Comparing the performances of four stochastic optimisation methods using analytic objective functions, 1D elastic full-waveform inversion and residual static computation

Angelo Sajeva*, Mattia Aleardi*, Bruno Galuzzi°, Eusebio Stucchi°, Emmanuel Spadavecchia[§], Alfredo Mazzotti*

*University of Pisa, Earth Sciences Department

°University of Milan, Earth Sciences Department

[§]ENI, Upstream and Technical Services Division

Corresponding Author: Angelo Sajeva (angelo.sajeva@for.unipi.it)

Abstract

We compare the performances of four different stochastic optimisation methods using four analytic objective functions and two highly non-linear geophysical optimisation problems: 1D elastic full-waveform inversion (FWI) and residual static computation. The four methods we consider, namely, adaptive simulated annealing (ASA), genetic algorithm (GA), neighbourhood algorithm (NA), and particle swarm optimisation (PSO), are frequently employed for solving geophysical inverse problems. Because geophysical optimisations typically involve many unknown model parameters, we are particularly interested in comparing the performances of these stochastic methods as the number of unknown parameters increases. The four analytic functions we choose simulate common types of objective functions encountered in solving geophysical optimisations: a convex function, two multi-minima functions that differ in the distribution of minima, and a nearly flat function. Similar to the analytic tests, the two seismic optimisation problems we analyse are characterized by very different objective functions. The first problem is a 1D elastic FWI, which is strongly ill-conditioned and exhibits a nearly flat objective function, with a valley of minima extended along the density direction. The second problem is the residual static computation, which

is characterized by a multi-minima objective function produced by the so-called cycle-skipping phenomenon. According to the tests on the analytic functions and on the seismic data, GA generally displays the best scaling with the number of parameters. It encounters problems only in the case of irregular distribution of minima, that is, when the global minimum is at the border of the search space and a number of important local minima are distant from the global minimum. The ASA method is often the best-performing method for low-dimensional model spaces, but its performance worsens as the number of unknowns increases. The PSO is effective in finding the global minimum in the case of low-dimensional model spaces with few local minima or in the case of a narrow flat valley. Finally, the NA method is competitive with the other methods only for low-dimensional model spaces; its performance stability sensibly worsens in the case of multi-minima objective functions.

Introduction

Geophysical optimisation problems are usually non-linear, multi-dimensional and characterized by complex objective functions with many local minima. There are essentially two strategies to tackle these problems: apply deterministic methods that involve the computation of the gradient of the objective function (e.g., the Gauss-Newton, conjugate-gradient and steepest-descent methods), or apply stochastic Monte Carlo (MC) methods that perform a direct search in the model space. Deterministic methods are attractive because they are the natural extension of linear methods (deterministic methods iteratively solve a linearized version of the problem) and because they usually converge rapidly. On the other hand, they need a starting model of sufficient quality in order to find the global minimum. Moreover, they require the computation of the first- and/or second-order derivatives of the objective function, a process that can be computationally demanding in the case of high-dimensional model spaces. Summarizing, deterministic methods are very poor at exploring the model space (that is, in finding the most-promising valleys of the objective function) and very good at exploitation (given the valley, finding its minimum). Differently, stochastic MC

methods do not require any calculation of the derivatives of the objective function, and compared to deterministic approaches, they are very good at exploration but are not as good at exploitation. These characteristics make their application particularly suitable when it is difficult to identify the valley in the model space that contains the global minimum or when the computation of the derivatives of the objective function is particularly expensive. The downside of the MC methods is that they usually need many model evaluations to converge, and this has prevented their application in inversions with many unknown model parameters and/or requiring expensive forward modelling.

MC methods can be applied to a wide range of problems in physics, chemistry, economics, biology, and mathematics. In geophysics, MC methods have proved to be a powerful tool to solve optimisation problems in different geophysical fields, including electromagnetic induction (Anderssen, 1970), Rayleigh wave attenuation (Mills and Fitch, 1977), regional magnetotelluric studies (Hermance and Grillot, 1974), estimation of plate rotation vectors (Jestin et al., 1994), and exploration and applied geophysics (Sen and Stoffa, 2013). The first applications of MC methods to geophysics were proposed in the 1960s (Keilis-Borok and Yanovskaya, 1967; Backus and Gilbert, 1968), whereas the first application to seismology was by Press (1968), who inverted travel times of body waves and 97 eigenperiods of the Earth's free oscillations for variations in the Earth's compressional wave velocity, shear wave velocity, and density as a function of depth. Since then, MC methods have been applied to solve an increasing number of geophysical problems.

An appealing property of MC methods is that they do not need the assumption of linearity of the inverse problem. This is convenient for geophysical problems, which are often non-linear, ranging from weakly non-linear, such as predicting earthquake hypocentre locations using the travel times of seismic events (Buland, 1976), to highly non-linear, such as the estimation of seismic velocities from high-frequency seismic waveforms (Mellman, 1980). MC sampling techniques can also be used to determine resolution estimates without computing derivative approximations (Wiggins, 1972). In fact, MC methods can address the non-uniqueness of the solution by solving for a region of acceptable models in the parameter space (Keilis-Borok and Yanovskaya, 1967).

In the ’70s, linear inversion techniques gained popularity at the expense of uniform MC, which was thought to be too inefficient and too inaccurate for problems involving a large number of unknowns. This is because the computational cost of uniform MC methods increases exponentially with the number of unknowns (a problem sometimes referred to as the “curse of dimensionality”). Nevertheless, the reliability of these linearized methods is limited to weakly non-linear problems if no guess of the solution exists, and it can only be extended to highly non-linear problems if a good enough guess of the solution is provided in advance.

The need for more powerful optimisation methods and the increasing computational power provided by modern computers have encouraged the creation and development of several partially stochastic heuristic methods, many of which have been successfully tested in the geophysical field. One of these is the simulated-annealing algorithm (SA; Kirkpatrick et al., 1983). This method models the physical process of heating a material and then slowly lowering the temperature to decrease defects, thus minimizing the system energy. Rothman (1985; 1986) first introduced the technique of simulated annealing into applied geophysics to solve surface-consistent residual statics computations. Many other applications of SA to geophysics were proposed in the following years, including Landa et al. (1989), Sen and Stoffa (1991), Ryden and Park (2006), and Pei et al. (2007), to name just a few. Another popular class of methods is constituted by the genetic algorithms (GA; Holland, 1975). They mimic the natural evolution of species by optimizing the fitness of a set of models through the operations of selection, recombination, and mutation. They were first used in geophysics in the early 1990s, when a number of papers addressed this subject in quick succession (Stoffa and Sen, 1991; Sen and Stoffa, 1992; Scales et al., 1992; Nolte and Frazer, 1994), largely in the area of seismic waveform fitting. Many other applications of GA to geophysics were proposed in the following years (Mallick, 1999; Mallick and Dutta, 2002; Padhi and Mallick, 2014; Li and Mallick 2015; Aleardi 2015; Aleardi and Mazzotti, 2017; Sajeva et al. 2016a; Aleardi et al. 2016; Sajeva et al. 2016b; Aleardi and Ciabarri, 2017). Other noteworthy MC methods that have been applied to solve geophysical optimisation problems are the neighbourhood algorithm (NA;

Sambridge 1999a), and particle swarm optimisation (PSO; Kennedy and Eberhart, 1995). A key foundation of NA is that it makes use of all the information from the ensemble of previously evaluated models by assuming that the best models are located in the neighbourhood of the previously evaluated good models. Geophysical applications of NA can be found in Wathelet et al. (2004), Marson-Pidgeon et al. (2000), and Fliedner et al. (2012). Finally, PSO is inspired by the social behaviour of bird flocking or fish schooling. Several variants of this algorithm exist, such as that of Clerc (1999). Some recent geophysical applications of PSO can be found in Shaw and Srivastava (2007), Fernández Martínez et al. (2010), Fernández Martínez et al. (2012), and Lagos et al. (2014).

It is of interest to compare different popular stochastic MC methods to determine their effectiveness in different scenarios. The reader can find comparisons of MC methods in Ingber and Rosen (1992), Horne and MacBeth (1998), Hassan et al. (2005), Shaw and Srivastava (2007), and Sambridge (1999b). In general, the results discussed in these papers cannot be extended to draw general conclusions. This is because the result of any single comparison depends on the complexity of the problem to be solved (nonlinearity, non-uniqueness, irregularity, dimensionality), as well as the particular implementation of the algorithm (for instance, several versions of GA and SA exist) and the choice of control parameter values.

Aware of these unavoidable limitations, we present a comparison of four MC methods frequently used as optimization tools in geophysical inversion problems. In particular, we test: adaptive simulated annealing (ASA), a method which pertains to the SA class; an algorithm representative of GA; an implementation of PSO, and finally NA. We apply each method to four very different analytic functions: a convex parabolic-shaped function (the De Jong function n°1), a multi-minima regular surface (the Rastrigin function), an irregular surface with many local minima (the Schwefel function), and a flat banana-shaped valley (the Rosenbrock function). These functions simulate a wide range of possible objective functions found in geophysical inverse problems, and for each of them, we analyse the pros and cons of the different MC optimization methods. In addition, we test

each function for dimensions of the model space ranging from 2 to 60. This permits us to estimate the rate of convergence as the number of unknowns increases, which is especially of interest in geophysical problems, which usually involve many unknowns. More specifically, the first test, on the simple convex function, is aimed at evaluating the rate of convergence in the case of a single minimum. The second and third tests, on multi-minima functions, help us verify the ability of each method to efficiently explore the model space and to escape local minima. The fourth test, on a convex function with a global minimum inside a long, narrow valley, allows us to compare the rate of convergence of each method when the objective function is nearly flat. Finally, we apply these four methods to two classic geophysical problems: 1D full waveform inversion (FWI) and residual statics estimation. As for the analytic functions, we approach the two problems by progressively increasing the number of unknowns.

To the best of our knowledge, a detailed comparison of the performances of these algorithms in different geophysical optimisations as the number of unknowns increases has not been performed. This lack is even more serious because applications of MC methods are becoming increasingly common in the geophysical community, thanks to the growing computational power that has progressively extended the field of applicability of these algorithms. We believe that this topic is worth a deeper investigation in order to increase the awareness on the limits, strengths and differences that characterize MC methods and to guide the choice of the most appropriate algorithm to tackle the problem at hand.

A brief introduction to the theoretical aspects of the four stochastic methods

In this section, we give a theoretical overview of the four stochastic methods we use in this work. References are given for the reader interested in the theoretical details of each method. We point out that the performances of a stochastic method in solving a particular problem may critically depend on the choice of the control parameters. Generally, it is difficult to give hard and fast rules that may work with a wide range of applications, although some guidelines and rules of thumb can be

dictated by experience. The control parameters used in the tests discussed in this paper have been determined from a trial and error procedure with the aim of balancing the probability and rapidity of convergence with the goodness of the final solution.

Adaptive simulated annealing

The simulated annealing (SA) method is an adaptation of the Metropolis-Hastings algorithm (Hastings, 1970). It is a stochastic optimisation method that mimics the metallurgical annealing process by using the concepts of "cooling" and "heating" (Kirkpatrick et al. 1983). The SA algorithm creates a sequence of models $\{x_k\}_{k>0}$, where k is the index of the iteration of the algorithm. For each iteration k, a new candidate model y is generated in a neighbourhood of the current model x_k, using the generation formula

$$y = x_k + \Delta x_k(T_g, p), \quad (1)$$

where Δx_k is the step size, which depends on a parameter T_g, called the generation temperature, and p is a random number uniformly distributed over $[0,1]$. The step size is proportional to T_g.

Once the candidate model y is created, the algorithm chooses whether the model is accepted or not. If the candidate is rejected, the algorithm proceeds from the current model, $x_{k+1} = x_k$, otherwise the candidate model becomes the current model, $x_{k+1} = y$. The new model is always accepted if the new value of the objective function is lower, whereas if this value is higher, the model is accepted with a probability dependent on a parameter called the acceptation temperature T_a. We can express this procedure with the following formula:

$$x_{k+1} = \begin{cases} y, & p \leq \min(1, e^{-\frac{f(y)-f(x_k)}{T_a}}), \\ x_k, & \text{otherwise} \end{cases} \quad (2)$$

where p is a random number uniformly distributed over $[0,1]$ and f is the objective function to be minimized. In the above formula, T_a depends on k, but it is not shown for readability. The process of randomly accepting models with higher objective function values permits to escape from local minima.

At the early stages of the optimisation, we set the initial generation and acceptation temperatures (T_{g0} , T_{a0}) to high values to allow a wide exploration of the model space. Subsequently, during the optimisation, their values are progressively reduced to the final values (T_{gf} , T_{af}) to focus the search on the most promising zones of the model space. As suggested by Aguiare et al. (2012), it is a good practice to increase the initial generation temperature and to reduce the cooling rate as the dimension of the model space increases.

Ingber (1989) introduced complex modifications to the original SA algorithm to reduce the computational cost and to decrease the probability of entrapment in local minima. In the current work, we use the Ingber variation of the SA algorithm, called adaptive simulated annealing (ASA). ASA is distinguished from standard SA by its use of different generation formulas for each *i-th* direction of the model space:

$$y^i = x_k^i + \Delta x_k^i\left(T_g^i, p^i\right), \qquad i = 1, \dots, n \tag{3}$$

where n is the dimension of the model space, T_g^i are the generation temperatures, Δx_k^i are the step sizes and p^i are random numbers uniformly distributed over [0,1]. In addition, in the ASA algorithm, the generation and acceptation temperatures, $T_g^{i,k}$ and T_a^k, are characterized by an exponential decrease with the number of iterations k and with the number of accepted models k_a, respectively:

$$\begin{cases} T_g^{i,k+1} = T_{g0}^{i,k} e^{-c_i \sqrt[n]{k}} \\ T_a^{k+1} = T_{a0}^{k} e^{-c_a \sqrt[n]{k_a}} \end{cases}, \tag{4}$$

where c_a and c_i are scalability factors. Both $T_g^{i,k}$ and T_a^k, after a given number of iterations n_{iter}, reach their final values and remain constant. Finally, ASA allows for multiple cooling/heating cycles (reannealing of both generation and acceptation temperatures) to reduce the possibility of becoming trapped in a local minimum. These modifications make the ASA less sensitive to user-defined parameters and more efficient than the classical SA method (Ingber, 1989).

