J Glob Optim (2011) 49:149–172
DOI 10.1007/s10898-010-9540-x

Study of multiscale global optimization based on parameter space partition

Weitao Sun · Yuan Dong

Received: 1 November 2006 / Accepted: 23 February 2010 / Published online: 8 March 2010

Abstract Inverse problems in geophysics are usually described as data misfit minimization problems, which are difficult to solve because of various mathematical features, such as multi-parameters, nonlinearity and ill-posedness. Local optimization based on function gradient can not guarantee to find out globally optimal solutions, unless a starting point is sufficiently close to the solution. Some global optimization methods based on stochastic searching mechanisms converge in the limit to a globally optimal solution with probability 1. However, finding the global optimum of a complex function is still a great challenge and practically impossible for some problems so far. This work develops a multiscale deterministic global optimization method which divides definition space into sub-domains. Each of these sub-domains contains the same local optimal solution. Local optimization methods and attraction field searching algorithms are combined to determine the attraction basin near the local solution at different function smoothness scales. With Multiscale Parameter Space Partition method, all attraction fields are to be determined after finite steps of parameter space partition, which can prevent redundant searching near the known local solutions. Numerical examples demonstrate the efficiency, global searching ability and stability of this method.

Keywords Inverse problem · Multiscale parameter space partition · Multi-grid seed growth · Deterministic global optimization

1 Introduction

Inverse problem for wave equations in complex media have been a focus of geophysics research for the last 30 years. The aim of inverse problems is to estimate parameters of the inner earth's conformation and material properties based on data measured on the surface.

W. Sun (✉)
ZHOU PEI-YUAN Center for Applied Mathematics, Tsinghua University, 100084 Beijing, China
e-mail: sunwt@tsinghua.edu.cn

Y. Dong
Department of Computer Science and Technology, Tsinghua University, 100084 Beijing, China

Let the true model be denoted by m and the data by d. From the data d one reconstructs an estimated model $\hat{m}$. This is called the estimation problem. The estimation problem is usually solved by fitting the model to the data. Let the ith datum d_i be related to the model m through the relation $d_i = G_i(m)$, where $G_i(m)$ is a nonlinear functional that maps the datum d_i. The data fitting can be achieved by minimizing the difference between d_i and the estimated data $G_i(\hat{m})$ as a function of the estimated model $\hat{m}$. In the simplest way, one can minimize the least squares error $S(\hat{m}) = \sum_i (d_i - G_i(\hat{m}))^2$ as a function of the estimated model $\hat{m}$. Descent methods, such as the steepest descent method, Newton method, Gauss–Newton method, etc., can be used to minimize the error function by moving 'downhill' in one way or another. However, the error function usually contains several minima for nonlinear inverse problems. A descent method may become trapped in local minima and lead to local model estimation, rather than the global optimal solution. Consequently, global searching approaches have been developed for many types of optimization problems [1,2].

Local optimization techniques can be applied to weakly nonlinear problems or some highly nonlinear problems with a good enough guess of the solution in advance. As the nonlinearity of inverse problem increases, the objective function can become more complex, including narrow valleys, non-smooth gradient changes and multiple minima [3,4]. As research of the inverse problems has progressed and intensified, non-deterministic optimization theories based on statistics and heuristic mechanisms have developed rapidly and show great potential. Among the existing methods, there are several well-studied and widely used approaches, including Simulated Annealing (SA), Genetic Algorithm (GA), Neural Network (NN), Chaotic optimization, etc. Rothman [5,6] introduced the SA method [7] into geophysics, which is a stochastic direct search method designed primarily for global optimization problems. More recently, GA have made the jump from their origins in the computer science literature [8,9] to geophysical problems [10–12]. This type of approach has found many applications in recent years [13,14]. Compared with local optimization techniques based on function gradient (such as steepest descent, conjugate gradients, Newton–Raphson, etc. [15]), global optimization methods based on stochastic search (such as GA, SA etc) do not need derivative or gradient information. Optimization problems are solved directly through global algorithms such as stochastic searching and the consequent mapping from data set space to model space. Global optimization (GO) methods only use evaluations of the objective function at points in parameter space, which leads to good qualities of global convergence. Some theoretical proofs suggest that GO algorithms may terminate with the global optimal solution in probability 1 [16,17]. However, GO algorithms, such as SA and GA, typically require an infinite sequence of sample points to guarantee theoretical convergence [16,18–22]. Convergence to global optima can only be realized in asymptotic sense and is only achieved when algorithm proceeds to infinite number of iterations. In many cases, stochastic GO algorithms give a good though not necessarily optimal solution within a reasonable computing time. The time required to ensure a significant probability of success will usually exceed the time required for a complete search of the solution space. Actually, the failure of the random-natured algorithms to correctly find the global optimal solution can often be seen in the literature of geophysical inversion [3,23,24].

Since the late 1990s, with the help of thorough study on different inverse methods, researchers began to focus on hybrid optimization methods. Cary and Chapman [25] proposed to use the Monte Carlo method to locate a globally approximated point and then use the local optimizer to find the optimal solution near the Monte Carlo solution. Gerstoft [26] suggested incorporating the Gauss-Newton method into GA in order to improve every member of the new generation. Fallat and Dosso [23], Liu and Hartzell [24] started with a simplex and then used SA algorithm to determine the direction and step of searching. Generally

speaking, there are two categories of hybrid method. In the first type of hybrid methods, combination of nonlinear methods with linear methods has been studied for differentiable objective functions [27,28]. When the objective functions are not differentiable, the second type of optimization methods combine nonlinear methods with direct optimization methods [3,24,29]. Although intensive computations are still not avoidable in hybrid methods, recent developments [30–35] show great potential to improve optimization efficiency. The key in hybrid optimization methods is how to propose general principles governing linear-nonlinear combination and how to propose rules for theoretical evaluation of hybrid GO methods.

GO methods are often based on the idea of transforming the function $f(x)$ into a new cost function [36–39]. Another essential idea is successive partitioning of the definition domain by direct use of interval arithmetic mathematics [40–46]. These kinds of deterministic global optimization algorithms have developed rapidly since late 1970s. The branch and bound method, a well studied deterministic GO method, shows great potentials in solving multi-extremum problems, especially in discrete and combinatorial optimization [47–49]. With the help of objective function structure characteristics, deterministic global optimizations aim to prevent redundant optimization in many sub-spaces, which become the leading edge of optimization method [50–65] and inverse problem [66–70].

Based on the relation between local extremum and neighboring sub-space, this study developed a multiscale deterministic GO method, which tries to provide an efficiently global optimization by partition of definition space at multi-scale levels. First, a local optimum is found by the conventional local optimization algorithm. The corresponding neighborhood, which contains less favorable models, are determined and eliminated from the whole parameter space by Attraction Field Searching (AFS) algorithm. Then, new starting points are distributed in the remaining parameter space to search for new local optima and their neighborhood. As the definition space was totally divided, the global optimum is expected to be estimated from the known local optima.

The rest of this paper is organized as follows. A Multi-Grid Seed-Growth (MGSG) algorithm for searching attraction basins is described in Sect. 2. Then, Sect. 3 is devoted to the formulation of Multiscale Parameter Space Partition method (MPSP). Numerical examples and discussions are given in Sect. 4. Final section presents a summary of our conclusions.

2 Multi-grid seed-growth algorithm for attraction field searching

2.1 Problem definition

Conventional local optimization methods have been very successful in searching convex functions. However, they cannot escape from valleys of local minima since the convex assumption does not hold for objective functions with a large number of local optima. Conventional optimization methods neglect the value-increasing directions along the function surface. In order to determine the attraction field of local minima accurately and efficiently, we propose a Multi-Grid Seed-Growth (MGSG) searching algorithm as described in following paragraphs.

Consider the global optimization problem: given a bounded set $\mathbf{D}$ in a n-dimensional real Euclidean space $\mathbf{R}^n$ and a continuous function $f : \mathbf{D} \to \mathbf{R}$, find

$$\min f(\mathbf{x}), \quad s.t. \quad \mathbf{x} \in \mathbf{D} \subset \mathbf{R}^n, \tag{1}$$

Denote the set of solutions by $\overline{\mathbf{X}}^* \subset \mathbf{D}$. For an arbitrary optimization solution $\mathbf{x}^* \in \overline{\mathbf{X}}^*$, $\mathbf{z}^* = f(\mathbf{x}^*)$ denote the optimum value. Here $\mathbf{R}^n$ is the n-dimensional real number space. The solution set $\overline{\mathbf{X}}^*$ is supposed to be countable throughout this paper.

Definition 1 Let $\mathbf{x}_i^* \in \overline{\mathbf{X}}^* \subset \mathbf{D}(i \in I)$ is a local optimum with a corresponding extremum value $\mathbf{z}_i^* = f(\mathbf{x}_i^*)$. The solution set $\mathbf{N}_i \subseteq \mathbf{D}$ is defined as the neighborhood of $\mathbf{x}_i^*$, written

$$\mathbf{N}_i = \mathbf{N}_i(\mathbf{x}_i^*)l, \tag{2}$$

if $\mathbf{N}_i(\mathbf{x}_i^*) \cap \left(\overline{\mathbf{X}}^* \backslash \mathbf{x}_i^*\right) = 0$ and there is no solution

$$\mathbf{x} \in \mathbf{N}_i(\mathbf{x}_i^*) \subseteq \mathbf{D}, \tag{3}$$

such that

$$f(\mathbf{x}) \leq f(\mathbf{x}_i^*). \tag{4}$$

Here $\overline{\mathbf{X}}^* \backslash \mathbf{x}_i^*$ indicates the solution set without $\mathbf{x}_i^*$.

