



Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Multimodal optimization: An effective framework for model calibration



Manuel Chica^{a,*}, José Barranquero^b, Tomasz Kajdanowicz^c, Sergio Damas^d,
Óscar Cordon^e

^a IN3 - Department of Computer Science, Open University of Catalonia, Rambla del Poblenou 156, 08018 Barcelona, Spain

^b Novelti (novelti.io), 28050 Madrid, Spain

^c Faculty of Computer Systems and Management, Wroclaw University of Technology, 50-370 Wroclaw, Poland

^d Department of Software Engineering, University of Granada, 18071 Granada, Spain

^e DECSAI and CITIC-UGR, University of Granada, 18071 Granada, Spain

ARTICLE INFO

Article history:

Received 7 October 2015

Revised 12 September 2016

Accepted 19 September 2016

Available online 21 September 2016

Keywords:

Model calibration

Multimodal optimization

Genetic algorithms

Sensitivity analysis

ABSTRACT

Automated calibration is a crucial stage when validating non-linear dynamic systems. The modeler must control the calibration results and analyze parameter values in an iterative way. In many non-linear models, it is usual to find sets of configuration parameters that may obtain the same model fitting. In these cases, the modeler needs to understand the results' implications and run a sensitivity analysis to check the model validity. This paper presents a framework based on niching genetic algorithms to provide modeler with a set of alternative calibration solutions which also ease the analysis of their parameters, model's response, and sensitivity analysis. The framework is called MOMCA, an integral and interactive solution for model validation which facilitates the implication of decision makers. The core component of MOMCA is its niching genetic algorithm, able to reach various optima in multimodal optimization problems by keeping the necessary diversity. The proposed framework is applied to two different case studies. The first case study is a biological growth model and the second one is a managerial model to improve brand equity. Both applications show the benefits of the framework when providing a set of calibrated models and a way to analyze and perform sensitivity analysis based on the set of solutions.

© 2016 Elsevier Inc. All rights reserved.

1. Introduction

Non-linear dynamic models are widely used as they characterize real-world systems and the key relationships between their elements. These models are particularly suitable for systems with a high number of interrelated variables, where all relevant data to build the system is not always available or precise. They also provide a way to carry out simulations, understand the effects of alternative strategies, and assist stakeholders in better planning and management [67]. An example of these models is system dynamics which presents methodologies and tools for developing mathematical models of complex systems for social, biological, and economic problems [34,51,62,68].

* Corresponding author.

E-mail addresses: manuel.chica.serrano@gmail.com (M. Chica), jose.barranquero@novelti.io (J. Barranquero), tomasz.kajdanowicz@pwr.edu.pl (T. Kajdanowicz), sdamas@ugr.es (S. Damas), ocordon@decsai.ugr.es (Ó. Cordon).

A decisive phase when modeling non-linear dynamic systems is model validation [67]. The validation requires testing a set of hypothesis, the significance of their behavioral components (by assuming that the behavior is a consequence of the system structure), and the historical model fitting [49]. Validation is also measured in terms of degrees of confidence or quality and this quality is usually difficult to obtain for most non-linear simulation models in use [25]. The search for better validation procedures and methodologies is still necessary to ensure an appropriate level of confidence in the models' performance [6].

Automated calibration, mainly based on gradient search methods and genetic algorithms (GAs) [3], is a useful tool for model validation, but its results must be analyzed with caution [49]. The modeler needs to use automated calibration methods judiciously and in iterative and controlled way in order to manually filter the different alternatives. Otherwise, if modelers blindly accept the calibrated parameters without studying them, these values will be forced to match the historical behavior, with the subsequent risk of treating the model as a black box [58].

The presence of a multimodal nature of parameters is another problem while calibrating the model [46]. The existence of several sub-optimal solutions in a multimodal search space [28] causes difficulties to find a unique solution for the parameters. This is also known as “system identifiability” [4]. Modelers also need to study the parameters and outputs of the model as non-linear simulation models cannot be properly understood without exploring their behaviors under different parameter settings [37]. Input/output exploration, sensitivity analysis, and parameters' distribution visualization are the most valuable validation techniques to help to understand the model's behavior [37,49].

In this contribution, we propose a novel calibration framework based on handling the parameter space multimodality to help and support the modeler in an integral model validation process. Our multimodal calibration framework, called multimodal optimization for model calibration (MOMCA), can obtain a set of different and *acceptable* calibration solutions for the same model in a single run. The framework generates different parameter configurations which show the same or a similar model behavior. This archive of valid calibration solutions are used by MOMCA to perform automatic parameter analysis and run sensitivity analysis to provide additional indications on the model validity [59].

The use of niching genetic algorithms (NGAs) for the optimization process [28,54,61] is a key strength of the presented framework. These methods allow the framework to obtain multiple alternatives (calibration solutions) in a single run and to enhance the exploration of possible combinations of parameters [71]. The majority of the existing NGA-related studies tried to find a single best solution without getting stuck in local optima and the assessment is made in terms of the number of found optima from the known set of solutions [61]. However, our MOMCA approach takes advantage of NGAs not only to improve the search performance of standard GAs but to also offer the modeler a set of equally preferable calibration solutions in terms of fitness (model fitting).

Up to our knowledge, this work represents the first NGA application to model calibration. Additionally, the framework extends current state-of-the-art solutions by incorporating interaction methods that achieve the primary MOMCA objectives: to serve as an integral framework for a whole model validation process [49,56] and facilitate the implication of decision makers and stakeholders [6,67]. The proposed MOMCA framework is decomposed in three main stages: 1) an optimization algorithm based on NGAs, 2) an evaluation and interactive filtering process on the set of calibration solutions, and 3) an assisted sensitivity analysis and parameters' study based on quantitative and visualization tools for the same set of calibration solutions.

After presenting the framework we apply it to two non-linear models for different real-world case studies, disparate both in the application field and the used modeling methodology. In the first case, MOMCA is applied to a biological model based on the dynamic energy budget (DEB) theory [36,41]. DEB interrelates several physiological processes of individual organisms such as ingestion, assimilation, respiration, growth, and reproduction to simulate non-trivial biological processes. The calibration of a DEB model is complex because the observations of some parameters of the models are not directly measurable [24,57]. We calibrate a growth prediction model for blue mussels using empirical data from a Norwegian bay area by also analyzing the parameters and DEB model response.

The second case study uses system dynamics for modeling a brand value management problem [1]. The model simulates the evolution of the brand equity of a television show, the Indian television show *Who wants to be a millionaire?* [45]. In this second case, modelers are assisted through MOMCA to estimate the parameters of the effects between the branding variables of the case and to understand the dependencies between those parameters that better fit the interest level of the television show. Our experimentation also shows the generic nature of the MOMCA framework which allows its application to any non-linear simulation model.

The rest of the paper is structured as follows. Section 2 extends this work motivation as well as the benefits of the framework to tackle existing challenges in model calibration. Then, Section 3 explains the MOMCA framework and its components. The computational setup is established in Section 4 while Sections 5 and 6 describe both case studies with more details and report the results analysis and modeling implications. Finally, we present our concluding remarks in Section 7.

2. Open problems and motivation for the MOMCA framework

In this section, we review related literature, explore the main existing problems in non-linear models validation, and highlight how our proposal can help to face them. A summary of these open problems and MOMCA benefits is shown in the diagram in Fig. 1.

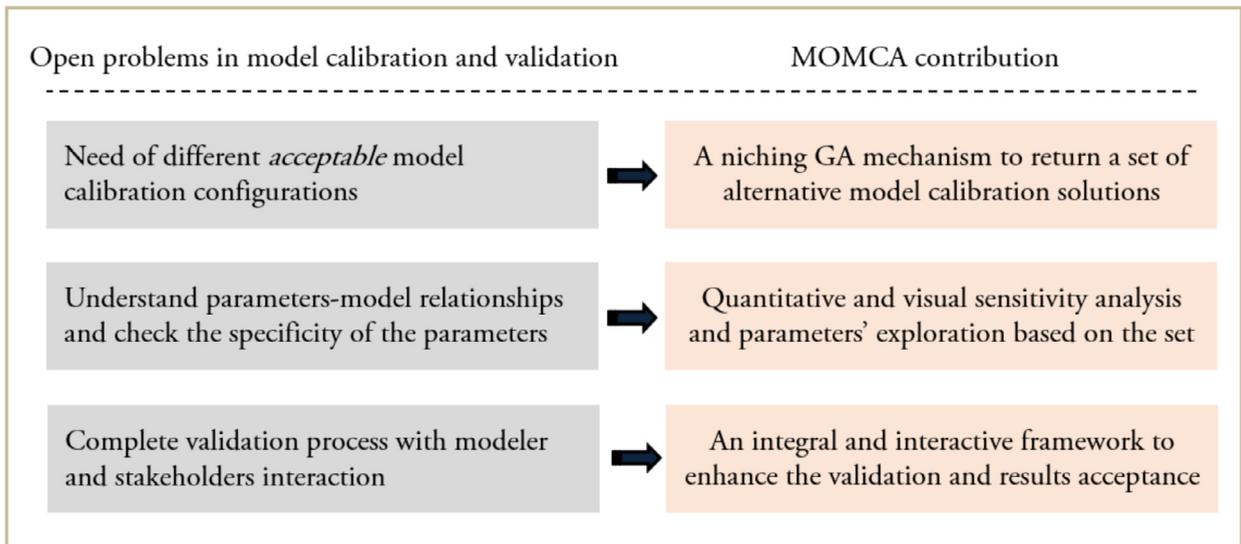


Fig. 1. Open problems and contributions of our framework to model validation and calibration.

2.1. Automated calibration methods

Model calibration is the process of estimating the parameters of a model to obtain a match between observed and simulated structures and behaviors [49,60]. The automated calibration of non-linear models is an optimization problem where the evaluation criterion is represented by an objective function (deviation function with respect to empirical data). In other words, it is a black-box optimization process [46] where model parameters are found by confronting the simulated outputs of the model with empirical data [49].

Among these automated methods, genetic algorithms (GAs) [3] are one of the most used optimization methods, already proved to be a useful tool for developing calibration and policy analysis in different kinds of non-linear models [21,30]. Miller [42] was the first to suggest the use of GAs for model calibration in classical system dynamics development. There is an extensive literature on the use of GAs and other metaheuristics to calibrate non-linear models from different application fields [9,29,31,39,71,72]. GAs show some additional advantages when automatically calibrating non-linear models as their capability to explore wider ranges of parameters and parameter settings (with a higher resolution) as well as to consider potentially non-linear interactions between those parameters [63,64].

Framework contribution: The existence of several near optimal answers demands automatically finding multiple solutions to the same calibration problem [70]. We take this assumption as a base to propose a calibration process in MOMCA where the modeler can examine and evaluate a set of different calibration alternatives.

Different parameter configurations, which show the same or very similar aggregated behavior, are thus obtained from one single run of the optimization method. Our framework uses NGAs, techniques able to find diverse existing optima with different decision variables [18,61], to obtain this set of calibration solutions with similar model behaviors. This is one of the main novelties of MOMCA. Unlike previous calibration methods based on standard GAs, we aim to exploit this NGA capability to partially or completely characterize the search space of the model calibration problem by obtaining the highest possible number of different optima. This phenomenon is called multi-solution-based efficacy [53].

2.2. Sensitivity analysis and parameter exploration

Sensitivity analysis reveals those parameters on which the behavior of the model highly depends. Generally, it is achieved by exploration of model's sensitivity to a particular parameter configuration and inputs [59]. Sensitivity analysis together with calibration are a key ingredient of the study of model's quality and crucial in deep validation of the model [49]. We say that the model is robust if the particular outputs are not dramatically affected by changing the parameter value, or if they are affected in a predictable way. In case the model is very sensitive to any parameter value, then it is necessary to investigate the underlying reason and analyze the impact on the model integrity.

