–125, Dec. 2017.
- [28] C. Zambrano-Vega, A. J. Nebro, J. García-Nieto, and J. F. Aldana-Montes, “Comparing multi-objective metaheuristics for solving a three-objective formulation of multiple sequence alignment,” *Prog. Artif. Intell.*, vol. 6, no. 3, pp. 195–210, Sep. 2017.
- [29] R. Hernández Gómez and C. A. Coello Coello, “Improved Metaheuristic based on the r2 indicator for many-objective optimization,” in *Proc. Annu. Conf. Genetic Evol. Comput.* New York, NY, USA: Association for Computing Machinery, Jul. 2015, pp. 679–686.
- [30] R. Saborido, A. B. Ruiz, and M. Luque, “Global WASF-GA: An evolutionary algorithm in multiobjective optimization to approximate the whole Pareto optimal front,” *Evol. Comput.*, vol. 25, no. 2, pp. 309–349, Jun. 2017.
- [31] J. A. Nelder and R. Mead, “A simplex method for function minimization,” *Comput. J.*, vol. 7, no. 4, pp. 308–313, Jan. 1965.
- [32] G. Eichfelder, *Adaptive Scalarization Methods in Multiobjective Optimization*. Berlin, Germany: Springer, 2008. [Online]. Available: <https://www.springer.com/gp/book/9783540791577>
- [33] G. Mavrotas, “Effective implementation of the  $\epsilon$ -constraint method in multi-objective mathematical programming problems,” *Appl. Math. Comput.*, vol. 213, no. 2, pp. 455–465, Jul. 2009.
- [34] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. da Fonseca, “Performance assessment of multiobjective optimizers: An analysis and review,” *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 117–132, Apr. 2003.
- [35] S. Farhadi, M. R. Nikoo, G. R. Rakhshandehroo, M. Akhbari, and M. R. Alizadeh, “An agent-based-Nash modeling framework for sustainable groundwater management: A case study,” *Agricult. Water Manage.*, vol. 177, pp. 348–358, Nov. 2016.
- [36] G. Narzisi, V. Mysore, and B. Mishra, “Multi-objective evolutionary optimization of agent-based models: An application to emergency response planning,” in *Proc. 2nd IASTED Int. Conf. Comput. Intell. (CI)*, 2006, pp. 224–230.
- [37] M. N. Read, K. Alden, L. M. Rose, and J. Timmis, “Automated multi-objective calibration of biological agent-based simulations,” *J. Roy. Soc. Interface*, vol. 13, no. 122, Sep. 2016, Art. no. 20160543.
- [38] S. Atamturktur, Z. Liu, S. Cogan, and H. Juang, “Calibration of imprecise and inaccurate numerical models considering fidelity and robustness: A multi-objective optimization-based approach,” *Struct. Multidisciplinary Optim.*, vol. 51, no. 3, pp. 659–671, Mar. 2015.
- [39] J. Guo, J. Zhou, J. Lu, Q. Zou, H. Zhang, and S. Bi, “Multi-objective optimization of empirical hydrological model for streamflow prediction,” *J. Hydrol.*, vol. 511, pp. 242–253, Apr. 2014.
- [40] W. Kurek and A. Ostfeld, “Multi-objective optimization of water quality, pumps operation, and storage sizing of water distribution systems,” *J. Environ. Manage.*, vol. 115, pp. 189–197, Jan. 2013.
- [41] Y. Liu, “Automatic calibration of a rainfall-runoff model using a fast and elitist multi-objective particle swarm algorithm,” *Expert Syst. Appl.*, vol. 36, no. 5, pp. 9533–9538, Jul. 2009.
- [42] Y. Liu and F. Sun, “Parameter estimation of a pressure swing adsorption model for air separation using multi-objective optimisation and support vector regression model,” *Expert Syst. Appl.*, vol. 40, no. 11, pp. 4496–4502, Sep. 2013.
- [43] D. Muraro and R. Dilão, “A parallel multi-objective optimization algorithm for the calibration of mathematical models,” *Swarm Evol. Comput.*, vol. 8, pp. 13–25, Feb. 2013.
- [44] Y. Zhang, Q. Shao, and J. A. Taylor, “A balanced calibration of water quantity and quality by multi-objective optimization for integrated water system model,” *J. Hydrol.*, vol. 538, pp. 802–816, Jul. 2016.
- [45] E. G. Bekele and J. W. Nicklow, “Multi-objective automatic calibration of SWAT using NSGA-II,” *J. Hydrol.*, vol. 341, nos. 3–4, pp. 165–176, Aug. 2007.