The neighborhood domain $\mathbf{N}_i = \mathbf{N}_i(\mathbf{x}_i^*) \subseteq \mathbf{N}$ satisfy non-overlap condition:

$$\mathbf{N} = \underset{i \in I}{\cup} \mathbf{N}_i, \tag{5}$$

$$\mathbf{N}_i \cap \mathbf{N} = \mathbf{N}_i, \tag{6}$$

$$\mathbf{N}_i \cap \mathbf{N}_j = \partial(\mathbf{N}_i) \cap \partial(\mathbf{N}_j), \quad \text{for } i, j \in I, i \neq j. \tag{7}$$

Here $\mathbf{N}$ is a collection of sets $\{\mathbf{N}_i : i \in I\}$ and I is a finite set of indices. $\partial(\mathbf{N}_i)$ denotes the relative boundary of $\mathbf{N}_i$.

Definition 2 Given a neighborhood $\mathbf{N}_i$, a local optimization operator L is represented by a solution $\mathbf{x}_i$ from $\mathbf{N}_i$ and the corresponding function value $\mathbf{z}_i = f(\mathbf{x}_i)$. The local optimization operator produces an optimum solution pair $\{\mathbf{x}_i^*, \mathbf{z}_i^* = f(\mathbf{x}_i^*)\}$ for $\mathbf{N}_i$.

$$\{\mathbf{x}_i^*, \mathbf{z}_i^*\} := L\{\mathbf{x}_i, \mathbf{z}_i\}, \quad i \in I \tag{8}$$

In order to determine attraction field of local optimum, the whole parameter space is divided into a grid system. For box-constrained optimization problems, the MGSG algorithm uses interval notation $[\mathbf{u}, \mathbf{v}]$ for rectangular grid system. Here $\mathbf{u}$ and $\mathbf{v}$ are n-dimensional vectors with $u_k < v_k$ for $k = 1, 2, \ldots, n$.

$$[\mathbf{u}, \mathbf{v}] := \{\mathbf{x} \in \mathbf{R}^n | u_k \leq x_k^1 < \cdots < x_k^{g_k} < \cdots < x_k^{G_k} \leq v_k, k = 1, 2\}, \quad G_k \geq 2. \tag{9}$$

Here $x_k^{g_i}$ is the center point coordinate value of grid g_k at kth dimension. The total grid number at kth dimension is G_k. Whenever a grid is split along each coordinates axis, 2^n sub-grids are created. The grids with initial size are called zero-level grids. When a grid of level s is split, the sub-grids with smaller size are called $s + 1$ level grids.

If the local optimization operator satisfies,

$$L\{\mathbf{x}^{g_k-1}, \mathbf{z}^{g_k-1}\} \neq L\{\mathbf{x}^{g_k}, \mathbf{z}^{g_k}\} = L\{\mathbf{x}^{g_k+1}, \mathbf{z}^{g_k+1}\} = \cdots = L\{\mathbf{x}^{g_k+m}, \mathbf{z}^{g_k+m}\}, \quad m > 1 \tag{10}$$

the coordinates of solution $\mathbf{x}^{g_k-1}$ and $\mathbf{x}^{g_k}$ define the boundary between two different neighborhoods of local optimum. Here $\mathbf{x}^{g_k-1}$ in Eq. 10 can be called the low neighborhood boundary of the optimum pair $L\{\mathbf{x}^{g_k}, \mathbf{z}^{g_k}\}$. The up boundary grid can be determined in the same way.

The irregular neighborhood boundary $\partial(\mathbf{N}_i)$ is determined by simulating seed growth from an initial zero-level grid point at $\mathbf{X}_{iopt}^0$ to outside domain. The capital letter $\mathbf{X}$ represents a grid and lowercase letter $\mathbf{x}$ represents the coordinate vector. Since local optimization methods converge to a deterministic local optimum (set), the neighborhood boundary can

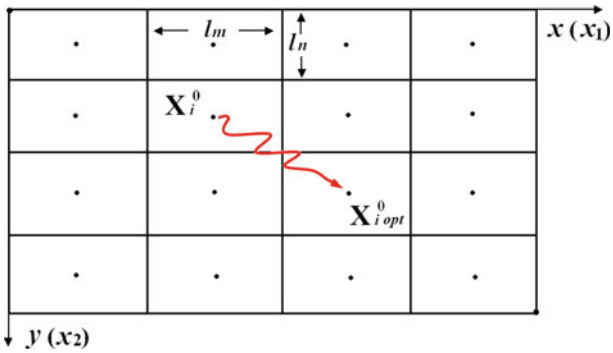

Fig. 1 The initialization step of MGSG algorithm. The *curved arrow* indicates the convergence of local optimization from initial solution $\mathbf{X}_i^0$ to local optimum $\mathbf{X}_{iopt}^0$

be located by finding the inflection points. By splitting the boundary grid into smaller grids, detailed boundary shape can be determined and the optimization accuracy increases. A recursive boundary searching is performed within boundary grids. Intensive computation is only carried out in the split boundary grids.

An analytic approach based on gradient information is much more efficient in finding a stationary point. From a point in the attraction basin of an optimum, local optimization method leads to quick convergence to this optimum. Taking the local optimum as a "seed", grids are searched outwards in each direction. Taking each searched grid as the initial solution, if a local optimization converges to "seed", the grid is inner part of the attraction basin. Otherwise, this grid is considered as boundary. As a result, a local optimization algorithm is especially useful to determine the local optimum neighborhood (attraction basin) boundary and divide the parameter space. Each subspace contains a known local optimum (set) and its neighborhood, which prevents taking solutions in the determined neighborhood in next iteration. As the definition space is totally divided, the actual global optima are to be determined. This conceptually straightforward method analyzes the subdomain distributions in feasible space instead of merely seeking global optimal solution, which shows great potential in searching a parameter space.

2.2 Algorithm

An elaborate outline of neighborhood boundary requires a complete, exhaustive search in small grids in whole definition space. The MGSG method only split intersection grid recursively on the neighborhood boundary, which forms a local refinement. For the sake of clear illustration, the algorithm is demonstrated step by step for 2 dimensional problems.

(1) Initialization. Given a bounded definition space, divide it into $m \times n$ grid system with grid lengths l_m, l_n in two dimensions. Center points with superscript 0 denote the grids at the initial zero-level (Fig. 1).

(2) Local optimization. Starting at a randomly distributed start grid $\mathbf{X}_i^0 \in \mathbf{D}$ in unsearched domain, compute the local optimum $\mathbf{X}_{iopt}^0$ by conventional local optimization algorithm.

(3) Primary seed-growth search. Taking $\mathbf{X}_{iopt}^0$ as a seed-growth point, attraction field boundary points are searched in x(or x_1) and y(or x_2) directions.

  (3.1) Search the negative direction in x axis for boundary point. Grids at left side of $\mathbf{X}_{iopt}^0$ are taken as the starting solution one by one for local optimization. If the

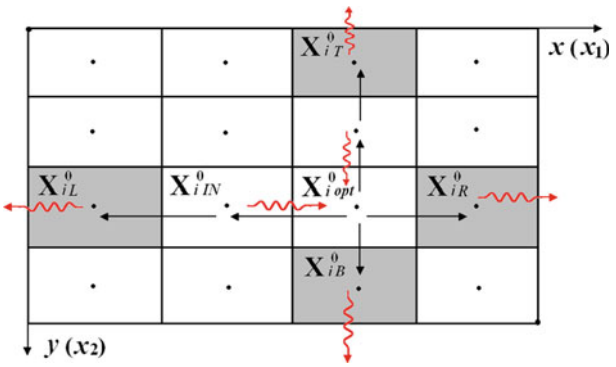

Fig. 2 The boundary searching step of MGSG algorithm. The *curved arrows* indicate the convergence of local optimization from a grid center to local optimum $\mathbf{X}^0_{iopt}$ (the case for inner attraction field grid) or to other optimum outside attraction field of $\mathbf{X}^0_{iopt}$ (the case for boundary grid)

local optimization converges to $\mathbf{X}^0_{iopt}$, this grid is considered as an inner neighborhood point of $\mathbf{X}^{opt}_i$ and named as $\mathbf{X}^0_{iIN}$ (Fig. 2). This inner neighborhood point is then pushed into a stack as a new seed for later boundary search process. Otherwise, this grid is the left boundary of $\mathbf{X}^0_{iopt}$ neighborhood and named as $\mathbf{X}^0_{iL}$. The right boundary point $\mathbf{X}^0_{iR}$ in positive x direction is found in the same way.

(3.2) Search the negative and positive y coordinate for top and bottom boundary grids in the way described in (3.1).

(4) Grid subdivision. The boundary grids on each dimension are split into smaller boxes (Fig. 3). The primary seed-growth search is performed in the higher-level grids (smaller grid size) to find the boundary grids with more reliability, such as X^1_{iL}. Grid subdivision will continue until a predetermined maximal level $s_{\max}$ is reached. Flags are set not only for local optimum grid, but also for neighborhood boundary grids and the inner neighborhood points.

(5) Recursive seed-growth search. After the primary seed growth search and grid split process, pop a new seed from the stack and repeat the search step 3 and 4 until there are no seeds in the stack. Then the local minimum $\mathbf{x}^*_i$ and its attraction neighborhood $\mathbf{N}_i$ are reduced from solution space $\mathbf{D}$.

After an iteration of local optimization and neighborhood boundary search related to a certain starting point, the determined neighborhood $\mathbf{N}_i$ will become an irregular sub-domain in the solution space. Eliminating $\mathbf{N}_i$ from the total solution space can prevent redundant local optimization in known attraction field and cuts down the computation costs (Fig. 4).