The exploratory analysis of the input/output variables space of a model is also employed to strengthen trust in the model realism and to eliminate model factors that have negligible influence on the output variability. By analyzing the distribution of the model variables and parameters, the modeler can move forward to a simpler and easier to understand setting. The use of this exploration together with sensitivity analysis provides information on influential factors that significantly affect the variability of model's results and allow modelers to gain a deeper understanding of the complexity of the model, its uncertainties, interrelationships, and its potential future scenarios [38].

Thiele et al. [65] showed the powerful use of calibration methods to induce model's responses departing far from the empirical, historical values, while constraining the search to a limited parameters range. Also, researchers have used GAs to search for parameters that yield specific emergent behaviors and sensitivity analysis [64].

Framework contribution: The above-mentioned approaches reported in literature performed their analysis by an additional experimentation or by sacrificing their calibration results to obtain the outputs' response. MOMCA promotes the automatic analysis of a set of alternative calibration configurations to discover hidden properties (or relationships) between the model design and its structure. This process is called *innovization process*. A set of trade-off optimal or near-optimal solutions, found using metaheuristics such as GAs, are analyzed to uncover useful relationships among problem entities. It thus provides a better understanding of the problem to a designer or a practitioner [18,19].

The filtering process of MOMCA allows the modeler to adjust the cardinality of the set of returned solutions as well as the similarity among those solutions to better focus on the posterior analysis. As stated by Ligmann-Zielinska et al. [38], a distribution of the model inputs and outputs, including the tails and means, provides an opportunity for exploration of extreme system behavior. MOMCA can automatically generate parameter distributions and measures to quantitatively describe them. Last but not least, all the information from sensitivity analysis and parameter exploration is performed without any additional computational effort from the set of alternatives in a single algorithm run.

2.3. Integral model validation

The validation and testing of any model or decision support system is a decisive step for ensuring its managerial adoption [14]. Decision makers are all rightly concerned whether results of each model are *correct* [49,60,63]. However, the validation of non-linear models such as system dynamics and their effectiveness for real-world problems is not straightforward. The validation process can be seen as a learning process where the modeler's understanding is enhanced through her/his interaction with the formal and mental model [43]. As this process evolves, both the formal and the mental perceptions of the modelers change, leading to a successive approximation of the formal model to reality.

Model calibration is only a step within model validation and should be considered as part of the model building process. For instance, Qudrat-Ullah [56] considered the calibration of a model as another structural validity test for checking the behavioral validity of the model. Sensitivity analysis, extreme cases, and the exploration of input/output variables are also important assets to model validation [49] but are not sufficient, in isolation, to guarantee the confidence and adoption of the model's results. Additionally, the utility and effectiveness of many non-linear models and their outputs are often judged by stakeholders and decision makers. Therefore, it is highly recommended to correctly perform the validation of the models [6,67].

Framework contribution: We propose an integral methodology where different steps within traditional model validation (i.e., calibration, sensitivity analysis, parameters exploration) are included and automated under the same framework. MOMCA can assist modelers in obtaining, analyzing, and interpreting information about their models. The three steps of the framework provide a way to better understand the models, follow a data-driven validation, and promote the participation of stakeholders.

The main MOMCA objective is to offer an enriched model validation framework to assist the modeler with more information about the problem. The MOMCA framework is an interactive framework and employs both quantitative and qualitative information because of the existing strong subjective and qualitative considerations when accepting and adopting a model [6]. Finally, returning a set of alternative calibration solutions as well as having an integrated parameter and sensitivity analysis will support the deployment of policy scenarios and will increase the confidence on the model's outputs.

3. Description of the MOMCA framework

In this section we describe the proposed framework and its main three stages (Sections 3.1, 3.2, and 3.3). Fig. 2 shows a diagram which summarizes the MOMCA framework, their inputs and outputs as well as its iterative validation loop.

3.1. Searching for a set of multiple calibration solutions by niching genetic algorithms

NGAs are multimodal optimization algorithms based on evolutionary computation [3]. These algorithms follow principles inspired in nature, keeping the necessary diversity to achieve a wide search in different promising areas, allowing the method to reach different optima. The advantage of most of the NGAs is that their population-based structure allows them to obtain multiple different solutions in a single run. Since the first pioneering NGA based on fitness sharing in 1987 [28], a significant number of algorithms have been proposed [61]. NGAs have been widely used in real-world optimization problems. For instance, Elsheikh et al. [22] applied them for the parameter estimation of flow models in an iterative stochastic ensemble method. de Magalhães et al. [40] used them for structure-based drug design, Will et al. [69] for a problem of estimation of solar radiation, and Aguilera et al. [2] for multi-classification systems, among other diverse application fields.

In the next sections we first describe the clearing method (Section 3.1.1) and later, the GA components and design used in the experimentation of the paper (from Sections 3.1.2 to 3.1.5). Additionally, reader can find the source code of the NGAs for calibration at <http://sci2s.ugr.es/soccer/mcalibration.html>.



Modeler's knowledge



Empirical data and observations

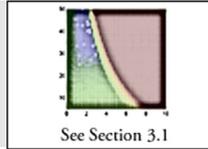


Non-linear model (software simulation)

MOMCA: Multimodal Optimization for Model CALibration

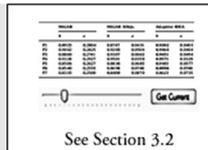
1. Searching for a set of calibration alternatives

Use of niching genetic algorithms to automatically search for a set of model calibrations with good fitting and diversity.



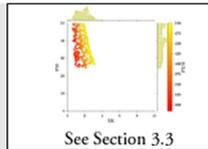
2. Evaluating performance and interactive filtering of the set

Evaluate the model calibration performance of the solutions and filter the set of solutions with respect to its fitting and diversity.



3. Automated quantitative and visual sensitivity analysis

Analyze and visualize the sensitivity and robustness of the parameters from the whole set of returned solutions.



Iterative loop for model validation with stakeholders' implication



Re-design the model from MOMCA insights, if needed



More confidence on a validated model.
Better understanding of the dynamics

Fig. 2. Diagram to illustrate the operation of the MOMCA framework and its three steps within the model validation environment.

3.1.1. Clearing as the niching method

We consider one of the most well-known NGA method: a standard generational GA [3] with a clearing method [54]. Clearing is a classical NGA that shows good trade-off between complexity and performance [52,61]. We follow the original proposal by Pétrowski [54] which is based on mimicking the concept of limited resources of the environment. This process allows the biodiversity generation and diminishes the competition among the individuals of the different species permitting the cohabitation in the same area.

The clearing process is performed after the evaluation and before the selection of the GA. First, the chromosomes of the population are sorted according to their fitness values, from the best to the worst chromosome. Later, the method removes those chromosomes showing worse fitness than the κ better chromosomes (called capacity of the niche) and within the clearing radius distance (R). Therefore, when the clearing process is over, the closer solutions to the best κ solutions are cleared and the remaining are considered as parents to generate offspring for the next generation.

Let us remark that the goal of this paper is to present and apply the proposed MOMCA framework to solve two disparate real case studies (presented in Sections 5 and 6). Therefore, we presented a study for several clearing configurations to present the usefulness of the framework omitting a comprehensive NGAs comparison for each case, which is out of the scope of this paper.

3.1.2. Representation scheme

The nature of the parameters to be calibrated in a model can be diverse. A modeler can choose integer (discrete-valued), real, or binary parameters for the calibration process even for the same non-linear model. There are many proposals where

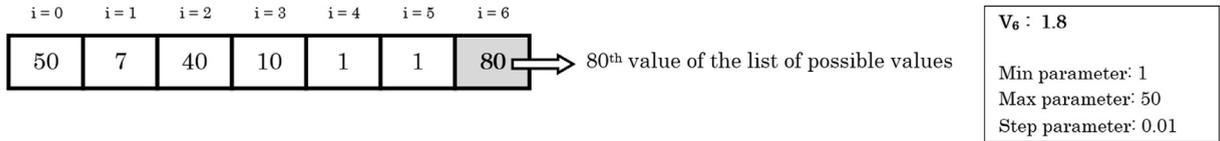


Fig. 3. Integer-based representation of the chromosomes of the NGA. For each gene, taking into account the nature of its corresponding calibration parameter, the NGA initially generates a list of valid values and avoids unfeasible genes. Note, that gene values are sorted indices of the valid values of the corresponding calibration parameter.

integer and binary variables are first encoded to real-form and the decoded back after the optimization process. However, there is a fair possibility of losing vital information on the mapping between different spaces [17]. Additionally, the modeler normally knows the granularity of these parameters, their maximum, and minimum values. For instance, one can set the minimum step of the parameter that is valid according to measurement characteristics or practical implications.

Therefore, we adopt a design and representation scheme that allow this generality and facilitate the injection of modelers' knowledge about the parameters to be calibrated. Particularly, the NGA is designed by using an integer representation and the framework includes three specification input variables for each model parameter i to be calibrated: a) minimum parameter value (min_i), b) maximum parameter value (max_i), and c) the step ($step_i$), that defines the granularity of parameter i within its range. Based on these three inputs, the algorithm can *a priori* calculate the total number of possible values for each parameter i (n_i), given by Eq. 1.

$$n_i = \left\lfloor \frac{max_i - min_i}{step_i} \right\rfloor + 1. \quad (1)$$

The chromosomes of the population have a fixed length and are the subject to the interval-based constraints of the decision variables (one integer gene for each calibration parameter). Every possible feasible value of parameter i , called v_i , is defined as $v_i = min_i + k \cdot step_i, \forall k \in \{0, \dots, n_i\}$. Fig. 3 shows a diagram with the considered representation for the chromosome. The NGA can then be generally based on this framework design and integer-based representation by allowing both integer/discrete and numerical calibration parameters.¹

3.1.3. Fitness function

We use a single-objective fitness function which measures the distance between the model's outputs and the available empirical data (model fitting). In order to calculate the output of the model we need to simulate the dynamic system with the parameters of the corresponding chromosome. There are different error deviation measurements in the literature for this comparison purposes. However, each type of model and its output variables might have specific characteristics to select the most appropriate measurement [10].

3.1.4. Initialization and breeding processes

The initialization process creates a population of chromosomes by generating feasible values for their genes. This process uses a uniformly distributed random procedure to select the value for each gene. The first step of the initialization of the gene is to create a list of possible values according the mechanism described in Section 3.1.2. Later on, the algorithm selects one of them at random. We use a generational GA approach with a tournament selection mechanism. Additionally, the design of the GA has weak elitism: the best parent is always preserved at every generation.

3.1.5. Crossover and mutation operators

Our designed crossover operator generates two offspring by crossing two parents gene by gene according to a probability p_c . We use a modified version of the BLX- α crossover [23,32] where its general mechanism is adapted for an integer representation scheme. The crossover operator truncates the selected values over the gene set of feasible values after randomly selecting the new value from interval $[c_{min} - I\alpha, c_{max} + I\alpha]$, where $c_{max} = \max(v_i^1, v_i^2)$, $c_{min} = \min(v_i^1, v_i^2)$ and $I = c_{max} - c_{min}$. We note v_i^1, v_i^2 as the feasible decoded values from the genes of the parents. Variable α is the exploration parameter of the operator. When α is set to 0, BLX- α is equivalent to the flat crossover. Fig. 4 shows an example of the application of the crossover operator.

Additionally, the mutation operator mutates the genes of the chromosomes of the population by a probability p_m (per gene). If a gene is to be mutated, the operator generates a new value for it by running a uniform random distribution over the feasible values of its corresponding model parameter.

¹ If needed, a real coding can be used for higher numerical precision. However, the selected representation presents a better trade-off among precision, convergence time, and model calibration quality.