Genetic algorithms

Genetic algorithms are a class of search algorithms developed by Holland (1975) that belong to the larger class of evolutionary algorithms. They use the mechanics of natural selection and evolution (the Darwinian principle "survival of the fittest") to search through the model space for optimal solutions. The optimisation process starts with randomly generated individuals, each one encoding a candidate solution, and the entire population of individuals evolves toward better solutions by means of three principal operators: mutation, cross-over, and selection. A complete mathematical description of this stochastic method would require too much space and tens of equations in order to be useful and understandable for the reader. For this reason, we opt to describe the GA optimization procedure with a diagram. Figure 1 gives an outline of how the algorithm works: at each generation, the fitness, namely, the goodness, of each individual is evaluated. Then, some individuals (parents) are stochastically selected from the current population on the basis of their fitness. Next, they are modified (using cross-over and mutation operators) to form a set of offspring that are used to replace the least-fit parents and to form the new population to be used in the next generation. The algorithm terminates when the optimization criteria are met, that is, when either a maximum number of generations has been produced or a satisfactory fitness level has been reached.

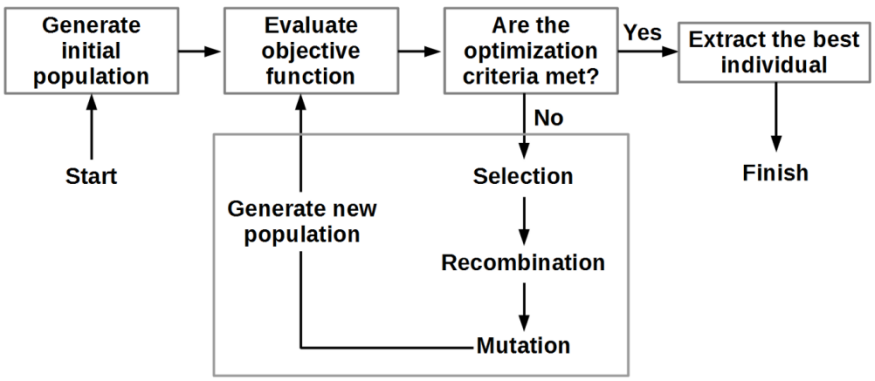

Figure 1: Diagram of a Genetic Algorithm optimization procedure.

There are many different GA implementations, and the one considered in this work is briefly described as follows. We use a real-coded GA implementation (Janikow and Michalewicz, 1991) in which the candidate solution is expressed by floating-point numbers instead of binary values. The selection probability associated with each individual is computed by applying a linear rank-based fitness assignment (Bäck and Hoffmeister, 1991), whereas the stochastic universal sampling method is used for selection. In the selection process, the percentage of individuals to be selected from the current population is given by the selection-rate parameter. The extended intermediate recombination method (Schlierkamp-Voosen and Mühlenbein H. 1993) is used in the cross-over step, which aims to produce new individuals by combining the information brought by two or more parents. In the mutation step, we use the mutation strategy proposed by Schlierkamp-Voosen and Mühlenbein H. (1993) in their "Breeder Genetic Algorithm". In this step, the probability of mutating a given variable is controlled by the mutation rate parameter. Schlierkamp-Voosen and Mühlenbein (1993) demonstrated that the best choice for the mutation rate is the reciprocal of the number of unknowns. Thus, on average, only one variable for each individual is mutated per generation. Finally, to derive the new population, we use an elitist strategy that preserves the fittest individuals of the previous generation in the new generation, combined with a fitness-based reinsertion in which the highest-fitness offspring replace the lowest-fitness parents.

More detailed information about the GA method and its control parameters can be found in Goldberg (1989), Mitchell (1998), and Sivanandam and Deepa (2008). Moreover, an extensive discussion about selection methods that can be used in a GA optimisation can be found in Blickle and Thiele (1995).

The neighbourhood algorithm

The neighbourhood algorithm (NA) is a direct-search method based on the concept of proximity. The method is composed of two parts, which are two different stages of the optimisation process:

sampling the model space (Sambridge, 1999a) and appraising the ensemble (Sambridge, 1999b). In this paper, we will employ only the part of the algorithm that involves the sampling of the model space, and we will refer to it as NA, for simplicity. The sampling part is concerned with finding models of acceptable data fit in a multi-dimensional parameter space. One of the key ideas of this algorithm is that the search is guided by the information brought by all the previous models for which the objective function has already been evaluated. To this end, the ensemble of previous models is used to approximate the objective function everywhere in the model space by employing Voronoi cells (Voronoi, 1908). We recall that given a set of points in a multi-dimensional space, Voronoi cells are simply the nearest neighbour regions about each point. Inside each Voronoi cell, the objective function is assumed to be flat and equal to the objective function of the model (also called "the seed") associated with the cell. This geometric construction is the core of the algorithm, as it enables creation of the approximated objective function surface.

The algorithm can be summarized in the following four steps:

1) Generate an initial set of n_p models uniformly (or otherwise) distributed in the parameter space;

2) Calculate the misfit function for the most-recently generated set of n_p models and determine the n_r models with the lowest misfit of all models generated so far;

3) Generate n_p new models by performing a uniform random walk in the Voronoi cell of each of the n_r chosen models (i.e., n_p / n_r samples in each cell);

4) Proceed to step 2.

Given an n-dimensional model space, it is natural to increase both n_p and n_r with n in order to grant an adequate exploration of the search space. Moreover, it is important to fix the ratio of the two parameters n_r/n_p, i.e., the ratio of the selected Voronoi cells over the newly created Voronoi cells, as it is a measure of the ratio of the exploitation of the optimisation process over the exploration. In addition, low n_r are optimal for convex objective functions or functions with a small number of local minima, whereas n_r values similar to n_p guarantee a more thorough exploration in

the case of complex functions with many local minima. From the above analysis, it is clear that this algorithm is conceptually simple, as it only requires two control parameters: the number of new models to be generated per iteration, n_p, and the number of best-fitting models to be selected among these new models per iteration n_r.

Particle swarm optimisation

Particle swarm optimisation (PSO) is inspired by the social behaviour of bird flocks (Kennedy at al., 2001). It uses a swarm of particles $\{x_i\}_{i=1}^{n}$ that moves in the model space with a specific speed $\{v_i\}_{i=1}^{n}$. The magnitude and direction of the velocities of the particles depend on three factors: the current velocity $\{v_{i0}\}_{i=1}^{n}$ of each particle (inertia), the current best model found by each particle $\{g_i\}_{i=1}^{n}$ (cognition), and the current global best model G shared by all the particles (sociality).

Initially, the PSO algorithm chooses candidate solutions randomly within the search space. Then, at each iteration k, the velocities (v_i) and positions (x_i) of the particles are updated according to the following formulas:

$v_i(k+1) = \omega * v_i(k) + a_1 * a_{loc} * \big(g_i - x_i(k)\big) + a_2 * a_{glob} * \big(G - x_i(k)\big),$	(5)
$x_i(k+1) = x_i(k) + v_i(k+1),$	(6)

where ω is the inertia, a_{loc} and a_{glob} are the local and global accelerations, respectively, and a_1 and a_2 are random numbers between 0 and 1. Originally, the PSO algorithm was used with $(\omega, a_{loc}, a_{glob}) = (1, 2, 2)$ (Kennedy and Eberhart, 1995), but this formulation can cause the particles to oscillate around their centre, preventing the system to achieve full convergence. To ensure convergence, Clerc (1999) proposed a variant of PSO based on the following formulas:

$v_i(k+1) = \phi * \Big(v_i(k) + a_1 * a_{loc} * \big(g_i - x_i(k)\big) + a_2 * a_{glob} * \big(G - x_i(k)\big)\Big),$	(7)
$x_i(k+1) = x_i(k) + v_i(k+1),$	(8)

where the function ϕ is a constraint function that depends on the two accelerations:

$$\phi(a_{loc}, a_{glob}) = \frac{2}{|2 - a_{loc} - a_{glob} - \sqrt{(a_{loc} + a_{glob})^2 - 4 * (a_{loc} + a_{glob})}|}. \quad (9)$$

These velocity and position updating steps determine the optimisation ability of the PSO algorithm. It is advisable to set a maximum velocity v_{max} for all the particles in order to limit the displacements in the search space and to set $a_{loc} + a_{glob} \geq 4$ (Clerc and Kennedy, 2002). Many versions of PSO exist, such as the Fernández Martínez et al. (2010; 2012) family of PSO optimizers. Among them, we have chosen the Clerc version, which is a popular version of PSO that has been implemented by many authors (e.g. Robinson and Rahmat-Samii, 2004; Shaw and Srivastava, 2007).

Tests on the analytic objective functions

We evaluate the performance of ASA, GA, NA, and PSO using four analytic test functions that exhibit very different characteristics. The test functions are the De Jong function n°1, which is convex; the Rastrigin function, which has a large number of regularly distributed local minima; the Schwefel function, which has the global minimum at the border of the search space and several irregularly distributed local minima; and the Rosenbrock function, in which the minimum is in an elongated flat valley. Each of these functions can be defined for an integer dimension *n* of the model space, except the Rosenbrock function, which is defined only for even dimensions. We use a rescaled version of each objective function such that we can consistently use the same search interval for the model parameters, that is, the hyper-cube $[-5,5]^n$. For simplicity, we use the same convergence criterion for all the algorithms in all the analytic functions. In particular, an algorithm converges when it finds a model $\boldsymbol{x} \in [-5,5]^n$ that satisfies the following accuracy criterion:

$$\sqrt{\frac{\sum_{i=1}^{n}(x_i - x_i^{glob})^2}{n}} < \epsilon \quad (10)$$

where $\boldsymbol{x}^{glob}$ is the global minimum and the accuracy ϵ is set to 0.05.

In this section, we aim to evaluate both the ability of the different algorithms to approach the global minimum and their rate of convergence, that is, the number of model evaluations needed to satisfy the convergence criterion. To produce statistically significant results, we perform 100 tests for each algorithm, for each test function, and for each dimension of the model space. From the subset of tests where the algorithm has reached the global minimum (within the predefined accuracy threshold), we derive the mean value and the standard deviation of the number of evaluated models. These values give us an estimate of the rate of convergence and its variability as a function of the model-space dimension. At the same time, the number of tests where convergence has been achieved is used to compute the probability of convergence.

Even if the convergence is always granted for a simple convex function, such as the De Jong function n°1 (after a sufficiently large number of evaluated models), this is not always the case for the Rastrigin or Schwefel functions, which exhibit a large number of local minima. When the entrapment in a local minimum is possible, measuring the probability of convergence as a function of the dimension of the model space is of particular interest. In this paper, we consider that a method is entrapped in a local minimum or fails to converge if the best-estimated model does not reach the global minimum within the selected accuracy after 10^7 evaluated models.

Setting the parameters of the algorithms

It is reasonable to set some control parameters of the algorithms according to the dimension of the model space and the complexity of the objective functions. The control parameters used for the tests on the four analytic objective functions are shown in Table 1, Table 2, Table 3, and Table 4.

ASA parameters	De Jong n°1	Rastrigin	Schwefel	Rosenbrock
T_{g0}	$10n$	$10n$	$10n$	$10n$
T_{gf}	$1^{-19}n$	$1^{-19}n$	$1^{-19}n$	$1^{-19}n$

n_{iter}	100*n*	100*n*	100*n*	100*n*
Reannealing	no	yes	yes	no

Table 1: Control parameters for the ASA method used in the tests on the analytic objective functions; n indicates the number of dimensions.

GA parameters	**De Jong n°1**	**Rastrigin**	**Schwefel**	**Rosenbrock**
Number of individuals	10*n*	10*n*	100*n*	10*n*
Selection rate	0.8	0.8	0.8	0.8
Mutation rate	n^{-1}	n^{-1}	n^{-1}	n^{-1}

Table 2: Control parameters for the GA method used in the tests on the analytic objective functions; n indicates the number of dimensions.

NA parameters	**De Jong n°1**	**Rastrigin**	**Schwefel**	**Rosenbrock**
n_p	10*n*	100*n*	100*n*	10*n*
n_r	2*n*	100*n*	100*n*	2*n*

Table 3: Control parameters for the NA method used in the tests on the analytic objective functions; n indicates the number of dimensions.

PSO parameters	**De Jong n°1**	**Rastrigin**	**Schwefel**	**Rosenbrock**
Number of particles	10*n*	100*n*	100*n*	10*n*
a_{glob}	1.2	1.2	1.2	1.2
a_{loc}	2.9	2.9	2.9	2.9
v_{max}	5	5	5	5

Table 4: Control parameters for the PSO method used in the tests on the analytic objective functions; n indicates the number of dimensions.

For ASA, it is good practice (Aguiare et al., 2012) to increase T_{g0}, T_{gf} and n_{iter} with the dimension of the model space *n*. We also use multiple cooling/heating cycles (reannealing) for multi-minima functions. For NA, it is advisable to increase n_p and n_r with *n* (Sambridge, 1999a). We increase the exploitation by setting a low $\frac{n_p}{n_r}$ value for convex functions, whereas in the case of multi-minima objective functions, we use a higher $\frac{n_p}{n_r}$ value to increase the exploration of the model space. For GA, we increase the size of the population with *n,* and we slow down the mutation rate linearly with n^{-1}. Finally, for PSO, we increase the size of the swarm linearly with *n* and set $v_{max} = 5$, a value that is half of the admissible range for each model parameter (Li-Ping et al., 2005). Moreover, we set $a_{loc} > a_{glob}$ in order to increase the exploration of the model space.

