

# HOMOTOPY OPTIMIZATION METHODS FOR GLOBAL OPTIMIZATION

DANIEL M. DUNLAVY\* AND DIANNE P. O'LEARY†

November 11, 2005

**Abstract.** We define a new method for global optimization, the Homotopy Optimization Method (**HOM**). This method differs from previous homotopy and continuation methods in that its aim is to find a minimizer for each of a set of values of the homotopy parameter, rather than to follow a path of minimizers. We define a second method, called **HOPE**, by allowing **HOM** to follow an ensemble of points obtained by perturbation of previous ones. We relate this new method to standard methods such as simulated annealing and show under what circumstances it is superior. We present results of extensive numerical experiments demonstrating performance of **HOM** and **HOPE**.

**Key words.** homotopy method, stochastic search, multistart, global optimization

**AMS subject classifications.** 65K10, 90C26, 90C30

**1. Introduction.** We are interested in solving the minimization problem

$$\begin{aligned} \text{Given } f : \mathcal{X} \subseteq \mathbb{R}^n \rightarrow \mathbb{R}, \\ \text{find } x^* \in \mathcal{X} \text{ such that } f(x^*) = \min_{x \in \mathcal{X}} f(x). \end{aligned} \quad (1.1)$$

We assume that such a global minimizer  $x^*$  exists.

The class of methods that we develop in this work is related to homotopy methods (also referred to as continuation, deformation, or embedding methods), which have been effectively used for solving systems of nonlinear equations [3] and nonlinear optimization problems [45, 47]. These methods trace a path from the solution of an easy problem to the solution of the given problem by use of a homotopy—a continuous transformation from the easy problem to the given one.

We define the *Homotopy Optimization Method* (**HOM**), a homotopy method designed to solve optimization problems by use of a homotopy but not necessarily by path-tracing. We also introduce the method of *Homotopy Optimization with Perturbations and Ensembles* (**HOPE**). This method is related to previous methods for global optimization (see, for example, [11, 17, 19, 20, 29, 31, 32, 33, 42].) In particular, stochastic search methods perform a random search of the function domain either by independent sampling or stochastic perturbations of current iterates. The *Improving Hit-and-Run* (**IHR**) method [49], for example, moves to a stochastic perturbation of the current iterate if the resulting function value is lower. *Simulated annealing* (**SA**) methods (e.g., *basin hopping* [44], or *fast annealing* [21]) exploit the analogy defined in [23] between simulating the annealing process of a collection of atoms [27] and solving optimization problems. In each iteration of **SA**, a candidate

---

\*Sandia National Laboratories, Albuquerque, NM 87185, USA; dmdunla@sandia.gov. This work was partially supported by the National Library of Medicine (National Institutes of Health) through Bioinformatics Individual Fellowship F37LM008162, by the Applied Mathematics Research Program of the Office of Science at the United States Department of Energy, and by Sandia National Laboratories, a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under contract DE-AC04-94AL85000.

†Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742, USA; oleary@cs.umd.edu. This work was supported by the Department of Energy under Grant DEFG0204ER25655 and by the National Science Foundation under Grants CCR 02-04084 and CCF 05-14213.

point is generated, the point is tested based on an acceptance criterion dependent on a nonnegative parameter  $T$  (representing temperature in the annealing process), and the parameter  $T$  is updated according to a cooling schedule. An important feature common to point generation methods is the ability to generate candidate points far from the current iterate, even when  $T$  is close to zero [44]. SA methods converge to a global minimizer with probability one (i.e., almost surely) [26, 43]. However, convergence results found in the literature often assume extremely slow cooling schedules or place requirements on the move class (i.e., candidate generation process) that are prohibitive for practical implementation. SA is sometimes associated with *evolutionary algorithms*, a class that also includes genetic algorithms, genetic programming, evolutionary programming, and evolutionary strategies [4]. These methods start with an ensemble, or population, of points and use *mutation* and *selection* procedures to move closer to a global minimizer. *Smoothing methods* [1, 2, 36] are typically used for finding global minimizers of continuously differentiable functions. In general, these methods start by finding a global minimizer of a less complicated approximation of the original function and then deforming the function into the original function in a series of steps, where a minimizer of the deformed function at one step is found starting from the minimizer found at the previous step. *Global homotopy optimization methods* have been developed that find all local minimizers (or stationary points) of a function [10, 41]; however, due to the amount of computation required in these methods, they are typically only applicable to problems with a small number of local minimizers.