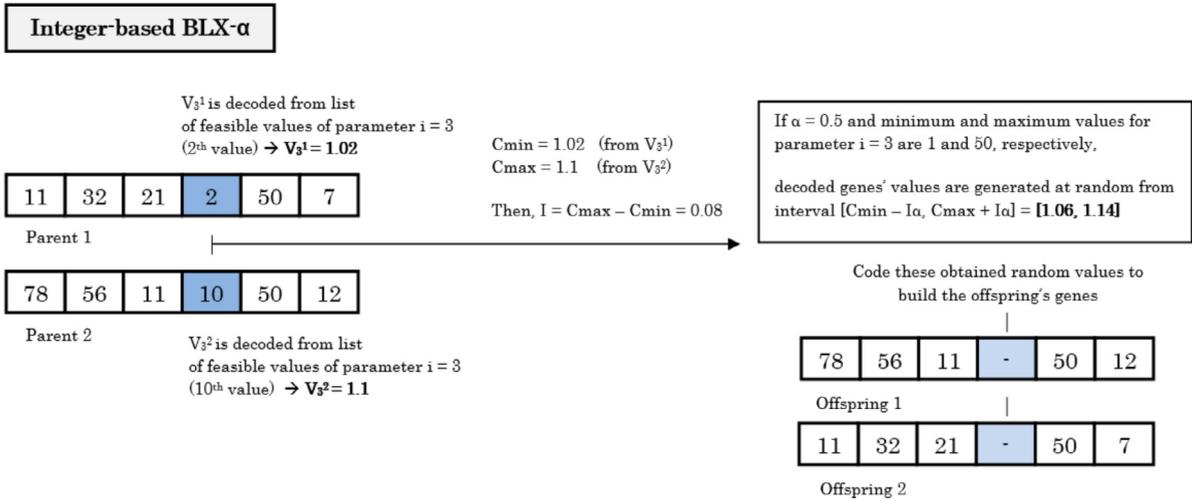


Fig. 4. Diagram to illustrate the generation of offspring genes from two parents by using our modified version of the BLX- α .

3.2. Evaluating calibration performance and filtering of the solution set

When the NGA finishes its optimization process, the modeler can inspect, evaluate, and decide the best calibration solutions for the model (i.e., the alternative parameter configurations obtained). Three different viewpoints are considered for the performance assessment of the results [53]: efficacy (calibration error), multi-solution based efficacy (capability to find multiple optima), and diversity in the final set of parameter configurations. This performance assessment checks if the calibration algorithm explores well when optima belong to different areas and exploits well an area where there are similar optimal solutions [52].

We define a set of quality indicators to measure the calibration performance of the returned set of calibration solutions. These metrics include the cardinality of the set of calibration solutions (number of solutions with different parameters' values of the set), the best fitness (model fitting) value of the set, differences between the fitness values of all the solutions of the set, and a measure of the diversity of the found solutions (e.g. average distance among the calibrated parameters of the solutions). Also, we use visualization tools such as heat-maps, where the color represents fitness values, to visualize the parameters search space and place the set of solutions returned by the NGAs on these heat-maps. Modelers can easily observe distances between solutions as well as possible clusters or unexplored areas and their position with respect to the fitness space (see Sections 5.2 and 6.2 of our experimentation).

Additionally to the presentation of the calibration performance to the modeler, our framework includes an interactive *a priori* and *a posteriori* processing based on filtering solutions by their fitness values (model fitting) or by the similarity of the parameter configurations. This interaction with the modeler permits the framework to show a greater or lesser number of calibration solutions when the NGA finishes. Stakeholders might also participate during this stage of the model validation to first, improve the model (allowing a more balanced view of the management issue) and second, to learn about the dynamic of their system [67]. The goal is filtering the returned NGA solutions with respect to two main criteria:

- (a) Fitness filtering: Modelers can filter the set of solutions by fitness in an *a priori* or *a posteriori* way (before or after the running of the NGA). This filtering is set by defining a minimum fitness threshold (min_{fit}) and/or a maximum fitness threshold (max_{fit}).
- (b) Similarity filtering: This *a posteriori* filtering reduces the number of solutions dynamically in terms of parameters' values distances among the set of calibration solutions. A similarity parameter σ_{fit} is used for this filtering based on the parameters' values distances.

By setting these thresholds, modelers can decide the focus of their calibration and better inspect the set of calibration solutions, which can be unaffordable for some models. In addition to them, the interactive filtering process is also related with the niching mechanisms of the algorithms. These mechanisms must also be controlled by the modeler to adjust the NGA behavior and obtain the desired final set of solutions with different features (i.e., cardinality and diversity of the solutions with respect to fitting and parameters' values). For instance, in the case of using a clearing method, niche capacity κ and radius R control the number of niches (clusters of calibration solutions with similar parameters' values) [28] and how similar the calibration solutions of the niches are. Sections 5.2 and 6.2 of the experimentation present a further discussion on the results of the filtering and NGA variants.

3.3. Automated quantitative and qualitative analysis on the solutions

One of the main advantages of MOMCA is its use of the set of model calibration alternatives to easily analyze and find relationships among the parameters and variables of the model. Therefore, we propose in this MOMCA step the use of quantitative measures and visualization tools to analyze the set of solutions in order to better explain and understand the dynamics.

The information of the parameters' distribution and their robustness is obtained when analyzing the set of calibration alternatives with equal fitting but different parameter values. First, a *pdf* is computed (i.e., histogram built from the frequency values of the parameter) for each parameter of the model by considering all the calibration solutions of the set. Histograms or box-plots can be used for observing the parameters' distribution of this set. These visualization techniques are focused on representing the solution set in terms of parameter configurations. Quantitative indicators are then computed from the *pdf* to numerically summarize each model parameter [6]:

- Median value: Unbiased calculation of *middle* value to define the central value of the parameter distribution.
- Spread: Range of the parameter (i.e., their maximum minus minimum values) that achieves good fitting models. It may be heavily affected by outliers.
- Kurtosis (k) [20]: Also known as the fourth standardized moment, it is an indicator of how peaked the data distribution is. A high kurtosis value indicates that the distribution has a sharp peak with long and fat tails.
- Skewness [33]: It measures the asymmetry of the distribution and some modality characteristics. A negative skew (left) has fewer low values and a positive skew (right) has fewer large values.
- Bimodality coefficient (BC) [26]: The computation of the bimodality coefficient can help, together with other measures such as the skewness, to measure the multimodality of the parameter distribution [55]. Eq. 2 shows its calculation by using the sample size of the distribution (defined by n), excess kurtosis (k), and skewness. Empirical values of $BC > 0.555$ can be taken to indicate bimodality: higher numbers point toward multimodality whereas lower numbers point toward unimodality.

$$BC = \frac{\text{skewness}^2 + 1}{k + \frac{3(n-1)^2}{(n-2)(n-3)}}. \quad (2)$$

The visualization of the parameters with respect to other input/output variables are an integral part of the sensitivity analysis [37]. MOMCA automated sensitivity analysis permits a comparison of the dependencies found from the set of solutions. For instance, scatter-plots directly and simply show relationships and dependencies between combinations of parameters and variables. As shown by Lee et al. [37], scatter-plots are especially recommended when dependencies are scalar and the number of enumerations between parameters and variables is manageable. Additionally, global sensitivity analysis methods such as the variance-based sensitivity analysis can be used to quantitatively measure the effects of some parameters and the outcomes [38,59]. For instance, the variance-based analysis calculates the partial variance of each parameter with respect to others for a given outcome.

4. Experimental setup

In the experimentation of the paper we compare our MOMCA framework based on NGAs with two traditional calibration algorithms. The first one is the Nelder–Mead simplex [47,50]. The second one is a standard generational GA [3]. The Nelder–Mead simplex is a classical optimizer previously used in different model calibration experiments [27] due to its ability to work under low restrictive conditions (e.g., it does not require the objective function to be differentiable). Also, it was used by Filgueira et al. [24] and Rosland et al. [57] to calibrate the models of our first case study. The standard GA follows the same design as the clearing NGA described in Section 3.1 but without its niching features (i.e. it is not a multimodal approach). However, the standard GA is also able to provide a set of calibration solutions because it is a population-based method and therefore, we can analyze their results with respect to the multimodal framework.

We run each GA 30 times with different random seeds given the stochastic nature of the considered GAs. All the algorithms were launched in the same computer: Intel Core i5 2400 (3.10 Ghz), 4 Gbytes of memory, and Windows 7 Professional (64-bits). We also use the same programming language (C++) for the development of the algorithms. Simile [44] was used as the software tool for modeling the non-linear model of the first case study, the biological growth model, called by the fitness function of the different calibration algorithms to evaluate the model's behavior with respect to the historical data of the model. The source code of the algorithms can be found at <http://sci2s.ugr.es/soccer/mcalibration.html>.

Table 1 shows the considered parameter values for all the launched algorithms. Preliminary experiments suggested the values presented in the table as appropriate values for the population size, operators' probabilities, and α parameter of the crossover operator. One important decision was to set the stopping criterion of the algorithms. We chose the number of evaluations (total number of runs of the non-linear model) because other stopping criteria such as execution time or number of generations are more algorithm-dependent and we could not fairly compare the different types of algorithms of the experimentation. For instance, the Nelder–Mead simplex is not a population-based algorithm. We set the maximum number of evaluations to 50,000 model simulations (fitness function calls).

Table 1
Parameter values of the calibration algorithms.

Parameter	Value
Common GA design	
Number of runs (different seeds)	30
Stopping criterion (evaluations)	50,000
Population size	500
Crossover probability (p_c)	0.8
Mutation probability per gen (p_m)	0.05
k-tournament selection process	3
α for the BLX operator	0.5
Clearing mechanism	
Capacity of the niche (κ)	1, 3, 7
Clearing radius for the same niche (R)	5, 10
Specific parameters of the first case (biological growth model)	
Fitness filtering (min. fitness value min_{fit})	13
Similarity filtering (min. Euclidean distance σ_{fit})	0.05
Specific parameters of the second case (brand management model of a television show)	
Fitness filtering (min. fitness value min_{fit})	9
Similarity filtering (min. Euclidean distance σ_{fit})	0.05

5. First case study: a biological growth model

5.1. Model description

The growth of bivalve species is widely studied due to their role in aquaculture and other ecosystem services such as water filtration. The need to make growth predictions encourages the creation of modeling tools to carry out prospective studies, and has promoted the development of individual bivalve growth models. Dynamic energy budget (DEB) is a non-linear individual-based modeling and one of the main approaches applied to model bivalve growth [35,57].

DEB theory [35] describes the individual in terms of three state variables: reserve(s), structure(s), and maturity/reproduction. The energy assimilated from food is stored as reserves. A fixed fraction of this energy (κ) is directed towards maintenance and growth of the structural body, and the remainder ($1 - \kappa$) is directed towards maturation, gamete production, and/or maintenance of the reproductive system depending on the life cycle stage of the organism. In the DEB theory, the description of these energetic processes in an organism is a function of its state and the environment [48].