Tests on the De Jong function n°1

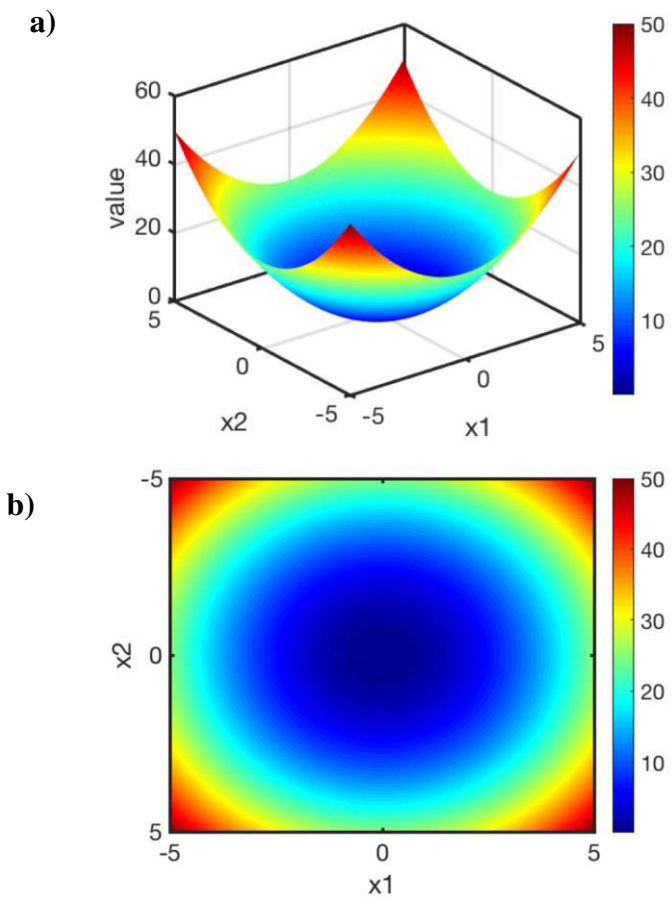

Figure 1: The De Jong function n°1 with n=2, represented a) as a surface in 3D space and b) as a 2D projection.

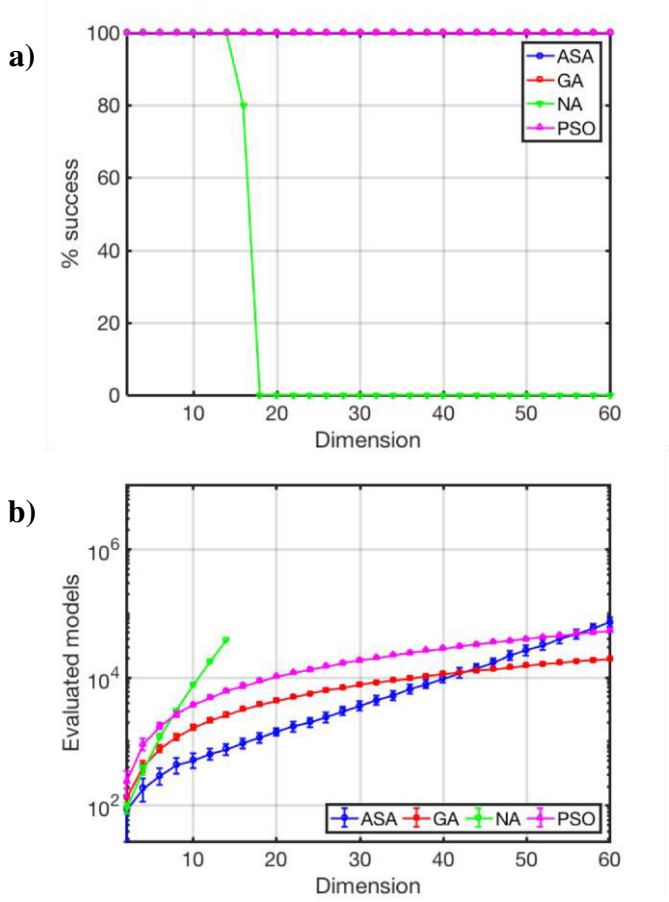

Figure 2: Summary of the results obtained on the De Jong function n°1. a) Percentage of successful tests (tests that have attained convergence within the chosen accuracy) as the dimension of the model space changes. b) Curves of convergence representing the mean number of evaluated models and the associated standard deviations needed to attain convergence within the chosen accuracy. These curves have been computed on the ensemble of 100 tests performed for each method and for each n. The curves highlight the different rates of convergence of the four algorithms.

In this first test, we compare ASA, GA, NA, and PSO on the De Jong function n°1:

$$f(x) = \sum_{i=1}^{n} x_i^2 \qquad (11)$$

This function is convex, symmetric, and unimodal, with a unique minimum in [0,…,0]. It is a very simple function, and we use it to evaluate the rate of convergence of the different algorithms in a favourable scenario in which there is a single minimum in the objective function.

Figure 1a and Figure 1b show this function in two dimensions (n=2). Figure 2a displays the percentage of success computed on the set of 100 tests for the four algorithms in a range of model-space dimensions from 2 to 60. ASA, GA, and PSO successfully converge for all the values of n; differently, NA converges until dimension 16 and quickly lowers its performance for higher dimensions, until it always fails to converge for dimensions higher than 20. Although the specific dimension for which NA fails to converge may depend on the choice of tuning parameters, we can still conclude that NA suffers the "curse of dimensionality" and that for high dimensions of the model space, it fails to converge even in the case of a convex objective function. This is because the sampling of the model space becomes increasingly sparse as the dimension of the model space increases, or, in other words, the approximation of the objective function surface given by the Voronoi cells becomes increasingly inaccurate. In fact, the number of Voronoi cells used to approximate the objective function is proportional to n_p, that is, it linearly increases with dimension n of the model space, while the "volume" of the model space exponentially increases with n.

In Figure 2b, we display the curves of convergence as the dimension n increases for the four methods. Note that NA and ASA exhibit an exponential trend (that is, a linear trend when using a semi-logarithmic plot), whereas GA and PSO are characterized by a polynomial trend (that is, a logarithmic trend when using a semi-logarithmic plot). In particular, NA displays good performances for low dimensions (n<8), but it shows the worst scalability as the dimension of the model space increases. We limit the analysis of NA to dimensions lower than 18 because of the large number of evaluated models required to attain the convergence criterion for higher values of n. For medium-low dimensions (2<n<40), ASA is the best-performing algorithm, while for dimensions higher than 44, GA outperforms the other methods. The PSO algorithm displays a polynomial trend similar to GA but with a number of evaluated models higher than GA. PSO

outperforms ASA for n>58. These results indicate that among the analysed algorithms, GA is the best-performing method for high-dimensional model spaces in the case of a convex objective function.

Tests on the Rastrigin function

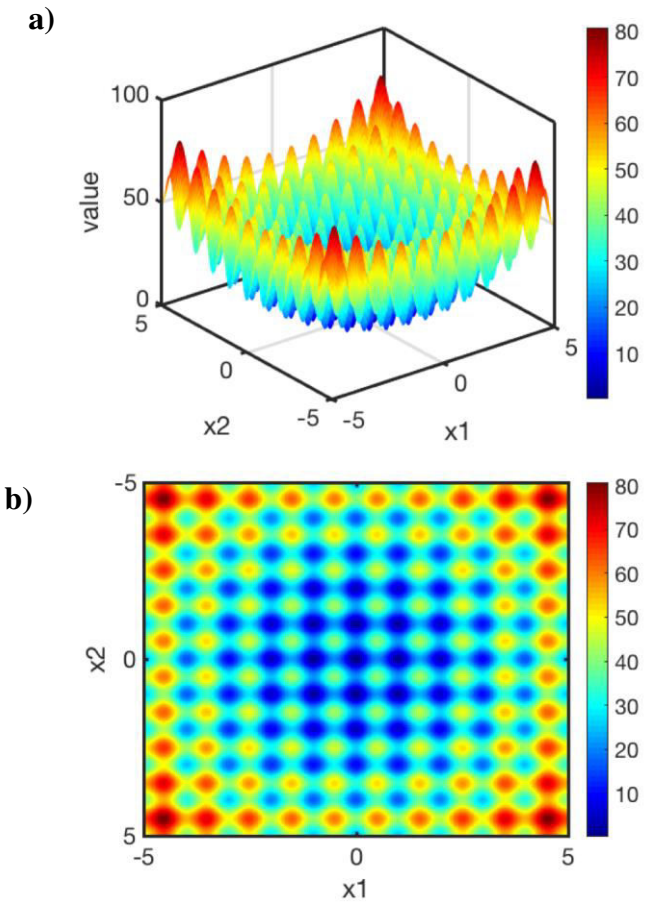

Figure 3: The Rastrigin function with n=2, represented a) as a surface in 3D space and b) as a 2D projection.

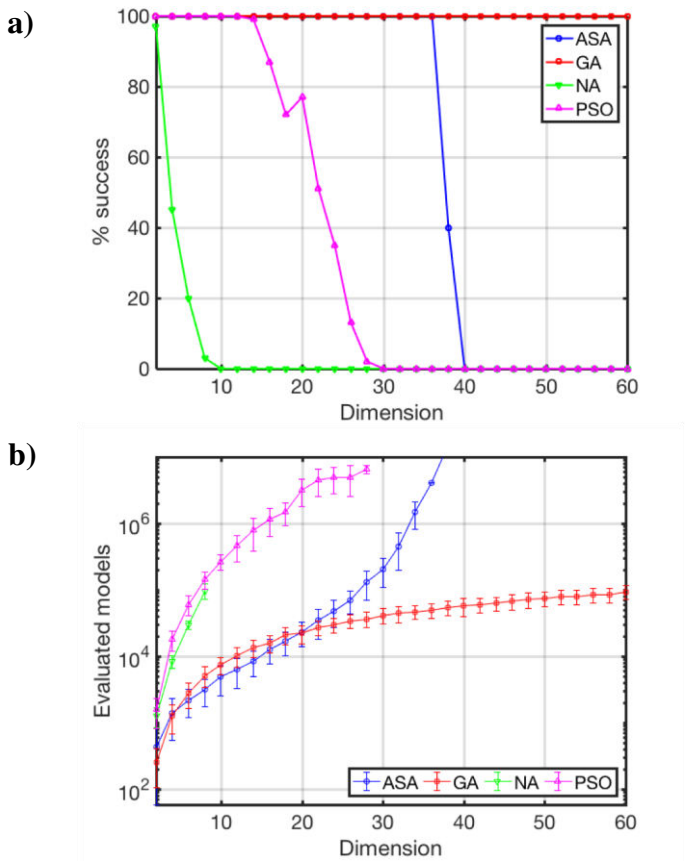

Figure 4: Summary of the results obtained on the Rastrigin function. a) Percentage of successful tests (tests that have attained convergence within the chosen accuracy) as the dimension of the model space changes. b) Curves of convergence representing the mean number of evaluated models and the associated standard deviations needed to attain convergence within the chosen accuracy. These curves have been computed on the ensemble of 100 tests performed for each method and for each n. The curves highlight the different rates of convergence of the four algorithms.

In this second test, we compare the four algorithms on the Rastrigin function:

$$f(x) = A\,n + \sum_{i=1}^{n} [x_i^2 - A\cos(2\pi x_i)] \qquad (12)$$

where $A = 10$. Figure 3a and Figure 3b show the Rastrigin function in two dimensions. This function is a typical example of a non-convex function, with a global minimum in [0,…,0] and a high number of local minima, which increases exponentially with the dimension of the model space.

Specifically, this function has 11^n local minima for this range of the model space, i.e., $[-5\ 5]^n$. The high number of local minima makes the optimisation procedure applied to this function practically impossible for a local method.

First, we analyse the percentage of success of the four algorithms (Figure 4a). The GA method successfully converges for all the dimensions, whereas the other methods fail after a certain *n* value. The ASA algorithm has full success until n=38, after which there is an increase in failures because many tests exceed the maximum number of model evaluations allowed (10^7). Concerning the other two algorithms, the percentage of successful tests drops quickly to zero within a range of *n* from 1 to 8 for NA and from 15 to 30 for PSO.

Figure 4b shows the curves of convergence for the four algorithms for *n* varying from 2 to 60. By comparing these curves of convergence for the Rastrigin function with the curves of convergence for the De Jong function n°1 (Figure 2b), we note an overall increment of the mean number of evaluated models for all the algorithms and for all *n*. This fact can be easily explained by the more complex nature of the Rastrigin function with respect to the De Jong function n°1. Nevertheless, the convergence trends for ASA and GA appear to be the same as before, i.e., an exponential trend for ASA and a polynomial trend for GA. The NA and PSO methods again display worse scalability with *n* than ASA and GA. We are unable to derive a significant trend for NA because of the great number of failures in the tests, whereas the PSO algorithm displays a polynomial trend analogous to GA for low dimensions (n<18). Summarizing, the ASA algorithm is the best method for medium-low dimensions (4<n<20), but it shows an exponential convergence trend; differently, GA, being characterized by a polynomial convergence trend, after a crossing point (at n=20), eventually overcomes ASA in terms of performance for higher dimensions.