3 Multiscale parameter space partition method

The critical step in studying the objective function landscape properties with the PSP method is to determine attraction basins of local minima in parameter space. We have already developed an AFS algorithm by lattice sampling to determine the attraction field boundaries, and have combined it with the multi-grid method to study more detailed subdomain boundary

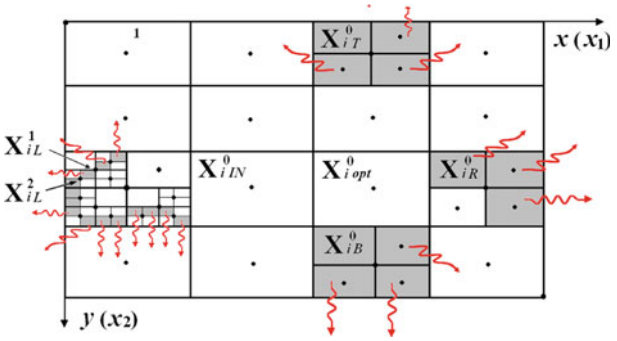

Fig. 3 The multi-grid boundary searching step of MGSG algorithm. The subscript of $\mathbf{X}^1_{iL}$ and $\mathbf{X}^2_{iL}$ indicate that left boundary grid centers at level 1 and level 2. For the case of grid split at level n, each grid is divided into 2^n sub-grids along coordinate axis

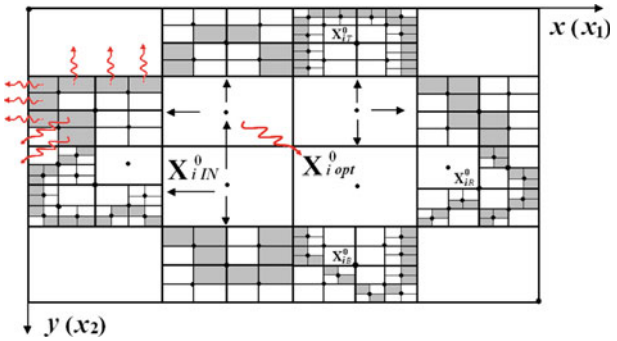

Fig. 4 The 2D MGSG algorithm for local optimum neighborhood determination. 4×4 grids division scheme is used for illustration. Seed grows from an initial zero-level grid point at $\mathbf{X}^0_{iopt}$ to left, right, top and bottom. Boundary grids in each direction are divided into sub-grids for enhanced searching of boundary grid. Grey grids are final traction field boundary

shapes. Multiscale PSP method is developed to determine the attraction field boundaries in parameter space subject to complex function landscapes. Then multiscale global optimization is studied.

3.1 Multiscale representation of objective functions

The global minimum of a unimodal function, which has only one local minimum, can be detected by a single run of local optimization. Highly multimodal functions provide more challenges and need a GO algorithm. The number of local optima is a critical issue for GO problems. However, the difficulty of a GO problem not only depends on the local optima number but also on how the local optima distribute in the search space. We can reasonably expect that strange landscapes, such as narrow attraction basin, make it a more difficult task to find global optimal solutions.

In order to overcome the difficulties arising from complex function landscapes, the objective function can be transformed into multiscale spaces by wavelet decomposition operator

[71–74]. The approximation of $f(x)$ at the resolution 2^j is the orthogonal projection $\mathbf{P}_{\mathbf{V_j}} f(x)$ on to a particular vector space $\mathbf{V}_j (\mathbf{V}_j \subset \mathbf{L}^2(\mathbf{R}))$ and thus given by

$$\mathbf{P}_{\mathbf{V}_j} f(x) = \sum_{n=-\infty}^{+\infty} < f, \phi_{j,n} > \phi_{j,n}(x) \tag{11}$$

where $<,>$ denotes the standard inner product in $\mathbf{L}^2(\mathbf{R})$. Here $\mathbf{R}$ denotes the set of real number. $\mathbf{L}^2(\mathbf{R})$ denotes the vector space of measurable square-integrable one-dimensional functions $f(x)$. The scaling function $\phi(x) \in \mathbf{L}^2(\mathbf{R})$ satisfies $\phi_j(x) = 2^j \phi(2^j x)$ for $j \in \mathbf{Z}$. Here $\mathbf{Z}$ is the set of integer number. $\phi_{j,n}(x)_{n\in\mathbf{Z}}$ is orthonormal basis of $\mathbf{V}_j$ denoted by scaling function $\phi(x)$ [74]:

$$\phi_{j,n}(x) = \sqrt{2^{-j}} \phi_j(x - 2^{-j} n), \quad (n, j) \in \mathbf{Z}^2 \tag{12}$$

The coefficient of $\sqrt{2^{-j}}$ in above formula normalizes the functions in the $\mathbf{L}^2(\mathbf{R})$ norm. Let $f(x) \in \mathbf{L}^2(\mathbf{R})$. The approximation of function $f(x)$ at resolution 2^j converge to the original function as the resolution scale j increase to $+\infty$. In $\mathbf{L}^2(\mathbf{R})$ space, a multi-resolution approximation is a sequence of closed sub-spaces $(\mathbf{V}_j)_{j\in\mathbf{Z}}$ [37,38]. The set of vector space $(\mathbf{V}_j)_{j\in\mathbf{Z}}$ is called a multiresolution approximation of $\mathbf{L}^2(\mathbf{R})$ if a series of properties are satisfied [74].

Since function $f(x)$ is at a resolution $\mathbf{V}_j \subset \mathbf{V}_{j+1}$, each space $\mathbf{V}_j$ can be decomposed into

$$\mathbf{V}_{j+1} = \mathbf{V}_j \oplus \mathbf{W}_j \tag{13}$$

where $\mathbf{W}_j$ is the orthogonal complement of $\mathbf{V}_j$ in $\mathbf{V}_{j+1}$. Let us denote by $\mathbf{P}_{\mathbf{W_j}}$ the orthogonal projection onto $\mathbf{W}_j$,

$$\mathbf{P}_{\mathbf{W}_j} f(x) = \sum_{n=-\infty}^{+\infty} \langle f, \psi_{j,n} \rangle \psi_{j,n}(x) \tag{14}$$

The function details, which is the difference between two approximations at resolution 2^j and 2^{j+1}, is obtained by decomposition of the function in a wavelet orthogonal basis $\psi_{j,n}(x)_{n\in\mathbf{Z}}$ [74].

$$\psi_{j,n}(x) = \sqrt{2^{-j}} \psi_j \left(x - 2^{-j} n\right), \quad (n, j) \in \mathbf{Z}^2 \tag{15}$$

$\psi_{j,n}(x)_{n\in\mathbf{Z}}$ is orthonormal basis of $\mathbf{W}_j$ and is denote by wavelet function $\psi(x)$. The Fourier transform of wavelet function $\psi(x)$ is given by:

$$\hat{\psi}(\omega) = G\left(\frac{\omega}{2}\right) \hat{\phi}\left(\frac{\omega}{2}\right) \tag{16}$$

$$G(\omega) = e^{-i\omega} \overline{H(\omega + \pi)} \tag{17}$$

The existence of wavelet $\psi(x)$, whose translation and dilation $\psi_j(x) = \sqrt{2^j} \psi(2^j x) (j \in \mathbf{Z})$ is an orthonormal basis in $\mathbf{L}^2(\mathbf{R})$, has been shown in [75]. In order to construct a wavelet, a proper definition of function $H(\omega)$ is needed. A well-defined $H(\omega)$ can produce a wavelet function $\psi(x)$ with good localization both in the spatial and Fourier domains [74–77].

From above formula, wavelet decomposition of function $f(x)$ at 2^{j+1} resolution scale is:

$$\mathbf{P}_{\mathbf{V}_{j+1}} f(x) = \mathbf{P}_{\mathbf{V}_j} f(x) + \mathbf{P}_{\mathbf{W}_j} f(x) \tag{18}$$

In numerical computations, resolution is limited by memory constraints and computation times. Instead of working with function $f(x)$, we consider its approximation up to a given

resolution 2^I. The function on scale i can be decomposed into the detail space $\mathbf{W}_j$ from $j(j < I)$ downward to resolution scale 2^J:

$$\mathbf{P}_{\mathbf{V}_I} f(x) = \mathbf{P}_{\mathbf{V}_J} f(x) + \sum_{j=J}^{I-1} \mathbf{P}_{\mathbf{W}_j} f(x) \tag{19}$$

Here a lower index j corresponds to a poorer (coarser) resolution.

3.2 Multiscale parameter space partition method

Let $\underset{\mathbf{S}}{L}(f(x))$ denote the Parameter Space Partition (PSP) process on $f(x)$ in definition space $\mathbf{S}$. The optimized solution $\mathbf{X}^{\min}$ and its attraction filed $\mathbf{S}^*$ are:

$$\{\mathbf{X}^{\min}, \mathbf{S}^*\} = \underset{\mathbf{S}}{L}(f(x)) \tag{20}$$

From Eq. 19 we have the Multiscale Parameter Space Partition (MPSP) optimization formula:

$$\underset{S}{L}\left(\mathbf{P}_{\mathbf{V}_I} f(x)\right) = \underset{S}{L}\left(\mathbf{P}_{\mathbf{V}_J} f(x) + \sum_{j=J}^{I-1} \mathbf{P}_{\mathbf{W}_j} f(x)\right) \tag{21}$$

First, the PSP optimization method is used at the J level, which is the coarsest scale. The optimized solution and its attraction field for scale 2^J are:

$$\{\mathbf{X}_J^{\min}, \mathbf{S}_J^*\} = \underset{S}{L}\left(\mathbf{P}_{\mathbf{V}_J} f(x)\right) \tag{22}$$

Then, the objective function is optimized in the attraction field $\mathbf{S}_J^*$ for approximation part on scale 2^{J+1}. The optimized solution and its attraction field for 2^{J+1} are:

$$\{\mathbf{X}_{J+1}^{\min}, \mathbf{S}_{J+1}^*\} = \underset{S_J^*}{L}\left(\mathbf{P}_{\mathbf{V}_{J+1}} f(x)\right) \tag{23}$$

The PSP method is used to optimize approximation projection $\mathbf{P}_{\mathbf{V}} f(x)$ at different scales sequentially until up to the original scale. The global optimization solution is found as:

$$\{\mathbf{X}_I^{\min}, \mathbf{S}_I^*\} = \underset{S_{I-1}^*}{L}\left(\mathbf{P}_{\mathbf{V}_I} f(x)\right) \tag{24}$$

The MPSP optimization method has two obvious advantages:

(1) The function surface is smoother on coarse scale, which leads to less local solutions. PSP optimization method can work very well at coarse scales.
(2) As up to the original scale, the multiscale function approximation approaches to the original function. In addition, the optimization on more detailed scale is only performed in a very limited space determined by the optimization on coarse scale, which saves a lot of computation cost.