In this paper, we discuss our two new methods and their properties. After introducing HOM and HOPE in Section 2.1 and 2.2, analysis of the methods is given in Section 3, numerical results are presented in Section 4, and conclusions are found in Section 5.

**2. Our methods.** We begin with HOM, an extension of a local optimization method, and then show how to generalize it to HOPE.

### 2.1. Homotopy Optimization Method.

We define HOM by choosing

- a function,  $f^0 : \mathbb{R}^n \rightarrow \mathbb{R}$ , for which a *local minimizer*, denoted by  $x^0$ , exists and is either known or trivial to compute.
- a continuous homotopy function,  $h : \mathbb{R}^{n+1} \rightarrow \mathbb{R}$ , a function of the original variables,  $x \in \mathbb{R}^n$ , plus a homotopy variable,  $\lambda \in \mathbb{R}$ , such that

$$h(x, \lambda) = \begin{cases} f^0(x), & \text{if } \lambda = 0 \text{ and} \\ f^1(x), & \text{if } \lambda = 1, \end{cases} \quad (2.1)$$

where  $f^1(x) = f(x)$ .

- an integer  $m \geq 1$  and a sequence of positive values  $\Delta^{(k)}$ ,  $k = 0, \dots, m - 1$ , that sum to one.

Figure 2.1 presents the details of HOM.

Each point in the sequence produced in Step 5 is a local minimizer of  $h$  with respect to  $x$ . Specifically, the last point in the sequence,  $(x^1, 1) = (x^{(m)}, \lambda^{(m)})$  is a local minimizer of  $h(x, 1) = f^1(x)$ .

HOM differs from a standard continuation method for solving nonlinear equations only in Step 5; a continuation method would solve a nonlinear system of equations instead of minimizing a function, and the intent would be to choose the steplength  $\Delta^{(k-1)}$  small enough so that a path of zeros could be traced. A small steplength also ensures that only a few iterations of the method in Step 5 are necessary. See [3] for

- 
1. Input:  $x^{(0)} = x^0$ , a local minimizer of  $f^0$ ;  $m \geq 1$ ;  
 $\Delta^{(k)} > 0$  ( $k = 0, \dots, m - 1$ ) summing to 1.
  2. Initialize:  $\lambda^{(0)} = 0$ ;
  3. **for**  $k = 1, \dots, m$
  4.      $\lambda^{(k)} = \lambda^{(k-1)} + \Delta^{(k-1)}$
  5.     Use a local minimization method, starting with  $x^{(k-1)}$ , to compute  
an (approximate) solution  $x^{(k)}$  to  $\min_{x \in \mathcal{X}} h(x, \lambda^{(k)})$ .
  6. **end**
  7. Output:  $x^1 = x^{(m)}$
- 

FIG. 2.1. Definition of **HOM**.

more details on such methods, and **LOCA** [35], **HOMPACK** [46, 48], **AUTO** [12, 13, 14], and **CONTENT/MATCONT** [9, 25] for high quality implementations. Handling difficulties associated with paths ending, bifurcating, and turning back on themselves can make these methods quite complicated. We allow **HOM** to jump past such troubles to another path if necessary and avoid these difficulties.

In the end, though, **HOM** is only guaranteed to find a local minimizer of  $f^1$ .

**2.2. HOPE.** **HOPE** is an extension of **HOM** that increases the likelihood of finding the global minimizer of  $f^1$ . Whereas **HOM** generates a sequence of points converging to a single local minimizer of  $f^1$ , **HOPE** generates a sequence of ensembles of points where each member of the  $k$ th ensemble is a local minimizer of the homotopy function for  $\lambda = \lambda^{(k)}$ . This sequence converges to an ensemble of local minimizers of  $f(x)$  as  $\lambda \rightarrow 1$ .

In Step 5 of **HOM**, the next local minimizer in the sequence,  $x^{(k)}$ , is found using a local minimization method starting at the previous point in the sequence,  $x^{(k-1)}$ . In **HOPE**, the next ensemble of local minimizers is found using local minimization starting at the points in the previous ensemble in the sequence along with one or more *perturbations* of each of those points. A perturbation of  $x$  is denoted by  $\xi(x)$ , where  $\xi : \mathcal{X} \rightarrow \mathcal{X}$  is a function that stochastically perturbs one or more of the variables in  $x$ . In the end, **HOPE** produces an ensemble of local minimizers of  $f^1$ , from which we choose the one with the lowest function value as the best approximation to the true solution.

