

# Globally convergent hybridization of particle swarm optimization using line search-based derivative-free techniques

A. Serani, M. Diez, E.F. Campana, G. Fasano, D. Peri and U. Iemma

**Abstract** The hybrid use of exact and heuristic derivative-free methods for global unconstrained optimization problems is presented. Many real-world problems are modeled by computationally expensive functions, such as problems in simulation-based design of complex engineering systems. Objective-function values are often provided by systems of partial differential equations, solved by computationally expensive black-box tools. The objective-function is likely noisy and its derivatives are not provided. On the one hand, the use of exact optimization methods might be computationally too expensive, especially if asymptotic convergence properties are sought. On the other hand, heuristic methods do not guarantee the stationarity of their final solutions. Nevertheless, heuristic methods are usually able to provide an approximate solution at a reasonable computational cost, and have been widely applied to real-world simulation-based design optimization problems. Herein, an overall hybrid algorithm combining the appealing properties of both exact and heuristic methods is discussed, with focus on Particle Swarm Optimization (PSO) and line search-based derivative-free algorithms. The theoretical properties of the hybrid algorithm are detailed, in terms of limit points stationarity. Numerical results

---

Andrea Serani, Matteo Diez, Emilio Fortunato Campana  
CNR-INSEAN, Via di Vallerano 139, Rome, Italy  
e-mail: matteo.diez@cnr.it, emiliofortunato.campana@cnr.it

Giovanni Fasano  
Department of Management,  
University Ca' Foscari of Venice, S.Giobbe, Cannaregio 873, Venice, Italy  
e-mail: fasano@unive.it

Daniele Peri  
CNR-IAC, Via dei Taurini 19, Rome, Italy  
e-mail: daniele.peri@cnr.it

Andrea Serani, Umberto Iemma  
Roma Tre University, Via della Vasca Navale 79, Rome, Italy  
e-mail: andrea.serani@uniroma3.it, umberto.iemma@uniroma3.it

are presented for a test function and for two real-world optimization problems in ship hydrodynamics.

**Key words:** Derivative-free optimization; global optimization; particle swarm optimization; line search algorithm; hybrid optimization algorithm; simulation-based design; ship design.

## 1 Introduction

There is plenty of challenging real applications in sciences where optimization is naturally involved, and sophisticated minimization techniques are definitely necessary in order to allocate resources. In particular, these scientific tough problems often involve a remarkably large computational cost, along with large time of computation and machine resources.

Up to 15-20 years ago, to a great extent the main interest of theoreticians in optimization was for methods based on the use of derivatives. This was basically due to the following three strong reasons:

- in several cases derivatives are available when solving computational problems. In particular, they are always ‘analytically’ available if the nonlinear functions involved are known in closed form (see for instance the work by Griewank in 2000 [13]), and they can be exactly computed (not simply approximated) at reasonable cost in small-medium scale problems (Nocedal and Wright 2006 [21], Fasano and Lucidi 2009 [11]);
- strong theoretical results have been developed, both in terms of convergence and computational performance, for optimization methods where derivatives (say of first/second order) are available;
- the use of machine resources at a cheaper cost has allowed the solution of problems where derivatives can be suitably approximated by finite differences, using either coarse or fine techniques.

On the other hand, engineering design offers a huge number of real-world problems where scientists are continuously asked to apply robust methods, using the most recent theoretical advances. In particular, design problems often include functions which are not differentiable or where the use of derivatives is possibly discouraged. The following issues motivate the latter statement and give more precise guidelines for analyzing and improving optimization procedures not involving derivatives.

- For **large scale** problems, computing derivatives by finite differences might be prohibitively costly, and also Automatic Differentiation (Griewank 2000 [13]) might be of difficult application. Furthermore, the computation of derivatives by finite differences proved to be very harmful when the scale of the problem increases.
- Most of the codes for complex design problems are **parameter dependent**, and the parameters need to be ‘properly’ assessed. Their correct choice in practice

implies that the overall performance of the code needs to be optimized with respect to those parameters. Thus, an implicit optimization problem with respect to these parameters requires a solution, and surely the derivatives of the functions involved are unavailable, being the output of a code *nondifferentiable*.

- Most of the design problems need solution procedures where expensive simulations are performed. Typically, simulations are affected by **noise**, systematic errors arise and stochastic parameters are used, so that derivatives are essentially unavailable or their use may lead to completely destroy the robustness of procedures.

The issues above contribute to motivate the use of efficient and effective derivative-free methods, in order to solve a wide range of challenging problems.

In this chapter we focus on a modification of the PSO algorithm (originally proposed by Kennedy and Eberhart in 1995 [17]), for the solution of the unconstrained *global optimization problem*

$$\min_{x \in \mathbb{R}^n} f(x), \quad f: \mathbb{R}^n \rightarrow \mathbb{R}. \quad (1)$$

At present  $f(x)$  is assumed to be a continuous nonlinear, non-convex and computationally expensive function. Observe that in (1) we aim at using a modified PSO algorithm in order to detect a global minimum of  $f(x)$ , i.e. a point  $x^* \in \mathbb{R}^n$  such that  $f(x^*) \leq f(x)$ , for any  $x \in \mathbb{R}^n$ .

The reason for which we focus on a modified PSO in order to tackle (1) is that when the function  $f(x)$  is computationally costly, exact methods may be definitely too expensive to solve (1). Moreover, some exact methods are possibly unable to provide a current satisfactory approximation of a solution. In the latter cases the use of heuristic approaches may be fruitful, in particular when the computational resources and/or the time allowed for the computation are severely limited, and asymptotically convergent procedures are unaffordable. On the basis of the latter observations, PSO proved to be both effective and efficient on several practical applications (Poli et al. 2007 [23], Corazza et al. 2013 [8], Peri and Diez 2013 [22]), so that it is often the heuristics of choice.

Recalling the above considerations, in the framework of derivative-free optimization, we think that combining heuristic procedures and exact methods could be amenable, provided that:

1. the overall hybridized scheme is *efficient*, i.e. it is possibly not too expensive. A legitimate expectation is that the overall computational cost of the combined scheme is in-between the cost of (not combined) PSO and the cost of the exact method;
2. the results provided by the combined procedure are endowed with some theoretical properties, which are guaranteed by an *effective* combination of PSO and the exact method. Typical theoretical properties characterize both the convergence of sequences of points, and the stationarity of limit points of the sequences generated by the hybridized scheme.

Thus, we focus here on some modifications of PSO, where converging subsequences of iterates are generated. As a consequence, in the next section we are committed to provide clear conditions, under which PSO particles trajectories can be controlled. On the other hand, our modifications proposed for PSO guarantee that the generated sequences of iterates have subsequences converging to stationary points of the objective function (see also Hart 1996 [14], Hart 2001 [15], Trelea 2003 [28], Van den Bergh and Engelbrecht 2006 [29]). In particular, since there are in the literature theoretical results for several exact derivative-free methods (Kolda et al. 2003 [18], Conn et al. 2009 [7]), we decided to combine PSO with a line search-based derivative-free algorithm, which is to our knowledge still an unexplored issue, apart from the analysis by Campana et al. in 2009 [1]. We consider here also a numerical experience on a simplified method proposed by Campana et al. in 2009 [1], where the choice of the search directions is particularly ‘intuitive’, and preserves some relevant theoretical results.

Observe that the aim of this paper is to provide robust methods with a twofold purpose. First we would like to exploit the capability of PSO to provide a satisfactory approximation to a global solution, within a few iterations. Then, by combining PSO with an exact method, we want to force the convergence of subsequences of points toward a stationary point, which satisfies first order optimality conditions for  $f(x)$ . This paper is specifically concerned with both reporting some theoretical results and performing a valid numerical experience, to prove our theoretical conclusions.

As regards the symbols we adopt in this chapter, subscripts are used to identify the particles in a PSO scheme, whilst the superscripts indicate the iteration. By  $I$  we denote the identity matrix, and  $\|\cdot\|$  represents the Euclidean norm of a vector/matrix. Finally,  $B(c, r)$  is the real ball with center in the vector  $c$  and radius  $r > 0$ , i.e.  $B(c, r) = \{y \in \mathbb{R}^n : \|y - c\| \leq r\}$ . All the other symbols follow a very standard notation.