Tests on the Schwefel function

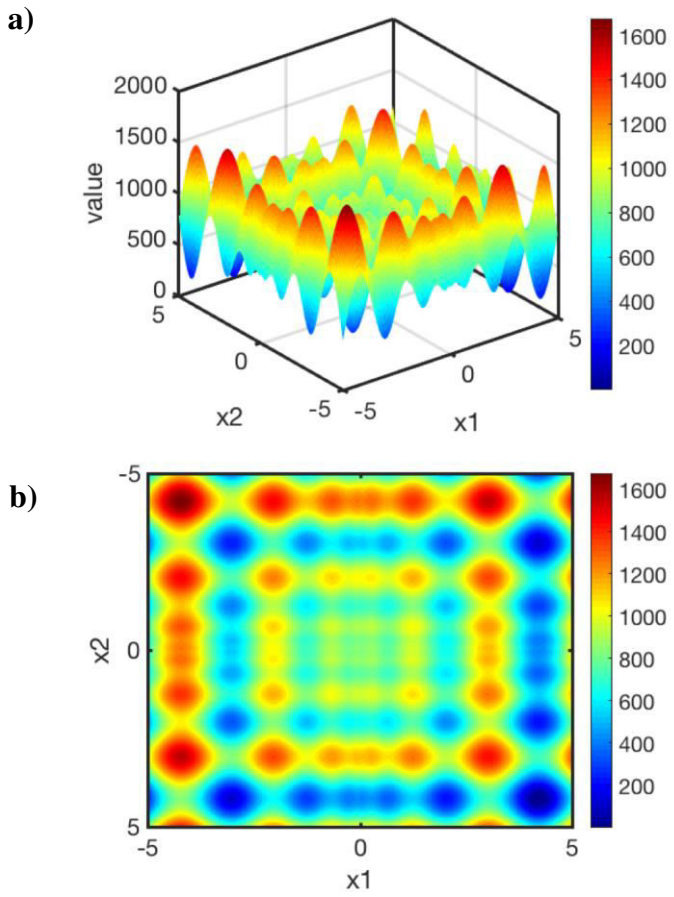

Figure 5: The Schwefel function with n=2, represented a) as a surface in 3D space and b) as a 2D projection.

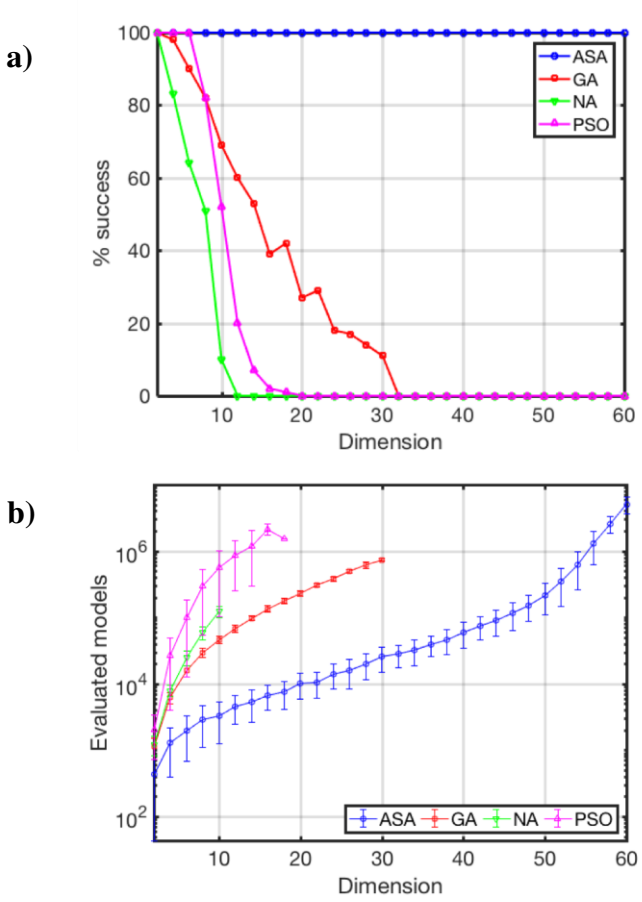

Figure 6: Summary of the results obtained on the Schwefel function. a) Percentage of successful tests (tests that have attained convergence within the chosen accuracy) as the dimension of the model space changes. b) Curves of convergence representing the mean number of evaluated models and the associated standard deviations needed to attain convergence within the chosen accuracy. These curves have been computed on the ensemble of 100 tests performed for each method and for each n. The curves highlight the different rates of convergence of the four algorithms.

The Schwefel function rescaled in the $[-5, 5]^n$ range is:

$$f(x) = A\,n + 100 * \sum_{i=1}^{n} [x_i * \sin(10 * \sqrt{|x_i|})] \qquad (13)$$

where A=418.9829. Similarly to the Rastrigin function, the Schwefel function has a large number of local minima equal to 7^n. However, differently from the Rastrigin function, in which the local minima surround the central global minimum, in the Schwefel function, the local minima are more

irregularly distributed, and important local minima are distant from the non-centred global minimum, which lies at [4.209687,…,4.209687], or are even located at the opposite edge of the model space.

Figure 5a and Figure 5b show the 2D Schwefel function as a surface in the three-dimensional space and as a projection onto a 2D map, respectively. Figure 6a displays the percentage of success of the four methods with n varying from 2 to 60. ASA successfully converges for all tests, displaying 100% success for all dimensions. Regarding the other methods, PSO reaches complete success for dimensions up to 6, while NA and GA are successful only for dimensions up to 2. For higher dimensions, only a fraction of the set of 100 tests successfully converges to the global minimum for NA, GA, and PSO. In particular, the percentage of successful tests in the ensemble of 100 tests quickly drops to zero in a range of n from 2 to 10 for NA and from 8 to 16 for PSO. Also, the percentage of successful tests for GA drops within a range from 2 to 30, but it does so more slowly than PSO and NA. The reduced number of failures for NA with respect to the Rastrigin function could be due to the lower number of local minima of the Schwefel function compared with the Rastrigin function. The increase of failures for PSO and GA could be because they remain entrapped in local minima that are far from the global minimum.

Figure 6b shows the curves of convergence of the four algorithms. The ASA algorithm has an exponential trend and is clearly the best-performing algorithm. For the GA and PSO methods, the convergence trends appear to be polynomial, even though the small number of successful tests used to derive the convergence curve is insufficient to infer reliable results (e.g., for GA with n=24, less than 25% of the tests successfully converged). The NA algorithm has a convergence curve close to the GA curve but is limited to the low dimensions (n<10).

To better understand the better (worse) performances of PSO and GA, for the Rastrigin (Schwefel) functions, we must consider that GA and PSO are stochastic methods in which the exploration of the model space in a given iteration is guided by the information collected in the previous iterations. For example, the information brought by each individual in a population (in

GA) or in a swarm (in PSO) is exploited and shared with the other models of the current iteration to find the most promising zones in the model space, which are the zones characterized by lower objective function values. However, this sharing process is not always successful, and it may not guide the exploration toward the global minimum if, somehow, the information brought by combining the model parameter values misguides the exploration of the search space. Let us consider the GA method and a simple heuristic example: in Figure 7a, we show a slice taken from the Rastrigin function in which we have two parents located in two suboptimal zones of the model space corresponding to two local minima. These two zones are not far from the global minimum, and hence, when the two individuals share their information, it is very likely that the generated offspring will draw closer to the global minimum. Differently, in the Schwefel function (Figure 7b), the position of the "second" best minimum is opposite the position of the global minimum. Therefore, in this case, if two individuals are exploring two zones corresponding to different local minima and if they share their information, it is very unlikely that the generated offspring will approach the global minimum. In other words, the GA and PSO methods seem to be more efficient in the case of regularly distributed minima, namely, in cases in which the global minimum is contained within the most-promising portion of the model space, that is, in a zone surrounded by many local minima.

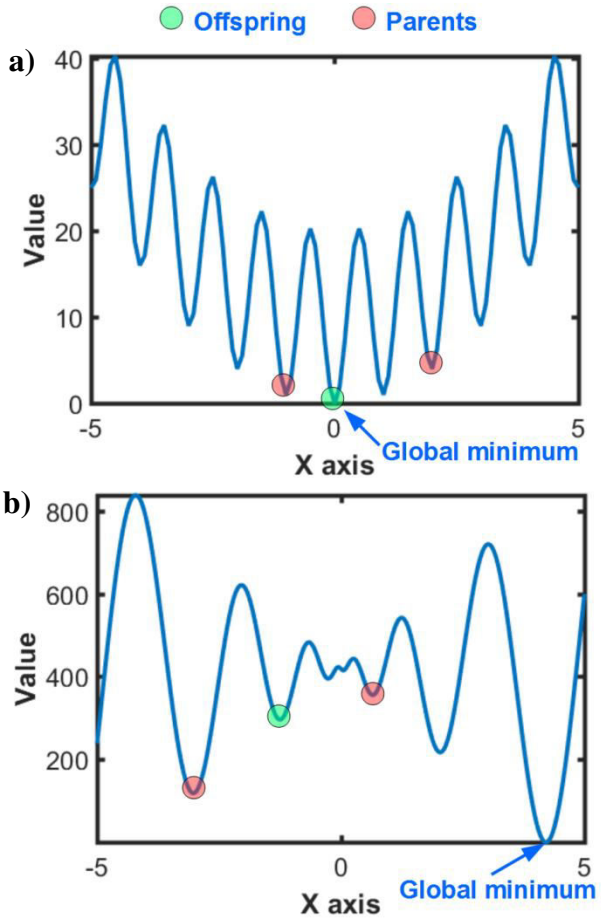

Figure 7: Schemes illustrating the exploration of the model space in a GA optimisation a) on the Rastrigin function and b) on the Schwefel function.

Tests on the Rosenbrock function

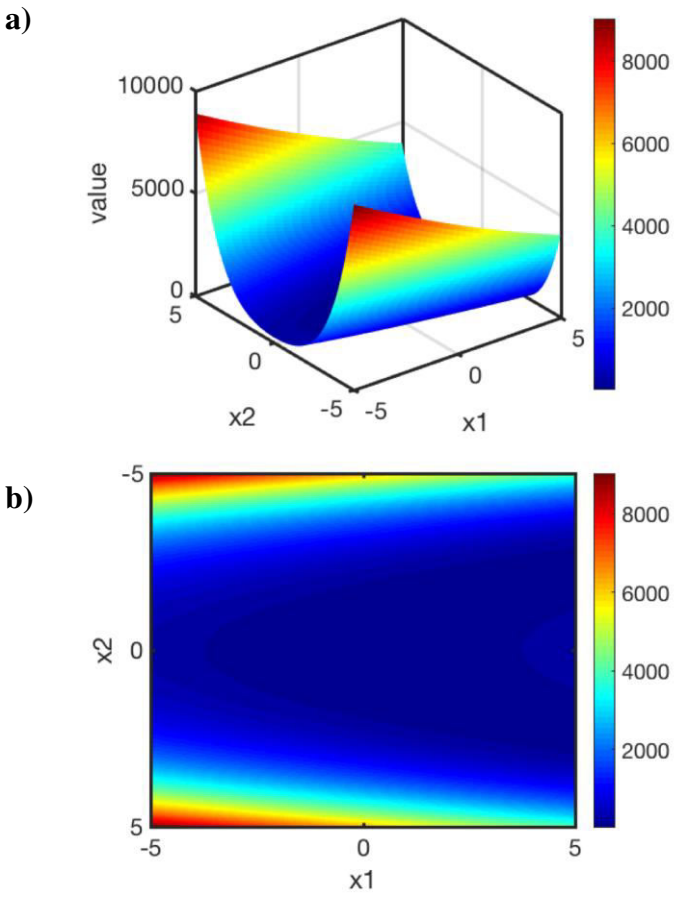

Figure 8: The Rosenbrock function with n=2, represented a) as a surface in 3D space and b) as a 2D projection.

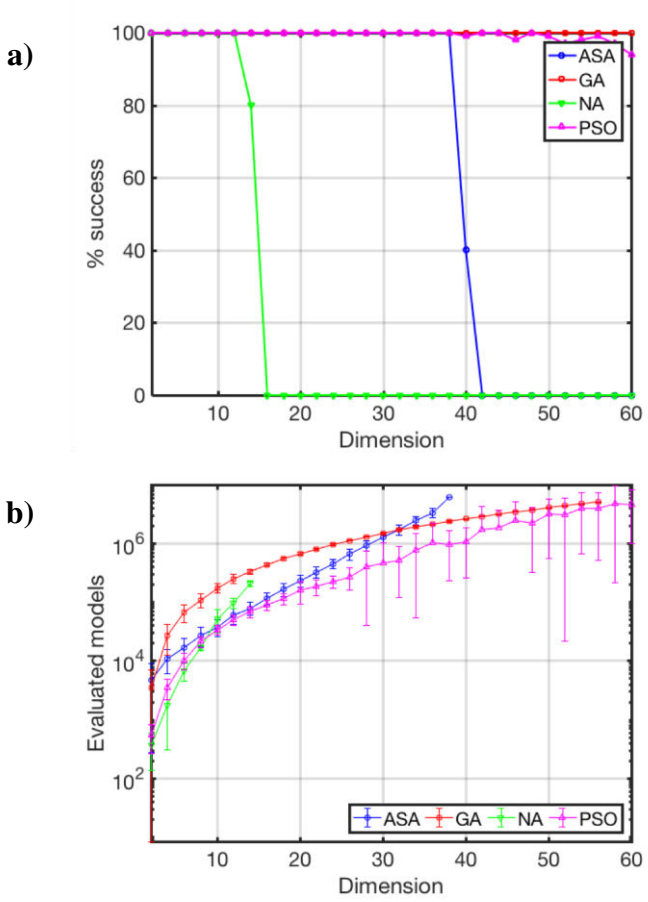

Figure 9: Summary of the results obtained on the Rosenbrock function. a) Percentage of successful tests (tests that have attained convergence within the chosen accuracy) as the dimension of the model space changes. b) Curves of convergence representing the mean number of evaluated models and the associated standard deviations needed to attain convergence within the chosen accuracy. These curves have been computed on the ensemble of 100 tests performed for each method and for each n. The curves highlight the different rates of convergence of the four algorithms.