4 Numerical example of parameter space partition method

Many constrained/unconstrained benchmark functions and test function construction methods have been proposed to verify validity of GO methods [78–87]. GO algorithm performance comparisons have been carried out to standard test functions [88,89], including Rastrigin's function, Weierstrass Function, Schwefel's function [90] and Ackley's Path function [91],

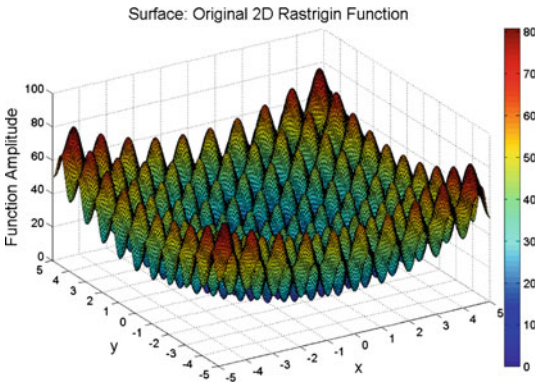

Fig. 5 The 2-D Rastrigin's function surface

etc. In addition, with a priori known local minima and their regions of attraction, differentiable and non-differentiable test functions can be constructed in a flexible way [84,85]. A method for generating box-constrained nonlinear programming test functions with known characteristics is also developed in [83].

In current study, the newly proposed deterministic global optimization method is applied to some standard box-constrained functions first. These functions have been used to test many classes of methods for solving optimization problems [89,92,93]. The algorithm performance in such continuous global minimization problems is compared with other non-deterministic GO method, such as SA. Then the MPSP method is verified in an inverse problem of 3-layer acoustic wave propagation model.

4.1 Two dimensional Rastrigin's function

Although a much more difficult example can be used to test the limits of the MPSP algorithm, it is more important to illustrate algorithm accuracy and efficiency by a relatively small typical example. In the first example section, MPSP method is applied to Rastrigin's function. Rastrigin's function is highly multimodal. Although the locations of the minima are regularly distributed, this example provides clear elucidation and intuitive understanding for how the MPSP algorithm works.

$$Rastrigin_n(x) = \sum_{i=1}^{n} \left[x_i^2 - 10\cos(2\pi x_i)\right] + 10n, \quad x_i \in [-5, 5] \tag{25}$$

The Rastrigin's function for case of $n = 2$ is shown in Fig. 5. The number of local minima increase exponentially with dimension number n. Local optima distribution is so dense in the x–y plane that the chance to be trapped in local optima increases greatly for local optimization method. It is nearly impossible to find out the global optimal solution by gradient-based algorithms, unless the initial solution is located in the basin of the global optimum.

The 2-D Rastrigin's function is discretized as a 255×255 lattice system in two-dimensional domain $[-5, 5] \times [-5, 5]$. The MPSP algorithm is carried out in reconstructed function surfaces based on approximation coefficients at level 0, 1 and 5. Daubechies 6 wavelet is used in multiscale analysis. Since the reconstructed surface at approximation level 0 is unimodal, the only optimal solution and its boundary is easy to find out. The computation parameters and results are listed in Table 1 (Fig. 6).

Table 1 The MPSP optimization results for 2-D Rastrigin's function

SL	DR	GS	MGS	GM	FV	TGM	TFV	EN
0	[−5.0, 5.0]×[−5.0, 5.0]	2.5	2.5	[−3.728897e-2, 4.193115e-4]	19.61723	(1.9608e-2, 1.9608e-2)	19.615144	575
1	[−2.518644, 2.481356]×[−2.499790, 2.500210]	0.5	0.5	[4.703217e-3, 4.325562e-3]	13.02914	(1.9608e-2, 1.9608e-2)	13.029144	2977
5	[−1.256971, 1.243029] ×[−1.247732, 1.250210]	0.25	0.2498	[4.703217e-3, 4.325562e-3]	0.1523587	(−1.960784e-2, −1.960784e-2), (−1.960784e-2, 1.960784e-2), (1.960784e-2, −1.960784e-2), (1.960784e-2, 1.960784e-2)	0.1523587	2551

SL scale level, *DR* definition range, *GS* grid size, *MGS* minimal grid size in boundary grid division, *GM* global minima *FV* function value of global minima, *TGM* true global minima of the discrete function data, *TFV* true global minima function value of the discrete function data at corresponding level, *EN*, function evaluation number

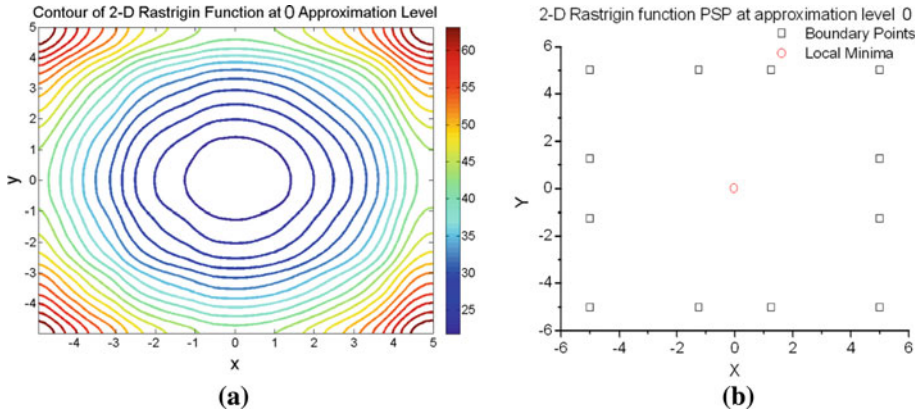

Fig. 6 **a** The reconstructed 2-D Rastrigin's function contour at approximation level 0. **b** 2-D Rastrigin's function PSP results at approximation level 0

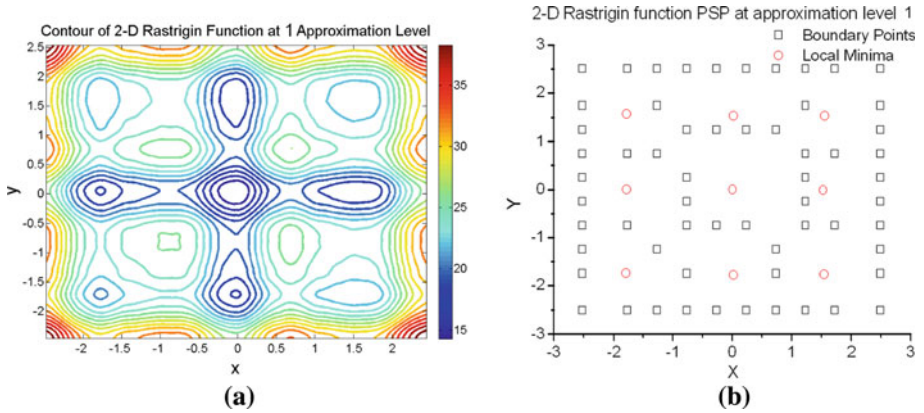

Fig. 7 **a** The reconstructed 2-D Rastrigin's function contour at approximation level 1. **b** 2-D Rastrigin's function PSP results at approximation level 1

In the sequential PSP optimization at approximation level 1, the definition domain shrinks to $[-2.479147, 2.520853] \times [-2.484032, 2.515968]$ with a center at the global minimum determined in approximation level 0. Although the parameter space is smaller, the reconstructed function surface at level 1 has more fluctuations than level 0, which costs more computations. The 9 local minima and their boundary points are shown in Fig. 7. Global optimum is found near the center of definition domain. The optimization results are shown in Table 1.

Since local minima distribution among the reconstructed surfaces at levels larger than 1 have much less variations, parameter space partition is performed directly at approximation level 5 (the original function level) to save computations. The partition $[-1.179147, 0.6159682] \times [-1.179147, 0.6159682]$ within attraction field boundary of global optimum at level 1 is taken as the new definition domain. In order to avoid the non-integer grid number, the searching step has a little adjustment to fit the new parameter space size. The local minima and their boundary points are shown in Fig. 8. The figures

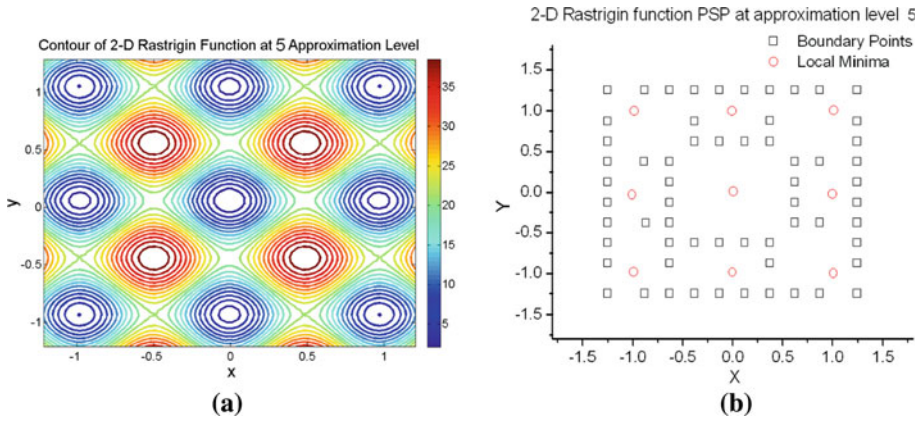

Fig. 8 **a** The reconstructed 2-D Rastrigin's function contour at approximation level 5. **b** 2-D Rastrigin's function PSP results at approximation level 5