In Sects. 2-3 we discuss some issues related to the stability of PSO iteration. Then, the Sects. 4-5 introduce both the theory and the motivations for our modification of PSO iteration. In Sect. 6 we describe our proposal and we carry out the related convergence analysis. Sect. 7 is then devoted to report a numerical experience on a test case and real problems from ship design. Finally, Sect. 8 contains some conclusions.

## 2 Stable and unstable trajectories for PSO

The strategy of PSO for solving (1) is that of generating the  $P$  sequences  $\{x_j^k\}$ ,  $j = 1, \dots, P$ , of iterates in  $\mathbb{R}^n$ , each associated with the  $j$ -th particle of the swarm. The particles share information on the point  $p_g^k$ , at any iteration  $k$ , satisfying the condition

$$f(p_g^k) \leq f(x_j^k), \quad \forall h \leq k, \quad \forall j \in \{1, \dots, P\}.$$

To our purposes we preliminarily refer to the following PSO iteration, for any  $k \geq 0$ :

$$\begin{aligned} v_j^{k+1} &= \chi_j \left[ w_j^k v_j^k + c_j r_j \otimes (p_j^k - x_j^k) + c_g r_g \otimes (p_g^k - x_j^k) \right], \\ x_j^{k+1} &= x_j^k + v_j^{k+1}, \end{aligned} \quad (2)$$

where  $j = 1, \dots, P$  represents the  $j$ -th *particle* (i.e. the  $j$ -th sequence of iterates),  $P$  is finite, while  $v_j^k$  and  $x_j^k$  are  $n$ -real vectors, which respectively represent the *speed* (i.e. the search direction) and the *position* of the  $j$ -th particle at step  $k$ . The real bounded coefficients  $c_j$  and  $c_g$  are typically given at the outset of iteration  $k = 0$ , and are possibly not modified unless stagnation arises. On the other hand, with  $r_j \otimes (p_j^k - x_j^k)$  (similarly with  $r_g \otimes (p_g^k - x_j^k)$ ) we indicate that every entry of the vector  $(p_j^k - x_j^k)$  is multiplied by a different value of  $r_j$ , which is a random parameter in the uniform distribution between 0 and 1. Finally, for a given  $k \geq 0$ , the  $n$ -real vectors  $\{p_j^k\}$  satisfy the conditions

$$f(p_j^k) \leq f(x_j^\ell), \quad \forall \ell \leq k, \quad p_j^k \in \{x_j^\ell\}, \quad (3)$$

moreover,  $\chi_j$  (*constriction coefficient*) and  $w_j^k$  (*inertia*) are positive bounded coefficients. In words the vector  $p_j^k$  represents the ‘best position’ in the  $j$ -th subsequence up to iteration  $k$ , while  $p_g^k$  is the ‘best position’ among all the vectors  $\{p_1^k, \dots, p_P^k\}$ . A keynote issue in PSO is that an effective choice of the coefficients  $\chi$ ,  $w^k$ ,  $c_j$  and  $c_g$  is often problem dependent, whereas several standard settings for them have been proposed in the literature (see Poli 2008 [24]). Notwithstanding the latter fact, more precise rules for assessing the coefficients in (2) are still sought, with a specific reference to eventually avoid stagnation of PSO iteration.

In order to possibly generalize the recurrence (2), we can assume that the speed  $v_j^{k+1}$  depends on all the  $P$  vectors  $(p_h^k - x_j^k)$  (see also Mendes 2004 [20]),  $h = 1, \dots, P$ , and not only on the pair of vectors  $(p_j^k - x_j^k)$ ,  $(p_g^k - x_j^k)$ . The resulting new iteration represents the so called Fully Informed PSO (FIPSO). The latter generalization is possibly unessential for our purposes, so that hereafter we limit our analysis to the more standard iteration (2).

Observe that in order to give rules, which ensure that PSO trajectories satisfy suitable conditions, we need to impose some restrictions to the coefficients in (2). In particular, after reviewing the literature we remark that the following (not mutually exclusive) conditions can be reasonably expected to hold for particles trajectories in PSO:

- (1) the sequence  $\{x_j^k\}$  converges to  $x_j^*$ , for any  $j = 1, \dots, P$ , with  $x_1^* = \dots = x_P^*$ ;
- (2) the sequence  $\{x_j^k\}$  converges to  $x_j^*$ , for any  $j = 1, \dots, P$ , but possibly  $x_j^* \neq x_\ell^*$ , with  $1 \leq x_j^* \neq x_\ell^* \leq P$ ;
- (3) the sequence  $\{x_j^k\}$  is not diverging for any  $j = 1, \dots, P$ , i.e.  $\|x_j^k\| < +\infty$ , for any  $j = 1, \dots, P$  and any  $k \geq 0$ .

We highlight that different bounds can be imposed on the coefficients  $\chi$ ,  $w^k$ ,  $c_j$  and  $c_g$  in (2), in order to ensure that either of the three conditions (1)-(3) is fulfilled. It is also not difficult to realize that (1) implies (2) (but not *viceversa*) and (2) implies (3) (but not *viceversa*). Thus, the conditions on the coefficients of PSO ensuring (3), are expected to be both weak enough and sufficiently general to allow a wide exploration of the search space. For the latter reason, in this paper we prefer to study and analyze the case (3), while the interested reader can possibly refer to PSO literature (see also Sect. 4) for the analysis on the cases (1)-(2). Now, by (2) let us preliminarily consider the following assumption, in order to simplify our notation.

**Assumption 1** *We assume in (2) that  $\chi_j = \chi > 0$ ,  $c_j = c > 0$  and  $r_j = r > 0$ , for any  $j = 1, \dots, P$ . Moreover,  $c_g = \bar{c} > 0$ ,  $r_g = \bar{r} > 0$  and  $w_j^k = w$ , for any  $j = 1, \dots, P$  and any  $k \geq 0$ .*

Then (see also Campana et al. 2010 [2]), using Assumption 1 the iteration (2) is equivalent to the *discrete stationary (time-invariant) system*

$$X_j(k+1) = \begin{pmatrix} \chi w I & -\chi(cr + \bar{c}\bar{r})I \\ \chi w I & [1 - \chi(cr + \bar{c}\bar{r})]I \end{pmatrix} X_j(k) + \begin{pmatrix} \chi(cr p_j^k + \bar{c}\bar{r} p_g^k) \\ \chi(cr p_j^k + \bar{c}\bar{r} p_g^k) \end{pmatrix}, \quad (4)$$

where

$$X_j(k) = \begin{pmatrix} v_j^k \\ x_j^k \end{pmatrix} \in \mathbb{R}^{2n}, \quad k \geq 0. \quad (5)$$

For a given  $j$ , the vectors  $\{X_j(k)\}$  identify a sequence of points in  $\mathbb{R}^n$  and represent indeed the trajectory of the  $j$ -th particle in the state space  $\mathbb{R}^{2n}$ . By definition, since  $X_j(k)$  represents a *state vector*, it can be split into the so called *free response*  $X_{j,\mathcal{L}}(k)$  and the *forced response*  $X_{j,\mathcal{F}}(k)$  (see also Sarachik 1997 [26]), such that

$$X_j(k) = X_{j,\mathcal{L}}(k) + X_{j,\mathcal{F}}(k), \quad (6)$$

being

$$X_{j,\mathcal{L}}(k) = \Phi_j(k)X_j(0), \quad X_{j,\mathcal{F}}(k) = \sum_{\tau=0}^{k-1} H_j(k-\tau)U_j(\tau), \quad (7)$$

and (with a little computation)

$$\Phi_j(k) = \begin{pmatrix} \chi w I & -\chi(cr + \bar{c}\bar{r})I \\ \chi w I & [1 - \chi(cr + \bar{c}\bar{r})]I \end{pmatrix}^k, \quad (8)$$

$$H_j(k-\tau) = \begin{pmatrix} \chi w I & -\chi(cr + \bar{c}\bar{r})I \\ \chi w I & [1 - \chi(cr + \bar{c}\bar{r})]I \end{pmatrix}^{k-\tau-1}, \quad (9)$$

$$U_j(\tau) = \begin{pmatrix} \chi(cr p_j^\tau + \bar{c}\bar{r}p_g^\tau) \\ \chi(cr p_j^\tau + \bar{c}\bar{r}p_g^\tau) \end{pmatrix}. \quad (10)$$

We highlight the important fact that the free response  $X_{j,\mathcal{L}}(k)$  in (6)-(7) only depends on the initial point  $X_j(0)$ , and is not affected by changes of the vectors  $p_j^\tau, p_g^\tau, \tau \geq 0$ . The latter observation is of great interest, in order to assess rules for the parameters  $\chi, w, c, r, \bar{c}$  and  $\bar{r}$ , as the next section shows.

