Int J Interact Des Manuf (2009) 3:237–245
DOI 10.1007/s12008-009-0080-x

ORIGINAL PAPER

Multi-objective preliminary ecodesign

Amadou Ndiaye · Patrick Castéra · Christophe Fernandez · Franck Michaud

Received: 10 June 2009 / Accepted: 25 August 2009 / Published online: 16 September 2009

Abstract The design by multi-objectives optimization implies the optimization of several contradictory objectives simultaneously. In fact there is no optimal solution for one particular objective if the other objectives are considered, but the aim is to simultaneously minimize all the objectives in order to reach an optimal compromise. Optimum is reached if any improvement of one objective induces the degradation of one other. Such an optimum is located on a front called Pareto front. The Pareto front, a set of optimal solutions that are not equivalent, allows us to choose an optimal solution with criteria external to optimization process (economic or functional). In this study, a multi-objective particle swarm optimization (a metaheuristic) algorithm has been used to optimize a wood plastic composite for decking application. This metaheuristic, based on evolutionary techniques, applies to a great diversity of functions objectives: continuous or discrete equations, qualitative knowledge rules and algorithms. The design variables are mainly variables of raw materials production, and the incorporation of a biopolymer, the control of timber particle sizes and chemical or thermal timber changes. The objective functions are equations and an algorithm integrating discrete data in the modelling of creep behavior, water resistance and fossil resources depletion.

A. Ndiaye (✉) · P. Castéra · C. Fernandez
UMR927 Sciences du Bois et des Biopolymères,
INRA, CNRS, Université Bordeaux 1, 33405 Talence, France
e-mail: ndiaye@us2b.pierroton.inra.fr; ndiaye@bordeaux.inra.fr

P. Castéra
e-mail: castera@us2b.pierroton.inra.fr

C. Fernandez
e-mail: fernandez@us2b.pierroton.inra.fr

F. Michaud
Ecole Supérieure du Bois, BP 10605, Atlanpole,
Rue Christian Pauc, 44306 Nantes Cedex 3, France
e-mail: Franck.michaud@ecoledubois.fr

Keywords Particle swarm optimization · Multi-objective problem · Composite material · Preliminary ecodesign

1 Introduction

1.1 The preliminary ecodesign problem

Taking into account environmental impact criteria in the preliminary ecodesign of semi-products or of completed functional units is becoming more and more an issue for industry. It implies going through a life cycle analysis (LCA) which is now the international standard to evaluate such impacts. It is in fact the only way to compare the environmental impact of different products that fulfill the same function; and this, from the production of raw materials to the final destination (Fig. 1). The fact that it is necessary to know the life cycle of a product makes it difficult to use the LCA during the preliminary ecodesign. One way to tackle the problem would be to focus on one of the stages of the life cycle of the product and to consider it as independent from the other stages.

The design process will be different if we are trying to: 1) improve the environmental characteristics of a product while disturbing as little as possible its production process, 2) optimize the environmental impact of a product defined by end-use performances without restricting oneself to a particular process. The first case, frequent with manufacturers, being guided by the manufacturing process, can make it impossible to meet both the technical and the environmental requirements in a given manufacturing scheme. The second approach, which is more prospective and open, is guided by the end-use properties that are required, and therefore can be tackled either in seeking and environmental optimum in a

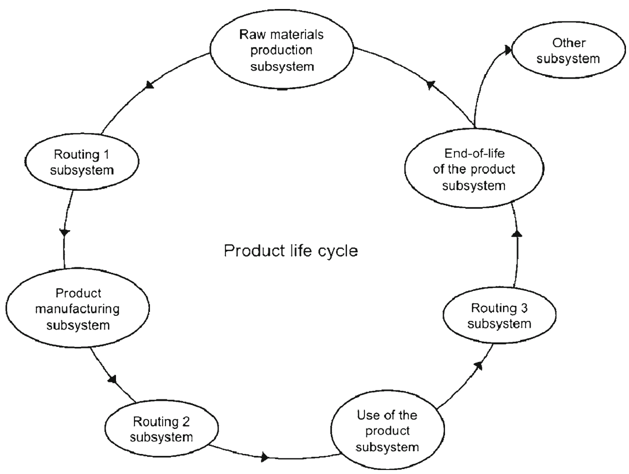

Fig. 1 Representation of a product life cycle as being a system made up of a sequence of subsystems

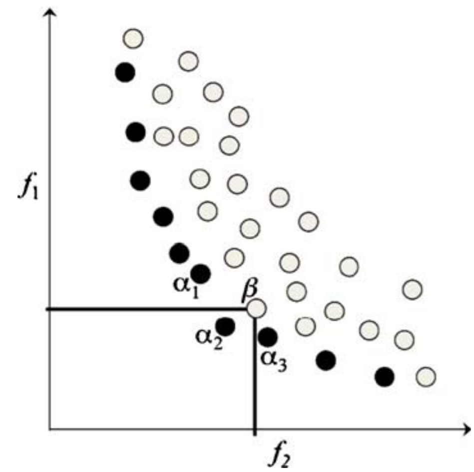

Fig. 2 Dominance relation in a bi-objective space: β is dominated by α_2 and α_1, α_2 and α_3 are on the Pareto front

search space that is constrained by functional specifications or through a multi-objective optimization.

