CWP-239
January 1997

A Complexity Analysis of Generic Optimization Problems: Characterizing the Topography of High-Dimensional Functions

Hongling Lydia Deng

— Doctoral Thesis —
Mathematical and Computer Sciences

Center for Wave Phenomena
Colorado School of Mines
Golden, Colorado 80401
303/273-3557

ABSTRACT

Optimization problems arise in every scientific and engineering field. It is often the case in practice that the function to be optimized (*objective function*) cannot be specified in closed form in terms of elementary functions, but must be evaluated pointwise via a computer program. For such black-box objective functions, an important issue is how to select appropriate optimization tools. Careful comparison of optimization techniques requires a meaningful way of characterizing the complexity of generic optimization problems.

An objective function can be thought of as a high-dimensional surface with "hills" and "basins" (local extrema) of different sizes scattered in the domain. Based on a catalog of diverse test functions, we consider three topographical factors to be important in influencing the performance of optimization algorithms. These factors are the number of local extrema in the domain, the widths of the corresponding basins (hills) and their relative depths (heights). In this thesis, I show how this topographic information can be used to characterize the complexity of generic optimization problems, independent of any particular search or optimization algorithm.

When the optimization problem is high-dimensional and the objective function can only be sampled point-wise, the topographical information is not accessible analytically. Estimations of the function topography, in such situations, can be obtained by statistical analysis of the results of many independent random local-descent searches.

This statistical estimation has two main sources of error: the sampling error and the error due to the numerical implementation. After defining *complexity*, I show that the distribution of the estimated complexity is approximately Gaussian. The mean value of this distribution is the exact complexity, and the variance goes to zero when the number of samples goes to infinity. A confidence-interval analysis is used to bound the error due to the finite sampling. Furthermore, I show that the numerical error of this complexity analysis depends on the accuracy of the local-descent search algorithms, the characteristics of the objective function, as well as the number of samples used. This type of error is bounded under different numerical situations, and the detailed implementation issues are discussed.

Numerical results will be presented for a variety of analytical test functions. Using the Griewank function, for example, we see that merely plotting hyper-planes of high-dimensional functions can be misleading. Furthermore, some functions become easier to optimize in high dimensions, but such conclusions can be supported only by a global analysis. On the other hand, the Rosenbrock function exhibits increasing numerical complexity because of ill-conditioning and the resulting numerical error.

Finally, as an example of a realistic optimization problem, I devote a complete chapter to the residual-statics estimation problem of explorational geophysics – a high-dimensional, highly multimodal optimization problem. After observing the specific

properties of this problem, I suggest two approaches for simplifying the objective function: multi-resolution analysis (MRA) and use of envelope information. The complexity analysis developed in this thesis is used for evaluating the performance of these strategies. In particular, I show that the results of the complexity analysis can be used to determine an optimal degree in simplifying objective functions.

TABLE OF CONTENTS

ACKNOWLEDGMENT

It is such a relief that I have finally finished the 7.5-year graduate-school life with two degrees (M.S. and Ph.D) from Colorado School of Mines (CSM). Besides being happy and proud, I cannot help feeling thankful. I am thankful for the opportunity of studying in this great school, for the very special education I have received from the Center for Wave Phenomena (CWP), for the rich diversity of culture I have been exposed to, for the friendship I have enjoyed over the years, and for the fresh air and beautiful mountains of Colorado. My stay at Mines would have been a much shorter one without beautiful mountains in Colorado, without the first meeting with Dr. Dave Hale who showed me what an electrical engineer can do with seismic data, or without the successful brain surgery at the St. Anthony Hospital.

During this long period of time, I have grown and transformed greatly. My knowledge has been enriched and my eyes opened by my interaction with many people of various cultural backgrounds. Many people have influenced me in many ways during these years. First and foremost, I want to thank two of my advisors at different stages of my study: Dr. Dave Hale, who got me started in a field I knew nothing about -- geophysics, whose continuous encouragement and enlightened ideas were always the stimulation during the period of my M.S. program; Dr. John Scales, my Ph.D. advisor, introduced me to other exciting fields – inversion, optimization, and object-oriented programming. His insights and experience in science has been refreshing and a precious resource of my research. His door is always open for discussions, anything from science to coffee. He gave me the freedom to choose research subjects, and always generously offered praise and encouragement whenever I felt in despair. This research is mainly a result of our continuous discussion.

I have also benefited from the special education and opportunities CWP provided. Besides Drs. Hale and Scales, other faculty members of CWP, Drs. Norman Bleistein, Ken Larner, and Jack Cohen, are also much appreciated. Dr. Larner's endless patience while proof-reading my work, correcting my grammatical mistakes, and rehearsing my presentations, is particularly appreciated. I also thank other committee members, Drs. Gregory Beylkin, Steve Pruess, and Manavendra Misra, for their time and comments at various stage of this research. Dr. Bill Navidi has offered much help with the statistical analysis during this research. I have also enjoyed and benefited from discussions with Dr. Erik Van Vleck on global optimization.

Many thanks go to people who helped me during the hospital days. CSM president, Dr. Ansell, who advised my father's Ph.D. dissertation, signed for my surgery so it could be successfully performed in time. Mrs. Ansell contacted my family and visited me many times at the hospital. Dr. Boes, then department head of Mathematical and Computer Sciences, not only visited me at the hospital, but also took me back to the doctor during the recovery stage. Dorothy Nelson's friendship and advice made

my difficult times easier. I appreciate the help and encouragement of all graduate students and faculty in the department and the Chinese community at Mines. Guy Fredrickson offered his precious and unique friendship that helped me going through most difficult times when I felt everything was turning away from me. Without his friendship, I could not imagine how I could have reached this point.

Friendship has been an important part of my life being at Mines. Besides Guy, I am also thankful for Gerardo Garcia, Katerina Papakonstantinu, Andreas Rüger, and Timo Tjan for being my friends, who I could always count on no matter what. Other friends are Tagir Galikeev, Boyi Ou, Wences Gouveia, Dong Chi, Craig Artley, Justin Hedley, Paul Fowler, Björn & Esther Rommel, and many more. These friends shared with me their colorful cultural background, cheered me up when I was frustrated, and helped me out when I was confused.

My industry friends, Alvin Tenpo, Bin Wang, Keh Pann, Wenjie Dong, and Gregg Zelewsky, have made the hot days much more enjoyable during my summer jobs. Particularly, I enjoyed working and talking with Keh, who has been my mentor and friend for the past few years. Whether the topic was about science, philosophy, or Beethoven, I have learned so much and enjoyed every moment of it. In addition, Drs. Zhiming Li, Terry Young, and Stew Levin not only provided opportunities of working with the oil industry, but also shared their insights in geophysics and gave me much encouragement.

My love of science and literature is mostly inspired by my parents. They have always encouraged me to seek higher standards, even though that meant I would have to be further and further away from home. My father's advice has guided me throughout these years across the distance, and will continue to guide me through my lifetime. Finally, I am deeply in debt to my dear husband, Zhiwei, who has always had faith in me. His unconditional love and support has been my main motivation during the past three years.

This research was partially supported by the sponsors of the Consortium Project on Seismic Inverse Methods for Complex Structures at the Center for Wave Phenomena, Colorado School of Mines, and Oak Ridge National Laboratory.

To Zhiwei, my dear husband, for his unconditional love and support.
To Dr. Keh Pann, my mentor and friend for the past few years and, I hope, for the years to come.

Chapter 1

INTRODUCTION

1.1 A Mathematical Model for Optimization Problems

Many problems in science and engineering can be formulated in terms of the maximization or minimization of a real-valued function of some number of parameters, possibly subject to constraints. We refer to the function being optimized as the *objective function.* The domain of a objective function is generally referred to as *model space* $\mathcal{M}$, in which each point $\mathbf{m}$ is a *model*[1].

If the model space is continuous, the optimization problem is said to be *continuous.* On the other hand if the model space consists of a finite number of points, the optimization problem is said to be *combinatorial.*

For both continuous and combinatorial optimization problems, a metric ρ can be defined to measure the distance between any two models in the model space. Such a metric can be either the Euclidean distance between two vectors in a continuous model space, or the number of models along the shortest path of connecting the two models for discrete model spaces. Therefore, a model space is a *metric space* (Goldburg, 1976). Next, several useful concepts are defined.

Definition 1 (Neighborhood Set) *Let $\mathcal{M}$ be the model space. A neighborhood set of a model, $\mathbf{N}(\mathbf{m}^*, r) \subset \mathcal{M}$, is a set about the model $\mathbf{m}^*$ so that*

$$\mathbf{N}(\mathbf{m}^*, r) = \{\mathbf{m} \in \mathcal{M} \mid \rho(\mathbf{m}, \mathbf{m}^*) \leq r\}. \tag{1.1}$$

Definition 2 (Local and Global Extrema) *Let the objective function F be a mapping $\mathcal{M} \subset \mathbf{R}^N$ into $\mathcal{Y} \subset \mathbf{R}$. A model $\mathbf{m}^* \in \mathcal{M}$ is a* local minimum *of F with respect to the neighborhood $\mathbf{N}(\mathbf{m}^*, r)$ if*

$$F(\mathbf{m}^*) \leq F(\mathbf{m}), \quad \forall \mathbf{m} \in \mathbf{N}(\mathbf{m}^*, r). \tag{1.2}$$

If equation (1.2) is satisfied for a neighborhood set $\mathbf{N}(\mathbf{m}^*, r) = \mathcal{M}$, then the model $\mathbf{m}^*$ is a global minimum of the function F.

Similarly, a local maximum *and* global maximum *of an objective function F are the local minimum and global minima of the function $-F$ with respect to the corresponding neighborhood set. An extremum can be either a minimum or a maximum.*

[1] A bold lower-case Roman letter $\mathbf{m}$ is used to denote a model in the model space.

In this thesis, I loosely refer to the model $\mathbf{m}^*$ as an extremum if it is a local extremum with respect to a neighborhood set $\mathbf{N}(\mathbf{m}^*, r)$ for some $r > 0$. When the objective function has a unique extremum, the function is *unimodal*. Functions with more than one extremum are said to be *multimodal*. Depending on the problem, the models sought in an optimization calculation may be either local or global extrema. All extrema are either *critical points*, where the gradient of the function vanishes ($F'(\mathbf{m}^*) = 0$), or boundary points. Therefore, many optimization algorithms are procedures that look for critical points.

Without loss of generality, I assume in this thesis that all optimization problems are minimization problems unless stated otherwise. Furthermore to simplify the analysis, I assume that the objective functions have finite numbers of *isolated extrema* in the model space. By that, I mean if $\mathbf{m} \in \mathbf{N}(\mathbf{m}^*, r)$ and $\mathbf{m} \neq \mathbf{m}^*$, then $F(\mathbf{m}^*) < F(\mathbf{m})$ in Definition 2.

1.1.1 A Generic Search Algorithm

Virtually all optimization algorithms are iterative searching procedures. I denote the iteration index by time step $t = 0, 1, 2, \ldots$. An *initial population* is chosen by some means, $\vec{\mathbf{m}}_0$[2] $\equiv \{\mathbf{m}_{0k}\}_{k=1,\cdots,K}$

where $K \geq 1$ is the size of this initial population. Then, the set of models $\vec{\mathbf{m}}_{t+1}$ evolves from the set $\vec{\mathbf{m}}_t$ under the action of a *transition operator* $\mathbf{T}(t)$,

$$\vec{\mathbf{m}}_{t+1} = \mathbf{T}(t)\vec{\mathbf{m}}_t. \tag{1.3}$$

I assume this procedure converges to a final set of models $\vec{\mathbf{m}}$,

$$\vec{\mathbf{m}}^* = \lim_{t\to\infty} \mathbf{T}(t)\, \vec{\mathbf{m}}_t. \tag{1.4}$$

In practice certain *stopping criteria*, such as a maximum computing time or an error threshold, need to be imposed, in which case $\vec{\mathbf{m}}^*$ depends implicitly on these criteria.

In order to be able to treat a broad variety of situations, we now define an abstract statement of a search algorithm, which has the following necessary elements: objective function F, initial population $\vec{\mathbf{m}}_0$, transition operator $\mathbf{T}$, and the stopping criterion S.

Algorithm 1 General Search (GS) $\vec{\mathbf{m}}^* = \mathrm{GS}(F, \vec{\mathbf{m}}_0, \mathbf{T}, S)$
Let $F : \mathcal{M} \subset \mathbf{R}^N \to \mathcal{Y} \subset \mathbf{R}$, $\vec{\mathbf{m}}_0$ *be an initial population with size* $K \geq 1$, $\mathbf{T}(t)$ *be a transition operator, and* S *be a stopping criterion.*

1. *Iteratively apply the transition operator to generate a new population of models, so that* $\vec{\mathbf{m}}_{t+1} = \mathbf{T}(t)\, \vec{\mathbf{m}}_t$*;*

[2] An over-vector $\vec{\mathbf{m}}$ is used to denote a set of models in the model space.

2. *Repeat (1) until S is satisfied. The final set of models $\vec{\mathbf{m}}^*$ is the output of the search.*

Different optimization algorithms differ by their strategies of choosing the initial population $\vec{\mathbf{m}}_0$ and the rules of transition $\mathbf{T}$. The stopping criterion S in numerical implementations is also an important factor for any optimization.

1.1.2 Basins of Attraction

There has been a rich mathematical theory on optimization algorithms for objective functions that are unimodal. These optimization algorithms developed under this theory correspond to realizations of Algorithm 1 when the initial population size $K = 1$ and the transition operator is deterministic and independent of the time step t. These transition operators, denoted by $\mathbf{T}_{local}$, involve taking the initial models "downhill" as far as possible, and are referred to as *local-descent* searches (Fletcher, 1987; Dennis & Schnabel, 1987). These local-descent search methods can be used to characterize local regions of objective functions.

Next, I introduce two of the most frequently used concepts in this thesis, the *basins of attraction* of an objective function and their widths. The definitions are given for the combinatorial case, but they can be naturally extended to the continuous case.

Definition 3 (Basin of Attraction) *Let the model space $\mathcal{M}$ have M models in total, where M is finite. Let $\{\check{\mathbf{m}}_l\}_{l=1,\ldots,n}$ be the set of models associated with isolated local minima, where $1 \leq n < M$. Define subsets of models $B_l \subseteq \mathcal{M}$, $l \in [1, n]$, so that*

$$B_l = \{\mathbf{m}_i | \lim_{t\to\infty} \mathbf{T}^t_{local}\mathbf{m}_i = \check{\mathbf{m}}_l, \quad \mathbf{m}_i \in \mathcal{M}\}, \tag{1.5}$$

Then, B_l is the lth basin of attraction of the objective function F in $\mathcal{M}$. These basins of attraction are mutually disjoint. That is,

$$B_l \cap B_j = \emptyset \quad l \neq j. \tag{1.6}$$

According to Definition 3, a basin of attraction is formed by all models that are on the paths of local-descent searches leading to the same local minimum. Since the local minima are assumed to be distinct for the problem studied in this thesis, the model space is divided into several disjoint sets $\{B_l\}_{l=1,\ldots,n}$, and these sets form the complete model space, *i.e.*,

$$\bigcup_{l=1}^{n} B_l = \mathcal{M}, \tag{1.7}$$

where n is the number of local minima. Then, the width of a basin of attraction can be defined as follows.

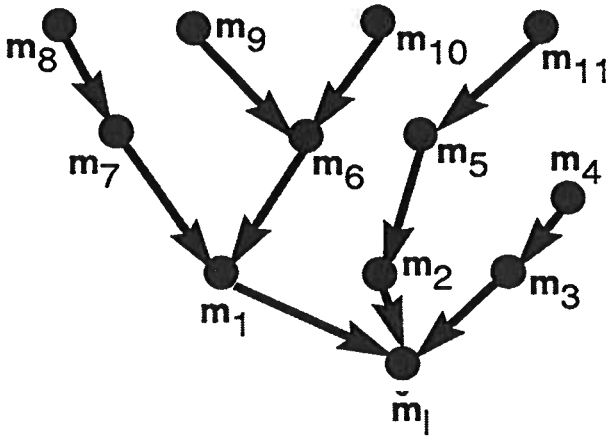

FIG. 1.1. Illustration of the directed-tree model used to represent the concept of basins of attraction.

Definition 4 (Width of a Basin of Attraction) *Let $\{B_l\}_{l=1,\ldots,n}$ be the basins of attraction as in Definition 3. Let the number of models in the set B_l be $|B_l|$. The width of the lth basin of attraction is defined as*

$$p_l = \frac{|B_l|}{M} \quad l \in [1, n]. \tag{1.8}$$

Notice that $\{p_l\}_{l=1,\ldots,n}$ forms a probability distribution since $\sum_{l=1}^{n} p_l = 1$.

Basins of attraction can be represented by directed-trees (Hu *et al.*, 1994). Figure 1.1 illustrates such a model with one basin of attraction. The tree has a unique root corresponding to the local minimum $\check{\mathbf{m}}_l$. The vertices on the tree are models that belong to the basin of attraction. Each of these vertices has a unique path leading to the root $\check{\mathbf{m}}_l$. The leaf vertices are the models with highest function values within the basin of attraction. It is easy to see that the number of vertices on the lth tree is proportional to the width of the corresponding basin of attraction. It is interesting to note that the tree may have different paths connecting the vertices, if different local-descent transition operators ($\mathbf{T}_{local}$) are used. However, the root, leaves, and all other vertices of the basin of attraction B_l are independent of local-descent algorithm used. Therefore, in this thesis I neglect the differences caused by local-descent algorithms and use the model of basins of attraction to represent objective functions. The numerical results of local-descent searches are produced by a non-linear conjugate-gradient algorithm (Deng *et al.*, 1996a; Deng *et al.*, 1996b). It is also important to notice that the graph-based representation is applicable to functions of any number of variables.

Suppose a local-descent search is performed with infinite precision and infinitely long computing time, then this search converges to an exact local minimum; such a search is referred as an *ideal local search*. According to Definition 3, the lth basin

of attraction is the subset of models from which the ideal local searches converge to the lth local minima. Imagine the objective function being a hyper-surface in an N-dimensional model space and an infinitesimal ball is randomly placed on this surface. Suppose there is no friction between the surface and the ball so that the ball rolls downwards on this surface. After a long enough time, the ball will stop at the local minimum $\breve{\mathbf{m}}_l$, if it was originally placed at any point $\mathbf{m}_i \in B_l$. Figure 1.2 illustrates such a "ball-rolling" procedure using 12 balls on a one-dimensional function with three basins of attraction. The balls are rolling downhill on the function surface at time $t1$. Finally by $t2$, all balls have stopped at the corresponding local minima; the number of these balls stopping at each of the local minima would be 3, 2, and 7, respectively, corresponding to the widths of the basins of attraction. This figure illustrates that if one ball is positioned randomly with uniform probability in this model space, the probability of the ball stopping at the lth local minimum is the width of the lth basin of attraction, p_l. In other words, rolling a randomly chosen ball downhill corresponds to tossing a weighted n-sided die, where n corresponds to the number of local minima, and the weight given to each side of the die corresponds to the width of that basin of attraction.

Let K_l, $l \in [1, n]$ be the random variables representing the number of balls rolling down to the lth local minimum when K balls are started at random positions in model space. Then the probability that $K_l = k_l$ for each $l \in [1, n]$ is

$$f(k_1, \ldots, k_n) = \frac{K!}{k_1! \cdots k_n!} p_1^{k_1} \cdots p_n^{k_n}, \tag{1.9}$$

where $\sum_{l=1}^{n} k_l = K$. For each $l \in [1, n]$, the mean value of the random variable K_l is $E[k_l] = K\, p_l$.

1.1.3 A Markov Model for Global Searches

Local-descent searches can be used to explore local features of objective functions. For global searches, however, some kind of stochastic process is useful so that the search can jump across the trees of basins of attraction. Broadly speaking, almost all global methods are some kind of *Monte Carlo* search. For most of these Monte Carlo processes, the current set of models, $\vec{\mathbf{m}}_t$, depends only on that of the previous generation, $\vec{\mathbf{m}}_{t-1}$. Such Monte Carlo processes can be described by *Markov chains*. When the transition operator $\mathbf{T}(t)$ is independent of the time step t, the searching process forms a *stationary* Markov chain; otherwise the procedure would be a *non-stationary* Markov chain.

For a combinatorial problem, let $\{\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_M\} = \mathcal{M}$ and $M = |\mathcal{M}|$. Then if the search algorithm is associated with a Markov chain, the transition operator $\mathbf{T}(t)$

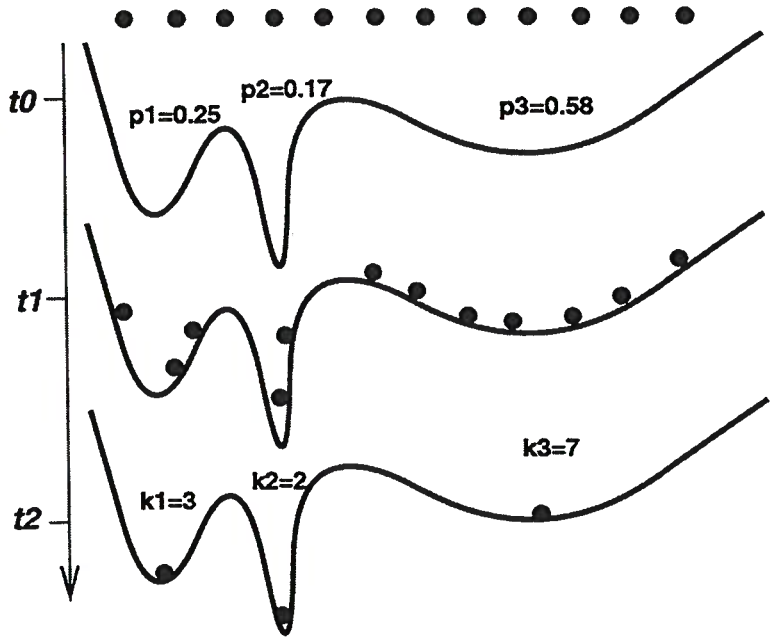

FIG. 1.2. Illustration of the "ball-rolling" experiment for a one-dimensional function with three basins of attraction. At the initial time step $t0$, 12 balls are positioned uniformly across the model space. The balls are rolling downhill on the function surface at time $t1$. Finally by $t2$, all balls have stopped at the three local minima. The number of balls stopped at each local minimum is proportional to the width of its associated basin of attraction.

of Algorithm 1 becomes a *stochastic matrix*[3]. This $M \times M$ matrix has elements of

$$T_{ij}(t) = Pr(\mathbf{m}_i|\mathbf{m}_j), \tag{1.10}$$

which is the probability of making a transition from the ith to the jth model at time step t. Each row sum of the transition matrix $\mathbf{T}$ is unity, that is $\sum_{j=1}^{M} T_{ij}(t) = 1$, and all the non-zero elements of the ith row $N(i)$ form the *neighborhood* of the ith model. Most of the widely used search methods defined via Algorithm 1 can be described by this Markov model.

A simple kind of Markov search would be to choose a model at each time step with uniform probability on $\mathcal{M}$, independent of the current model. This forms a Markov chain in which the the neighborhood of each point is all of $\mathcal{M}$. I refer to this kind of search as *Uniform Monte Carlo* (UMC). All elements of the transition matrix $\mathbf{T}$ have equal values, *i.e.*, $T_{ij} = \frac{1}{M}$, $\forall i, j \in [1, M]$. Therefore, the decision of choosing the next set of models is independent of the previous generation. If there are N parameters and each of them can take n possible values, the probability of finding a particular model is proportional to n^{-N} for each function evaluation.

While UMC does not learn anything from its history, the local-descent methods are slaves of their history. Many global search strategies have been developed to yield a compromise between the two extremes (two comprehensive surveys of global optimization are Törn & Žilinskas (1989) and Scheoen (1991)). Almost all of these global strategies incorporate some stochastic elements, especially in the construction of transition operators. It is important for the success of a global search to make the best use of information provided by the previous samples while avoiding being trapped in basins of attraction. Among all these strategies, the most widely used are *Simulated Annealing* (SA) (Kirkpatrick *et al.*, 1983), *Genetic Algorithms* (GA) (Holland, 1975), and random hill-climbing (RHC). SA and GA searching strategies use stochastic transition operators $\mathbf{T}$ that are biased to a certain degree towards good samples from the previous generations. On the other hand, RHC-searches use deterministic transition operators while the initial models are chosen randomly according to a certain probability.

Basically, SA is a non-stationary Markov chain with a probability matrix as the transition operator (Aarts & Korst, 1989; van Laarhoven & Aarts, 1987). The element in the ith row and jth column represents the probability of transition from the ith to the jth model (Morey, 1996; Morey *et al.*, 1996):

$$T_{ij}(t) = \begin{cases} 0, & \text{if } j \notin N(i); \\ \frac{1}{|N(i)|}, & \text{if } j \in N(i) \ \& \ F(\mathbf{m}_j) \leq F(\mathbf{m}_i); \\ \frac{1}{|N(i)|} \exp(\frac{F(\mathbf{m}_i) - F(\mathbf{m}_j)}{c(t)}), & \text{otherwise.} \end{cases} \tag{1.11}$$

where $c(t)$ is a temperature-like parameter that decreases with time t, and $N(i)$ is

[3] A stochastic matrix are such that the summation of all elements in each row is unity.

the neighborhood of the ith model. These two parameters are the control factors for the convergence of SA. If the temperature $c(t) \to 0$, then the process becomes a local search. On the other hand, a UMC process corresponds to another extreme wherein $c(t) \to \infty$ and $|N(i)| = M$. The convergence theory of SA is based on the analysis of this Markov process (Hajek, 1988).

A GA is an evolution process where the objective functions for a population of models are evaluated simultaneously and new models are generated by applying some random operations to the current population of models (Holland, 1975; Goldberg, 1989; Whitley, 1993). GAs can also be considered under the framework of the Markov chain representation, but the transition matrix is relatively complicated. Davis and Principe developed the Markov model for simple GAs, from which they have developed the convergence theory (Davis, 1991; Davis & Principe, 1991).

RHC-searches, on the other hand, apply deterministic transition operations $\mathbf{T}_{local}$ to a randomly chosen population of initial models $\vec{\mathbf{m}}_0$, where $K \gg 1$. This method is, in fact, a set of independent local-descent searches from randomly chosen initial models. These searches are the simplest instances of the *random multistart* methods (Scheoen, 1991; Hu *et al.*, 1993). In the framework of the generalized Markov chain, the deterministic transition operator $\mathbf{T}_{local}$ is a probability matrix of size $M \times M$, where each row has one non-zero element. Therefore,

$$T_{ij}^{t} = \begin{cases} 1, & \text{if } j \in N(i) \ \& \ F(\mathbf{m}_i) > F(\mathbf{m}_j); \\ 0, & \text{otherwise,} \end{cases} \tag{1.12}$$

which is a special case of equation (1.11).

Suppose we label the local minima in order of function values: $F(\breve{\mathbf{m}}_1) \leq F(\breve{\mathbf{m}}_2) \leq \ldots \leq F(\breve{\mathbf{m}}_n)$ in the direct-tree graph representation. The first set B_1 corresponds to a global minimum. As an example, Figure 1.3 shows this graph-based representation of a function with two basins of attraction: one global minimum $\breve{\mathbf{m}}_1$ and one local minimum $\breve{\mathbf{m}}_2$. For the situation depicted in Figure 1.3, the width of the basin of attraction associated with the global minimum is smaller than that of the local minimum.

The transition matrix of the local-descent search of the objective function shown in Figure 1.3 would be a 12×12 block matrix,

$$\mathbf{T}_{local} = \begin{pmatrix} P_1 & Z \\ Z^T & P_2 \end{pmatrix}, \tag{1.13}$$

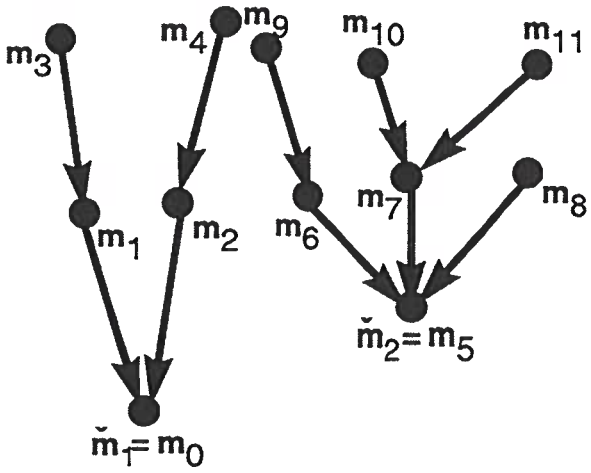

FIG. 1.3. The directed-tree model for an objective function of 12 models with two basins of attraction. The width of the basin of attraction associated with the global minimum has a smaller width than that of the local minimum.

where

$$P_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \quad P_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{1.14}$$

and Z is a matrix of zero elements with the size of 5×7. The columns with all zero elements in equation (1.14) correspond to the leaf vertices on the directed-trees, and the first columns correspond to the roots of each tree.

The initial population in an RHC is usually chosen *uniformly at random*, by which I mean that each parameter is randomly chosen under a uniform probability distribution within its range. The stopping criterion is that either the magnitude of the gradients are reduced to small enough value ϵ or that the maximum number of iterations *cmax* is reached. Such an RHC algorithm can be described in the context of Algorithm 1 as follows.

Algorithm 2 Random Hill Climbing ($\vec{\mathbf{m}}^* = \mathrm{RHC}(F, K, \epsilon, cmax)$)
Let K be the size of the initial population, and let the stopping criterion S be that either gradients of all samples are reduced to ϵ or the number of iterations reaches cmax. Let $\mathbf{T}_{local}$ be the local-descent transition operator.

1. *Choose K initial models $\vec{\mathbf{m}}_0$ randomly with a uniform probability distribution:*

2. *Apply Algorithm 1,* $\vec{\mathbf{m}}^* = \mathrm{GS}(F, \vec{\mathbf{m}}_0, \mathbf{T}_{local}, S)$.

The final population contains K *converged models,* $\vec{\mathbf{m}}^*$.

1.2 What Makes an Optimization Problem Hard

1.2.1 A Catalog of Two-Dimensional Objective Functions

Figure 1.4 illustrates some of the variety of objective functions that could be encountered in practical applications. For the visualization purpose, these two-dimensional functions are maximization problems.

- Function A shows an objective function that can be easily optimized by using some well-established local-descent search methods, such as Conjugate Gradient, Quasi-Newton, Downhill Simplex, *etc.* (Fletcher, 1987; Dennis & Schnabel, 1987). The convergence model is independent of the starting model of the optimization.

- B suggests that the function has many local extrema, but they may not represent important features of the problem. The results of the same local-descent algorithms could be any of the local extrema, depending on the starting models. In explorational geophysics, such situations are mostly caused by noise or other non-essential factors. Instead of using full-scale Monte-Carlo global searches, if we could somehow smooth the objective function, then the problem would be similar to that for function A.

- Function C represents types of problem that contain a small number of significant local extrema. It may be necessary to find all local/global extrema. Simply smoothing function C may cause the loss of important information. When the number of local maxima is not large, a set of local-descent searches may be used to explore some sub-divided regions of the function.

- Function D represents a problem with fundamental ambiguities. The null space of the problem is expressed as a flat region on the function surface. None of the search algorithms can help us determine the optimal model. It may be important that more information is needed for reducing the ambiguity of the problem.

- Function E suggests a problem containing a large number of significant local extrema, almost all of which represent some essential information about the problem. Some sophisticated Monte-Carlo searches, such as Simulated Annealing (SA) and Genetic Algorithm (GA), might be useful for such problems.

- Function F has an essentially flat landscape except for an isolated sharp maximum. For such cases, the global structure provides no information for searches; none of the "smart" searching strategies can be expected to work better than a pure random search or an exhaustive evaluation.

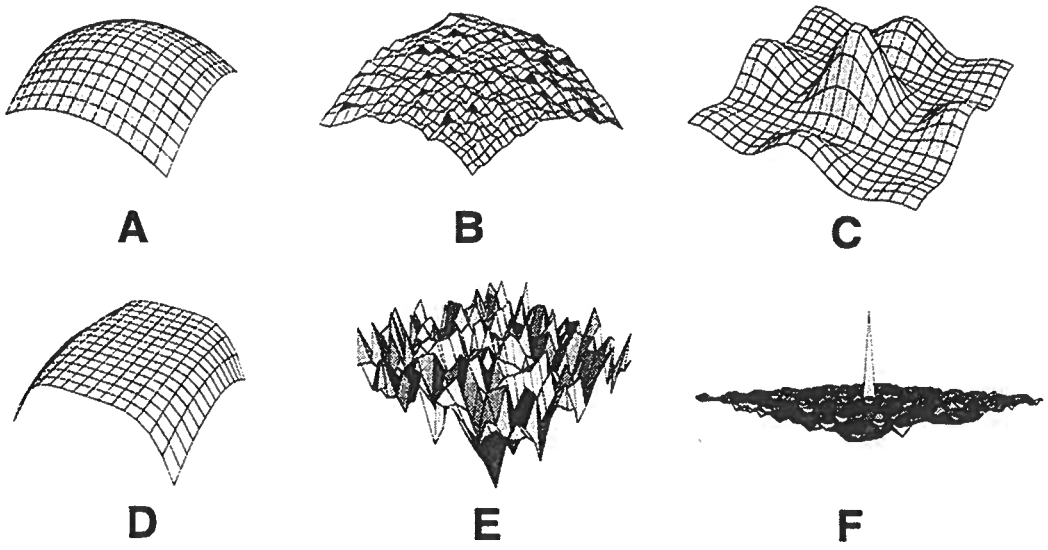

FIG. 1.4. The objective functions of two-dimensional maximization problems representing the diverse nature of such problems.