### 3 Issues on assessing parameters in PSO

As described in the last section, the fruitful choice of the parameters in PSO is to a large extent guided by a couple of issues: the efficiency of the overall scheme and the necessity of guaranteeing at least non-diverging trajectories of the particles. As regards the first issue, the literature of PSO provides several suggestions which have proved to be effective in most cases (see for instance Poli 2009 [25], Christopher et al. 2014 [4]). Conversely, some more recent papers, concerned with studying the stability of particles trajectory, have detailed some restrictions on PSO parameters in order to dynamically control the trajectory of particles, and make them more accurately predictable. On this guideline, papers like Kadiramanathan et al. in 2006 [16], and Gazi in 2012 [12] are advisable, since they contain clear indications on the latter issue.

Here we want to propose a unified approach for parameters assessment in PSO, using the reformulation (4)-(5). In practice (see also Sarachik 1997 [26]), as long as Assumption 1 holds, our perspective is that of using classic analysis for discrete linear systems in order to deduce bounds on PSO parameters. On the other hand, we want to carry on our analysis as rigorously as possible, so that we will separately develop formal conditions for the following three purposes:

- (a) define *necessary conditions* (possibly not sufficient) on PSO parameters, so that particles trajectories are not diverging, i.e. the quantities  $\|X_j(k)\|$  are limited, for any  $j = 1, \dots, P$  and any  $k \geq 0$ ;
- (b) ensure *no stagnation* (see Gazi 2012 [12]) for a modified PSO scheme;
- (c) possibly introduce simple modifications to PSO, so that the resulting scheme is *globally convergent* to stationary points, i.e. for any choice of the initial positions  $x_1^0, \dots, x_p^0$  of the particles, the scheme generates sequences of points like  $\{y^k\}$ , such that

$$\liminf_{k \rightarrow \infty} \|\nabla f(y^k)\| = 0.$$

Observe that the latter condition substantially guarantees that possibly ‘true’ minima for  $f(x)$  are eventually outreached, even in case stagnation arises. We strongly remark that though we used the symbol  $\nabla f(x)$ , the latter relation will be proved *without using any information* on  $\nabla f(x)$ , so that a completely derivative-free method will be developed.

In the current section we consider the issue (a), while in the next section issues (b) and (c) will be analyzed in detail.

As well known, for a discrete linear system like (4)-(5) (see for instance Sarachik 1997 [26]), if the  $j$ -th trajectory  $\{X_j(k)\}$  in (6) is non-diverging, then

$$\lim_{k \rightarrow \infty} X_j(k) = \lim_{k \rightarrow \infty} X_{j\mathcal{F}}(k), \quad j = 1, \dots, P.$$

In other words, the free response  $X_{j\mathcal{L}}(k)$  is bounded away from zero only for finite values of the index  $k$ , and

$$\lim_{k \rightarrow \infty} X_{j\mathcal{L}}(k) = 0. \quad (11)$$

This introduces a possible rule to address bounds for PSO parameters, since relation (11) imposes some restrictions to the eigenvalues of matrix  $\Phi_j(k)$  in (8). Observing that  $\Phi_j(k) = \Phi_j(1)^k$ , for any  $k \geq 0$ , the  $2n$  eigenvalues of the unsymmetric matrix  $\Phi_j(1)$  are real. In particular, by setting for the sake of simplicity in (8)

$$a = \chi w, \quad \omega = \chi(cr + \bar{c}\bar{r}), \quad (12)$$

we can prove that after some computation the matrix  $\Phi_j(1)$  has two distinct eigenvalues  $\lambda_{j1}$  and  $\lambda_{j2}$  given by

$$\begin{aligned} \lambda_{j1} &= \frac{1 - \omega + a - [(1 - \omega + a)^2 - 4a]^{1/2}}{2}, \\ \lambda_{j2} &= \frac{1 - \omega + a + [(1 - \omega + a)^2 - 4a]^{1/2}}{2}, \end{aligned} \quad (13)$$

each of them with algebraic multiplicity  $n$ . Thus, a necessary (but in general not sufficient) condition for the  $j$ -th trajectory  $\{X_j(k)\}$  to be non-diverging and satisfy  $\lim_{k \rightarrow \infty} X_{j\mathcal{L}}(k) = 0$ , is the following, which yields the required conditions on the coefficients of PSO iteration.

**Lemma 1.** *Consider the PSO iteration (2). Let Assumption 1 hold. Let the eigenvalues  $\lambda_{j1}$  and  $\lambda_{j2}$  in (13) satisfy the conditions*

$$|\lambda_{j1}| < 1 \quad |\lambda_{j2}| < 1, \quad (14)$$

*for any  $j = 1, \dots, P$ . Then, the sequence  $\{X_{j\mathcal{L}}(k)\}$  satisfies  $\lim_{k \rightarrow \infty} X_{j\mathcal{L}}(k) = 0$ , and condition (14) is a necessary condition for the trajectory  $\{X_j(k)\}$  to be non-diverging.*

Note that most of the typical settings for PSO parameters proposed in the literature (see e.g. Clerc and Kennedy 2002 [5], Poli 2009 [25], Christopher et al. 2014 [4]), satisfy the condition (14). Moreover, Lemma 1 does not guarantee that the sequence  $\{X_j(k)\}$ , for a given  $j$ , is converging, and indeed it possibly does not admit even limit points. This means that condition (14) only provides a result for item (a), but is definitely inadequate to treat items (b) and (c). This also implies that for instance



if (14) holds, then possibly the trajectory  $\{X_j(k)\}$  is not converging, or some of its subsequences converge to the point  $x_j^*$  which is not a minimum and does not satisfy the property

$$f(x_j^*) \leq f(x), \quad \forall x \in B(x_j^*, \varepsilon), \quad \varepsilon > 0.$$

In the next sections we focus on a rigorous analysis of the latter issue. I.e., under mild assumptions we propose a modified PSO scheme such that, if the function  $f(x)$  is continuously differentiable, the sequence  $\{x_1^1, \dots, x_p^1, \dots, x_1^k, \dots, x_p^k\}$  admits *stationary limit points* for  $f(x)$ , so that

$$\liminf_{k \rightarrow \infty} \|\nabla f(x_j^k)\| = 0 \quad \text{or} \quad \lim_{k \rightarrow \infty} \|\nabla f(x_j^k)\| = 0. \quad (15)$$

As the reader can expect, there is a theoretical and computational evidence that fulfilling condition (15) may be met (in several real applications) at the expense of a reasonably larger computational cost, with respect to the standard PSO iteration (2).

## 4 PSO and stationarity

In Sect. 1 we have detailed some motivations, to explain why in the last two decades design optimization and simulation-based optimization have required effective and robust derivative-free optimization methods. Exploiting also the recent advances on parallel computing, the last two decades have seen in particular the blow up of a remarkably effective class of optimization methods, endowed with complete convergence analysis and competitive performance: namely *direct search methods*. The latter class (see Kolda et al. 2003 [18]) counts several optimization methods, which do not use derivatives but basically rely on “*the ranks of a countable set of function values*” (Kolda et al. 2003 [18]), i.e. on comparing the objective function values in specific points of the search space.

Among direct search methods we focus here on a subclass of iterative techniques, which is usually addressed in the literature as *Generating Set Search* (GSS). In the latter class, the main idea is that of decreasing the objective function at each iteration, on a *cone* in  $\mathbb{R}^n$  generated by suitable search directions. *Pattern search methods* are in the GSS class, and have the distinguishing feature of enforcing, at each iteration, a simple decrease of the objective function. Conversely, also *line search-based* derivative-free methods are iterative schemes in GSS class, however they impose at each iteration a so called *sufficient reduction* of  $f(x)$ . We want to show that global convergence properties of a modified PSO scheme may be obtained by properly combining PSO with a line search-based derivative-free method, so that convergence to stationary points can be forced at a reasonable cost (see also items (b) and (c) of Sect. 3). On this guideline, there is plenty of examples where evolutionary strategies are combined with GSS schemes and yield globally convergent algorithms (see for instance Hart 1996 [14], Vaz and Vicente 2007 [30], Vaz and Vicente 2009 [31]). In particular, in the last reference PSO is hybridized within

a pattern search framework, and a resulting method converging to stationary points is given.