The second approach is closer to conventional preliminary design. However, as multi-objective optimization does not provide a single solution, but a set of possible solutions satisfying the design criteria among which the designer will be able to choose according to additional constraints, both approaches will be considered in this paper as preliminary ecodesign.

The example which is studied here concerns the preliminary design of an outdoor decking taking into account its environmental profile (first approach). The initial choice was of a wood–plastic composite, this choice allowing the use of industrial byproducts in a constrained search space. The optimum of the required properties will be obtained by multi-objective optimization.

1.2 A multi-objective optimization problem

Design by multi-objective optimization implies simultaneous optimization of various contradictory objectives. If we take a simple example consisting in minimizing simultaneously the two following functions: $f_1(x) = x_1$ and $f_2(x) = x_2/ax_1,$ the improvement of the first objective ($f_1(x)$) comes with a degradation of the second objective $f_2(x)$). This contradiction expresses the fact that there does not exist an optimal solution regarding the two objectives, there are only optimal compromises. With this example we see that for a minimal f_1 and thus x_1 the lowest possible, we need the lowest possible x_2 to minimize f_2. In addition, the absolute minimum f_2 is obtained with x_1 the highest possible and x_2 the lowest possible. It is the taking into account of this contradiction between minimization of f_1 and minimization of f_2 that introduces the notion of compromise whether one favors f_1 or f_2. We see that from a purely algebraic point of view x_1 cannot be null (division by zero). This observation introduces the fact that there is often a certain amount of constraints that must be met by the objective functions and/or their variables. These are also called parameters, optimization variables or conception variables. The constraints that are specifications of the problem limit the search spaces of the parameters and/or the determining, for example, bottom or top values. A general multi-objective optimization problem includes a set of k objective functions of n decision variables (parameters) constrained by a set of m constraint functions. It can be defined as below:

$$\begin{aligned}&\text{Optimize } \vec{f}(\vec{x}) = \left[f_1(\vec{x}), f_2(\vec{x}), \ldots, f_k(\vec{x})\right]\\&\text{subject to } g_j(\vec{x}) \leq 0 \quad \text{for} \quad j = 1, \ldots, p\\&\qquad \text{and } h_j(\vec{x}) = 0 \quad \text{for} \quad j = p+1, \ldots, m\end{aligned}$$

where $\vec{x} = (x_1, x_2, \ldots, x_n) \in \Re^n$ is the vector of decision variables, $f_i : \Re^n \rightarrow \Re$ for $i = 1, \ldots, k$ are the objective functions and $g_i, h_j : \Re^n \rightarrow \Re$ for $i = 1, \ldots, m$ and $j = 1, \ldots, p$ are the constraint functions of the problem.

A compromise will be said optimal if every improvement of an objective induces degradation of another objective. A compromise whose objectives can be improved is not optimal. It is said to be dominated by at least another compromise, which is the one obtained after improvement of its objective functions. The optimal compromises are located on a front named Pareto front (Fig. 2). Let's take as an example Fig. 2, where the plain dots constitute the Pareto front, the objective functions f_1 and f_2 at point β can still be improved to reach point α_2; therefore point β is dominated by at least point α_2. In addition, if, starting from point α_2 we minimize again function f_2, we will reach point α_3 which has an f_2 superior to the f_2 at point α_2; the same phenomenon would occur in

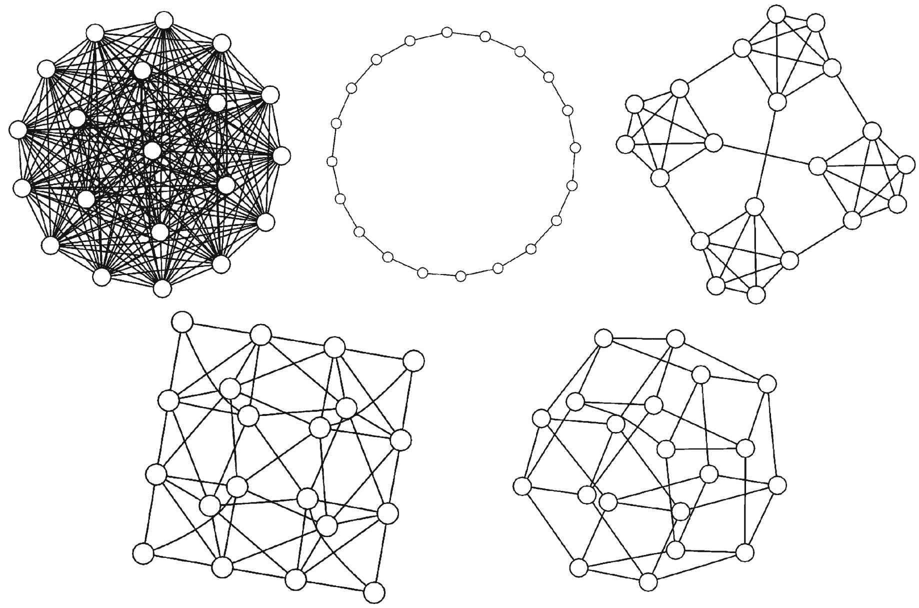

Fig. 3 Illustration of neighborhood topologies from [9]: *fully connected* (all), *ring*, *four clusters*, *pyramid* and *square*

function with f_1, α_2 and α_1. The Pareto Dominance can be defined as below:

$\vec{x} = (x_1, \ldots, x_n)$ is said to dominate $\vec{x}' = (x'_1, \ldots, x'_n)$ (denoted $\vec{x} \prec \vec{x}'$) if and only if

$\forall i \in \{1, \ldots, n\}, x_i \leq x'_i$ and $\exists j \in \{1, \ldots, n\}, x_j < x'_j$

The presence of a Pareto front, thus a set of optimal non-equivalent solutions, allows the choice of an optimal solution with regard to economical of functional criteria, which are external to the solved problem of multi-objective optimization. We will illustrate the procedure for the optimization of a wood–plastic composite decking with three objectives [3]. In this example, the optimization focuses on the creep, swelling, and exhaustion of abiotic resources functions. The design variables are mainly characteristics of raw materials such as timber particle sizes and chemical or thermal timber changes.

