Improving the performance of a preference-based multi-objective algorithm to optimize food treatment processes

Journal:	*Engineering Optimization*
Manuscript ID	GENO-2018-1020.R2
Manuscript Type:	Original Article
Date Submitted by the Author:	21-Apr-2019
Complete List of Authors:	Ferrández, Miriam; Universidad de Almeria, Department of Computer Science Redondo, Juana; Universidad de Almeria, Department of Computer Science Ivorra, Benjamin; Univ Complutense Madrid, Ramos, Angel; Universidad Complutense de Madrid Ortigosa, Pilar; Universidad de Almeria, Department of Computer Science Paechter, Ben; Edinburgh Napier University, School of Computing
Keywords:	preference-based multi-objective optimization algorithm, low-cost optimization, food treatment
Note: The following files were submitted by the author for peer review, but cannot be converted to PDF. You must view these files (e.g. movies) online.	
improving-wasfga.tex foodeng_and_epidemiology_miriam.bib	

ARTICLE TEMPLATE

Improving the performance of a preference-based multi-objective algorithm to optimize food treatment processes

M. R. Ferrández[a], J. L. Redondo[a], B. Ivorra[b],
A. M. Ramos[b], P. M. Ortigosa[a] and B. Paechter[c]

[a]Department of Computer Science, University of Almería, ceiA3, Ctra. Sacramento, La Cañada de San Urbano, Almería 04120, Spain;
[b]Instituto de Matemática Interdisciplinar (IMI) and Department of Mathematical Analysis and Applied Mathematics, Complutense University of Madrid, Plaza de las Ciencias, 3, Madrid 28040, Spain;
[c]School of Computing, Edinburgh Napier University, Edinburgh, Scotland, UK

ARTICLE HISTORY
Compiled April 21, 2019

ABSTRACT
This work focuses on the optimization of some high-pressure and temperature food treatments. In some cases, when dealing with real–life multi-objective optimization problems, such as the one considered here, the computational cost of evaluating the considered objective functions is usually quite high. Therefore, only a reduced number of iterations is affordable for the optimization algorithm. However, using fewer iterations can lead to inaccurate solutions far from the real Pareto optimal front. In this article, different mechanisms are analyzed and compared to improve the convergence of a preference-based multi-objective optimization algorithm called Weighting Achievement Scalarizing Function Genetic Algorithm. The combination of these techniques has been applied to optimize a particular food treatment process. In particular, one of the proposed methods, based on the introduction of an advanced population, achieves important improvements in the considered quality indicator measures.

KEYWORDS
preference-based multi-objective optimization algorithm; low-cost optimization; food treatment

1. Introduction

Recently, High-Pressure and Temperature (HPT) processes have emerged as a reference technology in the food industry, mainly due to the fact that they enhance the preservation of some good properties (e.g., organoleptic) and reduce the proliferation of damaging micro-organisms without using additives. To produce a processed food within predetermined quality levels, the food engineer must carefully determine the control temperatures and pressures to be applied during the HPT treatment. Nowadays, this challenging task is accomplished by several trial and error tests. However, since the target quality levels depend on the country in which they operate, these tests can lead to a considerable waste of time, money and product. In (Ferrández et al.

CONTACT M. R. Ferrández. Email: mrferrandez@ual.es

2019), a decision tool was proposed based on solving a multi-objective optimization problem to assist the food engineers in the design of HPT treatments. More precisely, the optimization problem consists in finding the initial and refrigeration temperatures and the pressure temporal profile to be provided to the HPT equipment such that the final enzymatic activity in the food and the maximum temperature reached during the whole process are minimal and the final vitamin activity is maximal. In general, this kind of optimization problems arising in real–life situations usually involve some mathematical models for describing the physical phenomena. Although these models are simplifications of the reality, they frequently imply computationally expensive calculations managing many variables and objectives (see, e.g., Gomez, Ivorra, and Ramos (2011a); Ivorra et al. (2013)). In this context, the meta–heuristic algorithms are quite suitable for dealing with their optimization. Some particular examples illustrating the wide range of applicability of this kind of algorithms in engineering problems are available in literature (see, for instance, Ahmadi et al. (2014); Ahmadi, Ahmadi, and Sadatsakkak (2015); Crespo et al. (2017); Ivorra et al. (2013)).

Considering a classical multi-objective optimization problem of the form

$$\begin{array}{ll} \min & \{f_1(\mathbf{x}), \ldots, f_m(\mathbf{x})\}, \\ \text{s.t.} & \mathbf{x} \in S \subseteq \mathbb{R}^n, \end{array} \tag{1}$$

the goal is to find a set of *decision vectors* $\mathbf{x} = (x_1, \ldots, x_n)$, with $n \in \mathbb{N}$, belonging to a set $S \subseteq \mathbb{R}^n$ called *feasible region*, such that the values obtained when evaluating the *objective functions* $f_1, \ldots, f_m : \mathbb{R}^n \longrightarrow \mathbb{R}$ at these decision vectors are minimal. However, minimizing all the objectives simultaneously is not always a trivial and feasible task because they frequently confront each other. So, the decision vectors of the solution set, known as *Pareto optimal set*, must be those having the best compromise among the considered objectives.

In this framework, a decision vector $\mathbf{x}^* \in S$ is said to be *efficient* if and only if there does not exist another feasible vector $\mathbf{x}$ in S dominating $\mathbf{x}^*$, that is, there does not exist another feasible vector $\mathbf{x} \in S$ satisfying that $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*)$, for all $i = 1, \ldots, m$, and $f_i(\mathbf{x}) < f_i(\mathbf{x}^*)$ for at least one index i (i. e., none of the objective values can be improved without worsening at least one of the others). The image of these non-dominated efficient vectors in the feasible objective region $F(S) \subseteq \mathbb{R}^n$ is known as Pareto optimal front. Additionally, a decision vector $\mathbf{x}^*$ in S is said to be *weakly efficient* if and only if there does not exist another feasible vector $\mathbf{x} \in S$ such that $f_i(\mathbf{x}) < f_i(\mathbf{x}^*)$, for all $i = 1, \ldots, m$.

As said before, the meta–heuristic algorithms are especially prescribed for solving complex problems due to the fact that they do not require any a-priori information about the objective functions (such as their gradient or Hessian matrix). More precisely, they provide a finite set of points composing a Pareto front approximation as a solution of (1). Among those algorithms, one of the most popular subfamily of methods are the so-called evolutionary multi-objective algorithms (EMOA), which are based on iterative procedures that continuously improve the set of approximated solutions bringing them closer to the Pareto front. When dealing with real–life applications, only a reduced number of iterations may be computationally affordable for the optimization algorithm, due to the high cost of evaluating the objective function (see, e.g., Carrasco, Ivorra, and Ramos (2015); Crespo et al. (2017); Gomez, Ivorra, and Ramos (2011b); Ivorra et al. (2006, 2013)). In particular, for the industrial problem of optimizing a HPT treatment considered here, a single evaluation lasts 46.5 seconds on average. However, using fewer iterations can lead to inaccurate solutions far from

the real Pareto optimal front.

A literature review provides some interesting ideas and methods aimed at enhancing the convergence and, consequently, achieving better results (i.e., solutions closer to the Pareto front) in fewer iterations. For instance, in (Ivorra, Mohammadi, and Ramos 2015), the authors apply a secant technique for improving the initialization of a generic mono-objective optimization algorithm. Other widely-used mechanisms are the local search methods. In particular, the single agent stochastic search (SASS) algorithm (Solis and Wets 1981) was adapted by Lancinskas, Ortigosa, and Žilinskas (2013) to work with multi-objective problems. This extended version called MOSASS showed successful results in (Filatovas et al. 2016; Redondo, Fernández, and Ortigosa 2017), where it is added as part of an evolutionary algorithm.

As done in the previous works (Ferrández et al. 2018, 2019), it is assumed that practitioners are only interested in calculating a specific region of the Pareto front determined by the food engineer and known as the *region of interest*. Therefore, the EMOA called Weighting Achievement Scalarizing Function Genetic Algorithm (WASF-GA) is used. According to Ruiz, Saborido, and Luque (2015), it seems to provide higher-quality results than other preference-based algorithms when the optimization problem has three or more objectives, as it is the case here.

Now, in this work, some variants of WASF-GA are proposed applying different mechanisms in order to achieve a set of solutions that closely approximate the Pareto front in the region of interest using a low number of iterations. The results obtained with those different WASF-GA variants when solving the considered food processing problem have been compared in terms of effectiveness using several quality indicator measure methods.

The article is organized as follows. First, in Section 2, the design of the HPT treatment is formulated as a multi-objective optimization problem. In Section 3, the original WASF-GA is described. Next, in Section 4, different mechanisms to improve WASF-GA are proposed. In Section 5, the computational experiments carried out to compare the original WASF-GA and its considered variants are detailed. Finally, in Section 6, the optimization results are discussed in terms of some state-of-the-art quality measures.

2. Food processing problem

In this work, the optimization problem consists in determining a set of temperature and pressure configurations for the HPT treatment that minimize the enzymatic activity (denoted as function f_1) and the maximum temperature reached during the food processing (denoted as function f_3) and maximize the vitamin activity (denoted as function f_2), at the same time (Ferrández et al. 2019) inside the food domain denoted as Ω_{F}. It is formulated as follows:

$$\begin{cases} \min & f_1(T_0, T_{\mathrm{r}}, \Delta P_1, \ldots, \Delta P_n), \\ \max & f_2(T_0, T_{\mathrm{r}}, \Delta P_1, \ldots, \Delta P_n), \\ \min & f_3(T_0, T_{\mathrm{r}}, \Delta P_1, \ldots, \Delta P_n), \\ \text{s.t.} & T_0, T_{\mathrm{r}} \in [10, 50](^{\circ}\mathrm{C}), \\ & \Delta P_1, \ldots, \Delta P_n \in [-250, 250](\mathrm{MPa}), \end{cases} \tag{2}$$

where the decision vectors $(T_0, T_{\mathrm{r}}, \Delta P_1, \ldots, \Delta P_n)$ are composed by the initial T_0 and refrigeration T_{r} temperatures (°C), and the set $\{\Delta P_1, \ldots, \Delta P_n\}$ corresponding to the

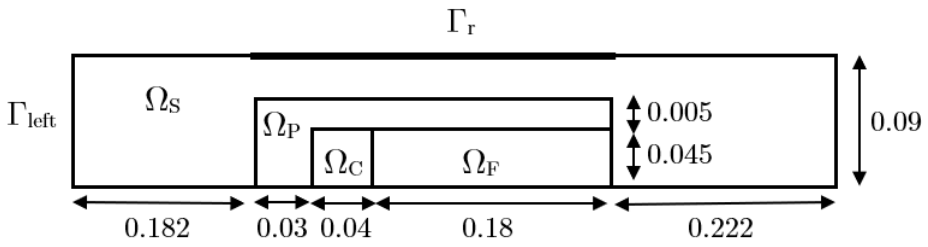


Figure 1. Computational domain $\Omega = \Omega_F \cup \Omega_C \cup \Omega_P \cup \Omega_S$.

pressure variations (MPa) applied to the High-Pressure equipment during the treatment.