The number of points produced at each value of  $\lambda$  grows exponentially, so constraints on computational resources generally requires limiting the size of the ensemble, thus limiting the number of paths of local minimizers to be followed in the next and subsequent steps in the method. Pruning duplicate points from ensembles also helps in efficiently using computational resources.

Before presenting the details of **HOPE**, we first introduce some notation. Let  $c^{(k-1)}$  be the number of points in the ensemble at the beginning of iteration  $k$ , with  $c^{(0)} = 1$  (i.e., a single starting point is used). The  $j^{\text{th}}$  point in the ensemble at the start of iteration  $k$  is denoted by  $x_j^{(k-1)}$ . We use a secondary index to keep track of perturbations of points in the ensemble. Thus, at the end of iteration  $k$ ,  $x_{j,0}^{(k)}$  is the point found by minimization starting at  $x_j^{(k-1)}$ , and  $x_{j,i}^{(k)}$  is the point found by minimization starting at the  $i^{\text{th}}$  perturbation of  $x_j^{(k-1)}$ . **HOPE** also requires two more

---

```

1. Input:  $x_1^{(0)} = x^0$ , a local minimizer of  $f^0$ ;  $m \geq 1$ ;  $c_{max} \geq 1$ ;  $\hat{c} \geq 0$ ;
    $\Delta^{(k)} > 0$  ( $k = 0, \dots, m - 1$ ) summing to 1.
2. Initialize:  $\lambda^{(0)} = 0$ ;  $c^{(0)} = 1$ 
3. for  $k = 1, \dots, m$ 
4.    $\lambda^{(k)} = \lambda^{(k-1)} + \Delta^{(k-1)}$ 
5.   for  $j = 1, \dots, c^{(k-1)}$ 
6.     Use a local method, starting with  $x_j^{(k-1)}$ , to compute
         an (approximate) solution  $x_{j,i}^{(k)}$  to  $\min_{x \in \mathcal{X}} h(x, \lambda^{(k)})$ .
7.   for  $i = 1, \dots, \hat{c}$ 
8.     Use a local method, starting with  $\xi(x_j^{(k-1)})$ , to
         compute an (approximate) solution  $x_{j,i}^{(k)}$ 
         to  $\min_{x \in \mathcal{X}} h(x, \lambda^{(k)})$ .
9.   end
10.  end
11.  Order the distinct points among  $x_{j,i}^{(k)}$  ( $j = 1, \dots, c^{(k-1)}, i = 0, \dots, \hat{c}$ )
      from “best” to worst as  $x_1^{(k)}, x_2^{(k)}, \dots$ , and discard any
      beyond  $c_{max}$ . Let  $c^{(k)} \leq c_{max}$  be the number retained.
12. end
13. Output:  $x^1$ , the point with lowest function value among
      the points  $x_j^{(m)}, j = 1, \dots, c^{(m)}$ 

```

---

FIG. 2.2. *Definition of HOPE.*

input values than **HOM**:  $c_{max}$ , the maximum number of points in an ensemble, and  $\hat{c}$ , the number of perturbations generated for each point in the ensemble.

**HOPE** is presented in Figure 2.2, where the overall structure of **HOM** is retained. (Note that Steps 7–9 should be skipped if  $\hat{c} = 0$ .) The differences between **HOPE** and **HOM** occur in Steps 5–11. Minimizing from a single starting point is replaced by minimizing from each of an ensemble of points and their perturbations, followed by the determination of the ensemble to be used in the next iteration. **HOPE** thus requires more work than **HOM** at each value of  $\lambda$ , making several calls to a local minimization method.

In Step 11 of **HOPE**, the ensemble of local minimizers to be used in the next iteration is determined. With no limit on the size of the ensemble and no pruning of duplicate points, the number of points in the ensemble at the end of iteration  $k$  would be  $(\hat{c}+1)^k$ . If the number of distinct local minimizers found at the current iteration is no more than the maximum ensemble size  $c_{max}$ , then all are used in the next iteration; otherwise we must choose the “best” subset. What constitutes best may differ with the specific problem or application area to which **HOPE** is applied and may depend on the iteration number ( $k$ ), the values of the algorithm parameters ( $m$ ,  $c_{max}$ , and  $\hat{c}$ ), or the choice of local minimization method. An obvious measure of what constitutes the best conformations—the one used in our numerical experiments—is homotopy function value: conformations with the lowest function values are considered the best. However, there may be other suitable (or perhaps even better) measures depending on the choices for  $f^0$  and  $h$ ; for example, when the minimizers of  $f^0$  and  $f^1$  are related

geometrically or when the homotopy function has been designed to deform  $f^0$  into  $f^1$  in a particular manner.