2 Background

2.1 Particle swarm optimization (PSO)

Particle swarm optimization (PSO) is an evolutionary computation technique developed by Kennedy and Eberhart [6]. This technique, motivated by the simulation of social behavior, has proved to be very efficient in hard optimization problems. The swarm is composed of particles randomly chosen in the search space. The search space, represented by the objective function and its constraints, is of n dimensions (a hyperdimension). Each particle knows its position in the search space, its best position ever visited, the best position of its neighborhood and has an instantaneous velocity. A particle position is described by its n coordinates in the search space and the corresponding fitness of the objective function. A particle position is best than an other one if its objective function fitness is best than that of the other one; best meaning less than if it is a minimization problem and greater than if it is a maximization problem. The neighborhood of a given particle (its social network) is composed by all the particles that influence its trajectory in the search space. The two most commonly used neighborhood topologies are the *fully connected* topology named *gbest* topology and the *ring* topology named *lbest* topology [8]. In the *fully connected* topology all particle is connected to every other; the trajectory of each particle is influenced by the best position found by any particle of the swarm as well as their own past experience. In the *ring* topology every particle is connected to its k immediate neighbors with toroidal wrapping; this allows parallel search and then subpopulations could converge in diverse regions of the search space. Usually the *ring* topology neighborhood comprises exactly two neighbors, every particle is connected to its two immediate neighbors one on each side. With a *fully connected* topology the swarm converges quickly on the problem solution but is vulnerable to the attraction of local optima, while, with *ring* topology, it better explore the search space and is less vulnerable to the attraction of local optima. Various neighborhood topologies have been investigated in [5,8,9] (Fig. 3). The structures experimented where from classical communications structures [2] and small-worlds networks [13]. The main conclusion was that the difference in performance depends on the topology implemented for a given function, with nothing suggesting that any topology was generally better than any other [11].

The movements of the particles are synchronized: at each time step all particles move at the same time, each particle choosing a direction and a velocity, the calculation of which includes its present position, the best position ever visited, and the position of its neighbors. A particle is equipped with means of communication, a small memory and a capacity to make decisions. Its communication means enable it to inform its neighbors of its own position, and to receive the position of each one of them. Its memory enables it to memorise its own position, the best position ever visited, and the best position of its neighborhood. Its capacity to make decisions enables it to decide which direction to go and at what velocity, and this at each time step. The standard PSO algorithm has been defined [6] for a search in a n-dimensional search space where the particles movements are synchronized: at the tth iteration, for the ith particle, the position and position change (velocity) vectors were respectively represented as:

$$X_i^t = (x_{i,1}^t, x_{i,2}^t, \ldots, x_{i,n}^t) \quad (1)$$
$$V_i^t = (v_{i,1}^t, v_{i,2}^t, \ldots, v_{i,n}^t) \quad (2)$$

The position $x_{i,j}^{t+1}$ and position change (velocity) $v_{i,j}^{t+1}$ updating rules are given as below:

$$x_{i,j}^{t+1} = x_{i,j}^t + v_{i,j}^{t+1} \quad (3)$$
$$v_{i,j}^{t+1} = w \cdot v_{i,j}^t + c_1 r_1 (p_{i,j}^t - x_{i,j}^t) + c_2 r_2 (g_j^t - x_{i,j}^t) \quad (4)$$

where $i = 1, 2, \ldots p$, $j = 1, 2, \ldots n$, p is the number of particles (the size of the swarm), and n is the dimension of search space; $x_{i,j}^{t+1}$ is the position of the particle i and $v_{i,j}^{t+1}$ its velocity; w is called inertia weight, it is used to control the impact of the previous history of velocity on the current one; r_1 and r_2 are uniformly distributed random numbers between 0 and 1; c_1 and c_2 are positive acceleration constants; $p_{i,j}$ is the value of jth dimension of the best position ever visited by the ith particle; g_j is the value of jth dimension of the global best position ever visited by all particles in the swarm.

2.2 Discrete binary particle swarm optimization (DPSO)

Kennedy and Eberhart [7] have introduced a discrete binary version of PSO (DPSO) that operates on binary variables (bit, symbol or string) rather than real numbers. The difference between the PSO and DPSO definitions is in the velocity updating rules $v_{i,j}^{t+1}$ where the position updating rule $x_{i,j}^{t+1}$ is based on a logistic function as below:

$$\begin{aligned} x_{i,j}^{t+1} &= 1 \quad \text{if } \varphi < S\left(v_{i,j}^{t+1}\right) \\ &= 0 \quad \text{otherwise} \end{aligned} \quad (5)$$

where

$$S\left(v_{i,j}^{t+1}\right) = \frac{1}{1 + e^{-v_{i,j}^{t+1}}} \quad (6)$$

and φ is an uniformly distributed random number between 0 and 1.