More precisely, the objective functions f_1 and f_2 are the average of the activities in Ω_F (assumed to be in 2D, see below) at the final time of the HPT process, denoted as t_f. They are computed as follows:

$$f_i(T_0, T_r, \Delta P_1, \ldots, \Delta P_n) = \frac{1}{|\Omega_F|} \iint_{\Omega_F} A_i(r, z, t_f) \mathrm{d}r\mathrm{d}z, \qquad i = 1, 2,$$

where A_1 refers to the enzyme activity and A_2 to the vitamin activity. Additionally, the maximum temperature registered in the food sample Ω_F during the processing time interval $[0, t_f]$ is given by:

$$f_3(T_0, T_r, \Delta P_1, \ldots, \Delta P_n) = \max_{(r,z)\in\Omega_F,\ t\in[0,t_f]} T(r, z, t).$$

Functions A_1, A_2 and T depend on spatial coordinates (here, $(r, z) \in \Omega_F$) and time $t \geq 0$ (measured since the beginning of the HPT process). As proposed in (Infante et al. 2009), an axisymmetrical 2D configuration of the cylindrical HPT device is assumed. Indeed, the rectangular spatial domain corresponding to the simulation domain is denoted by $\Omega = [0, L] \times [0, H]$. Ω contains four subdomains: an inscribed rectangle $\Omega_F = [L_1, L_2] \times [0, H_1]$ occupied by the food sample, the cap of this food sample Ω_C, the pressurizing fluid chamber surrounding them Ω_P, and the external steel vessel Ω_S (see Figure 1). Considering this geometry, three kind of boundaries are distinguished in Ω: the left edge of the rectangle is $\Gamma_{left} = \{0\} \times [0, H]$; the refrigeration frontier is $\Gamma_r = [L_{r1}, L_{r2}] \times \{H\}$; and the remaining boundaries are $\Gamma \backslash (\Gamma_r \cup \Gamma_{left})$ composed by the walls and the symmetry axis, which are thermally isolated. A representation of the computational domain Ω and its measures considered in this work are given in Figure 1.

In this work, the activity A_i registered at a point $(r, z) \in \Omega_F$ at time $t > 0$ is given by:

$$A_i(r, z, t) = A_i(r, z, 0) \exp\left(- \int_0^t \kappa_i(P(\sigma), T(r, z, \sigma)) \mathrm{d}\sigma \right), \tag{3}$$

where κ_i is the inactivation rate of the enzyme ($i = 1$) or the vitamin ($i = 2$) given by the following combination of the Arrhenius and Eyring equations (see, e.g., Denys

et al. (2000)):

$$\kappa_i(P,T) = \kappa_{\mathrm{ref},i} \exp\left(-B_i\left(\frac{1}{T} - \frac{1}{T_{\mathrm{ref},i}}\right)\right) \exp\left(-C_i(P - P_{\mathrm{ref},i})\right); \tag{4}$$

$T_{\mathrm{ref},i}$ (K) and $P_{\mathrm{ref},i}$ (MPa) are reference values for the temperature and the pressure, respectively; $\kappa_{\mathrm{ref},i}$ (min^{-1}) is the inactivation rate at these reference conditions; B_i (K) and C_i (MPa^{-1}) are temperature and pressure dependence parameters. Furthermore, the initial activity $A_i(r,z,0)$ is set to one, which implies that the resulting activity $A_i(r,z,t)$ is within [0, 1] and represents the percentage value of the initial activity that is retained. Thus, the values of the objective functions f_1 and f_2 are in the interval [0, 1].

Therefore, evaluating those objective functions requires the numerical simulation of the HPT treatment by using a heat transfer model for describing the evolution of the pressure and the temperature inside the HPT device. To do so, the following two-dimensional heat transfer equations are considered:

$$\begin{cases} \rho C_p \dfrac{\partial T}{\partial t} - \dfrac{1}{r}\dfrac{\partial}{\partial r}\left(rk\dfrac{\partial T}{\partial r}\right) - \dfrac{\partial}{\partial z}\left(k\dfrac{\partial T}{\partial z}\right) = \alpha \dfrac{\mathrm{d}P}{\mathrm{d}t} T & \text{in} \quad \Omega \times (0, t_{\mathrm{f}}), \\ k\dfrac{\partial T}{\partial \mathbf{n}} = 0 & \text{on} \quad \Gamma \backslash (\Gamma_{\mathrm{r}} \cup \Gamma_{\mathrm{left}}) \times (0, t_{\mathrm{f}}), \\ k\dfrac{\partial T}{\partial \mathbf{n}} = h(T_{\mathrm{amb}} - T) & \text{on} \quad \Gamma_{\mathrm{left}} \times (0, t_{\mathrm{f}}), \\ T = T_{\mathrm{r}} & \text{on} \quad \Gamma_{\mathrm{r}} \times (0, t_{\mathrm{f}}), \\ T(0) = T_0 & \text{in} \quad \Omega, \end{cases} \tag{5}$$

where $\rho = \rho(T,P)$ ($\mathrm{Kg\,m}^{-3}$) is the density; $C_p = C_p(T,P)$ ($\mathrm{J\,Kg}^{-1}\mathrm{K}$) is the heat capacity; $k = k(T,P)$ ($\mathrm{W\,m}^{-1}\mathrm{K}^{-1}$) is the thermal conductivity; $\alpha = \alpha(T,P)$ (K^{-1}) is the thermal expansion coefficient; T_{amb} (K) is the ambient temperature; h ($\mathrm{W\,m}^{-2}\mathrm{K}^{-1}$) is the heat transfer coefficient; T_{r} (K) is the refrigeration temperature; T_0 (K) is the initial temperature in the whole domain; and $\mathbf{n}$ is the outward unit normal vector.

For a detailed explanation about System 5 and the computational domain, see (Ferrández et al. 2019).

3. The original optimization algorithm

The Weighting Achievement Scalarizing Function Genetic Algorithm (Ruiz, Saborido, and Luque 2015), shortly called WASF-GA, is a multi-objective evolutionary optimization algorithm based on a population of points, named as individuals, to which some genetic operators inspired by the Darwin theory of evolution are applied. In this way, it consists of an iterative procedure where, at each step, a new offspring population is generated from the previous-step population using these genetic operators. Then, among the individuals belonging to both populations, a selection mechanism is applied to decide which individuals will survive composing the next-step population. Since WASF-GA is a preference-based algorithm, this survival selection is performed according to the closeness of their objective values to certain predetermined values expressing the preferences of the person solving the problem. Consequently, the resulting

set of solutions is an approximation to a specific region of the Pareto optimal front referred to as the region of interest.

In addition to the multi-objective function $\mathbf{f} = (f_1, \ldots, f_m)$, the input parameters required by WASF-GA are the following ones: (i) the size N of the population of individuals, (ii) the total number $h_{\max}$ of iterations to be considered in the evolutionary procedure, (iii) the values p_m, d_m, p_c, d_c for the probabilities and the distributions of the genetic operators (mutation and crossover), (iv) a reference point $\mathbf{q} = (q_1, \ldots, q_m)$ giving the preferred values for the m objective functions, and (iv) a sample of N_μ weight vectors $W = \{\mu^1, \ldots, \mu^{N_\mu}\}$.

In WASF-GA, as usually done when considering meta–heuristic algorithms, the initial population of N individuals is randomly generated in the search space. Then, at each iteration, three main stages are performed:

(1) *Reproduction*: The individuals of the previous-step population (called *parent population*) are matched between them for applying the crossover operator by pairs generating new individuals, which are later modified by the mutation providing the *offspring population*. Then, the *joint population* is built by the union of both parent and offspring populations.

(2) *Classification*: Next, the classification is performed separating the individuals from this joint population into several groups, called fronts. To do that, in WASF-GA, the following Wierzbicki's Achievement Scalarizing Function (ASF) (Ruiz, Saborido, and Luque 2015) is considered for the given reference point $\mathbf{q}$ and the set of weight vectors $W = \{\mu^1, \ldots, \mu^{N_\mu}\}$:

$$s^j(\mathbf{x}) = s(\mathbf{q}, \mathbf{f}(\mathbf{x}), \mu^j) = \max_{i=1,\ldots,m} \{\mu_i^j (f_i(\mathbf{x}) - q_i)\} + \eta \sum_{i=1}^{m} \mu_i^j (f_i(\mathbf{x}) - q_i).$$

Notice that the first term of this expression is based on the L_∞ distance and the other term is used to guarantee the efficiency of the solutions using an augmentation coefficient (here, $\eta = 0.001$). Then, each front is filled as follows. First, among the unclassified individuals, the one providing the lowest value of the ASF with the first weight vector is chosen to be the first individual of the front. After that, if there are still any unclassified individuals, among them, the one with the lowest value of the ASF considering the second weight vector is now selected and copied into the front. This process is repeated until all the individuals are classified or until the front contains exactly as many individuals as weight vectors in the sample (i.e., N_μ). When the latter occurs and there are still individuals without classifying, a new front is created and filled by applying the same procedure.

(3) *Selection*: The selection consists of building the parent population for the next iteration or for the outcome (if the number of iterations is equal to $h_{\max}$). This new or final population is formed by the individuals of the first fronts until reaching the total size of N individuals. If there is any front that cannot be included completely, then its individuals having the lowest ASF values are chosen. In this work, the population and the weight vector sample are assumed to have the same size: $N = N_\mu$. Therefore, the new population built at the end of each iteration is composed exactly by the individuals from the first front.