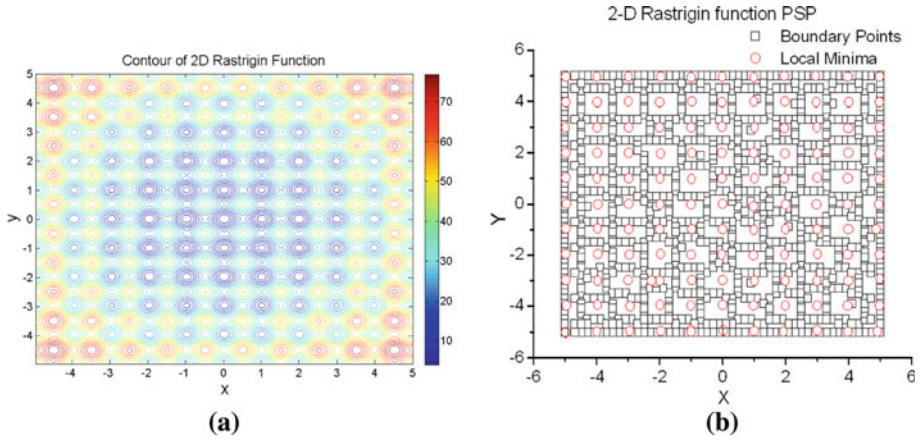

Fig. 9 **a** The original 2-D Rastrigin's function contour. **b** 2-D Rastrigin's function PSP results

suggest that although the function is a unimodal basin within current shrunk domain at level 1, there are more local minima emerged in a more detailed definition space of level 5.

The PSP algorithm is also used to solve optimization problem at the original 2-D Rastrigin's function surface. Figure 9 shows the parameter space partitions and the local minima distribution in the definition domain $[-5.0, 5.0] \times [-5.0, 5.0]$. High density peak distribution becomes a difficult task to improve partition efficiency. In comparison with the proposed multiscale parameter space partition method, the continuous 2-D Rastrigin's function is also optimized by simulated annealing (SA) method. The initial annealing temperature was 10 and the cooling factor was 0.5. The maximal iteration number at each annealing temperature was 10. The maximal inner loop bound at each temperature was 20. The maximal function evaluation number for SA was 1e5. The floating-point precision is 1.0e-6. The SA final temperature was 1.490116e-7. The computation parameters and results are listed in Table 2.

The computation results and comparisons show that single level parameter space partition methods have difficulties in optimizing functions with dense local minima. However,

Table 2 The single level PSP and simulated annealing (SA) optimization results for 2-D Rastrigin's function surface

OM	DR	GS	MGS	GM	FV	TGM	TFV	EN
PSP	$[-5.0, 5.0] \times [-5.0, 5.0]$	0.2	2.0e − 1	[5.468750e − 3, −3.125000e − 3]	1.523587e − 1	(−1.960784e − 2, 1.960784e − 2) (−1.960784e − 2, 1.960784e − 2) (1.960784e − 2, −1.960784e − 2) (1.960784e − 2, 1.960784e − 2)	0.152359	70989
SA	$[-5.0, 5.0] \times [-5.0, 5.0]$	1.0	−9.7e − 8	[9.520362e − 7, −9.692928e − 8]	1.816787e − 10	(0,0)	0	10801

OM optimization method, *DR* definition range, *GS* grid size. For *SA*, it is the original step length, *MGS* minimal grid size in boundary grid division. For SA, it is the final step length, *GM* global minima, *FV* function value of global minima, *TGM* true global minima of the discrete function data, *TFV* true global minima function value of the discrete function data at corresponding level, *EN* function evaluation number

MPSP method decomposes complex functions in a hierarchy of subspaces, which improve the computation efficiency. Although there will be many oscillated peaks at more detailed levels, this local minimum at coarse level provides a predetermination of the global minimum position. Given a shrunk search space near the predetermined position, the global optimization can be carried out more easily. Here a0 and a1 level approximations are used to estimate the global optimum position. Then the predetermined global optimum is taken as the initial value for the function optimization at detail resolution space (scale level 5). After the optimization at each resolution level, the definition domain shrinks around the determined global optimum, which saves a lot of computation cost.

In this example, the function evaluation of MPSP is less than 1/11 of single-level PSP method. Compared with other global optimization methods, such as SA, multiscale PSP algorithm also shows some superiority in dealing with complex 2D objective functions with dense local minima and fluctuations. In addition, multiscale parameter space partition method is suitable for parallel computation and can be implemented on large scale parallel computers easily.

4.2 MPSP optimization on standard box-constrained functions

In this section, we attempt to solve optimization problems for following box-constrained test functions. These functions have been used widely to test different types of optimization methods [89,92,93]. For each test function $f(x)$, we give the formula and bound on the variables.

1. Drop wave function (DRWV)

$$f(x_1, x_2) = -\frac{1 + \cos\left(12\sqrt{x_1^2 + x_2^2}\right)}{\frac{1}{2}\left(x_1^2 + x_2^2\right) + 2}$$

   where $x_1 \in [-5.12, 5.12]$, $x_2 \in [-5.12, 5.12]$.$\mathbf{x}_{opt} = (0, 0)$, $f_{opt} = -1$.
2. Schwefel's function (SCH) [90]

$$f(x) = \sum_{i=1}^{n}\left(-x_i \sin\left(\sqrt{|x_i|}\right)\right)$$

   where $x_i \in [-500, 500]^n$, $n = 2$. Global minimum is $f(x) = -n \cdot 418.9829, x_i = 420.9687$.
3. Griewank's function (GRW)

$$f(x) = \sum_{i=1}^{n}\frac{x_i^2}{4000} - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

   where $x_i \in [-100, 100]^n$, $n = 2$. Global minimum is $f(x) = 0, x_i = 0$.
4. Ackley's function (ACK) [91]

$$f(x) = -a\exp\left(-b\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos\left(cx_i\right)\right) + a + \exp\left(1\right)$$

   where $x \in [-32, 32]^n$, $n = 2, a = 20, b = 0.2, c = 2\pi$. Global minimum is $f(x) = 0, x_i = 0$.

5. Langermann's function (LAN)

$$f(x) = -\sum_{i=1}^{m} c_i \left[\exp\left(-\frac{\sum_{j=1}^{n} \left(x_j - a_{ij}\right)^2}{x} \cos\left(\pi \sum_{j=1}^{n} \left(x_j - a_{ij}\right)^2 \right) \right) \right]$$

where $x_j \in [0, 10]^n$, $m = 5$ and $n = 2$. $a_{i1} = [3, 5, 2, 1, 7]$. $a_{i2} = [5, 2, 1, 4, 9]$. $c_i = [1, 2, 5, 2, 3]$.

6. Goldstein–Price's function (GOLPRI) [95]

$$\begin{aligned} f(x_1, x_2) = &\left[1 + (x_1 + x_2 + 1)^2 \left(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2\right)\right] \\ &\times \left[30 + (2x_1 - 3x_2)^2 \left(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2\right)\right] \end{aligned}$$

where $x_1 \in [-2, 2], x_2 \in [-2, 2]$. Global minimum is $f(x_1, x_2) = 3, (x_1, x_2) = (0, -1)$.

7. Shekel's foxholes function (SHFX)

$$f(x) = -\sum_{i=1}^{m} \left(\sum_{j=1}^{n} \left[\left(x_j - a_{ij}\right)^2 + c_j \right] \right)^{-1}$$

where $c_i, i = 1, 2, \ldots, m$ and $a_{ij}, j = 1, 2, \ldots, n; i = 1, 2, \ldots, m$ are predetermined constants. Here $m = 30$. For two dimensional Shekel's foxhole problem, the function is:

$$f(x_1, x_2) = -\sum_{j=1}^{m} \left[\left(x_1 - a_j\right)^2 + \left(x_2 - b_j\right)^2 + c_j \right]^{-1}$$

where $x_1 \in [-5, 15], x_2 \in [-5,15]$. Here we use

a_j = [9.681, 9.400, 8.025, 2.196, 8.074, 7.650, 1.256, 8.314, 0.226,
7.305, 0.652, 2.699, 8.327, 2.132, 4.707, 8.304, 8.632, 4.887, 2.440,
6.306, 0.652, 5.558, 3.352, 8.798, 1.460, 0.432, 0.679, 4.263, 9.496, 4.138]

b_j = [0.667, 2.041, 9.152, 0.415, 8.777, 5.658, 3.605, 2.261, 8.858, 2.228, 7.027, 3.516,
3.897, 7.006, 5.579, 7.559, 4.409, 9.112, 6.686, 8.583, 2.343, 1.272, 7.549, 0.880,
8.057, 8.645, 2.800, 1.074, 4.830, 2.562]

c_j = [0.806, 0.517, 0.100, 0.908, 0.965, 0.669, 0.524, 0.902, 0.531, 0.876, 0.462,
0.491, 0.463, 0.714, 0.352, 0.869, 0.813, 0.811, 0.828, 0.964, 0.789, 0.360,
0.369, 0.992, 0.332, 0.817, 0.632, 0.883, 0.608, 0.326]

1. Six-hump camel back function (SIXHUMP) [94]

$$f(x_1, x_2) = \left(4 - 2.1x_1^2 + \frac{x_1^4}{3}\right)x_1^2 + x_1x_2 + \left(-4 + 4x_2^2\right)x_2^2$$

where $x_1 \in [-3,3], x_2 \in [-2,2]$.

The definition space and global solutions for test functions are given in the function definition sections. The test functions are optimized by both MPSP method and SA method. Table 3 provides the computation results for comparison.