With the introduction of DPSO, Kennedy and Eberhart [7] extend the use of PSO to optimization to discrete binary functions as well as to functions of continuous and discrete binary variables at the same time. To be able to handle the optimization of functions including discrete n-ary variables, Michaud et al. [10] have generalized the discrete binary version of PSO to a discrete n-ary version of PSO as below:

$$\begin{aligned} x_{i,j}^{t+1} &= n_k \quad \text{if } \varphi_{k-1} < S\left(v_{i,j}^{t+1}\right) \\ &= n_l \quad \text{if } \varphi_{l-1} < S\left(v_{i,j}^{t+1}\right) \le \varphi_l \text{ with } 1 < l \le k-1 \\ &= n_1 \quad \text{if } \varphi_1 \ge S\left(v_{i,j}^{t+1}\right) \end{aligned} \quad (7)$$

where $\varphi_1, \ldots, \varphi_{k-1}$ are strictly ordered uniformly distributed random numbers between 0 and 1.

2.3 The PSO algorithm

The original procedure for implementing PSO is a simple and easy to implement six steps algorithm:

1. Initialize a population of particles with random positions and velocities on n dimensions in the problem space.
2. For each particle calculate its fitness (the function to optimize in n variables).
3. Compare particle's fitness with the fitness of its best position ever visited (*pbest*). If current value is better than *pbest*, then it becomes *pbest*
4. Identify the particle in the neighborhood with the best fitness, it becomes the leader of the neighbourhood.
5. Change the velocity and position of the particle according to its velocity and position updating rules (Eqs. 3, 4 and/or 7).
6. Loop to step 2. Until the end condition is met, usually a sufficiently good fitness or a maximum number of iterations.

2.4 Multi-objective particle swarm optimization (MOPSO)

With the PSO algorithm, the determining of the leader that influences the updating of the position of a particle is function of the established neighborhood topology. However, in a multi-objective optimization problem the determining of the leader is function of the set of leaders already founded in the search space. Such set of leaders is usually stored in a specific memory [1,4,12] called *extended memory* or *external archive*. When a particle dominates some leaders in the extended memory, it is added to the leaders set and the dominated ones are discarded from the extended memory. The set of leaders is reported as the final Pareto optimal set or Pareto front. The particles in the Pareto front are equivalent in the absence of any preference among the objectives.

3 The wood–plastic composite preliminary ecodesign problem

The wood–plastic composites (WPC) constitute a good example of an ecodesign approach: initially developed in North America for recycling materials—plastics and papers—they also enable a significant reduction of the plastic coming from the petrochemical industry. There is thus in their development both a definite economic advantage and a potential environmental interest. Nevertheless when decking is used outdoor, these products exhibit a certain amount of weakness points and contradictions: in order to allow a homogeneous extrusion and to prevent the material from becoming too fragile, a minimal quantity of thermoplastic (about 30 % in the case of a PEHD/wood composite) is necessary. In addition, in order to improve compatibility between the two components—one being polar, the other being apolar—chemical additives are included in the formula.

The WPC preliminary ecodesign requires first that the designer solves a multi-objective optimization problem. Usually one of the three strategies below is used: 1) optimizing one objective with constraints on the others, 2) optimizing a weighted function including the different objectives or 3) Pareto optimization. The first two strategies lead to a single solution while the third one leads to a set of optimal compromises between the objectives that is well distributed in the space of solutions. The evolutionary techniques—genetic algorithm (GA), ant colony (AC), particle swarm optimization (PSO), etc… are well adapted to the third strategy with more or less efficiency. The PSO technique, like other evolutionary techniques, finds optima in complex optimization problems. Like GA, the system is initialized with a population and searches for optima by updating generations. However, unlike GA, PSO has no evolution operators such as crossover and mutation. PSO while traversing the search space is focused on the optimum, whereas GA explores the search space and then takes more time to find the optimum. In the WPC preliminary ecodesign the main objective is to find the relevant optima to be able to choice an optimum with regard to economical of functional criteria; knowing that completely different composite formulations lead to equivalent composites in reference to the objective functions. Multi-objective PSO technique is specially and fully suitable for this problem.

3.1 The wood–plastic composite preliminary ecodesign modelling

The modelling of WPC for decking application preliminary ecodesign has taken a multidisciplinary team (physicists and computer scientists). The modelling process has been to generate knowledge by some experiments, collect knowledge generated and those from the literature and build up the influence graphs of relationships between the problem variables [10]. The three objectives considered in the preliminary ecodesign of wood–plastic composite (creep, swelling and exhaustion of fossil resources functions) have been identified as critical weak points of the product [10]. From an environmental point of view, exhaustion of fossil resources is, with the green house effect, the weak point of this material. We will recall their definition in order to highlight the algorithmic nature of these functions.