Consequently, the WASF-GA population can be thought of as a set of points, each one solving the minimization of a mono-objective function which is the ASF with a

determined weight vector. Indeed, at each iteration $h < h_{\max}$, for each weight vector $\mu^j = (\mu_1^j, \ldots, \mu_m^j) \in W$, with $j = 1, \ldots, N_\mu$, the algorithm tries to obtain a point closer to the root of the ASF weighted using μ^j and denoted as $s^j(\mathbf{x})$. Then, the individual j in the population is the one that minimizes $s^j(\mathbf{x})$.

4. The proposed improvements

This section is aimed to explain some ideas and techniques that have been added to the WASF-GA implementation in order to improve its performance when considering a low number of iterations.

Two categories have been distinguished depending on if the mechanisms affect the reproduction phase or the classification stage. Notice that the selection procedure of WASF-GA remains unchanged (i.e., as explained in Section 3). For the sake of completeness and usability of the notation, the original reproduction and classification methods have also been included in the enumerations at the first position.

In addition to the novelties applied at the reproduction and classification stages, an external list has been introduced with the aim of increasing the number of non-dominated points in the final population obtained with a short number of iterations. This external list is created at the beginning of the iterative process containing only the non-dominated points of the initial population. Then, each time a new population is generated, the external list is updated to include those new individuals which are non-dominated and remove those old individuals which are dominated by the new ones.

4.1. *Reproduction stage*

At this stage, four alternatives have been considered: the WASF-GA original reproduction method and three variations involving different techniques. The main difference among these alternative methodologies is the composition of the population named *joint population*. Notice that this population is created at the end of the reproduction phase and that the individuals included in it will be the ones that the algorithm will consider for the next classification and selection procedures.

- $Reproduction_1$: The joint population is the union of the parent population and the offspring population created by applying crossover and mutation.
- $Reproduction_2$: In addition to the parent and offspring populations, the *joint population* also contains two more populations:
  - The population of individuals obtained by improving the dominated solutions.

    For this strategy, in addition to the external list, another list is built including those points which have not been included in the external list or which have been deleted from it because they are dominated. Then, for each one of those dominated points:
    (1) Look for the non-dominated point in the external list that is closest to it in the objective space using the Euclidean distance.
    (2) Compute a new point randomly chosen in the segment that unites them in the search space.
  - The secant population.

    This mechanism is inspired by a similar technique that was successfully

applied in a multilayer algorithm in (Ivorra, Mohammadi, and Ramos 2015) for accelerating the convergence of genetic algorithms.

The main idea is to generate at each iteration a new set of individuals, called *secant population*, by considering the populations obtained at the two previous iterations. As explained in Section 3, the WASF-GA population can be seen as a set of points, each one solving the minimization of a mono-objective function $s^j(\mathbf{x})$, $j = 1, \ldots, N_\mu$. Since the secant method aims to approximate the zeros of a function without using any information about its gradient, it is applied here at each iteration $2 < h < h_{\max}$ to obtain a set of new individuals $\{\mathbf{x}^{h,j}\}_{j=1,\ldots,N_\mu}$ as follows:

$$\mathbf{x}^{h,j} = \text{proj}_S\left(\mathbf{x}^{h-1,j} - s^j(\mathbf{x}^{h-1,j})\frac{\mathbf{x}^{h-1,j} - \mathbf{x}^{h-2,j}}{s^j(\mathbf{x}^{h-1,j}) - s^j(\mathbf{x}^{h-2,j})}\right),\ j = 1, \ldots, N_\mu,\ h = 3, \ldots, h_{\max},$$

where $\mathbf{x}^{h-1,j}$ and $\mathbf{x}^{h-2,j}$ belong to the previous-steps populations P^{h-1} and P^{h-2}, respectively. Those individuals were selected because they minimize the function $s^j(\mathbf{x})$ at the steps $h-1$ and $h-2$, respectively. Notice that proj_S is a projection operator of the point into the search space S for guaranteeing that the new point remains in S, i.e., $\text{proj}_S(\mathbf{x}) = (\min(\max(x_1, lb_1), ub_1), \ldots, \min(\max(x_n, lb_n), ub_n))$, where $(lb_1, \ldots, lb_n)$ and $(ub_1, \ldots, ub_n)$ are the lower and upper bounds of the decision variables.

- *Reproduction*$_3$: As stated in literature, local search procedures aim to improve the convergence to the optimum. In this case, the multi-objective version of the SASS method, which is called MOSASS, has been used. MOSASS was proposed in (Redondo, Fernández, and Ortigosa 2017). Among its advantages, it must be highlighted that during the search process many interesting points are usually generated close to the individual to which the method is applied. These points will be introduced in the external list of non-dominated points or in a new list containing the dominated individuals. Nevertheless, as a counterpart, this implies an increase in the number of evaluations. Then, in the proposed implementation, MOSASS has been applied only to the individual of the current population that is the closest one to the reference point in the objective space, considering the Euclidean distance.

  MOSASS is based on an iterative procedure aimed at obtaining an improved point starting from an initial one. In addition to this initial point, it receives as input the normalized maximum radius $\sigma_{ub} \in \mathbb{R}$ allowed for modifying the variables to avoid the generation of new points outside the region of interest. In this proposal, the value of σ_{ub} is the normalized distance between the initial point and the point of the parent population being the furthest from it in the search space. There are other parameters that are fixed to pre-determined values such as: the maximum number of iterations $ic_{\max}$ and of consecutive failures $Maxfcnt$ used by MOSASS as stopping rule; the threshold number of consecutive failure trials F_{cnt} and of consecutive successful trials S_{cnt}, and the contraction ct and expansion ex coefficients. More precisely, $ct \in (0, 1)$ and $ex > 1$ are constants that regulate the intensification of the local search multiplying the search radius to reduce it or to extend it, respectively.

  In MOSASS, at each iteration, the starting individual $\mathbf{x}$ is perturbed by adding a multivariate Gaussian random vector $\xi = N(\mathbf{b}, \sigma I) \in \mathbb{R}^n$, where $\mathbf{b} \in \mathbb{R}^n$ is a

normalized bias vector directing the search and $\sigma \in \mathbb{R}$ is the radius of the most preferred search region. At the beginning of the iterative procedure, $\mathbf{b} = \mathbf{0}$ and $\sigma = \sigma_{ub}$. Then, the new point $\mathbf{x}+\xi$ is evaluated and, if it dominates the original individual $\mathbf{x}$, it replaces $\mathbf{x}$ for the next iteration. On the contrary, $\mathbf{x}+\xi$ is tried to be included in the external list but it will be accepted in this list only if it is not dominated by any individual on the list, otherwise, it will be transferred into the dominated list. This dominated list also receives the individuals that may be removed from the external list because they are dominated by the new individual $\mathbf{x}+\xi$. If $\mathbf{x}+\xi$ cannot be stored in the external list because it is dominated, then the same procedure is repeated but now with $\mathbf{x}-\xi$. First, it is tested if it dominates $\mathbf{x}$ and, if not, it is tried to be included on the external list. At each iteration, a new Gaussian random vector is generated but updating the deviation σ that specifies the size of the sphere that most likely contains the perturbation vector, and the bias term $\mathbf{b}$ indicating the center of this sphere. They are updated according to the number of successful trials achieved at the previous iterations. A trial point is considered successful if it dominates the previous point or if it is finally included in the non-dominated external list. If more than *Scnt* consecutive successes have been reached, the perturbation radius is increased $\sigma = ex \cdot \sigma$. On the contrary, if more than *Fcnt* consecutive failures have occurred, it is decreased $\sigma = ct \cdot \sigma$. Furthermore, if the perturbation is too small or too big, it will be reset to σ^{ub}. In the first case, the bias vector is also reset to zero. For more details about MOSASS, see (Lancinskas, Ortigosa, and Žilinskas 2013) and (Redondo, Fernández, and Ortigosa 2017).

Finally, in *Reproduction*$_3$, the joint population is formed by the offspring population, the individuals belonging to the external list, and the dominated individuals.

- *Reproduction*$_4$: Here, an alternative method for generating the offspring population is proposed consisting of building an *advanced population*. This advanced population is generated using the parent population, i.e., the population of the previous step, and the external list (see Algorithm 1). For each weight vector μ^j, two points are selected: (i) the point $\mathbf{x}^j_{\text{el}}$ of the external list which minimizes the ASF weighted using μ^j and, (ii) analogously, the point $\mathbf{x}^j$ of the parent population minimizing the same ASF with μ^j. Then, a new point $\mathbf{x}^j_{\text{ad}}$ is randomly generated in the neighborhood between both points $\mathbf{x}^j_{\text{el}}$ and $\mathbf{x}^j$ as follows:

$$\mathbf{x}^j_{\text{ad}} = \mathbf{x}^j_{\text{el}} + \lambda(\mathbf{x}^j_{\text{el}} - \mathbf{x}^j), \tag{6}$$

where $\lambda = (\lambda_1, \ldots, \lambda_n)$ is a vector of random numbers. Next, this new point $\mathbf{x}^j_{\text{ad}}$ is included in the advanced population. The idea is to achieve new points closer to those non-dominated points in order for them to have more probabilities to be non-dominated also. In some cases, modifying all the n decision variables, as in (6), can lead to an objective value far from the neighborhood. Then, for generating the new point, it may be more efficient to maintain some values of the individual of the external list (i.e, $x^j_{\text{ad},i} = x^j_{\text{el},i}$, for some $i = 1, \ldots, n$). Thus, a fixed probability value $p_{\text{ad}} \in (0, 1]$ has been considered, which is used to make the decision of when to apply the modification to a certain variable (Lancinskas, Ortigosa, and Žilinskas 2013). More precisely, for each variable $x_{\text{ad},i}$, $i \in \{1, \ldots, n\}$, a random number *rnd* is generated and, if $rnd \leq p_{\text{ad}}$, then the expression (6) is applied, otherwise, $x_{\text{ad},i} = x_{\text{el},i}$ (see Algorithm 2).