Clearly, it is important to understand the characterizations of various kinds of problems, so that appropriate strategies for optimization can be developed.

1.2.2 Why Develop a Measure of Complexity?

For treating a wide varieties of objective functions as shown in Figure 1.4, many global optimization algorithms have been developed. A great deal of research has been done on the comparison of various global optimization methods. Unfortunately, these comparisons are mostly either empirical or conclusions are drawn from the study of a small class of special problems. While some proofs of asymptotic convergence of global optimization strategies are available (Hajek, 1988; Davis, 1991), it is not clear how these results can be applicable to a wide variety of problems within a finite computing time.

Wolpert and Meagready (1995) have shown that all global optimization algorithms have exactly the same performance when averaged over all possible objective functions. Known as the *No Free Lunch* (NFL) theorem, this result implies that there does not exist a universal optimization strategy independent of the nature of the problem. In their recent work, Culberson (1996) and Radcliffe-Surry (1996) also show that performance of any global optimization strategy depends greatly on the specialty of the problem. Therefore, the blind faith of developing a "once-for-all superior" global algorithm cannot be justified.

For treating a wide variety of optimization problems of different nature, it is necessary to have a large collection of optimization algorithms. The software library

COOOL was developed for such purpose (Deng *et al.*, 1996a; Deng *et al.*, 1996b). Appendix A gives a description of the library, as well as its design within an object-oriented programming paradigm.

Meagready and Wolpert (1995-1996) found that while no optimization problem is intrinsically harder than others when averaged over all possible search algorithms, for any given optimization problem there exist optimal search strategies for solving the particular problem. Unfortunately, there is no generally agreed strategy for choosing appropriate algorithms. In order to be able to usefully compare different optimization techniques, it is **necessary** that we have a meaningful way of characterizing the problems. Such a characterization is referred to a measure of *complexity* of objective function or *hardness* of optimization problem.

1.3 Overview of the Thesis

As a first step in studying the hardness of generic optimization problems, most of this thesis is devoted to developing a practical tool for characterizing objective functions, especially when the dimensionality is high.

From the experience with a catalog of test problems, I find that the performance of any optimization procedure is influenced mostly by the topographical surface of the objective function. Therefore, the nature of an optimization problem can be represented by the complexity of the topography of the objective function. In the following chapter, I propose an entropy-based criterion for characterizing the topography of high-dimensional functions. The proposed measure takes into account important information on topographical features, such as the number of local minima, the widths of basins of attraction, and the depths of these basins of attraction. Such a measure aims to represent the important global feature of objective functions, independent of any search algorithm. As a practical matter, I then discuss the statistical estimation of this entropy measure by repeatedly applying random local-descent searches in the model space. Finally, I show some numerical examples for analytic test functions.

In Chapter 3, I discuss the error caused by numerical and sampling limitations. Both finite sampling and numerical implementation contribute to the error of final result of this complexity analysis. First, statistical issues are investigated. Confidence-interval analysis is used to bound the error caused by finite sampling. Second, the estimated complexity values could be dependent on the local-descent algorithm and the computation time allowed. For the purpose of the complexity analysis, however, it is not necessary that local-descent searches converge all models to exact local minima. The error caused by numerical inaccuracy will be important only when the error of imperfect searches exceeds the size of the smallest basins of attraction. In addition, while numerical inaccuracy can cause discrepancies between the estimated and exact complexity, I show that this search-dependent error can sometimes help us to evaluate the ill-posedness of the problem.

As an example of a hard optimization problem in explorational geophysics, Chapter 4 is devoted to the study of the seismic residual-statics-estimation problem. By formu-

lating the high-dimensional, stacking-power function as an objective function, the task becomes a global optimization problem. This problem is difficult because the number of parameters is large, and the oscillatory nature of seismic signals causes the objective function to be highly multimodal. After studying special properties of this problem, I propose two approaches for this optimization problem. First, I apply a shift-invariant multi-resolution analysis (MRA) to simplify the objective function. I use the complexity analysis to determine the optimal level of simplification. The second approach is to use an envelope of the multimodal objective function. Both approaches greatly reduce the number of local extrema. The complexity analysis developed in Chapter 2 is used to support this claim.

Finally in Chapter 5, I summarize the thesis and give some suggestions for possible further research following this work.

Appendix A is a document on the CWP Object-Oriented Optimization Library (COOOL), the software primarily used for this thesis work. I describe the object-oriented design strategy of the library and optimization algorithms included in the library so far. In Appendix B, I give some background knowledge of the MRA that I have used in Chapter 4.

In Appendix C, I discuss some practical issues of residual-statics estimation in explorational geophysics. In particular, the objective function is formulated under the widely used surface-consistent assumption. In addition, I give extensive discussions on two conventional residual-statics approaches, and investigated their advantage and disadvantages in accordance with the physical insights provided in Chapter 4. Finally, I present the residual-statics correction of two field data based on the envelope approach, as described in Chapter 4.

Chapter 2

ENTROPY-BASED COMPLEXITY ANALYSIS

2.1 Measures of Performance for Optimization Problems

For local optimization, the performance is generally measured by the *rate of convergence*[1] (Fletcher, 1987). Global optimization, however, does not have a generally agreed on performance measure. Here, I list several recent-developed criteria for comparing global optimization. Wolpert and Magready (1995) suggested analyzing the distribution of function values at the converged models, the algorithms that converge to models with low function-values are considered good. With this criterion, the authors proved the No Free Lunch (NFL) theorem (see page 11). In the framework of Markov chains, Shonkwiler and Van Vleck (1994) propose using the *expected hitting time* (EHT)[2] for measuring the performance of a global algorithm. The EHT is used to study the speedup of the convergence rate for independent identical processors (IIP) (Shonkwiler & Van Vleck, 1994; Hu *et al.*, 1993), and to develop a methodology for dynamical adjustment of searching parameters for SA searches (Morey, 1996). For studying the performance of GAs, Jones (1994) uses a directed-graph to represent what he calls "landscape" of GA procedures.

None of the above measures, however, is generic enough to be useful in designing optimal search strategies. To usefully compare different global-search methods, a meaningful way of characterizing objective functions needs to be developed. In this chapter, I go beyond the efforts of measuring performance of algorithms. Instead, I develop a criterion for representing the complexity of the problem itself, independent of searching algorithms. From the discussions in Section 1.2 and inspection of Figure 1.4, we sense that the complexity of an optimization problem can be represented by some topographic features of the objective function. Similar issues are also studied by Törn and Žilinska (1989) and Kaufmann (1993). Kaufmann summarized the situation for SA and GA by the following (Kaufmann, 1993):

> Annealing works well only in landscapes in which deep energy wells also drain wide basins. It does not work well on either a random landscape or a "golf course" potential, which is flat everywhere except for a unique "hole". In the latter case, the landscape offers no clue to guide the search.
>
> Recombination (in GAs) is useless on uncorrelated landscapes but useful under two conditions (1) when the high peaks are near one another and

[1] The rate of convergence measures how rapidly the trial models converge to the local optima.
[2] EHT is defined as the expected number of steps of reaching a particular goal-state (extremum).

> hence carry mutual information about their joint locations in genotype space and (2) when parts of the evolving system are quasi-independent of one another and hence can be interchanged with modest chances that the recombined system had the advantage of both parents.

On the other hand, since RHC uses local descent transition operators $\mathbf{T}_{local}$, the performance is even more strongly influenced by the topography of the objective function. I consider the most influential topographic features for optimization to be the following three: the number of local extrema, the widths of the corresponding basins of attraction, and the depths of these basins. These topographic features of an objective function are generally referred to as the *landscape* of this function. Characterizing this landscape is an important step toward understanding the complexity issues in optimization.

For many practical applications, objective functions cannot be expressed in closed forms in terms of elementary functions, but can only be evaluated point-wise by computer programs. Little is known about such objective functions because of the lack of the global knowledge. Furthermore, while it is still possible to examine all possible models in the model space when the dimension is low ($N \leq 2$), this situation is especially serious for high-dimensional problems.

One natural characterization of the landscape is the *spatial correlation* (Weinberger, 1990; Stadler, 1992). Some prototyped combinatorial optimization problems have been investigated through study of the correlation in landscapes: the Traveling Salesman Problem (TSP) (Stadler & Schnabl, 1992), the graph-bipartitioning problem (Stadler & Happel, 1992), and the NK model problems, a spin-glass-like problem in biology (Kauffman & Weinberger, 1989). Using the spatial correlation of an objective function as a criterion, these authors study the effectiveness of particular global algorithms as a function of the correlation length.

On the other hand, local minima and their associated basins of attraction can be used to directly characterize surface topography. Berry and Breitengraser-Kunz (1995) studied topography and dynamics of multidimensional inter-atomic potential surfaces by analyzing a population of local minima, each of which has two saddle points connected to it. By connecting these samples in a certain order, the high-dimensional function surface is represented by a series of lines. In this way, the important features of the landscape are represented by the distribution of the primary, secondary or tertiary basins of attraction (Berry & Breitengraser-Kunz, 1995).

Alternatively, complexity of high-dimensional Hamiltonians has also been studied by means of *entropy* (Falcioni *et al.*, 1995). For an N-dimensional Hamiltonian, some local extrema are found by some means[3]. The width of each basin of attraction p_i is

[3]In (Falcioni *et al.*, 1995), the authors did not indicate the method used of finding the local extrema.

then estimated by the product,

$$p_i \propto \Delta^{(i)}(N) \equiv \prod_{k=1}^{N} \delta_k^i. \tag{2.1}$$

Here, δ_k^i is the largest distance along the kth coordinate for the models belonging to the ith basin of attraction. Falcioni *et al.* (1995) characterize the complexity of the N-dimensional surface by the following entropy expression,

$$S(N) = -\sum_i p_i \ln p_i. \tag{2.2}$$

Using $S(N)$ to characterize a high-dimensional surface is attractive in the sense that it incorporates the concept of information (entropy) in the characterization. However, it lacks an important factor that influences the hardness of optimization - the depths of the basins of attraction. Recall that Figure 1.4 in Chapter 1, the function B and E seem to have the same number of local extrema and similar widths. These two objective functions, however, have very different topographical features, and different strategies are required for maximizing these two functions. In addition, the estimation of the width of basins of attraction p_i by equation (2.1) assumes that the geometric shape of all basins of attraction are N-dimensional hyper-cubes, which is not true in general.

2.2 Entropy-Based Measure of Complexity

As mentioned above, the hardness of an optimization problem is closely related to the topographical complexity of the objective function. Such topographic features are characterized primarily in terms of the three basic factors mentioned in the previous section. Independent of the work reported in (Falcioni *et al.*, 1995), I have developed a similar entropy-based complexity measure. I estimate the widths of basins of attraction p_i by a statistical analysis of the results of RHC-searches instead of calculating the volumes of hyper-cubes as in equation (2.1). Furthermore, the contribution of each basin of attraction to the complexity is represented by a related probability q_i that takes into account both the widths and depths of basins of attraction, rather than the widths p_i solely.

Definition 5 (Entropy-Based Complexity) *Let $F : \mathcal{M} \subset \mathbf{R}^N \rightarrow \mathcal{Y} \subset \mathbf{R}$ be bounded and have n isolated local minima, where $1 \leq n < \infty$. Let $\{\check{\mathbf{m}}_i\}_{i=1,\cdots,n}$ be these distinct local minima, $\{y_i = F(\check{\mathbf{m}}_i)\}_{i=1,\cdots,n}$ be their corresponding function values, and p_i be the width of the ith basin of attraction, as in Definition 3.*

Define $\{q_i\}_{i=1,\cdots,n}$ to be a probability distribution

$$q_i \propto \begin{cases} p_i, & \text{if } \sigma = 0; \\ p_i\, e^{-\frac{y_i - y_m}{\sigma}}, & \text{otherwise,} \end{cases} \tag{2.3}$$

where $i \in [1, n]$, $y_m = \min_{i \in [1,n]} y_i$ *is the value at the global minimum, and*

$$\sigma = \frac{1}{n} \sum_{j=1}^{n} (y_j - y_m).$$

The entropy-based complexity is defined to be,

$$\begin{aligned} C_e &= -\sum_{i=1}^{n} q_i \ln q_i \\ &= \left\langle \ln \left(\frac{1}{q_i} \right) \right\rangle, \end{aligned} \tag{2.4}$$

where the angle brackets denote the expected value with respect to the probability distribution $\{q_i\}_{i=1,\ldots,n}$.

As an entropy quantity, C_e measures the "disorder" of the function surface, and it is always true that

$$0 \leq C_e \leq \ln n,$$

where n is the number of local minima. The probability q_i represents the *significance* of the ith local minimum. A high C_e value indicates a relatively large number of significant local minima. If all minima of a function are equivalent, *i.e.* n basins of attraction have the same widths and depths ($q_i = \frac{1}{n}, \quad \forall i \in [1, n]$), then C_e is $\ln n$, and C_e increases with the number of minima n. On the other hand, a low C_e implies that the function does not have many local minima or the local minima are not significant. For the latter case, all q_i corresponding to non-global minima are small ($q_i \ll 1$). Therefore, a significant local minimum corresponds to either a wide (*i.e.*, large p_i) or a deep (i.e. small $(y_i - y_m)/\sigma$) basin of attraction whose corresponding q_i is comparable to $\max(q_i)$. A unimodal function has the complexity value $C_e = 0$.

An alternative complexity could also be defined as a normalized quantity in Definition 5 so that $0 \leq C_e \leq 1$. However since both $\{p_i\}$ and $\{q_i\}$ are normalized probability distributions, for simplicity, I use the classic definition of entropy by Shannon (1948). In addition, complexity C_e encapsulates topographical information about the objective function into one value. Although C_e represents the function topography in a meaningful way, it is important to keep in mind that the topography about each of the basins of attraction can be analyzed individually with the available information of $\hat{p}_i$ and $\hat{q}_i$, $\forall i \in [1, n]$. This detailed information may provide addition understanding about the objective function.

It is important to notice that the complexity measure C_e in Definition 5 does not explicitly depend on the dimensionality N. Rather, it depends on the number of local minima (n), the widths of the basins of attraction (p_i), as well as the relative values of these local minima ($\exp(\frac{y_i - y_m}{\sigma})$). In addition, the complexity measure in Definition 5 has the following scale- and shift-invariant properties.

Theorem 1 *Let $C_e(F)$ and $C_e(G)$ be the complexity measures of objective functions F and G in the model spaces $\mathcal{M}_F$ and $\mathcal{M}_G$, respectively. Let $a, b \in \mathbf{R}$ be arbitrary constants, then we have the following,*

(a) If $F(\mathbf{m}) = a + bG(\mathbf{m})$ and their model spaces are the same, $\mathcal{M}_F = \mathcal{M}_G$, then

$$C_e(F) = C_e(G); \tag{2.5}$$

(b) If $F(\mathbf{m}) = G(a + b\,\mathbf{m})$, and the model spaces are related in the same way, $\mathcal{M}_F = a + b\,\mathcal{M}_G$, then

$$C_e(F) = C_e(G). \tag{2.6}$$

Theorem 1 shows that a linear transformation of either the model space or the objective function would not alter the complexity of the problem. These results can be easily verified from Definition 5. For the directed-tree representation (see page 4), functions that differ by a linear transformation of either domain or range have identical representations. Clearly, the invariant properties of Theorem 1 are desirable for the complexity criterion.

Figure 2.1 shows two one-dimensional functions,

$$\begin{aligned} f_1(x) &= 0.5 - |\cos(3\,x)| \\ f_2(x) &= 0.5 - |\cos(3\,x)|\, e^{-0.1\,x^2}, \end{aligned} \tag{2.7}$$

where the model space $\mathcal{M} = [-5, 5]$. Both functions have the same number of local minima and the same widths of the basins of attraction. Function $f_1(x)$ has identical basins of attraction, while $f_2(x)$ has a dominant global minimum at $x = 0$ and decreasingly significant local minima away from the center. Suppose $f_1(x)$ and $f_2(x)$ are two misfit functions and $x = 0$ is the goal value for both problems, it is easy to see that achieving this goal for $f_1(x)$ is more difficult than that for $f_2(x)$. The complexity measure of Definition 5 gives a higher C_e value to $f_1(x)$ ($C_e = 2.2$) than to $f_2(x)$ ($C_e = 1.5$).

2.3 Statistical Complexity Analysis

In most practical problems, the objective function can be evaluated only pointwise. Even for many analytical functions, the location of all stationary points cannot be computed easily when the dimensionality is high. Under such circumstances, some degree of global sampling is essential to estimate the information we need for the complexity analysis. As discussed earlier, RHC-searches described in Algorithm 2 can explore various regions of the model space and take initial samples downhill to the bottoms of basins of attraction. Therefore, a statistical analysis of results of RHC-searches may be used to estimate the topographic quantities.

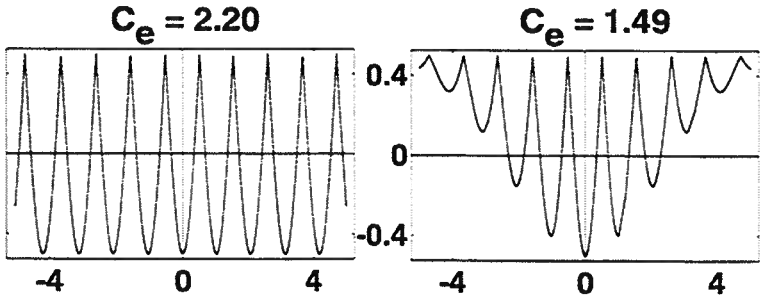

FIG. 2.1. Two one-dimensional functions with the same number and widths of basins of attraction. The function on the left has complexity of $C_e = 2.2$, while the function on the right a complexity of $C_e = 1.5$.

If the local-descent search used in the RHC is ideal (see page 4), the number of models converging to each local minimum k_i from K randomly chosen initial models has a multinomial probability distribution (see page 5). If these K initial models are chosen uniformly, the multinomial distribution of equation (1.9) is the joint probability of these random variables. For such multinomial distributions, the mean value for each random variable is proportional to the width of the corresponding basin of attraction, *i.e.* $E[k_i] = K p_i$, $\forall i \in [1, n]$. Therefore, the clustering of the converged models that result from an RHC can be used to estimate the topographical features needed in Definition 5: the number of distinct converged models estimates the number of local minima (n), and the portion of models converging to the ith local minimum (k_i/K) estimates the width of the ith basin of attraction (p_i). Using the "ball-rolling" experiment shown in Figure 1.2, we randomly place K friction-free, infinitesimal balls with uniform probability onto the function surface. Then, we analyze the distribution of the balls after all of them have completely stopped (in principle, after infinitely long time). Therefore, the algorithm for estimating the entropy-based complexity measure is described as follows.

Algorithm 3 Estimating Entropy-Based Complexity
Let $F : \mathcal{M} \subset \mathbf{R}^N \to \mathcal{Y} \subset \mathbf{R}$ be bounded and have a finite number of isolated local minima. The entropy-based complexity of this objective function $C_e(F)$ is estimated by the following steps:

(a) Uniform random hill-climbing (RHC):
Choose K initial models randomly with uniform probability in the model space, perform an RHC algorithm as in Algorithm 2, $\vec{\mathbf{m}}^* = \mathbf{RHC}(F, K, \epsilon, cmax)$[4]

(b) Clustering and evaluating:
Suppose among the resulting models $\vec{\mathbf{m}}$ of the RHC, there are $\hat{n}$ distinct

[4] F is the objective function, K is the number of models in the initial population, ϵ is the tolerance on gradients, and *cmax* is the maximum number of iterations allowed in each local-descent search.

ones $\{\check{\mathbf{m}}\}_{i=1,\cdots,\hat{n}}$, *each of which occurs with frequency of* k_i, $i \in [1, \hat{n}]$, *so that* $K = \sum_{i=1}^{\hat{n}} k_i$. *Evaluate values of the objective function at these distinct models,* $\{y_i = F(\check{\mathbf{m}}_i)\}_{i=1,\cdots,\hat{n}}$.

(c) *Estimating the complexity:*
Let $\hat{q}_i$ *be a probability distribution defined as*

$$\hat{q}_i \propto \begin{cases} \frac{k_i}{K}, & \text{if } \sigma = 0; \\ \frac{k_i}{K}\, e^{-\frac{|y_i - y_m|}{\sigma}}, & \text{otherwise,} \end{cases} \tag{2.8}$$

in which $y_m = \min\{y_i\}$ *and* $\sigma = \frac{1}{\hat{n}} \sum_{j=1}^{\hat{n}} |y_j - y_m|$. *Then, the estimated complexity* $\hat{C}_e$ *is*

$$\begin{aligned} \hat{C}_e &= -\sum_{i=1}^{\hat{n}} \hat{q}_i \ln(\hat{q}_i) \\ &= \left\langle \ln\left(\frac{1}{\hat{q}_i}\right) \right\rangle. \end{aligned} \tag{2.9}$$

The quantity $\hat{C}_e$ obtained from Algorithm 3 is a sample estimate of C_e of Definition 5. Since C_e characterizes topographical features of the objective function, it is independent of numerical computation and any searching techniques. $\hat{C}_e$, however, may be influenced by many practical issues. The discrepancy between C_e and $\hat{C}_e$ has two causes: shortcomings in both the statistical sampling and the numerical implementation.

First, accuracy of the statistical estimation is limited by the finite samples K used in the RHC. Some detailed confidence-interval analysis will be given in the next chapter. On the other hand, ideal local searches are not possible in reality due to numerical factors. If, for example, the curvature of the function is nearly zero, which is equivalent to an ill-conditioned Hessian matrix, gradient-based local descent searches may not be able to converge to the exact local minima within a reasonable amount of time. The estimated value of complexity $\hat{C}_e$ in such a situation may be higher than the true complexity C_e. Hence, a high estimated complexity value $\hat{C}_e$ maybe caused by either a large number of local minima or the flatness of the function surface. In practice, however, it is often difficult to distinguish the results of such ill-conditioning from multi-modality. Therefore, taking such numerical issues into account can represent an important aspect in characterizing the difficulty of optimization. Detailed discussions on these numerical issues will be given in the following chapter.

2.4 Numerical Examples of Complexity Analysis

In this section, I show examples of applying the complexity analysis to some analytical functions. First, some two-dimensional functions are examined. For 2-D functions we can confirm the statistical analysis graphically. On the other hand,

when the number of dimension is high, this is not possible. The statistical complexity analysis is essential for understanding the behavior of such functions. As an illustration of these ideas, I also study the behavior of the Griewank function (Griewank, 1981) as the dimensionality increases.

2.4.1 Complexity of a Family of Two-Dimensional Functions

Let us consider a family of two-dimensional functions,

$$F(m_0, m_1) = a\,(m_0^2 + m_1^2) - \cos(f\,\pi\,m_0) - \cos(f\,\pi\,m_1), \tag{2.10}$$

where $a, f \in \mathbf{R}$. This function is multimodal with quadratic long-wavelength components. The oscillation causes local minima whose number varies with the frequency parameter f. The quadratic trend, on the other hand, is determined by the curvature factor a. The oscillatory trend competes with the quadratic trend when the constants a and f vary. By appropriate choice of f and a, we can generate functions of widely varying complexity.

For the RHC algorithm used in this analysis, I choose the initial population to be $K = 500$. The stopping criterion is that either the magnitude of gradient is reduced to $\epsilon = 10^{-5}$ or the number of non-linear Conjugate Gradient iterations reaches $cmax = 200$.

Figures 2.2(a) and (b) show the function surface when $a = 0.1$ and $f = 1,\ 10$ respectively; both functions have the same quadratic term, but have different number of local minima in the same domain. Figures 2.2(c) and (d) show the converged models using the RHC-search when either of the stopping criteria are met. The converged models are more clustered at bottoms of basins on the low-frequency function surface than on the high-frequency one, and the corresponding complexities $\hat{C}_e$ are 2.14 and 3.93, respectively. Figure 2.3 shows the estimated complexity $\hat{C}_e$ as a function of spatial frequency f when the curvature $a = 0.1$ is fixed. We see that $\hat{C}_e$ increases with increasing oscillations in the domain.

Fixing the spatial frequency f, the spatial curvature a controls the significance of these same number of local minima. Figure 2.4(a) and (b) show the function surfaces of equation (2.10) when $f = 10$ and $a = 1,\ 10$ respectively; both functions have the same number of local minima as does in Figure 2.2(b). However, since the quadratic trend dominates the global feature in Figure 2.2(b), the oscillatory feature becomes less important, the local minima are not as significant as when a is smaller. This feature is reflected in the result of the RHC as shown in Figures 2.4(c) and (d). Compared with Figure 2.2(d), the converged models are more clustered with the increase of spatial curvature despite the fact that all three functions have the same number of local minima in the model space. The estimated complexity $\hat{C}_e$ of these two functions is 3.17 and 1.02, respectively. Figure 2.5 shows the estimated complexity values of the function in equation (2.10) for an increasing spatial curvature a when the spatial frequency is fixed at $f = 10$.

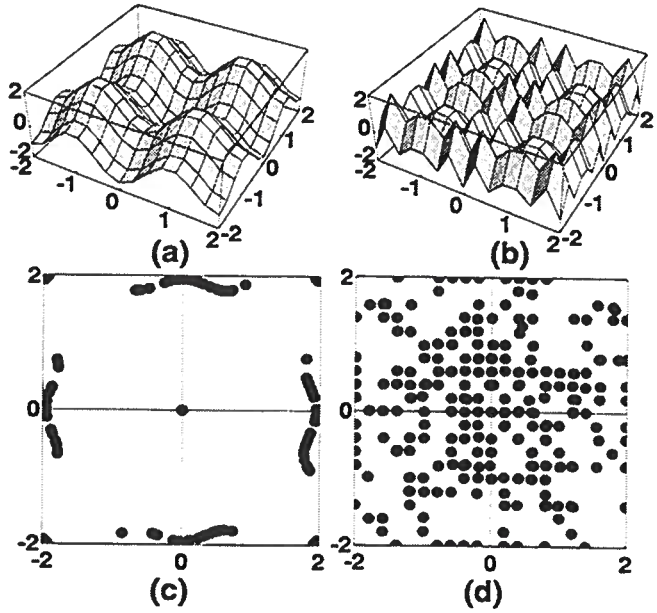

FIG. 2.2. Random hill-climbing with population of $K = 500$. (a) and (b) show the function surface defined in equation (2.10) when $a = 0.1$ and $f = 1,\ 10$ respectively. (c) and (d) show the convergence of 500 samples in the 2-D model space for functions (a) and (b). In the first case $\hat{C}_e = 2.14$ and second case $\hat{C}_e = 3.93$.

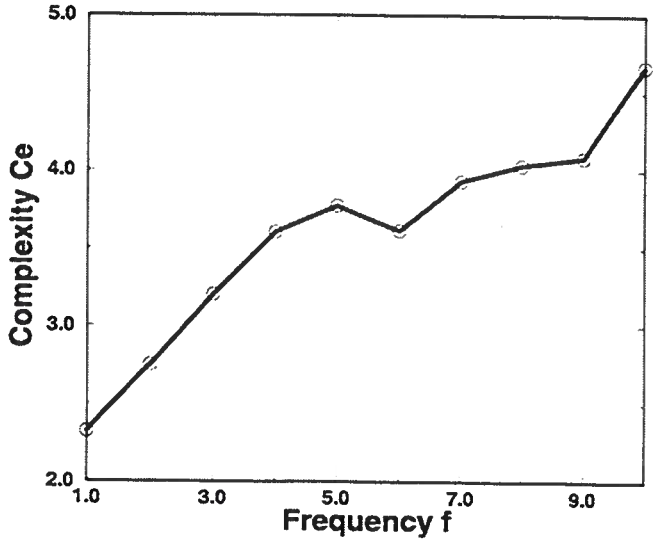

FIG. 2.3. Plot of $\hat{C}_e$ as a function of the spatial frequency f for the function defined in equation (2.10) when $a = 0.1$.

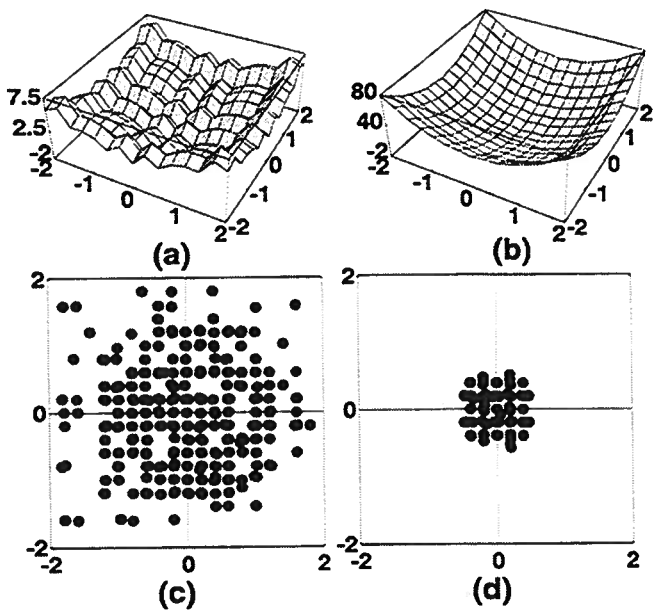

FIG. 2.4. Random hill-climbing with population of $K = 500$. (a) and (b) show the function surface defined in equation (2.10) when $f = 10$ and $a = 1,\ 10$ respectively. Both functions have the same number of local minima, though those in (b) are too small to be noticed. (c) and (d) show the convergence of 500 initial models in the 2-D model space for functions (a) and (b). In the first case, $\hat{C}_e = 3.17$, in the second $\hat{C}_e = 1.02$.

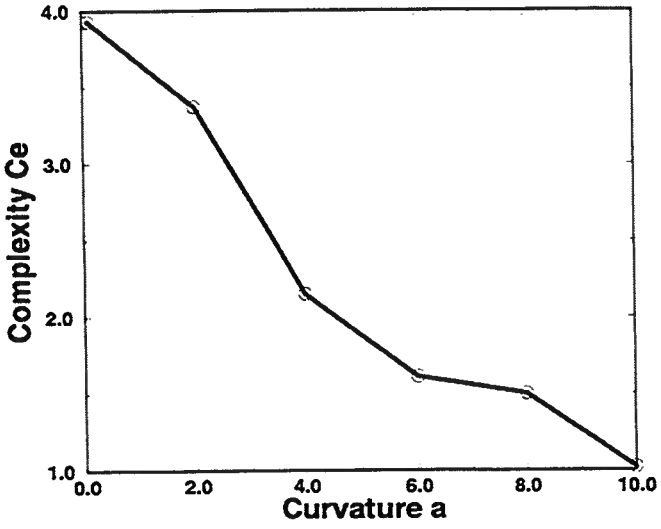

FIG. 2.5. Plot of $\hat{C}_e$ as function of the spatial curvature a for the function in equation (2.10) when $f = 10$.

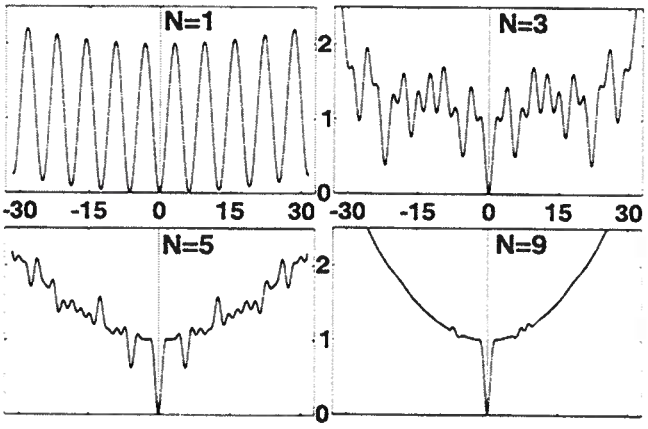

FIG. 2.6. Diagonal slices of N-dimensional Griewank functions.

2.4.2 N-Dimensional Griewank Functions

The Griewank function is a commonly used test function in optimization (Griewank, 1981):

$$g(\mathbf{m}) = 1 + \sum_{i=1}^{N} \frac{m_i^2}{4000} - \prod_{i=1}^{N} \cos\left(\frac{m_i}{\sqrt{i}}\right), \tag{2.11}$$

where the model space is an N-dimensional hypercube with $-a \leq m_i \leq a$, $\forall i \in [1, N]$. The cosine term in equation (2.11) produces many local minima; both the number and significance of these local minima vary with N and a. However, there is a unique global minimum at the origin of any dimension. It was noted by Törn and Žilinskas (1989) that there are some 500 local minima within the range of $-100 \leq m_i \leq 100$ when the dimension $N = 2$, while the number of local minima increases to up to some thousand if $N = 10$ and $a = 600$.

On the other hand, when studying the significance of these local minima, Whitley *et al.* (1995c; 1995a) pointed out

> ... the summation term (of $g(\mathbf{m})$) induces a parabolic shape while the cosine function in the product term creates "waves" over the parabolic surface. Thus, as the dimensionality of the search space (of $g(\mathbf{m})$) is increased the contribution of the product term involving the cosine becomes smaller and the local optima become smaller. The function becomes simpler and smoother in numeric space, and thus easier to solve, as the dimensionality of the search space is increased ...