The last objective function we study is the Rosenbrock function:

$$f(x) = \sum_{i=1}^{n/2} \mathrm{A}(x_{2i-1} - x_{2i}^2)^2 + (1 - x_{2i-1})^2$$	(14)

with $A = 100$ and n assuming only even values (for odd dimensions, the nature of the function changes, becoming a multi-minima function). Figure 8a and Figure 8b show that the Rosenbrock function is constituted by a very long, narrow, and parabolic-shaped valley. This function is convex with a single minimum located in the flat valley at [1,…,1]. Finding the valley is trivial, but converging to the minimum is very difficult. For this reason, the number of evaluated models required to converge is usually very high. In Figure 9a, we represent the percentage of success for dimensions varying from 2 to 60 for the four algorithms. GA has complete success for all the dimensions. The PSO algorithm exhibits a success rate higher than 90% for all the dimensions. In contrast, the NA algorithm fails completely for n>14, probably because of limits in the Voronoi approximation of the misfit surface, as previously discussed (see the comments on the NA results for the De Jong function n°1). The ASA algorithm is successful for n<40, whereas it fails for higher dimensions. In Figure 9b, we represent the curves of convergence for the four algorithms up to n=60. We note that their behaviour is similar to the De Jong case. The NA and ASA algorithms have an exponential trend, while GA and PSO show polynomial trends. NA is the best method for low dimensions (n<8), but it has the worst scalability with increasing n, confirming the characteristics shown in the previous test functions. The ASA algorithm exhibits good performance until n=40; for dimensions higher than 40, it exceeds the maximum number of evaluated models, so we consider it as failing to converge. The PSO and GA methods reach the global minimum within a reasonable number of evaluated models. Among the two, PSO is the best method for n<50, whereas the performances of PSO and GA are very similar for n>50.

1D elastic full-waveform inversion

Pre-stack waveform inversion has been widely solved using global-search algorithms (Sen and Stoffa, 1991; 1992; Mallick and Dutta, 2002; Fliedner et al., 2012; Sajeva et al. 2014; Aleardi and Mazzotti, 2017). We perform 1D elastic full-waveform inversions on synthetic seismic data, using a 1D reference elastic model composed of 12 layers, with a water depth of 400 m and a maximum

depth of 1400 m. In the inversion, the P-wave velocity (*Vp)*, S-wave velocity (*Vs)*, and density values are unknown, whereas the number of layers, their depths, the source signature and the water properties (*Vp*, density and water depth) are assumed known. Similarly to the optimisations on the analytic functions, we increase the number of unknowns in the tests, that is, we perform three tests characterized by an increasing number of layers to be inverted (3, 7, and 12), corresponding to 9, 21, and 36 unknowns, respectively. For each stochastic method and for each test, we perform five independent inversions from which the best result is extracted and discussed here. In the inversions, we used the L_2 norm between the predicted and observed seismograms as the objective function to be minimized. For the forward modelling, we use the reflectivity method (Kennett, 1983). For each inversion, we simulate a single shot gather with 30 receivers—spaced at 100 metres each, for a minimum offset of 100 m and a maximum offset of 3000 m—using a 5-Hz Ricker source wavelet. The ranges delimiting the search space for the model parameters are centred on the true parameter values and have a width of 600 m/s for both compressional and shear waves velocities and 0.6 g/cm^3 for densities. Finally, since the number of unknowns increases from test one to test three (from 9 to 36 unknowns), the maximum number of model evaluations allowed in each test increases accordingly. In particular, in the first, second, and third tests, we stop the inversions after 4100, 10000, and 12300 model evaluations, respectively. Table 5 summarizes the control parameters used in the FWI tests for the different algorithms. In all the FWI tests for the GA method, we use a number of individuals equal to ten times the number of the unknowns, a selection rate of 0.8, and a mutation rate of *1/n*, whereas the stochastic universal sampling is used as the selection method together with a linear ranking. For the ASA method, we use an initial generation temperature equal to 10*n* and a final generation temperature of 10^{-18}, and we apply the reannealing process. For PSO, we use a number of particles equal to ten times the number of unknowns, a local acceleration of 2.5 and a global acceleration of 1.6. For NA, the tuning parameters increase linearly with *n* to guarantee the exploration of the model space, and we use n_p=15*n* and n_r=2*n*.

ASA parameters		GA parameters		NA parameters		PSO parameters	
T_{g0}	10n	*N° individuals*	10*n*	n_p	15*n*	*N° particles*	10*n*
T_{gf}	10^{-18}	*Sel. rate*	0.8	n_r	2*n*	a_{glob}	1.6
Reannealing	yes	*Mut. rate*	1/*n*			a_{loc}	2.5
						v_{max}	100

Table 5: The principal control parameters used in the FWI tests for each stochastic method; n indicates the number of model parameters considered in the inversion.

Before discussing the inversion results, we want to more thoroughly investigate the shape of the objective function that characterizes the 1D elastic FWI. This analysis allows us to determine which elastic parameters are the most significant in determining the seismic response. In other words, it allows us to understand which parameters will be better resolved in the inversion. For graphical purposes, we limit our attention to a simple case with three unknown model parameters. To this end, we consider a very simple 1D elastic model with a single reflecting interface separating two half spaces. The overlying layer simulates the water column with a *Vp* equal to 1500 m/s, a null *Vs* and a density of 1g/cm^3, whereas the underlying layer has a *Vp* equal to 1800 m/s, a *Vs* of 800 m/s and a density of 1.6 g/cm^3. In this model, we compute the associated seismogram using the same acquisition configuration described previously. We assume the properties of the overlying layer to be known, and hence, we keep them fixed to their true values and we perturb the properties of the underlying layer. For each perturbed model, we derive the associated seismogram and compute the objective function, which is the L_2 norm difference between this seismogram and the seismogram obtained on the original unperturbed model. Figure 10 represents the so-derived objective function. We observe an elongated valley of low misfit mainly stretching along the density axis, which indicates the minor influence of this model parameter in determining the seismic response. This elongated valley demonstrates the ill-conditioning of the 1D elastic FWI and the difficulty of a reliable density estimation when the seismic data are single-component and have a limited offset

range (Fliedner et al., 2012; Aleardi and Mazzotti, 2017), whereas higher resolution characterizes the *Vs* parameter and, particularly, the *Vp* parameter. Note that the objective function in Figure 10 is somewhat similar to the Rosenbrock function. Although examining the objective function within a higher-dimensional context could provide further information, it would be more difficult to visualize as well as highly expensive to compute. The example shown here, even if oversimplified because it is limited to a 3D objective function, nonetheless provides significant insight into some of the most problematic aspects of the 1D elastic FWI.

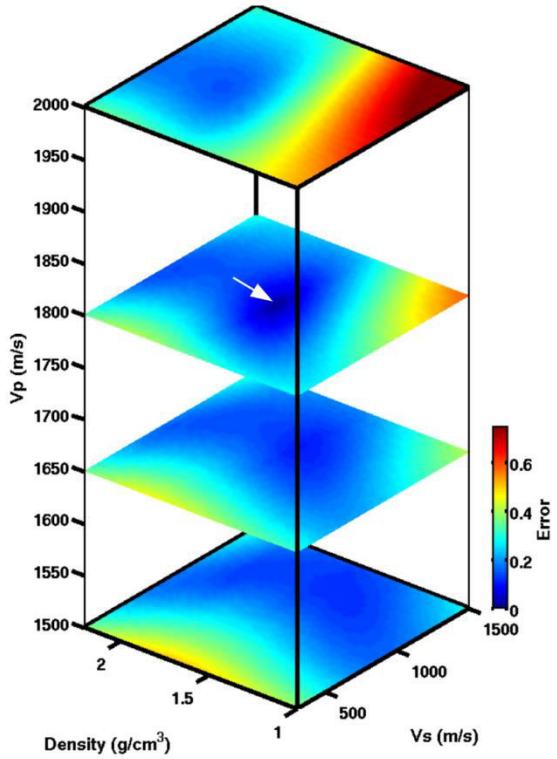

Figure 10: Objective function for the 1D elastic FWI computed on a simple model composed of 2 half spaces. The objective function has been computed by keeping the elastic properties (Vp, Vs and density) of the overlying layer fixed to their correct values and by perturbing the properties of the underlying layer. The white arrow points to the true property values of the underlying layer.

In Figure 11, Figure 12 and Figure 13, we show the subsurface elastic models predicted by each stochastic method for the three-layer, seven-layer, and twelve-layer tests, respectively. Each figure shows comparisons between the *Vp*, *Vs*, and densities predicted by each method and the true

property values. For the three-layer inversion, we observe that the estimated seismic velocities for the GA, ASA, NA, and PSO methods are very close to the true velocity values. The estimated density depth trend is also very close to the true one for the GA, PSO, and ASA methods, whereas the NA method produces less-accurate results. Similar conclusions can be drawn from the seven-layer inversion. Also, in this case, the velocity trends are very well predicted by the four stochastic methods. However, note that the elastic properties estimated by the GA, PSO, and ASA methods are closer to the true values than the NA results. For the twelve-layer test, we observe that the GA and PSO methods seem to perform better than the ASA and NA algorithms, and they yield final velocity and density depth trends that are very close to the true ones. Moreover, in this test, the GA and PSO methods are able to reproduce the numerous increases and decreases in velocity and density that characterize the true subsurface reference model. Conversely, ASA and NA return less-satisfactory predictions. In particular, the elastic properties predicted by the NA method are very far from the true ones. To better quantify the differences between the predicted and true elastic properties, we compute the percentage mean errors for *Vp*, *Vs*, and density for each method and for each inversion test (three-, seven-, and twelve-layer). The formula of the percentage mean error for a given model parameter is as follows:

$$Error(\%) = \frac{100}{N}\sum_{i=1}^{N}\frac{|m_i^{pred} - m_i^{true}|}{|m_i^{true}|} \quad (15)$$

where N is the number of layers considered in the inversion, and m^{true} and m^{pred} indicate the property of the true and predicted models (*Vp*, *Vs,* or density), respectively. These percentage mean errors are displayed in Figure 14. Independently from the number of inverted layers and from the analysed method, we note that the percentage error generally tends to increase when passing from *Vp* to *Vs* and to density. This is obviously expected from the analysis of Figure 10, which demonstrated that the *Vs* and, particularly, the density parameters play a very minor role with respect to *Vp* in determining the observed seismogram. For the NA method, we observe a decrease in the mean percentage error for the density parameter as the number of unknowns increases. This

anomalous result can be ascribed to the severe ill-conditioning that characterizes this model parameter and the difficulty the NA method has in efficiently exploring the model space in the case of an elongated valley in the objective function (see also the results obtained for the Rosenbrock function discussed previously). This fact makes the density estimation for the NA method very problematic and poorly reliable. Figure 14 makes clear that the percentage mean error tends to increase when passing from GA to PSO to ASA and to NA in all the inversion tests, although PSO and GA return very similar mean model errors. The NA method is by far the worst-performing method in the twelve-layer inversion. The ASA tests return a model misfit comparable to those of GA and PSO for the three-layer and seven-layer tests, whereas ASA worsens its performance for the twelve-layer test.

Figure 15 shows the observed seismograms and the differences between the observed and predicted seismograms for the three-, seven-, and twelve-layer inversions. In all cases, the GA, PSO, and ASA methods return final predicted seismograms with nearly the same differences with respect to the observed data. The NA method returns final predicted seismic data very similar to the observed data in the three-layer test, whereas the differences between the observed and predicted data are much more prominent in the seven-layer and, particularly, the twelve-layer inversions. Summarizing, the analysis of the data misfit confirms the better performance of GA, PSO, and ASA with respect to NA.

After observing that the Rosenbrock function has a fair similarity to the objective function in the 1D elastic FWI, we can conclude that the FWI results shown here confirm the conclusions drawn in the analytic tests on the Rosenbrock function. The GA and PSO methods are very effective at exploring the model space in the case of an error function characterized by a nearly flat valley of low values. Differently from ASA and NA, the performances of GA and PSO are also less affected by the increase of unknown model parameters. The ASA method confirms its efficiency for low-dimensional model spaces, and this efficiency decreases as the number of unknowns increases. NA

severely suffers from the increase of unknown model parameters, and it is unable to efficiently explore the model space in the case of a severely ill-conditioned problem.

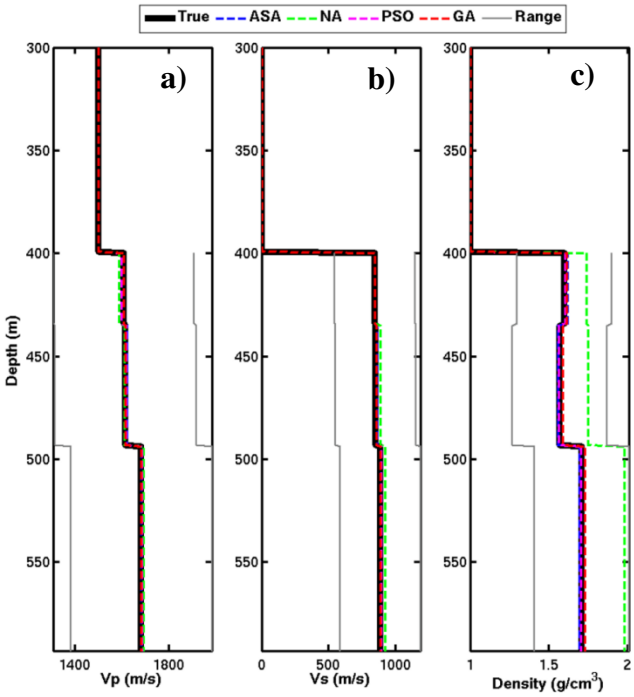

Figure 11: Comparison between the true subsurface elastic model and the models predicted by the different algorithms in the three-layer test. Vp, Vs, and density are represented in a), b), and c), respectively.

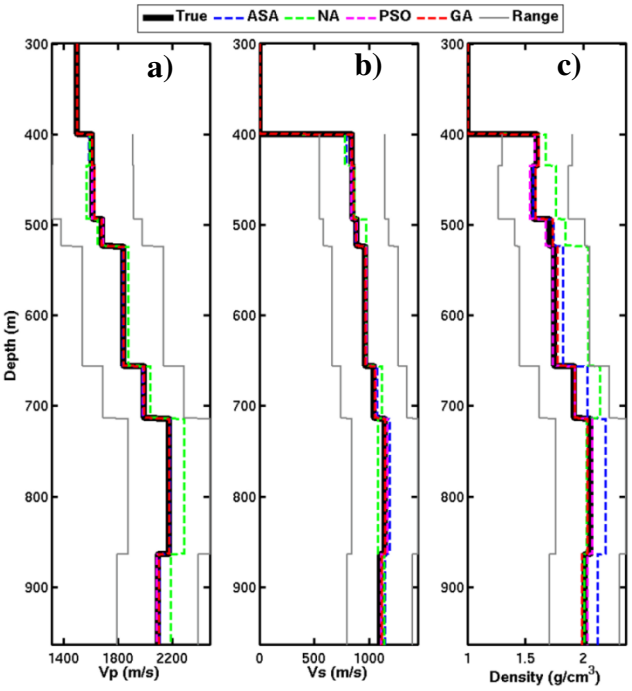

Figure 12: Comparison between the true subsurface elastic model and the models predicted by the different algorithms in the seven-layer test. Vp, Vs, and density are represented in a), b), and c), respectively.