Algorithm 1 *calculateAdvancedPop*

calculateAdvancedPop *in: population, externalList out: advancedPop*
Create *advancedPop* as an empty list of individuals;
for $j = 0$ **to** $problem.getNumberOfWeightVectors()$ **do**
  double $ASFmin = ASF.evaluate(externalList.get(0), \mu^j)$;
  int $indexASFmin = 0$;
  for $l = 0$ **to** $externalList.size()$ **do**
    $aux = ASF.evaluate(externalList.get(l), \mu^j)$;
    if $aux < ASFmin$ **then**
      $indexASFmin = l$;
    end if
  end for
  $popIndiv = population.get(j)$;
  {Notice that the individual j in the population is the one that minimizes the ASF using the weight vector μ^j}
  $externalListIndiv = externalList.get(indexASFmin)$;
  $advancedIndividual = calculateAdvancedIndividual(popIndiv, externalListIndiv)$;
  $advancedPop.add(advancedIndividual)$;
end for
$evaluatePopulation(advancedPop)$;
return *advancedPop*

Algorithm 2 *calculateAdvancedIndividual*

calculateAdvancedIndividual *in: popIndividual, externalListIndividual out: advancedIndividual*
Create *advancedIndividual* as an individual with $problem.getNumberOfVariables()$ variables;
double $probab = 1/problem.getNumberOfVariables()$;
double $x0, x1, x2, lambda$;
boolean $isthesame = true$;
while $isthesame == true$ **do**
  for $i = 0$ **to** $problem.getNumberOfVariables()$ **do**
    if $rnd.nextDouble() <= probab$ **then**
      $x0 = popIndividual.getVariableValue(i)$;
      $x1 = externalListIndividual.getVariableValue(i)$;
      $lambda = rnd.nextDouble()$;
      $x2 = x1 + lambda * (x1 - x0)$;
      $x2 = Double.max(x2, getLowerBound(i))$;
      $x2 = Double.min(x2, getUpperBound(i))$;
      $advancedIndividual.setVariableValue(i, x2)$;
      $isthesame = false$;
    end if
  end for
end while
return *advancedIndividual*

In order to maintain a good compromise between exploitation of the already

known non-dominated points and exploration of the search space, the advanced individuals' generation and the generation of the offspring population by means of the crossover and mutation operators, have been alternated. More precisely, the former has been used at the even iterations and the latter at the odd iterations. Notice that, as the advanced population has as many individuals as weight vectors N_μ and it has been set $N_\mu = N$, the number of evaluations does not increment, i.e., the resulting reproduction method called $Reproduction_4$ maintains the same number of evaluations per iteration than WASF-GA. This is very important for problems such as the one considered here, in which the evaluation of the objective function is computationally expensive.

Remark 1. The number of individuals per iteration in the joint population is not the same for all the variants. In particular, for $Reproduction_2$ and $Reproduction_3$, it is not a constant number. Therefore, to perform a fair comparison, the stopping criterion is formulated in terms of the number of evaluations (see Section 5.1). Furthermore, since this work is focused on problems whose objective functions evaluations are computationally expensive, the reproduction, classification and selection procedures have a negligible cost in comparison to a single function evaluation.

4.2. *Classification stage*

The main core of the WASF-GA classification procedure detailed in Section 3 remains for all the studied versions. The only distinctive feature that has been modified is the choice of the population to which the classification will be applied.

- $Classification_A$: As in the original WASF-GA, those individuals belonging to the joint population will be considered for their classification into fronts. Since the selection procedure explained in Section 3 is based on preferences without any dominance criterion, this kind of classification can lead to select some dominated individuals for the final population.
- $Classification_B$: At the first iterations where the size of the external list is lower than the number of required individuals in the final population (i.e., N), the joint population is considered, as in $Classification_A$. Otherwise, when the external list has enough individuals (i.e., more than N), only these individuals belonging to the external list are considered for the classification and selection procedures. As a consequence, the final population is guaranteed to be formed only by non-dominated individuals.

5. Computational experiments

According to Section 4, the different variants of WASF-GA that have been implemented and tested are denoted as WASF-GA$_{rc}$, where the sub-index $r = 1, 2, 3, 4$ refers to the considered reproduction method and $c = A, B$ indicates one of the proposed classification options. Hence, the original WASF-GA corresponds to WASF-GA$_{1A}$. To compare their solutions, the execution of each variant WASF-GA$_{rc}$ has been performed 20 times considering the food processing problem (2). Then, the quality of the obtained Pareto front approximations has been analyzed by using the average and the standard deviation values of some state-of-the-art effectiveness measures.

The experiments were performed on a cluster node which has 8 Intel Xeon E7

8860v3 3.2 GHz processors with 16 cores each and 2.3 TB of RAM. The shared-memory parallel version of WASF-GA proposed in (Ferrández et al. 2018) has been used adapting it to hold all these WASF-GA$_{rc}$ variants. Our implementations have been built in the framework of the optimization package jMetal (Durillo and Nebro 2011), taking as reference the existing implementation of WASF-GA available there [1].

The target multi-objective problem is given by (2) where ten decision variables have been considered: the initial and refrigeration temperatures T_0, $T_r \in$ [10,50] °C, and eight pressure variations $\Delta P_1, \ldots, \Delta P_8 \in$ [-250, 250] MPa to be applied at eight times t_i, $i = 1, \ldots, 8$, uniformly distributed in the processing interval $[t_0, t_f]$=[0, 900] seconds.

For this industrial problem, evaluating the objective functions implies the numerical simulation of the HPT treatment with the temperature and pressure provided by those decision variables. This numerical simulation consists in solving the heat transfer system (5) coupled with the activity equation (3) formulated in Section 2. To do so, the Finite Element Method (FEM), which is a common technique to approximate the solution of partial differential equations models, has been used. It is based on describing the unknown fields as polynomials of a determined degree at each point of a spatial mesh that covers the whole computational domain (Ramos 2012). Despite considering the simplifications detailed in Section 2, the computational cost of the numerical model is quite high. Looking for a compromise between computational cost and accuracy of the solution, a spatial mesh with 2881 elements has been considered for the discretized domain. The computational time of a single evaluation is 46.5 seconds on average using this mesh in the cluster node described before with a single core per evaluation. Considering a mesh with a lower number of points may lead to a poor description of the HPT treatment not being able to capture the effects of the pressure and temperature over the micro-organism activities.

In these experiments, the considered HPT treatment involves the following materials: tylose, which is a solid gel similar to meat, in the food sample Ω_F; water, as pressurizing fluid in Ω_P; rubber, for the cup of the food sample holder Ω_C, and steel for the external vessel Ω_S. The thermo-physical properties of tylose and water have been computed depending on temperature and pressure using a shifting approach (Otero et al. 2006); those of rubber and steel are assumed to be constants (Infante et al. 2009): $\rho = 1110$ Kg m^{-3}, $C_p = 1884$ J Kg^{-1}K, $k = 0.173$ W m^{-1}K^{-1} in Ω_C, and $\rho = 7833$ Kg m^{-3}, $C_p = 465$ J Kg^{-1}K, $k = 55$ W m^{-1}K^{-1} in Ω_S. For the parameter α, the approach and values proposed in (Otero, Molina-García, and Sanz 2002) have been considered. The convection parameters on boundary Γ_{left} have been set to $h = 28$ W m^{-2}K^{-1} and $T_{\text{amb}} = 19.3$ K.

The accomplished experiments focus on reducing the activity of the enzyme called BSAA (Denys et al. 2000) and on retaining the activity of the vitamin C (Verbeyst et al. 2013). Concerning the inactivation rates, for BSAA the parameters have been set to $B_1 = 10097$ K and $C_1 = -8,7$e-4 MPa^{-1} with $T_{\text{ref},1} = 313$ K, $P_{\text{ref},1} = 500$ MPa, and $\kappa_{\text{ref},1} = 3.9$e-2 min^{-1}; while for the vitamin C they have been set to $B_2 = 9071.4$ K and $C_2 = -5.55$e-3 MPa^{-1} with $T_{\text{ref},1} = 373$ K, $P_{\text{ref},1} = 700$ MPa, and $\kappa_{\text{ref},1} = 7.92$e-2 min^{-1}.

Considering the model explained in Section 2 and the set of parameters given above, the third objective function f_3 takes values within [10, 80]°C during the experiments.

According to the decision maker preferences, who in this case is the food engineer, the following unachievable reference point (0.0, 1.0, 30.0) has been considered. Re-

[1] https://github.com/jMetal/jMetal

call that the activities take values in [0.0, 1.0]. Then, the first component indicates the ideal scenario where the undesired enzyme is completely inactivated. The second component reflects the objective of retaining all the vitamin activity. Finally, the last value corresponds to the temperature preferences and it is fixed to 30°C close to the ambient temperature since high temperatures may damage the organoleptic properties of the food (e.g., its flavor).

5.1. *Optimization settings*

According to previous works based on a similar industrial problem (Ferrández et al. 2018, 2019), a population size of at least $N = 200$ individuals is needed for the optimization algorithm. Considering fewer individuals in the population, the obtained set of optimal points conforming the Pareto front approximation may not sufficiently cover the region of interest. As a consequence, it cannot be guaranteed that some required quality scenarios for the vitamin conservation and the enzyme reduction will be attended.

Concerning the maximum number of iterations, the higher its value, the higher the quality of the Pareto front approximation in both senses, the better distributed the points and the closer they are from the real Pareto optimal front. However, in view of the required amount of individuals and the high computational cost of their evaluation, to solve the industrial problem in an affordable time, a maximum number of iterations of $h_{\max} = 50$ has been considered.

The stopping criterion of the original algorithm WASF-GA$_{1A}$ is based on the number of iterations. More precisely, its evolutionary process is running until the value $h_{\max}$ is reached. Since WASF-GA$_{1A}$ consumes as many evaluations per iteration as individuals in the population, when it finishes, the total number of evaluations done is $N \cdot h_{\max}$. Nevertheless, for some of the proposed variants of WASF-GA$_{1A}$, the number of evaluations at each iteration is not a constant number. In particular, it occurs for WASF-GA$_{2c}$, whose reproduction method involves the improvement of the individuals that are dominated, and for WASF-GA$_{3c}$, which is based on the use of the local search technique MOSASS, no matter the classification method $c = A$ or $c = B$ used. Thus, for the sake of the fairness of their comparison, the following stopping rule is considered: at each iteration h, the algorithm stops if $cont\_eval_{h-1} + N \geq N \cdot h_{\max}$, where $cont\_eval_{h-1}$ is the cumulative number of evaluations consumed until the $h-1$ iteration.

Next, the used settings concerning the remaining parameters are included. For them, a fine-tuned analysis falls out of the scope of this article, nevertheless, the most extended values used in literature have been selected (Solis and Wets 1981; Ruiz, Saborido, and Luque 2015). Regarding the genetic operators, the Simulated Binary Crossover (SBX) and the polynomial mutation have been employed with probabilities $p_c = 0.9$ and $p_m = 0.1$, respectively, and a distribution of 20 for both. In particular, for those parameters, a more exhaustive study was performed in (Ferrández et al. 2019) and those values proved to be the ones leading to the higher quality Pareto front approximations.