Both intuitive speculations are reasonable at first sight. Furthermore, Figure 2.6 shows a one-dimensional slice of the Griewank function along the diagonal of the hypercube for dimensions 1, 3, 5 and 9, where values of all components of the models are equal. It seems that this partial information in the high-dimensional model space

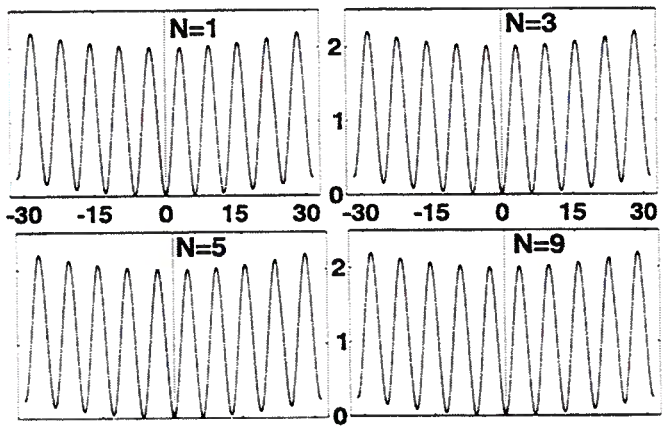

FIG. 2.7. Slices of N-dimensional Griewank functions. All variables but one are fixed at 0.

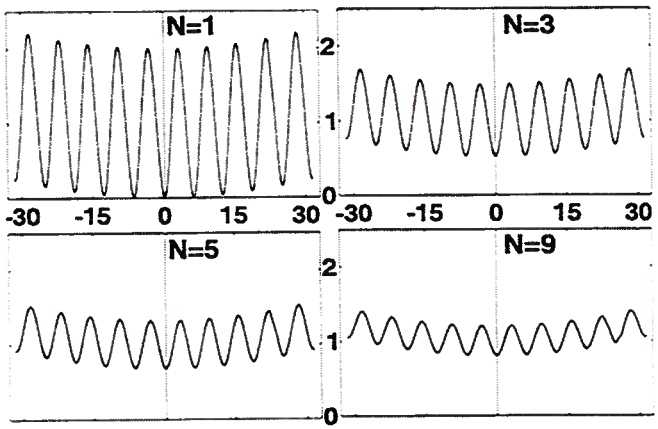

FIG. 2.8. Slices of N-dimensional Griewank functions. All variables but one are fixed at random between $[-2, 2]$.

confirms the previous conclusion that Griewank function becomes simpler when the dimensionality increases.

However, such pictures may be misleading since they tell us only about low-dimensional projections of the function. Figure 2.7 shows slices of the same functions when all but one variables are fixed to be 0. The increasing dimensionality does not change the oscillation around the global minimum. In addition, Figure 2.8 shows slices of this function when all but one variables are fixed and randomly chosen between $[-2, 2]$. These different slices seemingly give us different pictures about the global behavior of the Griewank function when the dimension increases. Therefore, studying the overall behavior of high-dimensional functions could be tricky. As far as I know, there has not been a systematic study on the number of minima in Griewank as a function of the dimensionality and their significance for optimization.

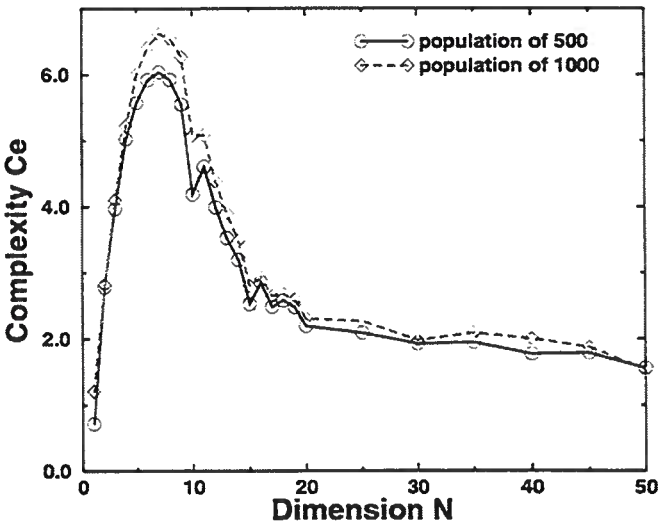

FIG. 2.9. Complexity $\hat{C}_e$ as a function of dimension N for the Griewank function with populations of 500 and 1000.

The complexity analysis developed in this thesis provides a means for study of the behavior of the Griewank function with increasing dimensionality. Figure 2.9 shows the estimated complexity values $\hat{C}_e$ as a function of the dimensionality. In the numerical computation, the initial populations are chosen to be $K = 500$ and 1000 in two independent experiments, and the model spaces are a N-dimensional hyper-cube where $-10 \leq x_i \leq 10$ for $i = 1, \ldots, N$. Both curves in Figure 2.9 give us consistent results that the complexity of Griewank function increases with number of dimensions till around nine dimensions, then it decreases as the number of dimension continuous to increase.

Given such consistency in results, I gain confidence in use of the entropy-based complexity as a criterion for understanding the dimensional-dependence of complexity of certain functions. This measure appears to be more useful than is study of hyper-planes. In the next chapter, I present a mathematical analysis of implementation issues with this complexity measure, as well as of errors caused by both the finite sampling and the finite arithmetic.

Chapter 3

IMPLEMENTATION ISSUES IN THE COMPLEXITY ANALYSIS

Implementation of the complexity analysis proposed in Algorithm 3 involves many practical issues. First, the statistical elements in Algorithm 3 are likely to introduce errors due to finite sampling. I calculate the confidence intervals for both the widths of the basins of attraction (p_i) and the estimated complexity ($\hat{C}_e$). These confidence intervals are functions of the number of samples (K), but do not explicitly depend on the number of dimensions (N).

In addition, the local-descent searches have finite precision, and only a finite number of iterations are allowed in reality. Therefore, the results also depend more or less on implementation issues, especially on the accuracy of local-descent searches in the RHC. In the final part of this chapter, I present an analysis of implementation issues and the errors. In particular, I show that errors associated with the flatness of the landscape may cause the estimated value of C_e to differ significantly from its exact value. I believe that this difference represents an important aspect of what makes an optimization problem hard.

3.1 How Many Samples Are Necessary?

The following analyses are based on the assumption that the RHC used in the complexity estimation is ideal. As defined in Chapter 1, an ideal RHC is a special case of Algorithm 2 where an infinitely large *cmax* is allowed and ϵ is infinitely small (see page 4). For such an ideal RHC, all K models converge exactly to one of the local minima at the end of the procedure. The number of converged models at each local minimum (k_i) has a multinomial probability, as in equation (1.9). The marginal distribution for each k_i has a binomial distribution,

$$b(k_i; K, p_i) = \frac{K!}{k_i!\,(K - k_i)}\, p_i^{k_i}\,(1 - p_i)^{K - k_i}, \tag{3.1}$$

$\forall i \in [1, n]$. From equation (3.1), the following results can be easily verified.

Lemma 1 *Let the objective function have n isolated local minima, where* $1 \leq n < \infty$. *Let* $\{p_i\}_{i=1,\ldots,n}$ *be the widths of the basins of attraction. Perform an ideal RHC with* K *randomly sampled models with uniform probability. Let* k_i *be the random variable representing the number of models converging to the ith local minimum. The probability of the ith local minimum being found* ($k_i \geq 1$) *is the following,*

$$Pr(k_i \geq 1) = 1 - (1 - p_i)^K. \tag{3.2}$$

(a) Equation (3.2) is a monotonically increasing function of p_i and K, where $0 < p_i \leq 1$.

(b) If $p_m = \min\{p_i\}$, then for a fixed K

$$Pr(k_m \geq 1) \quad \leq \quad Pr(k_i \geq 1),$$

$\forall i \in [1, n]$.

Lemma 1 implies that the basin of attraction with the smallest width is the most difficult to find. To reliably estimate the complexity, it is important to use a large enough number of samples K so that the minimum basin attraction can be found with a reasonably large probability. Any random variable with binomial distribution, $b(x; K, \theta)$, can be approximated by a normal distribution when the number of trials K is large. The general rule of thumb for such an approximation is (Freund, 1992),

$$K \geq \max(\frac{5}{\theta}, \frac{5}{1-\theta}). \tag{3.3}$$

If K satisfies the equality of equation (3.3), the probability of having at least one model converging to the basin of attraction of width θ is

$$P(\theta) = \begin{cases} 1 - (1-\theta)^{5/\theta}, & \text{if } 0 < \theta \leq 0.5 \\ 1 - (1-\theta)^{5/(1-\theta)}, & \text{if } 0.5 < \theta \leq 1. \end{cases} \tag{3.4}$$

Figure 3.1 shows such a probability $P(\theta)$ as a function of θ. The "kink" at $\theta = 0.5$ represents a discontinuity of the first derivative of $P(\theta)$. Nonetheless, equation (3.4) is a monotonically increasing function with asymptote of probability 1. Therefore if the inequality in equation (3.3) is satisfied, the probability of finding the basin of attraction of width θ is larger than 99% for any $\theta > 0$. The following lemma summarizes the above arguments.

Lemma 2 *Let an objective function have n isolated local minima, and $\{p_i\}_{i=1,\ldots,n}$ be the widths of the basins of attraction. Let $p_m = \min\{p_i\}$. In order to find all the basins of attraction with probability of higher than 99%, the number of initial samples K needs to satisfy*

$$K \geq \frac{5}{p_m}. \tag{3.5}$$

To get some ideas of the magnitudes of the population size, here is an example.

Example 1 Suppose an objective function has $p_m = 0.01$, then the initial population size K needs to be at least as $K > 500$. In general, for an objective function with $p_m = 10^{-n}$, then $K > 5 \times 10^n$.

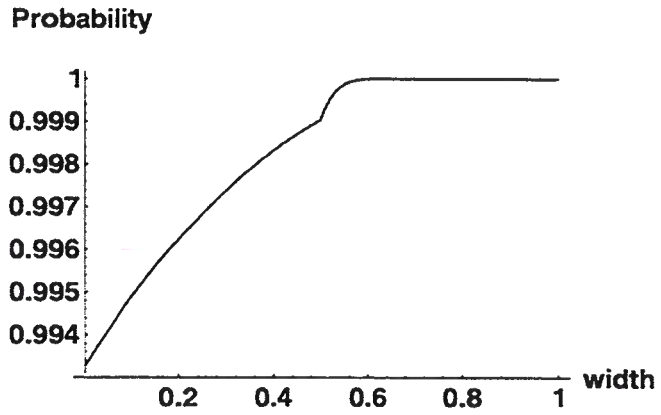

FIG. 3.1. The probability of having at least one model converge to a local minimum as a function of its width, when K satisfies the condition for approximating a binomial distribution by a normal distribution as in equation (3.3).

3.1.1 Curse of Dimensionality?

It is important to note that none of the arguments so far have directly involved the dimensionality of the objective function N. Rather, the number of samples needed is inversely proportional to the minimum width of the basins of attraction p_m.

The minimum width of the basins of attraction may or may not related to the dimensionality of the problem. For many problems, the number of local minima increases substantially when the dimension increases. In such cases, p_m decreases as the dimension increases. For some other problems, however, the number of local minima does not change with the dimensionality. Then, p_m is independent of N. Or better yet, the number of significant basins of attraction decreases with increasing dimensionality, as it does in the Griewank function equation (2.11) studied in Chapter 2. Therefore, it is not necessarily true that the number of samples needed to explore the model space increases exponentially with the number of dimensions, as it may often be thought. Although the dimensionality could be important for some problems, it is not the direct reason that an optimization problem is hard. Törn and Žilinska (1989) give a similar analysis on this issue from a different perspective.

This result is important for studying the complexity of optimization problems, since it makes it possible to study global features of some high-dimensional objective functions with a finite-sampling of the model space. From this perspective, rather than speaking of *the curse of dimensionality*, it may be more appropriate to speak of *the curse of small basins of attraction.*

3.2 Statistical Analysis on Complexity Estimation

Next I derive confidence intervals for the estimation of the basins of attraction and the complexity.

3.3 Clustering in Model Space

In reality, however, there are no friction-free infinitesimal balls which can roll down to the exact bottoms of the function surface. Similarly, we do not have ideal RHCs and so the local-descent searches may converge to models that are not exactly local minima. Therefore, a process of *clustering* is needed to classify the converged models to the basins of attraction in the implementation of Algorithm 3(b). The strategy of such a process is related to the objective function itself, as well as the numerical accuracy in the RHC. In this section, I discuss the clustering strategy and the errors in the complexity estimation due to the clustering.

3.3.1 Criteria for Clustering

For the purpose of the complexity analysis developed in the previous chapter, the critical quantities needed are (1) the number of samples converging to each basin of attraction, and (2) the function values at the local minima. Hence, the estimated complexity $\hat{C}_e$ in Algorithm 3 is not only influenced by the limited sampling K, but also by the numerical accuracy of the RHC used in the process.

Assume that there are no large jumps in function values for models close to each other. For continuous optimization problems, this assumption is equivalent to assuming a continuous objective function. Then, $\hat{C}_e$ would not be heavily influenced by the error of $f(\check{\mathbf{m}})$ as long the model $\check{\mathbf{m}}$ is not far from a local minimum. For the following discussion, in addition to the assumption of no statistical error in the complexity analysis, I ignore the influence of the inaccurate function values $f(\check{\mathbf{m}})$ at local minima. For the complexity analysis of Algorithm 3, it is important that the models belonging to the same basin of attraction are identified correctly.

Clustering analysis techniques are used in many fields for dividing a finite set of objects or points into subsets so that the objects in a subset are more similar to one another according to some measure than to those objects outside the subset (Zupan, 1982). When these objects are models in optimization problems, the clustering ideas can be used for benefiting global searches. Many such algorithms have been developed in attempts to solve global optimization problems; Törn and Žilinska (1989) devoted a complete chapter for summarizing such algorithms. Although differing in detail, the same basic idea applies to almost all of those algorithms in that repeated local-descent searches are used to explore the model space globally, then these models are grouped together around the local minima according to some dissimilarity measure.

In this thesis, I use a straight forward strategy for the step of clustering, wherein the model space is divided into a certain number of hyper cubes, each of which is referred to as a *bin*. If each axis of the model space is divided into b sections, the N-dimensional model space becomes an N-dimensional mesh with b^N hyper-cubic bins. Then, the final models from RHC-searches can be clustered to different basins of attraction according to the bins into which they fall. The strategy for choosing the bin-size depends on our knowledge of the objective function, the number of samples,

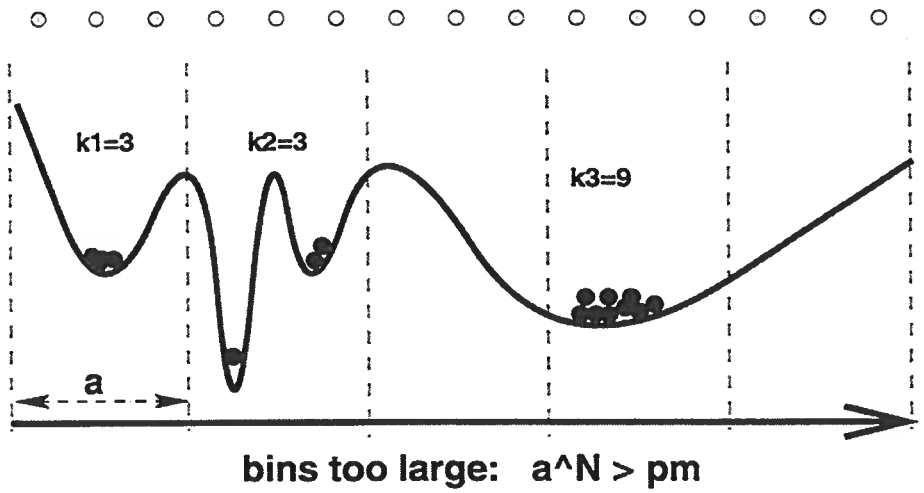

FIG. 3.2. Illustration of the discretization in a one-dimensional model space, when the local searches are relatively accurate and the number of samples is sufficiently large. Choosing a bin-size of larger than the minimum width of basins of attraction introduces error in estimating basins of attraction.

and our confidence in the accuracy of the RHC performed in Algorithms 3(a).

The combined actions of local-descent search operator ($\mathbf{T}_{local}$), finite number of iterations ($cmax$), and non-zero error tolerance (ϵ) in the RHC prevent the models from converging to the exact local minima. Resulting errors may well cause errors in the complexity estimation. Since local-descent searches are well understood and the results are easy to control and observe, it is reasonable to assume an estimate of the magnitude of the numerical error (α), which represents the maximum distance of the final models from their nearest local minima. For instance, α could be defined as follows,

$$\alpha = \max_{j \in [1,K]} \left(\min_{i \in [1,n]} \frac{\rho(\mathbf{m}_j^*, \check{\mathbf{m}}_i)}{D} \right), \tag{3.26}$$

where ρ is the metric for measuring the distance between two models in the model space, and D is the maximum possible distance between any two models. Also, $\{\mathbf{m}_j^*\}_{j=1,\ldots,K}$ are the final models from the RHC and $\{\check{\mathbf{m}}_i\}_{i=1,\ldots,n}$ are the exact locations of the local minima. The other possible influential factors include the minimum width of the basins of attraction (p_m) and the number of samples (K).

The Minimum Basins of Attraction When the inequality

$$p_m > \max(\alpha^N, \frac{1}{K}) \tag{3.27}$$

is satisfied, this corresponds to the situation when the RHC is relatively accurate and the number of samples sufficiently large. In this case, the dominating factor for the

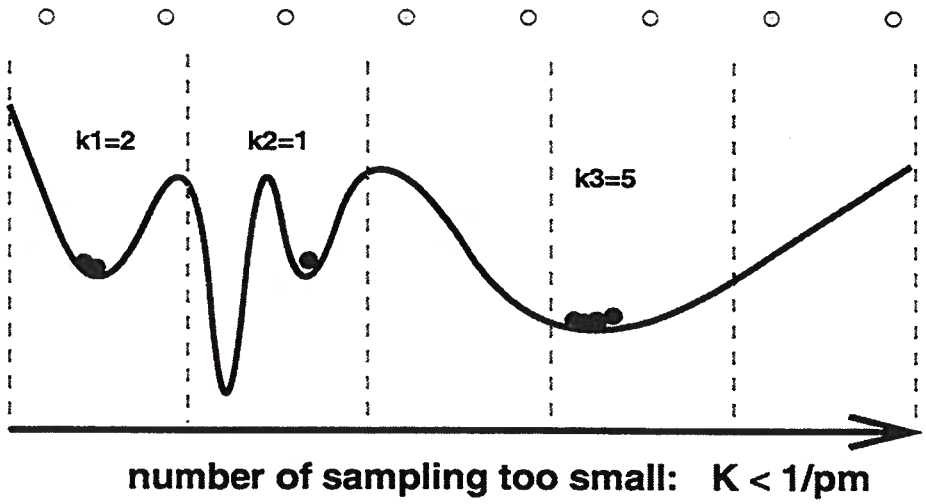

FIG. 3.3. Illustration of the discretization in a one-dimensional model space, when the local-descent searches are relatively accurate and the number of samples is not sufficiently large. It is not necessary to choose bin-size larger than $\hat{p}_m = 1/K$.

bin-size should be the minimum width of all the basins of attraction, p_m. Figure 3.2 shows such a situation when equation (3.27) is satisfied. If the bin-size is chosen larger than the minimum width of basins of attraction, the small basins of attraction would not be identified, as illustrated in Figure 3.2. On the other hand, choosing a bin-size of too small (*i.e.*, $a < \alpha$), error will also be introduced by splitting the models belonging to the same basin of attraction to different bins. Therefore, a good choice for the clustering would be to choose the number of bins along each axes in the model space b as

$$b \approx \frac{1}{\sqrt[N]{p_m}}. \tag{3.28}$$

In this case, each model is correctly clustered to the bin corresponding to the basin of attraction it belongs to. The imperfect convergence of the RHC does not introduce error in the complexity estimation. An accurate complexity analysis can be achieved even if the models stopped on the "slopes" of basins of attraction at the end of the RHC.

Number of Samples Figure 3.3 shows the situation in which the number of samples K is not sufficient for estimating the smallest basins of attraction. Since the smallest convergence is $k_i = 1$, then the smallest basin of attraction that can be estimated is $\hat{p}_m = \frac{1}{K}$. Therefore, a necessary requirement for the number of samples is

$$K \geq \frac{1}{p_m}, \tag{3.29}$$

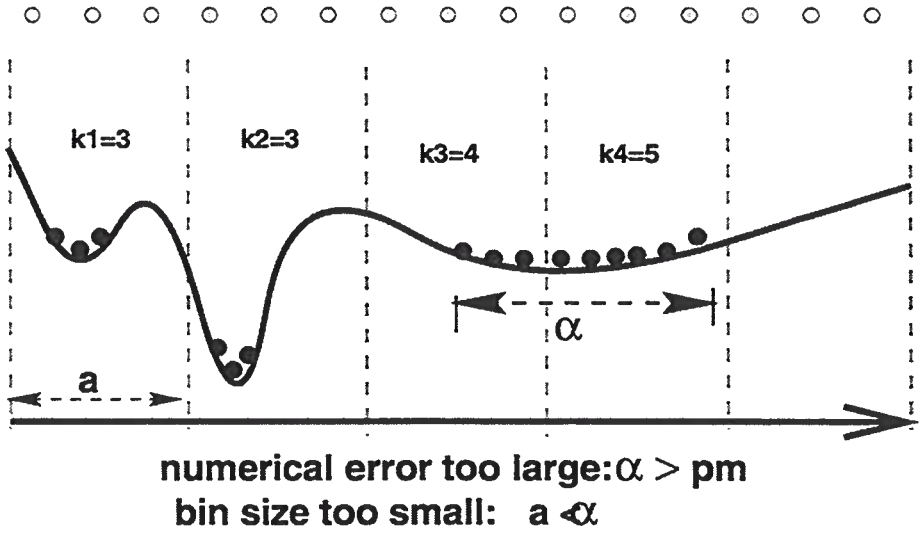

FIG. 3.4. Illustration of the discretization in a one-dimensional model space, when the errors caused by the RHC-searches are large compared with p_m and $\hat{p}_m$. The discretization shown here has bin-size that is too small, which causes the splitting of some basins of attraction.

where p_m is the smallest width of the basins of attraction. If equation (3.29) is violated, then any basin of attraction of width $p_i < \hat{p}_m = 1/K$ cannot be correctly identified. This situation is similar to that of aliasing in sampled signals, I will refer to it as *aliasing* hereafter. Figure 3.3 shows such aliased sampling when equation (3.29) is violated.

Suppose sampling of the model space is aliased and the local-descent searches are still relatively accurate, the following relation is satisfied,

$$K < \min(\frac{1}{\alpha^N}, \frac{1}{p_m}). \tag{3.30}$$

In this case, it is not necessary to choose the bin-size smaller than what can be identified; *i.e.*, $\hat{p}_m = \frac{1}{K}$. Figure 3.3 illustrates such a situation when equation (3.30) is satisfied. With K fixed, the erroneous identification of the smallest basin of attraction in Figure 3.3 cannot be compensated by choosing smaller bins, or more accurate RHC-searches. The number of bins along each axis of the model space can be estimated by

$$b \approx \sqrt[N]{K}. \tag{3.31}$$

Numerical Error Caused by the RHC As previously discussed, the purpose of the RHC in Algorithm 3(a) is to cluster the sampled models to local minima. Therefore, it is not mandatory that the RHC converge all models to local minima exactly. However, when the RHC-searches have a slow convergence rate, whether

caused by the poor choice of algorithm or the small curvature of the function surface, it is likely that the result of the complexity analysis will be influenced by the numerical error. Figure 3.4 illustrates such a situation when the following equation is satisfied,

$$\alpha > \max(\sqrt[N]{p_m}, \sqrt[N]{\frac{1}{K}}). \tag{3.32}$$

The discretization of the one-dimensional model space shown in Figure 3.4 causes the splitting of some basins of attraction by choosing the bin-size to be too small. To avoid this splitting, the number of bins along each direction of the model space can be chosen as

$$b \approx \frac{1}{\alpha}. \tag{3.33}$$

Combining the above discussion on three major factors on the discretization of the model space, the following algorithm is designed for the implementation of Algorithm 3.

Algorithm 4 (Identifying the Basins of Attraction) *Let $F : \mathcal{M} \subset \mathbf{R}^N \to \mathcal{Y} \subset \mathbf{R}$ be bounded and have isolated and finite number of local minima. Step 1 of Algorithm 3 has been performed, $\vec{\mathbf{m}}^* = \mathbf{RHC}(F, K, \epsilon, cmax)$. We assume that p_m is the smallest width of the basins of attraction on F in $\mathcal{M}$, and the estimated numerical error of the RHC is α, as defined by equation (3.26).*

1. *Discretize the model space to an N-dimensional mesh where each axis is evenly divided into b bins, where b can be roughly determined by*

$$b \approx \min(\frac{1}{\alpha}, \sqrt[N]{K}, \frac{1}{\sqrt[N]{p_m}}). \tag{3.34}$$

2. *Cluster the models $\vec{\mathbf{m}}^*$ in accommodance with the bins on the mesh to which each of them belongs.*

The output of this algorithm consists of: $\hat{n}$, the number of bins which contain at least one model, and k_i, the number of models that are clustered $\forall i \in [1, \hat{n}]$.

Alternative clustering techniques may also be used in the numerical implementation of Algorithm 3(b). One such technique starts from a seed point (possibly, the final model $\mathbf{m}$ with the smallest function value), then dynamically classifies the rest of the models according to the density of models in the space (Törn, 1977). If the new model does not belong to any existing cluster, a new cluster is created for the future use. That approach, however, is relatively complicated to implement, and the results may be difficult to evaluate.

Next I show the error estimates when Algorithm 4 is used for the complexity analysis described by Algorithm 3.

Theorem 4 (Errors Caused by the Clustering) *For a realization of Algorithm 4, suppose the initial number of samples is K, and $\{p_i\}_{i=1,\ldots,n}$ are the exact widths of the n basins of attraction in the model space. Suppose these K initial samples are distributed uniformly in the model space, which means the error estimated in Theorem 2 is zero. Also assume that the function values of the calculated local minima are exact, i.e., $F(\hat{\mathbf{m}}_i) = F(\check{\mathbf{m}}_i)$. Then, if the discretization of the model space is as in equation (3.34), the error of $\hat{C}_e$ caused by this clustering can be estimated by the following.*

1. *If the number of samples is sufficiently large to avoid aliasing, and the error of the RHC is smaller than the smallest size of the basins of attraction, i.e.,*

$$p_m > \max(\alpha^N, \frac{1}{K}), \tag{3.35}$$

then estimates of the basins of attraction are exact, i.e.,

$$\hat{p}_i = p_i \quad \forall i \in [1, n], \tag{3.36}$$

and the complexity estimation is also exact due to the assumption of exact function values,

$$C_e = \hat{C}_e.$$

2. *When the number of samples K is not sufficiently large to find the smallest basins of attraction, yet the error caused of the RHC is small compared with the size of the smallest basins of attraction, i.e.,*

$$K < \min(\frac{1}{\alpha^N}, \frac{1}{p_m}), \tag{3.37}$$

then, the complexity would be under-estimated, i.e., $\hat{C}_e < C_e$, and the difference would be bounded by

$$|C_e - \hat{C}_e| \leq \frac{n - \hat{n}}{K} \ln K. \tag{3.38}$$

3. *When the error of the RHC is inaccurate to the extent that*

$$\alpha > \max(\sqrt[N]{p_m}, \frac{1}{\sqrt[N]{K}}), \tag{3.39}$$

results of the complexity estimation would be heavily influenced by this error. If the discretization of the model space is chosen as $b = \frac{1}{\alpha}$, the complexity would be under-estimated; that is

$$\hat{C}_e < C_e.$$

The error of the estimated complexity $\hat{C}_e$ from the true complexity C_e would be

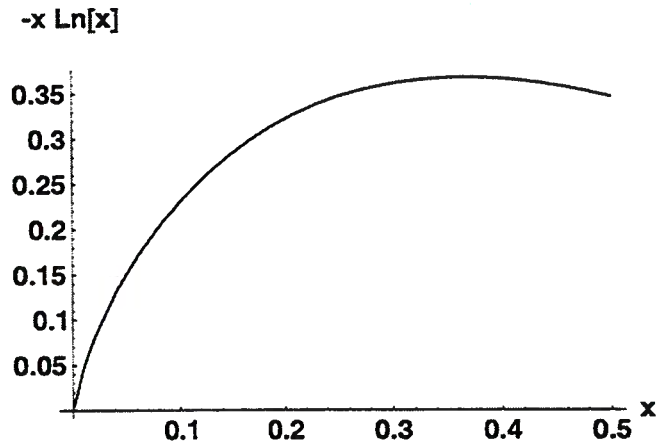

FIG. 3.5. The function $f(x) = -x \ln x$ for $(0, 0.5)$ has a maximum at $x \approx 3.68$.

bounded by

$$|C_e - \hat{C}_e| \leq (\hat{n} - n)\, \alpha^N \ln \alpha^N. \tag{3.40}$$

Proof:

Without loss of generality, suppose the basins of attraction are labeled in descending order, such that $p_1 \geq p_2 \geq \ldots \geq p_n$. Before proving the above results, let us consider a function that is frequently used in the following proof,

$$f(x) = -x \ln x. \tag{3.41}$$

Figure 3.5 shows a plot of this one-dimensional function $f(x)$ for $0 < x \leq 0.5$. It is easy to verify that $f(x) > 0$ and it increases monotonically when $0 < x < \frac{1}{e} \approx 0.368$.

1. For the case of equation (3.35), the minimum width of the basins of attraction p_m is the dominating factor. Equation (3.34) for the discretization of the model space becomes $b \approx \frac{1}{\sqrt[N]{p_m}}$. Furthermore, since all K samples are uniformly distributed within the model space and $K > \frac{1}{p_m}$, then at least one model converges to each of the basins of attraction (see page 30). The clustered distribution after the discretization would be

$$\hat{k}_i = \lfloor K p_i \rfloor \geq 1 \quad \forall i \in [1, n]. \tag{3.42}$$

Then, the estimation of the widths of basins of attraction would be exact.

If the assumption of exact values of local minima is used, the estimated complexity $\hat{C}_e$ would be exact as well. Therefore, it is possible to get an accurate estimation of the complexity even if the RHC is not ideal.

2. For the case of equation (3.37), aliasing is the dominating factor. Equation (3.34) for the discretization of the model space becomes $b \approx \sqrt[n]{K}$. Furthermore, since all K samples are uniformly distributed within the model space, and $K < \frac{1}{p_m}$, there will be no models converging to the basins of attraction whose width $p_i < \frac{1}{K}$. The clustered distribution after the discretization would be

$$\hat{k}_i = \begin{cases} \lfloor K\, p_i \rfloor, & \text{if } p_i \geq \frac{1}{K}: \\ 0, & \text{otherwise.} \end{cases} \tag{3.43}$$

So, we have $n > \hat{n}$. In addition, since the function values at the local minima are exact, we can have

$$\hat{q}_i = \frac{1}{c}\, q_i; \quad \forall i \in [1, \hat{n}], \tag{3.44}$$

where $c = \sum_{j=1}^{\hat{n}} q_j < 1$. Then

$$\begin{aligned} C_e - \hat{C}_e &= -\sum_{i=1}^{n} q_i \ln q_i + \sum_{i=1}^{\hat{n}} \hat{q}_i \ln \hat{q}_i \\ &= \sum_{i=1}^{\hat{n}} (-q_i \ln q_i + \frac{q_i}{c} \ln(\frac{q_i}{c})) - \sum_{i=1+\hat{n}}^{n} q_i \ln q_i \\ &= I_1 + I_2. \end{aligned}$$

For the situation considered here $K < \frac{1}{p_m}$, it is almost always true that $n \gg 2$ and $p_i \ll 0.5$ for all $i \in [1, n]$. Since $q_i \leq p_i$ and $c < 1$, we see that $f(q_i) < f(q_i/c)$. Then,

$$I_1 = \sum_{i=1}^{\hat{n}} [f(q_i) - f(q_i/c)] \; < \; 0.$$

Other the other hand, if the model space is not severely under-sampled, which should be the case, we have $c \approx 0$. So, $I_1 \approx 0$. Furthermore, since

$$I_2 = \sum_{i=1+\hat{n}}^{n} f(q_i) \; > \; 0,$$

so $C_e > \hat{C}_e$, and

$$\begin{aligned} |C_e - \hat{C}_e| &\leq I_2 \\ &= |\sum_{i=1+\hat{n}}^{n} q_i \ln q_i| \\ &\leq |\sum_{i=1+\hat{n}}^{n} p_i \ln p_i| \end{aligned}$$

$$\begin{aligned} &\leq |(n-\hat{n})\hat{p}_m \ln \hat{p}_m| \\ &= \frac{n-\hat{n}}{K} \ln K. \end{aligned}$$

3. For the case of equation (3.39), the numerical error caused by the RHC is the dominating factor. Equation (3.34) for the discretization of the model space becomes $b \approx \frac{1}{\alpha}$. Then, the smallest estimated basin of attraction is $\hat{p}_m = \alpha^N$. Since all K samples are uniformly distributed within the model space, the basins of attractions having width $p_i > \hat{p}$ can be correctly identified.

   Suppose there are m basins of attraction whose widths are larger than $\hat{p}_m$; then we have,

$$\begin{aligned} C_e - \hat{C}_e &= \sum_{i=1}^{n} f(q_i) - \sum_{i=1}^{\hat{n}} f(\hat{q}_i) \\ &= \sum_{i=m+1}^{n} f(q_i) - \sum_{i=m+1}^{\hat{n}} f(\hat{q}_i) \\ &= I_1 - I_2, \end{aligned}$$

   where $I_1 > 0$ and $I_2 > 0$. Since $n > \hat{n}$, then $I_1 > I_2$. Therefore, $C_e > \hat{C}_e$, and again the complexity is under-estimated.