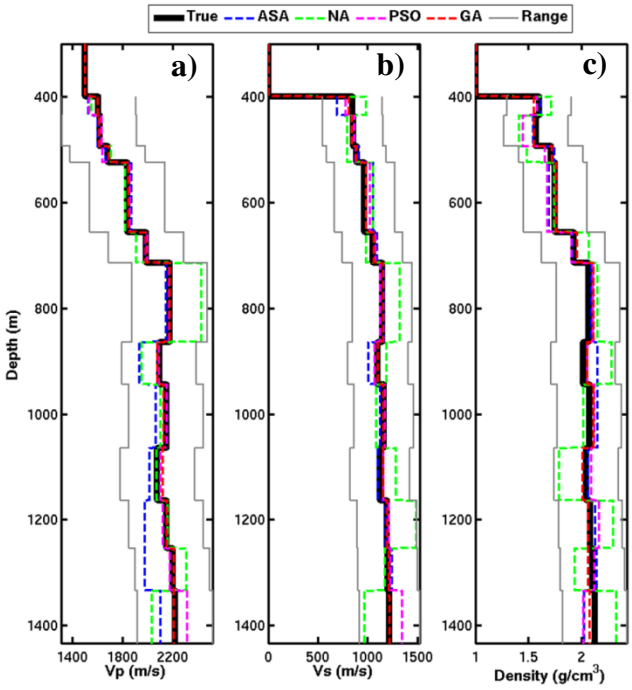

Figure 13: Comparison between the true subsurface elastic model and the models predicted by the different algorithms in the twelve-layer test. Vp, Vs, and density are represented in a), b), and c), respectively.

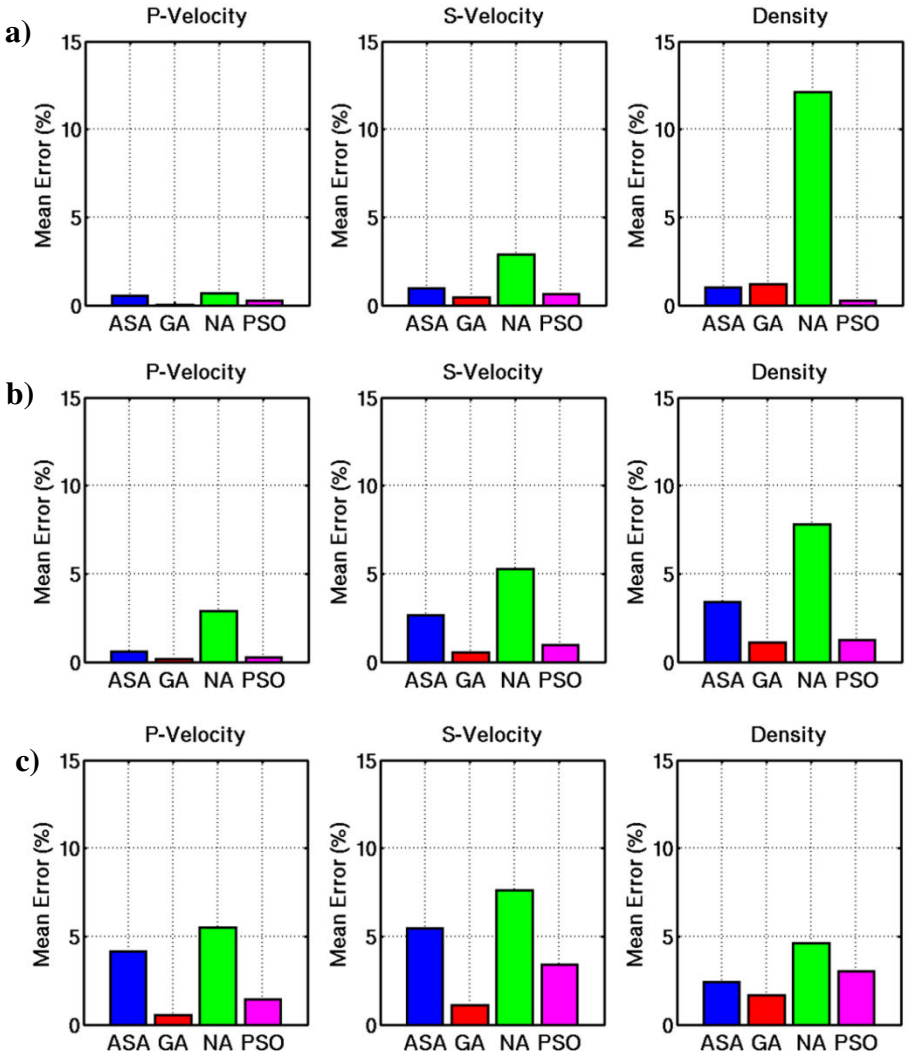

Figure 14: a), b), and c) Percentage mean errors for each model parameter (Vp, Vs, and density) for the three-layer, seven-layer and twelve-layer tests, respectively. The Vp, Vs, and density errors are represented from left to right in a), b), and c).

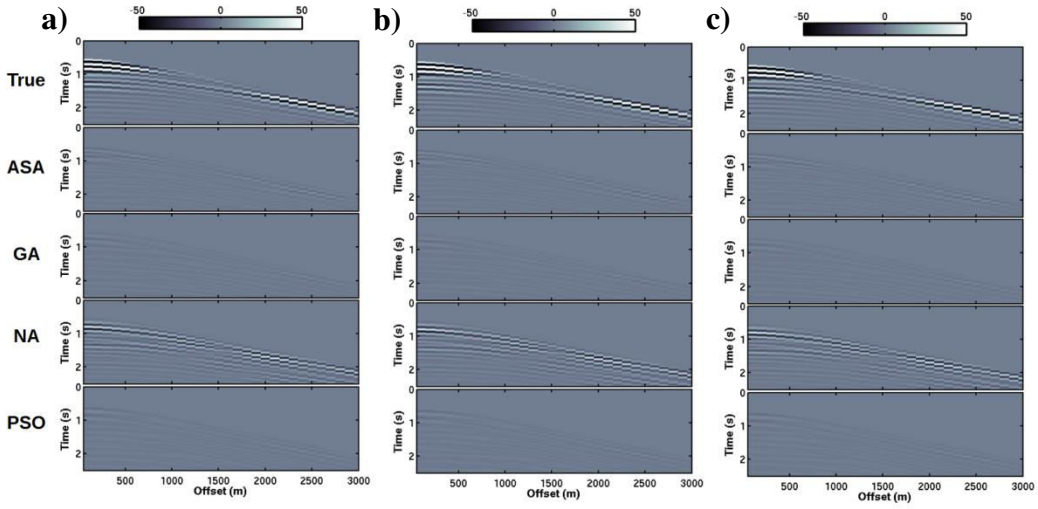

Figure 15: From top to bottom, the seismograms computed on the true model and the differences between these seismograms and those computed on the best models predicted by each stochastic method. a), b), and c) refer to the three-layer, seven-layer, and twelve-layer tests, respectively.

CMP-consistent residual static corrections

The second geophysical example concerns the computations of the residual statics correction. Rothman (1985; 1986) was the first to apply a stochastic method (simulating annealing) to solve this highly non-linear multi-minima optimisation problem. In his work, Rothman focused on the solution of the surface-consistent residual static computation. However, for the sake of simplicity, we limit our attention to the CMP-consistent residual static for a single CMP gather. Like the previous tests on the analytic functions and the FWI examples, we analyse the performance of each stochastic method as the number of unknown model parameters increases. In this case, the unknowns are the time shifts that must be applied trace by trace to the CMP in order to maximize the energy of the associated stack trace. We discuss three cases in which the CMP gather is constituted by 10, 20, and 40 traces, corresponding to 10, 20, and 40 unknown time shifts. As in the FWI tests, the outcomes of each method are compared after a fixed number of model evaluations (1000, 2000, and 4000 for the 10-, 20-, and 40-trace tests, respectively). Similarly to the FWI examples, we perform five independent inversions for each stochastic method, and for each test, we

extract and analyse the best result, that is, the model producing the stack trace with the highest energy.

To generate the reference CMP gather (without residual static), we use the *Vp*, *Vs*, and density values extracted by actual well log data and a 1D convolutional forward modelling with a 50-Hz Ricker wavelet as the source signature and with a sampling interval of 2 ms. To simulate residual statics in the data, we apply to each trace in the reference CMP a time shift randomly generated with a uniform probability over the range -15/+15 ms, whereas in the subsequent optimisation process, we allow time shifts within the range -25/+25 ms.

In the following tests for the GA method, we use a number of individuals equal to five times the number of unknowns, a selection rate of 0.8, and a mutation probability of 1/*n*. Stochastic universal sampling is used as selection method together with a linear ranking. For the ASA method, we employ an initial generation temperature equal to 10*n* and a final generation temperature of 10^{-18}. We also apply the reannealing process. For PSO, we choose a number of particles equal to 5 times the number of unknowns, a local acceleration of 2, and a global acceleration equal to 2. For NA, we use an n_p value equal to 5 times the number of unknowns and an n_r value equal to 1/5 of the number of unknowns. These parameters are summarized in Table 6.

ASA parameters		**GA parameters**		**NA parameters**		**PSO parameters**	
T_{g0}	10*n*	*N° individuals*	5*n*	n_p	5*n*	*N° particles*	5*n*
T_{gf}	10^{-18}	*Sel. rate*	0.8	n_r	*n/5*	a_{glob}	2
Reannealing	yes	*Mut. rate*	1/*n*			a_{loc}	2
						v_{max}	10

Table 6: The principal control parameters used in the residual static tests for each stochastic method; n indicates the number of model parameters used in the inversion.

Before discussing the performances of the four stochastic methods in the residual static computation, we analyse the objective function that characterizes this optimisation problem. The objective function we aim minimize is the negative of the stack trace energy associated with the considered CMP gather. Similarly to the analysis performed for the 1D elastic FWI, we consider an oversimplified problem with only two unknowns, that is, the reference CMP in which only two traces have been randomly time-shifted: the first trace with a time shift equal to -15 ms and the second trace with a time shift equal to 10 ms. Then, for all the possible combinations of time shifts for these two traces, we compute the associated energy of the stack trace. The resulting objective function is represented in Figure 16. As expected, the global minimum is located at time shifts equal to 15 ms and -10 ms for the first and second traces, respectively, which correspond to the original time positions of the two traces in the reference CMP gather. The objective function is characterized by a closely spaced distribution of minima generated by the well-known cycle-skipping effect. The distribution of these minima is fairly irregular and is not symmetric with respect to the 0 time translations, and some minima valleys are narrower than others. From the above considerations and from the comparison of Figure 16 with the analytic functions previously described, it is found that the residual static corrections are characterized by an objective function with some similarities to both the Rastrigin and Schwefel functions (see Figure 3b and Figure 5b).

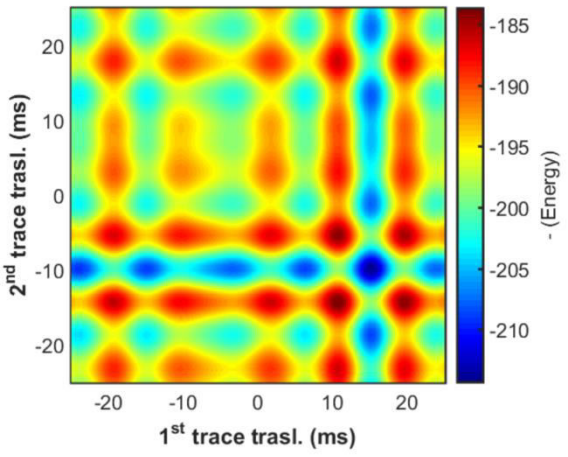

Figure 16: The objective function associated with the CMP-consistent residual static corrections. To compute this objective function, we consider the reference CMP gather in which

only two traces are shifted, while the other traces are kept fixed to their correct time position. Note that this objective function presents some similarities with both the Rastrigin and Schwefel functions (see Figure *3b and* Figure 5*b).*

The results obtained with the four stochastic methods in the three tests with 10, 20, and 40 traces in the CMP gather are shown in Figure 17, Figure 18, Figure 19 and Figure 20. We note that in the 10-trace case, the ASA, GA, PSO, and NA methods are able to perfectly reconstruct the reference CMP gather and the energy of the stack trace generated by the reference CMP. In the 20-trace case, only the ASA algorithm correctly predicted the time shifts that must be applied to each trace to perfectly reproduce the reference CMP. The outcomes of the GA and PSO methods are very similar but worse than the ASA result. Again, in the last case, with 40 traces, the ASA method shows the best performance, followed by the PSO and GA methods. In the 20- and 40-trace cases, it is the NA method that shows the worst performances. These results are very similar to those discussed previously for the Schwefel function. This fact can be ascribed to the high similarity between the objective function associated with the residual static computation and the Schwefel function. The GA and PSO methods seem to suffer from the uneven distribution of minima, especially for high-dimensional model spaces. This lack of convergence is even more pronounced for the NA method, which again shows its limits in efficiently exploring the model space in the case of a high-dimensional model space and in the case of a multi-minima objective function. This makes the NA method easily prone to becoming trapped in local minima.