The MOSASS parameters used in the implementation of WASF-GA$_{3c}$ (with $c = A, B$) are the ones recommended by Solis and Wets (1981) and successfully validated by Lancinskas, Ortigosa, and Žilinskas (2013) and Redondo, Fernández, and Ortigosa (2017). The coefficient values used for updating the bias term $\mathbf{b}$ are 0.4, 0.2 and 0.5, and the values for the threshold number of successes and failures involved at the

σ computation are $Scnt = 5$ and $Fcnt = 3$, respectively. For the contraction and expansion coefficients, also the recommended values $ct = 0.5$ and $ex = 2$, respectively, have been considered. In order to avoid an excessive increment of the number of evaluations derived from the application of the MOSASS mechanism, the maximum number of consecutive failures for its stopping rule has been fixed to $Maxfcnt = 5$ for each WASF-GA$_{3c}$ iteration. In both methods (i.e., MOSASS in WASF-GA$_{3c}$ and the generation of the advanced individuals in WASF-GA$_{4c}$), a probability $p_{\text{ad}} = 1/n$, where n is the number of variables, has been considered for deciding when the decision variables are modified or not (Lancinskas, Ortigosa, and Žilinskas 2013).

5.2. *Quality indicators*

To measure the effectiveness of the algorithms, the most extended methodology in literature has been used, which is the *quality indicator* procedure. It consists of quantifying the quality of each Pareto front approximation and, then, analyzing the resulting numbers for their comparison. More precisely, let Ω be the set of all Pareto front approximations, a *unary quality indicator* is a function $I : \Omega \to \mathbb{R}$ which assigns to each Pareto front approximation $PFA \in \Omega$ a real value $I(PFA)$.

As the studied algorithms are heuristics, every particular instance has been run 20 times. Thus, for each algorithm rc, with $r \in \{1, \ldots, 4\}$ and $c \in \{A, B\}$, 20 different Pareto set approximations $PS^1_{rc}, \ldots, PS^{20}_{rc}$ have been obtained. All these resulting sets for all the algorithms compose the set of all the Pareto-set approximations denoted as SPS. Since the real Pareto front is required for the computation of some quality indicators, in problems where it is unknown, an approximated reference set RS is used. In this work, RS has been generated by merging all the individuals of the SPS Pareto-set approximations and obtaining their images in the objective space. Then, the solutions which are non-dominated have been selected. Furthermore, for a fairness contribution of all the objectives, RS and all the Pareto front approximations have been normalized before computing the quality indicator values (Ferrández et al. 2019). The following standard normalization has been used:

$$f_i(\mathbf{x})' = \frac{f_i(\mathbf{x}) - f_i^{(\min)}}{f_i^{(\max)} - f_i^{(\min)}}, \qquad i = 1, \ldots, m,$$

where $f_i^{(\min)}$ (resp. $f_i^{(\max)}$) denotes the minimum (resp. maximum) value of f_i when considering all the solutions in SPS.

In general, there exist three kinds of quality indicators depending on what feature is measured: proximity, diversity and global indicators. The former focuses on computing the distance between the real Pareto front (or, in this case, RS) and the approximation obtained with the optimization algorithm. Therefore, since the main goal here is comparing the algorithms in terms of their convergence, some proximity indicators have been used. In particular, two of those measures have been employed, namely, the *additive epsilon* (Zitzler et al. 2003) and the *inverted generational distance plus* indicator (IGD+) (Ishibuchi et al. 2015). Additionally, to evaluate the global quality, the well-known *hypervolume* indicator has been used. It is based on computing the hypervolume of the piece of the decision space which is weakly dominated by the Pareto front approximation (While, Bradstreet, and Barone 2012; Zitzler and Thiele 1998). Since the considered algorithms are devoted to approximate the part of the Pareto front preferred by the user, it has been calculated the *hypervolume in the region of*

interest as proposed in (Ruiz, Saborido, and Luque 2015). To bound the space for that hypervolume computation, the point that is dominated by all the points in the region of interest must be provided. In this work, this point $\mathbf{z}^{\text{nad}}$ has been obtained by considering all the approximations of the Pareto front $f(SPS)$ as follows:

$$\mathbf{z}^{\text{nad}} = \left(\max_{\mathbf{x} \in SPS} f_1(\mathbf{x}), \ldots, \max_{\mathbf{x} \in SPS} f_m(\mathbf{x}) \right).$$

On the one hand, given a Pareto set approximation PS, the additive epsilon indicator is calculated as the minimum distance by which $\mathbf{f}(PS)$ needs to be moved in each dimension of the objective space such that the reference set RS is weakly dominated. Mathematically, it can be expressed as follows:

$$I_{\epsilon+}(\mathbf{f}(PS)) = \min_{\epsilon \in \mathbb{R}} \left\{ \forall \mathbf{b} \in RS, \quad \exists \mathbf{a} \in PS : \right. \\ \left. \frac{f_i(\mathbf{a}) - f_i^{(\min)}}{f_i^{(\max)} - f_i^{(\min)}} - \epsilon \leq \frac{f_i(\mathbf{b}) - f_i^{(\min)}}{f_i^{(\max)} - f_i^{(\min)}}, \quad \forall i \in \{1, \ldots, m\} \right\}. \tag{7}$$

On the other hand, the inverted generational distance metric plus (IGD+), which was recently proposed by Ishibuchi et al. (2015), uses the following formula:

$$\text{IGD+}(\mathbf{f}(PS)) = \frac{1}{|RS|} \left(\sum_{l=1}^{|RS|} (d_l^+)^p \right)^{1/p}$$

where a new modified distance d^+ is employed instead of the Euclidean distance for calculating the distance d_l^+ from $\mathbf{b}_l \in RS$ to its nearest objective vector in $\mathbf{f}(PS)$. For a minimization problem, this modified distance is defined as:

$$d^+(\mathbf{a}, \mathbf{b}) = \sqrt{(\max\{a_1 - b_1, 0\})^2 + \cdots + (\max\{a_m - b_m, 0\})^2},$$

for two objective vectors $\mathbf{a} \in PS$ and $\mathbf{b} \in RS$. In this work, it has been used $p = 1$ as it was originally proposed in (Ishibuchi et al. 2015).

Notice that all those three indicators that have been employed are *Pareto compliant*. It means that whenever a Pareto-set approximation A is preferable to a Pareto-set approximation B with respect to weak Pareto dominance, the indicator value for $\mathbf{f}(A)$ should be at least as good as the indicator value for $\mathbf{f}(B)$. To compute them, the implementations available at the optimization package jMetal (Durillo and Nebro 2011) have been used.

6. Results and discussion

As a preliminary step, not detailed here, the most promising WASF-GA variants were selected to focus the analysis on those most competitive methods. For instance, $Reproduction_2$ and $Reproduction_3$ combined with $Classification_B$ lead to a stagnation of the population, since those reproduction techniques are mainly based on managing dominated points meanwhile the cited classification only considers non-dominated individuals for building the population. Therefore, in order to present a

Table 1. Results of the Wilcoxon test considering the values of the following indicators: additive epsilon (Epsilon), inverted generational distance plus (IGD+), hypervolume (HV) and number of non-dominated solutions (NumND). "–" indicates that the hypothesis of a significant difference between the algorithms cannot be accepted; "▲" represents that they are significantly different and that the average of the algorithm corresponding to the row is better than the algorithm of the column, and "▽" means that the algorithm indicated on the column is significantly better, on average, than the one of the row for the considered indicator.

	Epsilon, IGD+, HV, NumND																			
	$1A$				$2A$				$3A$				$4A$				$4B$			
$1A$					–	–	–	–	▽	▽	–	–	–	▲	▲	▽	▽	▽	▽	▽
$2A$	–	–	–	–					–	–	–	–	–	–	▲	▽	–	–	▽	▽
$3A$	▲	▲	–	–	–	–	–	–					▲	▲	▲	▽	–	–	–	▽
$4A$	–	▽	▽	▲	–	–	▽	▲	▽	▽	▽	▲					▽	▽	▽	▽
$4B$	▲	▲	▲	▲	–	–	▲	▲	–	–	–	▲	▲	▲	▲	▲				

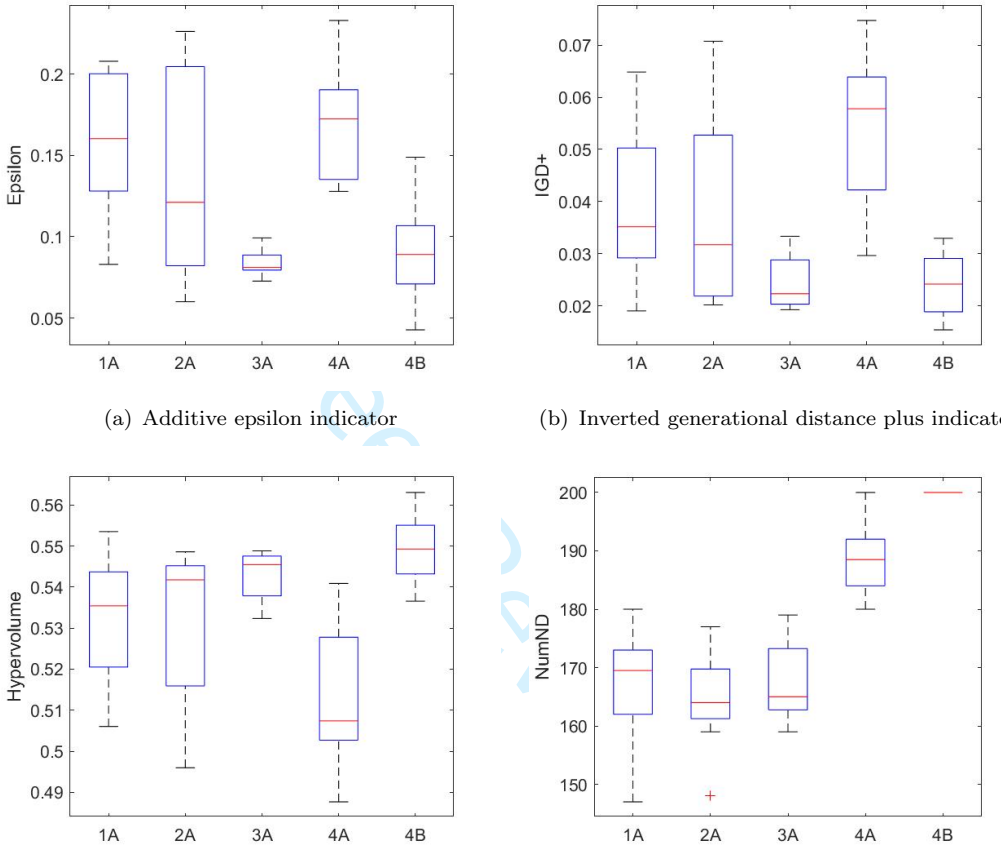
concise comparison, the variants involving those combinations have not been included ahead in the discussion.