Filgueira et al. [24] proposed various DEB models applied to the growth prediction in culture areas of blue mussels. In this experimentation we use our MOMCA framework to help modelers with the DEB model validation which was proposed in Filgueira et al. [24]. Briefly, the description of the model and main modeler's problems when validating it are summarized below:

- **Calibration parameters:** Three real-value parameters, highlighted by Rosland et al. [57] to be auto-calibrated, are defined as the main parameters to calibrate. These parameters are the *half-saturation coefficient* (X_k), the *maximum energy ingestion rate* ($\{\dot{p}_{X_m}\}$), and the *specific somatic maintenance rate* ($[\dot{p}_M]$). Additionally, the model has a higher number of parameters which are fixed by modelers. See the complete model diagram in Fig. 5 where the three calibration parameters are framed by red bounding boxes. All the parameters of the model are mathematically defined in Rosland et al. [57] and Filgueira et al. [24]. We refer the reader to Rosland et al. [57] and Filgueira et al. [24] for further details about DEB and the specific model.
- **Modeler's knowledge about the parameters:** Practical ranges of the three parameters are $X_k \in [0, 10]$, $\{\dot{p}_{X_m}\} \in [50, 500]$, and $[\dot{p}_M] \in [5, 50]$. As suggested by authors of Filgueira et al. [24], the granularity of the three parameters is defined by the following numerical steps: $step_{X_k} = 0.01$, $step_{\{\dot{p}_{X_m}\}} = 0.01$, and $step_{[\dot{p}_M]} = 0.1$.
- **Output of interest:** The model has two main output variables: the weight and the length of the mussel shell. We use an empirical data-set to calculate the deviation of the simulated outputs with respect to the real growth of the mussels. The data-set is obtained from *in situ* experiments from commercial mussel farms in southern Norway (Flødevigen, from March to November, 2007).
- **Deviation measure:** The mean absolute relative error (MARE) is considered as the deviation measure because of the need of aggregating errors from the two output variables (shell weight and length) on different scales. This is the method previously used with DEB models in Filgueira et al. [24]. Eq. 3 defines MARE for these two output variables.

$$MARE = \frac{1}{N} \sum_{i=1}^N \left(\frac{|V_M(i) - V_E(i)|}{V_E(i)} + \frac{|V'_M(i) - V'_E(i)|}{V'_E(i)} \right), \quad (3)$$

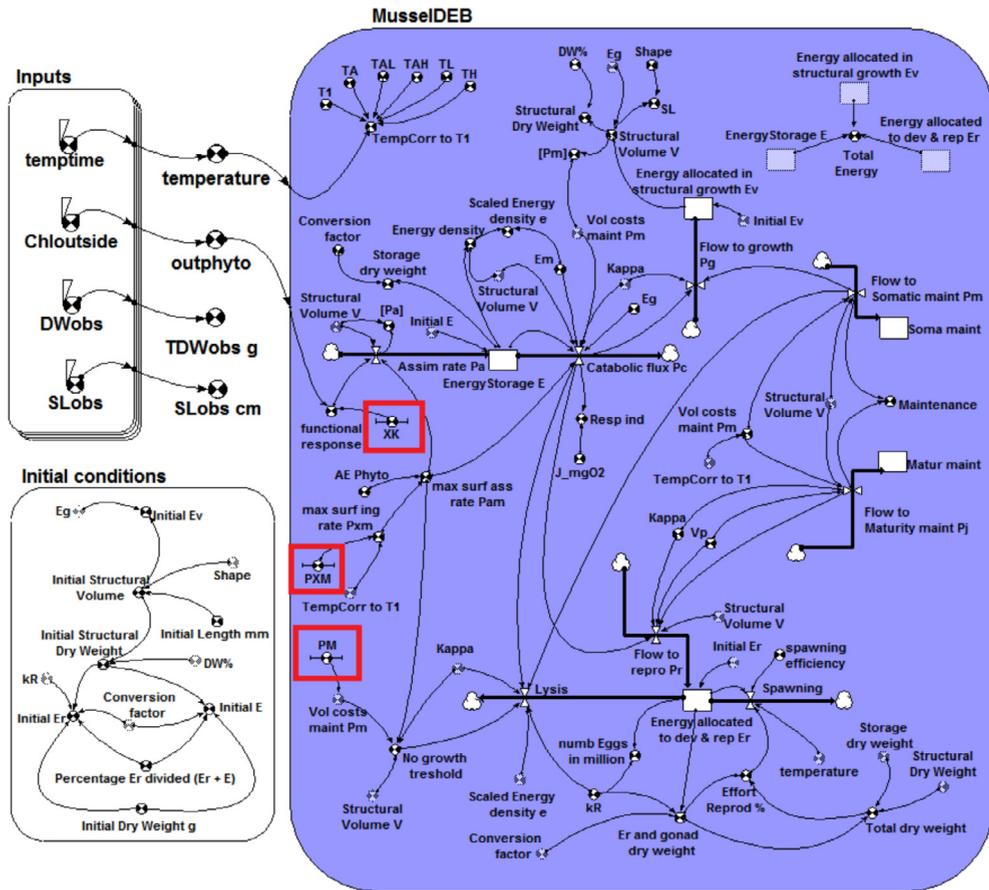


Fig. 5. Model structure for the DEB case study by using the Simile system dynamics tool [44]. We have highlighted the three calibration parameters of the model by three red bounding boxes. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

where N is the number of steps of the model (e.g. days of the historical evolution); and V_M, V'_M and V_E, V'_E are the two output vectors of the simulated model and the empirical data, respectively.

- **Modeler's objective and concerns:** Filgueira et al. [24] already pointed out that the parameter calibration of a given DEB model for different species is one of the challenges in DEB modeling. In previous studies, DEB models were calibrated using a simplex optimization method [24,57]. However, these authors noted that, even when achieving low deviation from historical data, some estimated X_k values were unrealistic, extremely high, and not sensitive. Some X_k values also generated a negative reproductive buffer in the DEB model. Main concerns and challenges described in Filgueira et al. [24] were to check the robustness and sensitivity of the three calibration parameters rather than obtaining a higher model's output accuracy. DEB modelers also want to verify if different ranges of parameters' values can obtain similar good model behavior as already suggested by Filgueira et al. [24].

5.2. Calibration performance of the niching genetic algorithms

Table 2 shows the results of applying the MOMCA framework and the baseline algorithms to this first calibration problem. The table contains the mean (\bar{x}) and standard deviation (σ) of 30 runs of the algorithms. Results for the Nelder–Mead algorithm are obtained by running it 30 different times and changing its initial solution (i.e., initial parameters' values of the calibration problem). The values in columns are organized in following order: number of solutions of the returned set, the best calibration solution, the averaged fitness, and the averaged distance among the set of solutions. The average distance is obtained by calculating the average Euclidean distance of the three calibration parameters (with min-max normalization to $[0, 1]$) for all the possible pairs of solutions of the set.

The final sets of calibrated solutions were filtered by fitness not to have a fitness function value higher than 13 which is considered by the modelers of this case as the minimum acceptable fitting value (fitness filtering min_{filt}). Table 2 also contains two versions of each NGA variant. The first one does not filter the set of final calibration solutions while the second version includes the similarity filter σ_{filt} to clean similar solutions within the final set of solutions. This parameter

Table 2

Calibration results of the baseline and NGA methods for the DEB model in 30 independent runs of the algorithms.

Calibration algorithm	Number of returned solutions	Best solution fitness	Average fitness all solutions	Average distance among solutions
$\bar{x}(\sigma)$ statistics for baseline methods				
Nelder–Mead	1 (0)	12.85 (0.45)	12.85 (0.45)	0 (0)
Standard GA	135.77 (88.89)	12.65 (0)	12.78 (0.02)	0.18 (0.04)
$\bar{x}(\sigma)$ statistics for clearing $\kappa = 1$				
Niche radius $R = 10$	14.47 (3.18)	12.65 (0)	12.78 (0.02)	0.26 (0.03)
Filtered, niche radius $R = 10$	9.13 (2.08)	12.65 (0)	12.77 (0.02)	0.28 (0.02)
Niche radius $R = 5$	14.3 (4.01)	12.65 (0)	12.78 (0.03)	0.23 (0.04)
Filtered, niche radius $R = 5$	9.6 (2.13)	12.65 (0)	12.78 (0.03)	0.26 (0.02)
$\bar{x}(\sigma)$ statistics for clearing $\kappa = 3$				
Niche radius $R = 10$	15.07 (3.33)	12.65 (0)	12.79 (0.02)	0.24 (0.03)
Filtered, niche radius $R = 10$	9.13 (1.98)	12.65 (0)	12.78 (0.03)	0.27 (0.02)
Niche radius $R = 5$	15.43 (4.03)	12.65 (0)	12.8 (0.03)	0.22 (0.02)
Filtered, niche radius $R = 5$	9.10 (1.94)	12.65 (0)	12.79 (0.03)	0.24 (0.02)
$\bar{x}(\sigma)$ statistics for clearing $\kappa = 7$				
Niche radius $R = 10$	16.9 (3.82)	12.65 (0)	12.80 (0.03)	0.23 (0.03)
Filtered, niche radius $R = 10$	9.6 (2.13)	12.65 (0)	12.78 (0.03)	0.26 (0.02)
Niche radius $R = 5$	16.07 (5.3)	12.65 (0)	12.8 (0.02)	0.21 (0.02)
Filtered, niche radius $R = 5$	9.9 (1.75)	12.65 (0)	12.78 (0.02)	0.24 (0.02)

σ_{flt} is set to 0.05 which is the minimum Euclidean distance among the normalized parameter values of a solution to be considered as a final solution.

We analyze the performance results of the calibration algorithms in Table 2 from two points of view: the fitting results (measured by the fitness) and the diversity and cardinality of the solutions set. The most significant results are the following:

- Calibration solutions found by the GA design (both standard and all the NGA variants) are better than those obtained by the Nelder–Mead algorithm. The Nelder–Mead algorithm has no diversity in its final set of solutions (i.e. the cardinality of the set of solutions is always 1).
- The NGA variant with a clearing mechanism of $\kappa = 1$ and $R = 10$ is the best performing algorithm in terms of averaged fitness of the solutions (12.77 and 12.78) and good diversity in the set of calibration solutions (average distances of 0.26 and 0.28).
- The NGA variants and the GA always, i.e., in all the 30 runs ($\sigma = 0$), obtain a solution with a fitness fitting value of 12.65.
- The standard GA returns a set of solutions with high cardinality (135.77 averaged number of solutions in the 30 runs). However, the standard GA solutions are more similar among them (averaged Euclidean distance of 0.18) than those returned by all the NGA variants.

We can see that differences between NGA variants are low both in fitting performance and diversity within the set of calibration solutions. In fact, the behavior of the clearing mechanisms are coherent. The number of returned solutions (cardinality) increases when clearing uses a higher capacity value κ (with and without filtering). When clearing makes use of a higher niche radius R , the clearing process keeps less niches and therefore, the similarity of the solutions decreases (higher averaged distance among them). On the other hand, the results of Table 2 show that the number of returned solutions is not affected by changes on the clearing niche radius R . As expected, the filtered variants of the NGAs have a lower number of solutions in the final set of calibration solutions and their averaged distance is higher after applying the filtering.

Finally, we visualize, in Fig. 6, the solutions found by the best performing NGA (clearing $\kappa = 1$ and $R = 10$, with and without filtering). A grid search was used to systematically sample the parameters' values [37] and obtain the cell values by combining pairs from the three calibration parameters while the remaining is fixed to ease the graph understanding. The six plots of Fig. 6 show that the NGA can get good calibration solutions partially distributed through the best fitting area of the search space (blue cells of the heat-map). The similarity filtering (second row of plots) cleans quite similar solutions without removing the insights coming from the analysis.

5.3. Sensitivity analysis and validation implications when using the framework

In this section we take the set of calibrated solutions of the best performing NGA, clearing $\kappa = 1$ and $R = 10$, and we use MOMCA to analyze the parameters' distribution and run a direct sensitivity analysis. Table 3 shows quantitative indicator values for each of the three calibration parameters ($\{p_M\}$, X_k , and $\{p_{X_m}\}$), obtained from the set of solutions. Fig. 7 presents the direct sensitivity analysis performed on two parameters, $\{p_M\}$, X_k , with respect to the model sensitivity of $\{p_{X_m}\}$. Please notice that sensitivity can be measured with respect to any other parameter or input/output variable of the DEB model. Fig. 7 also shows the histograms and computed *pdfs* of the parameters distribution based on the set of solutions.