3.1.1 The creep function (def)

The creep function, $def(t_{ref})$, is an empirical non linear power function that has been fitted to bending experimental results. The magnitude of creep deformation is related to the elastic compliance $1/E$. The kinetics of creep deformation is related to the viscosity of the composite, ν. The fiber size distribution parameter k_{GRAN} used in Eq. 9 is a discrete variable that can take three different values between 0.3 (random) and 1 (unidirectional) with an intermediate value calculated at 0.69 (partially oriented)—see Michaud et al., *op. cit.*, whereas the other variables used in the Eqs. (8), (9) and (10) are continuous. In fact the *def* function (Eq. 8), in its developed formula has an algorithm form due to the conditions on the discrete k_{GRAN}.

$$def(t_{ref}) = \frac{A\left(\frac{\sigma_0}{\sigma_{MOR}}\right) \cdot t_{ref}^{\frac{N \cdot e^{\left(\frac{\sigma_0}{\sigma_{MOR}}\right)}}{\nu}}}{E} \tag{8}$$

where A and N are fitted parameters of the creep function model, σ_0 is applied stress, σ_{MOR} is modulus of rupture of the composite material, t_{ref} is the time to reach a limit state deflection, E is the modulus of elasticity and ν is the apparent viscosity of the composite at room temperature. E and ν are calculated through a simple mixture law, as shown in Eqs. (9) and (10). These equations reveal the main optimization variables, i.e., material properties, volume fractions and fibre orientation.

$$E = \lambda_m(\alpha_{bio}E_{bio} + (1-\alpha_{bio}-\alpha_{add})E_m) + \lambda_{add}E_{add} + (k_{GRAN}) \cdot (1-\lambda_m-\lambda_{add})E_f \tag{9}$$

$$\nu = \lambda_m(\alpha_{bio}\nu_{bio} + (1-\alpha_{bio})\nu_m) + \lambda_{add}\nu_{add} + (1-\lambda_m-\lambda_{add})\nu_f \tag{10}$$

See Table 1 for the meaning of other variables.

3.1.2 Water swelling function (SW)

The swelling function due to water absorption, *SW*, is defined by Eq. (11). It expresses the fact that the swelling of the composite is the sum of the swelling deformations of all hygroscopic components present in the composite and accessible to water, e.g., wood, biopolymers… The part representing the

Table 1 Variables $X = \{x_1, x_2, \ldots, x_{12}\}$ related to the composite formulation

x_j	Description	Main relations
$x_1 = \lambda_f$	Fiber ratio in composite formulation	$0 \le x_1 \le 1$ and $x_1 = x_1(x_4 + x_5 + x_6)$
$x_2 = \lambda_{add}$	Additives ratio in composite formulation	$0 \le x_2 \le 1$
$x_3 = \lambda_m$	Matrix ratio in composite formulation	$0 \le x_3 \le 1, x_3 = 1 - x_1 - x_2$ and $x_3 = x_3(x_7 + x_8 + x_9)$
$x_4 = \alpha_f$	Fiber ratio in fiber component	$0 \le x_4 \le 1$ and $x_4 + x_5 + x_6 = 1$
$x_5 = \alpha_{frec}$	Recycled fiber ratio in fiber component	$0 \le x_5 \le 1$
$x_6 = \alpha_{reinf}$	Other reinforcement ratio in fiber component	$0 \le x_6 \le 1$
$x_7 = \alpha_m$	Thermoplastic ratio in matrix component	$0 \le x_7 \le 1$and $x_7 + x_8 + x_9 = 1$
$x_8 = \alpha_{bio}$	Biopolymer ratio in matrix component	$0 \le x_8 \le 1$
$x_9 = \alpha_{trec}$	Recycled thermoplastic ratio in matrix	$0 \le x_9 \le 1$
$x_{10} = GRAN$	Fiber size distribution factor	Discrete variable $x_{10} = \{1, 2, 3\}$
$x_{11} = k_t$	Fiber treatment factor	Discrete variable $x_{11} = \{0, 1, 2, 3\}$
x_{12}	Viscoelastic properties of constituents	E, n

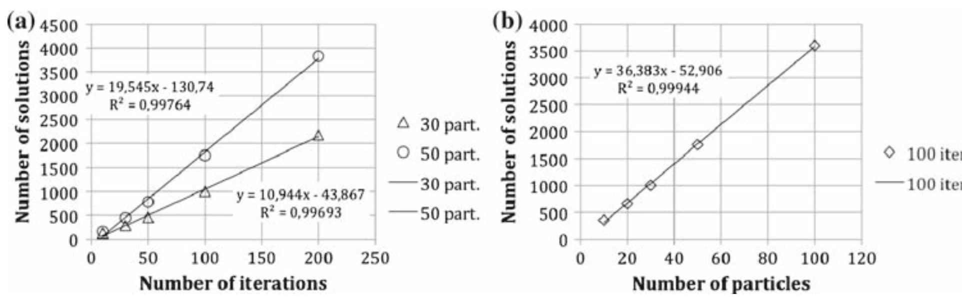

Fig. 4 Stability of the Pareto front: **a** constant number of particles, **b** constant number of iterations

swelling of the fibres vanishes when the fibres are not accessible to water (below a given percolation threshold λ_0). In addition the swelling capacity of wood fibres can be changed by thermal or chemical wood modification, which is expressed in Eq. (11) by the discrete variable k_t that can take three different values (low, medium or high effect). The *SW* function is also an algorithm: there are conditions on the discrete variables (k_t, m and ω) and on the threshold variable λ_0.

$$\begin{aligned} SW &= (1 - \alpha_{frec}(1 - k_{fr}))k_t(1 - e^{-m.\lambda_f^{\omega+1}})\lambda_f SW_f \\ &\quad + \alpha_{bio}\lambda_m SW_m \quad \text{if } \lambda_f + \alpha_{bio}\lambda_m \ge \lambda_0 \\ &= \alpha_{bio}\lambda_m SW_m \text{ otherwise} \end{aligned} \tag{11}$$

where λ_0 is the percolation threshold; k_{fr} is the user defined coefficient for influence of recycled fiber onto swelling; k_t is the user defined coefficient for influence of treatment onto swelling; m, ω, SW_f and SW_m are swelling function parameters. See Table 1 for the meaning of other variables.