Observe that in the literature of derivative-free methods we can also find PSO-based approaches combined with a trust-region framework (see Vaz and Vicente 2007 [30], Vaz and Vicente 2009 [31]), in order to provide again globally convergent methods to stationary points.

In this section we consider the solution of the problem (1), and we focus on a modified PSO scheme, combined with a line search-based derivative-free algorithm. We study in particular the nature of limit points of the sequences  $\{x_j^k\}$ ,  $j = 1, \dots, P$ , when Assumption 1 holds. However, we think that to have a better insight in our analysis, the following very preliminary results (see also Conn et al. 2009 [7]) can help the reader grasp the importance of the GSS class, in order to ensure convergence to stationary points.

**Definition 1.** Given the set of vectors  $D = \{d_1, \dots, d_m\}$  of  $\mathbb{R}^n$ , we say that  $D$  is a Positively Spanning Set (PSS) if for any vector  $u \in \mathbb{R}^n$  we have

$$u = \sum_{i=1}^m \alpha_i d_i, \quad \alpha_i \geq 0,$$

i.e. any vector  $u$  of  $\mathbb{R}^n$  can be expressed as the weighted sum of the vectors in  $D$ , using nonnegative weights.

Thus, a PSS substantially provides a set of vectors which positively span the space  $\mathbb{R}^n$ . It can be easily proved that if  $D$  is a PSS of  $\mathbb{R}^n$ , then its cardinality must be at least  $n + 1$ . It is very easy to define PSSs; simple examples of them in  $\mathbb{R}^2$  are given in Figure 1, where  $m = 4$  (*top* and *bottom*) and  $m = 3$  (*middle*). In addition, there is the following nice property of PSSs that we are going to exploit in our proposal. If the point  $x \in \mathbb{R}^n$  is not stationary for  $f$  in (1) (i.e.  $\nabla f(x) \neq 0$ ), given the PSS  $D$  in  $\mathbb{R}^n$ , there exists at least one vector, say  $\hat{d} \in D$ , such that  $\nabla f(x)^T \hat{d} < 0$ , meaning that the direction  $\hat{d}$  is of descent for  $f(x)$  at  $x$ . The latter fact ensures that if the current point is not stationary, and a PSS is available, roughly speaking there is at least one *direction of descent* for  $f(x)$  in the PSS.

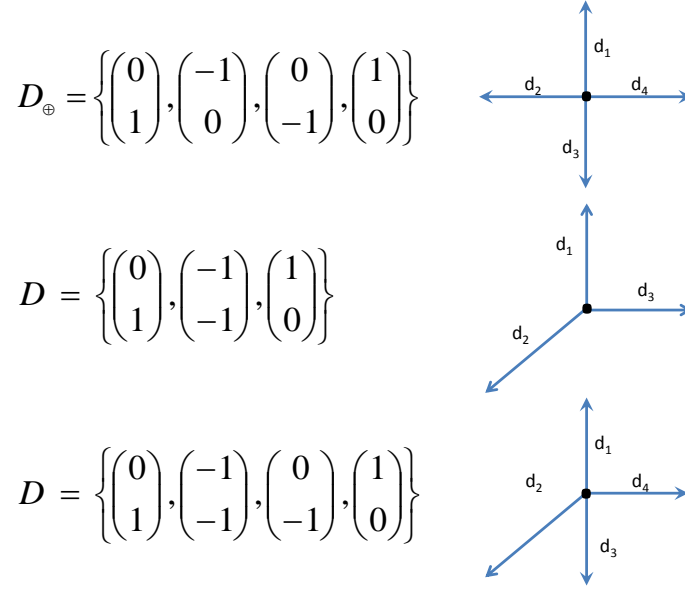
A so called *cosine measure*  $cm(D)$  can be associated to the PSS  $D$  of  $\mathbb{R}^n$ , defined as follows.

**Definition 2.** Given the PSS  $D = \{d_1, \dots, d_m\}$  in  $\mathbb{R}^n$ , we define the cosine measure  $cm(D)$  of  $D$  as

$$cm(D) = \min_{v \in \mathbb{R}^n \setminus \{0\}} \max_{d_i \in D} \left\{ \frac{v^T d_i}{\|v\| \|d_i\|} \right\},$$

being always  $cm(D) > 0$ .

By Definition 2 the quantity  $cm(D)$  clearly represents a measure of the least orthogonal projection of any vector in  $\mathbb{R}^n$  on vectors in  $D$ . As a consequence, if  $cm(D) \rightarrow 0$  then  $D$  might be not a 'good' PSS and consequently it might be difficult to find a descent direction for  $f$  in  $D$ . The following result clarifies the importance of introducing PSSs in derivative-free optimization, in order to characterize stationary points, without using any information on the gradient of  $f$  in (1).



**Fig. 1** Examples of PSSs in  $\mathbb{R}^2$ . The subscript ‘ $\oplus$ ’ in the uppermost PSS means that the vectors in the set are the coordinate unit vectors  $\pm e_i$ ,  $i = 1, \dots, n$ .

**Theorem 1.** Let  $D = \{d_1, \dots, d_m\}$  be a PSS. Suppose the function  $f(x)$  in (1) is continuously differentiable in  $\mathbb{R}^n$  and the gradient  $\nabla f(x)$  satisfies the Lipschitz condition

$$\|\nabla f(y) - \nabla f(x)\| \leq v\|y - x\|, \quad \forall y \in \overset{\circ}{B}(x, \alpha\delta), \quad \delta = \max_{1 \leq i \leq m} \|d_i\|,$$

for some  $v > 0$  and  $\alpha > 0$ . If  $f(x) \leq f(x + \alpha d_i)$ ,  $i = 1, \dots, m$ , then the following bound holds for the gradient of  $f(x)$

$$\|\nabla f(x)\| \leq \frac{v}{2} \frac{1}{cm(D)} \alpha \delta. \quad (16)$$

*Proof.* Since  $D$  is a PSS, there exists at least one vector in the set  $\{d_i\}$ , say  $\hat{d} \in D$ , such that

$$cm(D) \leq \frac{-\nabla f(x)^T \hat{d}}{\|\nabla f(x)\| \|\hat{d}\|},$$

hence, recalling that  $\alpha$  is positive, we have equivalently

$$cm(D) \|\nabla f(x)\| \|\hat{d}\| \alpha \leq -\nabla f(x)^T (\alpha \hat{d}). \quad (17)$$

On the other hand, the hypothesis of continuous differentiability of  $f(x)$  allows to apply the Mean Value Theorem in the integral form, being for any  $d \in D$

$$f(x + \alpha d) = f(x) + \int_0^1 \nabla f[x + t\alpha d]^T (\alpha d) dt,$$

or equivalently

$$0 \leq f(x + \alpha d) - f(x) = \int_0^1 \nabla f[x + t\alpha d]^T (\alpha d) dt. \quad (18)$$

Combining (17) and (18) we obtain for the direction  $\hat{d}$

$$\begin{aligned} cm(D) \|\nabla f(x)\| \|\hat{d}\| \alpha &\leq -\nabla f(x)^T (\alpha \hat{d}) + \int_0^1 \nabla f[x + t\alpha \hat{d}]^T (\alpha \hat{d}) dt \\ &\leq \int_0^1 |\nabla f[x + t\alpha \hat{d}]^T (\alpha \hat{d}) - \nabla f(x)^T (\alpha \hat{d})| dt \\ &\leq \int_0^1 \|\nabla f[x + t\alpha \hat{d}] - \nabla f(x)\| \|\alpha \hat{d}\| dt \\ &\leq v \int_0^1 t \|\alpha \hat{d}\|^2 dt \leq \frac{v}{2} \alpha^2 \|\hat{d}\|^2, \end{aligned}$$

which immediately yields (16).