Next, an statistical analysis has been performed to determine if the proposed variants are significantly different among them or, contrarily, they come from the same probability distribution showing variations because of the stochasticity of the algorithms. To do that, the Wilcoxon test (Wilcoxon 1945) has been applied, since it is widely used in literature for the pairwise comparison of evolutionary algorithms and it does not need any assumption about the normality of the distributions (Derrac et al. 2011; Ruiz, Saborido, and Luque 2015). In this analysis, a confidence level of 95% has been considered. It means that the distributions of the results for two compared algorithms are accepted to be different if the p-value obtained by the Wilcoxon test is lower than 0.05. In Table 1, the results of the Wilcoxon test are summarized for all the considered quality indicators. For the sake of completeness, the distributions of the quality measures values for the different variants are shown in Fig. 2.

More precisely, Fig. 2(a) and Fig. 2(b) corresponds to the epsilon and the IGD+ indicators, respectively. Notice that for these metrics the lower the value, the better the quality of the Pareto front approximation. Therefore, the versions WASF-GA$_{3A}$ and WASF-GA$_{4B}$ based on the introduction of the MOSASS technique and the advanced population with classification applied to the external list, respectively, clearly outperform the original algorithm WASF-GA$_{1A}$. It means that using those mechanisms to solve the considered industrial problem, an enhancement in the convergence is achieved. Moreover, according to the Wilcoxon test, both WASF-GA$_{3A}$ and WASF-GA$_{4B}$ are significantly different from WASF-GA$_{1A}$ regarding the epsilon and the IGD+ measures (see Table 1), so they provide a meaningful improvement.

For the hypervolume global indicator, the higher the value, the better the solution. Thus, as can be seen in Fig. 2(c), the algorithm WASF-GA$_{4B}$ using the advanced population for the reproduction phase and the external list for the classification procedure, also overcomes the basic version WASF-GA$_{1A}$ in terms of global quality. Furthermore, the Wilcoxon test confirms that this improvement is not due to the randomness of the algorithms, as WASF-GA$_{4B}$ presents a distribution of hypervolume values significantly different from WASF-GA$_{1A}$ (see Table 1).

According to those results for the considered industrial problem, it seems that the goal of improving WASF-GA has been achieved in terms of convergence and proximity to the reference Pareto front and also in global performance. Now, in previous studies (Ferrández et al. 2019), the authors realized that the population obtained with WASF-GA does not reach the total number of individuals being non-dominated, in



(a) Additive epsilon indicator

(b) Inverted generational distance plus indicator

(c) Hypervolume indicator

(d) Number of non-dominated points in the Pareto front approximation

Figure 2. Distribution of the quality measures values for the different variants: $1A$ corresponds to the original WASF-GA, $2A$ to its version based on improving the dominated individuals and building a secant population, $3A$ to the one using the MOSASS mechanism, $4A$ to the one considering the advanced population, and $4B$ the algorithm also using the advanced population but only considering the points of the external list for the classification procedure. For (a) Epsilon and (b) IGD+, the lower the value, the better the quality of the Pareto front approximation. Contrarily, for (c) Hypervolume and (d) NumND, the higher the value, the better the quality. In these boxplot graphics, the lower and upper bounds of the box represent the first and third quartile, respectively; the line inside the box is the median; the lower and upper limits of the whiskers are the minimum and maximum values, respectively, without considering the outliers points represented by the "+" symbol.

Table 2. Average and standard deviation values of the considered quality indicators for $h_{\max} = 50$ iterations. $1A$ corresponds to the original WASF-GA, $2A$ to its version based on improving the dominated individuals and building a secant population, $3A$ to the one using the MOSASS mechanism, $4A$ to the one considering the advanced population, and $4B$ the algorithm also using the advanced population but only considering the points of the external list for the classification procedure. The best average values for each of the indicators have been highlighted in bold.

rc	Epsilon	IGD+	HV	Number of ND
$1A$	1.60E-01$_{\pm 4.0E-02}$	3.89E-02$_{\pm 1.5E-02}$	5.33E-01$_{\pm 1.4E-02}$	1.68E+02$_{\pm 9.2E+00}$
$2A$	1.34E-01$_{\pm 6.5E-02}$	3.69E-02$_{\pm 1.9E-02}$	5.32E-01$_{\pm 1.9E-02}$	1.65E+02$_{\pm 8.2E+00}$
$3A$	**8.40E-02**$_{\pm 8.6E-03}$	2.45E-02$_{\pm 5.4E-03}$	5.43E-01$_{\pm 6.3E-03}$	1.67E+02$_{\pm 7.4E+00}$
$4A$	1.72E-01$_{\pm 3.9E-02}$	5.39E-02$_{\pm 1.5E-02}$	5.13E-01$_{\pm 1.7E-02}$	1.89E+02$_{\pm 6.1E+00}$
$4B$	9.02E-02$_{\pm 3.0E-02}$	**2.39E-02**$_{\pm 6.0E-03}$	**5.50E-01**$_{\pm 8.2E-03}$	**2.00E+02**$_{\pm 0.0E+00}$

this case $N = 200$. For that reason, here, it has been introduced the variant denoted as $Classification_B$ which only considers the individuals belonging to the external list for the classification into fronts. As a consequence, all the individuals obtained as outcome of WASF-GA$_{4B}$ are non-dominated. As can be seen in Fig. 2(d), in comparison to the other considered versions, this approach produces a great improvement. In this problem, it implies that the solution set provides the food engineer a larger amount of temperature and pressure configurations for the HPT equipment allowing to attend more quality demands maybe from different countries.

In Table 2, the average and standard deviation values of the quality indicators are shown, highlighting in bold the best average values for each of the indicators. It is important to mention that, according to these values, the algorithms WASF-GA$_{3A}$ and WASF-GA$_{4B}$ are not only better on average than the others for the epsilon, IGD+ and hypervolume indicators but they also seem to exhibit a low variability of the results since their standard deviation values are lower.

To sum up, in general, the proposed algorithm WASF-GA$_{4B}$ obtains significantly better results for the food processing problem than the other considered versions for all the analyzed measures. Moreover, it guarantees that all the individuals in the outcome population are non-dominated solutions.

For the sake of fairness, it is worthy to mention that the hypervolume measure depends on the number of non-dominated solutions (While, Bradstreet, and Barone 2012; Zitzler and Thiele 1998). Therefore, in order to complete the comparison among the proposed variants, the performance index known as *set coverage* or C-metric (Zhang and Li 2007) has been computed. For two Pareto front approximations P_1 and P_2, it is calculated as the number of individuals in P_2 which are dominated by at least one point in P_1 divided by the number of individuals in P_2. Thus, it provides the percentage of dominated-by-P_1 solutions in P_2. In Table 3, the averages of the C-metric values of all the pairwise combinations of the proposed variants are reported. According to those results, it can be concluded that the variant WASF-GA$_{4B}$ is the best proposal among all the studied algorithms for the considered problem since it has the highest percentages (on average) of points dominated by its solutions in the other algorithms' outcomes.

Remark 2. It should be recalled that those results have been obtained considering the reference point (0.0, 1.0, 30.0). This unachievable point expresses the food engineers preferences in a general and unconstrained context, allowing them to deal with a wide variety of food quality scenarios using a single Pareto front approximation (Ferrández et al. 2019). Despite it is the reference point recommended for this industrial problem,

Table 3. Averages of the set coverage metric $C(P_1, P_2)$ between the Pareto front approximations P_1 and P_2 for all the pairwise combinations of the considered algorithms. The best values have been highlighted in bold.

P_1 \ P_2	$1A$	$2A$	$3A$	$4A$	$4B$
$1A$	-	26.83	24.71	**30.19**	14.47
$2A$	4.62	-	14.98	13.59	8.08
$3A$	4.46	16.21	-	15.78	8.57
$4A$	7.01	13.97	10.68	-	6.18
$4B$	**16.65**	**30.59**	**33.01**	27.91	-

analogous comparisons have been performed for another achievable reference point, as it is (0.4, 0.97, 50.0), obtaining similar conclusions.

7. Conclusions

In this work, some mechanisms improving the convergence of evolutionary multi-objective optimization algorithms have been analyzed and tested to solve a complex and computationally expensive industrial problem. Some of them are inspired by successful techniques found in literature as the secant method or the MOSASS local search. Another, such as the one based on the construction of an advanced population, has been completely designed by the authors.

More precisely, they have been applied to enhance the performance of the preference-based algorithm known as WASF-GA when a low number of iterations is considered. The different WASF-GA versions have been tested to solve a particular industrial problem based on the optimization of the High-Pressure and Temperature food processing treatment. Then, an exhaustive analysis of their results using some measures of the Pareto front quality has been carried out. From these quality results, it can be concluded that:

- Implementing the creation of an advanced population, the obtained Pareto front approximations show significantly better values than the ones of the original WASF-GA for all the considered proximity and global quality indicators. This advanced population method allows generating new points closer to the non-dominated individuals of the external list, such that they have more probabilities of also being non-dominated.
- Introducing an external list for storing the non-dominated individuals found during the iterative process and using this list for the classification stage, all the points in the final population are achieved to be non-dominated.

As future work, it will be attempted to also improve some quality measures based on the distribution of the points in the Pareto front. Furthermore, the use of surrogate models will be explored. These kinds of models, which are becoming more popular over the last few years, are also prescribed for problems whose evaluation is expensive and involves complex simulations.