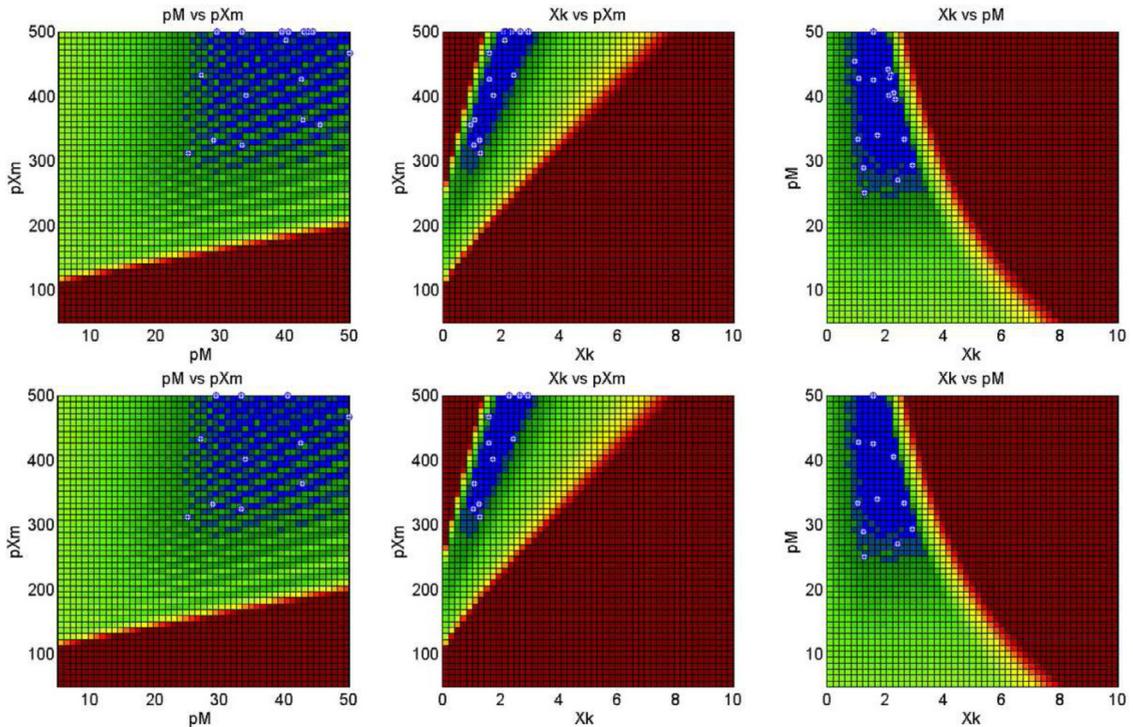


Fig. 6. Representation of the set of clearing NGA ($\kappa = 1, R = 10$) calibration solutions on the 2D plots of the multimodal search space of the DEB model. The three plots of the first row correspond to the non-filtered version of the NGA and the second row to the variant filtered by similarity. Blue cells represent the best calibrated models in terms of historical fitting. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 3

Analysis of the calibration parameters of the DEB model by using quantitative indicators from the set of calibration solutions (statistics obtained from 30 runs of clearing with $\kappa = 1$ and $R = 10$).

$\bar{x}(\sigma)$ for the 30 runs of clearing $\kappa = 1, R = 10$								
Parameter	Median	Spread	Max	Min	k	Skewness	BC	
$\{\dot{p}_M\} \in [5, 50]$	38.68 (1.8)	21.96 (2.73)	49.34 (1.34)	27.38 (1.96)	-1.05 (0.41)	0.02 (0.33)	0.53 (0.10)	
$X_k \in [0, 10]$	1.81 (0.25)	1.92 (0.28)	2.81 (0.24)	0.89 (0.14)	-0.9 (0.41)	0.02 (0.42)	0.51 (0.09)	
$\{\dot{p}_{Xm}\} \in [50, 500]$	439.51 (29.87)	184.09 (18.29)	499.91 (0.47)	315.82 (18.2)	-1.06 (0.79)	-0.37 (0.47)	0.63 (0.16)	

Columns of **Table 3** summarize the NGA results of the 30 runs by mean (\bar{x}) and standard deviation (σ). These values and the histograms of **Fig. 7** show that $\{\dot{p}_M\}$ can be set to a wide range of values without influencing the fitting of the model. The distribution of this parameter does not have skewness and it is unimodal according to its kurtosis k , skewness measure, and BC . In contrast, parameters X_k and $\{\dot{p}_{Xm}\}$ have a narrower relative spread and their histograms and quantitative values suggests more modes (i.e., more than one frequent value for the parameters in the set of solutions). $\{\dot{p}_{Xm}\}$ has a BC value greater than 0.55 and a skewness of -0.37 and therefore, its distribution is clearly right-skew and bimodal, at least. Parameter X_k presents the majority of its *acceptable* values around its median ($\bar{x} = 1.81, \sigma = 0.25$).

The scatter-plot of **Fig. 7** shows a clear linear dependence between the three parameters being evaluated. This fact could suggest the modeler a re-design of the DEB model or a simplification in the number of parameters used to control the growth of the biological process. A modeler can, for instance, inspect if this linear dependence is correct from the biological point of view and if one parameter can be given by the other two in order to better define the model. Let us again remark that this additional analysis of the third MOMCA step permits a better understanding of how the model and its dynamics works.

6. Second case study: a brand management model of a television show

6.1. Model description

A concrete case of strategic management is brand value management [1]. Brand value management is a complex, adaptive, and dynamic environment. This environment is usually a system with a high number of variables and contains non-

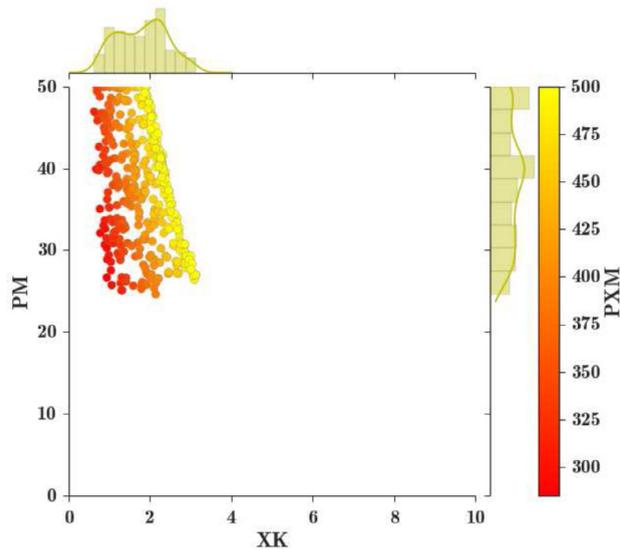


Fig. 7. Scatter-plot with a sensitivity analysis of parameters $\{p_M\}$ and X_k on $\{p_{Xm}\}$. Histograms and computed pdfs are also shown based on the set of solutions of the 30 runs of clearing with $\kappa = 1$ and $R = 10$.

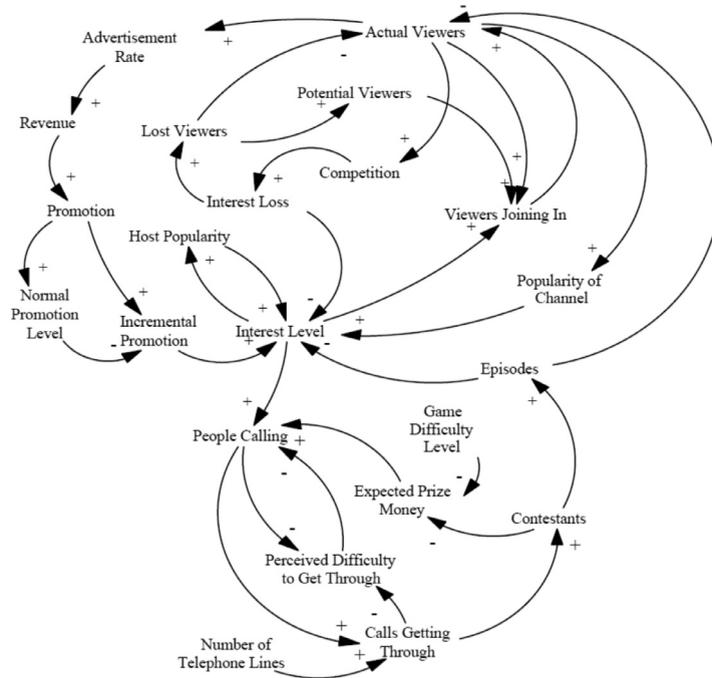


Fig. 8. Structure of the system dynamics model for the television show case study (from Mukherjee and Roy [45]).

linearities, inertia, delays, and bi-directional network feedback loops. System dynamics is an ideal methodology for complex feedback systems like brand management where all brand related components are treated as resources, which grow or erode over time [45]. As pointed out by Crossland [16], system dynamics allows brand managers to evaluate both the structure and the dynamic relationships between components of a brand management system.

The brand management case study published in [45] is a good example of this system dynamics application. In this work, authors modeled the management of the equity of the Indian television show *Who wants to be a millionaire?*. Fig. 8 shows the system dynamics diagram of the brand management model. We have adopted the latter model but using an extension of the sensitivity model [66] to allow managers to easily model the effects or relationships between the variables of the model structure [5,13]. Within this type of modeling, effects between variables are given by their temporal delay, intensity, and values of change; and variables are defined by some attributes such as their initial, optimum, and current value. Main description about this calibration problem is given below:

Table 4

Calibration results of the baseline and NGA methods for the brand management model in 30 independent runs of the algorithms.

Calibration algorithm	Number of returned solutions	Best solution fitness	Average fitness all solutions	Average distance among solutions
$\bar{x}(\sigma)$ statistics for baseline methods				
Nelder–Mead	1 (0)	9.36 (1.78)	9.36 (1.78)	0 (0)
Standard GA	436.53 (39.47)	7.80 (0.02)	7.98 (0.11)	0.05 (0.01)
$\bar{x}(\sigma)$ statistics for clearing $\kappa = 1$				
Niche radius $R = 10$	245.80 (60.27)	7.75 (0.03)	7.88 (0.13)	0.12 (0.02)
Filtered, niche radius $R = 10$	44.2 (14.58)	7.75 (0.03)	8.28 (0.1)	0.14 (0.01)
Niche radius $R = 5$	104.17 (10.94)	7.81 (0.02)	8.51 (0.03)	0.08 (0)
Filtered, niche radius $R = 5$	98.67 (10.68)	7.81 (0.02)	8.53 (0.03)	0.08 (0)
$\bar{x}(\sigma)$ statistics for clearing $\kappa = 3$				
Niche radius $R = 10$	295.33 (42.25)	7.76 (0.02)	8.05 (0.07)	0.1 (0.01)
Filtered, niche radius $R = 10$	48.27 (10.72)	7.76 (0.02)	8.32 (0.08)	0.13 (0.01)
Niche radius $R = 5$	111.00 (8.75)	7.82 (0.02)	8.53 (0.03)	0.07 (0)
Filtered, niche radius $R = 5$	106.90 (7.79)	7.82 (0.02)	8.54 (0.03)	0.07 (0)
$\bar{x}(\sigma)$ statistics for clearing $\kappa = 7$				
Niche radius $R = 10$	335.40 (34.34)	7.75 (0.02)	7.92 (0.08)	0.08 (0.01)
Filtered, niche radius $R = 10$	47.83 (10.2)	7.75 (0.02)	8.24 (0.07)	0.12 (0.01)
Niche radius $R = 5$	114.13 (10.25)	7.84 (0.02)	8.54 (0.03)	0.06 (0)
Filtered, niche radius $R = 5$	110.63 (10.04)	7.84 (0.02)	8.55 (0.03)	0.06 (0)

- **Calibration parameters:** 25 parameters that define the intensity of those system effects (relationships between the variables) which have the lowest modeler’s confidence level. These parameters P_i are integer values and referred to P_0, \dots, P_{24} .
- **Modeler’s knowledge about calibration parameters:** The 25 calibration parameters can have 50 possible discrete values ($P_i = \{1, \dots, 50\}$). Therefore, their granularity is $step_{P_i} = 1$.
- **Output of interest:** The goal of this calibration problem is to adjust the evolution of one of the main variables of the system, the *interest level*, which can be considered as the brand equity of the television show. This variable ranges from 0 to 100 and has a set of historical values during the last two years (with a periodicity of 1 month).
- **Deviation measure:** We use the root-mean-square error (RMSE), defined in Eq. 4, as the fitness function which calculates the deviation between the historical data and the model’s output of the variable.