Loosely speaking, in order to suggest the reader the importance of Theorem 1, it can be rephrased in the following simple way. If the PSS  $D$  is available and the value  $f(x)$  cannot be decreased on points along all the directions in  $D$ , then it means that the iterate  $x$  is a stationary point. Indeed, in the latter case, from (16) we have  $\lim_{\alpha \rightarrow 0} \|\nabla f(x)\| = 0$ . Also note that if the PSS  $D$  is poor (i.e.  $cm(D)$  is small), then the bound on the gradient (16) is poor accordingly.

Since in our proposal we combine PSO with a line search-based derivative-free method, which relies on the use of PSSs, with our proposal we will be able to characterize stationarity conditions for a modified PSO scheme, without recurring to any information on the gradient  $\nabla f(x)$ . In the next section we describe some basic properties of the line search-based derivative-free method we couple with PSO.

## 5 Preliminaries on the line search-based method adopted

We consider in this section the proposal by Lucidi and Sciandrone in 2002 [19], which includes some line search-based derivative-free methods. Since it is our intention to reduce the complexity of our proposal as much as possible, the next result represents a simplified version of the material of Lucidi and Sciandrone in 2002 [19].

**Table 1** The Line search-based Derivative-free Algorithm LS-DF (see also Lucidi and Sciandrone 2002 [19]).

<p><b>Step 0.</b> Set <math>k = 0</math>; choose <math>x^0 \in \mathbb{R}^n</math>, set <math>\bar{\alpha}^0 &gt; 0</math>, <math>\gamma &gt; 0</math>, <math>\theta \in (0, 1)</math>.</p> <p><b>Step 1.</b> If there exists <math>y^k \in \mathbb{R}^n</math> such that <math>f(y^k) \leq f(x^k) - \gamma\bar{\alpha}^k</math>, then go to <b>Step 4</b>.</p> <p><b>Step 2.</b> If there exists <math>j \in \{1, \dots, m\}</math> and an <math>\alpha^k \geq \bar{\alpha}^k</math> such that</p> $f(x^k + \alpha^k d_j^k) \leq f(x^k) - \gamma(\alpha^k)^2,$ <p>then set <math>y^k = x^k + \alpha^k d_j^k</math>, set <math>\bar{\alpha}^{k+1} = \alpha^k</math> and go to <b>Step 4</b>.</p> <p><b>Step 3.</b> Set <math>\bar{\alpha}^{k+1} = \theta\bar{\alpha}^k</math> and <math>y^k = x^k</math>.</p> <p><b>Step 4.</b> Find <math>x^{k+1}</math> such that <math>f(x^{k+1}) \leq f(y^k)</math>, set <math>k = k + 1</math> and go to <b>Step 1</b>.</p>
--

**Proposition 1.** Let  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ , with  $f$  continuously differentiable in  $\mathbb{R}^n$ . Suppose that the points in the sequence  $\{x^k\}$  are bounded. Suppose the directions  $d_1^k, \dots, d_m^k$  are bounded and form a positively spanning set of  $\mathbb{R}^n$ . Then, the following stationarity condition holds

$$\lim_{k \rightarrow \infty} \|\nabla f(x^k)\| = 0 \quad \text{if and only if} \quad \lim_{k \rightarrow \infty} \sum_{j=1}^m \min \left\{ 0, \nabla f(x^k)^T d_j^k \right\} = 0. \quad (19)$$

Let us consider the sequence  $\{x^k\}$  in (19) and Theorem 1. By Proposition 1 necessary and sufficient conditions of stationarity for the sequence  $\{x^k\}$  can be accomplished by simply exploiting at any iterate  $x^k$  the function  $f(x)$  (through its directional derivative  $\nabla f(x^k)^T d_j^k$ ), along the search directions  $d_1^k, \dots, d_m^k$ . Table 1 details a line search-based derivative-free method for unconstrained minimization, which uses the results of Proposition 1. In Lucidi and Sciandrone (2002) [19] a complete convergence analysis was developed for the Algorithm LS-DF and the following conclusion was proved (see e.g. Proposition 5.1 in Lucidi and Sciandrone (2002) [19]).

**Proposition 2.** Suppose the directions  $d_1^k, \dots, d_m^k$  satisfy Proposition 1. Consider the sequence  $\{x^k\}$  generated by the Algorithm LS-DF and let the level set  $\mathcal{L}_0 = \{x \in \mathbb{R}^n : f(x) \leq f(x^0)\}$  be compact. Then we have

$$\liminf_{k \rightarrow \infty} \|\nabla f(x^k)\| = 0. \quad (20)$$

Note that the condition (20) is weaker than (19), and in principle is met only asymptotically by the Algorithm LS-DF. Of course, recalling also the results in Theorem 1, a practical stopping condition of Algorithm LS-DF could be obtained by moni-

toring the steplength  $\bar{\alpha}^k$  at Steps 2 and 3. The algorithm can stop when  $\bar{\alpha}^k$  becomes sufficiently small. Also observe that at Step 4 the point  $x^{k+1}$  might be computed for instance by any heuristic procedure; nevertheless, we can set in any case  $x^{k+1} \equiv y^k$ , since convergence analysis does not require  $f(x^{k+1}) < f(y^k)$ .

As a final consideration, note that relation (20) may be strongly strengthened by choosing a different (and computationally more expensive) strategy at Step 2 of the Algorithm LS-DF. Indeed, instead of requiring that at Step 2 just one direction is of sufficient decrease for the objective function (i.e.  $f(x^k + \alpha^k d_j^k) \leq f(x^k) - \gamma(\alpha^k)^2$  for *at least one*  $j \in \{1, \dots, m\}$ ), we can exploit  $f(x)$  along *all* the directions  $\{d_1^k, \dots, d_m^k\}$  in the PSS. The resulting algorithm is Algorithm LS-DF<sub>+</sub> in Table 2. We recall that, for the sake of simplicity and in order to keep the computational cost as low as possible, we will couple PSO only with Algorithm LS-DF. In the following proposition we summarize convergence properties also for Algorithm LS-DF<sub>+</sub>: we remark that they are definitely stronger than the result in Proposition 2.

**Proposition 3.** *Suppose the directions  $d_1^k, \dots, d_m^k$  satisfy Proposition 1. Consider the sequence  $\{x^k\}$  generated by the Algorithm LS-DF<sub>+</sub> and let the level set  $\mathcal{L}_0 = \{x \in \mathbb{R}^n : f(x) \leq f(x^0)\}$  be compact. Then we have*

$$\lim_{k \rightarrow \infty} \|\nabla f(x^k)\| = 0. \quad (21)$$

It is evident that in Algorithm LS-DF<sub>+</sub> the stronger convergence result is obtained at the expense of a larger computational cost in Step 2. In addition, the procedure `Line search()` is aimed to determine a possible expansion of the steplength  $\alpha_j^k$ .

## 6 A hybrid algorithm

In this section we propose an hybrid algorithm, obtained by coupling the PSO scheme described in Sect. 2 with the algorithm in Table 1. We remind the reader that the resulting method should be endowed with both the (local) convergence properties of Algorithm LS-DF and the (global) strategies of exploration of PSO (see also Sect. 1).

Of course, it is far obvious that simply alternating a finite sequence of steps of PSO and a finite sequence of steps of Algorithm LS-DF would provide a method satisfying a property similar to Proposition 2. However, the latter strategy might be a blind sequential application of two different algorithms, which does not exploit their peculiarities. On the contrary, we prefer to consider a scheme which at once both *exploits* (local strategy) and *explores* (global strategy) the objective function. Thus, we consider here a PSO-based method which attempts to detect a global minimum of the objective function, while retaining the asymptotic convergence properties of Algorithm LS-DF.

Our proposal (namely Algorithm LS-DF\_PSO) is summarized in Table 3 and its convergence properties to stationary points are summarized in the next proposition (see also [19]).

**Table 2** The Line search-based Derivative-free Algorithm LS-DF<sub>+</sub> in Lucidi and Sciandrone (2002) [19].

**Step 0.** Set  $k = 0$ . Choose  $x^0 \in \mathbb{R}^n$  and  $\bar{\alpha}_j^0 > 0$ ,  $j = 1, \dots, n+1$ ,  $\gamma > 0$ ,  $\delta \in (0, 1)$ ,  $\theta \in (0, 1)$ .