Objective functions in above test function set are decomposed at 5 scale levels. Most of the functions are optimized at scale level 1, 2 and 5. Since Goldstein–Price's function

Table 3 The MPSP and simulated annealing (SA) optimization results for box-constrained functions

Functions	MPSP			SA			
SL	GM	FV	EN	GM	FV	EN	
DRWV	1	(0.000000e + 000, −6.250000e − 001)	−4.903406e − 001	698	(1.116187e − 004, −5.513575e − 005)	−9.999994e − 001	16801
	2	(−5.331755e − 002, 5.235715e − 001)	−9.558241e − 001	3411			
	5	(2.189399e − 3, −3.955205e − 3)	−9.997988e − 1	3092			
SCH	1	(4.339452e + 002, 4.328566e + 002)	7.507002e + 002	3623	(−3.040000e + 002, −3.030000e + 002)	−6.030125e + 002	18401
	2	(4.195241e + 002, 4.208265e + 002)	−8.405229e + 002	3807			
	5	(4.221059e + 002, 4.215345e + 002)	8.387200e + 002	4284			
GRW	1	(0.000000e + 000, 0.000000e + 000)	1.000077e + 000	858	(−3.202847e − 004, −1.054665e − 003)	3.296745e − 007	33601
	2	(8.178711e − 002, 5.053711e − 001)	4.155208e − 002	2966			
	5	(8.649826e − 003, −1.612091e − 002)	2.885167e − 004	3695			
ACK	1	(0.000000e + 000, 0.000000e + 000)	3.872181e + 000		(1.881673e − 006, −3.080803e − 007)	5.393134e − 006	18401
	2	(0.000000e + 000, 0.000000e + 000)	2.411640e + 000	4442			
	5	(0.000000e + 000, 0.000000e + 000)	5.887218e − 016	3471			
LAN	1	(2.079712e + 000, 1.923340e + 000)	−3.143258e + 000	870	(2.793402e + 000, 1.597241e + 000)	−4.155809e + 000	19201
	2	(2.929434e + 000, 1.273700e + 000)	−4.396301e + 000	3961			
	5	(2.971649e + 000, 1.050730e + 000)	−4.059696e + 000	5461			
GOLPRI	5	(−5.719863e − 004, −9.998738e − 001)	3.000105e + 000	3425	(4.737213e − 006, −1.000004e + 000)	3.000000e + 000	20001
SHFX	1	(7.746887e + 000, 8.774719e + 000)	−5.144611e + 000	699	(8.024091e + 000, 9.146601e + 000)	−1.211901e + 001	16801
	2	(8.042749e + 000, 9.160522e + 000)	−1.151351e + 001	3763			
	5	(8.031433e + 000, 9.136528e+000)	−1.210358e + 001	4391			
SIXHUMP	5	(8.924693e − 002, −7.142694e − 001)	−1.031605e + 000	4304	(−8.983792e − 002, 7.127082e − 001)	−1.031628	17601

SL multiscale level used in MPSP, *GM* global minima, *FV* function value of global minima, *EN* function evaluation number

(GOLPRI) and Six-hump camel back function (SIXHUMP) are pretty smooth, only level 5 (the original function level) are used in MPSP. The global optima solution (GM), function values (FV) and function evaluation number (EN) are given in Table 3 for MPSP method at different scale levels. The optimization results by Simulated Annealing method are also given in Table 3 for comparison. Control parameters for SA method are as following: Initial temperature is 10.0. Loop number for step adjustment is 20. Loop number for temperature decrease is 20. Convergence error for terminating SA computation is 0.001. Temperature decrease ratio is 0.6. But for Griewank's function (GRW), temperature decrease ratio must be set as 0.8 to obtain an accurate estimation of the global optimal solution. Table 3 shows that the function evaluations for MPSP method are only 1/2 (or less) of that for SA. Meanwhile, the solution accuracies of the both method are at the same level. What we want to deliver by comparing MPSP with SA is that deterministic optimization method based on parameter space partition is more favorable in dealing with complex functions with multiple local optima. However, more comparisons with state-of-the-art GO methods are expected in our future work to provide more evidences.

Computation results suggest that the deterministic optimization method based on multiscale parameter space partition has better performance in dealing with objective functions with complex landscape. Wriggling basin paths, steep cliffs and fluctuating valley areas can be smoothed at coarse scale level, which make it easier to estimate the approximated global optimal solutions. As the optimizations are carried out from coarse scale to detail scale level, the functions at multiscale levels will approach the original function. Based on the knowledge of approximated location of global optimal solution, the searching domain can be shrunk from coarse scale levels to detail levels, which saves computation costs greatly. In the case of smooth objective functions, the evidence of better performance of MPSP method will not be as remarkable as for complex functions.

4.3 Inversion problem for two-dimensional acoustic wave propagation

A three-layer acoustic wave model is considered to check MPSP inversion method. Figure 10 shows the three layer model parameters. The units for velocity and density are m/s and kg/m^3.

The model size is 500×500 m in x–z space. Explosive source is located at (250,50 m).The maximal source frequency is 60 Hz. Receivers are located at surface with 5 meters interval. Acoustic wave propagation is simulated with finite difference method in three-layer model. Absorbing boundary conditions are applied to model boundaries. The size of finite difference

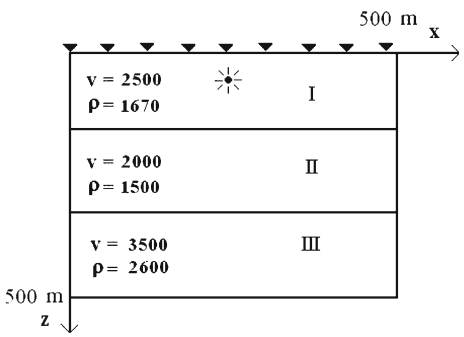

Fig. 10 The parameters for three-layer acoustic wave propagation model

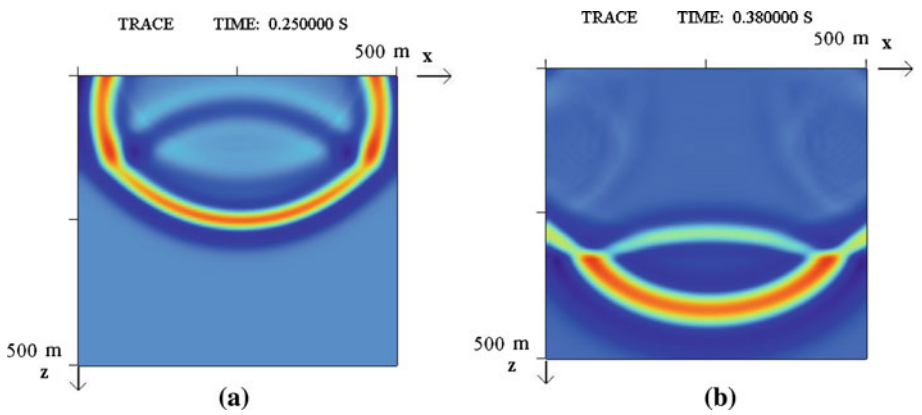

Fig. 11 **a** Wave field snapshot at $t = 0.25$ s. **b** Wave field snapshot at $t = 0.38$ s

grid is 5×5 m. Total simulation time is 1.2 s. The wave field snapshot at $t = 0.25$ and 0.38 s show wave reflections at layer interfaces (Fig. 11).

The velocity and density of layer II are selected as inversion parameters. Objective function is defined as the errors between observed and simulated data:

$$\Delta f = \sqrt{\frac{\sum_{i=1}^{N_T} \sum_{j=1}^{N_S} \left(S_{sim}^{ij} - S_{obs}^{ij} \right)^2}{N_T N_S}} \tag{26}$$

Here N_T, N_S are receiver number and sample number in each receiver channel. S_{sim}^{ij} is simulated signal amplitude of jth sample in ith receiver channel. S_{obs}^{ij} is observed signal amplitude of jth sample in ith receiver channel. Here the so called "observed" data are from simulation with true model parameters.

In order to have an overall understanding of how the objective function depends on parameters to be optimized, Fig. 12 provides the Δf function surface in velocity-density (Layer II) domain. The velocity range is [1000, 3500] (m/s). Density range is [1000, 3500] (kg/m^3). The whole velocity-density domain is divided into rectangle meshes with 50 m/s velocity interval and 50 kg/m^3 density interval. The objective function values are calculated for all velocity-density parameter pairs at the uniformly distributed grid nodes.

According to the analysis of objective function surface, there are multiple local minima in density-velocity inversion problem. In addition, the global optimal solutions are located at the bottom of a very narrow valley with steep and high cliff edges, which is like searching a needle in straws.

MPSP method is used to search the global optimal velocity-density values for layer II. The velocity searching domain for MPSP is [1000, 5000] (m/s). The density searching domain is [500, 3500] (kg/m^3). Uniform design algorithm is used in selecting initial solutions in the rectangular domain at different scale. The optimization will stop when objective function converges to a pre-determined value or the function evaluation reaches a predetermined maximal number.