Note that different parameter choices for  $m$  and  $\hat{c}$  reduce **HOPE** to **HOM** ( $\hat{c} = 0$ ), to the local minimization method used in Step 6 ( $\hat{c} = 0, m = 1$ ), or to a stochastic search method ( $\hat{c} > 0, m = 1$ ). We thus view **HOPE** as a framework for generalizing a local method for use in solving global optimization problems.

**3. Analysis of HOPE and HOM.** We begin the analysis by showing that with proper choices for the homotopy function, perturbation function, and algorithm parameters, **HOPE** is equivalent to other methods for which convergence results exist. We then derive results giving insight on homotopy choices.

**3.1. Analysis by analogy with other methods.** In this section we present several theoretical results regarding the performance of **HOM** and **HOPE** in relation to other methods for solving optimization methods.

**HOM as a probability-one homotopy method.** The first result involves **HOM** applied to a convex function and follows from the analysis of probability-one homotopy methods [45]. In that work, if  $f^1 \in C^3(\mathcal{X})$  is convex on  $\mathcal{X} = \mathbb{R}^n$ , it was shown that for almost every point  $x^0$ , an equilibrium curve of  $\nabla_x h$ , with

$$h(x, \lambda) = (1 - \lambda) \frac{1}{2} (x - x^0)^T (x - x^0) + \lambda f^1(x), \quad (3.1)$$

exists, contains both  $x^0$  and  $x^*$ , has finite length, and contains only minimizers of  $h$ . Thus, **HOM** applied to solving this problem using a globally convergent local minimization method (e.g., Newton's method with a trust region) will produce a sequence of minimizers of  $h$ , converging to the unique minimizer  $(x^*, 1)$ , starting from almost any point  $x^0$ . Note that these results hold for any  $m \geq 1$ . Thus **HOM** converges to the global minimizer of  $f^1$  with probability one.

**HOPE as a stochastic search method.** **HOPE** is an Improving Hit-and-Run (**IHR**) method when  $c_{max} = 1, m = 1$ , the perturbation function is the Hit-and-Run (**HR**) algorithm [7, 39], homotopy function value is used as the measure to constitute the best ensemble points, and the number of iterations of the local minimization method used in **HOPE** is set to 0. Thus, following the analysis in [49], we conclude that **HOPE** converges in that case with probability one as  $\hat{c} \rightarrow \infty$ . For  $m > 1$ , the result holds as well. Moreover, for one class of functions (Lipschitz, elliptical), the number of points required to guarantee convergence is linear in the dimension of the problem [49].

We note that using similar assumptions, **HOPE** can be shown to be a Pure Random Search (**PRS**) method as well. The convergence of **HOPE** under those conditions can be shown, but the complexity results of **PRS** show that the number of points required for this convergence is exponential in the dimension of the problem. However, these results verify that convergence with probability one can be guaranteed for **HOPE** using perturbation functions other than **HR**.

**HOPE as a simulated annealing method.** We next discuss the conditions for which **HOPE** is an **SA** method. Theoretical convergence proofs exist for many variants of **SA** methods, and results of numerical experiments show that **SA** is an effective method for solving problems for which several standard minimization methods fail [26].

Define the perturbation function as

$$\xi_{T(\lambda)}(x) = \begin{cases} \delta x & \text{with probability } P_{T(\lambda)}(x, \delta x) \\ x & \text{with probability } 1 - P_{T(\lambda)}(x, \delta x) \end{cases} \quad (3.2)$$

where  $\delta x$  is a point sampled from a distribution in which every point in the domain has a positive probability of being sampled;  $P_T$  is the Metropolis acceptance criterion often used in SA methods; and  $T(\lambda)$  is a continuous function of  $\lambda$  defining the temperature such that  $T(0) = T^0$  and  $T(1) = 0$  (e.g.,  $T(\lambda) = T^0(1 - \lambda)$ ).

Combining the perturbation function,  $\xi_{T(\lambda)}$ , with the identity homotopy function  $h(x, \lambda) = f^1(x)$ , HOPE becomes an SA method. Thus, convergence results that apply to SA methods can be applied to HOPE as well. In [8] it was shown that by placing an upper bound on the rate of temperature decrease and allowing generation of points in the entire domain (or feasible region), an SA method converges to a global minimizer almost surely as  $m \rightarrow \infty$ . Here  $m$  is the number of iterations (steps in  $T$ ) taken.