**Step 1.** Set  $j = 1$  and  $y_1^k = x^k$ .

**Step 2.** If  $f(y_j^k + \bar{\alpha}_j^k d_j^k) \leq f(y_j^k) - \gamma(\bar{\alpha}_j^k)^2$  then

compute  $\alpha_j^k$  by Line search( $\bar{\alpha}_j^k, y_j^k, d_j^k, \gamma, \delta$ ) and set  $\bar{\alpha}_j^{k+1} = \alpha_j^k$ ;

else set  $\alpha_j^k = 0$ , and  $\bar{\alpha}_j^{k+1} = \theta \bar{\alpha}_j^k$ .

Set  $y_{j+1}^k = y_j^k + \alpha_j^k d_j^k$ .

**Step 3.** If  $j < m$  then set  $j = j + 1$  and go to **Step 2**.

**Step 4.** Find  $x^{k+1}$  such that  $f(x^{k+1}) \leq f(y_{n+1}^k)$ , set  $k = k + 1$  and go to **Step 1**.

Line search( $\bar{\alpha}_j^k, y_j^k, d_j^k, \gamma, \delta$ ):

Compute the steplength  $\alpha_j^k = \min \{ \bar{\alpha}_j^k / \delta^h, h = 0, 1, \dots \}$  such that

$$f(y_j^k + \alpha_j^k d_j^k) \leq f(y_j^k) - \gamma(\alpha_j^k)^2,$$

$$f\left(y_j^k + \frac{\alpha_j^k}{\delta} d_j^k\right) \geq \max \left[ f(y_j^k + \alpha_j^k d_j^k), f(y_j^k) - \gamma \left(\frac{\alpha_j^k}{\delta}\right)^2 \right].$$

**Proposition 4.** Suppose the directions  $d_1^k, \dots, d_m^k$  satisfy Proposition 1. Consider the sequence  $\{x^k\}$  generated by the Algorithm LS-DF\_PSO and let the level set  $\mathcal{L}_0 = \{x \in \mathbb{R}^n : f(x) \leq f(x^0)\}$  be compact. Then we have

$$\liminf_{k \rightarrow \infty} \|\nabla f(x^k)\| = 0. \quad (22)$$

*Proof.* Observe that the Algorithm LS-DF and the Algorithm LS-DF\_PSO differ only at Step 1 and Step 4. Indeed, Step 1 and Step 4 of Algorithm LS-DF\_PSO are simply obtained from the corresponding steps of Algorithm LS-DF, observing that the iterates  $y^k$  (Step 1) and  $x^{k+1}$  (Step 4) are computed by using PSO. Therefore,

convergence properties of the sequence generated by the Algorithm LS-DF\_PSO are basically inherited from Proposition 2.

**Table 3** The Line search-based Derivative-free Algorithm LS-DF\_PSO.

<p><b>Data.</b> Set <math>k = 0</math>; choose <math>x^0 \in \mathbb{R}^n</math> and <math>z_j^0, v_j^0 \in \mathbb{R}^n</math>, <math>j = 1, \dots, P</math>. Set <math>\bar{\alpha}^0 &gt; 0</math>, <math>\gamma &gt; 0</math>, <math>\theta \in (0, 1)</math>.</p> <p><b>Step 1.</b> Set <math>h_k \geq 1</math> integer. Apply <math>h_k</math> PSO iterations considering the <math>P</math> particles with respective initial velocities and positions <math>v_j^k</math> and <math>z_j^k</math>, <math>j = 1, \dots, P</math>. Set <math>y^k = \operatorname{argmin}_{1 \leq j \leq P, \ell \leq h_k} \{f(z_j^\ell)\}</math>. If <math>f(y^k) \leq f(x^k) - \gamma \bar{\alpha}^k</math>, then set <math>v_j^k = v_j^{k+h_k}</math> and <math>z_j^k = z_j^{k+h_k}</math>, and go to <b>Step 4</b>.</p> <p><b>Step 2.</b> If there exists <math>j \in \{1, \dots, m\}</math> and an <math>\alpha^k \geq \bar{\alpha}^k</math> such that</p> $f(x^k + \alpha^k d_j^k) \leq f(x^k) - \gamma (\alpha^k)^2,$ <p>then set <math>y^k = x^k + \alpha^k d_j^k</math>, <math>\bar{\alpha}^{k+1} = \alpha^k</math> and go to <b>Step 4</b>.</p> <p><b>Step 3.</b> Set <math>\bar{\alpha}^{k+1} = \theta \bar{\alpha}^k</math> and <math>y^k = x^k</math>.</p> <p><b>Step 4.</b> Set <math>q_k \geq 1</math> integer. Apply <math>q_k</math> PSO iterations considering the <math>P</math> particles with respective initial velocities and positions <math>v_j^k</math> and <math>z_j^k</math>, <math>j = 1, \dots, P</math>. Set <math>x^{k+1} = \operatorname{argmin}_{1 \leq j \leq P, \ell \leq q_k} \{f(z_j^\ell)\}</math>; if <math>x^{k+1}</math> satisfies <math>f(x^{k+1}) \leq f(y^k)</math>, then set <math>k = k + 1</math> and go to <b>Step 1</b>.</p>
---

We conclude this section by highlighting that similarly to Algorithm LS-DF\_PSO, instead of using Algorithm LS-DF, we can couple PSO with the Algorithm LS-DF<sub>+</sub>. The resulting hybrid scheme, would be much similar to Algorithm LS-DF\_PSO but more expensive than the algorithm in Table 3. Moreover, it would be also endowed with convergence properties similar to those reported in Proposition 3, so that the condition (22), which is worth for Algorithm LS-DF\_PSO, would be reinforced with a condition like (21).

## 7 Numerical results

Numerical results are presented for the Rosenbrock function and for two real-world optimization problems in ship hydrodynamics. Specifically, a four-parameters shape optimization of a catamaran is investigated for (a) resistance reduction in calm water at fixed speed, and (b) expected resistance reduction in wave, taking into account stochastic sea state and speed. For all optimization problems, the deterministic implementation of PSO presented by Serani et al. in 2014 [27] is used for extension to

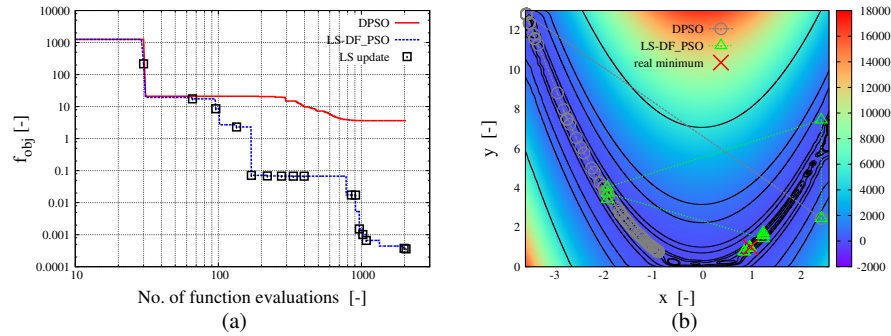


LS-DF\_PSO. A number of particles equal to  $4n$  is used, with initialization over the variables domain by Hammersely sequence sampling, and PSO coefficients given by Clerc in 2006 [6], i.e.,  $\chi = 0.721$ ,  $c_1 = c_2 = 1.655$  (see the work by Serani et al. in 2014 [27] for details). LS-DF\_PSO parameters are set as  $h_k = 1$ ,  $\gamma = 10^{-3}$ ,  $\theta = 0.5$ ,  $\alpha^k = 0.25$  of the design variable range.