- [46] R. B. Confesor and G. W. Whittaker, "Automatic calibration of hydrologic models with multi-objective evolutionary algorithm and Pareto optimization," *J. Amer. Water Resour. Assoc.*, vol. 43, no. 4, pp. 981–989, 2007.
- [47] X. Zhang, R. Srinivasan, and M. V. Liew, "On the use of multi-algorithm, genetically adaptive multi-objective method for multi-site calibration of the SWAT model," *Hydrol. Processes*, vol. 24, no. 8, pp. 955–969, Apr. 2010.
- [48] X. Zhang, R. Srinivasan, and M. Van Liew, "Multi-site calibration of the SWAT model for hydrologic modeling," *Trans. ASABE*, vol. 51, no. 6, pp. 2039–2049, 2008.
- [49] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *Int. J. Forecasting*, vol. 22, no. 4, pp. 679–688, Oct. 2006.
- [50] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257–271, Nov. 1999.
- [51] I. Moya, M. Chica, and Ó. Córdón, "A multicriteria integral framework for agent-based model calibration using evolutionary multiobjective optimization and network-based visualization," *Decis. Support Syst.*, vol. 124, Sep. 2019, Art. no. 113111.
- [52] B. Libai, E. Müller, and R. Peres, "Decomposing the value of word-of-mouth seeding programs: Acceleration versus expansion," *J. Marketing Res.*, vol. 50, no. 2, pp. 161–176, Apr. 2013.
- [53] E. K. Macdonald and B. M. Sharp, "Brand awareness effects on consumer decision making for a common, repeat purchase product: A replication," *J. Bus. Res.*, vol. 48, no. 1, pp. 5–15, Apr. 2000.
- [54] F. Wu and B. A. Huberman, "Novelty and collective attention," *Proc. Nat. Acad. Sci. USA*, vol. 104, no. 45, pp. 17599–17601, 2007.
- [55] J. Yang and J. Leskovec, "Modeling information diffusion in implicit networks," in *Proc. IEEE Int. Conf. Data Mining*, Dec. 2010, pp. 599–608.
- [56] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, Oct. 1999.
- [57] M. Newman, A.-L. Barabási, and D. J. Watts, *The Structure and Dynamics of Networks*. Princeton, NJ, USA: Princeton Univ. Press, 2006.
- [58] I. Moya, M. Chica, J. L. Sáez-Lozano, and Ó. Córdón, "An agent-based model for understanding the influence of the 11-M terrorist attacks on the 2004 Spanish elections," *Knowl.-Based Syst.*, vol. 123, pp. 200–216, May 2017.
- [59] A. J. Nebro, J. J. Durillo, and M. Vergne, "Redesigning the jMetal multi-objective optimization framework," in *Proc. Companion Publication Annu. Conf. Genetic Evol. Comput.* New York, NY, USA: ACM, Jul. 2015, pp. 1093–1100.
- [60] C. M. Fonseca and P. J. Fleming, "On the performance assessment and comparison of stochastic multiobjective optimizers," in *Parallel Problem Solving from Nature—PPSN IV*, H.-M. Voigt, W. Ebeling, I. Rechenberg, and H.-P. Schwefel, Eds. Berlin, Germany: Springer, 1996, pp. 584–593.
- [61] L. Sanchez and J. Villar, "Obtaining transparent models of chaotic systems with multi-objective simulated annealing algorithms," *Inf. Sci.*, vol. 178, no. 4, pp. 952–970, Feb. 2008.
- [62] B. R. Campomanes-Álvarez, O. Córdón, and S. Damas, "Evolutionary multi-objective optimization for mesh simplification of 3D open models," *Integr. Comput.-Aided Eng.*, vol. 20, no. 4, pp. 375–390, Aug. 2013.
- [63] S. García, D. Molina, M. Lozano, and F. Herrera, "A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: A case study on the CEC'2005 special session on real parameter optimization," *J. Heuristics*, vol. 15, no. 6, pp. 617–644, May 2008.
- [64] H. Ishibuchi, Y. Setoguchi, H. Masuda, and Y. Nojima, "Performance of decomposition-based many-objective algorithms strongly depends on Pareto front shapes," *IEEE Trans. Evol. Comput.*, vol. 21, no. 2, pp. 169–190, Apr. 2017.
- [65] T. Matsumoto, N. Masuyama, Y. Nojima, and H. Ishibuchi, "Performance comparison of multiobjective evolutionary algorithms on problems with partially different properties from popular test suites," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2018, pp. 769–774.
- [66] G. Yücel and Y. Barlas, "Automated parameter specification in dynamic feedback models based on behavior pattern features," *Syst. Dyn. Rev.*, vol. 27, no. 2, pp. 195–215, Apr. 2011.
- [67] G. Yücel and Y. Barlas, "Pattern recognition for model testing, calibration, and behavior analysis," *Anal. Methods Dyn. Modelers*, vol. 27, pp. 173–206, Nov. 2015.
- [68] F. Lamperti, A. Roventini, and A. Sani, "Agent-based model calibration using machine learning surrogates," *J. Econ. Dyn. Control*, vol. 90, pp. 366–389, May 2018.