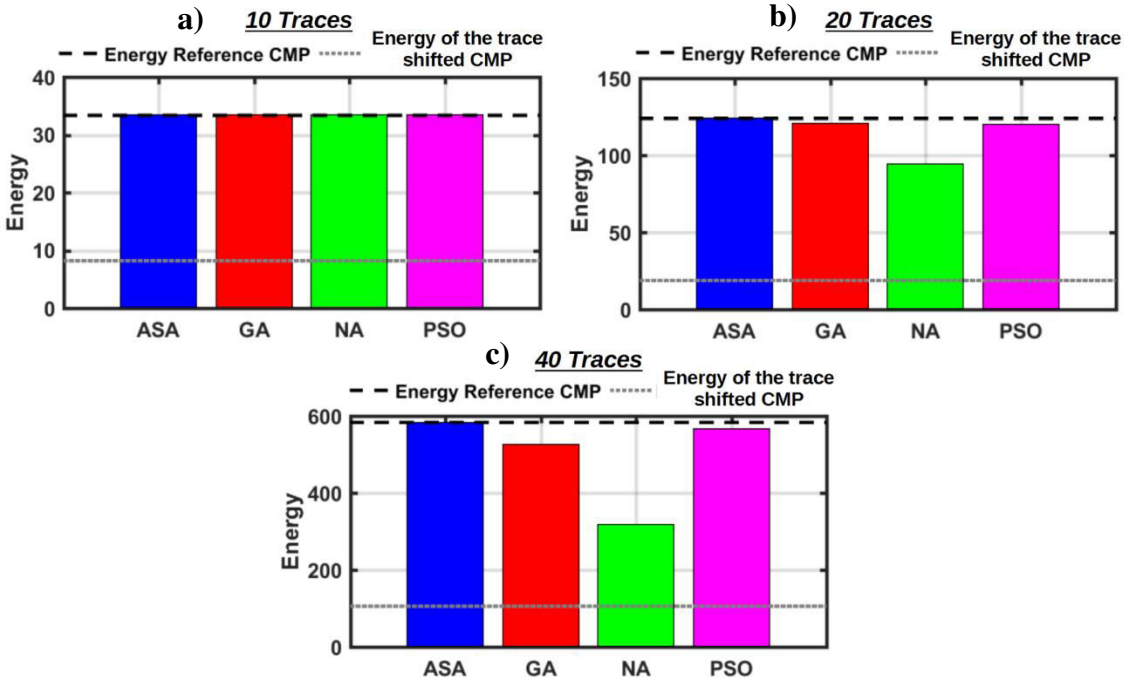

Figure 17: Comparison of the performances of each method in the residual static computation. a), b), and c) represent the energy of the stack trace associated with the CMP gather before static correction (grey dotted line), the energy of the stack trace generated by the reference CMP (black dashed line) and the energy of the stack trace associated with the CMP gathers after residual static corrections with the time shifts predicted by each stochastic method (coloured bars). a), b) and c) refer to the cases with 10, 20, and 40 traces in the CMP gather.

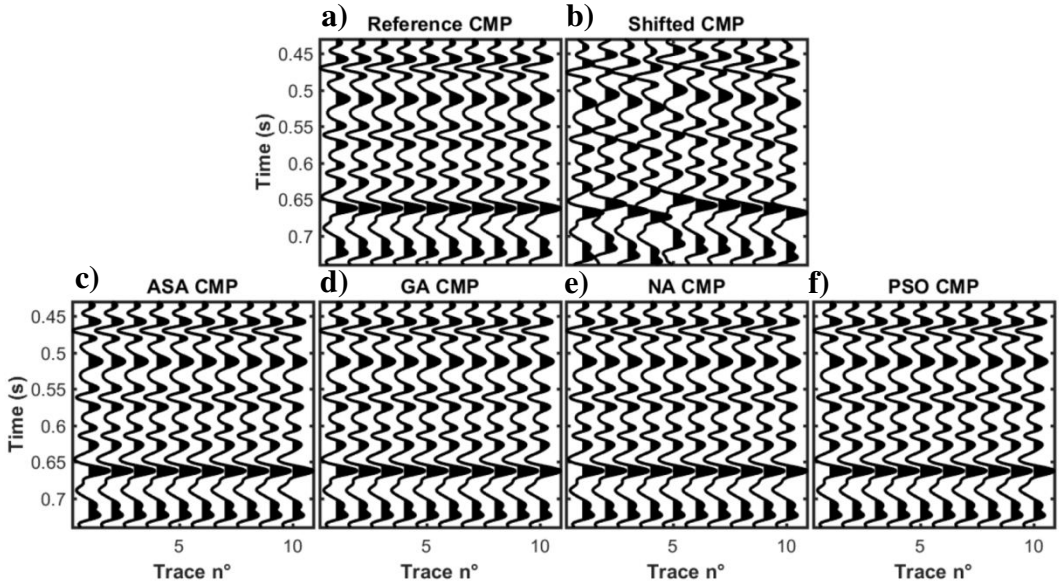

Figure 18: Comparison of a) the reference CMP gather, b) the trace time-shifted CMP gather, which simulates a CMP gather before residual static corrections, and c) the final CMP gathers after residual static corrections with the time shifts predicted by each method. The CMP gathers reconstructed by ASA, GA, NA, and PSO are represented in c), d), e), and f), respectively. This figure refers to the test in which the CMP gather contains 10 traces.

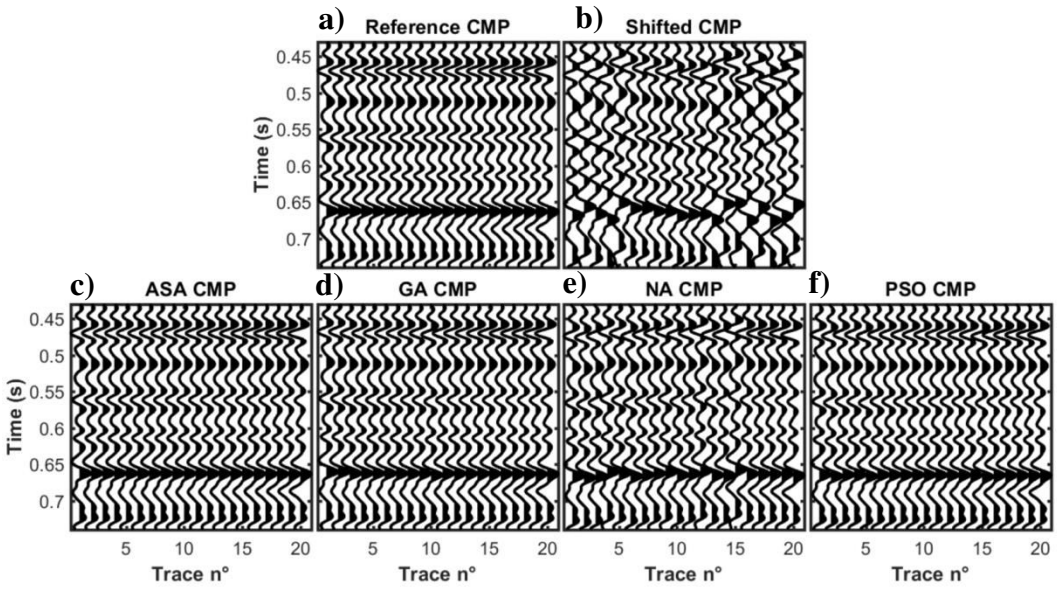

Figure 19: Comparison of a) the reference CMP gather, b) the trace time-shifted CMP gather, which simulates a CMP gather before residual static corrections, and c) the final CMP gathers after residual static corrections with the time shifts predicted by each method. The CMP gathers reconstructed by ASA, GA, NA, and PSO are represented in c), d), e), and f), respectively. This figure refers to the test in which the CMP gather contains 20 traces.

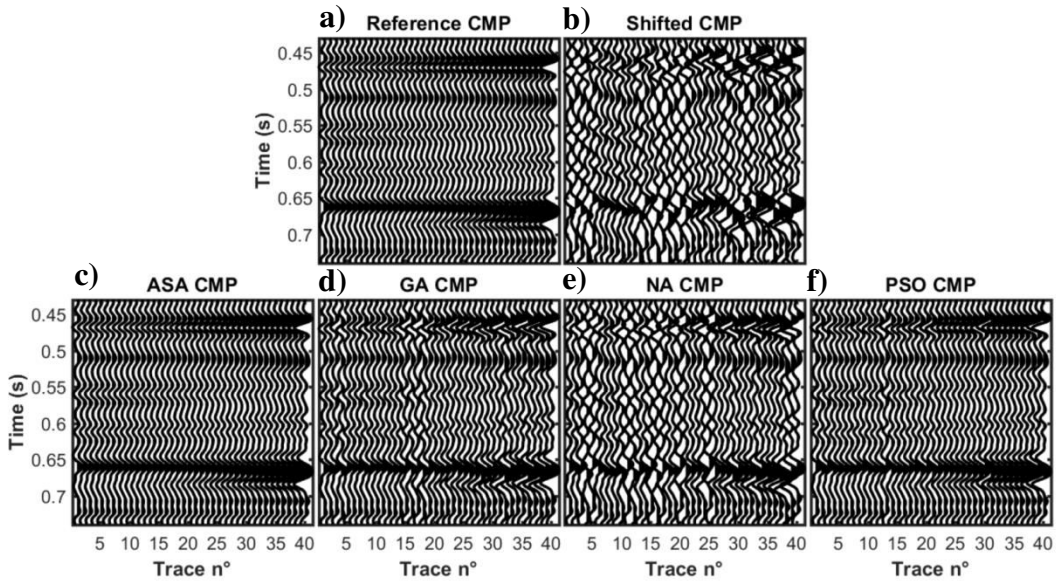

Figure 20: Comparison of a) the reference CMP gather, b) the trace time-shifted CMP gather, which simulates a CMP gather before residual static corrections, and c) the final CMP gathers after residual static corrections with the time shifts predicted by each method. The CMP gathers reconstructed by ASA, GA, NA, and PSO are represented in c), d), e), and f), respectively. This figure refers to the test in which the CMP gather contains 40 traces.

Discussion

The results presented in this paper have been obtained by applying four global optimization methods that are frequently used for solving geophysical inverse problems. However, the classes of stochastic or semi-stochastic methods include many other techniques that we have not tested. Moreover, many different implementations exist for each stochastic method and, in our study, we only selected the most popular ones to test their performances and their convergence properties on a subset of the many objective functions that characterise geophysical inverse problems. However, a complete and comprehensive review of all stochastic methods with their many variants and for all the possible kinds of objective functions is beyond the scope of this work and is likely beyond the reach of any single scientific paper. Therefore, the results obtained in this paper cannot be extended to predict the performance of each variant of the four stochastic methods, but they can be useful as a general guide to determining the suitability of each method in solving particular problems. In this

respect, an analysis of the objective function of the problem at hand, even for a limited dimension of the model space, can yield useful insights into the most appropriate approach to use.

Regarding the parameter setting, among the stochastic methods we analysed, NA seems to be the method whose results are most affected by the user-defined parameters. Therefore, in applying this method, particular care must be taken in setting the n_p and n_r values. Unfortunately, this fine tuning can be very time consuming in the case of optimisation problems with expensive forward modelling or with a large number of unknowns.

Concerning the computational time, reducing the cost of an optimisation problem can be crucial, both in the case of expensive objective-function evaluation (i.e., FWI) and in the case of a large-dimensional model space. To this end, the ability to simultaneously evaluate all models in a given iteration may greatly reduce the computational cost. An iteration consists in a generation for GA, a set of n_p models for NA, and a swarm in PSO. Obviously, the number of simultaneously evaluated models is limited by the number of available cpu's. Note that this parallel approach can be applied to GA, NA, and PSO, whereas it cannot be applied to the ASA method (or to any other simulated annealing version). This is because simulated annealing is a largely sequential Monte Carlo algorithm in which a single model is explored at a given iteration and it entirely depends on the model explored in the previous iteration. This characteristic often makes the standard ASA method inapplicable in the case of optimisation problems with expensive forward modelling. Different strategies have been developed to speed up the simulated annealing method, and some of these strategies hybridize simulated annealing with some of the principles of the GA method (Chen and Flann, 1994; Chen and Shahandashti, 2009).

Conclusions

We compared the performances of four stochastic methods: adaptive simulated annealing, a real-coded genetic algorithm, the neighbourhood algorithm, and Clerc's version of particle swarm optimisation. A set of four analytic functions and two geophysical simulation tests (1D elastic FWI

and residual statics computation) were used. The four algorithms were tested on a wide range of dimensions of the model space (varying from 2 to 60), and the four analytic functions were chosen to test the methods in very different scenarios. For all analytic functions (the De Jong n°1, Rastrigin, Schwefel, and Rosenbrock functions), we observed that the curve of convergence as a function of the dimension of the model space has an exponential trend for ASA and NA and a polynomial trend for PSO and GA. This suggests that PSO and GA are more appropriate for large-dimensional optimisation problems (n>40), even though these two methods are subject to failure in the case of many local minima (PSO) or in the case of irregularly distributed minima (GA). Differently, because of their exponential trend with n, ASA and NA can be only used for optimisation algorithms for small to intermediate dimensions of the model space (n<10 for NA and n<40 for ASA, in our tests). In addition, NA is subject to failure in the case of multiple minima, even with small model space dimensions (n<10).

The results for the synthetic seismic optimisation problems confirm the conclusions drawn in the tests on the analytic functions. In more detail, in the 1D elastic FWI, GA is the best-performing method for all dimensions (3-, 7- and 12-layer tests, that is, 9, 21 and 36 unknowns, respectively). ASA and PSO yield very similar and good performances for small to intermediate dimensions of the model space (3- and 7-layer inversions), whereas for the 12-layer test, the performance of the ASA algorithm clearly worsens compared to the performances of GA and PSO, again confirming that ASA efficiency decreases with increased model-space dimensionality. The NA method exhibits a limited capability to explore the model space and a sudden decrease in performance as the dimension of the model space increases. The residual statics test confirms that the performance of GA is affected by irregularly distributed minima and that NA is prone to becoming trapped in local minima in the case of high-dimensional model spaces and multi-minima objective functions. For this test, the ASA method seems to be the most efficient algorithm, closely followed by PSO.