   Since $p_i < \hat{p}_m$ for all $i > m$, we have

$$\begin{aligned} I_1 &= \sum_{i=m+1}^{n} f(q_i) \\ &< (n-m)\, f(p_m). \end{aligned}$$

   In addition, since $\hat{p}_i \geq \hat{p}_m \ \forall i \in [1, \hat{n}]$, then for $\forall i \in [m+1, \hat{n}]$, it is the case that $\hat{p}_i = \hat{p}_m$. So, we have

$$\begin{aligned} I_2 &= \sum_{i=m+1}^{\hat{n}} f(\hat{q}_i) \\ &\approx (\hat{n}-m)\, f(p_m), \end{aligned}$$

   and

$$\begin{aligned} |C_e - \hat{C}_e| &= |I_1 - I_2| \\ &\leq |(n-m)f(\hat{p}_m) - (\hat{n}-m)f(\hat{p}_m)| \\ &= (n-\hat{n})\, \alpha^N \ln \alpha^N. \end{aligned}$$

□

Theorem 4 gives a reasonable analysis on error in practical implementations.

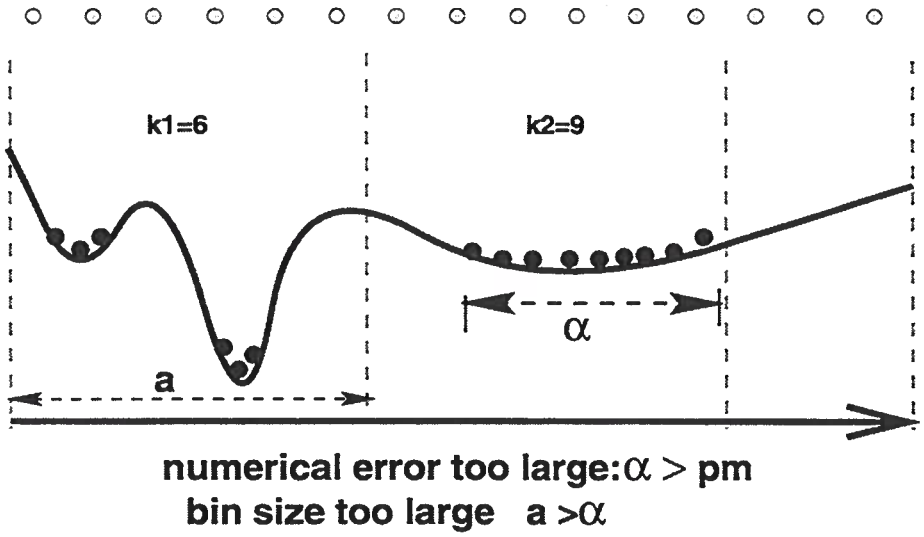

FIG. 3.6. Illustration of the discretization in a one-dimensional model space, when the error caused by the RHC-searches are large compared with p_m and $\hat{p}_m$. The discretization of the model space shown here has the bin-size too large. Some basins of attractions could not be identified.

However, some *ad hoc* assumptions have to be made, such as the knowledge of p_m and α. We need to be careful at interpreting the result of such an estimation: a single value of $\hat{C}_e$ calculated from one realization of the implementation may be meaningless if a careful comparison is not made with other similar situations. It can be a useful tool, on the other hand, for comparing similar objective functions with the same local search parameters, $(\mathbf{T}_{local}, cmax, \epsilon)$. In Chapter 2, I have shown examples of two types of comparison: a family of two-dimensional multi-modal function whose complexity changes tremendously with the change of some parameters, and the Griewank function whose complexity changes in an unusual way with increasing dimensionality.

3.3.2 Another Aspect of Complexity – Numerical Difficulty

Generally speaking, the situation of equation (3.39) should be avoided because of the possible severe distortion to the global picture of the objective function. If large numerical error α is caused by the algorithm, a faster convergence algorithm or more iterations should be used in the RHC. However, when the curvature of the objective function is nearly zero, which is equivalent to an ill-conditioned Hessian matrix, local-descent searches by *any* algorithm may give very poor convergence of the models within a finite amount of computing time. If we follow the implementation of Algorithm 4, the discretization of the model space would be too coarse to give us any useful information. Figure 3.6 shows the same situation as that in Figure 3.4, but with a coarser discretization, in which case the smaller basins of attraction could not be identified.

For representing such a numerical difficulty in optimizing objective functions with very small curvature, we need to choose smaller bin sizes. Instead of following the guideline of equation (3.34), we can discretize the model space sufficiently fine that $b < \frac{1}{\alpha}$. Figure 3.4 shows such a discretization of model space. The result of complexity analysis would take into account the spreading of the final models from the RHC. Thus, the estimated value of complexity $\hat{C}_e$ is higher than the true complexity C_e, which is an indication of the ill-posedness of the optimization problem. Taking such numerical issues into account in the complexity analysis therefore can represent an important aspect in the numerical difficulties of optimization. It can be shown that the following results are true.

Theorem 5 *For the same problem as in Theorem 4, if the numerical error dominates the final result of RHC, i.e.,*

$$\alpha > \max(\sqrt[N]{p_m}, \frac{1}{\sqrt[N]{K}}), \tag{3.45}$$

then the numerical difficulties of global searches are taken into account in the complexity analysis of Algorithm 3.

If the discretization parameter b is chosen that

$$b > \frac{1}{\alpha},$$

then complexity of the function would be over-estimated. The estimated complexity value is bounded as

$$C_e < \hat{C}_e \leq \min(\ln K, N \ln b). \tag{3.46}$$

Next, I show some examples of complexity analysis that are heavily influenced by their numerical errors, yet correctly indicates the numerical difficulties caused by the flatness of the landscape of the objective function.

Two-Dimensional Quadratic Functions For the two-dimensional function given by equation (2.10), if the oscillation frequency is zero ($f = 0$), the function $F(m_0, m_1)$ becomes quadratic with a unique global minimum. Figures 3.7(a) and (b) show the function surfaces when $f = 0$ and $a = 1$, 0.001 respectively. Both functions are quadratic. Noticing the scale of the two plots, however, Figure 3.7(b) is much flatter than (a). This difference in the curvature of surfaces significantly influences the difficulty of optimization. Figures 3.7(c) and (d) show the converged models for the RHC-search when the stopping criterion are met. In Figure 3.7(c), we see that all 500 models converge to the global minimum when the curvature of the quadratic function is large enough. As a result, $\hat{C}_e = 0$. However, when the function surface is almost constant, as in Figure 3.7(b), the convergence is so slow and insignificant

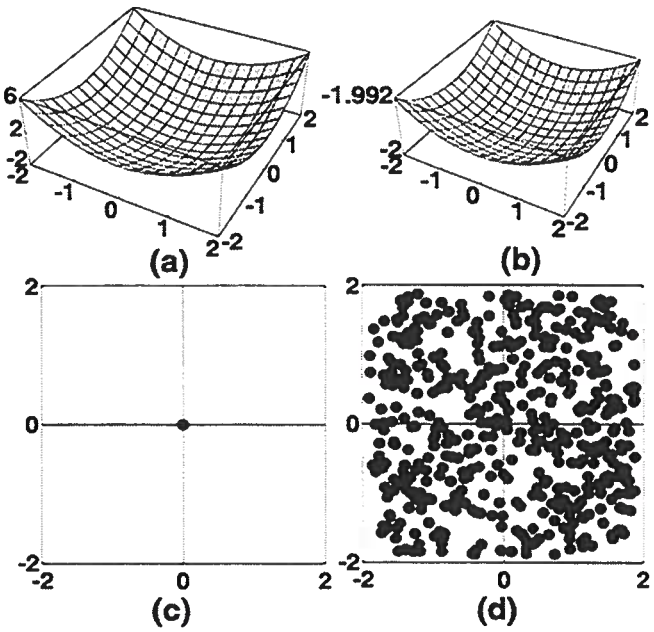

FIG. 3.7. Random hill-climbing with population of $K = 500$. (a) and (b) show the function surface defined in equation (2.10) when $f = 0$ and $a = 1$, 0.001 respectively. (c) and (d) show the convergence of 500 initial models in the 2-D model space on functions (a) and (b). In the first case $\hat{C}_e = 0$; in the second $\hat{C}_e = 4.11$.

that the final models are scattered about the domain when the maximum number of iterations is exceeded. For such a numerical experiment, the estimated complexity $\hat{C}_e = 4.1$. This demonstrates that even for unimodal objective functions, the hardness of searches could depend on the curvature of the landscape.

N-Dimensional Rosenbrock Function An N-dimensional Rosenbrock function (Fletcher, 1987) can be written as

$$R(\mathbf{x}) = \sum_{i=1}^{N-1} \left[100(x_i - x_{i-1}^2)^2 + (1 - x_{i-1})^2\right], \tag{3.47}$$

where $\mathbf{x} = (x_1, \ldots, x_N)$. It can be proved that this function is unimodal for any dimension N, with unique minimum is at $\{1, 1, \ldots, 1\}$. From Definition 5, the complexity measure for this function C_e would have been zero. However, the "long" and "narrow" basin around the minimum point still poses a challenge for any searching algorithms. This dimensionality-dependent ill-posedness can be reflected in the estimated complexity.

Figure 3.8 shows the function surface and its contour when $N = 2$. When $N \geq 2$, the function is still unimodal, but it is not easy to see how the increase of dimensionality alters the difficulty of optimization. Using the complexity-estimation approach in

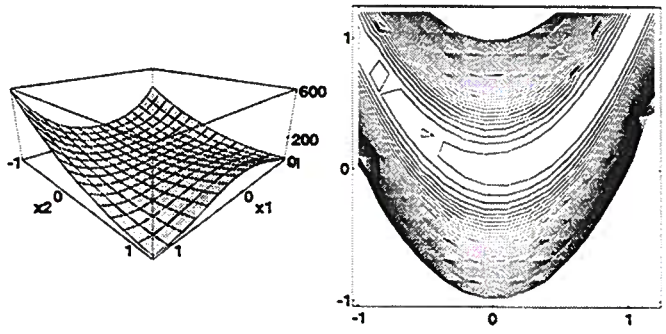

FIG. 3.8. Two-dimensional Rosenbrock function. The figure on the left is a 3-D plot of the function surface, while the one on the right shows the contour plot of the same function.

Algorithm 3, the value of $\hat{C}_e$ obtained from a finite-precision, finite-iteration RHC is largely influenced by the spatial curvature of the function surface; the estimated complexity would be larger than the exact complexity ($C_e = 0$) to an extent that varies with spatial curvature. Therefore, we can use the estimated complexity to estimate the "flatness" of a function surface.

One way of studying the spatial curvature of functions is by looking at the ratio of largest and smallest eigenvalues (*condition number*) of the Hessian at a point. The Hessian for equation (3.47) is a tri-diagonal matrix,

$$\begin{pmatrix} a_0 & c_0 & 0 & \cdots & 0 \\ b_1 & a_1 & c_1 & 0 & \cdots \\ 0 & b_2 & a_2 & c_2 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & b_{N-1} & a_{N-1} \end{pmatrix} \tag{3.48}$$

where

$$a_i = \begin{cases} 2 + 1200\,x_0^2 - 400\,x_1, & \text{if } i = 0; \\ 202 + 1200 x_{i-1}^2 - 400 x_i, & \text{if } 0 < i < N-1; \\ 200, & \text{if } i = N-1, \end{cases}$$

$$\begin{aligned} b_i &= -400\,x_{i-1}, \quad 0 < i \leq N-1, \\ c_i &= -400\,x_{i-1}, \quad 0 \leq i < N-1. \end{aligned}$$

At the global minimum $(1, 1, \cdots, 1)$, the tri-diagonal matrix equation (3.48) becomes Toeplitz except for a_0 and a_{N-1}. The condition number of the Hessian at the global

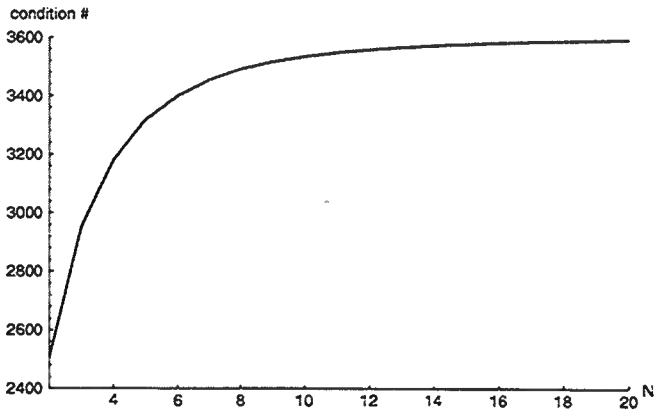

FIG. 3.9. Condition number of the Hessian matrix for the N-dimensional Rosenbrock function at the global minimum.

minimum reaches an asymptote with increasing dimension, as shown in Figure 3.9.

Figure 3.10 shows the estimated complexity $\hat{C}_e$ as a function of the number of dimensions; it shows the same asymptotic trend as does the condition number. Thus the increasing complexity for low dimensions is the result of increasing ill-conditioning of the Hessian and has nothing to do with local minima.

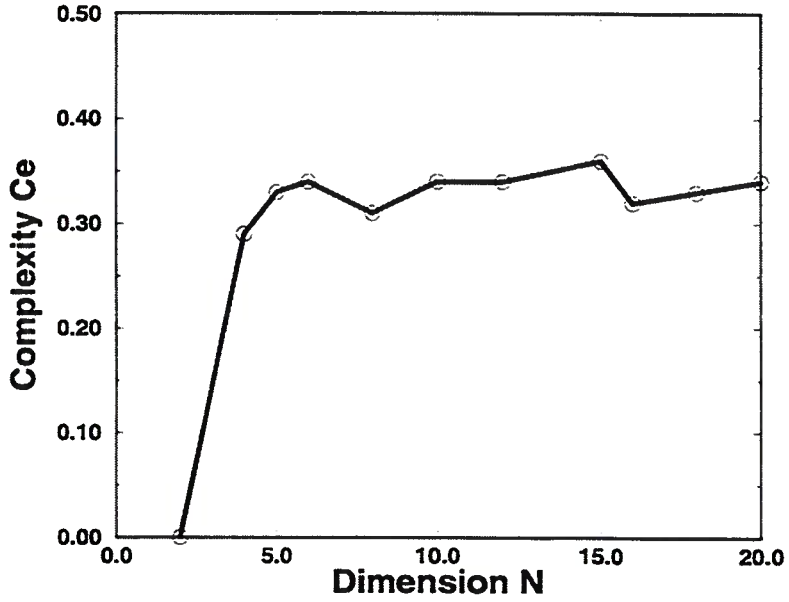

FIG. 3.10. Complexity $\hat{C}_c$ for N-dimensional Rosenbrock functions as a function of N.

Chapter 4

A HARD OPTIMIZATION PROBLEM IN GEOPHYSICS

In previous chapters, a method for studying the complexity of generic optimization problems was developed, and the implementation issues and error bounds were also discussed. To demonstrate applications of this approach, in this chapter I study a hard optimization problem encountered in explorational geophysics – *residual-statics* estimation.

The basic theme of the thesis is that we cannot usefully compare optimization techniques unless we understand the problem well. Therefore, the first step in dealing with an optimization problem is to study the physical nature of the problem, and take the best advantage of special characteristics of the problem. In this chapter, I first introduce the residual-statics estimation problem, and show that this problem is naturally formulated as an optimization problem. With insights gained from an analysis of the details of the problem, I propose two strategies for simplifying the objective function of this optimization problem: simplification of the data via multi-resolution analysis (MRA) and via use of envelope information. I show that the complexity analysis developed in the previous chapters can be used to evaluate these strategies.

4.1 Estimating Near-Surface Heterogeneities

For studying the earth's subsurface structure, explorational geophysicists usually send signals generated by *seismic sources* into the ground, and record the reflected echos from discontinuities of the earth material properties (*reflectors*) by *receivers*. These recorded data are usually referred to as *seismograms*, and the recordings from each receiver is a *seismic trace*.

The left portion of Figure 4.1 illustrates such a recording situation, and the right plot shows the assumed recording seismogram. In a seismogram, the high-amplitude wiggles, as shown on the right of Figure 4.1, usually represent reflections from strong discontinuities of the earth material properties such as velocity. The amplitudes and their positions on the time-axis (*traveltimes*) of reflections carry important information for recovering subsurface material properties and structure. As shown on the left of Figure 4.1, when the waves propagate vertically through the earth and are reflected from a horizontal reflector, the traveltime of the reflections should be the same for all traces in the same seismogram. However, the reflections in the seismogram of Figure 4.1 do not have the same traveltime. This is a common phenomenon for seismic data recorded on land. The heterogeneous material properties in the earth's near surface can cause time-shifts of reflections on seismic traces. These time-shifts

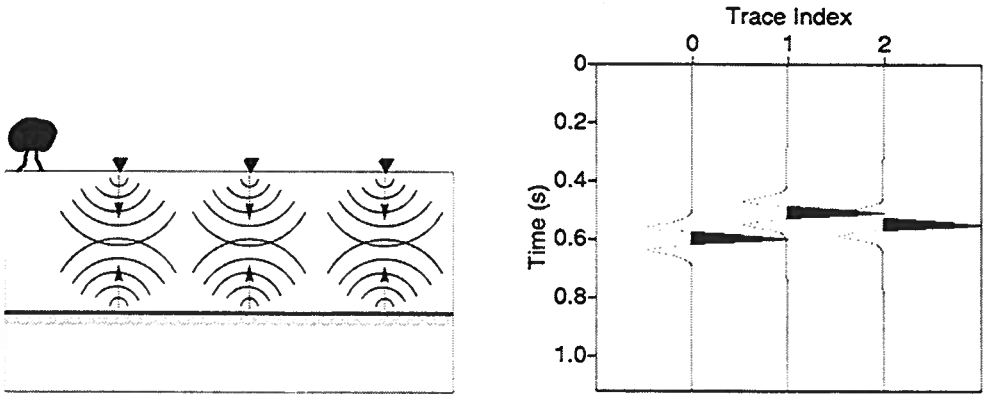

FIG. 4.1. The recording of reflection seismic data is illustrated on the left, and on the right are depicted three synthetic seismic traces time-shifted by (presumably) random statics.

are referred to as *statics*, since they cause time distortions that often can be usually approximated as time-shifts of the entire trace under the assumption that the energy is mostly propagating vertically through the near-surface.

If the seismic waves did not propagate vertically through the earth media or were not reflected from horizontal reflectors, the reflection events would show regular time distortions among nearby traces. These propagating factors are usually estimated and corrected (*normal moveout* correction) so that the reflections on seismogram appear as if travel paths were vertical. When these processing steps are done correctly, the misalignments among the nearby traces can be attributed to the near-surface variations and are often assumed to be static shifts.

After the propagating factors are corrected, the seismic data need to go to the *stacking* step, where the traces representing waves traveling through the same media are averaged in order to compress the data and enhance the data quality. To achieve good quality in stacked sections, the reflections on the traces representing the same changes in earth properties need to be aligned. This not only requires a good correction of propagating factors, but also requires a correction of the statics contamination in the data. The contamination of the alignment by statics are usually corrected by *residual-statics estimation*. The goal of residual-statics estimation is to look for the time-shift of each trace that maximizes the alignment of the traces to be stacked together. Since statics corrections are applied to virtually all seismic data recorded on land, the estimation of these corrections is key to the quality of the images seismologists extract from such data.

Generally speaking, the ultimate goal for statics corrections is to improve the images after stacking. Hence, a measure of stacking quality can be used as an objective function in such a problem. One measure of stacking quality is the sum of squares of the stacked traces – stacking power (Neidell & Taner, 1971; Ronen & Claerbout,

1985; Rothman, 1985). The stacking power of N seismic traces may be written as

$$\sum_t \left[\sum_{i=0}^{N-1} d_i(t + \tau_i) \right]^2 , \tag{4.1}$$

where $d_i(t)$ is a one-dimensional function of time t representing the ith trace and each component of vector $\vec{\tau}$ is the time-shift for each seismic trace. It can be easily shown that maximizing the function in equation (4.1) is equivalent to minimizing a negative summation of cross-correlation functions.

$$F(\vec{\tau}) = - \sum_{i=0}^{N-1} \sum_{j=i-1}^{N-1} \left[\Phi_{ij}(\tau_j - \tau_i) \right], \tag{4.2}$$

where $\Phi_{ij}(\tau)$ is the cross-correlation of the ith and jth traces evaluated at $\tau = \tau_j - \tau_i$. Equation (4.2), rather than (4.1), is mostly used in implementations because cross-correlations $\Phi_{ij}(\tau)$ may be pre-computed and stored.

Because of the oscillatory nature of seismic traces, cross-correlations $\Phi_{ij}(\tau)$ are also oscillatory. Positive and negative peaks of the correlation functions $\Phi_{ij}(\tau)$ are the indications of time-shifts when the traces, $d_i(t)$ and $d_j(t)$, are aligned *in-phase* (aligned with the same polarity) or *out-phase* (aligned with the opposite polarity). Hence, objective functions $F(\vec{\tau})$ in equation (4.2) are generally highly multimodal. Consider a simple example where we need to align three otherwise identical traces, similar to those shown in Figure 4.1. Fixing the first trace, we look for time-shifts of the second and third traces, t_1, t_2, so that the sum of squares of the stacked traces (stacking-power) is maximized. Figure 4.2 shows an example of such a two-dimensional objective function $-F(t_1, t_2)$, where the basins of attraction are shown as "hills" scattered on the landscape. When the objective function is high-dimensional (such as when the number of seismic traces being analyzed is large, and often $N > 10^4$, one could imagine that the landscape would be even more complex. Smith *et al.* (1992) showed that an arbitrary two-dimensional projection of one such function presents a highly irregular shape.

Furthermore for seismic traces in explorational geophysics, it is usually the case that each trace cannot be shifted independently. This corresponds to the case of equation (4.2) where each components of $\vec{\tau}$ cannot be arbitrarily chosen. In this case, a set of independent parameters needs to be chosen as unknowns for the problem. For example, the *source-statics* and *receiver-statics* are the independent unknowns under the *surface-consistent* assumption, as discussed in Appendix C. In general, the time-shift of each trace τ_i is a linear combination of the unknown parameters. Then, a general residual statics problem becomes a mathematical optimization problem in which we look for parameter vectors $\mathbf{m}$ so that

$$\min_{\mathbf{m}} F(\mathbf{m}) = - \sum_i \sum_{j \neq i} \Phi_{ij}(\tau_{ij}(\mathbf{m})), \tag{4.3}$$

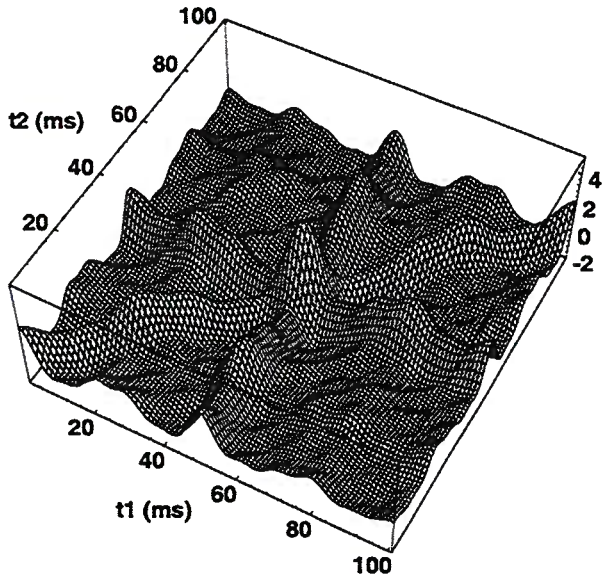

FIG. 4.2. Landscape of a 2-D residual-statics objective function.

where $\tau_{ij}(\mathbf{m})$ is the linear function of $\mathbf{m}$. The function $F(\mathbf{m})$ in equation (4.3) is a generic formulation of a *stacking-power function.*

4.1.1 Behavior of Stacking-Power Functions

As complex as it might be, the stacking-power function in equation (4.3) is far from arbitrary. Treating this function with some understanding may indeed help us in choosing strategies for dealing with the statics-estimation problem. To gain insights in these stacking-power functions, let us take a close look at the function of equation (4.3):

Observation 1 (Stacking-Power Function) *The stacking-power function $F(\mathbf{m})$ of equation (4.3) has the following generic properties:*

1. *F is a summation of a set of one-dimensional functions, Φ_{ij}. Each Φ_{ij} can be computed prior to the optimization procedure.*

2. *For models where all correlation functions Φ_{ij} are simultaneously maximized, the function F is guaranteed to have a minimum (see Lemma 3).*

3. *Not all minima of F correspond to the maxima of all Φ_{ij}. The local minima of F that do not correspond to local maxima of Φ_{ij} are not related to the alignment of traces.*

4. *There exists a linear transformation such that the function F can be reduced to a set of one-dimensional optimization problems.*

To verify the above observations, let us take a close look at the stacking-power function. For a generic stacking-power function $F(\mathbf{m})$ in equation (4.3), each component of the gradient vector can be written as

$$\frac{\partial F(\mathbf{m})}{\partial m_k} = -\sum_i \sum_{j \neq i} \Phi'_{ij}(\tau_{ij}(\mathbf{m})) \frac{\partial \tau_{ij}}{\partial m_k}, \quad \text{for } \forall k. \tag{4.4}$$

With equation (4.4), the following statements can be easily verified.

Lemma 3 *If there exists* $\mathbf{m}^*$ *such that*

$$\nabla \Phi_{ij}(\mathbf{m}) \mid_{\mathbf{m}=\mathbf{m}^*} = \mathbf{0}, \quad \forall i, j, \tag{4.5}$$

then,

$$\nabla F(\mathbf{m}) \mid_{\mathbf{m}=\mathbf{m}^*} = \mathbf{0}. \tag{4.6}$$

Lemma 3 states that if a point in the domain is the maximum of all correlation functions simultaneously, then it must be a maximum of the stacking-power function. This lemma is equivalent to the first statement of Observation 1, which can be proved as follows.

Proof: Since $\nabla \Phi_{ij}(\mathbf{m}) \mid_{\mathbf{m}=\mathbf{m}^*} = \mathbf{0}$, for $\forall i, j$, we can also write that for each Φ_{ij},

$$\frac{\partial \Phi_{ij}(\mathbf{m})}{\partial m_k} \mid_{\mathbf{m}=\mathbf{m}^*} = \Phi'_{ij}(\mathbf{m}) \frac{\partial \tau_{ij}}{\partial m_k} = 0, \quad \text{for } \forall k. \tag{4.7}$$

From equation (4.4), we have that $\nabla F(\mathbf{m}) \mid_{\mathbf{m}=\mathbf{m}^*} = \mathbf{0}$. Therefore, $\mathbf{m}^*$ is a minimum for the stacking-power function equation (4.3).

□

However, as stated in Observation 1, the converse of Lemma 3 is not necessarily true. It is easy to see that local minima of F could be caused by the compensation of positive- and negative- slopes between correlation functions Φ_{ij}. That is, if equation (4.6) is true, equation (4.5) is not necessarily true.

Significantly, a change of variables can be applied to equation (4.3) so that the function of the new set of variables, $\tilde{F}(\tilde{\mathbf{m}})$, becomes *separable*, where no non-linear interactions exist in the function (Whitley *et al.*, 1995b). Then, the minimization of the transformed function $\tilde{F}(\tilde{\mathbf{m}})$ can be solved as a set of one-dimensional optimization problems. Detailed discussion and application of this point are given in Appendix C.

To demonstrate these observations, we start with a simple, schematic statics problem: consider three identical seismic traces that can be shifted independently of each other, and look for the independent time-shifts (τ_i, $i = 0, 1, 2$) that best align the traces. Without loss of generality, we use the first trace as a reference ($\tau_0 \equiv 0$) and look for time-shifts of other two traces (τ_1 and τ_2). The stacking power is a

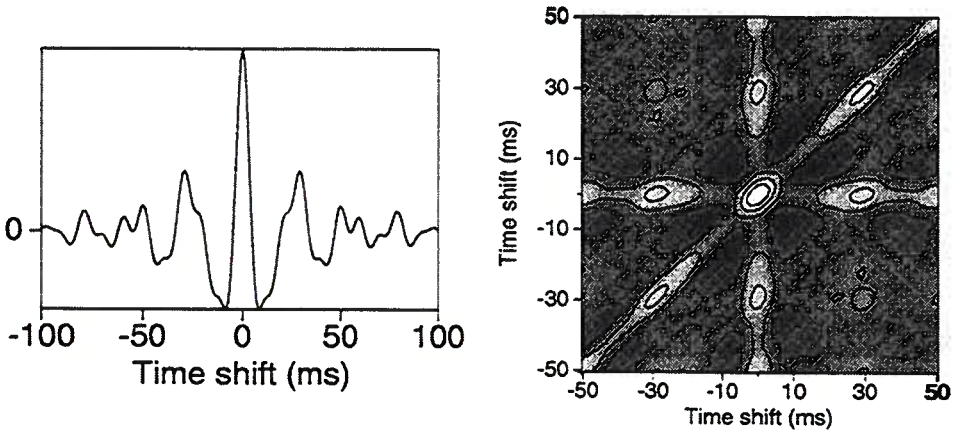

FIG. 4.3. The correlation function is shown on the left. The right figure shows the contour of the stacking-power function formed from the correlation on the left as a function of τ_1 and τ_2.

two-dimensional function,

$$F(\tau_1, \tau_2) = -[\Phi_{01}(\tau_1) + \Phi_{02}(\tau_2) + \Phi_{12}(\tau_2 - \tau_1)]. \tag{4.8}$$

If the three traces are identical, all correlation functions are auto-correlations. The left of Figure 4.3 shows one example of such correlations. The two-dimensional stacking-power function, shown on the right of Figure 4.3, demonstrates that the function is multimodal even for such a simple case. The converged models of local-search algorithms will strongly depend on the initial models, and the estimated statics may easily be at a local minimum if a local search is used. Notice also that the observable minima in Figure 4.3 are scattered regularly on the function surface. The global minimum is located at the point where maximum positive peaks of all three correlation functions coincide (the correct time-shift), and the observable local minima are regularly distributed at locations where positive peaks of all three correlation functions coincide. Among these local minima, those that coincide with the maximum positive peak of one correlation function have significant values. In Figure 4.3, the dominant stripes (vertical, horizontal, and diagonal) are located in areas where any of the three correlation functions is at its highest function value.

Observations from Figure 4.3 suggest that contributions to significant minima of stacking-power function are mostly due to positive peaks of the correlations from which the stacking-power function is made. Small and negative correlation values are of little interest to us.

4.2 Using a Multi-Resolution Analysis (MRA) to Simplify Stacking-Power Functions

As with the stacking-power functions, many objective functions in seismic inverse problems are highly multimodal. This situation is caused primarily by the oscillatory nature of seismic signal. Therefore, it can be expected that the number of local minima of such objective functions would be largely influenced by the band-width of seismic signals in the traces. That is, high-frequency signals in seismic traces produce more local extrema in the objective functions, and low-frequency signals result in smoother ones. Hence, one way of making such optimization problems easier would be to reduce the frequency content in the seismic data.

Multi-scale analysis is a strategy for simplifying such objective functions. A sequence of low-pass filters can be applied to the seismic data so the data are decomposed into several sets, each of which contains progressively higher frequencies (Saleck *et al.*, 1993; Chen, 1994). Then, optimization procedures are applied to these data sets iteratively in order to increase the chance of finding global minima. In this thesis, a similar approach is used for simplifying stacking-power functions in equation (4.3). Instead of using low-pass filters, I use a shift-invariant multi-resolution analysis (MRA) technique.

4.2.1 Multi-Resolution Analysis

The basic concept in the MRA is a set of nested, closed subspaces $\{V_j;\ j \in \mathbf{Z}\}$ of $\mathbf{L}^2(\mathbf{R})$, such that (Daubechies, 1992; Jawerth & Sweldens, 1994)

$$\ldots V_3 \subset V_2 \subset V_1 \subset V_0 \ldots. \tag{4.9}$$

The basis for the subspace V_j is a set of orthonormal, translated functions, and each of these function sets is a fixed dilation of a *scaling function*, $\{\phi_{j,k};\ k \in \mathbf{Z}\}$. These subspaces have the property (Daubechies, 1992; Jawerth & Sweldens, 1994)

$$f(x) \in V_0 \Longleftrightarrow f(2^{-j}x) \in V_j; \quad \forall j \in \mathbf{Z}. \tag{4.10}$$

Define W_j to be the *orthogonal complement* of V_j in V_{j-1}, then

$$V_{j-1} = V_j \ominus W_j. \tag{4.11}$$

A set of *wavelets* $\{\psi_{j,k}; k \in \mathbf{Z}\}$ forms an orthonormal basis of the subspace W_j. Therefore, for $j < n_0$, we can have

$$V_j = V_{n_0} \ominus W_{n_0} \ominus W_{n_0-1} \ldots \ominus W_{j+1}. \tag{4.12}$$

Figure 4.4 illustrates the nesting of subspaces V_j and their orthogonal complements W_j. In Figure 4.4, V_0 contains the original data which has the finest resolution; the projection of the data on $\{V_j;\ j = 1, 2, 3\}$ has increasingly coarse resolution. The data

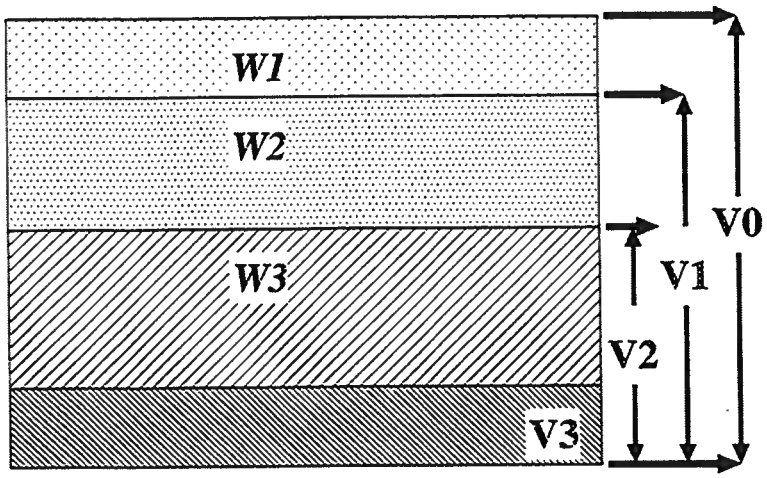

FIG. 4.4. Illustration of the sequence of multi-resolution analysis subspaces V_j. W_j is the orthogonal complement of V_j in V_{j-1}. Space V_0 represents the space that contains the finest resolution data, and $V_0 = V_3 \ominus W_3 \ominus W_2 \ominus W_1$.

projected onto the subspace V_j is referred as the decomposition of data at resolution level j.