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N |V_M(i) - V_E(i)|^2}, \tag{4}$$

where N is the number of months of the historical evolution, V_E the empirical data, and V_M the output vector of the model.

- **Modeler’s objective and concerns:** Modelers want to know the best calibration results to match the brand equity of the model with respect to reality. Their goal is also to understand the most important correlations between the values of the effects that obtain the best accuracy. These correlations can help modelers to find strong dependencies between the effects and variables of the system dynamics model. The sensitivity analysis and robustness of the effects’ parameters are also useful for them to locate those effects that do not affect the output of the system, i.e., brand equity of the show.

6.2. Calibration performance of the niching genetic algorithms

Table 4 shows the calibration results obtained by the baseline algorithms and six NGA variants with and without similarity filtering. As done in Section 5.2, the table presents the number of solutions of the final set, the best fitness value, and the average fitness and average distance among the solutions of the set. The values of the table are the mean (\bar{x}) and standard deviation (σ) for the 30 runs of the algorithms. We have independently run the Nelder–Mead algorithm 30 times by changing the initial parameter values randomly (stochastically). The final solutions of the 30 runs of the algorithm were grouped and considered as the set of calibration solutions.

The set of solutions shown in the table are those with a fitness value lower than 9 (*a priori* fitness filtering min_{filt} according to the modeler’s acceptable fitting). Additionally, we have used a similarity filter $\sigma_{filt} = 0.05$ which is the minimum Euclidean distance among the 25 normalized calibration parameter values of a solution to be considered a final solution. Again in this case study, we compute the averaged distance by calculating the Euclidean distance between the 25 normalized parameter values of the solutions of the set.

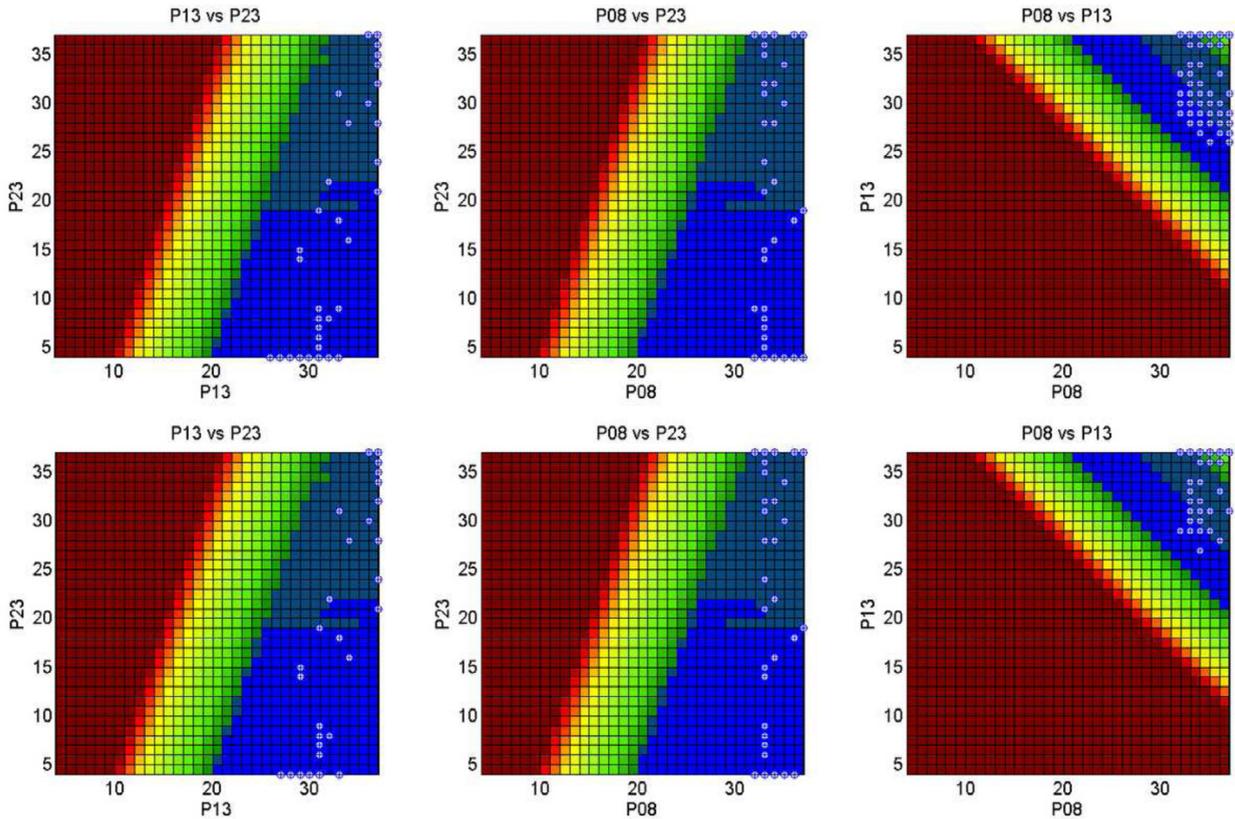


Fig. 9. Representation of the set of solutions obtained by clearing $\kappa = 1$ and $R = 10$ on the 2D plots of the multimodal search space of the brand management model. The three plots of the first row correspond to the non-filtered version of the NGA and the second row to the variant filtered by similarity. Blue cells represent the best calibrated models in terms of historical fitting. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

The analysis of the calibration performance results obtained by this experimentation suggests the following findings:

- NGA variants using a niche radius $R = 10$ get the best calibrated models for this brand management case. Independently from the capacity κ of the niche, their fitness results are better than the standard GA (which shares a common design but the multimodal component). The Nelder–Mead algorithm obtains the worst calibration results with a fitting of 9.36.
- The number of valid calibration solutions returned by the NGAs and the standard GA is high and all of them have better fitting than the solutions obtained by the Nelder–Mead (all show a fitness value lower than 9).
- The standard GA obtains the least diverse set of solutions (average distance of 0.05) while the clearing NGA with $\kappa = 1$ and $R = 10$ obtains very diverse solutions (average distance of 0.12 and 0.14).
- Averaged fitness of the solutions provided by the clearing variant with $\kappa = 1$ and $R = 10$ is slightly better than the standard GA and the rest of the NGA variants. However, the averaged fitness deteriorates when filtering the NGAs solutions and the number of solutions drastically falls (e.g., from 245.8 to 44.2). This is a common trade-off the modeler has to deal with in some cases: to get more diverse solutions with respect to the calibration parameters or to get a higher number of very similar solutions with respect to the calibration parameters.

With respect to the NGA variants we can see that the number of returned solutions increases without an important fitness loss when increasing the capacity of the niche (κ value). On the other hand, the averaged distance among the solutions of the set (fourth numerical column of Table 4) increases when the niche radius R increases as well.

The number of returned solutions when filtering is lower if we set the niche radius to $R = 10$ although a worse averaged fitness is obtained (third numerical column). However, if the NGA niche radius is set to $R = 5$, the similarity filtering does not practically affect the number of returned solutions, fitness values, and distances among the solutions. The reason is the use of a higher similarity filter σ_{filt} value than the niche radius R . As stated in Section 3.2, the filtering process and niching mechanisms of the NGA must be jointly used by modelers to adjust the cardinality and features of the returned set of calibration solutions.

Finally, Fig. 9 highlights, on six 2D search space plots, the solutions obtained by one run of the clearing NGA with $\kappa = 1$ and $R = 10$. We again ran a grid search to sample the parameters of the model and plot 2D plots of combinations of three arbitrary calibration parameters (P_{08} , P_{13} , and P_{23}) and its model fitting with respect to the historical data. To ease and

Table 5

Analysis of the calibration parameters of the television show model by using quantitative indicators from the set of calibration solutions (statistics obtained from 30 runs of clearing with $\kappa = 1$ and $R = 10$).

$\bar{x}(\sigma)$ for the 30 runs of clearing $\kappa = 1, R = 10$							
$P_i \in \{1, \dots, 50\}$	Median	Spread	Max	Min	k	Skewness	BC
P_0	16.86 (0.73)	9.31 (1.09)	17 (0)	7.69 (1.09)	33.95 (37.64)	-4.98 (2.8)	0.87 (0.1)
P_1	13.31 (2.79)	9.90 (0.3)	16.97 (0.18)	7.07 (0.25)	8.38 (12.72)	-0.93 (2.46)	0.61 (0.21)
P_2	13.1 (3.04)	9.66 (0.6)	16.86 (0.43)	7.21 (0.48)	8.73 (21.43)	-0.58 (2.7)	0.61 (0.25)
P_3	13.38 (3.07)	9.69 (0.83)	17 (0)	7.31 (0.83)	7.29 (12.89)	-0.38 (2.34)	0.65 (0.22)
P_4	10.50 (2.34)	9.52 (1.45)	16.62 (1.3)	7.1 (0.3)	15.78 (62.2)	0.35 (3.99)	0.62 (0.19)
P_5	11.79 (3.03)	9.72 (0.64)	16.83 (0.53)	7.1 (0.4)	4.49 (10.04)	0.16 (1.58)	0.57 (0.24)
P_6	14.24 (2.25)	9.31 (1.09)	16.79 (0.66)	7.48 (0.9)	11.02 (15.81)	-0.87 (2.32)	0.57 (0.25)
P_7	9.66 (2.4)	9.76 (0.68)	16.83 (0.46)	7.07 (0.36)	6.45 (15.08)	1.16 (2.2)	0.67 (0.19)
P_8	33.14 (3)	7.41 (3.7)	36.69 (1.32)	29.28 (3.44)	23.37 (55.04)	0.19 (4.48)	0.66 (0.24)
P_9	7.31 (0.59)	9.52 (0.81)	16.52 (0.81)	7.00 (0)	21.57 (21.17)	3.96 (1.93)	0.78 (0.15)
P_{10}	12.52 (3.01)	9.62 (0.89)	16.79 (0.66)	7.17 (0.59)	9.57 (17.67)	0.32 (2.6)	0.61 (0.2)
P_{11}	9.03 (2.51)	9.9 (0.3)	16.93 (0.25)	7.03 (0.18)	4.41 (8.04)	1.38 (1.62)	0.69 (0.19)
P_{12}	12.9 (3.21)	9.9 (0.3)	16.97 (0.18)	7.07 (0.25)	7.87 (15.96)	-0.42 (2.75)	0.69 (0.21)
P_{13}	31.21 (3.19)	9.41 (4.07)	36.69 (0.91)	27.28 (3.77)	34.41 (74.17)	-0.29 (5.52)	0.71 (0.23)
P_{14}	12.86 (3.46)	9.62 (0.81)	16.86 (0.43)	7.24 (0.68)	10.48 (20.44)	-0.81 (2.9)	0.65 (0.19)
P_{15}	12.28 (3.11)	9.9 (0.3)	16.97 (0.18)	7.07 (0.25)	5.12 (6.85)	-0.30 (2.15)	0.65 (0.18)
P_{16}	7.62 (1)	9 (1.39)	16 (1.39)	7 (0)	15.28 (12.08)	3.22 (1.64)	0.71 (0.2)
P_{17}	36.72 (0.94)	29.03 (6.16)	37 (0)	7.97 (6.16)	26.18 (49.35)	-4.06 (2.95)	0.89 (0.07)
P_{18}	5.83 (3.5)	32.41 (2.19)	36.55 (1.89)	4.14 (0.73)	6.37 (12.75)	2.23 (1.2)	0.87 (0.18)
P_{19}	7.91 (1.8)	9.38 (1.1)	16.38 (1.1)	7 (0)	18.33 (20.71)	3.27 (2.44)	0.77 (0.16)
P_{20}	12.76 (2.9)	9.76 (0.5)	16.93 (0.25)	7.17 (0.46)	6.2 (11.66)	-0.31 (2.09)	0.59 (0.26)
P_{21}	13.48 (3.08)	9.69 (0.53)	16.93 (0.25)	7.24 (0.5)	9.23 (18.19)	-0.87 (2.2)	0.64 (0.2)
P_{22}	14.24 (2.48)	9.79 (0.41)	16.9 (0.3)	7.1 (0.3)	15.94 (42.42)	-2.11 (3.21)	0.67 (0.2)
P_{23}	6.52 (2.86)	30.07 (7.13)	34.38 (6.48)	4.31 (1.32)	22.28 (60.59)	3.21 (3.49)	0.86 (0.19)
P_{24}	7.03 (0.18)	9.17 (1.21)	16.17 (1.21)	7 (0)	36.99 (24.76)	5.55 (1.99)	0.87 (0.08)