3.1.3 Exhaustion of fossil resources function (efr)

The exhaustion of fossil resources function, *efr*, is defined as an addition of two factors (Eq. 12): one for fibres used and one for the non renewable part of the polymer if the polymer is a blend.

$$efr = a_1\lambda_f + a_2\,(1 - \alpha_{bio})\left(1 - \lambda_f\right) \tag{12}$$

where the coefficient a_1 represents the impact of fiber processing and treatment on the exhaustion of fossil resources, and the coefficient a_2 reflects the impact of non renewable thermoplastic and additives production and processing. Other factors have an impact on *efr*, such as consumption of non renewable energy during composite assembly, production of additives… For simplification they have not been considered. Normally a_2 is expected to be higher than a_1. The balance between the two coefficients influences the environmental optimization. See Table 1 for the meaning of other variables.

3.2 Application of the MOPSO algorithm

In the design of wood–plastic composite (WPC), the creep and swelling functions are conflicting: the swelling of the composite growth when the creep decreases with the rate of fibers (wood). The MOPSO deals with such conflicting objectives; even if the representation of each objective is an

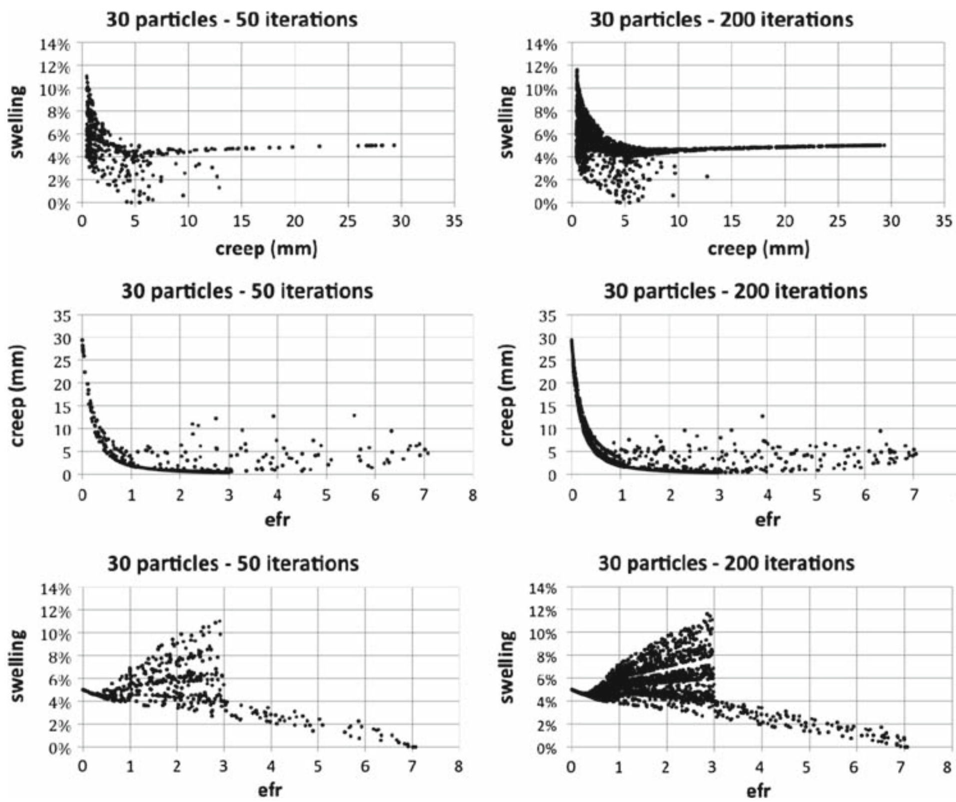

Fig. 5 The distribution of particles in the Pareto front is statistically stable. The 3D Pareto front is represented here by two objective functions

Table 2 Examples of solutions in the Pareto front

Solution	λ_m (%)	α_{bio} (%)	*GRAN*	α_{frec} (%)	k_t	efr	Creep (mm)	Swelling (%)
a	33	0	2	9	2	4.67	1.0	3
b	59	39	2	2	2	4.10	1.9	3

algorithm and thus with a high number of functions. In our WPC preliminary design we have three objective functions with two of them represented each by an algorithm utilizing several variables.

3.2.1 Dealing with continuous and discrete variables

Equations (3) and (7) are used as *position* updating rule of respectively, real and discrete variables. The Equation (4) is used as *velocity* updating rule for all variables. During the optimization process, the real variables converge to their optima according to the objective functions, whereas each discrete variable randomly traverses its space of definition and consequently its best solution is identified.