**3.2. Insight on homotopy choice.** In this section we give analysis to explain how the definition of the homotopy function and choice of algorithm parameters determines the performance of HOPE.

**Influence of basins of attraction.** Our aim is to show that when  $f^1$  is well-behaved, then HOPE converges, and we can provide a bound on the number of steps in  $\lambda$  required for HOPE to converge with some given probability. We assume in this section that  $\hat{c} = 1$  and  $c_{max} = 1$  and that  $\xi$  generates points uniformly on a bounded domain  $\mathcal{X}$ .

We define the *basin of attraction*  $\mathcal{B}(\lambda)$  of a local minimizer of  $h(x, \lambda)$  to be the set of points  $x$  such that the local minimization method, started at  $x$ , will converge to the minimizer. (Note that this definition depends on the choice of the local minimization method.) Assume that  $x^*$  is the unique global minimizer of  $f^1$ , and that it is an isolated minimizer. Then the Implicit Function Theorem guarantees that in some neighborhood of  $(x^*, 1)$ , there exists a unique curve of isolated minimizers that passes through  $(x^*, 1)$ . Furthermore, since the global minimizer of  $f^1$  is unique, then there is some value,  $\lambda_*$ , such that for all  $\lambda \in [\lambda_*, 1]$ , these isolated minimizers are global minimizers of  $h(x, \lambda)$ . Let  $\mathcal{B}^*(\lambda)$  be the basin of attraction for the global minimizer at  $\lambda$ . Assume  $V(\mathcal{B}^*(\lambda)) > 0$ , where  $V(\cdot)$  denotes the volume, normalized so that  $V(\mathcal{X}) = 1$ .

In Figure 3.1, we show an example homotopy function in one dimension where  $\mathcal{X} = [x_l, x_u]$ . The solid curves are curves of local minimizers of  $h$  and the dashed curves are the boundaries of the basins of attraction of the minimizers of  $h$ . The shaded region represents  $\mathcal{B}^*$ , the basin of attraction of the global minimizers of  $h$  for all  $\lambda \in [\lambda_*, 1]$ , and the solid curve in that region is the curve of global minimizers.

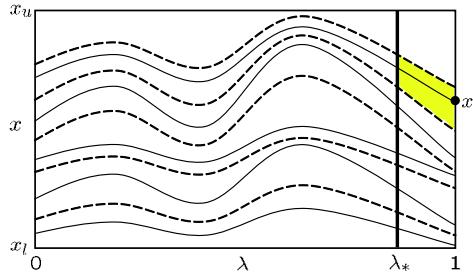


FIG. 3.1. Depiction of the point  $\lambda_*$  for a homotopy function in one dimension. The solid lines represent the curves of local minimizers of  $h$  and the dashed lines represent the boundaries of the basins of attraction associated with those curves. The shaded region is  $\mathcal{B}^*$ .

**THEOREM 3.1.** Assume that HOPE generates uniformly distributed  $\xi$  and that  $\hat{c} = 1$  and  $c_{max} = 1$ . Let  $k$  be the smallest integer so that  $\lambda^{(k)} \geq \lambda_*$ , and assume that the steps  $\Delta^{(j)}$  are small enough so that if  $(x^{(j)}, \lambda^{(j)}) \in \mathcal{B}^*(\lambda^{(j)})$  then  $(x^{(j+1)}, \lambda^{(j+1)}) \in \mathcal{B}^*(\lambda^{(j+1)})$  for  $k-1 \leq j < m$ . Then either  $x^{(k-1)} \in \mathcal{B}^*(\lambda^{(k)})$  and HOPE converges to the global minimizer of  $f^1$ , or the probability that HOPE converges to the global minimizer of  $f^1(x)$ , i.e., the probability that at least one point generated by HOPE for  $\lambda \in [\lambda_*, 1]$  is in  $\mathcal{B}^*$ , is given by

$$p_m = 1 - (1 - V_k)(1 - V_{k+1}) \dots (1 - V_m),$$

with  $V_k = V(\mathcal{B}^*(\lambda^{(k)}))$ .