We define the projection of a function $f \in V_0$ on V_j to be $f^j(x)$. Then the jth resolution level of the function has the form

$$f^j(x) = \sum_k s_{j,k}\, \phi_{j,k}(x). \tag{4.13}$$

where $s_{j,k}$ is the component of the function $f(x)$ with respect to the basis $\phi_{j,k}$; that is,

$$s_{j,k} = \int f(x)\, \phi_{j,k}(x)\, dx.$$

Next, define the projection of $f(x)$ on the subspace W_j to be

$$df^j(x) = \sum_k d_{j,k}\, \psi_{j,k}(x), \tag{4.14}$$

where $d_{j,k}$ is the compoment of function $f(x)$ with respect to the basis $\psi_{j,k}$

$$d_{j,k} = \int f(x)\, \psi_{j,k}(x)\, dx.$$

Then, equation (4.12) implies that the original function $f(x) \in V_0$ can be represented by

$$f(x) \;=\; f^{n_0}(x) + \sum_{j=n_0}^{i} df^j(x)$$

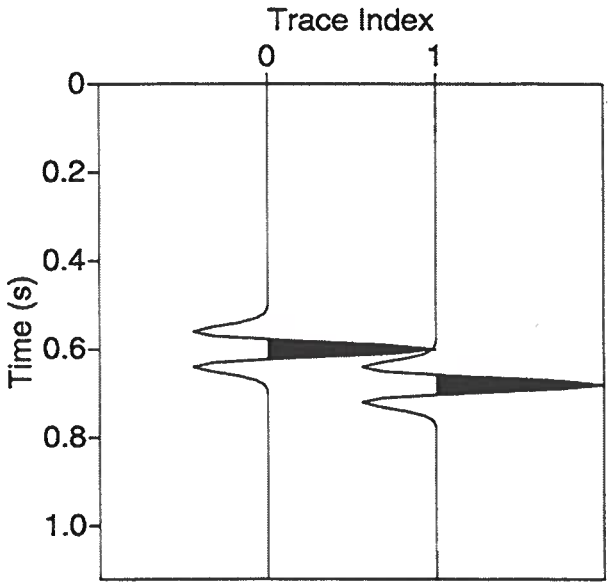

FIG. 4.5. The observed data in the first example. Two traces contain identical waveforms, a Ricker wavelet with a 30 Hz peak frequency. The relative time-shift is the unknown we are seeking.

$$= \sum_k s_{n_0,k}\,\phi_{n_0,k}(x) + \sum_{j=n_0}^{1}\sum_k d_{j,k}\,\psi_{j,k}(x). \tag{4.15}$$

Therefore using MRA, signals can be decomposed into various resolution levels. The data with coarse resolution contain less detailed information and lower frequencies, while keeping major features of the original signal. These less-informative data can serve as a relaxation to optimizations in the sense that, by using data at coarser resolution levels, complexity of objective functions may be reduced. One special kind of MRA, a shift-invariant basis developed by Saito and Beylkin (1993) is used in this work for decomposing seismic data to several sets with various level of resolution. A detailed description of MRA can be found in Appendix B.

4.2.2 Applying an MRA in Optimization

Let us first consider a trace-alignment problem, which involves a simple one-dimensional aspect of the residual-statics problem. Consider a trace containing one Ricker wavelet; and duplicate the trace with an unknown shift. Figure 4.5 shows two such traces. Now, we look for the time-shift between the two by applying an optimization, that is, searching for the time-shift that maximally aligns the two traces. This is a simplistic residual-statics estimation problem with one unknown. The corresponding

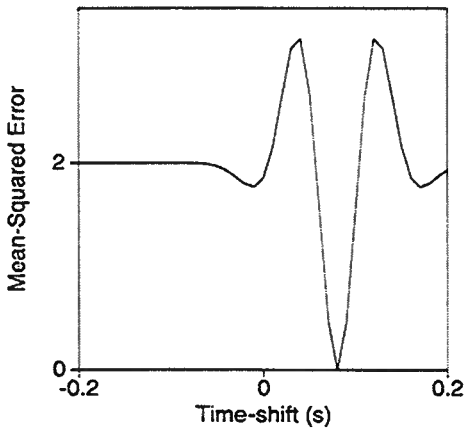

FIG. 4.6. The error-fitting function with respect to the relative time-shift between two traces. The goal is to find the optimal point where the mean-squared error is minimum.

stacking-power function is

$$F(\tau) = -\Phi(\tau),$$

where τ is unknown time-shift. Minimizing such a stacking-power function is equivalent to solving a least-squared problem with objective function

$$E(\tau) = \sum_{i=0}^{N-1} [P_0(i-\tau) - P_1(i)]^2, \tag{4.16}$$

where $P_0(t)$ and $P_1(t)$ are the two data traces, N is the number of samples per trace, and τ is the unknown time-shift. The goal is to find the time-shift τ that minimizes the error function $E(\tau)$. Such a minimization problem would be equivalent to minimizing a corresponding stacking-power function.

Figure 4.6 shows the error function of equation (4.16) for the fitting of these two traces. In addition to possible problems caused by the local minima, the basin of attraction leading to the global-minimum is steep and narrow, while the two areas to the sides are flat. The global structure of this objective function suggests that the global minimum point may be hard to find by directly applying local-descent methods. Assuming that we know *a priori* that the time-shift between the traces lies in the range of $[-0.2, 0.2]$ s, the searching range is restricted to this interval. Figure 4.7 shows the histogram of the converged models after an RHC with $K = 50$. As expected, the chance of finding the correct global minimum is small. For this test, 8 out of 50 experiments found the correct time-shift ($\hat{p} = 0.16$).

Let us decompose the observed data into various resolution levels by representing them with the wavelet bases described in Appendix B. For the above example, the traces $\{P_i(x);\ i = 0, 1\}$ of lengths $N = 2^J$ can be represented in the form of

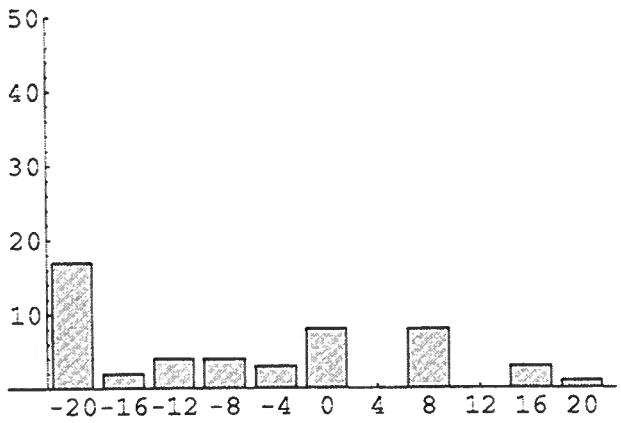

FIG. 4.7. Histogram of the obtained time-shifts of 50 conjugate-gradient optimization experiments starting from uniformly distributed random initial models between $[-0.2, 0.2]$ s. The horizontal axis is the number of shift-samples, where the sample interval is 0.01 s, and the grid size of the histogram is four samples. The number of times that found the true global minimum (shift of eight samples) is 8 out of 50.

equation (4.15),

$$P_i(x) = P_i^{n_0}(x) + \sum_{j=n_0}^{1} \sum_{k=0}^{N-1} d_{j,k}\, \psi_{j,k}(x), \tag{4.17}$$

where $1 \leq n_0 \leq J$ and $P_i^{n_0}(x)$ is the projection of the original data onto the subspace V_{n_0}. Therefore, equation (4.16) can be rewritten as,

$$E^{n_0}(\tau) = \sum_{i=0}^{N-1} [P_0^{n_0}(i-\tau) - P_1^{n_0}(i)]^2 + R^{n_0}(\tau), \tag{4.18}$$

where $R^{n_0}(\tau)$ is the residual error term which is related to the detailed information being projected onto subspaces $\{W_j;\ j = 1, ..., n_0\}$.

Ignoring certain levels of fine-resolution information, *i.e.*, ignoring the residual term in equation (4.18), the resolution level n_0 representation of the seismic traces can be used for optimization. Figure 4.8 shows the objective function $E^{n_0}(\tau)$ at various resolution levels, $n_0 = 1, 2, 3, 4$. It shows that the global complexity of the objective function is reduced with the increasingly coarse level of resolution, and that the basins of attraction leading to the global minimum become wider as well. Figure 4.8(d), however, shows that the global structure of the objective function is severely distorted when too much detail is ignored.

The distortion of Figure 4.8(d) is caused by the shift-variant nature of compactly supported, orthonormal wavelet bases. In order to keep the global shape of the ob-

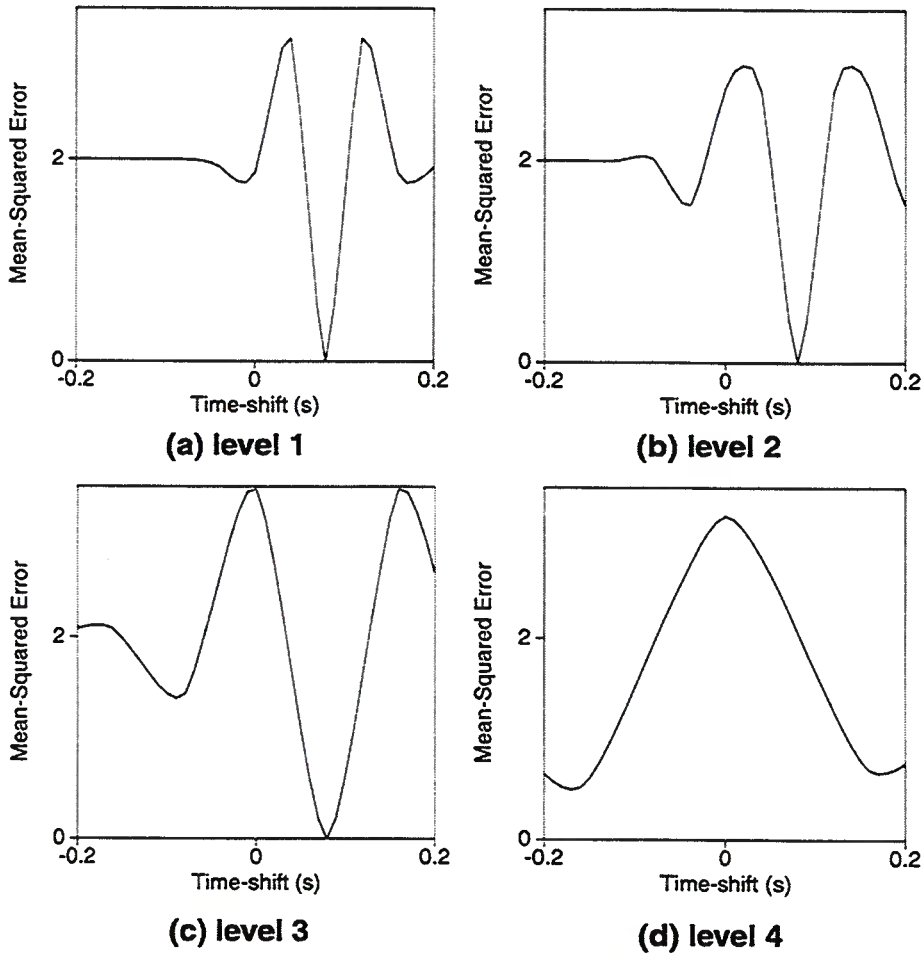

FIG. 4.8. The mean-squared error functions for two seismic traces at various resolution levels. The traces are decomposed in the Daubechies basis with two vanishing moments.

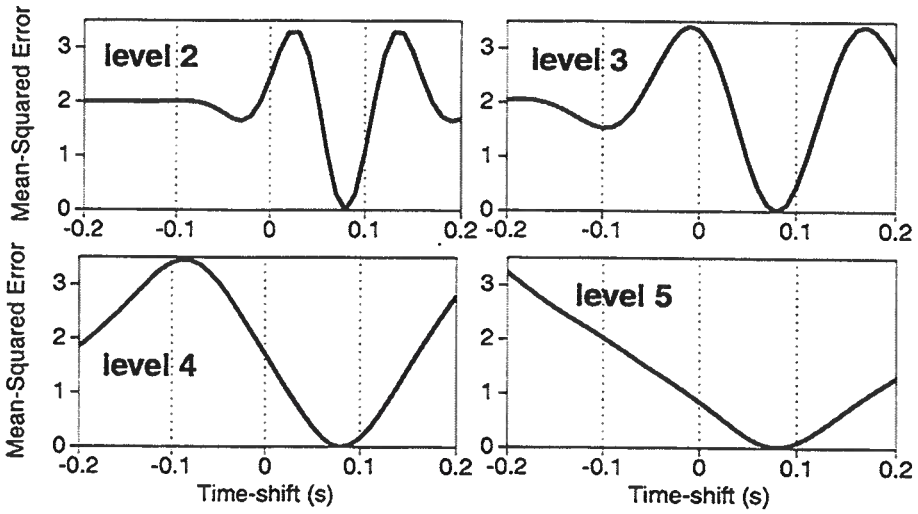

FIG. 4.9. The mean-squared error functions for two seismic traces at various resolution levels. The traces are decomposed in the auto-correlation shell of the Daubechies basis with two vanishing moments.

jective function in this problem, it is necessary that a shift-invariant basis is used to decompose the signal. From Appendix B, two families of bases are shift-invariant. Because the auto-correlation shell of orthonormal bases (see page 103) is symmetric, this bases is used for this study. From this point on, the paper uses only the auto-correlation shell of orthonormal bases to decompose the signal, unless otherwise indicated.

Figure 4.9 shows the objective function at resolution levels $n_0 = 2, 3, 4, 5$ in an auto-correlation shell of the Daubechies wavelet basis with two vanishing moments (Daubechies, 1992). The global structure of the objective function also shows the desired simplification seen in Figure 4.8, such as a wider basin of attraction leading to the global minimum, fewer oscillations and smaller "flat" area in the searching range. Moreover, the global minimum is not shifted at any decomposed resolution level. Figure 4.9(d) shows that the whole searching range has been transformed to one wide basin of attraction, which would lead all initial models to the global minimum.

Figure 4.10 shows histograms such as that shown in Figure 4.7, except the data used for optimizations are decomposed at various resolution levels. These results confirm our prediction that there are increasing chances for local-search optimizations to find the global minimum when coarse-resolution data are used. For Figure 4.10(d), all searches converge to the global minimum when data are decomposed to resolution level 5.

For the simple problem, five levels of decomposition are needed to reduce the objective function to a convex function in the searching range. In addition, the global minimum of this simplified objective function coincides with that of the original objective function. Therefore, the correct solution is reached when only coarse resolution

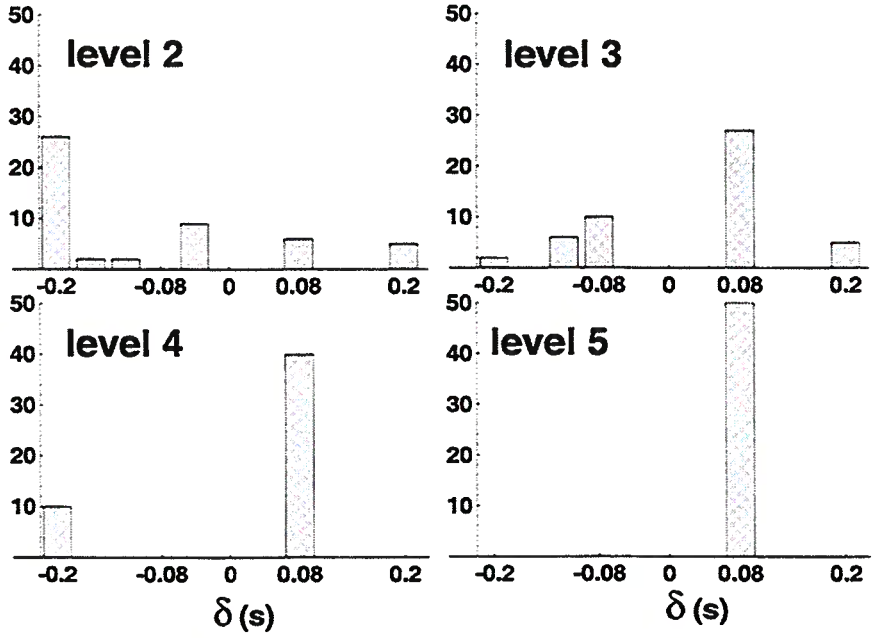

FIG. 4.10. Histograms of the obtained time-shifts of 50 conjugate-gradient optimization experiments for data at various resolution levels. Initial models are chosen randomly between $[-0.2, 0.2]$ s. The horizontal axis is the number of shift-samples, where the sample interval is 0.01 s, and the grid size of the histograms is four samples. All 50 experiments found the true solution when the data were decomposed to resolution level 5.

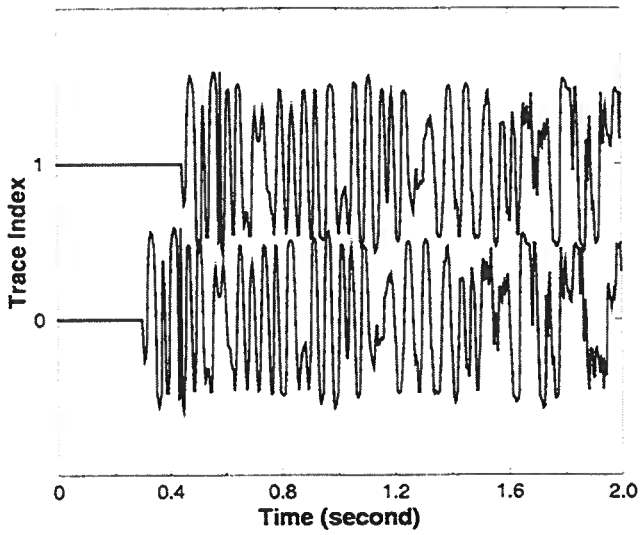

FIG. 4.11. A field seismic trace and its duplicate with an unknown shift.

data are used in this example. In the next example, an optimization applied to the coarse-resolution data will not suffice.

For more complex optimization problems, further reduction of resolution may be needed to make objective functions convex. In the next section, I show that an optimal level of decomposition can be determined using the complexity analysis developed in previous chapters.

Figure 4.11 shows a trace taken from a field seismic record, and its duplicate with some time-shift. Figure 4.12 shows the objective function for this problem. Due to the oscillatory nature of the seismic field data, the objective function shows complicated local and global structure. The basin of attraction leading to the global minimum point is extremely narrow and steep, which would make it difficult for any searching method to find the correct solution.

Again, the auto-correlation shell of the Daubechies basis is used to decompose the traces to coarse resolution levels. Figure 4.13 shows objective functions for various level of decomposition applied to traces in Figure 4.11. As expected, the complexity of the objective function is greatly reduced after the data have been decomposed to coarse levels.

It is worth noticing, however, that the global minimum point is slightly shifted in Figure 4.13(d), though the objective function shows a nice, convex shape. This problem may be caused by the loss of information when too much resolution was discarded from the data. In this case, an iterative process similar to a multi-grid iteration can be used to enhance the resolution progressively; *i.e.*, the solution of a coarse-level optimization is used as the initial model to a following optimization at a

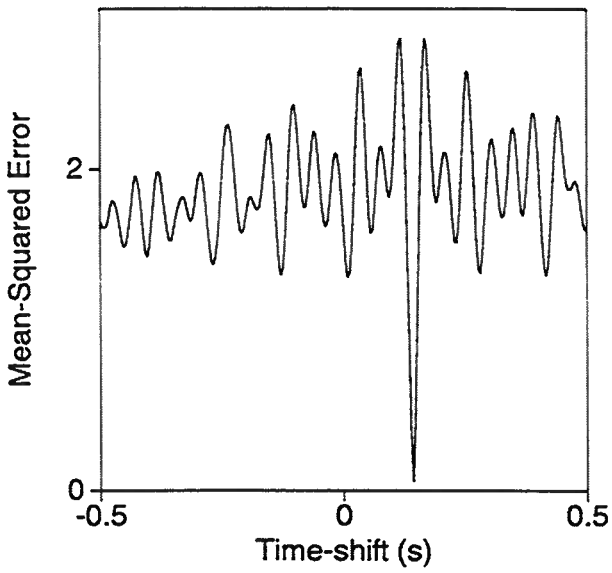

FIG. 4.12. The mean-squared error function for the two field seismic traces shown in Figure 4.11.

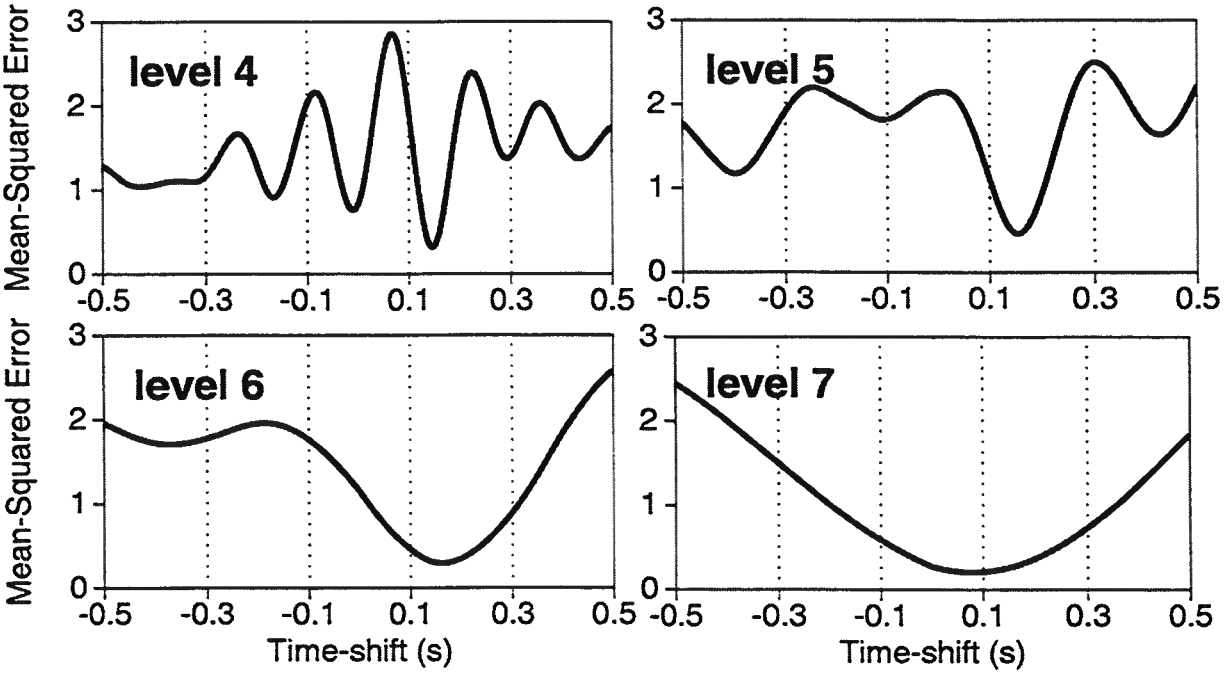

FIG. 4.13. Mean-squared error functions for the two field seismic traces shown in Figure 12 at various resolution levels. The traces are decomposed with the auto-correlation shell of the Daubechies basis with two vanishing moments.

finer level. Such an algorithm can be described as follows:

Algorithm 5 MRHC ($\vec{\mathbf{m}} = \mathrm{MRHC}(F, L, \epsilon, cmax)$)
Let $\{\mathbf{S}_i\}_{i=L,\cdots,0}$ *be a sequence of decreasingly smooth operators to be defined below, with* $\mathbf{S}_0$ *an identity operator.*

1. *Let* $f_L = \mathbf{S}_L F$*; choose an initial population* $\{\mathbf{m}_k^0\}_{k=1,\cdots,K}$ *with size* K *at random; apply Algorithm 2, so* $\{\mathbf{m}_k\}_{k=1,\cdots,M_L} = \mathrm{RHC}(f_L, K, \epsilon, cmax)$*, and* $i = L - 1$.

2. *Let* $f_i = \mathbf{S}_i F$*,* $(L > i \geq 0)$ *and* $\mathbf{m}^0 = \{\mathbf{m}_k\}_{k=1,\cdots,M_{i-1}}$*; run Algorithm 2,* $\{\mathbf{m}_k\}_{k=1,\cdots,M_i} = \mathrm{RHC}(f_i, K, \epsilon, cmax)$.

3. *Decrease the level index* i *by* 1*, repeat* 2 *until* $i = 0$*. The final set of models* $\mathbf{m}$ *is the solution.*

The smoothing operators $\{\mathbf{S}_i\}_{i=L,\cdots,0}$ could be a sequence of low-pass filters with increasingly wider pass-band (Bunks *et al.*, 1995), or a sequence of increasingly fine wavelet operators (Deng, 1995) for decomposing the input seismic data. The sequence of smoothing operators should be such that the resulting functions, $\{f_i\}_{i=L,\cdots,0}$, have the same global shape as does the objective function F for all levels and have a decreasing number of local optima when the level increases, and $f_0 = F$. I showed with the above one-dimensional examples that this could be achieved using a shift-invariant wavelet basis.

4.2.3 Using Complexity Analysis to Evaluate the Behavior of the MRA

The simple one-dimensional examples above show that MRA may be used for simplifying objective functions. However, it is not clear that an MRA can help with any kind of difficult optimization problem. According to the No-Free-Lunch (NFL) Theorem (see page 11), the MRA approach cannot be a superior strategy for solving all optimization problems. For the stacking-power function, in particular, it is necessary to study more realistic examples in order to draw a reasonable conclusion on whether or not this is a useful strategy. Unfortunately, when the dimensionality is high, we do not have a way of visualizing behavior of stacking-power functions as we did with one-dimensional functions.

Using the tool of the complexity analysis developed in this thesis, the behavior of the objective function when applied MRA can be closely studied. Next, I study the residual statics problem for a synthetic data set when surface-consistency (see page 105) is assumed. Figure 4.14 shows the recording geometry of a data set which has 20 sources, 35 distinct receivers and 320 traces. All traces are identical except for random source and receiver statics. These traces are generated by repeatedly shifting a single trace of field data. Thus, equation (C.1) of Appendix C is used as the objective function, which is of dimension 55. When there are no static-shifts in the data, the global minimum of the function is at the origin ($\mathbf{s}_i = \mathbf{0}$, $\mathbf{r}_j = \mathbf{0}$). Figure 4.15

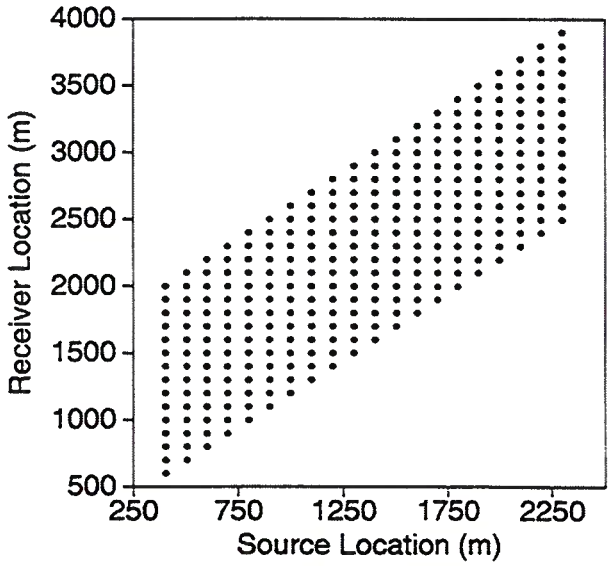

FIG. 4.14. Recording geometry of a synthetic data set. The horizontal axis is the source position and vertical axis the receiver position.

shows a 2-D projection of this 55-dimensional stacking-power function in which all parameters are fixed except for two components. The left plot is the slice when the 53 other components are fixed at their correct values. For this situation, the apparent minima are regularly distributed on the function surface, and the global minimum is located at the origin of this hyper-plane. In contrast, for the right plot, which is the projection when the fixed components are chosen at random, the structure is more irregular, likely with probably more local minima.

For such highly multi-modal functions, local-descent searches usually fail to find global optima. Some form of Monte Carlo global search has been almost always necessary for solving large-scale statics problems (Rothman, 1985; Rothman, 1986). However, most of these global search algorithms, such as SA or GA, are computational intensive, and there is no guarantee that these searches will find the global minima within a reasonable time.

I apply the complexity analysis to evaluate the behavior of the MRA on this 55-dimensional problem. Figure 4.16 shows $\hat{C}_e$ as a function of the wavelet decomposition level using a population size of $K = 1000$; the mean and one-standard deviation error bars are obtained from 32 independent calculations. Results are shown for six levels of decomposition, using a wavelet operator $\{\mathbf{S_i}\}_{i=0,\ldots,5}$ where $i = 0$ is an identity operator, corresponding to use of the original data. These results indicate that for this particular problem a complexity minimum is achieved for a wavelet decomposition of level 4. Higher levels of decomposition actually increase the estimated complexity; presumably this results from the objective function being too flat for local optimization. Thus, the complexity measure gives us a way of choosing a wavelet decomposition level to

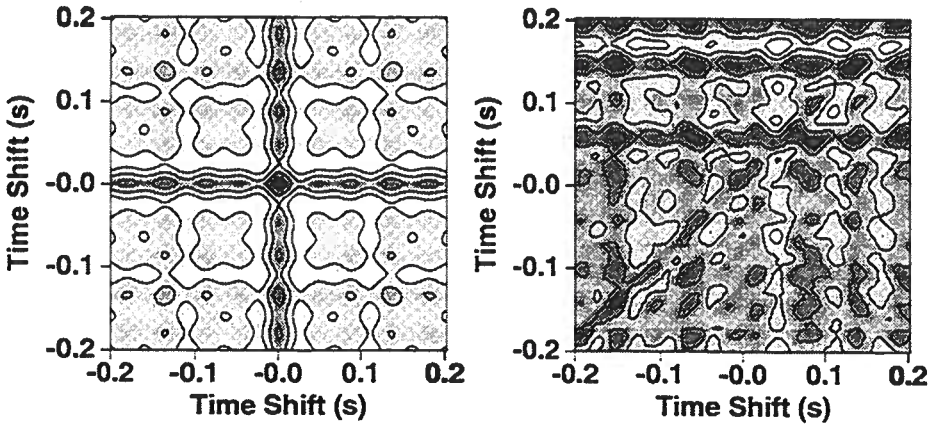

FIG. 4.15. Contour plots of 2-D projects of a 55-dimensional stacking-power function with all statics values held fixed, with the exception of two components. All other components are fixed to their correct values for the left figure, and those components are chosen at random for the right one.

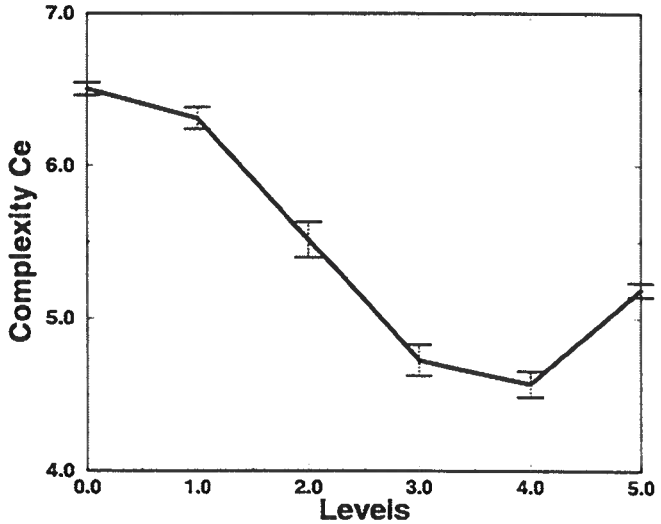

FIG. 4.16. The estimated complexity $\hat{C}_e$ as a function of the level of wavelet decomposition.

achieve optimal simplification of an objective function.

4.3 An Envelope Approach to Simplify Stacking-Power Functions

An alternative approach of suppressing high-frequency information in seismic waveforms is to extract the *envelope* of the signal. As discussed in Appendix C, the conventional linear residual-statics approach (see page 108) uses too little information from the data. When the seismic data are severely contaminated by noise, the time-picking (one-dimensional optimization) could be very difficult. On the other hand, when the global optimization approach (Rothman, 1985; Rothman, 1986) uses the full data set, an additional complication is introduced by optimizing a objective function with unnecessary local extrema. As a compromise between using "too little" (linear approach) and "too much" (global optimization) information from correlations, "partial" information given by the correlation functions can be used for such a high-dimensional optimization problem. Since the complexity of stacking powers is due mostly to complexity of correlation functions, suppressing high-frequency components of the oscillatory correlations will result in a stacking-power function with reduced complexity.