Specifically, the minimization in  $\mathbb{R}^2$  of the Rosenbrock function,

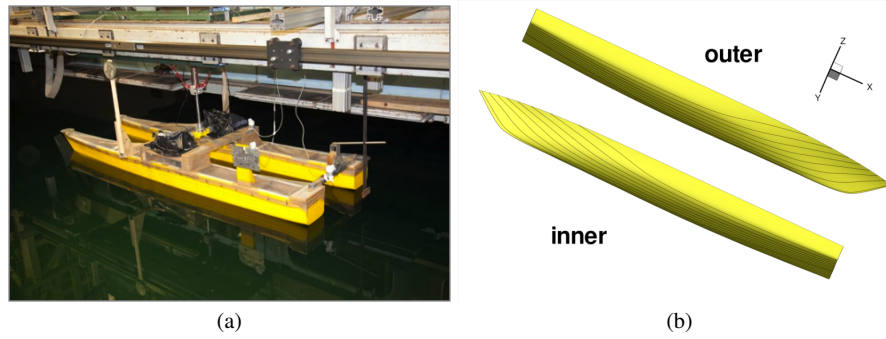
$$f(x,y) = (a-x)^2 + b(y-x^2)^2 \tag{23}$$

with  $a = 1$ ,  $b = 100$ ,  $-20 \leq x \leq 20$  and  $-20 \leq y \leq 20$ , is used as an explanatory test problem. Figure 2 (a) shows the convergence of the LS-DF\_PSO algorithm, compared to the standard PSO. Black squares indicates LS-DF\_PSO iterations where LS is used to improve the optimum location. Figure 2 (b) shows a comparison of the algorithms' convergence in a close up of the variables domain. The global-optimum location history is depicted, along with the real minimum, which is located at  $x = 1, y = 1$ . The beneficial effects of using PSO with LS are evident, providing a faster and more effective convergence to the optimum, along with the identification of the proper region of the global optimum.



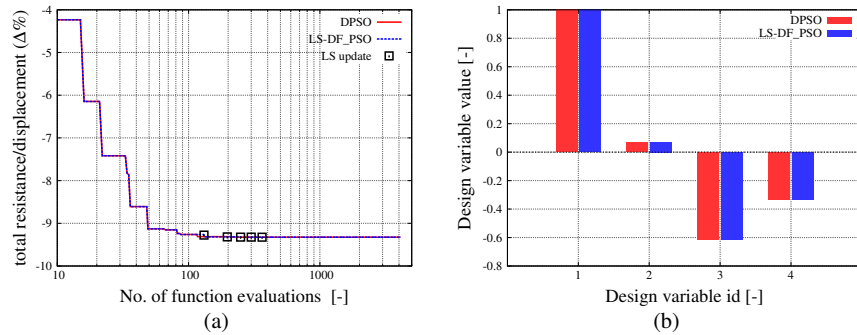
**Fig. 2** Rosenbrock function minimization: convergence of the objective function (a) and global-optimum location history (b)

The shape optimization of the Delft catamaran is shown as an example of industrial design problems. The Delft catamaran is a concept ship used for experimental and numerical benchmarks (see, e.g., the numerical studies presented by Diez et al. in 2013 [10]). Figure 3 shows the Delft catamaran model during towing tank experiments at CNR-INSEAN, along with a detail of the original hull shape. The design optimization problems are taken from the work by Chen et al. in 2014 [3] and Diez et al. in 2013 [9] respectively, and solved by means of stochastic radial-basis functions interpolation (details may be found in the work by Volpi et al. in 2014 [32]) of high-fidelity URANS simulations. For the first problem, the design objective is the minimization of the resistance in calm water at Froude number equal to 0.5 (see Chen et al. 2014 [3] for details). For the second problem, the design objective is



**Fig. 3** Delft catamaran: towing tank experiments at CNR-INSEAN (a) and detail of the original geometry (b)

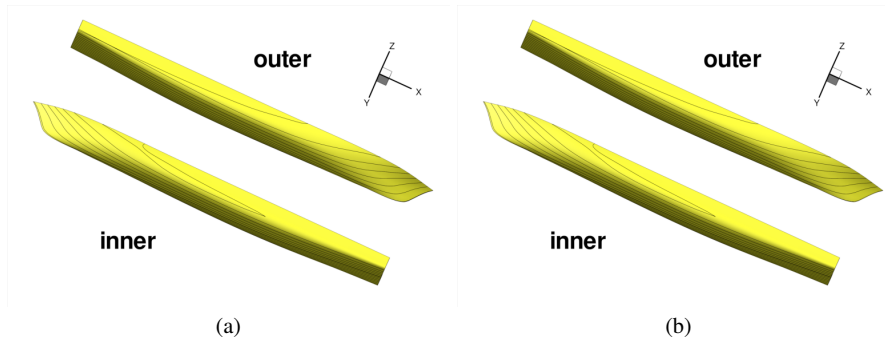
the reduction of the expected value of the mean resistance in head waves, taking into account stochastic sea state in the North Pacific ocean and variable speed (see Diez et al. 2013 [9] for details). For both problems, four design variables control global shape modifications, based on the Karhunen-Loève expansion of the shape modification vector (Diez et al. 2013 [10]).



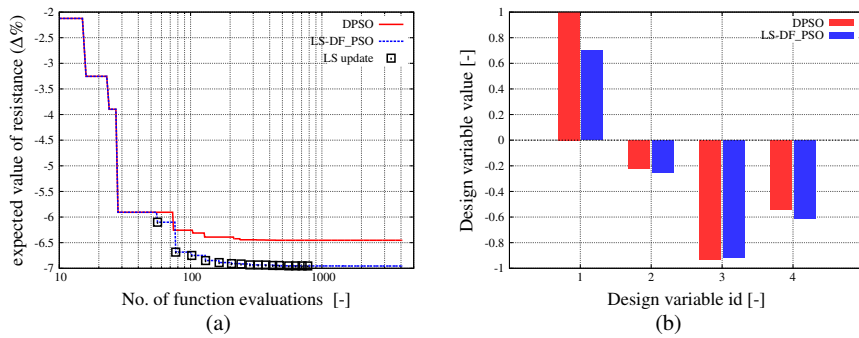
**Fig. 4** Minimization of calm-water resistance for the Delft catamaran: convergence of the objective function (a) and global-optimum design variables (b)

Figure 4 (a) shows the convergence of the minimization procedure for the first problem, comparing PSO with LS-DF\_PSO. LS is required only few times, as indicated by the black squares, and the minima provided by the two algorithms are extremely close, as shown in Figure 4 (b) and Figure 5. Nevertheless, it may be noted that, at very reduced additional cost, LS-DF\_PSO provides a solution certified with stationarity properties.

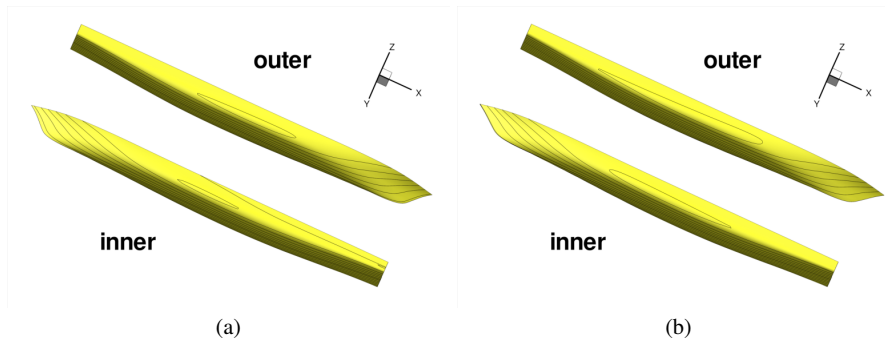
Figure 6 (a) shows the convergence of the minimization procedure for the second problem, comparing PSO with LS-DF\_PSO. LS is required and applied a larger number of times than in the previous problem, and is essential to identify the global



**Fig. 5** Minimization of calm-water resistance for the Delft catamaran: optimal shape design by PSO (a) and LS-DF\_PSO (b)



**Fig. 6** Minimization of expected value of mean resistance in head waves for the Delft catamaran: convergence of the objective function (a) and global-optimum design variables (b)



**Fig. 7** Minimization of expected value of mean resistance in head waves for the Delft catamaran: optimal shape design by PSO (a) and LS-DF\_PSO (b)

optimum, as shown in Figure 6 (b). As a results, optimal shape design provided by PSO and LS-DF\_PSO are noticeably different (within the context of current ap-

plication's variation), as shown in Figure 7. Additionally, it may be noted that the solution given by LS-DF\_PSO is also endowed with stationarity properties.

## 8 Conclusions

In this chapter we have detailed some globally convergent modifications of PSO iteration (2), for the solution of the unconstrained global optimization problem (1). Under mild assumptions, Proposition 4 proved that at least a subsequence of the iterates generated by our modified PSO, namely Algorithm LS-DF\_PSO, converges to a stationary point, which is possibly a minimum point. We recall that using the standard PSO iteration, by no means we can guarantee convergence towards stationary points, unless we consider trivial cases of no practical interest. Thus, our result reinforces the theoretical properties of modified PSO schemes. To the best of our knowledge, our result is also among the first attempts to couple PSO with line search-based derivative-free schemes (see also Vaz and Vicente 2007 [30], Vaz and Vicente 2009 [31] for extensions to trust-region derivative-free approaches), where a modified PSO scheme is proved to satisfy conditions like (20) or (21).