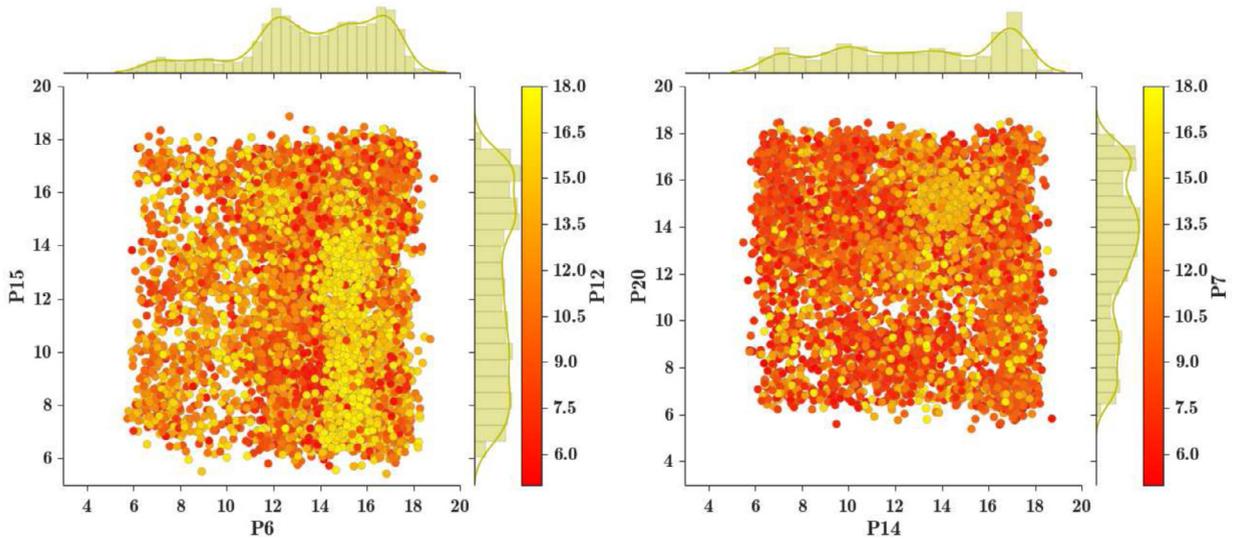


Fig. 10. Sensitivity analysis of parameters P_6/P_{15} , P_{14}/P_{20} on P_{12} and P_7 are shown in two scatter-plots. A jittering method (i.e., adding random noise to the integer data points) was used to prevent overplotting. Histograms and computed *pdfs* are also shown based on the set of solutions of the 30 runs of clearing with $\kappa = 1$ and $R = 10$.

simplify the visualization, the fitness values of the cells of the plots are obtained by fixing the values of the remaining 22 calibration parameters.

6.3. Sensitivity analysis and validation implications when using the framework

Table 5 shows the values of the quantitative indicators presented in MOMCA for exploring the parameters of the television show model. As presented in Section 5.3, we show statistics of the 30 runs for the best performing NGA, clearing with $\kappa = 1$ and $R = 10$. Fig. 10 shows two scatter-plots for direct sensitivity analysis on variables P_7 and P_{12} as well as the histograms and computed *pdfs* of parameters P_6 , P_{15} , P_{14} and P_{20} .

The modeler can identify those parameters with higher variability, more modes, and skew distributions by observing the values of Table 5 and histograms of Fig. 10. P_{17} , P_{18} , and P_{23} are parameters with high spread. From a modeler point of view it means their values do not have an important effect to obtain good calibrated models. On the contrary, P_8 and P_{13} are parameters with more restrictive ranges. Therefore, their values must be strictly set to integer values close to 33 and 31 to have a calibrated model with respect to the brand equity of the television show. Table 5 suggests that the majority of the parameters have either positive or negative skewness and their distributions are, at least, bimodal (BC values greater than 0.55). Histograms and *pdfs* shown in Fig. 10 also visualize the existence of more than one mode in the parameters' distributions.

Jittered scatter-plots of Fig. 10 show a sensitivity analysis on P_{12} and P_7 . These variables were chosen at random and any other sub-sets of input/output variables of the model can be visualized and analyzed directly from the set of solutions of the NGA. In the first scatter-plot we can see a cluster of yellow points when $P_6 = 15$. This fact reflects a relationship between P_{12} and P_6 . The same occurs in the second plot when $P_{20} = 15$ and $P_{12} = 15$. All these dependencies between those parameters can help the modeler to identify effects between the variables of the system dynamics model that are strongly related. The modeler must check the model design and the observed reality to validate if these dependencies, pointed out by the automated sensitivity analysis of the framework, are correct.

7. Concluding remarks

In this work we have presented MOMCA, a novel framework for applying multimodal optimization to the calibration of non-linear models. The framework is based on NGAs and has three different stages: the search of a set of calibration solutions, an interactive filtering process, and an automated analysis of the set of solutions for a model sensitivity and parameters' understanding. The main goal of the presented MOMCA framework is to assist modelers with an integral set of methods for model validation in order to increase the confidence and acceptance of the model's results by decision makers and stakeholders.

We have applied the framework to two different case studies: a biological process using DEB and a television show brand management using system dynamics. The results and benefits of the proposed framework for both case studies were consistent even if they belong to different application fields and modeling methodologies. The proposed multimodal optimization accomplished by GA-based methods obtained better results than the Nelder–Mead algorithm in terms of the model fitting ability (the final goal of the calibration problem). In addition, NGAs based on the clearing method outperformed the standard GA for calibrating the television show brand case, while they obtained the same results in the DEB model. Clearing variants with $\kappa = 1$ and $R = 10$ were the best algorithms for both case studies in terms of fitting and diversity of the set of solutions.

The exploitation of the set of *valid* returned solutions, after the NGA running, and the interactive nature of MOMCA (e.g., its filtering process) is an important toolbox for validation purposes. It allows a general view of the model behavior at first, and in a second place, a focus on the more interesting modeler's concern. The sensitivity analysis performed on the set of results showed the range of valid values for the model's parameters and how other variables respond to changes in the parameters. We think that this information, directly obtained from the results of the calibration method, is useful for the modeler as it can increase her/his knowledge and understanding of the system and its dynamics. It can also generate rich insights about the model behavior by identifying the possible alternative calibration solutions and their input parameters' distributions. The use of both quantitative and visualization tools are complementary and help when understanding the model and its validity.

Our framework could present several limitations with respect to the number of the parameters to be calibrated and outputs of the model. Although NGAs do not have a limitation to the number of parameters to calibrate, more evaluations (i.e., computational time), number of niches, population size, and/or additional GA mechanisms might be required. Other limitations related to the inspection and sensitivity analysis can also appear taking into account the psychological nature and human-effort aspects while analyzing the indicators and visualization tools. In the case of a high number of parameters, modelers cannot control the model itself and cannot quickly analyze the parameters behavior and their dependencies. However, we must state that dealing with many calibration parameters at the same time when validating a model is not a convenient approach and modelers must focus and reduce the problem to different calibration sub-problems in an iterative way [49].

Our future research may first focus on proposing more advanced visualization techniques and extending the *innovation process* on the set of solutions with data science methods to analytically summarize dependencies between the model's components [18,19] in order to overcome the above-mentioned limitations. We would also like to broaden our study by using multiobjective GAs [12,15] and robust optimization mechanisms [7,11] to calibrate several output variables at the same time and consider their inherent uncertainty. Finally, it is our aim to apply MOMCA to other modeling methodologies such as agent based modeling [8,14].

Acknowledgments

This work is supported by Spanish Ministerio de Economía y Competitividad under the NEWSOCO project (ref. TIN2015-67661), including European Regional Development Funds (ERDF), statutory activities of the Faculty of Computer Science and

Management of Wroclaw University of Technology and by the European Commission under the 7th Framework Programme, Coordination and Support Action, Grant Agreement Number 316097, Engine project. Authors also wish to thank Dr. Ramón Filgueira for his suggestions, help, and paper revision.