In this example, the pre-determined converge condition is $|\Delta f_i - \Delta f_{i-1}| < 1.0$. MPSP method takes totally 522 function evaluations at 8 scale levels, which means 522 wave propagation simulations. The solution distribution is shown in Fig. 13a. It can be easily found that sampling domain at different scale levels are approaching to the global solutions. The final

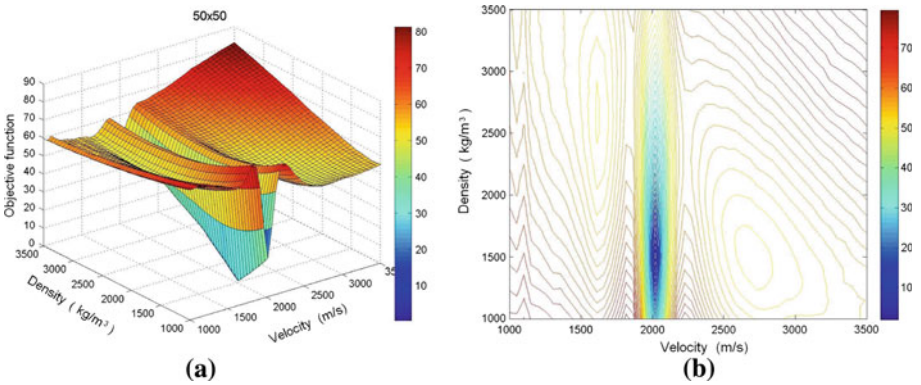

Fig. 12 Objective function with respect to velocity-density (layer II) parameter pairs **a** function surface, **b** function contour

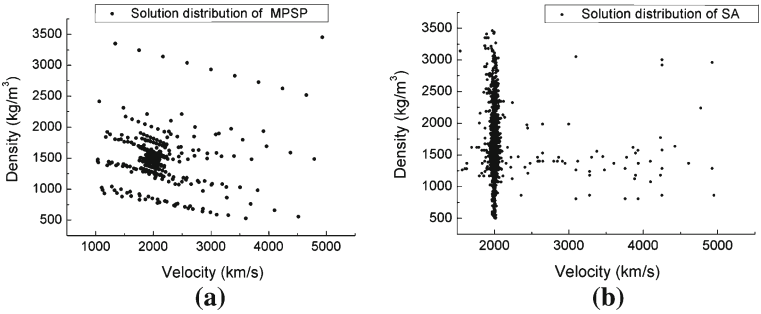

Fig. 13 **a** Solution distribution of MPSP method with uniform design sampling at each scale(522 sampling point at 8 scale levels), **b** solution distribution of SA method(1,500 sampling point)

solution after 522 function evaluation is $v = 2000.314530$ (m/s), $\rho = 1495.671055\,(\mathrm{kg/m^3})$. The corresponding objective function value is $\Delta f = 0.796985$.

Simulated Annealing (SA) method is used for comparison. In order to reach the same predetermined converge condition, SA need at least 1,500 sampling point in parameter domain, which means 1,500 wave propagation simulations. Figure 13b shows the sampling point distribution in velocity-density domain. The final optimized solution is $v = 2,000\,(\mathrm{m/s})$, $\rho = 1,510\,(\mathrm{kg/m^3})$. The corresponding objective function value is $\Delta f = 0.907128$. Control parameters for SA method are as following: Initial temperature is 6.0. Temperature decrease ratio is 0.8. Loop number for step adjustment is 20. Loop number for temperature decrease is 20. The initial solution are $v = 3,000.0\,\mathrm{m/s}$ and $\rho = 2,000.0\,\mathrm{kg/m^3}$.

From the solution distributions for MPSP and SA, some interesting conclusions can be achieved. First, sampling points of SA method show great stochastic features. SA can quickly find out the solutions near the global optimal solution and intensively search better solutions in nearby domain. In other domains, there are not any searching trials. SA has capability to escape from local solutions at the early stage. However, as SA finds better solutions, searching step becomes smaller and smaller, which constrain the optimization at a small domain with concentrated sampling points. Second, MPSP persist selecting solutions uniformly at each scale level. From coarsest to finest scale, MPSP shrinks searching domains and approaches to the global solutions. Although the searching step becomes smaller and smaller, the sampling

point will not increase and so as the computation costs. Compared with SA, the computing costs of MPSP is only 1/3 (522/1,500) with same pre-determined converge condition.

5 Conclusions

This paper developed a multiscale parameter space partition method, which divides the definition space into subdomains containing the same local optimal solutions. All the local optimization solutions (sets) are to be determined after finite times of parameter space partition steps. The new method has a stable convergence speed and the optimization solution is independent on the selection of initial solutions. This study provides a new method for nonlinear inverse problems.

Acknowledgments The work in this paper was sponsored by the National Natural Science Foundation of China (Grant No. 10402015) and Open Project Program of State Key Laboratory of Petroleum Resource and Prospecting (China University of Petroleum, Beijing) (Grant No. 2009010). The computation works were partly supported by CNPC Innovation Fund (Grant No. 05E7010), National Natural Science Foundation of China (Grant No. 60573017) and Hi-Tech Research And Development Program of China (Grant No. 2008AA01Z102).

References

1. Bomze, I.M., Csendes, T., Horst, R., Pardalos, P.M.: Developments in Global Optimization: Nonconvex Optimization and Its Applications. Kluwer, Dordrecht (1997)
2. Gray, P., Hart, W., Painton, L., et al.: A Survey of Global Optimization Methods. Technical report, Sandia National Laboratories (1997)
3. Scales, J.A., Smith, M.L., Fischer, T.L: Global optimization methods for multimodal inverse problems. J. Comput. Phys. **103**, 258–268 (1992)
4. Deng, L.H., Scales, J.A.: Estimating the Topography of Multi-dimensional Fitness Functions. Colorado School of Mines (1999)
5. Rothman, D.H.: Nonlinear inversion, statistical-mechanics, and residual statics Estimation. Geophysics **50**, 2784–2796 (1985)
6. Rothman, D.H.: Automatic estimation of large residual statics corrections. Geophysics **51**, 332–346 (1986)
7. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. Science **220**, 671–680 (1983)
8. Holland, J.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor (1975)
9. Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, Reading (1989)
10. Stoffa, P.L., Sen, M.K.: Nonlinear multiparameter optimization using genetic algorithms—inversion of plane-wave seismograms. Geophysics **56**, 1794–1810 (1991)
11. Sambridge, M., Drijkoningen, G.: Genetic algorithms in seismic wave-form inversion. Geophys. J. Int. **109**, 323–342 (1992)
12. Gallagher, K., Sambridge, M., Drijkoningen, G.: Genetic algorithms—an evolution from Monte-Carlo methods for strongly nonlinear geophysical optimization problems. Geophys. Res. Lett. **18**, 2177–2180 (1991)
13. Gallagher, K., Sambridge, M.: Genetic algorithms—a powerful tool for large-scale nonlinear optimization problems. Comput. Geosci. **20**, 1229–1236 (1994)
14. Sen, M., Stoffa, P.L.: Global Optimization Methods in Geophysical Inversion. Elsevier, Amsterdam (1995)
15. Gill, P.E., Murray, W., Wright, M.H.: Practical Optimization. Academic Press, New York (1981)
16. Granville, V., Krivanek, M., Rasson, J.P.: Simulated annealing—a proof of convergence. IEEE Trans. Pattern Anal. Mach. Intell. **16**, 652–656 (1994)
17. Greenhalgh, D., Marshall, S.: Convergence criteria for genetic algorithms. SIAM J. Comput. **30**, 269–282 (2000)
18. Locatelli, M.: Convergence and first hitting time of simulated annealing algorithms for continuous global optimization. Math. Methods Oper. Res. **54**, 171–199 (2001)

19. Locatelli, M.: Convergence of a simulated annealing algorithm for continuous global optimization. J. Glob. Optim. **18**, 219–234 (2000)
20. Locatelli, M.: Simulated annealing algorithms for continuous global optimization: convergence conditions. J. Optim. Theory Appl. **104**, 121–133 (2000)
21. Locatelli, M.: Convergence properties of simulated annealing for continuous global optimization. J. Appl. Probab. **33**, 1127–1140 (1996)
22. Belisle, C.J.P.: Convergence theorems for a class of simulated annealing algorithms on R(D). J. Appl. Probab. **29**, 885–895 (1992)
23. Fallat, M.R., Dosso, S.E.: Geoacoustic inversion via local, global, and hybrid algorithms. J. Acoust. Soc. Am. **105**, 3219–3230 (1999)
24. Liu, P.C., Hartzell, S., Stephenson, W.: Nonlinear multiparameter inversion using a hybrid global search algorithm—applications in reflection seismology. Geophys. J. Int. **122**, 991–1000 (1995)
25. Cary, P.W., Chapman, C.H.: Automatic 1-D waveform inversion of marine seismic refraction data. Geophys. J. Int. **93**, 527–546 (1988)
26. Gerstoft, P.: Inversion of acoustic data using a combination of genetic algorithms and the Gauss–Newton approach. J. Acoust. Soc. Am. **97**, 2181–2190 (1995)
27. Hibbert, D.B.: A hybrid genetic algorithm for the estimation of kinetic-parameters. Chemometr. Intell. Lab. **19**, 319–329 (1993)
28. Chunduru, R.K., Sen, M.K., Stoffa, P.L.: Hybrid optimization methods for geophysical inversion. Geophysics **62**, 1196–1207 (1997)
29. Calderon-Macias, C., Sen, M.K., Stoffa, P.L.: Artificial neural networks for parameter estimation in geophysics. Geophys. Prospect. **48**, 21–47 (2000)
30. Chelouah, R., Siarry, P.: A hybrid method combining continuous tabu search and Nelder–Mead simplex algorithms for the global optimization of multiminima functions. Eur. J. Oper. Res. **161**, 636–654 (2005)
31. Gil, C., Marquez, A., Banos, R. et al.: A hybrid method for solving multi-objective global optimization problems. J. Glob. Optim. **38**, 265–281 (2007)
32. Olensek, J., Burmen, A., Puhan, J., Tuma, T.: DESA: a new hybrid global optimization method and its application to analog integrated circuit sizing. J. Glob. Optim. **44**, 53–77 (2009)
33. Yiu, K.F.C., Liu, Y., Teo, K.L.: A hybrid descent method for global optimization. J. Glob. Optim. **28**, 229–238 (2004)
34. Xu, P.L.: A hybrid global optimization method: the multi-dimensional case. J. Comput. Appl. Math. **155**, 423–446 (2003)
35. Hedar, A.R., Fukushima, M.: Hybrid simulated annealing and direct search method for nonlinear unconstrained global optimization. Optim. Methods Softw. **17**, 891–912 (2002)
36. Barhen, J., Protopopescu, V., Reister, D.: TRUST: a deterministic algorithm for global optimization. Science **276**, 1094–1097 (1997)
37. Basso, P.: Iterative methods for the localization of the global maximum. SIAM J. Numer. Anal. **19**, 781–792 (1982)
38. Shubert, B.O.: Sequential method seeking global maximum of a function. SIAM J. Numer. Anal. **9**, 379–388 (1972)
39. Floudas, C.: Global Optimization: Theory, Methods and Applications. Kluwer, Dordrecht (2000)
40. Hansen, E.: Global optimization using interval-analysis—the multidimensional case. Numer. Math. **34**, 247–270 (1980)
41. Hansen, E.R.: Global optimization using interval analysis— one-dimensional case. J. Optim. Theory Appl. **29**, 331–344 (1979)
42. Hansen, E.: Global Optimization Using Interval Analysis. Marcel Dekker, New York (1992)
43. Ichida, K., Fujii, Y.: Interval arithmetic method for global optimization. Computing **23**, 85–97 (1979)
44. Kearfott, R.B.: Rigorous Global Search: Continuous Problems. Kluwer, Dordrecht (1996)
45. Ratschek, H., Rokne, J.: New Computer Methods for Global Optimization. Ellis Horwood, Chichester (1988)
46. Tarvainen, M., Tiira, T., Husebye, E.S.: Locating regional seismic events with global optimization based on interval arithmetic. Geophys. J. Int. **138**, 879–885 (1999)
47. Land, A.H., Doig, A.G.: An automatic method for solving discrete programming problems. Econometrica **28**, 497–520 (1960)
48. Clausen, J.: Branch and bound algorithms—principles and examples. In: Department of Computer Science, University of Copenhagen (1999, March)
49. Tawarmalani, M., Sahinidis, N.V.: Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming: Theory, Algorithms, Software, and Applications. Kluwer, Boston (2002)
50. Khajavirad, A., Michalek, J.J.: A deterministic Lagrangian-based global optimization approach for quasi-separable nonconvex mixed-integer nonlinear programs. J. Mech. Design **131**, 051009 (8pp)