*Proof.* Let  $p_j$  be the probability of being in the basin of attraction for the global minimizer when  $\lambda = \lambda^{(j)}$ . The desired result follows, by an induction argument, from the relation

$$p_j = p_{j-1} + (1 - p_{j-1})V_j = (1 - V_j)p_{j-1} + V_j.$$

□

Note that if  $V^*$  is a known lower bound on the volumes  $V_k$ , then we have a computable lower bound  $1 - (1 - V^*)^{m-k+1}$  on the probability of success.

This theorem gives insight into desirable properties for homotopy functions in HOPE. First, we want the basin of attraction for the global minimizer  $x^*$  to contain global minimizers for as large an interval of  $\lambda$  as possible, so that the desired point will not be pruned. Second, we want the average volume  $V_j$  to be larger than  $V_m$  for  $j \geq k$ ; otherwise, simulated annealing or stochastic search would do as well.

**Influence of  $m$  and the perturbation function.** We assign to the basin of attraction of each local minimizer a state  $S_\ell$  in a Markov chain, and order so that  $S_1$  corresponds to the global minimizer. Our stochastic perturbation sets up a branching random walk [5]. Assume that the steplengths  $\Delta^{(k)}$  are small enough so that a point in basin  $\ell$  at step  $j$  is still in basin  $\ell$  at step  $j+1$  for  $0 \leq j < m$ .

Let the  $(i, \ell)$  element of the transition matrix  $P(\lambda)$  be defined by

$$P_{i,\ell}(\lambda) = \Pr[\xi_\lambda(x) \in S_\ell : x \in S_i], \quad (3.3)$$

and let  $e_i$  be the  $i$ th unit vector. Consider the HOPE algorithm started at the local minimizer of  $h(x, 0)$  in  $S_i$ . After  $\ell$  iterations, we potentially have  $(\hat{c}+1)^\ell$  points in our ensemble, each with a different probability of being in each state. For example, counting duplicates, after 1 step we have two points, with probabilities  $e_i^T e_1$  and  $e_i^T P(\lambda^{(1)})e_1$ . After two steps, the probabilities for these two points are unchanged but we have two new points with probabilities  $e_i^T P(\lambda^{(2)})e_1$  and  $e_i^T P(\lambda^{(1)})P(\lambda^{(2)})e_1$ . In the next step we add points with associated probabilities  $e_i^T P(\lambda^{(3)})e_1$ ,  $e_i^T P(\lambda^{(1)})P(\lambda^{(3)})e_1$ ,  $e_i^T P(\lambda^{(2)})P(\lambda^{(3)})e_1$ , and  $e_i^T P(\lambda^{(1)})P(\lambda^{(2)})P(\lambda^{(3)})e_1$ . The pattern is now clear. For equal steplengths  $\Delta^{(k)} = 1/m$ , for example, the goal is to choose the smallest  $m$  so that the expected number of points is at least 1. Thus, a good method will have  $m$  large enough that sufficient points are generated, with the perturbation function chosen so that transitions into the state corresponding to the global minimizer are likely.

Certain special cases in which the matrix  $P(\lambda)$  has simple structure have been analyzed in [15].

**4. Numerical results.** We apply HOPE and HOM to several standard test problems found in the unconstrained optimization literature and to one new problem. We compare our methods to a quasi-Newton method for local minimization, and to a stochastic search method for global minimization. We highlight some advantages of using HOPE and HOM over other methods and show that HOPE is an effective method for solving general unconstrained minimization problems. Results showing HOPE outperforming simulated annealing on simple protein structure prediction problems can be found in [15, 16].

In our experiments, local minimization in HOPE and HOM was performed by a quasi-Newton method with cubic line search and BFGS update, which we denote **QNewton**. Note that **QNewton** requires only first derivative information.

Experiments using **QNewton** by itself provide a benchmark to which we compare the results using HOPE and HOM. **QNewton** terminates when the change in function values between iterates drops below **TolFun**, the maximum change in any of the variables in  $x$  between iterates drops below **TolX**, the number of iterates reaches **MaxIter**, or the number of function evaluations reaches **MaxFunEval**. The values used in our experiments are **TolFun** =  $10^{-6}$ , **TolX** =  $10^{-12}$ , **MaxIter** = 400, and **MaxFunEval** = 800. When **QNewton** is used for minimization in HOPE and HOM, **MaxIter** is reduced, as specified below, since multiple calls to **QNewton** are made.

We used equal steplengths  $\Delta^{(k)} = 1/m$  and the convex homotopy

$$h(x, \lambda) = (1 - \lambda)f^0(x) + \lambda f^1(x). \quad (4.1)$$

Note that when  $m = 1$ , HOPE is just a stochastic search method, and we include this in our experiments.