Acknowledgments

This research has been funded by grants from the Spanish Ministry of Economy and Competitiveness (TIN2015-66680-C2-1-R and MTM2015-64865P); Junta de Andalucía (P12-TIC301), in part financed by the European Regional Development Fund (ERDF). The work has been performed under the Project HPC-EUROPA3 (INFRAIA-2016-1-730897), with the support of the EC Research Innovation Action under the H2020 Programme; in particular, the author gratefully acknowledges the support of Ben Paechter from the School of Computing at Edinburgh Napier University and the computer resources and technical support provided by EPCC (Edinburgh Parallel Computing Center). This work used EPCC's Cirrus HPC Service (https://www.epcc.ed.ac.uk/cirrus). The authors want to thank the reviewers for their valuable contributions to the final version of this work.

References

Ahmadi, M. H., M. A. Ahmadi, A. H. Mohammadi, M. Feidt, and S. M. Pourkiaei. 2014. "Multi-objective optimization of an irreversible Stirling cryogenic refrigerator cycle." *Energy Conversion and Management* 82: 351–360.

Ahmadi, M. H., M. A. Ahmadi, and S. A. Sadatsakkak. 2015. "Thermodynamic analysis and performance optimization of irreversible Carnot refrigerator by using multi-objective evolutionary algorithms (MOEAs)." *Renewable and Sustainable Energy Reviews* 51: 1055–1070.

Carrasco, M., B. Ivorra, and A. M. Ramos. 2015. "Stochastic topology design optimization for continuous elastic materials." *Computer Methods in Applied Mechanics and Engineering* 289: 131–154.

Crespo, M., B. Ivorra, A. M. Ramos, and A. Rapaport. 2017. "Modeling and optimization of activated sludge bioreactors for wastewater treatment taking into account spatial inhomogeneities." *Journal of Process Control* 54: 118–128.

Denys, S., L. R. Ludikhuyze, A. M. Van Loey, and M. E. Hendrickx. 2000. "Modeling Conductive Heat Transfer and Process Uniformity during Batch High-Pressure Processing of Foods." *Biotechnology Progress* 16 (1): 92–101.

Derrac, J., S. García, D. Molina, and F. Herrera. 2011. "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms." *Swarm and Evolutionary Computation* 1 (1): 3–18.

Durillo, J. J., and A. J. Nebro. 2011. "jMetal: A Java framework for multi-objective optimization." *Advances in Engineering Software* 42: 760–771.

Ferrández, M. R., S. Puertas-Martín, J. L. Redondo, B. Ivorra, A. M. Ramos, and P. M. Ortigosa. 2018. "High-performance computing for the optimization of high-pressure thermal treatments in food industry." *The Journal of Supercomputing* 75: 1187–1202.

Ferrández, M. R., J. L. Redondo, B. Ivorra, A. M. Ramos, and P. M. Ortigosa. 2019. "Preference-based multi-objectivization applied to decision support for High-Pressure Thermal processes in food treatment." 79: 326–340.

Filatovas, E., A. Lančinskas, O. Kurasova, and J. Žilinskas. 2016. "A preference-based multi-objective evolutionary algorithm R-NSGA-II with stochastic local search." *Central European Journal of Operations Research* 1–20.

Gomez, S., B. Ivorra, and A. M. Ramos. 2011a. "Optimization of a pumping ship trajectory to clean oil contamination in the open sea." *Mathematical and Computer Modelling* 54 (1): 477–489.

Gomez, S., B. Ivorra, and A. M. Ramos. 2011b. "Optimization of a pumping ship trajectory to clean oil contamination in the open sea." *Mathematical and Computer Modelling* 54 (1–2): 477 – 489.

Infante, J. A., B. Ivorra, A. M. Ramos, and J. M. Rey. 2009. "On the Modelling and Simulation

of High Pressure Processes and Inactivation of Enzymes in Food Engineering." *Mathematical Models and Methods in Applied Sciences (M3AS)* 19 (12): 2203–2229.

Ishibuchi, H., H. Masuda, Y. Tanigaki, and Y. Nojima. 2015. "Modified distance calculation in generational distance and inverted generational distance." In *International Conference on Evolutionary Multi-Criterion Optimization*, 110–125. Springer.

Ivorra, B., B. Mohammadi, and A. M. Ramos. 2015. "A multi-layer line search method to improve the initialization of optimization algorithms." *European Journal of Operational Research* 247 (3): 711–720.

Ivorra, B., B. Mohammadi, P. Redont, L. Dumas, and O. Durand. 2006. "Semi-deterministic versus genetic algorithms for global optimisation of multichannel optical filters." *International Journal of Computational Science and Engineering* 2 (3-4): 170–178.

Ivorra, B., J. L. Redondo, J. G. Santiago, P. M. Ortigosa, and A. M. Ramos. 2013. "Two-and three-dimensional modeling and optimization applied to the design of a fast hydrodynamic focusing microfluidic mixer for protein folding." *Physics of fluids* 25 (3): 032001.

Lancinskas, A., P. M. Ortigosa, and J. Žilinskas. 2013. "Multi-objective single agent stochastic search in non-dominated sorting genetic algorithm." *Nonlinear Analysis: Modelling and Control* 18 (3): 293–313.

Otero, L., A. D. Molina-García, and P. D. Sanz. 2002. "Some Interrelated Thermophysical Properties of Liquid Water and Ice. I. A User-Friendly Modeling Review for Food High-Pressure Processing." *Critical Reviews in Food Science and Nutrition* 42 (4): 339–352.

Otero, L., A. Ousegui, B. Guignon, A. Le Bail, and P. D. Sanz. 2006. "Evaluation of the thermophysical properties of tylose gel under pressure in the phase change domain." *Food hydrocolloids* 20 (4): 449–460.

Ramos, A. M. 2012. *Introducción al análisis matemático del método de elementos finitos*. Editorial Complutense.

Redondo, J. L., J. Fernández, and P. M. Ortigosa. 2017. "FEMOEA: a Fast and Efficient Multi-Objective Evolutionary Algorithm." *Mathematical Methods of Operations Research* 85 (1): 113–135.

Ruiz, A. B., R. Saborido, and M. Luque. 2015. "A preference-based evolutionary algorithm for multiobjective optimization: the weighting achievement scalarizing function genetic algorithm." *Journal of Global Optimization* 62 (1): 101–129.

Solis, F. J., and R. J. B. Wets. 1981. "Minimization by random search techniques." *Mathematics of operations research* 6 (1): 19–30.

Verbeyst, L., R. Bogaerts, I. Van der Plancken, M. Hendrickx, and A. M. Van Loey. 2013. "Modelling of vitamin C degradation during thermal and high-pressure treatments of red fruit." *Food and Bioprocess Technology* 6 (4): 1015–1023.

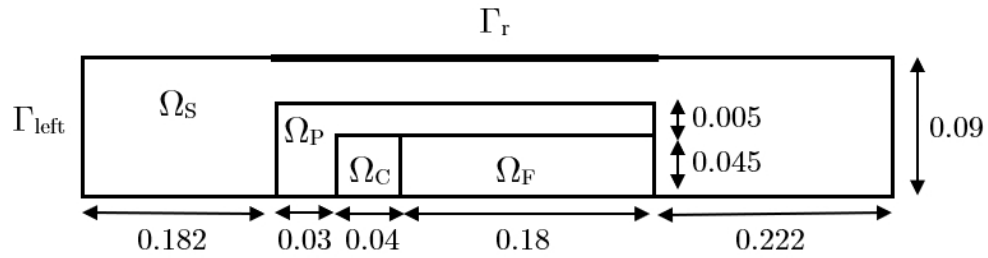
While, L., L. Bradstreet, and L. Barone. 2012. "A fast way of calculating exact hypervolumes." *IEEE Transactions on Evolutionary Computation* 16 (1): 86–95.

Wilcoxon, F. 1945. "Individual comparisons by ranking methods." *Biometrics Bulletin* 1 (6): 80–83.

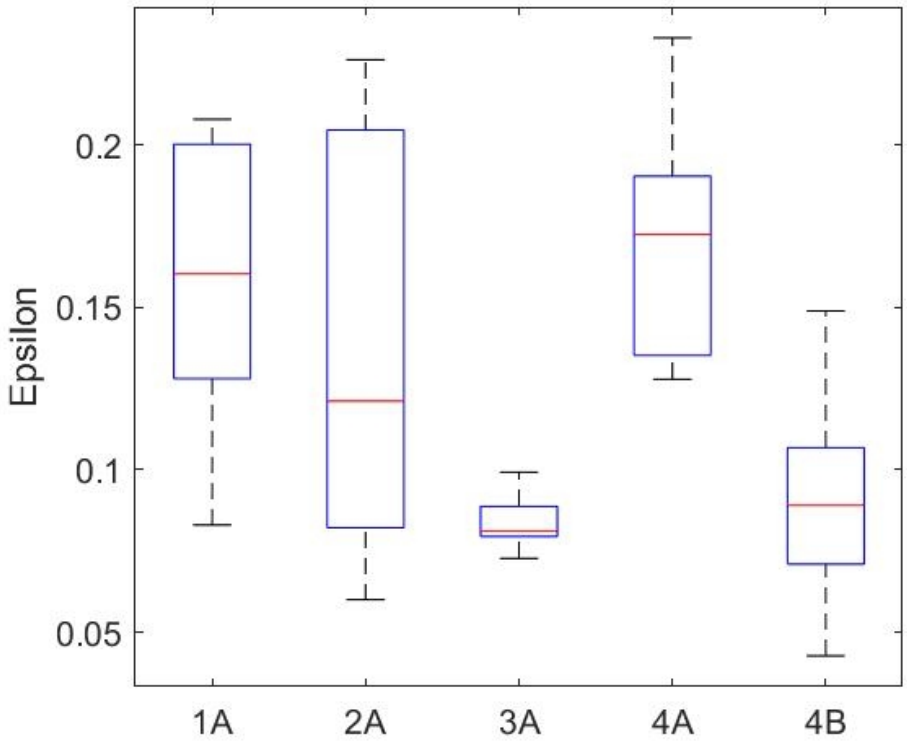
Zhang, Q., and H. Li. 2007. "MOEA/D: A multiobjective evolutionary algorithm based on decomposition." *IEEE Transactions on Evolutionary Computation* 11 (6): 712–731.

Zitzler, E., and L. Thiele. 1998. "Multiobjective optimization using evolutionary algorithms - a comparative case study." In *Parallel problem solving from nature V*, edited by A. E. Eiben, Amsterdam, September, 292–301. Springer-Verlag.