51. Qu, S.J., Ji, Y., Zhang, K.C.: A deterministic global optimization algorithm based on a linearizing method for nonconvex quadratically constrained programs. Math. Comput. Model. **48**, 1737–1743 (2008)
52. Jiao, H.W., Chen, Y.Q.: A note on a deterministic global optimization algorithm. Appl. Math. Comput. **202**, 67–70 (2008)
53. Wu, Y., Lai, K.K., Liu, Y.J.: Deterministic global optimization approach to steady-state distribution gas pipeline networks. Optim. Eng. **8**, 259–275 (2007)
54. Long, C.E., Polisetty, P.K., Gatzke, E.P.: Deterministic global optimization for nonlinear model predictive control of hybrid dynamic systems. Int. J. Robust Nonlinear Control **17**, 1232–1250 (2007)
55. Lin, Y.D., Stadtherr, M.A.: Deterministic global optimization of nonlinear dynamic systems. AICHE J. **53**, 866–875 (2007)
56. Ji, Y., Zhang, K.C., Qu, S.H.: A deterministic global optimization algorithm. Appl. Math. Comput. **185**, 382–387 (2007)
57. Lin, Y., Stadtherr, M.A.: Deterministic global optimization for parameter estimation of dynamic systems. Ind Eng Chem Res **45**, 8438–8448 (2006)
58. Long, C.E., Polisetty, P.K., Gatzke, E.P.: Nonlinear model predictive control using deterministic global optimization. J. Process Control **16**, 635–643 (2006)
59. Lin, Y.D., Stadtherr, M.A.: Deterministic global optimization of molecular structures using interval analysis. J. Comput. Chem. **26**, 1413–1420 (2005)
60. Sun, W.T., Shu, J.W., Zheng, W.M.: Deterministic global optimization with a neighbourhood determination algorithm based on neural networks. In: Advances in Neural Networks—ISNN 2005, Pt 1, Proceedings, vol. 3496, pp. 700–705 (2005)
61. Messine, F.: Deterministic global optimization using interval constraint propagation techniques. Rairo Oper. Res. **38**, 277–293 (2004)
62. Adjiman, C.S., Papamichail, I.: A deterministic global optimization algorithm for problems with nonlinear dynamics. Front. Glob. Optim. **74**, 1–23 (2004)
63. Gau, C.Y.T., Schrage, L.E.: Implementation and testing of a branch-and-bound based method for deterministic global optimization: operations research applications. Front. Glob. Optim. **74**, 145–164 (2004)
64. Lin, Y., Stadtherr, M.A.: Advances in interval methods for deterministic global optimization in chemical engineering. J. Glob. Optim. **29**, 281–296 (2004)
65. Bartholomew-Biggs, M.C., Parkhurst, S.C., Wilson, S.R.: Global optimization—stochastic or deterministic?. Stoch. Algorithms Found. Appl. **2827**, 125–137 (2003)
66. Sambridge, M.: Geophysical inversion with a neighbourhood algorithm—II. Appraising the ensemble. Geophys. J. Int. **138**, 727–746 (1999)
67. Sambridge, M.: Geophysical inversion with a neighbourhood algorithm—I. Searching a parameter space. Geophys. J. Int. **138**, 479–494 (1999)
68. Sambridge, M., Braun, J., Mcqueen, H.: Geophysical parametrization and interpolation of irregular data using natural neighbors. Geophys. J. Int. **122**, 837–857 (1995)
69. Locatelli, M., Wood, G.R.: Objective function features providing barriers to rapid global optimization. J. Glob. Optim. **31**, 549–565 (2005)
70. Locatelli, M.: On the multilevel structure of global optimization problems. Comput. Optim. Appl. **30**, 5–22 (2005)
71. Daubechies, I., Mallat, S., Willsky, A.S.: Special issue on wavelet transforms and multiresolution signal analysis—introduction. IEEE Trans. Inf. Theory **38**, 529–531 (1992)
72. Daubechies, I.: The wavelet transform, time-frequency localization and signal analysis. IEEE Trans. Inf. Theory **36**, 961–1005 (1990)
73. Daubechies, I.: Ten Lectures on Wavelets. SIAM, Philadelphia (1992)
74. Mallat, S.: A theory for multiresolution signal decomposition: the wavelet representation. IEEE Trans. Pattern Anal. Mach. Intell. **11**, 674–693 (1989)
75. Meyer, Y.: Principle d'incertitude, basis Hilbertiennes et algebras d'operateurs. In: Bourbaki Seminar (1885–1986)
76. Daubechies, I.: Orthonormal bases of compactly supported wavelets. Commun. Pure Appl. Math. **41**, 909–996 (1988)
77. Daubechies, I., Paul, T.: Time frequency localization operators—a geometric phase-space approach 2. The use of dilations. Inverse Probl. **4**, 661–680 (1988)
78. Kalantari, B., Rosen, J.B.: Construction of large-scale global minimum concave quadratic test problems. J. Optim. Theory Appl. **48**, 303–313 (1986)
79. Floudas, C., Pardalos, P.M.: A collection of test problems for constrained global optimization algorithms. In: Goos GaH, J. Lecture Notes in Computer Science, Springer, Berlin (1990)
80. Khoury, B.N., Pardalos, P.M., Du, D.Z.: A test problem generator for the Steiner problem in graphs. ACM Trans. Math. Softw. **19**, 509–522 (1993)

81. Schoen, F.: A wide class of test functions for global optimization. J. Glob. Optim. **3**, 133–137 (1993)
82. Mathar, R., Zilinskas, A.: A class of test functions for global optimization. J. Glob. Optim. **5**, 195–199 (1994)
83. Facchinei, F., Judice, J., Soares, J.: Generating box-constrained optimization problems. ACM Trans. Math. Softw. **23**, 443–447 (1997)
84. Gaviano, R., Lera, D.: Test functions with variable attraction regions for global optimization problems. J. Glob. Optim. **13**, 207–223 (1998)
85. Gaviano, M., Kvasov, D.E., Lera, D., Sergeyev, Y.D.: Algorithm 829: software for generation of classes of test functions with known local and global minima for global optimization. ACM Trans. Math. Softw. **29**, 469–480 (2003)
86. Mishra, S.: Some new test functions for global optimization and performance of repulsive particle swarm method. In: MPRA (2006)
87. Addis, B., Locatelli, M.: A new class of test functions for global optimization. J. Glob. Optim. **38**, 479–501 (2007)
88. Jones, D.R., Perttunen, C.D., Stuckman, B.E.: Lipschitzian optimization without the Lipschitz constant. J. Optim. Theory Appl. **79**, 157–181 (1993)
89. Liang, J.J., Suganthan, P.N., Deb, K.: Novel composition test functions for numerical global optimization. In: 2005 IEEE Swarm Intelligence Symposium, Pasadena, pp. 68–75. IEEE Press (2005)
90. Schwefel, H.-P.: Numerical Optimization of Computer Models. Wiley, New York (1981)
91. Ackley, D.H.: A Connectionist Machine for Genetic Hillclimbing. Springer, Boston (1987)
92. Conn, A.R., Gould, N.I.M., Toint, P.L.: Testing a class of methods for solving minimization problems with simple bounds on the variables. Math. Comput. **50**, 399–430 (1988)
93. Branch, M.A., Coleman, T.F., Li, Y.Y.: A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems. SIAM J. Sci. Comput. **21**, 1–23 (1999)
94. Dixon, L.C.W., Szego, G.P.: The optimization problem: an introduction. In: Dixon, L.C.W., Szego, G.P. (eds.) Towards Global Optimization II., North Holland, New York (1978)
95. Goldstei, A.A., Price, J.F.: Descent from local minima. Math. Comput. **25**, 569–574 (1971)