Two types of perturbations were used in testing HOPE. The first, denoted by  $\xi_{hr}$ , uses the **HR** method with a uniform distribution of perturbation lengths between 0 and a fixed maximum perturbation length  $p_{max}$ . The other, denoted by  $\xi_{pct}$ , is a variant of **HR** where the maximum perturbation length is a percentage of  $\|x\|_2$ , where  $x$  is the point being perturbed. The percentage is fixed throughout each run of an experiment and is denoted by  $\tilde{p}_{max}$ .

All of the experiments were run under Linux on a 2.5 GHz Intel Pentium 4 processor using Matlab 6.5 and the Optimization Toolbox 2.2 from Mathworks, Inc. In Matlab, **QNewton** is implemented in the routine **fminusub** and is accessed from the unconstrained minimization driver, **fminunc**.

**4.1. Comparison of methods applied to standard test problems.** The problems in this section are a subset of the test functions in [30] and consist of sums of squares of  $m_s$  functions of  $n$  variables. Using the starting points reported in that paper, **QNewton** converged to the correct solution for all but 5 of the 35 test problems:

- Freudenstein and Roth Function (**Freu**) [18]  $n = 2, m_s = 2$ .
- Jennrich and Sampson Function (**Jenn**) [22]  $n = 2, m_s = 10$ .
- Meyer Function (**Mey**) [28]  $n = 3, m_s = 16$ .
- Biggs EXP6 Function (**Be6**) [6]  $n = 6, m_s = 13$ .
- Trigonometric Function (**Trig**) [40]  $n = 10, m_s = 10$ .

The homotopy function for HOPE and HOM used

$$f^0(x) = \frac{1}{2}(x - x^0)^T(x - x^0), \quad (4.2)$$

using the starting points as reported in [30]. This generic function is often associated with probability-one homotopy methods used to solve optimization problems [45, 47].

|      | $f^1(x^*)$ | QNewton<br>$f^1(x^1)$ | HOM                  |     | HOPE       |             |
|------|------------|-----------------------|----------------------|-----|------------|-------------|
|      |            |                       | $f^1(x^1)$           | $m$ | $f^1(x^1)$ | $m$         |
| Freu | 0          | 48.98                 | 48.98                | —   | 0          | 9           |
| Jenn | 124.36     | 2020                  | 124.36               | 2   | 124.36     | 2           |
| Mey  | 87.95      | $5.0 \times 10^8$     | 146.18               | 81  | 87.95      | $3^\dagger$ |
| Be6  | 0          | $5.7 \times 10^{-5}$  | $5.7 \times 10^{-5}$ | —   | $10^{-14}$ | 1           |
| Trig | 0          | $2.8 \times 10^{-5}$  | $2.8 \times 10^{-5}$ | —   | $10^{-14}$ | 5           |

$^\dagger$ For  $1 \leq m \leq 10$ , there was only a single successful run, and it was when  $m = 3$ .

TABLE 4.1

Results of QNewton, HOM, and HOPE applied to Moré test functions. The lowest function values found and the fewest number of steps in  $\lambda$  (HOM and HOPE) to find the corresponding minimizer are presented.

Table 4.1 presents results of minimizing these functions using QNewton, HOM, and HOPE. In HOM and HOPE, QNewton was used for local minimization with MaxIter = 20. The second and third columns contrast the global minimum with the function value at the point  $x^1$  found by QNewton. The points produced by QNewton for Be6 and Trig are documented local minima [24] with function values close to the global minimum.

For the experiments using HOM, values of  $m = 1, \dots, 100$  were used. Only marginal improvement was achieved as the number of steps is increased, despite a large increase in computational cost. Columns 4 and 5 in the table show the lowest function value generated using HOM and the value of  $m$  at which that value was attained (with “—” signifying that no improvement was made in using HOM over QNewton for any value of  $m = 1, \dots, 100$ ).

HOM correctly predicted only one global minimizer (Jenn) and found only one other local minimizer (Mey) with a significantly lower function value than the one found using QNewton. In the latter case, the improvement was dramatic, as measured by relative error in the function value

$$r_{f^1} = \frac{|f^1(x^1) - f^1(x^*)|}{|f^1(x^*)|}. \quad (4.3)$$

HOM achieved  $r_{f^1} \approx 6.62 \times 10^{-1}$  versus  $r_{f^1} \approx 5.73 \times 10^7$  for QNewton. However, this improvement came at the cost of  $m = 81$  steps in the homotopy parameter and a correspondingly large number of function evaluations: a total of 7675 function evaluations for HOM versus 45 for QNewton. In general, such an increase in the number of function evaluations may not produce as significant an improvement.