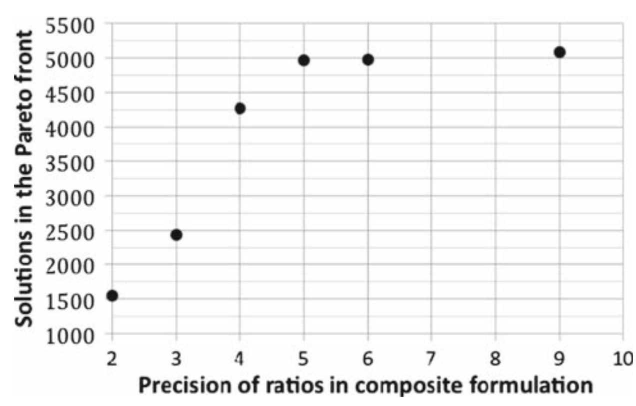

Fig. 6 The influence of the precision of ratios in composite formulation on the number of solutions in the Pareto front

3.2.2 Multi-objective optimization

In this work we have applied the MOPSO method described in [1]. In this method only the *fully connected* topology is used to calculate the position of each particle for each objective function and then the Pareto dominance test is applied to each particle regarding the particle's positions stored in the *extended memory*. If the position of a particle dominates some particle's positions in the *extended memory*, the position of the particle is stored in the *extend memory* and the ones dominated are discarded from the *extended memory*. We used, as end condition of the optimization process, a given maximum number of iterations. Of course the swarm is randomly initialized and the number of its particles is given. The Pareto front is constituted by the particle's positions in the *extended memory* at the end of the optimization process.

The efficiency of the optimization is hardly influenced by the constant parameters w, c_1 and c_2 in Equation (4). Such parameters have to be experimentally adapted to each optimization problem. For our problem the parameters w, c_1 and c_2 have been respectively settled to 0.63, 1.45 and 1.45.

4 Results and discussion

4.1 Stability of the Pareto front

The Pareto front is stable regarding the swarm size and the number of generations of particles (number of iterations used as end-condition of the optimization process):

1. for a given swarm size, the number of particles in the Pareto front increases with the increasing number of generation of particles according to an affine law (Fig. 4a), but the shape of the front remains the same; and for a given number of generation of particles, the number of particles in the Pareto front increases with the increase of the swarm size (Fig. 4b).
2. for a given swarm size and two different numbers of generation (200 and 50), the distribution of particles in the Pareto front is statistically stable (Fig. 5), as the coordinates of the solutions in the front may differ.

The size of the Pareto front can be rather large and therefore the swarm size and the number of iterations should be fitted in order to obtain a reasonable front size.

4.2 Analysis of MOPSO solutions on composite formulations

Table 2 illustrate the differences that can be exist between two solutions, a and b, very close in the Pareto front. The two solutions refer to the two completely different composite

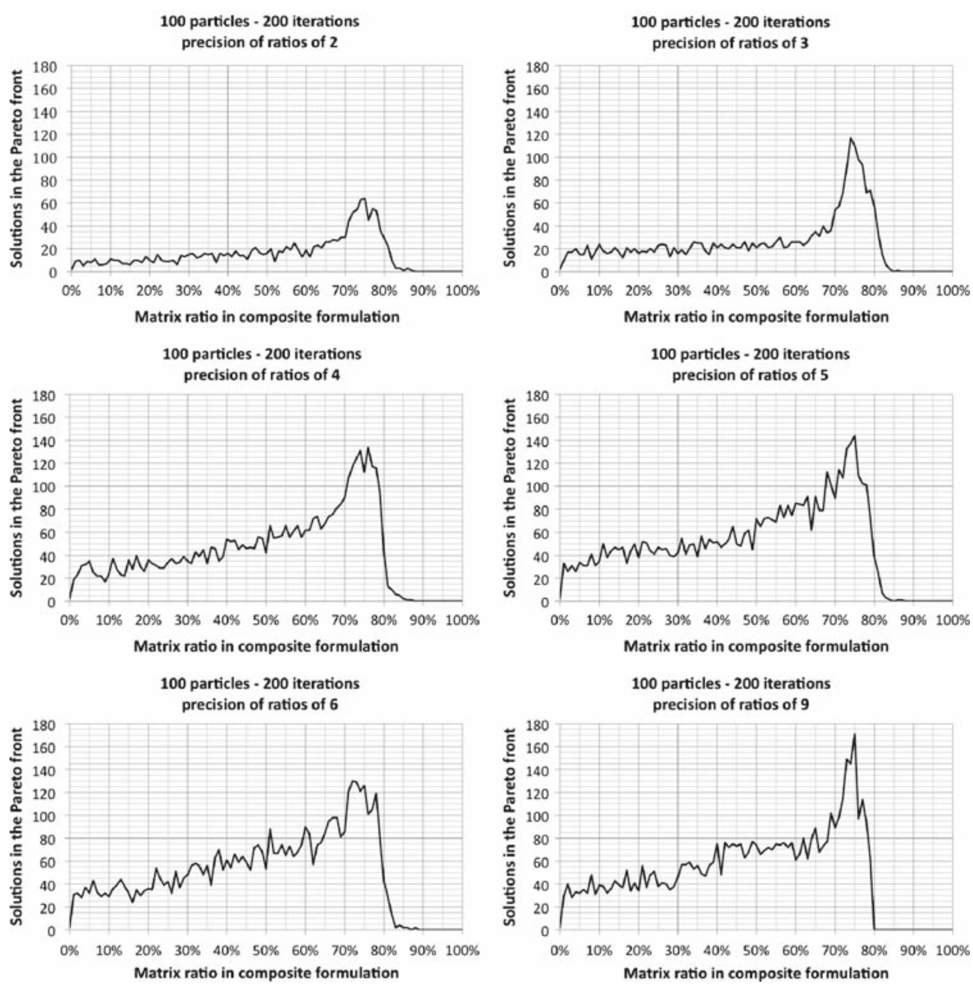

Fig. 7 The influence of the matrix ratio in composite formulation on the number of solutions in the Pareto front

formulations: the solution (*a*) contains a low rate of plastic (36%) without biopolymer, randomly oriented short fibers with 9% of recycled ones and a high treatment level; the solution (*b*) contains a high plastic content (59%) with 39% of biopolymer thermoplastic, randomly oriented short fibers with 2% of recycled ones and a high treatment level. These two solutions are rather equivalent regarding the objective functions values: for (a) 1 mm/3%/4.67 for creep/swelling/efr and 1.9 mm/3%/4.10 for (b). These results show a significant gap for raw materials content and underline the power of such optimization process offering new possibilities of preliminary design.