On the basis of our experience, which seems confirmed by the results reported here, we are persuaded that a fruitful coupling of PSO with an iterative globally convergent derivative-free method, should yield a compromise, between the fast progress of PSO (global search) in the early iterations, and the capability to exploit (local search) the objective function.

We also have reported numerical experiences on a significant test function and two ship design problems, which confirm that LS-DF\_PSO is more effective (and to a great extent equally efficient) than PSO. Indeed, LS-DF\_PSO is able to achieve better solutions/designs, and provides stationarity properties at the associated optimal points.

**Acknowledgements** The work of M. Diez is supported by the US Navy Office of Naval Research, NICOP Grant N62909-11-1-7011, under the administration of Dr. Ki-Han Kim and Dr. Woei-Min Lin. The work of A. Serani, E.F. Campana and G. Fasano is supported by the Italian Flagship Project RITMARE, coordinated by the Italian National Research Council and funded by the Italian Ministry of Education, within the National Research Program 2011-2013.

## References

1. Campana, E.F., Fasano, G., Peri, D.: Globally convergent modifications of Particle Swarm Optimization for Unconstrained Optimization. In Particle Swarm Optimization: Theory, Techniques and Applications, Nova Publishers, Series: Advances in Engineering Mechanics, Ed. Bohua Sun (2011)
2. Campana, E.F., Fasano, G., Pinto, A.: Dynamic system analysis for the selection of parameters and initial population, in Particle Swarm Optimization. Journal of Global Optimization, **48(3)**, 347–397 (2010)

3. Chen, X., Diez, M., Kandasamy, M., Zhang, Z., Campana, E.F., Stern, F.: High-fidelity global optimization of shape design by dimensionality reduction, metamodels and deterministic particle swarm. *Engineering Optimization*, in press, DOI: 10.1080/0305215X.2014.895340 (2014)
4. Christopher, W., Cleghorn, C., Engelbrecht, A.: A generalized theoretical deterministic particle swarm model. *Swarm Intelligence Journal*, **8**(1), 35–59 (2014)
5. Clerc, M., Kennedy, J.: The Particle Swarm - Explosion, Stability, and Convergence in a Multidimensional Complex Space. *IEEE Transactions on Evolutionary Computation*, **6**(1), 58–73 (2002)
6. Clerc, M.: Stagnation analysis in particle swarm optimization or what happens when nothing happens. <http://hal.archives-ouvertes.fr/hal-00122031>, Tech. Rep. (2006)
7. Conn, A.R., Scheinberg, K., Vicente, L.N.: *Introduction to Derivative-Free Optimization*. MPS-SIAM Series on Optimization, Philadelphia (2009)
8. Corazza, M., Fasano, G., Gusso, R.: Particle Swarm Optimization with non-smooth penalty reformulation for a complex portfolio selection problem. *Applied Mathematics and Computation*, **224**, 611–624 (2013)
9. Diez, M., Chen, X., Campana, E.F., Stern, F.: Reliability-based robust design optimization for ships in real ocean environment. *Proceedings of 12th International Conference on Fast Sea Transportation, FAST2013, Amsterdam, The Netherlands* (2013)
10. Diez, M., He, W., Campana, E.F., Stern, F.: Uncertainty quantification of Delft catamaran resistance, sinkage and trim for variable Froude number and geometry using metamodels, quadrature and Karhunen-Loève expansion, *Journal of Marine Science and Technology*, **19**(2), 143–169, DOI: 10.1007/s00773-013-0235-0 (2014)
11. Fasano, G., Lucidi, S.: A nonmonotone truncated Newton-Krylov method exploiting negative curvature directions, for large scale unconstrained optimization. *Optimization Letters*, **3**(4), 521–535 (2009)
12. Gazi, V.: Stochastic stability analysis of the particle dynamics in the PSO algorithm. In: *Proceedings of the IEEE International Symposium on Intelligent Control, Dubrovnik, Croatia*, IEEE Press 708–713 (2012)
13. Griewank, A.: *Evaluating Derivatives*. SIAM Frontiers in applied mathematics, Philadelphia (2000)
14. Hart, W.E.: A stationary point convergence theory for evolutionary algorithms. *Foundations of Genetic Algorithms 4*, Morgan Kaufmann, San Francisco, CA, pp. 325–342 (1996)
15. Hart, W.E.: *Evolutionary Pattern Search Algorithms for Unconstrained and Linearly Constrained Optimization*. In *Proc. Evolutionary Programming VII*, Springer-Berlin, pp. 303–312 (2001)
16. Kadirkamanathan, V., Selvarajah, K., Fleming, P.: Stability analysis of the particle dynamics in particle swarm optimizer. *IEEE Transactions on Evolutionary Computation*, **10**(3), 245–255 (2006)
17. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. *Proceedings of the 1995 IEEE International Conference on Neural Networks (Perth, Australia)*, IEEE Service Center, Piscataway, NJ, IV: pp. 1942–1948 (1995)
18. Kolda, T.G., Lewis, R.M., Torczon, V.: Optimization by Direct Search: New Perspectives on Some Classical and Modern Methods. *SIAM Review*, **45**(3), 385–482 (2003)
19. Lucidi, S., Sciandrone, M.: On the global convergence of derivative-free methods for unconstrained optimization. *SIAM Journal of Optimization*, **13**, 97–116 (2002)
20. Mendes, R.: *Population Topologies and Their Influence in Particle Swarm Performance*, PhD Dissertation, University of Minho, Departamento de Informatica Escola de Engenharia Universidade do Minho (2004)
21. Nocedal, J., Wright, S.J.: *Numerical Optimization*, Second Edition. Springer Series in Operations Research and Financial Engineering, New York (2006)
22. Peri, D., Diez, M.: Ship Optimization by globally convergent modification of PSO using surrogate-based Newton method. *Engineering Computations*, **30**(4), 548–561, DOI: 10.1108/02644401311329361 (2013)

23. Poli, R., Kennedy, J., Blackwell, T.: Particle swarm optimization. *Swarm Intelligence*, **1**(1), 33–57 (2007)
24. Poli, R.: Analysis of the Publications on the Applications of Particle Swarm Optimisation. *Journal of Artificial Evolution and Applications*, **article ID 685175**, 1–10 (2008)
25. Poli, R.: Mean and variance of the sampling distribution of particle swarm optimizers during stagnation. *IEEE Transactions on Evolutionary Computation* **13**(4), 712–721 (2009)
26. Sarachik, P.E.: Principles of linear systems. Cambridge University Press, New York (1997)
27. Serani, A., Diez, M., Leotardi, C., Peri, D., Fasano, G., Iemma, U., Campana, E.F.: On the use of synchronous and asynchronous single-objective deterministic Particle Swarm Optimization in ship design problems. *Proceeding of OPT-i, International Conference on Engineering and Applied Sciences Optimization*, Kos Island, Greece (2014)
28. Trelea, I.C.: The Particle Swarm Optimization Algorithm: Convergence Analysis and Parameter Selection. *Information Processing Letters*, **85**, 317–325 (2003)
29. Van den Bergh, F., Engelbrecht, A.P.: A Study of Particle Swarm Optimization Particle Trajectories. *Information Sciences*, **176**, 937–971 (2006)
30. Vaz, A.I.F., Vicente, L.N.: A particle swarm pattern search method for bound constrained global optimization. *Journal of Global Optimization*, **39**, 197–219 (2007)
31. Vaz, A.I.F., Vicente, L.N.: PSwarm: A hybrid solver for linearly constrained global derivative-free optimization. *Optimization Methods and Software*, **24**, 669–685 (2009)
32. Volpi, S., Diez, M., Gaul, N.J., Song, H., Iemma, U., Choi, K.K., Campana, E.F., Stern, F.: Development and validation of a dynamic metamodel based on stochastic radial basis functions and uncertainty quantification, *Structural Multidisciplinary Optimization*, in press, DOI: 10.1007/s00158-014-1128-5 (2014)