References

- [1] D.A. Aaker, Measuring brand equity across products and markets, *California Manage. Rev.* 38 (1996) 102–120.
- [2] J.J. Aguilera, M. Chica, M.J. del Jesus, F. Herrera, Niching genetic feature selection algorithms applied to the design of fuzzy rule based classification systems, in: *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'07)*, London (UK), 2007, pp. 1794–1799.
- [3] T. Back, D.B. Fogel, Z. Michalewicz, *Handbook of Evolutionary Computation*, IOP Publishing Ltd., Bristol (UK), 1997.
- [4] J. Banga, E. Balsa-Canto, Parameter estimation and optimal experimental design, *Essays Biochem.* 45 (2008) 195–210.
- [5] J. Barranquero, M. Chica, O. Cordón, S. Damas, Detecting key variables in system dynamics modelling by using social network metrics, in: F. Amblard, F.J. Miguel, A. Blanchet, B. Gaudou (Eds.), *Advances in Artificial Economics, Lecture Notes in Economics and Mathematical Systems*, 676, Springer International Publishing, 2015, pp. 207–217.
- [6] N.D. Bennett, B.F. Croke, G. Guariso, J.H. Guillaume, S.H. Hamilton, A.J. Jakeman, S. Marsili-Libelli, L.T. Newham, J.P. Norton, C. Perrin, Characterising performance of environmental models, *Environ. Model. Softw.* 40 (2013) 1–20.
- [7] H. Beyer, B. Sendhoff, Robust optimization – a comprehensive survey, *Comput. Methods Appl. Mech. Eng.* 196 (2007) 3190–3218.
- [8] E. Bonabeau, Predicting the unpredictable, *Harv. Bus. Rev.* 80 (2002) 109–116.
- [9] B. Calvez, G. Hutzler, Automatic tuning of agent-based models using genetic algorithms, in: *Multi-Agent-Based Simulation VI*, Springer, 2005, pp. 41–57.
- [10] T. Chai, R.R. Draxler, Root mean square error (RMSE) or mean absolute error (MAE)?—arguments against avoiding RMSE in the literature, *Geosci. Model Dev.* 7 (2014) 1247–1250.
- [11] M. Chica, J. Bautista, O. Cordón, S. Damas, A multiobjective model and evolutionary algorithms for robust time and space assembly line balancing under uncertain demand, *Omega* 58 (2016) 55–68.
- [12] M. Chica, O. Cordón, S. Damas, J. Bautista, Multiobjective, constructive heuristics for the 1/3 variant of the time and space assembly line balancing problem: ACO and random greedy search, *Inf. Sci.* 180 (2010) 3465–3487.
- [13] M. Chica, O. Cordón, S. Damas, V. Iglesias, J. Mingot, Identimod: Modeling and managing brand value using soft computing, *Decis. Support Syst.* 89 (2016) 41–55.
- [14] M. Chica, W. Rand, Building agent-based decision support systems for word-of-mouth programs. A freemium application, *J. Marketing Res.* (2016). <http://ssrn.com/abstract=2821171>.
- [15] C.A.C. Coello, D.A. Van Veldhuizen, G.B. Lamont, *Evolutionary Algorithms for Solving Multi-objective Problems*, Springer, 2002.
- [16] P. Crossland, Value creation in fine arts: a system dynamics model of inverse demand and information cascades, *Strategic Manage. J.* 23 (2002) 417–434.
- [17] D. Datta, J.R. Figueira, A real–integer–discrete-coded differential evolution, *Appl. Soft Comput.* 13 (2013) 3884–3893.
- [18] K. Deb, S. Bandaru, D. Greiner, A. Gaspar-Cunha, C.C. Tutum, An integrated approach to automated innovization for discovering useful design principles: case studies from engineering, *Appl. Soft Comput.* 15 (2014) 42–56.
- [19] K. Deb, A. Srinivasan, Innovization: innovating design principles through optimization, *ACM*, 2006. Proceedings of the 8th annual conference on Genetic and evolutionary computation, 1629–1636.
- [20] L.T. DeCarlo, On the meaning and use of kurtosis, *Psychol. Methods* 2 (1997) 292.
- [21] C. Eksin, Genetic algorithms for multi-objective optimization in dynamic systems, in: *Proceedings of the 26th International System Dynamics Conference*, 2008.
- [22] A.H. Elsheikh, M.F. Wheeler, I. Hoteit, Clustered iterative stochastic ensemble method for multi-modal calibration of subsurface flow models, *J. Hydrol.* 491 (2013) 40–55.
- [23] L.J. Eshelman, J.D. Schaffer, Real-coded genetic algorithms and interval-schemata, *Found. Genet. Algorithms* 2 (1993) 187–202.
- [24] R. Filgueira, R. Rosland, J. Grant, A comparison of scope for growth (SFG) and dynamic energy budget (DEB) models applied to the blue mussel (*mytilus edulis*), *J. Sea Res.* 66 (2011) 403–410.
- [25] J.W. Forrester, System dynamics: the next fifty years, *Syst. Dyn. Rev.* 23 (2007) 359–370.
- [26] J.B. Freeman, R. Dale, Assessing bimodality to detect the presence of a dual cognitive process, *Behav. Res. Methods* 45 (2013) 83–97.
- [27] M. Gilli, P. Winker, A global optimization heuristic for estimating agent based models, *Comput. Stat. Data Anal.* 42 (2003) 299–312.
- [28] D.E. Goldberg, J. Richardson, Genetic algorithms with sharing for multimodal function optimization, Hillsdale, NJ: Lawrence Erlbaum, 1987. Genetic algorithms and their applications: Proceedings of the Second International Conference on Genetic Algorithms, 41–49.
- [29] R. Guzmán-Cruz, R. Castañeda Miranda, J. García-Escalante, I. López-Cruz, A. Lara-Herrera, J. De la Rosa, Calibration of a greenhouse climate model using evolutionary algorithms, *Biosyst. Eng.* 104 (2009) 135–142.
- [30] C. Hamarat, J.H. Kwakkel, E. Pruyt, E.T. Loonen, An exploratory approach for adaptive policy-making by using multi-objective robust optimization, *Simul. Model. Pract. Theory* 46 (2014) 25–39.
- [31] A.J. Heppenstall, A.J. Evans, M.H. Birkin, Genetic algorithm optimisation of an agent-based model for simulating a retail market, *Environ. Plann. B* 34 (2007) 1051–1070.
- [32] F. Herrera, M. Lozano, J.L. Verdegay, Tackling real-coded genetic algorithms: operators and tools for behavioural analysis, *Artif. Intell. Rev.* 12 (1998) 265–319.
- [33] D. Joanes, C. Gill, Comparing measures of sample skewness and kurtosis, *J. Royal Stat. Soc.* 47 (1998) 183–189.
- [34] N.B. Kamath, R. Roy, Capacity augmentation of a supply chain for a short lifecycle product: A system dynamics framework, *Eur. J. Oper. Res.* 179 (2007) 334–351.
- [35] S.A.L.M. Kooijman, Energy budgets can explain body size relations, *J. Theor. Biol.* 121 (1986) 269–282.
- [36] S.A.L.M. Kooijman, *Dynamic Energy and Mass Budgets in Biological Systems*, Cambridge University Press, 2000.
- [37] J.S. Lee, T. Filatova, A. Ligmann-Zielinska, B. Hassani-Mahmoei, F. Stonedahl, I. Lorscheid, A. Voinov, J.G. Polhill, Z. Sun, D.C. Parker, The complexities of agent-based modeling output analysis, *J. Artif. Soc. Social Simul.* 18 (2015) 4.
- [38] A. Ligmann-Zielinska, D.B. Kramer, K.S. Cheruvellil, P.A. Soranno, Using uncertainty and sensitivity analyses in socioecological agent-based models to improve their analytical performance and policy relevance, *PLoS one* 9 (2014) e109779.
- [39] S. Liu, D. Butler, R. Brazier, L. Heathwaite, S.T. Khu, Using genetic algorithms to calibrate a water quality model, *Sci. Total Environ.* 374 (2007) 260–272.
- [40] C.S. de Magalhães, D.M. Almeida, H.J.C. Barbosa, L.E. Dardenne, A dynamic niching genetic algorithm strategy for docking highly flexible ligands, *Inf. Sci.* 289 (2014) 206–224.
- [41] J. Van der Meer, An introduction to dynamic energy budget (DEB) models with special emphasis on parameter estimation, *J. Sea Res.* 56 (2006) 85–102.
- [42] J.H. Miller, Active nonlinear tests (ANTs) of complex simulation models, *Manage. Sci.* 44 (1998) 820–830.
- [43] J. Morecroft, *Strategic Modelling and Business Dynamics: A Feedback Systems Approach*, John Wiley & Sons, 2007.
- [44] R. Muetzelfeldt, J. Massheder, The simile visual modelling environment, *Eur. J. Agron.* 18 (2003) 345–358.
- [45] A. Mukherjee, R. Roy, A system dynamic model of management of a television game show, *J. Model. Manage.* 1 (2006) 95–115.

- [46] M.A. Muñoz, Y. Sun, M. Kirley, S.K. Halgamuge, Algorithm selection for black-box continuous optimization problems: a survey on methods and challenges, *Inf. Sci.* 317 (2015) 224–245.
- [47] J. Nelder, R. Mead, A simplex method for function minimization, *Comput. J.* 7 (1965) 308–313.
- [48] R. Nisbet, E. Muller, K. Lika, S. Kooijman, From molecules to ecosystems through dynamic energy budget models, *J. Anim. Ecol.* 69 (2000) 913–926.
- [49] R. Oliva, Model calibration as a testing strategy for system dynamics models, *Eur. J. Oper. Res.* 151 (2003) 552–568.
- [50] R. O'Neill, Algorithm AS 47: function minimization using a simplex procedure, *Appl. Stat.* 20 (1971) 338–345.
- [51] S. Overbeek, P. van Bommel, H.E. Proper, Statics and dynamics of cognitive and qualitative matchmaking in task fulfillment, *Inf. Sci.* 181 (2011) 129–149.
- [52] E. Pérez, F. Herrera, C. Hernández, Finding multiple solutions in job shop scheduling by niching genetic algorithms, *J. Intell. Manuf.* 14 (2003) 323–339.
- [53] E. Pérez, M. Posada, F. Herrera, Analysis of new niching genetic algorithms for finding multiple solutions in the job shop scheduling, *J. Intell. Manuf.* 23 (2012) 341–356.
- [54] A. Pétrowski, A clearing procedure as a niching method for genetic algorithms, *IEEE, 1996. Evolutionary Computation, 1996., Proceedings of IEEE International Conference on*, 798–803.
- [55] R. Pfister, K.A. Schwarz, M. Janczyk, R. Dale, J. Freeman, Good things peak in pairs: a note on the bimodality coefficient, *Front. Psychol.* 4 (2013) 700.
- [56] H. Qudrat-Ullah, On the validation of system dynamics type simulation models, *Telecommun. Syst.* 51 (2012) 159–166.
- [57] R. Rosland, O. Strand, M. Alunno-Bruscia, C. Bacher, T. Strohmeier, Applying dynamic energy budget (DEB) theory to simulate growth and bio-energetics of blue mussels under low seston conditions, *J. Sea Res.* 62 (2009) 49–61.
- [58] M. Saleh, R. Oliva, C.E. Kampmann, P.I. Davidsen, A comprehensive analytical approach for policy analysis of system dynamics models, *Eur. J. Oper. Res.* 203 (2010) 673–683.
- [59] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, S. Tarantola, *Global Sensitivity Analysis: The Primer*, John Wiley & Sons, 2008.
- [60] R.G. Sargent, Verification and validation of simulation models, in: *Proceedings of the 37th Conference on Winter Simulation, 2005*, pp. 130–143.
- [61] G. Singh, K. Deb, Comparison of multi-modal optimization algorithms based on evolutionary algorithms, in: *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, New York, USA, 2006, pp. 1305–1312.
- [62] J.D. Sterman, System dynamics modeling: tools for learning in a complex world, *California Manage. Rev.* 43 (2001) 8–25.
- [63] F. Stonedahl, W. Rand, When Does Simulated Data Match Real Data? Comparing Model Calibration Functions Using Genetic Algorithms, in: *Advances in Computational Social Science*, in: *Agent-Based Social Systems*, 11, Springer, Japan, 2014, pp. 297–313.
- [64] F. Stonedahl, U. Wilensky, Evolutionary robustness checking in the artificial Anasazi model, in: *Proceedings of the AAAI Fall Symposium on Complex Adaptive Systems: Resilience, Robustness, and Evolvability*, 2010, pp. 120–129.
- [65] J.C. Thiele, W. Kurth, V. Grimm, Facilitating parameter estimation and sensitivity analysis of agent-based models: a cookbook using netlogo and r, *J. Artif. Soc. Social Simul.* 17 (2014) 11.
- [66] F. Vester, *The Art of Interconnected Thinking: Tools and Concepts for a New Approach to Tackling Complexity*, MCB Verlag, 2007.
- [67] A. Voinov, F. Bousquet, Modelling with stakeholders, *Environ. Model. Softw.* 25 (2010) 1268–1281.
- [68] S. Wei, H. Yang, J. Song, K.C. Abbaspour, Z. Xu, System dynamics simulation model for assessing socio-economic impacts of different levels of environmental flow allocation in the weihe river basin, china, *Eur. J. Oper. Res.* 221 (2012) 248–262.
- [69] A. Will, J. Bustos, M. Bocco, J. Gotay, C. Lamelas, On the use of niching genetic algorithms for variable selection in solar radiation estimation, *Renewable Energy* 50 (2013) 168–176.
- [70] E. Yu, P.N. Suganthan, Ensemble of niching algorithms, *Inf. Sci.* 180 (2010) 2815–2833.
- [71] G. Yücel, Y. Barlas, Automated parameter specification in dynamic feedback models based on behavior pattern features, *Syst. Dyn. Rev.* 27 (2011) 195–215.
- [72] E.C.T. Zúñiga, I.L.L. Cruz, A.R. García, Parameter estimation for crop growth model using evolutionary and bio-inspired algorithms, *Appl. Soft Comput.* 23 (2014) 474–482.