Formally, the envelope of an analytical function is the square-root of the sum of the function and its *Hilbert transform*. The envelope extracts the low-frequency amplitude information while removing the "carrying-frequency" of the signal, which represents the phase information. For a narrow-band correlation function

$$\Phi(\tau) = \exp[-\frac{x^2}{100}]\cos(x), \tag{4.19}$$

for example, its envelope function obtained by the Hilbert transform is

$$\Phi(\tau) = \exp[-\frac{x^2}{100}].$$

Figure 4.17 shows the function in equation (4.19) and its envelope. Such an envelope clearly has fewer oscillations than does the original function, as it carries information on the positions of energy concentration, independent of the locations of individual peaks and troughs of the correlations.

It has been proposed that extracting envelope information from seismic data can facilitate the solution of waveform-inversion problems (Shaw & Orcutt, 1985). Unfortunately, Shaw and Orcutt found that the envelope of seismic data was sensitive to noise. For the residual-statics problem, however, the original objective function is a summation of a set of pre-computed one-dimensional correlation functions. By finding envelopes of the cross-correlation functions rather than of the seismic traces, we work with information that is less sensitive to noise. Figure 4.18 shows an auto-correlation function of a seismic trace from field recorded data and its envelope function. The envelope function clearly has lower frequency content than that of the original signal,

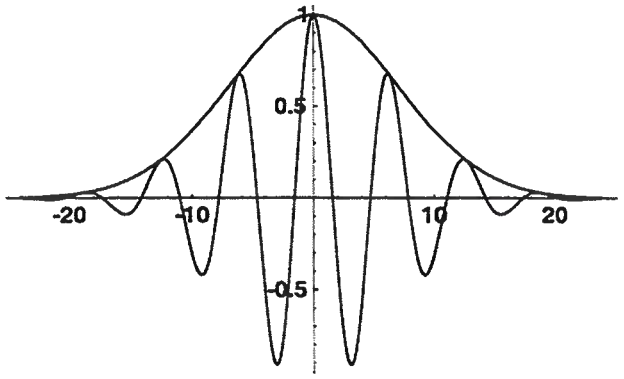

FIG. 4.17. The correlation function, $\exp[-(x/10)^2]\cos(x)$ and its envelope.

but this in itself does not necessarily make the envelope function a good appropriate representation of the original signal. For cross-correlation functions, however, the negative peaks represent out-of-phase alignments among traces. Contributions to significant minima of stacking-power function are primarily due to positive peaks of the correlations of which the stacking-power function is composed of. Therefore, small and negative correlation values are of little interest to us. I propose an alternative definition of envelope function in the residual-statics estimation. This new definition of envelope is obtained by an interpolation of all positive peaks of correlation functions. The cubic-convolution interpolation scheme (Keys, 1981; Keys & Pann, 1993) is used in implementations presented later in this section. This slow-varying function represents global feature of large positive correlation values; henceforth, I refer to this smooth approximation of a one-dimensional signal as *the envelope*. Figure 4.19 shows such an envelope function of the same correlation seismic signal as that shown in Figure 4.18 when cubic-convolution interpolation (Keys, 1981; Keys & Pann, 1993) is used as the interpolation scheme. It can be seen that the envelope obtained by the interpolation of positive peaks can closely resemble general features of the correlation function.

Using the envelope function, we define a new measure of stacking power,

$$\hat{F}(\tau) = \sum_{i=0}^{N-1} \sum_{j=i+1}^{N-1} \left[\hat{\Phi}_{ij}(\tau_j - \tau_i)\right], \tag{4.20}$$

where function $\hat{\Phi}_{ij}(\tau)$ is the envelope of the correlation $\Phi_{ij}(\tau)$. The function in equation (4.20) uses "partial" information from each correlation functions; that is this function ignores all information given by correlations except for the location and amplitude of all positive peaks. Envelope functions contain fewer peaks than do the correlation functions themselves; therefore, one expects that the summation of these envelope functions, equation (4.20), will contain fewer local minima than does the original stacking-power function, *i.e.*, the objective function has its complexity reduced. I shall call $\hat{F}(\tau)$ the *reduced stacking-power* function.

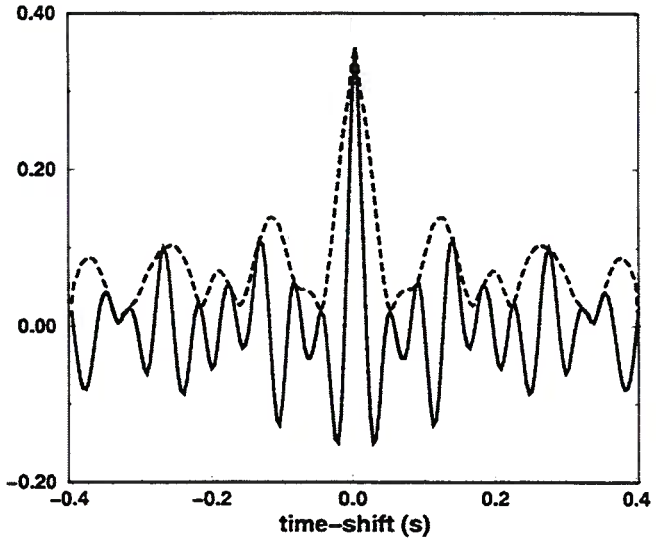

FIG. 4.18. This figure shows a correlation function of a seismic data (solid) and its envelope from Hilbert transform (dashed).

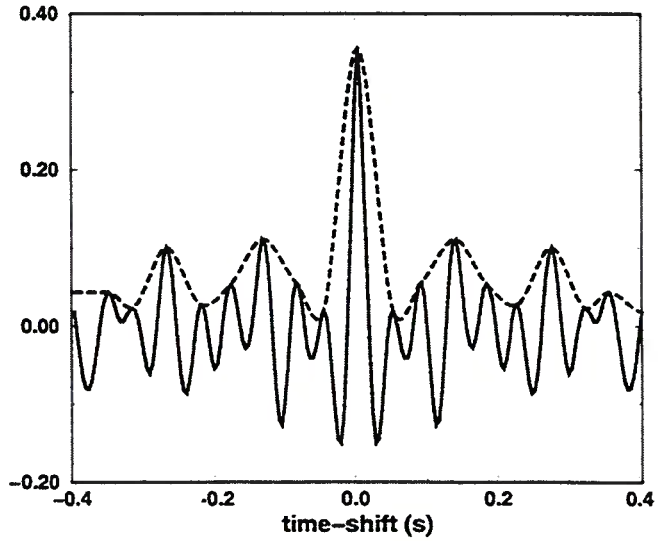

FIG. 4.19. This figure shows the same correlation function as that in Figure 4.18 (solid) and its envelope formed by interpolating the positive peaks (dashed).

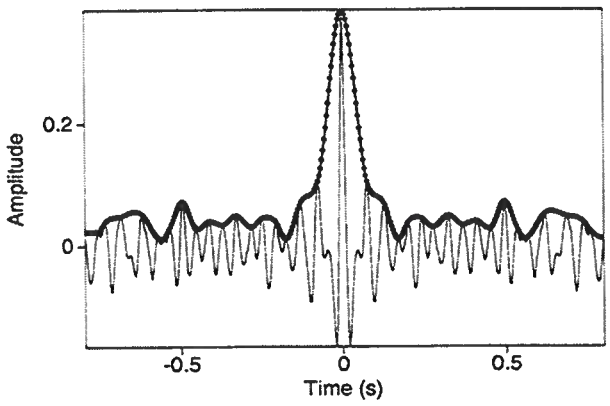

FIG. 4.20. A correlation function between two identical seismic traces and its envelope function.

In order for equation (4.20) to be a useful objective function, it needs to be a qualified measure for the stacking quality. Following Lemma 3 (see page 57), we see that the reduced stacking-power function is, in effect, an interpolated function of many local minima of the original stacking-power function. Following this, we have the following observation about the reduced stacking-power function, which can be easily verified.

Observation 2 (Reduced Stacking-Power Function) *The reduced stacking-power function $\hat{F}$ has three properties:*

1. *$\hat{F}$ contains fewer local minima than does the corresponding original stacking-power function, F.*
2. *$\hat{F}$ maintains the global minimum of the original stacking-power function F, as well as the values of the original stacking-power function at locations of all local minima caused by the maxima of all correlation function simultaneously.*
3. *$\hat{F}$ is an interpolated function of some local minima on its corresponding F, so it maintains significant topographical features of F.*

To demonstrate the above observation, let us again consider the alignment of three identical traces, where the correlation functions are the narrow-band function as shown in equation (4.19). For simplicity, the envelope function is obtained by linearly interpolating local peaks of the correlation function. Both the function and the envelope are shown on the left of Figure 4.21, and the stacking power function $F(\tau_1, \tau_2)$ is shown on the right of Figure 4.21. The left of Figure 4.22 is obtained from a linear interpolation of all local minima of the stacking-power function $F(\tau_1, \tau_2)$. The contour plot on the right in Figure 4.22 is the corresponding reduced stacking-power function. It can be seen that these two plots have the same global feature.

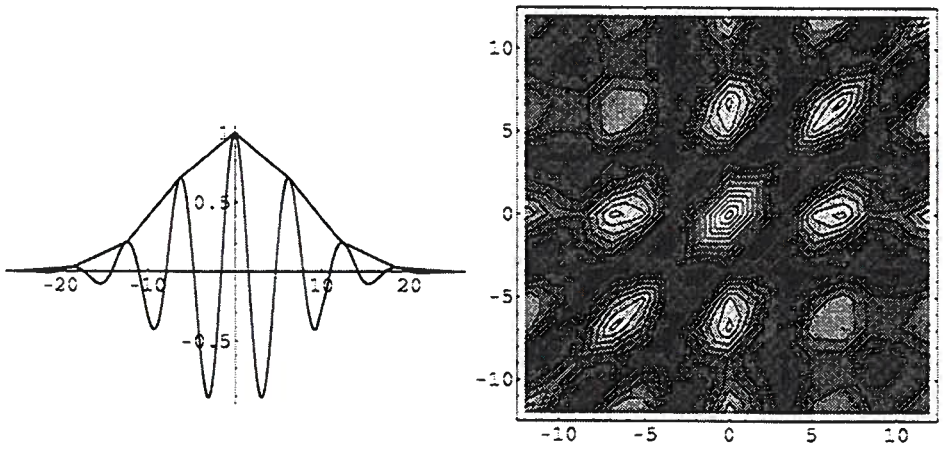

FIG. 4.21. The correlation function, $\exp[-(x/10)^2]\cos(x)$ and its envelope are shown on the left. The two-dimensional stacking-power function $F(\tau_1, \tau_2)$ formed by this correlation is shown on the right.

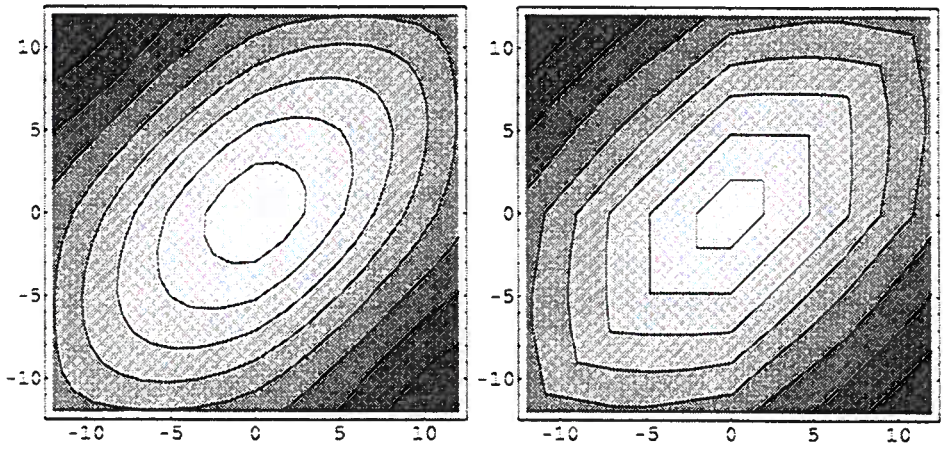

FIG. 4.22. Contour plot of the interpolated function from positive peaks of the original stacking-power function $F(\tau_1, \tau_2)$ is shown on the left. The contour plot of the reduced stacking-power function is shown on the right.

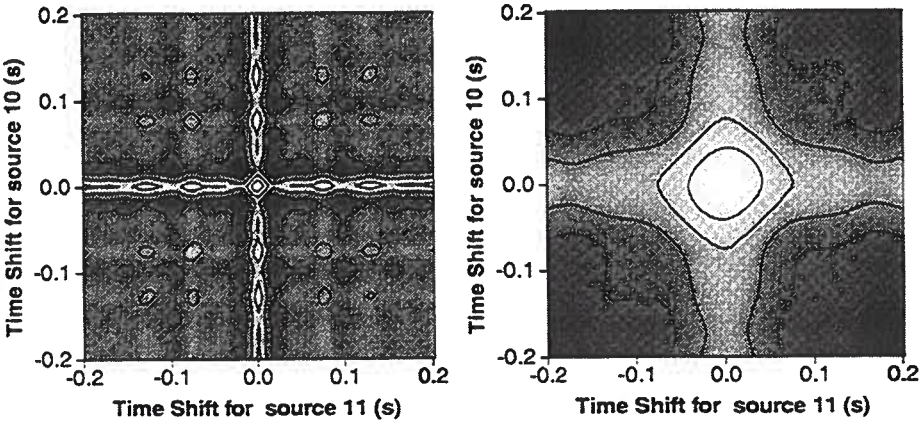

FIG. 4.23. Contour plots of stack powers as a function of static time-shift at the 10th and 11th source points, with all other statics held to be correct. The left figure shows that of the original stacking-power function, while the right figure shows that of the reduced stacking-power function (using envelopes).

4.3.1 Application to a Synthetic Data-Set

Section C.3 of Appendix C describes a practical residual-statics algorithm using the envelope algorithm. The surface-consistency condition is assumed (see page 105), and equation (C.5) is the reduced stacking-power function under the surface-consistent assumption. In this section, this approach is demonstrated by applying Algorithm 7 to a synthetic data set, and the complexity analysis is used to confirm the simplification of the problem.

Once again, the 55-parameter synthetic data set, with recording geometry shown in Figure 4.14 is used. Since all traces are generated by copying and shifting of a field trace, correlations of these traces are therefore shifted versions of the auto-correlation of the field trace. This auto-correlation function and its envelope are shown in Figure 4.20. For this problem, the stacking-power function and its reduced form both have dimensions of 55. When no static-shifts are added to the data, the global minimum of the objective function is at the origin of the model space.

Not being able to visualize the 55-dimensional objective function in this problem, we can observe two-dimensional hyper-planes. Figure 4.23 and 4.24 show behavior of the original stacking power function and its reduced form from slices through the 55-dimensional space. Similar to Figure 4.22, the original stacking-power function appears to have many local maxima, which are scattered regularly on the 2-D slices. On the other hand, the reduced stacking-power function is far more simple.

Using the complexity analysis, we find that $\hat{C}_e = 4.46$ for the reduced stacking-power function when exactly the same initial sampling and local-descent searches are used as those used to obtain the results of Figure 4.16. Compared with the estimated complexity of the original stacking-power function (level 0, $\hat{C}_e = 6.44$) shown in

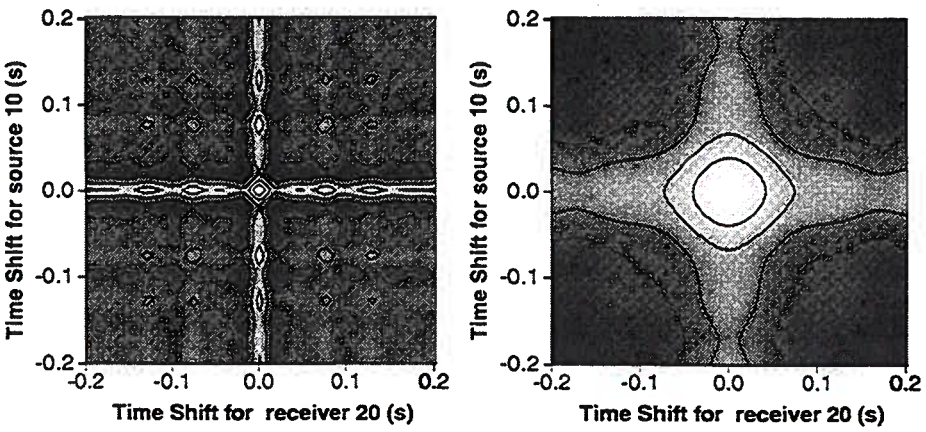

FIG. 4.24. Contour plots of stack power as a function of static time-shift at the 10th source and the 20th receiver points, with all other statics values held to be correct. The left figure shows that of the original stacking-power function, while the right figure shows that of the reduced stacking-power function (using the envelope).

Figure 4.16, this confirms our expectation that the complexity of the reduced stacking-power function is much reduced.

When the above data are contaminated by severe noise and large source and receiver statics, the traces are no longer identical. The left of Figure 4.25 shows the stacked section of this data set contaminated by surface-consistent statics and 50% additive random noise (the ratio of the RMS amplitude of signal to that of noise is two). The noise is band-limited Gaussian noise with the same bandwidth as the original signal. The added statics are generated by a uniformly-distributed random-number generator between $[-100, 100]$ ms. The added and estimated statics are shown in Figure 4.26. The estimated statics in general agree well with the added statics; the observed large error at the beginning and end of the receiver statics are caused by the lack of information when the number of folds of the common-midpoint (CMP) section is low (see page 105). The stacked section after estimated residual-statics are corrected is shown on the right of Figure 4.25. Note the continuity of the horizontal events except at the ends of the section.

In Appendix C, I show the application of this envelope approach to two field data-set. With the analysis given above and these examples, I conclude that the envelope approach in the residual-statics estimation is a practical algorithm, especially for dealing with large static-shifts.

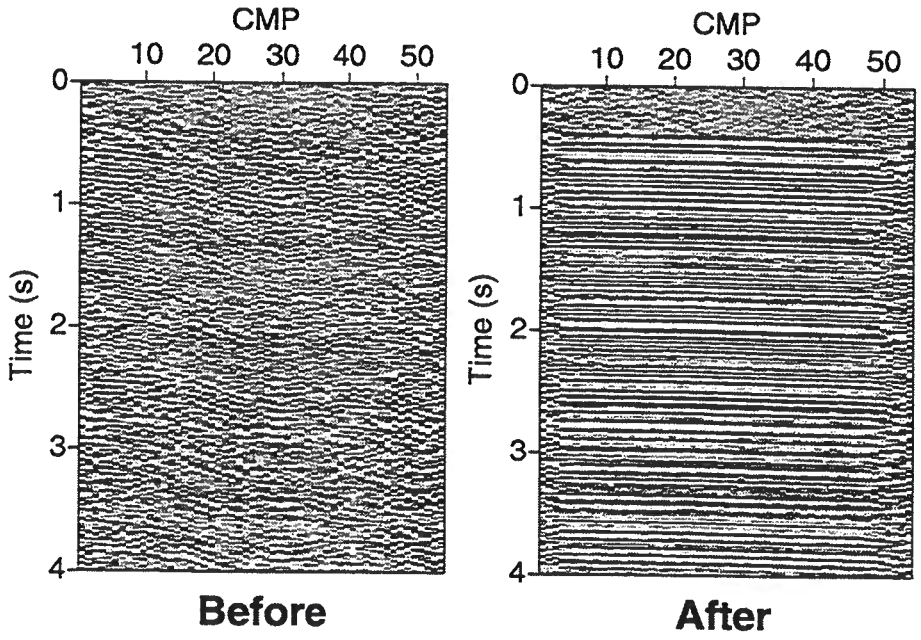

FIG. 4.25. The stacked section in the presence of surface-consistent static-shifts and noise is shown on the left. The stacked section after the source and receiver statics are estimated and corrected is shown on the right. The two figures are plotted with the same amplitude scale.

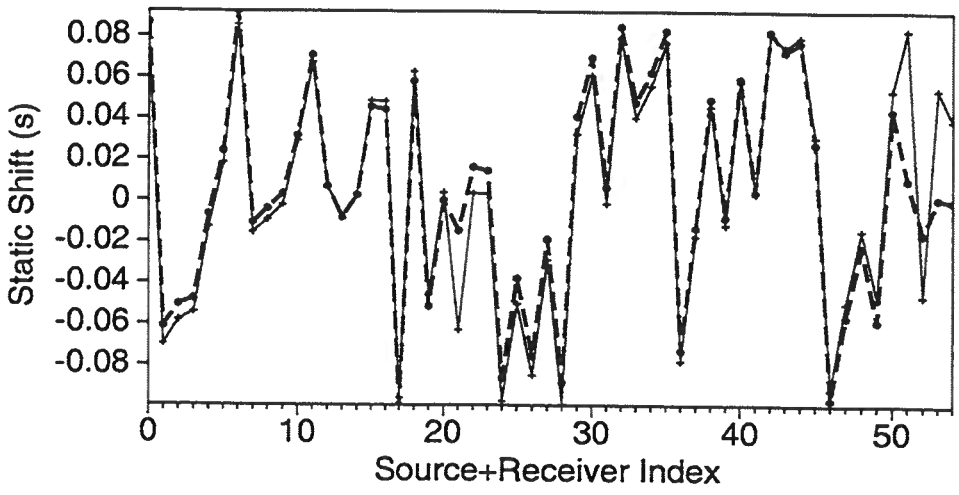

FIG. 4.26. Comparison of the added statics and estimated statics. Horizontal axis is the index of unknowns: $0 - 19$ are sources and $20 - 54$ are receivers. The solid-thin line shows the added statics, while the thick-dashed line shows the estimated statics.

Chapter 5

CONCLUSIONS

5.1 Summary of Major Contributions

Optimization is a widely used tool for solving scientific and engineering problems. The diversity of such problems makes it difficult to have a single criterion for choosing the appropriate optimization algorithms. Especially when the dimensionality is high and our knowledge about the objective function is limited, there is no general rule as to how to design optimization techniques. The No Free Lunch (NFL) theorem (Wolpert & Magready, 1995) states that the performance of any optimization algorithm is the same when averaged over all possible objective functions. The central theme of this thesis is that we cannot usefully compare optimization techniques unless we have a meaningful way of characterizing an optimization problem. Specifically, the main effort of this thesis is to develop a methodology for such a characterization.

In this thesis, I find that the hardness of an optimization problem is directly related to some topographical features of the corresponding objective function. I propose that a generic optimization problem can be characterized by an entropy-based complexity (see Algorithm 3), regardless of the dimensionality of the problem or whether or not the function is available only point-wise. This complexity measure encapsulates three important pieces of topographical information about the function: the number of local minima (n), the width (p_i) and relative depth ($f(\check{\mathbf{m}}_i)$) of these minima. More importantly, I show that for any objective function these topographical features can be estimated by applying some independent local-descent searches (RHC), followed by a statistical analysis on the results of the RHC.

In actual implementations, results of the complexity estimation are influenced by both the number of samples used in the RHC (K) and the numerical implementation of the complexity estimation. The reliability of such results is evaluated using confidence-intervals and numerical error-analysis in Chapter 4. For the purpose of the complexity estimation, it is not necessary that the RHC-searches converge all models to the exact local extrema. The two essential steps in the RHC are: (1) estimation of the width of each of the basins of attraction correctly; (2) evaluation of the function values at the local minima.

Ill-conditioned optimization problems are usually associated with flatness of the objective function. In this case, pre-mature convergence may mimic the presence of local minima. This numerical effect can be reflected in the results of the complexity estimation. Since it is influenced by numerical difficulties in optimization, the estimated complexity represents yet another important aspect of what makes optimization problems hard.

The complexity analysis developed in this thesis can be used for studying any real-valued function whether it is analytical or only available point-wise, especially when the dimensionality is high. Results of this analysis can represent the hardness both caused by the multi-modality and by the ill-posedness of the problem. A promising application of this complexity analysis would be to evaluate the effectiveness of certain strategies in solving hard optimization problems. In this thesis, this application is demonstrated by using complexity analysis to determine the optimal level of decomposition in using MRA on an optimization problem.

Another important result of this research is its insight into the influence of dimensionality on the hardness of generic optimization problems. From the analyses and examples in the thesis, one can see that dimensionality is not directly related to the hardness of problems. The most essential factor is the minimum width of the basins of attraction p_m. Therefore, I suggest replacing the notion of "the curse of dimensionality" with that of "the curse of small basins of attraction". Similarly, the number of samples K needed in the complexity analysis is also directly related to the minimum width of the basins of attraction rather than the dimensionality.

An important application of this complexity analysis is in geophysical inversion problems where high-dimensional, highly multimodal optimization problems are encountered frequently. As an example, I have investigated the problem of residual-statics estimation problem in Chapter 4. I first demonstrate the importance of carefully identifying and taking into account specifics of the problem (see page 56). With the insights gained from these observations, two strategies are proposed for simplifying the objective function: applying a multi-resolution analysis (MRA) to the seismic data, and using the envelope information of the correlation functions. I show that the complexity analysis can serve as an important tool for evaluating the behavior of these strategies. In particular, the optimal level of decomposition for the MRA can be chosen so as to minimize the complexity of the transformed problem.

Finally, development of the CWP Object-Oriented Optimization Library (COOOL) is also part of the contribution of this research. The library provides a wide variety of optimization algorithms, sharing a uniform interface to generic objective functions, making it easy to mix and match optimization algorithms with objective functions. COOOL does not require much knowledge of C++ or objective-oriented programming, and the user-defined objective function can be written in any programming language. In addition to this flexibility, further extension of COOOL is easy due to its object-oriented design.

5.2 Further Studies and Limitations

Many local-descent searches are required for doing this complexity analysis. On the other hand, since sampling of the model space is independent for each search, this process is easy to parallelize. It will also be practical to explore subsets of the model space simultaneously. Therefore, this complexity analysis is a potentially practical tool for studying hard optimization problems.

Furthermore, the convergence rate of global-search methods, such as GA and SA, may be studied under the criterion of this complexity. For example, the complexity may be used in the same way as the expected hitting time (Shonkwiler & Van Vleck, 1994; Morey, 1996); to determine the optimal parameters in SA, such as the neighborhood structure and the temperature (see page 8).

It is necessary for such an analysis, however, that we have a pre-estimated minimum width of the basins of attractions, p_m. This quantity has essential importance for the number of samples required for adequate complexity analysis and implementation strategies discussed in section 3.3. Even with the error estimates obtained from chapter 3, some uncontrollable factors still can compromise results of the complexity estimation. It is extremely important to treat the estimated complexity $\hat{C}_e$ carefully. The reliability of the analysis can be increased by repeated experiments and a Monte Carlo analysis as illustrated in Figure 4.16.

In addition to the value $\hat{C}_e$ obtained by Algorithm 3, the availability of topographical information about the basins of attraction is important for further analysis of the problem. Taking advantage of data obtained during the complexity estimation, the widths and significance of the basins of attraction, $\{p_i\}$ and $\{q_i\}$ (see pages 17 and 18), can be used to analyze the cause of the complexity and each basins of attraction can be studied individually.

When the basin of attraction that leads to the global minimum has small width, it is essentially impossible for any strategy or analysis to work. I do not treat such an extremely pathological case in this study.

REFERENCES

Aarts, E.H.L., & Korst, Jan. 1989. *Simulated Annealing and Boltzman Machines.* N.Y: Wiley.

Berry, R. S., & Breitengraser-Kunz, R. 1995. Topography and Dynamics of Multidimensional Interatomic Potential Surface. *Physical Review Letters*, **74**, 3951–3954.

Bunks, C., Saleck, F. M., Zaleski, S., & Chavent, G. 1995. Multiscale seismic waveform inversion. *Geophysics*, **60**(5), 1457–1473.

Chen, T. 1994. *Multilevel Differential Semblance Optimization for Waveform Inversion.* Tech. rept. CWP-153. Center for Wave Phenomena, Colorado School of Mines.

Culberson, J. C. 1996 (July). *On the Futility of Blind Search.* Tech. rept. TR 96-18. Department of Computing Science, The University of Alberta, Edmonton, Alberta, Canada.

Daubechies, I. 1992. *Ten Lectures on Wavelets.* SIAM.

Davis, T. E. 1991. *Toward an Extrapolation of the Simulated Annealing Convergence Theory Onto the Simple Genetic Algorithm.* Ph.D. thesis, University of Florida, Gainesville, Florida.

Davis, T. E., & Principe, J. C. 1991. A simulated annealing like convergence theory for the simple genetic algorithm. *In:* Belew, R. K., & Booker, L.B. (eds), *Proceedings of the fourth international conference on genetic algorithms.* San Mateo, Calif.: Morgan Kaufmann Publishers.

Deng, H. L. 1995. *Using Multi-Resolution Analysis to Study the Complexity of Inverse Calculations.* Tech. rept. CWP-183. Center for Wave Phenomena, Colorado School of Mines.

Deng, H. L., Gouveia, W., & Scales, J. A. 1996a. The CWP Object-Oriented Optimization Library. *The Leading Edge*, **15**(5), 365–369.

Deng, H. L., Gouveia, W., & Scales, J. A. 1996b. An Object-Oriented Toolbox for Studying Optimization Problems. *Pages 320–330 of:* Jacobsen, B. H., Moosegard, K., & Sibani, P. (eds), *Inverse Methods, Interdisciplinary Elements of Methodology, Computation, and Applications.* Berlin, Germany: Springer-Verlag.

Dennis, J. E., & Schnabel, R. B. 1987. *Numerical methods for unconstrained optimization and nonlinear equations.* Prentice-Hall Inc.

Falcioni, M., Marconi, U. M. B., Ginanneschi, P. M., & Vulpiani, A. 1995. Complexity of the Minumum Engery Configurations. *Physical Review Letters*, **75**, 637–640.

Fletcher, R. 1987. *Practical Methods of Optimization.* John Wiley & Sons.

Freund, J. E. 1992. *Mathematical Statistics.* 5 edn. Englewood Cliffs, New Jersey: Prentice Hall.

Goldberg, D. E. 1989. *Genetic Algorithms in Search, Optimization, & Machine Learning.* Reading, Massachusetts: Addison-Wesley.

Goldburg, R. R. 1976. *Methods of Real Analysis.* 2 edn. New York: John Wiley & Sons, Inc.

Griewank, A. O. 1981. Generalized descent for global optimization. *JOTA*, **34**(34), 11–39.

Hajek, B. 1988. Cooling schedules for optimal annealing. *Mathematics of Operations Research*, **13**, 311–329.

Holland, J. H. 1975. *Adaptation in Natural and Artificial Systems.* Ann Harbor, MI: University of Michigan Press.

Hu, X., Shonkwiler, R., & Spruill, M.C. 1993. Random Restart in Global Optimization. *SIAM Journal on Optimization.* submitted.

Hu, X., Shonkwiler, R., & Spruill, M.C. 1994. Approximate Speedup by Independent Identical Processing. *preprint.*

Jawerth, B., & Sweldens, W. 1994. An overview of wavelet based multiresolution analyses. *SIAM Review*, **36**, 377–412.

Jones, Terry. 1994. *A Model of Landscape.* Tech. rept. 94-02-002. Santa Fe Institute.

Kauffman, S. A., & Weinberger, E. D. 1989. The NK model of rugged fitness landscapes and its application to maturation of the immune response. *Journal of Theoretical Biology*, **141**(2), 211.

Kaufmann, S. 1993. *The Origins of Order: Self-Organization and Selection in Evolution.* New York: Oxford University Press. Chap. 2-3, pages 33–117.

Keys, R. G. 1981. Cubic convolution interpolation for digital image processing. *IEEE Trans. on Acoustics, Speech and Signal Processing.*

Keys, R. G., & Pann, K. 1993. Geophysical Applications of Cubic Convolution Interpolation. *Pages 162–165 of: 63nd Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts.* Soc. Expl. Geophys.

Kirkpatrick, S., Gelatt, C.D., & Vecchi, M.P. 1983. Optimization by simulated annealing. *Science*, **220**, 671–680.

Magready, W. G., & Wolpert, D. H. 1995-1996. What Makes an Optimization Problem Hard? *Complexity*, **1:5**, 40–46.

Mallat, S. 1989. A theory for multiresolution signal decomposition. *IEEE Trans. Pattern Anal. Machine Intell.*, **11**(Jul.), 674–693.

Meyer, Y. 1992. *Wavelets and Operators*. Cambridge: Cambridge University. translated by D.H. Salinger.

Monzon, Lucas Alejandro. 1994 (May). *Constructive Multiresolution Analysis and the Structure of Quadrature Mirror Filters*. Ph.D. thesis, Yale University.

Morey, C. 1996 (July). *Dynamically Determining Search Parameters for Monte Carlo Optimization*. Ph.D. thesis, Colorado School of Mines, Golden, Colorado.

Morey, C., Scales, J. A., & Van Vleck, E. S. 1996. Dynamically Determining Search Parameters for Monte Carlo Optimization. *preprint*.

Neidell, N. S., & Taner, M. T. 1971. Semblance and other coherency measures for multichannel data. *Geophysics*, **36**, 482–497.

Radcliffe, N. J., & Surry, P. D. 1996. *Fundamental Limitations on Search Algorithms: Evolutionary Computing in Perspective*.