Zitzler, E., L. Thiele, M. Laumanns, C. M. Foneseca, and V.G. da Fonseca. 2003. "Performance assessment of multiobjective optimizers: An analysis and review." *IEEE Transactions on Evolutionary Computation* 7 (2): 117–132.

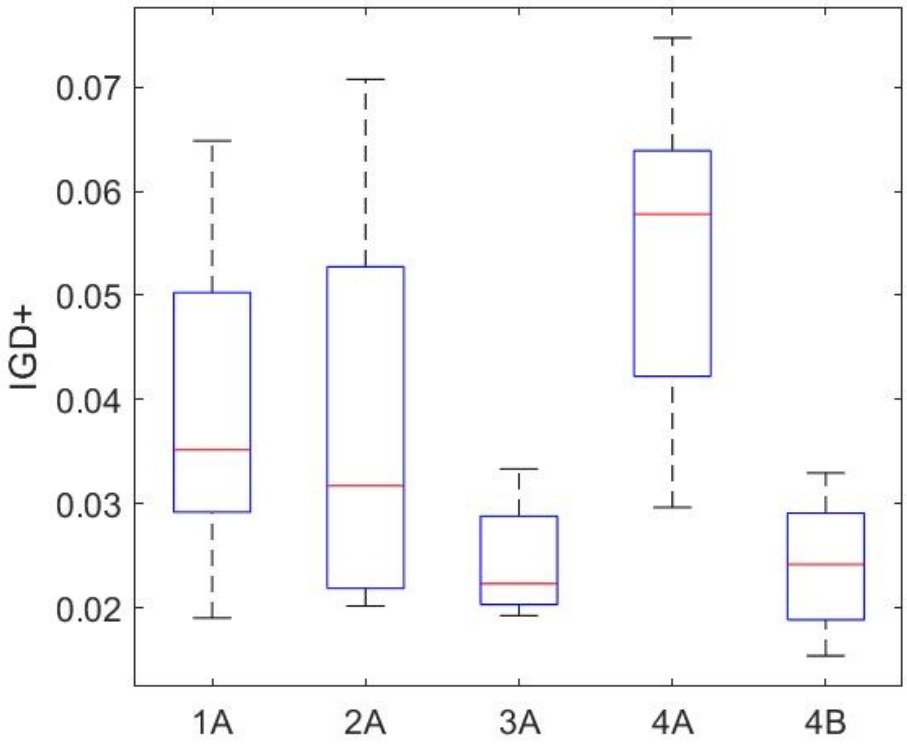


Computational domain $\Omega=\Omega_{\rm F} \cup \Omega_{\rm C} \cup \Omega_{\rm P} \cup \Omega_{\rm S}$.



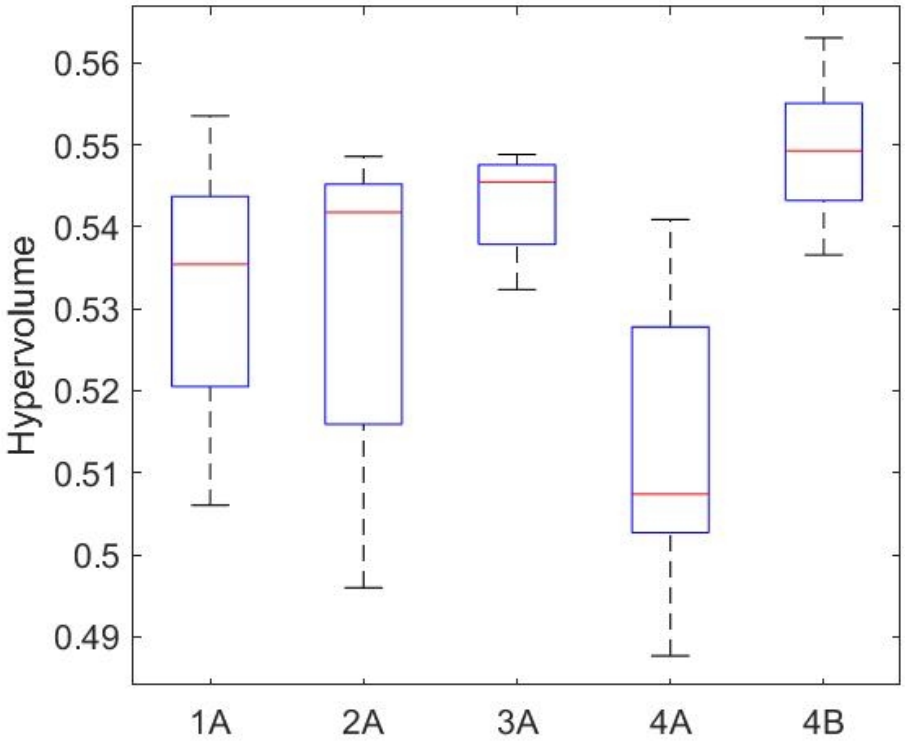
Additive epsilon indicator.
Distribution of the quality measures values for the different variants: $1A$ corresponds to the original WASF-GA, $2A$ to its version based on improving the dominated individuals and building a secant population, $3A$ to the one using the MOSASS mechanism, $4A$ to the one considering the advanced population, and $4B$ the algorithm also using the advanced population but only considering the points of the external list for the classification procedure. For (a) Epsilon and (b) IGD+, the lower the value, the better the quality of the Pareto front approximation. Contrarily, for (c) Hypervolume and (d) NumND, the higher the value, the better the quality. In these boxplot graphics, the lower and upper bounds of the box represent the first and third quartile, respectively; the line inside the box is the median; the lower and upper limits of the whiskers are the minimum and maximum values, respectively, without considering the outliers points represented by the ``+" symbol.

216x180mm (72 x 72 DPI)



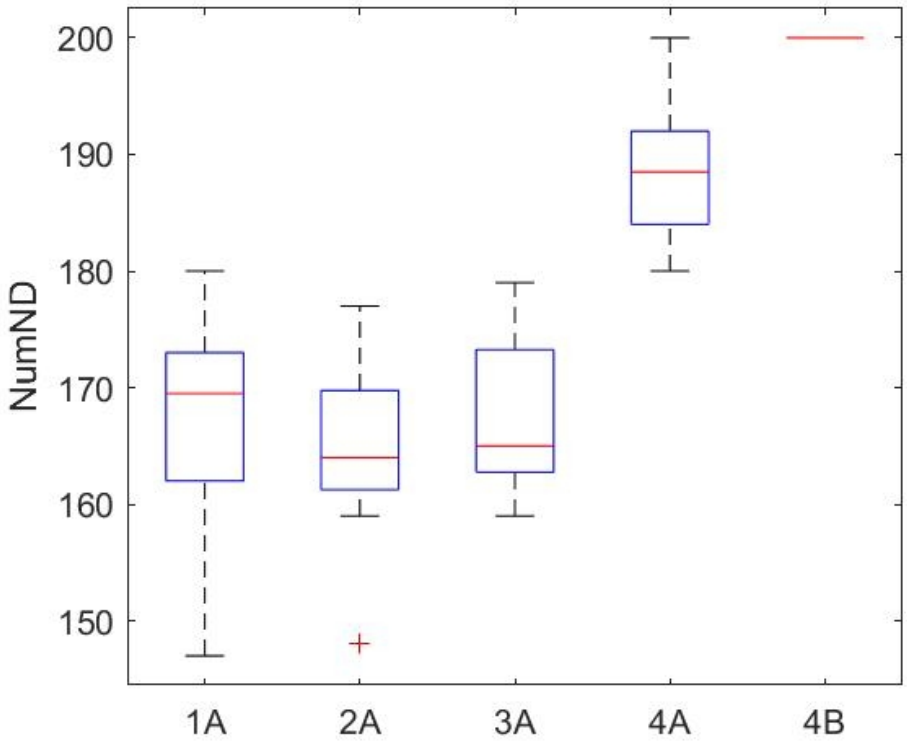
Inverted generational distance plus indicator.
Distribution of the quality measures values for the different variants: $1A$ corresponds to the original WASF-GA, $2A$ to its version based on improving the dominated individuals and building a secant population, $3A$ to the one using the MOSASS mechanism, $4A$ to the one considering the advanced population, and $4B$ the algorithm also using the advanced population but only considering the points of the external list for the classification procedure. For (a) Epsilon and (b) IGD+, the lower the value, the better the quality of the Pareto front approximation. Contrarily, for (c) Hypervolume and (d) NumND, the higher the value, the better the quality. In these boxplot graphics, the lower and upper bounds of the box represent the first and third quartile, respectively; the line inside the box is the median; the lower and upper limits of the whiskers are the minimum and maximum values, respectively, without considering the outliers points represented by the ``+" symbol.

216x180mm (72 x 72 DPI)



Hypervolume indicator.
Distribution of the quality measures values for the different variants: $1A$ corresponds to the original WASF-GA, $2A$ to its version based on improving the dominated individuals and building a secant population, $3A$ to the one using the MOSASS mechanism, $4A$ to the one considering the advanced population, and $4B$ the algorithm also using the advanced population but only considering the points of the external list for the classification procedure. For (a) Epsilon and (b) IGD+, the lower the value, the better the quality of the Pareto front approximation. Contrarily, for (c) Hypervolume and (d) NumND, the higher the value, the better the quality. In these boxplot graphics, the lower and upper bounds of the box represent the first and third quartile, respectively; the line inside the box is the median; the lower and upper limits of the whiskers are the minimum and maximum values, respectively, without considering the outliers points represented by the ``+" symbol.

216x180mm (72 x 72 DPI)



Number of non-dominated points in the Pareto front approximation.
Distribution of the quality measures values for the different variants: $1A$ corresponds to the original WASF-GA, $2A$ to its version based on improving the dominated individuals and building a secant population, $3A$ to the one using the MOSASS mechanism, $4A$ to the one considering the advanced population, and $4B$ the algorithm also using the advanced population but only considering the points of the external list for the classification procedure. For (a) Epsilon and (b) IGD+, the lower the value, the better the quality of the Pareto front approximation. Contrarily, for (c) Hypervolume and (d) NumND, the higher the value, the better the quality. In these boxplot graphics, the lower and upper bounds of the box represent the first and third quartile, respectively; the line inside the box is the median; the lower and upper limits of the whiskers are the minimum and maximum values, respectively, without considering the outliers points represented by the ``+" symbol.

216x180mm (72 x 72 DPI)