4.3 A large number of solutions

The number of MOPSO solutions on composite formulations depends on the ratios between the components and their desired precision. The number of solutions grows in function with the precision of ratios between the components using a logarithm-like law. It starts at 1,500 solutions for a precision of 2 (the lowest possible precision) to more than 5,000 for a precision greater than 5 (Fig. 6). The matrix ratio in composite formulation generates a peak of solutions around 75% for any precision of ratio (Fig. 7). This large number of solutions makes them difficult to handle. One solution is to take into account, in the system process, the user of the system so he could fix the precision of ratios, and for each ratio, its desired range; the latter being included in the domain of validity of the variable representative of the ratio. For example if you want to formulate a wood–plastic composite with a matrix ratio lying between 30 and 40% without biopolymer, it is sufficient to restrict the range of the variable representative of the matrix ratio (λ_m) between 0.3 and 0.4 and the one representative of the biopolymer ratio in matrix component (α_{bio}) between 0.0 and 0.0. In this case the number of solutions in the Pareto front fall down to 20.

5 Conclusion

In this paper we have shown the easiness of handling the multi-objective particle swarm optimization (MOPSO) method and its interest in preliminary ecodesign. The method provides a set of "interesting" solutions among which the designer will be able to refine the design process, introducing for instance processes, availability of raw materials and economic viability. There is no restriction on the number of objectives, provided their expressions and interactions between them can be clearly defined. We have used a MOPSO algorithm based on an extended memory technique to calculate a stable Pareto front for three objective functions: creep, swelling and exhaustion of fossil resources in the context of the environmental optimization of the wood–plastic composite. The creep and swelling functions are in fact algorithms using in the same time continuous and discrete variables. A flexible and multiplatform (unix, windows and mac osx) computer program has been developed.

Acknowledgments We acknowledge ANR (National Research Agency) for funding the research.

References

1. Alvarez-Benitez, J.E., Everson, R.M., Fieldsend J.E.: A MOPSO algorithm based exclusively on pareto dominance concepts. Evolutionary multi-criterion optimization, Guanajuato, Mexico. Lecture Notes in Computer Science (Issue), pp. 459–473. Springer (2005)
2. Bavelas, A.: Communication patterns in task-oriented groups. J. Acoust. Soc. Am. **22**, 271–282 (1950)
3. Castéra, P., Ndiaye, A., Fernandez, C., Michaud, F.: L'optimisation par essaim particulaire appliquée à la conception de composites à renforts lignocellulosiques. Revue Des Composites Et matériaux avancés **18**(2), 185–190 (2008)
4. Hu, X., Eberhart R., Shi, Y.: Particle swarm with extended memory for multiobjective optimization. IEEE Swarm Intelligence Symposium 2003. Indianapolis, IN, USA. (issue), pp. 193–197. IEEE (2003)
5. Kennedy, J.: Small worlds and mega-minds: effects of neighborhood topology on particle performance. In: Proceedings of the IEEE Congress on Evolutionary Computation. pp. 1931–1938. IEEE, Piscataway (1999)
6. Kennedy, J., Eberhart, R.: Particle swarm optimization. IEEE 530 International Conference on Neural Networks. Part 1 (of 6) 531 **4**(Issue), pp. 1942–1948. IEEE (1995)
7. Kennedy, J., Eberhart, R.C.: A discrete binary version of the particle swarm algorithm. In: Proceedings of the World Multiconference on Systemics, Cybernetics and Informatics. pp. 4104–4109. Piscataway, NI (1997)
8. Kennedy, J., Mendes, R.: Population structure and particle swarm performance. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC). pp. 1671–1676. IEEE, Honolulu, HI Piscataway (2002)
9. Mendes, R., Kennedy, J., Neves, J.: The fully informed particle swarm: simpler, maybe better. IEEE Trans. Evol. Comput. **8**(4), 204–210 (2004)
10. Michaud, F., Castera, P., Fernandez, C., Ndiaye, A.: Meta-heuristic methods applied to the design of wood–plastic composites, with some attention to environmental aspects. J. Compos. Mater. **43**(5), 533–548 (2009)
11. Poli, R., Kennedy, J., Blackwell, T.: Particle swarm optimization: an overview. Swarm Intell. **1**(1), 33–57 (2007)
12. Reyes-Sierra, M., Coello Coello, C.: Multi-objective particle swarm optimizers: a survey of the state-of-the-art. Int. J. Comput. Intell. Res. **2**(3), 287–308 (2006)
13. Watts, D.J., Strogatz, S.H.: Collective dynamics of 'small-world' networks. Nature **393**, 440–442 (1998)