Ronen, J., & Claerbout, J. 1985. Surface-consistent residual statics estimation by stack-power maximization. *Geophysics*, **50**, 2759–2767.

Rothman, D. H. 1985. Nonlinear inversion, statistical mechanics, and residual statics estimation. *Geophysics*, **50**, 2797–2807.

Rothman, D. H. 1986. Automatic estimation of large residual statics corrections. *Geophysics*, **51**, 332–346.

Saito, N., & Beylkin, G. 1993. Multiresolution representations using the auto-correlation functions of compactly supported wavelets. *IEEE Transactions on Signal Processing*, **41**, 3585–3590.

Saito, Naoki. 1994 (December). *Local Feature Extraction and Its Applications Using a Library of Bases*. Ph.D. thesis, Yale University.

Saleck, F. M., Bunks, C., Zaleski, S., & Chavent, G. 1993. Combining the multigrid and gradient methods to solve the seismic inversion problem. *Pages 688–691 of: 63rd Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts*. Soc. Expl. Geophys.

Scales, J. A., & Smith, M. 1994. *Introductory Inverse Theory.* Samizdat Press. http://landau.mines.edu/~samizdat.

Scheoen, F. 1991. Stochastic Techniques for Global Optimization: a Survey of Recent Advances. *J. of Global Optimization*, **1**, 207–228.

Shannon, C. E. 1948. The mathematical theory of communication. *Bell Syst. Techn. Journ.*, **27**, 623–656.

Shaw, P., & Orcutt, J. 1985. Waveform inversion of seismic refraction data and applications to young Pacific crust. *Geophysical Journal of the Royal Astronomical Society*, **82**, 374–414.

Shonkwiler, R., & Van Vleck, E. S. 1994. Parallel Speed-Up of Monte Carlo Methods for Global Otimization. *Journal of Complexity*, **10**, 64–95.

Siegel, Dirk. 1991 (September). *A new iterative approach to solving the statics problem.* Tech. rept. Amoco Production Company Research Center, Tulsa, OK.

Smith, M., Scales, J. A., & Fischer, T. 1992. Global search and genetic algorithms. *The Leading Edge of Exploration.* **11**, 22–26.

Stadler, P. F. 1992. Correlation in landscapes of combinatorial optimization problems. *Europhysics Letters*, **20**(6), 479–482.

Stadler, P. F., & Happel, R. 1992. Correlation structure of the landscape of the graph-bipartitioning problem. *Journal of Physics. A*, **25**(11), 3103–3110.

Stadler, P. F., & Schnabl, W. 1992. The landscape of the traveling salesman problem. *Physics Letters A*, **161**, 337–344.

Taner, M. T., Koehler, F., & Alhilali, K. A. 1974. Estimation and correction of near-surface time anomalies. *Geophysics*, **39**, 441–463.

Törn, A. A. 1977. Cluster Analysis Using Seed Points and Density-Determined Hyperspheres as an Aid to Global Optimization. *IEEE Transactions on System, Man, and Cybernetics*, **SMC-7**(8), 610–616.

Törn, A. A., & Žilinskas, A. 1989. *Global Optimization.* Berlin, Germany: Springer-Verlag.

van Laarhoven, P.J.M., & Aarts, E.H.L. 1987. *Simulated Annealing: Theory and Practice.* Dordrecht: Reidel.

Weinberger, E. D. 1990. Correlated and uncorrelated fitness landscapes and how to tell the difference. *Biological Cybernetics*, **63**, 325–336.

Whitley, D. 1993. *A generic algorithm tutorial.* Tech. rept. CS-93-103. Colorado State University.

Whitley, D., Mathias, K., Rana, S., & Dzubera, J. 1995a. Building Better Test Functions. San Mateo, Calif.: Morgan Kaufmann Publishers.

Whitley, D., Mathias, K., Rana, S., & Dzubera, J. 1995b. *Evaluating Evolutionary Algorithms: The Perils of Poor Empiricism.* Tech. rept. Department of Computer Science, Colorado State University.

Whitley, D., Beveridge, R., Graves, C., & Mathias, K. 1995c. Test Driving Three 1995 Genetic Algorithms: New Test Functions and Geometric Matching. *Journal on Heyristics*, **1**, 77–104.

Wiggins, R. A., Larner, K. L., & Wisecup, R. D. 1976. Residual statics analysis as a general linear inverse problem. *Geophysics*, **41**, 922–938.

Wolpert, D. H., & Magready, W. G. 1995. No Free Lunch Theorems for Search. *Oper. Res.*, **submitted.**

Zupan, Jure. 1982. *Clustering of Large Data Sets.* Chemometrics Research Studies Series. Research Studies Press.

Appendix A

CWP OBJECT-ORIENTED OPTIMIZATION LIBRARY (COOOL)

The CWP object-oriented optimization library (COOOL) is a tool for studying optimization problems and for aiding in the development of new optimization software. COOOL consists of a collection of C++ *classes* (encapsulation of abstract ideas), a variety of optimization algorithms implemented using these classes, and a collection of test functions, which can be used to evaluate new algorithms. To use one of the optimization algorithms, a user codes an objective function (and its derivatives if available) according to a simple input/output model — in any language. Moreover, the classes themselves can facilitate the development of new optimization and numerical linear-algebra software since the routine aspects of such coding will benefit from the reusability of code inherent in the object-oriented philosophy.

COOOL is the basic tool kit for the research in this thesis. All numerical results, especially those of RHC, are produced by the local-descent search algorithms in the COOOL library. I give a brief description of this library in this appendix.

A.1 Design of COOOL

The three criteria behind the development of COOOL are:

1. It provides a consistent application programming interface (API) for solving optimization problems.

2. It allows incremental development of the library.

3. It allows application packages to be easily built from the existing library.

By consistent, I mean that the formats of the optimization algorithms and objective function should be relatively transparent to application users. Figure A.1 schematically illustrates the design of COOOL. Such design of the library provides flexibility for users to choose optimization methods easily and concentrate on their specific problems rather than struggling to fit the requirements of the various formats of optimization algorithms.

By using the object-oriented paradigm in the design and programming, COOOL is able to achieve the goal stated above. Figure A.2 shows the layered structure used in the design of the COOOL library. The base level contains the simplest class templates in COOOL, such as **Vector**, **Matrix**, **List**, **Astring**, *etc.* These classes are the basic elements of COOOL for handling algebraic computations. Special classes, such as **SpaMatrix** and **DiagMatrix**, handle sparse and diagonal matrices efficiently. The

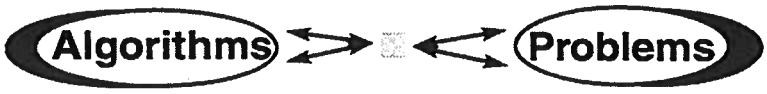

FIG. A.1. The design goal of COOOL is to be able to easily include a large variety of practical optimization problems as well as a large collection of optimization algorithms. The interface, however, between these two large varieties should be small and consistent.

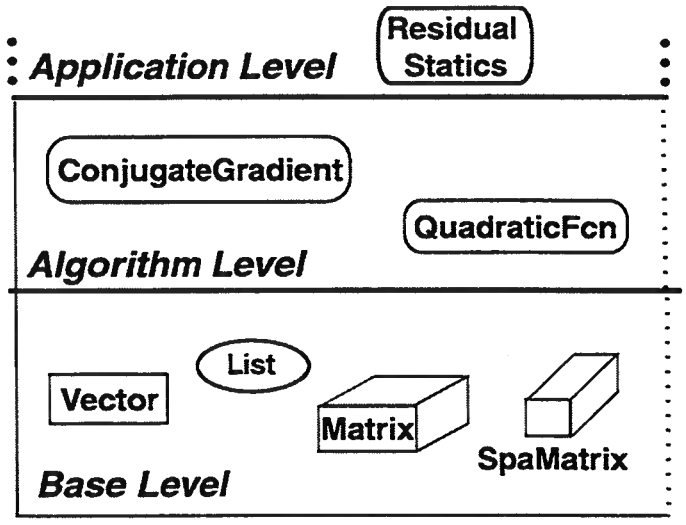

FIG. A.2. Structure of COOOL. The base level contains generic, simple algebraic data structures. The main level of COOOL is the algorithm level, which contains prototypes for the necessary components in any optimization problems (see text). Both levels are extensible with minimal influence on existing classes. Application packages at the top can be easily built with lower-level objects.

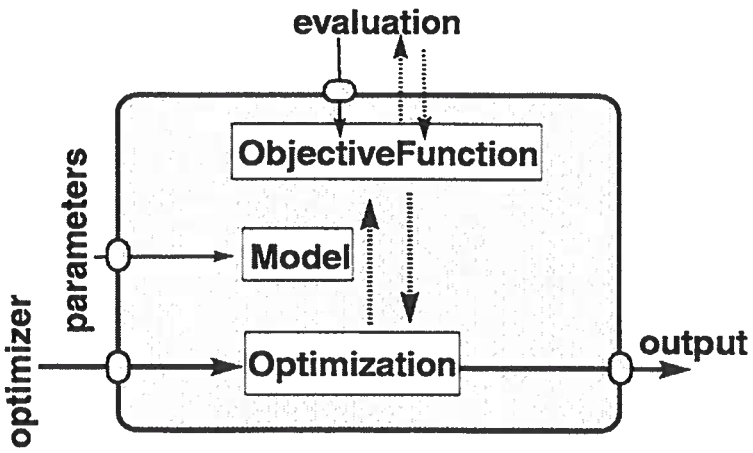

FIG. A.3. There are three necessary elements in any optimization procedure: the model space, the objective function, and the optimization algorithm. They each interact with different aspects of the problem, and the interaction among themselves is taken care of by COOOL.

algorithm level is the main part of COOOL, containing classes for three necessary components in any optimization problems: unknown parameters (**Model**), objective functions (**ObjFcn**), and optimization algorithms (**Optima**). The application level is the most special purpose and sophisticated level where problem-specific packages can be built with existing classes from lower levels. For example, we should be able to build a package for a certain residual-statics problem with the flexibility of choosing any of the mathematical optimization methods. The structure of COOOL shown in Figure A.2 is easily extensible, where all classes at each levels can be easily accessed (by either adding or modifying) with minimum influence on others.

A key point in COOOL is that communication between optimization algorithms and objective functions is transparent to users. To use COOOL, a user needs only to construct several objects by choosing from the library; namely, a **Model**, an **ObjFcn**, and an **Optima**. The interior computation and communication among these objects are handled by COOOL. Figure A.3 shows that, from this level of COOOL, there are only three interfaces where users need to choose the correct objects from the derived classes of each abstract class.

A.2 Optimization Methods in COOOL

Figure A.4 shows a classification of the optimization methods included in the current release of the library. The two main types of optimization algorithm are: linear solvers and local optimization methods. Global optimization methods will be part of a future release.

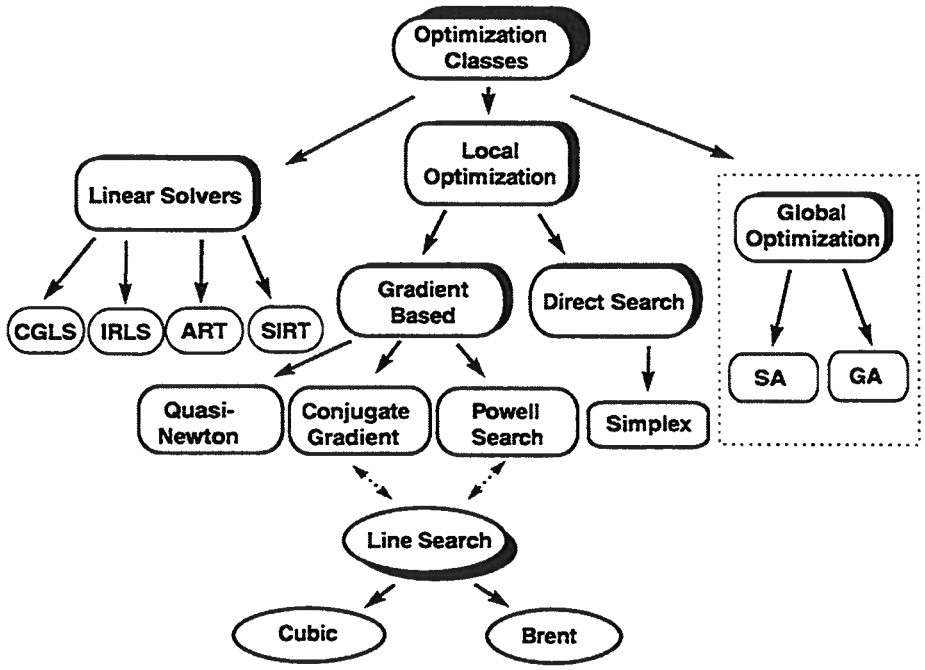

FIG. A.4. An overview of the optimization algorithms contained in COOOL. Included are linear solvers for rectangular linear systems using both least squares and general ℓ_p norms, and local methods based on direct search and quasi-Newton methods. Monte Carlo global optimization methods will be part of a future release of COOOL.

A.2.1 Linear Solvers

All of the linear solvers in COOOL are iterative. Descriptions of most algorithms can be found in Scales and Smith (1994). Presently COOOL implements two flavors of row action method: **ART** (Algebraic Reconstruction Technique) and **SIRT** (Simultaneous Iterative Reconstruction Technique). These methods are especially attractive when problems are too big to fit into memory. When the non-zero matrix elements fit into memory, **CGLS** (Conjugate Gradient Least Squares) is very attractive. Finally, a version of the **IRLS** (Iteratively Re-weighted Least-Squares) algorithm is also included to efficiently solve linear systems in the ℓ_p norm. This algorithm takes advantage of certain approximations to achieve nearly the speed of conventional least squares methods while being able to robustly handle long-tailed noise distributions.

A.2.2 Local Optimization Methods

The algorithms presented here can be applied to quadratic and non-quadratic objective functions alike. The term "local" refers both to the fact that only information about a function from the neighborhood of the current approximation is used in updating the approximation, and that these methods usually converge to extrema near the starting models. As a result, the global structure of an objective function is unknown to a local method. Some of these techniques, such as Downhill Simplex and Powell's method do not require explicit derivatives of the objective function. Others, such as the quasi-Newton methods, require at least the gradient. In the latter case, if analytic expressions are not available for the derivatives, a module for finite-difference calculation of the gradient is provided. COOOL also includes non-quadratic generalizations of the conjugate gradient method incorporating two different kind of line search procedures.

A.3 Objective Functions

Objective functions are those functions to be minimized (or maximized). Several analytical test functions are implemented in the library. They include a generalized N-dimensional quadratic function, N-dimensional Rosenbrock function, N-dimensional Griewank function and a two-dimensional multimodal analytical function.

For realistic problems, the formulation of objective functions may vary tremendously. A user can write the objective function as a stand-alone Unix executable which reads models from its standard input and writes numbers to its standard output. COOOL initiates this executable (using ideas developed by Don Libes in his *Expect* package) in such a way as to take over the objective function's input and output. (This communication between COOOL and the objective function is called a pseudotty in Unix and is somewhat like a pipe.) The net result is that the objective function can be written in any language and may take advantage of hardware-specific features. This communication model is illustrated in Fig. A.5. COOOL sends out a flag to the user-defined objective-function file, indicating whether the function value or its

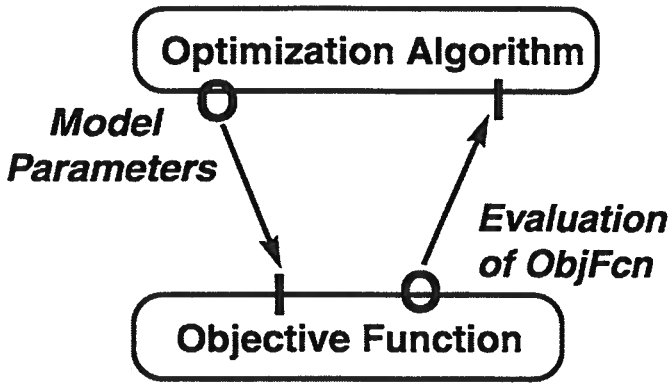

FIG. A.5. A model of communication between an objective function and an optimization class. The user-defined stand-alone objective function is responsible for evaluating the value or gradient for a set of parameters according to the message sent by the optimization algorithm.

derivatives are needed, followed by a set of model parameters. The objective function reads the flag and the model from standard input, and writes the result to standard output; COOOL will then capture this information and supply it to an optimization algorithm for an updated model. If no analytical gradient information is available, COOOL will approximate the gradient vector using a finite-difference technique. This simple communication model allows us to consider algorithms as diverse as downhill simplex, Newton's method, and simulated annealing (SA) within a unified framework.

A.4 Model Spaces

The third necessary component in any optimization problem is the domain of the objective function - model space. For representing specific problems, an object belonging to the class **Model** needs to be constructed by the user. The attributes of the model space are encapsulated in the **Model** class constructed by the user, and these features are handled internally, independent of the **ObjFcns** and **Optima**. With such an encapsulation, the **ObjFcns** and **Optima** classes do not have to deal with whether the model space is an N-dimensional real space $\mathbf{R}^N$ (continuous without boundary), a hyper-cube (continuous with boundary), an N-dimensional integer space $\mathbf{I}^N$ (discrete without boundary), or a hyper-lattice (discrete with boundary).

A recent addition to COOOL is that inequality constraints on the model space can be handled by the **Model** class as well. The user needs to code a function that returns an integer indicating whether or not the current model satisfies the constraints.

A pseudo-code for solving a descretized optimization problem using COOOL is shown in Figure A.6. Although COOOL is written in C++, little knowledge of C++ is needed for using it. Figure A.7 is the pseudo-code for evaluating the objective function and the gradients. This code can be written in FORTRAN or any other language and

```
main(argc, **argv)
{
        // Construct a constrained Model with n unknowns
        Model m(n, upper, lower, Δm);
        // Construct an ObjFcn: user-defined function, statics
        ObjFcn *f = new ObjFcn("statics", ...);
        // Construct an Optima: choose an optimization method
        Optima *opt = new ConjugateGradient(f, imax, tol, ...);

        // Initialize the model to m0
        m = m0;

        // COOOL returns an optimal model optm
        Model optm = opt → optimizer(m);
}
```

FIG. A.6. Pseudo-code for using COOOL to solve an optimization problem. Constraints of model parameters are separated from optimization algorithms. The model space is bounded by the interval $[lower, upper]$. If the model is descretized, the grid-size Δm is specified when constructing the **Model** object.

compiled independently without linking to COOOL. Evaluations of objective functions may be the most computationally intensive step for many optimization problems; COOOL allows users to code them in the most efficient language available.

```
...
while (cin ≫ iflag) {
        cin ≫ m;
        if (iflag == 0) {
                evaluate the objective function at m;
                cout ≪ F(m);
        }
        else {
                evaluate the gradient at m;
                cout ≪ g⃗(m);
        }
}
...
```

FIG. A.7. Pseudo-code for constructing an objective-function-evaluation problem. This code could be written in either C, C++, or FORTRAN.

Appendix B

MULTI-RESOLUTION ANALYSIS (MRA)

Multiresolution analysis (MRA) was formulated based on the study of orthonormal, compactly supported wavelet bases. Wavelet theory and its applications are rapidly developing fields in applied mathematics and signal analysis. Wavelet basis representation of certain signals show advantages over the traditional Fourier basis representation both theoretically and practically. The MRA concept, initiated by Meyer (1992) and Mallat (1989), provides a natural framework for the understanding of wavelet bases. Here, I give a brief description of orthonormal, compactly supported wavelet bases; detailed information can be found, for example, in Daubechies (1992) and Jawerth and Sweldens (1994).

An orthonormal, compactly-supported wavelet basis of $\mathbf{L}^2(\mathbf{R})$ is formed by the dilation and translation of a single function $\psi(x)$, called the wavelet function:

$$\psi_{j,k}(x) = 2^{-j/2}\psi(2^{-j}x - k); \quad j,k \in \mathbf{Z}. \tag{B.1}$$

where $\mathbf{Z}$ is the set of integers. In equation (B.1), the function ψ has M vanishing moments up to order $M-1$, and it satisfies the following "two-scale" difference equation,

$$\psi(x) = \sqrt{2}\sum_{k=0}^{L-1} g_k \psi(2x-k). \tag{B.2}$$

The wavelet function $\psi(x)$ has a companion, the scaling function $\phi(x)$, which also forms a set of orthonormal bases of $\mathbf{L}^2(\mathbf{R})$,

$$\phi_{j,k}(x) = 2^{-j/2}\phi(2^{-j}x - k); \quad j,k \in \mathbf{Z}. \tag{B.3}$$

The scaling function $\phi(x)$ satisfies,

$$\int_{-\infty}^{+\infty} \phi(x)dx = 1. \tag{B.4}$$

and the "two-scale difference" equation,

$$\phi(x) = \sqrt{2}\sum_{k=0}^{L-1} h_k \phi(2x-k). \tag{B.5}$$

In equations (B.2) and (B.5), two coefficient sets $\{g_k\}$ and $\{h_k\}$ have the same finite length L for a certain basis, where L is related to the number of vanishing mo-

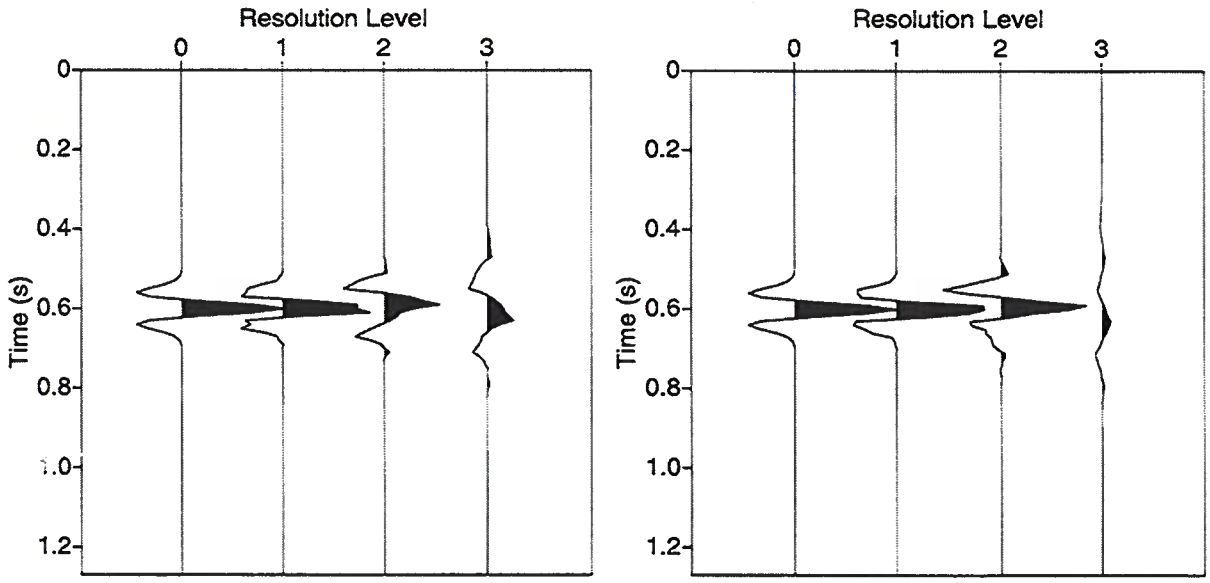

FIG. B.1. Decomposition of a Ricker wavelet at increasingly more coarse resolution levels. The bases of the decompositions are Daubechies wavelets with two and three vanishing moments for the left and right figure, respectively. The first traces shows the signal at the finest level, that is the original signal.

ments M in $\psi(x)$. For example, $L = 2M$ in the Daubechies wavelets. In the wavelet representation of signals, $\{h_k\}_{k=0,...,L-1}$ behaves as a low-pass filter and $\{g_k\}_{k=0,...,L-1}$ behaves as a high-pass filter to signals. These two filters are related by

$$g_k = (-1)^k h_{L-k}; \quad k = 0, ..., L-1, \tag{B.6}$$

and are called *quadrature mirror filters* (QMF). An extensive study of the QMF can be found in (Monzon, 1994).

Figure B.1 shows the decomposition of a simple synthetic seismic trace at various resolution levels for two different wavelet functions. The original trace is a *Ricker wavelet*, *i.e.*, a normalized second-order derivative of a Gaussian function, with a peak frequency of 30 Hz. The left figure shows the decomposition by a Daubechies orthonormal basis with two vanishing moments, while the right figure shows the same decomposition with three vanishing moments. The Ricker wavelet (f) is the leftmost trace in each box, while the remaining traces correspond to f^j of equation (4.13), where $j = 1, 2, 3$, respectively. From Figure B.1, it can be seen that the decomposed traces contains progressively lower frequencies with the increase of decomposition levels. Comparing the two plots in Figure B.1, we also observe that the increasing the number of vanishing moments increases the smoothness of the decomposed signal.

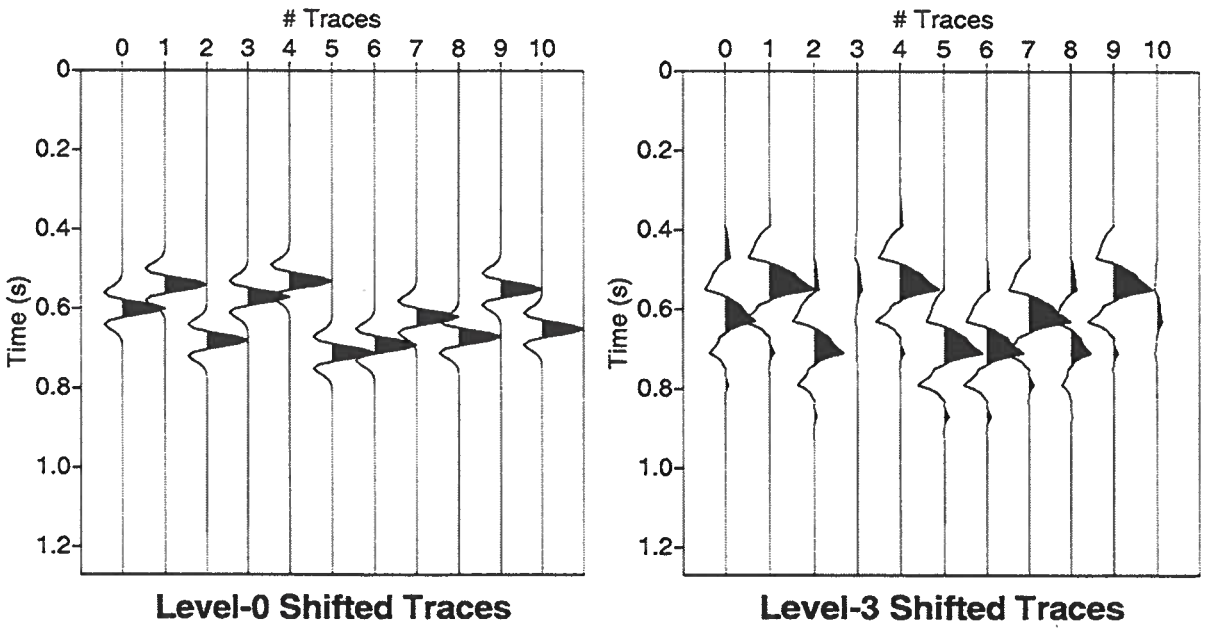

FIG. B.2. Ten traces of randomly shifted Ricker-wavelet traces (left) and their decomposition at resolution level 3 in the Daubechies wavelet bases with two vanishing moments (right).

A Symmetric and Shift-Invariant Wavelet Basis In many applications, it is required that the processes applied to the obtained signals be shift-invariant. For example, in examining the multi-scale property in residual-statics correction problems, it is important that the error-fitting function at each scale have a common — or at least close to common — global minimum. Therefore, we expect that the relative time-shifts among traces at each scale to be almost the same as it was in the original data, and that the waveforms are not deformed from one trace to another. However, the orthonormal wavelet basis representations are generally not shift-invariant. This shift-variance of the orthonormal, compactly-supported wavelet an be seen directly from the construction of their bases, equations (B.2) and (B.5), because of the change of step sizes among different scales in these definitions. Therefore, the Daubechies wavelet bases are not suitable for our purpose. Figure B.2 shows ten copies of randomly shifted Ricker-wavelet traces, and their projections onto the subspace V_3 in the Daubechies bases with two vanishing moments. The decomposed waveforms on the right of Figure B.2 are deformed to different shapes among traces with different time-shifts, and they do not have the same relative time shifts as those shown on the left of Figure B.2.

Saito and Beylkin (1993) suggested using the *shell* of an orthonormal basis when shift-invariance is required. Without loss of generality, let us assume that the signal

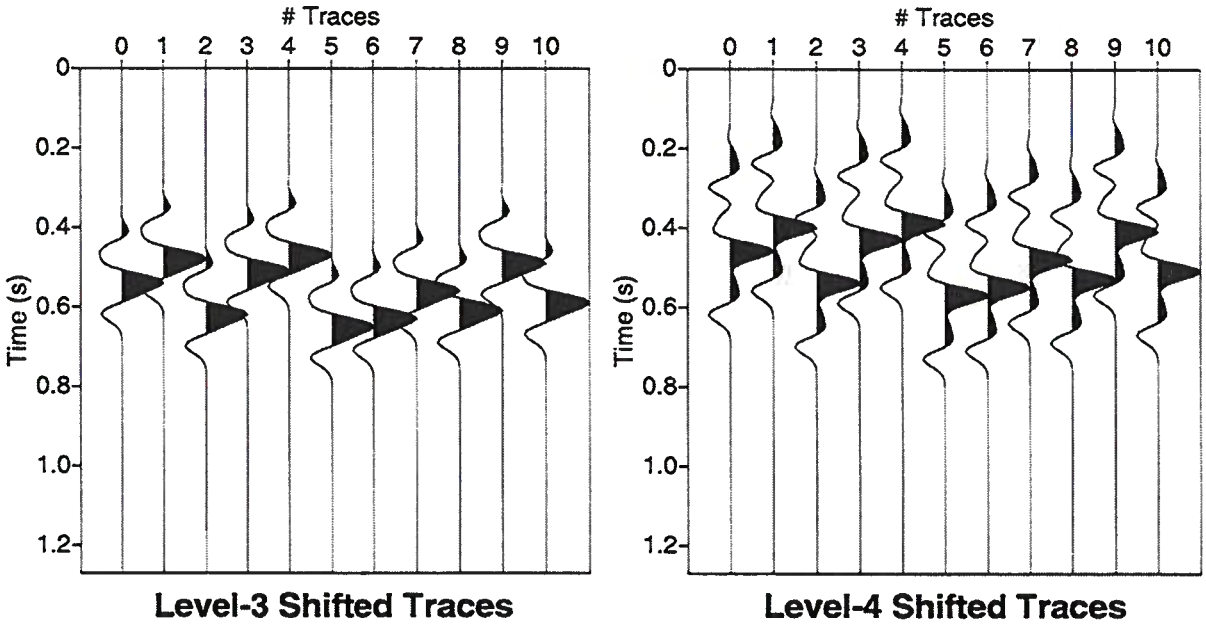

FIG. B.3. The decomposition of ten copies of randomly shifted Ricker-wavelet traces, in the **shell** of the Daubechies basis with two vanishing moments, at resolution levels 3 and 4. The original traces are shown on the left of Figure 3.

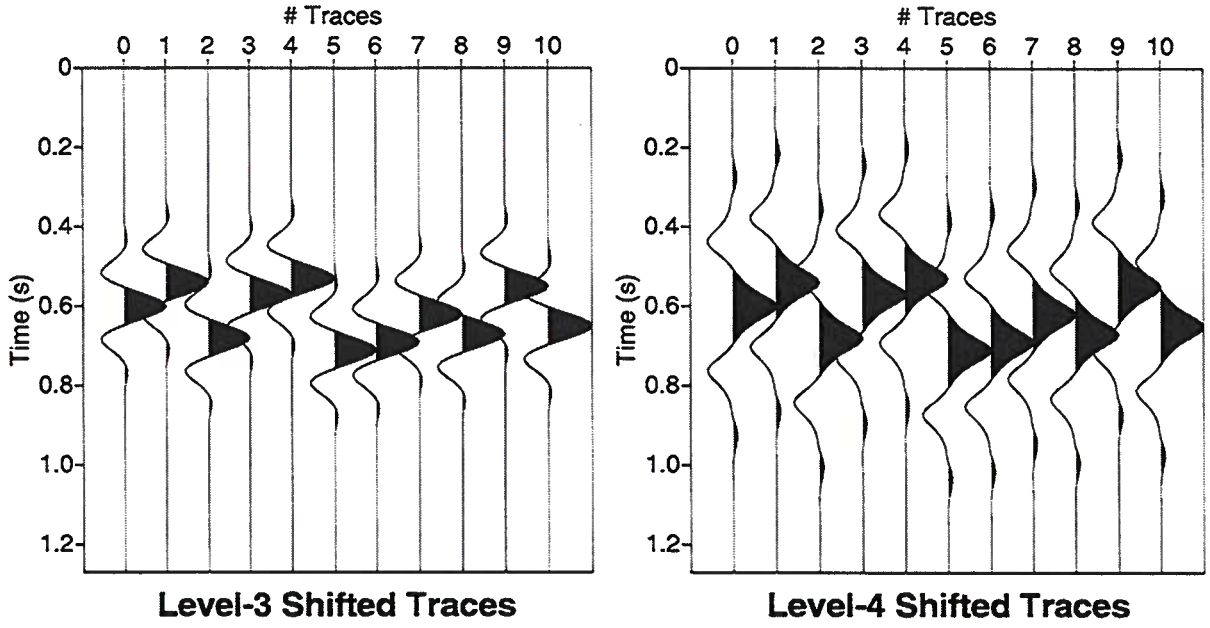

FIG. B.4. The decomposition of ten copies of randomly shifted Ricker-wavelet traces, in the **auto-correlation shell** of Daubechies basis with two vanishing moments, at resolution levels 3 and 4. The original traces are shown on the left of Figure 3.

we consider has finite length $N = 2^J$. Consider a family of functions

$$\{\tilde{\psi}_{j,k}(x)\}_{1\le j\le J,\, 0\le k\le N-1} \quad \text{and} \quad \{\tilde{\phi}_{j,k}(x)\}_{1\le j\le J,\, 0\le k\le N-1},$$

where

$$\tilde{\psi}_{j,k}(x) = 2^{-j/2}\psi(2^{-j}(x-k)), \tag{B.7}$$

$$\tilde{\phi}_{j,k}(x) = 2^{-j/2}\phi(2^{-j}(x-k)), \tag{B.8}$$

where the functions $\psi(x)$ and $\phi(x)$ are a wavelet and scaling function, respectively. The new family of functions defined by equations (B.7) and (B.8) can also serve as bases for subspaces V_j and W_j in MRA. They are complete, but they are redundant and not orthonormal (Saito, 1994). Therefore, the decomposition of a function in these bases is not unique. However, by forcing an additional constraint on the projection, a function $f \in V_0$ may still be decomposed in the shell of an orthonormal basis in much the same way as it was in an orthonormal wavelet basis itself. In this case, the basis functions in equations (4.13) and (4.14) are replaced by $\tilde{\psi}_{j,k}(x)$ and $\tilde{\phi}_{j,k}(x)$.