**IGNACIO MOYA** received the B.Sc. and M.Sc. degrees in computer science from the Complutense University of Madrid, Spain, in 2012 and 2013, respectively. From 2014 to 2016, he joined the European Centre for Soft Computing as a Research Assistant. He is currently pursuing the Ph.D. degree with the University of Granada, with a Scholarship granted by the Spanish Ministry of Economy. His research interests include agent-based modeling and social simulation, with special interest in model calibration and validation.



**MANUEL CHICA** Ph.D. degree in computer science and artificial intelligence in 2011 from the University of Granada. He is currently a "Ramón y Cajal" Senior Researcher at the University of Granada, a Conjoint Lecturer at the University of Newcastle, Australia, and the Chief A.I. Officer for a SME, ZIO, which applies computational intelligence and agent-based modeling to marketing. His current research interests include agent-based modeling, complex systems, evolutionary game

theory, machine learning, and multi-objective evolutionary optimization. He has published more than 90 scientific papers in peer-reviewed conferences and high-impact journals, such as IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, *Omega*, *California Management Review*, *Information Sciences*, IEEE TRANSACTIONS ON CYBERNETICS, *IEEE Computational Intelligence Magazine*, and the *Journal of Marketing Research* (totally and up to this date, 42 JCR-ranked papers, 33 of them being published in the first quartile). His work has been applied to a diverse range of applications, which include marketing, industrial engineering, social systems, climate change, and healthcare systems. Consequently, he is a Co-Inventor of an international patent, registered in the EU and U.S. and under exploitation; and has participated in more than 20 Research and Development Projects, where he played the role of a Principal Investigator in 12 of them.



**ÓSCAR CORDÓN** (Fellow, IEEE) was the Founder and a Leader of the Virtual Learning Center, University of Granada, from 2001 to 2005, and the Vice President of Digital University, from 2015 to 2019. From 2006 to 2011, he was one of the Founding Researcher with the European Centre for Soft Computing, being contracted as Distinguished Affiliated Researcher until December 2015. He is currently a Full Professor with the University of Granada. He has been,

for 25 years, an internationally recognized contributor to Research and Development Programs in fundamentals and real-world applications of computational intelligence. He has published more than 380 peer-reviewed scientific publications, including a research book on *Genetic Fuzzy Systems* (with 1400 citations in Google Scholar) and 111 JCR-SCI-indexed journal papers (67 in Q1 and 38 in D1), advised 19 Ph.D. dissertations, and coordinated 37 research projects and contracts (with an overall amount of > 9M€). From February 2021, his publications had received 5357 citations (H-index = 39), being included in the 1% of most-cited researchers in the world (source: Thomson's Web of Knowledge with 14625 citations and H-index = 58 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, the National Award on Computer Science ARITMEL by the Spanish Computer Science Scientific Society in 2014, and the IFSA Fellowship in 2019. He is currently or was an Associate Editor of 19 international journals. He was recognized as an Outstanding Associate Editor of IEEE TRANSACTIONS ON FUZZY SYSTEMS in 2008 and of IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION in 2019. Since 2004, he has taken many different representative positions with EUSFLAT and the IEEE Computational Intelligence Society.

...