Acknowledgments

These results were obtained within a research project with Eni. We thank Eni for the permission to publish this paper and Nicola Bienati of Eni for useful discussions and insightful comments.

References

Aguiare H., Junior O., Ingber L., Petraglia A., Petraglia M.R. and Machado M.A.S. 2012. Stochastic global optimization and its applications with fuzzy adaptive simulated annealing. Springer Heidelberg New York Dorddrecht London.

Aleardi M. and Ciabarri, F. 2017. Assessment of different approaches to rock-physics modeling: A case study from offshore Nile Delta. Geophysics **82**(1), MR15-MR25. DOI: 10.1190/geo2016-0194.1

Aleardi M. and Mazzotti A. 2017. 1D elastic full-waveform inversion and uncertainty estimation by means of a hybrid genetic algorithm-gibbs sampler approach. Geophysical Prospecting, **65** (1), 64-85. DOI: 10.1111/1365-2478.12397

Aleardi M., Tognarelli A. and Mazzotti A. 2016. Characterisation of shallow marine sediments using high-resolution velocity analysis and genetic-algorithm-driven 1D elastic full-waveform inversion. Near Surface Geophysics **14**(5), 449-460. DOI: 10.3997/1873-0604.2016030

Aleardi M. 2015. Seismic velocity estimation from well log data with genetic algorithms in comparison to neural networks and multilinear approaches. Journal of Applied Geophysics **117**, 13-22. DOI: 10.1016/j.jappgeo.2015.03.021

Anderssen R.S. 1970. The Character of Non-Uniqueness in the Conductivity Modelling Problem for the Earth. Pure and Applied Geophysics **80** (1), 238–259. DOI: 10.1007/BF00880211

Bäck T. and Hoffmeister F. 1991. Extended selection mechanisms in genetic algorithms. Morgan Kaufmann publisher.

Backus G. and Gilbert F. 1968. The Resolving Power of Gross Earth Data. Geophysical Journal of the Royal Astronomical Society **16**, 169–205. DOI: 10.1111/j.1365-246X.1968.tb00216.x

Blickle T. and Thiele L. 1995. A comparison of selection schemes used in genetic algorithms. TIK report, 11.

Buland R. 1976. The Mechanics of Locating Earthquakes. Bulletin of the Seismological Society of America **66** (1), 173–187.

Chen H. and Flann N. S. 1994. Parallel simulated annealing and genetic algorithms: a space of hybrid methods. In Parallel Problem Solving from Nature. Springer Berlin Heidelberg. DOI: 10.1007/3-540-58484-6_286

Chen P.H. and Shahandashti S.M. 2009. Hybrid of genetic algorithm and simulated annealing for multiple project scheduling with multiple resource constraints. Automation in Construction **18** (4), 434-443. DOI: 10.1007/3-540-58484-6_286

Clerc M. 1999. The Swarm and Queen: Towards A Deterministic and Adaptive Particle Swarm Optimization. Proceedings of the IEEE Congress on Evolutionary Computation, 1951-1957.

Clerc M. and Kennedy J. 2002. The particle swarm−explosion, stability, and convergence in a multidimensional complex space. IEEE Transactions on Evolutionary Computation **6**(1), 58-73. DOI: 10.1109/4235.985692

Fernández Martínez J.L., Gonzalo E.G., Álvarez J.P.F., Kuzma H.A. and Pérez C.O.M. 2010. PSO: a powerful algorithm to solve geophysical inverse problems: application to a 1D-DC resistivity case. Journal of Applied Geophysics **71**(1), 13-25. DOI: 10.1016/j.jappgeo.2010.02.001

Fernández Martínez J.L., Mukerji T., García Gonzalo E., and Suman, A. 2012. Reservoir characterization and inversion uncertainty via a family of particle swarm optimizers. Geophysics **77** (1), M1-M16. DOI: 10.1190/1.3513319

Fliedner M. M., Treitel S. and MacGregor L. 2012. Full-waveform inversion of seismic data with the Neighborhood Algorithm. The Leading Edge **31** (5), 570-579. DOI: 10.1190/tle31050570.1

Goldberg D. E. 1989. Genetic algorithms in search, optimisation, and machine learning. Reading Menlo Park: Addison-Wesley.

Hassan R., Cohanim B., De Weck O. and Venter G. 2005. A Comparison of Particle Swarm Optimisation and the Genetic Algorithm. In Proceedings of the 1st AIAA Multidisciplinary Design Optimisation Specialist Conference, 18–21.

Hastings W.K. 1970. Monte Carlo sampling methods using Markov Chain and their applications, Biometrika **57**, 97–109.

Hermance, J. F. and Grillot, L. R. 1974. Constraints on temperatures beneath Iceland from magnetotelluric data. Physics of the Earth and Planetary Interiors **8**(1), 1-12. DOI: 10.1016/0031-9201(74)90104-6.

Horne S. and Macbeth C. 1998. A Comparison of Global Optimisation Methods for near-offset VSP Inversion. Computers and Geosciences **24** (6), 563–572. DOI: 10.1016/S0098-3004(98)00023-5.

Holland J.H. 1975. Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence. University Michigan Press.

Ingber L. 1989. Very fast simulated re-annealing. Mathematical and Computer Modelling **12** (8), 967-973.

Ingber L. and Rosen B. 1992. Genetic Algorithms and Very Fast Simulated Reannealing: A Comparison. Mathematical and Computer Modelling **16** (11), 87–100. DOI: 10.1016/0895-7177(89)90202-1.

Janikow C.Z. and Michalewicz, Z. 1991. An experimental comparison of binary and floating point representations in genetic algorithms. In Proceedings of international conference on genetic algorithms, 31-36.

Jestin F., Huchon P. and Gaulier J.M. 1994. The Somalia Plate and the East-African Rift System - Present-Day Kinematics. Geophysical Journal International **116** (3), 637–654. DOI: 10.1111/j.1365-246X.1994.tb03286.x

Keilis-Borok V.I. and Yanovskaja T.B. 1967. Inverse Problems of Seismology (Structural Review). Geophysical Journal International **13** (1-3), 223–234. DOI: 10.1111/j.1365-246X.1967.tb02156.x

Kennedy J. and Eberhart R. 1995. Particle swarm optimisation. In Proceedings of the IEEE International Conference on Neural Networks **4**, 1942– 1948.

Kennedy J., Kennedy J.F., Eberhart R.C. and Shi Y. 2001. Swarm intelligence. Morgan Kaufmann.

Kennett B.L. 1983. Seismic wave propagation in stratified media. Cambridge University Press.

Kirkpatrick S., Gelatt C.D. and Vecchi M.P. 1983. Optimisation by Simulated Annealing. Science **220** (4598), 671–680.

Lagos S.R., Sabbione J.I. and Velis D.R. 2014. Very Fast Simulated Annealing and Particle Swarm Optimisation for Microseismic Event Location. 84th SEG meeting, Denver, USA, Expanded Abstracts, 2188–2192. doi: 10.1190/segam2014-1216.1.

Landa E., Beydoun W. and Tarantola A. 1989. Reference Velocity Model Estimation from Prestack Waveforms: Coherency Optimisation by Simulated Annealing. Geophysics **54**(8), 984-990. DOI: 10.1190/1.1442741

Li-Ping Z., Huan-Jun Y. and Shang-Xu H. 2005. Optimal choice of parameters for particle swarm optimization. Journal of Zhejiang University Science A **6**(6), 528-534. DOI: 10.1007/BF02841760

Li T., and Mallick, S. 2015. Multicomponent, multi-azimuth pre-stack seismic waveform inversion for azimuthally anisotropic media using a parallel and computationally efficient non-dominated sorting genetic algorithm. Geophysical Journal International **200**(2), 1134-1152. DOI: 10.1093/gji/ggu445

Mallick, S. 1999. Some practical aspects of prestack waveform inversion using a genetic algorithm: An example from the east Texas Woodbine gas sand. Geophysics **64** (2), 326-336. DOI: 10.1190/1.1444538

Mallick S. and Dutta N. C. 2002. Shallow water flow prediction using prestack waveform inversion of conventional 3D seismic data and rock modeling. The Leading Edge **21**(7), 675-680. DOI: 10.1190/1.1497323

Marson-Pidgeon K., Kennett B.L.N. and Sambridge M. 2000. Source Depth and Mechanism Inversion at Teleseismic Distances Using a Neighborhood Algorithm. Bulletin of the Seismological Society of America **90** (6), 1369–1383. DOI: 10.1785/0120000020

Mellman G.R. 1980. A Method of Body-Wave Waveform Inversion for the Determination of Earth Structure. Geophysical Journal of the Royal Astronomical Society **62**, 481–504. DOI: 10.1111/j.1365-246X.1980.tb02587.x

Mills J.M. and Fitch T.J. 1977. Thrust Faulting and Crust-upper Mantle Structure in East Australia. Geophysical Journal International **48** (3), 351–384. DOI: 10.1111/j.1365-246X.1977.tb03677.x

Mitchell M. 1998. An introduction to genetic algorithms. MIT press.

Nolte B. and Frazer L.N. 1994. Vertical Seismic Profile Inversion with Genetic Algorithms. Geophysical Journal International **117** (1), 162–178. DOI: 10.1111/j.1365-246X.1994.tb03310.x

Padhi, A., and Mallick S. 2014. Multicomponent Pre-Stack Seismic Waveform Inversion in Transversely Isotropic Media Using a Non-Dominated Sorting Genetic Algorithm. Geophysical Journal International **196** (3), 1600–1618. DOI: 10.1093/gji/ggt460

Pei D., Louie J.N. and Pullammanappallil S.K. (2007). Application of simulated annealing inversion on high-frequency fundamental-mode Rayleigh wave dispersion curves. Geophysics **72**(5), R77-R85. DOI: 10.1190/1.2752529

Press F. 1968. Earth Models Obtained by Monte Carlo Inversion. Journal of Geophysical Research **73** (16), 5223–5234. DOI: 10.1029/JB073i016p05223

Robinson J. and Rahmat-Samii Y. 2004. Particle Swarm Optimization in Electromagnetics. IEEE Transactions on Antennas and Propagation **52**(2), 397-407. DOI: 10.1109/TAP.2004.823969

Rothman D.H. 1985. Nonlinear inversion, statistical mechanics, and residual statics estimation. Geophysics **50**(12), 2784-2796. DOI: 10.1190/1.1441899

Rothman D.H. 1986. Automatic estimation of large residual statics corrections. Geophysics **51** (2), 332-346. DOI: 10.1190/1.1442092

Ryden N. and Park C.B. 2006. Fast Simulated Annealing Inversion of Surface Waves on Pavement Using Phase-Velocity Spectra. Geophysics **71**(4), R49–R58. DOI: 10.1190/1.2204964

Sajeva A., Aleardi M., Mazzotti A., Bienati N. and Stucchi E. 2014. Estimation of velocity macro-models using stochastic full-waveform inversion. 84th SEG meeting, Denver, USA, Expanded Abstracts, 1227-1231. doi: 10.1190/segam2014-1088.1.

Sajeva A., Aleardi M., Stucchi E., Bienati N. and Mazzotti A. 2016a. Estimation of acoustic macro-models using genetic full-waveform inversion: applications to the Marmousi model. Geophysics **81** (4), 1-12. doi: 10.1190/geo2015-0198.1.

Sajeva A., Aleardi M. and Mazzotti A. 2016b. Combining Genetic Algorithms, Gibbs Sampler, and Gradient-based Inversion to Estimate Uncertainty in 2D FWI. 78th EAGE Conference and Exhibition, Vienna, Austria. doi: 10.3997/2214-4609.201601543.

Sambridge M. 1999a. Geophysical inversion with a neighbourhood algorithm—I. Searching a parameter space. Geophysical Journal International **138** (2), 479-494. DOI: 10.1046/j.1365-246X.1999.00876.x

Sambridge M. 1999b. Geophysical inversion with a neighbourhood algorithm—II. Appraising the ensemble. Geophysical Journal International **138** (3), 727-746. DOI: 10.1046/j.1365-246x.1999.00900.x

Scales J. A., Smith M. L. and Fischer T. L. 1992. Global Optimisation Methods for Multimodal Inverse Problems. Journal of Computational Physics **103** (2), 258–268.

Schlierkamp-Voosen D. and Mühlenbein H. 1993. Predictive models for the breeder genetic algorithm. Evolutionary Computation **1**(1), 25–49.

Sen, M. K. and Stoffa P. L. 1991. Nonlinear one-dimensional seismic waveform inversion using simulated annealing. Geophysics **56** (10), 1624-1638.

Sen, M.K. and Stoffa P.L. 1992. Rapid sampling of model space using genetic algorithms: examples from seismic waveform inversion. Geophysical Journal International **108** (1), 281-292.

Sen M.K. and Stoffa P.L. 2013. Global optimisation methods in geophysical inversion. Cambridge University Press.

Sivanandam S.N. and Deepa S.N. 2008. Genetic Algorithm Optimisation Problems. Springer Berlin Heidelberg.

Shaw R., and Srivastava S. 2007. Particle Swarm Optimisation: A New Tool to Invert Geophysical Data. Geophysics **72**(2), F75–F83.

Stoffa P.L. and Sen M.K. 1991. Nonlinear multiparameter optimization using genetic algorithms: Inversion of plane-wave seismograms. Geophysics **56**(11), 1794-1810.

Voronoi, G. 1908. Nouvelles applications des paramètres continus à la théorie des formes quadratiques. Premier mémoire. Sur quelques propriétés des formes quadratiques positives parfaites. Journal für die reine und angewandte Mathematik **133**, 97-178.

Wathelet M., Jongmans D. and Ohrnberger M. 2004. Surface-wave inversion using a direct search algorithm and its application to ambient vibration measurements. Near Surface Geophysics **2** (4), 211-221.

Wiggins R.A. 1972. The General Linear Inverse Problem: Implication of Surface Waves and Free Oscillations for Earth Structure. Reviews of Geophysics and Space Physics **10** (1), 251–285.