The representation of signals using this family of bases are shift-invariant among different scales. Figure B.3 shows the same numerical experiment as that in Figure B.2, except using the shell of orthonormal bases expansion at resolution levels 3 and 4. The relative time-shifts among traces are preserved while the waveforms are deformed to the same extent. However, the original symmetric waveforms are deformed to asymmetric waveforms. This deformation of the waveforms is not desirable, and may cause problems for some applications.

To overcome this problem, a family of symmetric, shift-invariant bases are introduced by Saito and Beylkin (1993). Let $\Phi(x)$ and $\Psi(x)$ be auto-correlation functions of the scaling function and wavelet function, respectively:

$$\Phi(x) = \int \phi(y)\, \phi(y-x)\, dy, \tag{B.9}$$

$$\Psi(x) = \int \psi(y)\, \psi(y-x)\, dy, \tag{B.10}$$

where ψ and ϕ satisfy equations (B.2) and (B.5) respectively. Construct a family of bases

$$\{\Psi_{j,k}(x)\}_{1\le j\le J,\, 0\le k\le N-1} \quad \text{and} \quad \{\Phi_{j,k}(x)\}_{1\le j\le J,\, 0\le k\le N-1},$$

where

$$\Phi_{j,k}(x) = 2^{-j/2}\Phi(2^{-j}(x-k)), \tag{B.11}$$

$$\Psi_{j,k}(x) = 2^{-j/2}\Psi(2^{-j}(x-k)). \tag{B.12}$$

Now, we have an *auto-correlation shell* of an orthonormal basis that is both symmetric and shift-invariant. Figure B.4 shows the expansion of shifted Ricker-wavelet traces in the auto-correlation shell of Daubechies basis. It can be seen that both the symmetry of the waveforms and the relative time-shifts are preserved at resolution levels 3 and

4.

There exists a fast algorithm for expanding a function $f \in V_0$ using the auto-correlation shell of orthonormal basis (Saito & Beylkin, 1993). I give only the formulas for the discrete expansion; detailed derivation can be found in (Saito & Beylkin, 1993).

Suppose that S_k^j and D_k^j are the projected signal onto the subspaces V_j and W_j at the sampled positions, respectively; that is

$$S_k^j = f^j(k\Delta), \quad D_k^j = Df^j(k\Delta),$$

where Δ is the sampling interval. Then, two symmetric filters, $P = \{p_k\}_{-L+1 \le k \le L-1}$ and $Q = \{q_k\}_{-L+1 \le k \le L-1}$, are applied recursively to the signal we wish to decompose,

$$\begin{aligned} S_k^j &= \sum_{l=-L+1}^{L-1} p_l \, S_{k+2^{j-1}l}^{j-1} \\ D_k^j &= \sum_{l=-L+1}^{L-1} q_l \, S_{k+2^{j-1}l}^{j-1}; \end{aligned} \tag{B.13}$$

where $0 \le k < N$, $1 \le j \le J$, and L is the filter length in the "two-scale difference" equations of wavelet and scaling functions as in equations (B.2) and (B.5). In equation (B.13), $N = 2^J$ is the number of samples of the signal, and the filter coefficients p_k and q_k are,

$$p_k = \begin{cases} 2^{-1/2}, & \text{for } k = 0, \\ 2^{-3/2} a_{|k|}, & \text{otherwise;} \end{cases} \tag{B.14}$$

and

$$q_k = \begin{cases} 2^{-1/2}, & \text{for } k = 0, \\ -p_k, & \text{otherwise.} \end{cases} \tag{B.15}$$

In equations (B.14) and (B.15), coefficients $\{a_k\}_{k=1,\ldots,L-1}$ are the correlation of the low-pass filter $\{h_l\}_{l=0,\ldots,L-1}$ in equation (B.5),

$$a_k = \begin{cases} 2\sum_{l=0}^{L-1-k} h_l \, h_{l+k}, & \text{for } k \text{ odd}, \\ 0, & \text{for } k \text{ even}. \end{cases} \tag{B.16}$$

Appendix C

SURFACE-CONSISTENT RESIDUAL-STATICS ESTIMATION

C.1 The Surface-Consistency Assumption

Statics are the time-shift approximations to the time-distortions in seismic-reflection data caused by heterogeneous material properties in the Earth's near surface. The goal of residual-statics estimation is to look for the time-shift of each trace by maximizing the alignment of the traces to be stacked together. For realistic residual-statics estimation problems, it is usually the case that the traces cannot be shifted independently.

It is mostly the case in reality that the near-surface heterogeneity is comprised of lower-velocity material than those in the deeper layers. Hence for most residual-statics problems, it is often assumed that waves are reflected from deep interfaces and are traveling through the near-surface with approximately vertical ray paths. Although not always true, it has been a good enough working assumption for many practical problems. Therefore, residual-statics are *surface consistent*; that is, statics of each trace are the additive time distortion of near-source and near-receiver heterogeneities (*source-statics* $\mathbf{s}$ and *receiver-statics* $\mathbf{r}$), regardless of the ray paths connected with it. Figure C.1 illustrates the similarity of travel paths near each source and each receiver.

The recorded reflection seismic signals are usually sorted into the common-midpoint (CMP) domain, where the two major axes are the *midpoint* y (of the source and receiver locations) and *half-offset* h (half distance between the source and receivers). The number of different offsets in a CMP section is usually referred to as the number of *folds* of this CMP section. Assuming that the time-shifts between the input traces for statics correction are caused only by the near-surface heterogeneity, the unknown parameters to be estimated would be $\mathbf{s}$ and $\mathbf{r}$. For this particular case, the stack-power function becomes

$$F(\mathbf{s},\mathbf{r}) = -\sum_{y}\sum_{h_1\neq h_2}\Phi^{y}_{h_1,h_2}(\tau(\mathbf{s},\mathbf{r})), \tag{C.1}$$

where y denotes the midpoint index, t is the time index over a specified window, and h is the offset index, and where $\Phi^{y}_{h_1,h_2}(\tau)$ is the cross-correlation between the normal-moveout-corrected traces (see page 54) of offsets h_1 and h_2 at midpoint y. When ignoring the non-corrected (residual) normal-moveout and time-shift caused by subsurface structure, function $\tau(\mathbf{s},\mathbf{r})$ is determined by the recording geometry,

$$\tau = s_{i(y,h_1)} + r_{j(y,h_1)} - s_{i(y,h_2)} - r_{j(y,h_2)},$$

and $i(y,h)$ and $j(y,h)$ are the source and receiver indices for midpoint y and offset h,

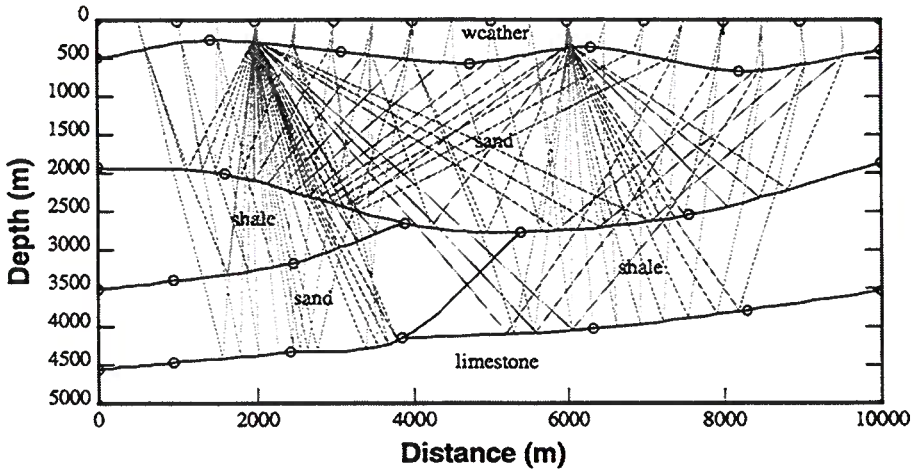

FIG. C.1. A hypothetical model of the Earth's upper crust showing rays associated with seismic waves propagating down from sources on the surface, reflecting off geologic boundaries and traveling upwards to receivers on the surface. Static-shifts of the seismic traces are caused by the combined time-distortions of near-source and near-receiver heterogeneities in the weathering layer.

respectively. Therefore, solving a residual-statics problem is a nonlinear optimization wherein we look for $\mathbf{s}, \mathbf{r}$ that

$$\min_{\mathbf{s},\mathbf{r}} F(\mathbf{s}, \mathbf{r}).$$

This is a non-linear approach developed by Rothman (1985; 1986).

In some cases, some factors other than statics can cause the misalignment of nearby traces. For example, we can include a *residual normal-movemout* term, when the correction for the propagation (normal moveout) is not perfect, and a *structural factor*, which is the zero-offset travel-time difference caused by the subsurface structure. When these are considered, τ_i should be a linear function of all these contributing factors (Taner *et al.*, 1974; Wiggins *et al.*, 1976).

C.2 Conventional Residual-Statics Approaches Revisited

From Observation 1 in Chapter 4, a stacking-power function could be separable after a linear transformation. Therefore, this high-dimensional optimization problem can be reduced to a set of one-dimensional optimization problems, followed by the solution of a set of linear system. Such a two-step approach can be described by the following algorithm.

Algorithm 6 (Linear Approach for Residual-Statics Estimation) *The goal of this algorithm is to find a set of parameters* $\mathbf{m}^*$, *such that the function* $F(\mathbf{m})$ *of*

equation (4.3) at this point is the global minimum in the model space, i.e.,

$$F(\mathbf{m}^*) \leq F(\mathbf{m}),$$

where $\mathbf{m}$ *is an arbitrary model in the model space.*

1. *Define a new set of variables,*

   $$T_{k(i,j)} = \tau_{i,j}(\mathbf{m}).$$

   After this linear transformation, equation (4.3) becomes,

   $$\tilde{F}(\mathbf{T}) = -\sum_i \sum_{j \neq i} \Phi_{ij}(T_{k(i,j)}), \tag{C.2}$$

   where k *is an index for each of correlations functions.*

2. *Independently, find global maxima* $T^*_{k(i,j)}$ *for each of the one-dimensional correlation functions* $\Phi(T_k)$ *i.e.,*

   $$\Phi(T_k^*) \geq \Phi(T_k) \quad \forall k.$$

   Then, the global minimum of function $\tilde{F}(\mathbf{T})$ *would be at* $\mathbf{T}^*$*, whose components are* T_k^**. That is,*

   $$\tilde{F}(\mathbf{T}^*) \leq \tilde{F}(\mathbf{T}).$$

3. *Solve a linear system of equations in unknown parameters* $\mathbf{m}$*,*

   $$\tau(\mathbf{m}) = \mathbf{T}. \tag{C.3}$$

   By solving the linear system equation (C.3), we find the global minimum of the stacking-power function equation (4.3).

The output of this algorithm is a set of parameters $\mathbf{m}^*$*, such that,*

$$F(\mathbf{m}^*) \leq F(\mathbf{m}) \quad \forall \mathbf{m} \in \mathcal{M}.$$

In Algorithm 6, we solve a high-dimensional optimization problem by a two-step approach: a series of one-dimensional optimization followed by solving a linear system. With the assumption of surface-consistency and taking into account the residual normal-moveout and the subsurface structure, the linear system in equation (C.3) is comprised of linear equations,

$$T_{k(i,j)} = s_i + r_j + g_k + m_k x_{ij}^2, \tag{C.4}$$

where s_i is the static time-shift associated with the ith source;
r_j is the static time-shift associated with the jth receiver;
g_k is the normal-incident travel-time from the datum plane to a surface reflector at the kth CMP gather, where $k = (i + j)/2$;
m_k is the time-averaged residual-moveout coefficient at the kth CMP;
x_{ij} is the known offset between the jth receiver and the ith source.
Equation (C.4) is a familiar equation used in conventional residual-statics estimation (Taner *et al.*, 1974; Wiggins *et al.*, 1976). Therefore, one special case of Algorithm 6 is, in fact, the conventional linear approach in statics, wherein the maximum correlation time is picked from each of the correlations. The time-picking procedure is, in effect, the procedure of one-dimensional optimization, at step 2 of Algorithm 6.

From the above analyses the decomposition of high-dimensional optimization to a set of one-dimensional problems is exactly. As long as the time-picking on correlation functions is correct, the statics results obtained from solving the linear system equation (C.3), is no worse than that of the non-linear approach of directly optimizing the corresponding high-dimensional stacking-power function.

Due to its computational efficiency, the conventional linear approach is the one most commonly used in industry for residual-statics estimation. However, this is overly optimistic about the seismic data in the sense that the only information used from the seismic data are the time-lags corresponding to the correlation peaks. Results of the statics estimation rely entirely on the time-picking $T_{k(ij)}$. When the seismic data are contaminated by severe noise and large statics, time-picking becomes difficult and ambiguous. If incorrect peaks are picked from the correlation function, (*i. e.*, cycle skips) resulting statics estimation would be incorrect and seismic events would be misaligned.

For problems with severe noise and large statics, Ronen and Claerbout (1985) proposed using equation (4.3) directly as the objective function for the optimization. Global searches, such as simulated annealing (SA) and genetic algorithms (GAs), are generally needed for searching the global optimization of stacking-power functions (Rothman, 1985; Rothman, 1986; Smith *et al.*, 1992). These global searches require intensive computation.

While oscillations in seismic data cause local extrema of stacking-power functions, they are not the only contributing factor, according the statement 3 of Observation 1. There exist local extrema on stacking-power functions that are not caused by the alignments of seismic traces. We are not interested in these *spurious minima* in residual-statics estimation. Therefore, as also pointed out by (Siegel, 1991), although direct optimization of the stacking-power functions uses full information in the seismic data, it suffers from the unnecessary complications of spurious local minima, and the computation cost is high due to the high complexity of the objective function.

C.3 An Algorithm of for Envelope Approach

C.3.1 The Algorithm

Ignoring residual normal-moveout and sub-surface structure, the mis-alignments of seismic traces within a CMP section are caused by only the source- and receiver-statics. Under such conditions, the objective function is the stacking-power in equation (C.1). As discussed in section C.3, this high-dimensional, highly multimodal objective function can be simplified by using the envelope data. The corresponding reduced stacking-power function becomes

$$\hat{F}(\mathbf{s},\mathbf{r}) = \sum_{y} \sum_{h_1 \neq h_2} \hat{\Phi}^{y}_{h_1,h_2}(\tau(\mathbf{s},\mathbf{r})), \tag{C.5}$$

where $\hat{\Phi}^{y}_{h_1,h_2}(\tau)$ is the envelope of correlation functions of $\Phi^{y}_{h_1,h_2}(\tau)$. To keep the smoothness, the cubic-convolution interpolation scheme is used throughout the project (Keys, 1981; Keys & Pann, 1993).

Now, we can design an algorithm for estimating residual statics for NMO-corrected, CMP data contaminated by surface-consistent statics as follows,

Algorithm 7 Residual-Statics Estimation using Reduced Stacking-Power

(a) *(Pre-compute and store:) For all cross-correlation $\Phi(\tau)$, find positive-peaks on all cross-correlation between traces for all CMP. Use the cubic convolution technique (Keys & Pann, 1993) to form envelope of the sequences $\hat{\Phi}(\tau)$.*

(b) *(Gross estimation of large-magnitude statics:) Local searches on the reduced stacking-power function $\hat{F}(\vec{s},\vec{r})$ for the initial model $\vec{s}_0,\vec{r}_0$; correct data for the estimated statics.*

(c) *(Refinement:) After large-magnitude statics are corrected in step (ii), results can be refined by optimizing the stacking-power function using local optimization with 0 initial guess, or a conventional linearized statics-estimation procedure. Depending upon results from step (b), this step may not be necessary.*

Where the reduced stacking-power function causes a loss of resolution, the last step of refinement is important for further improvement of the stacking quality.

The optimization method used Algorithm 7 is the same as that used in a conventional non-linear residual-statics approach. The only difference in Algorithm 7 is that the input data consist of the envelope functions, instead of the correlation functions themselves. Therefore, the implementation of Algorithm 7 can use existence software,

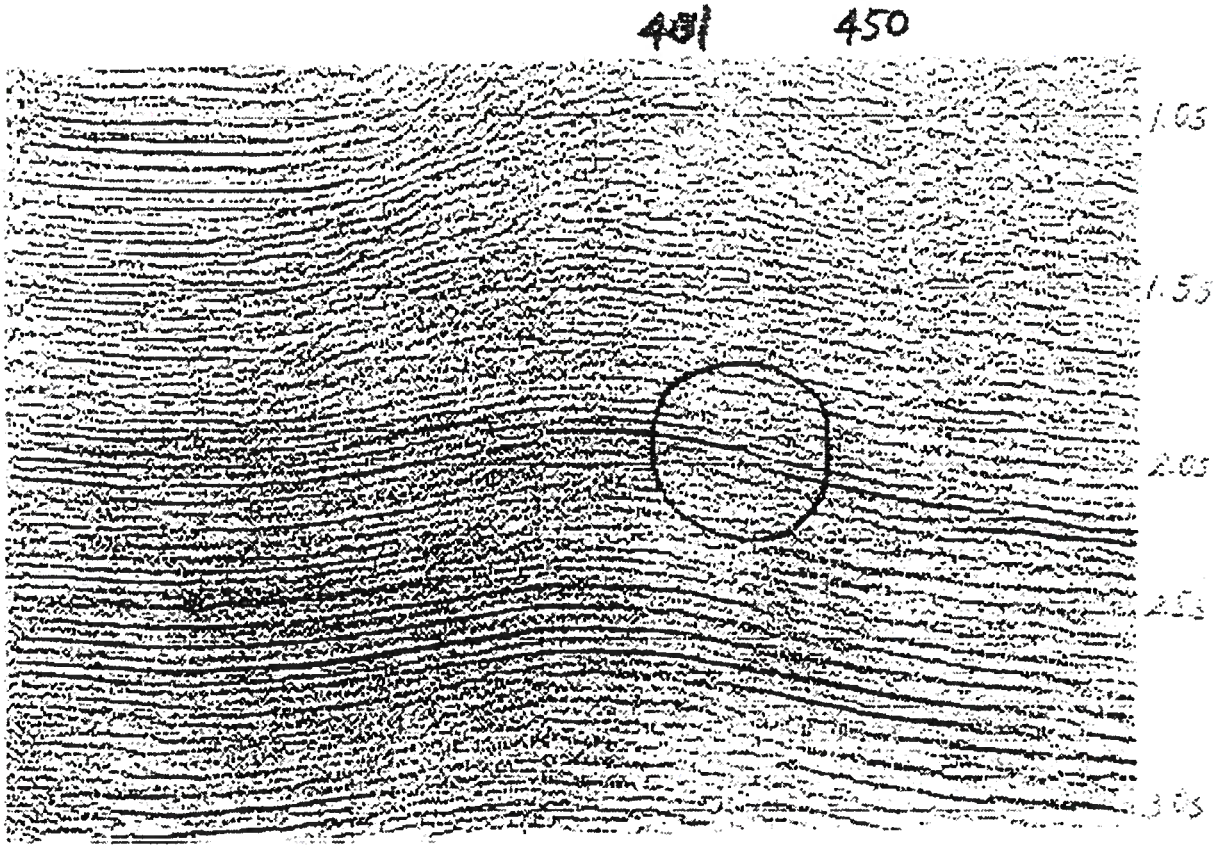

FIG. C.2. Alberta Foothills data: conventional statics-corrected stacked section. Cycle skipping has disrupted continuity in the circled area.

with only the addition of software for extracting envelope information from correlation data.

In Section 4.3.1, I showed an application of Algorithm 7 to a synthetic data set. The analysis show that this approach can indeed simplify the objective function, and only local-descent searches are needed in the implementation. The synthetic data-set also show that this approach works well for large static-shifts (about 100 ms) and is robust in the presence of random noise. Now, I show the application to two field data-set contaminated by large statics. Use of conventional methods has introduced cycle-skips for both data sets.

C.3.2 Application to Alberta Foothill Data

The Alberta Foothill data is contaminated by large statics which disrupt continuity of strong reflection events in stacked sections. Figure C.2 shows a stacked section, corrected with residual statics using the ProMax software. The result of cycle skipping can be observed at about CMP 420.

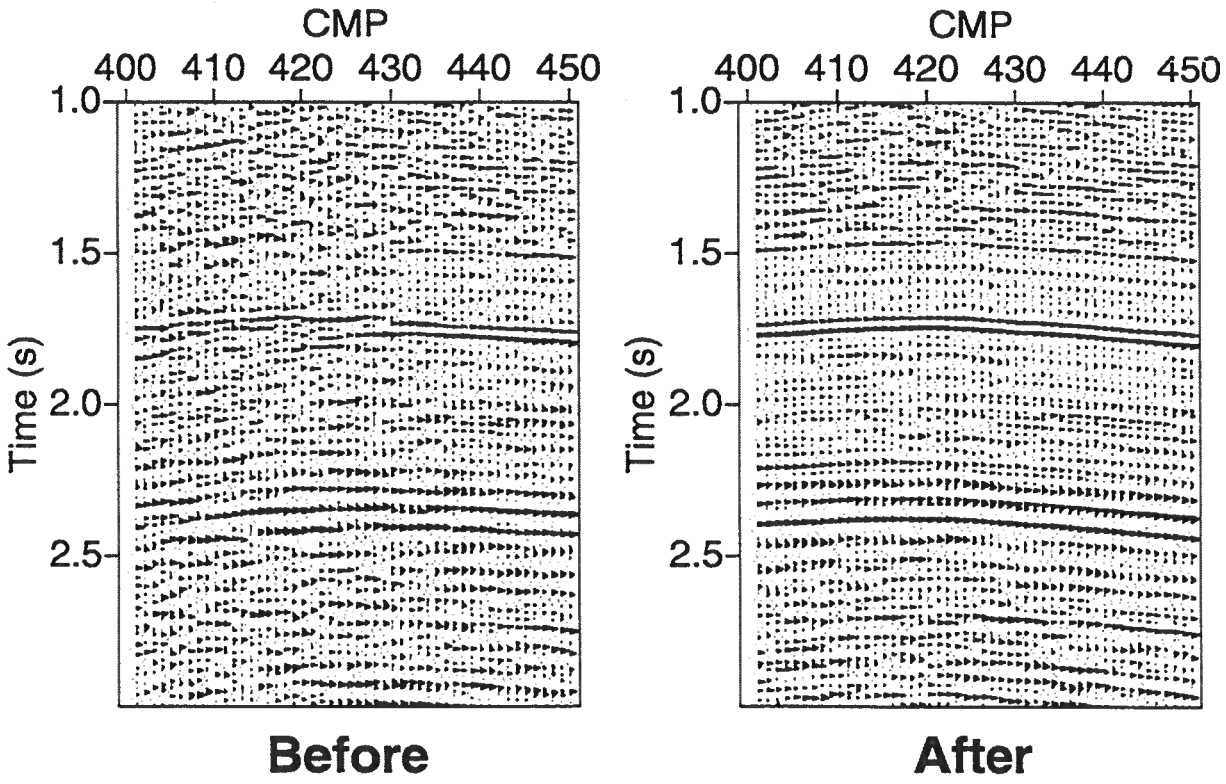

FIG. C.3. Left figure shows 50 stacked CMP gathers from the Alberta Foothills data set before residual statics are corrected. The right figure shows the same stacked section after residual statics are corrected using the envelope approach.

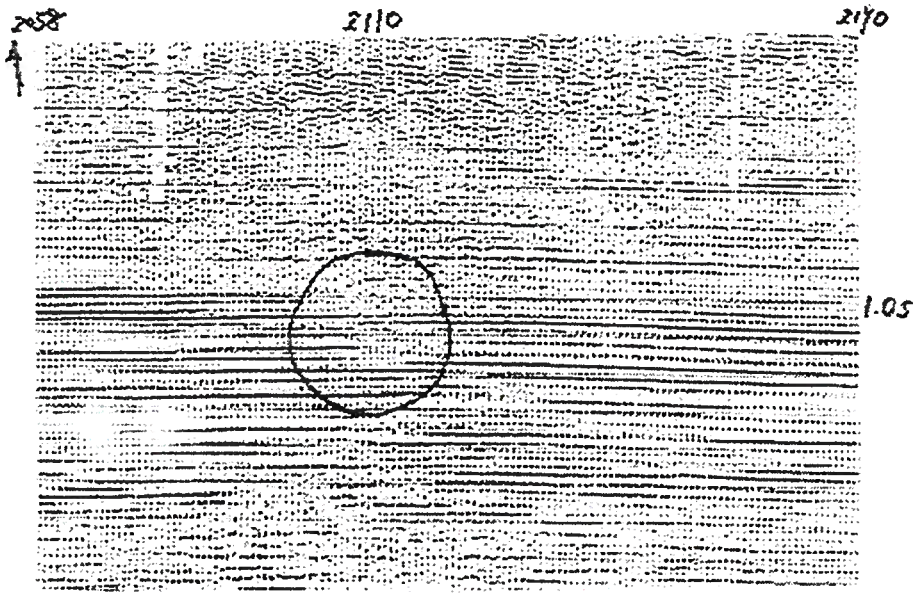

FIG. C.4. Paradox basin data: conventional statics-corrected stacked section. Cycle skipping has disrupted continuity in the circled area.

This data set has an average of 60 fold. I choose 50 CMP gathers from this dataset, which was obtained with 88 sources and 88 receivers. Therefore, this statics-estimation problem is an optimization of 176 dimensions. The stacked section for CMP 401-450 is shown on the left of Figure C.3. Discontinuity of events is severe in this section (see, for example, the reflections between 1.5 and 2.0 s). The statics-corrected stacked section using the proposed residual-statics correction algorithm is shown on the right of Figure C.3, which shows no discontinuity of reflection events.

The reflection events between 1.5 and 2.0 s shows different shapes across the section between Figures C.2 and C.3. This is caused by an additional elevation correction done internally by ProMax in Figure C.2. To solve this 176-dimensional global optimization problem, run time was around 48 minutes on an SGI Indy.

C.3.3 Application to Paradox Basin Data

I test Algorithm 7 on another data set. Figure C.4 shows a stacked section corrected with residual statics estimated by a conventional method in ProMax. This output shows a strong discontinuity at around CMP 2110, especially for the event of 0.8 s.

This data set has an average fold of 15. I processed 100 CMPs, with 35 sources and 146 receivers. This residual-statics estimation is a 181-dimensional global optimization problem. A stacked section containing 100 CMPs before the residual statics are corrected is shown on the left of Figure C.5. The right figure in C.5 shows the stacked section after residual statics are estimated and corrected using Algorithm 7.

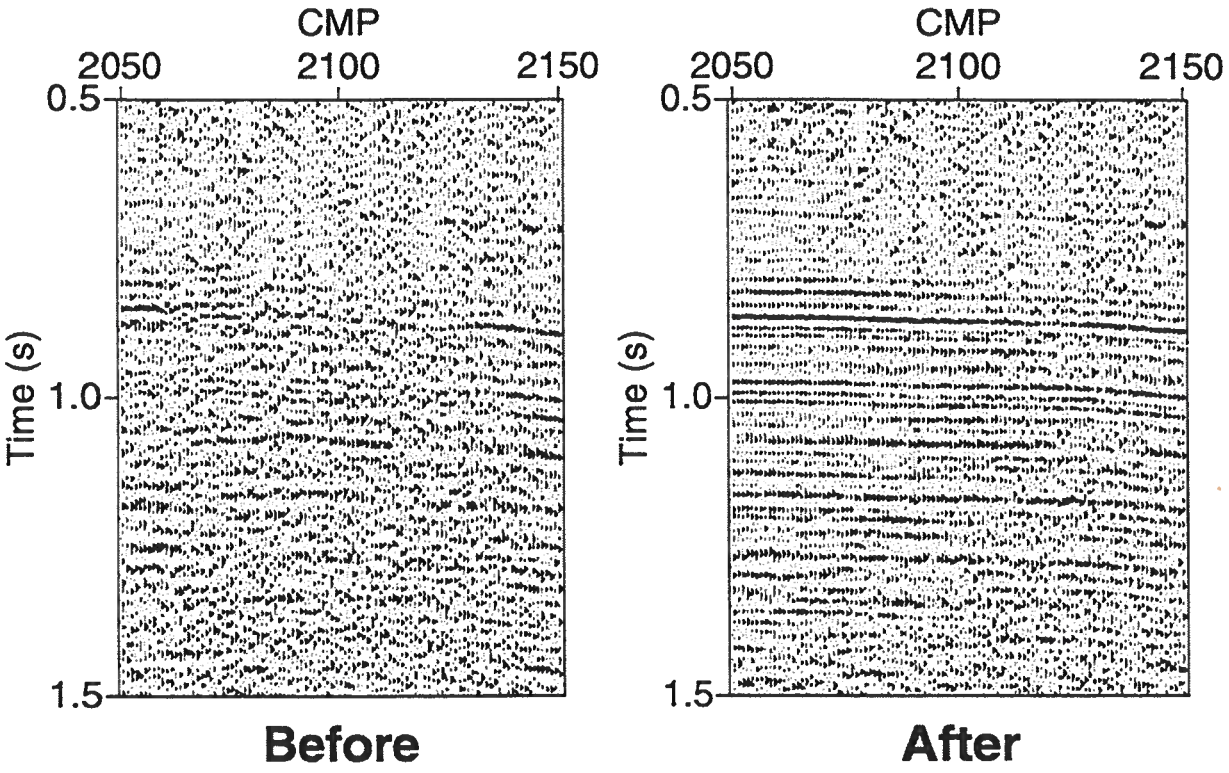

FIG. C.5. Stacked section of Paradox data before residual statics are corrected is shown on the left, and the stacked Paradox data after statics are estimated and corrected using the envelope approach is shown on the right.

The strong event at around 0.8 s is almost continuous across the section, and some stratigraphic details can be seen.

The continuity of the event at 0.8 s, however, is still not satisfactory around CMP 2110. This results from the loss of resolution in the reduced stacking-power function. A second iteration of optimizing the stacking-power function is needed for refinement. Figure C.6 shows the result of this second iteration using a non-linear conjugate-gradient search with 0 initial guess. The lateral continuity of the strong event at around 0.8 s has notably improved. Each iteration of the statics estimation for this data set is about 25 minutes on an SGI Indy.

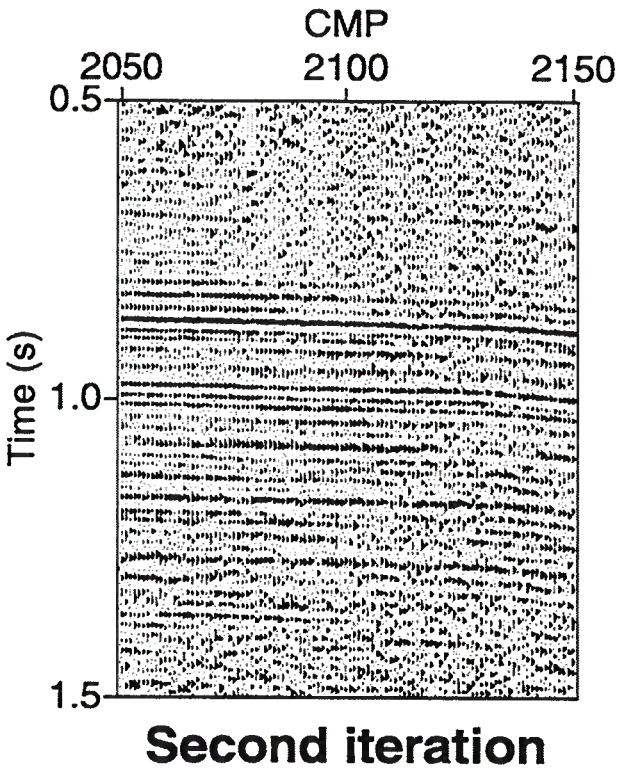

FIG. C.6. Stacked section of Paradox data after the second iteration of residual-statics correction. The statics are estimated by optimizing the stacking-power function formed from the right of Figure C.5 by a non-linear conjugate-gradient search with 0 initial guess. The continuity of events around 0.8 s has notably improved.