The results of experiments using HOPE to minimize these functions are presented in the remaining columns of the table, using  $\hat{c} = 1$  and  $c_{max} = 2^m$ , with perturbations generated using  $\xi_{hr}$  with  $p_{max} = 10^{-3}$  (i.e., very local perturbations). The column labeled  $f^1(x^1)$  shows the lowest function value attained in the experiments using HOPE, where 10 runs were performed for each of the values  $m = 1, \dots, 10$ . (Note that when  $m = 1$ , HOPE is a stochastic search method.) Since no points were discarded from the ensembles in HOPE, the runs at  $m = 10$  require a considerable amount of computation compared to those for HOM and QNewton. The last column shows the lowest value of  $m$  ( $1 \leq m \leq 10$ ) for which all 10 runs were successful (i.e., 100% success rate). The best results were for Be6 and Jenn, where only 1 and 2 steps in  $\lambda$ , respectively, were required. Compared to the results for these problems where one fewer step in  $\lambda$  was taken, the increases in success rates were dramatic (0% success at  $m = 0$  for Be6 and

30% at  $m = 1$  for **Jenn**). This suggests that for these problems the perturbations were most responsible for the success of **HOPE**. For **Trig** and **Freu**, more steps in  $\lambda$  ( $m = 5$  and  $m = 9$ , respectively) were required before achieving a perfect set of runs. Furthermore, the increases in success rates were more gradual for these two problems (80% successes for **Trig** at  $m = 4$  and for **Freu** at  $m = 8$ ), suggesting that the performance of **HOPE** depends on more than perturbations alone.

For **Mey**, **HOPE** had difficulty for all values  $m \leq 10$ . The only successful result is shown in the table, where the global minimizer was found in 1 of the 10 runs for  $m = 3$  steps in  $\lambda$ . For **Mey**, there are several orders of magnitude difference in  $x_1^*$  and  $x_2^*$ , the first two elements of the global minimizer. The amount of perturbation ( $p_{max} = 10^{-3}$ ) used in these experiments was not enough to lead **HOPE** to success for  $m \leq 10$ . In followup experiments using larger perturbations ( $p_{max} = 100$ ), though, **HOPE** was successful in finding the global minimizer of **Mey** in 100% of the runs for each  $m = 1, \dots, 10$ .

We conclude that for these problems **HOPE** outperformed **QNewton** and **HOM** in terms of successfully finding the global minimizer of a function. Moreover, the results suggest that a small amount of perturbation can dramatically increase the performance of **HOPE** over **HOM**.

**4.2. Influence of parameter choice.** In this experiment we see that when we limit the ensemble size, a larger amount of perturbation may be required to produce comparable results. We applied **HOPE** to the **Freu** problem using various amounts of perturbation, maximum ensemble sizes, and numbers of steps in  $\lambda$  to illustrate the impact of the algorithm parameters on performance. In **HOPE**, **QNewton** was used with **MaxIter** = 60. Perturbations of ensemble members were generated using  $\xi_{hr}$  and  $\hat{c} = 1$ . A total of 100 runs were performed for each combination of  $p_{max} = 1, 2, 4, 8$ ,  $c_{max} = 2, 4, 8, 16$ , and  $m = 1, 2, 4, 8$ .

Table 4.2 presents the results of this experiment. The first two columns show the amount of perturbation and the maximum ensemble size. The next four columns show the number of runs where **HOPE** correctly predicted the global minimizer for  $m = 1, 2, 4, 8$ , respectively. Denoting  $N_f$  as the total number of function evaluations performed over the 100 runs, the last four columns show the number of function evaluations per success for  $m = 1, 2, 4, 8$ , respectively.

The general trend of these results show that as the amount of perturbation and the amount of computational effort increases (as controlled by  $c_{max}$  and  $m$ ) the chances of correctly predicting the global minimizer increases as well. However, the amount of perturbation appears to be the most important parameter affecting the success of **HOPE**, where a trend of increasing success is coupled with a trend of decreasing number of function evaluations per successful run. This was expected for the **Freu** problem since there is a relatively high barrier between the standard starting point and the global minimizer that prevents many methods from converging to the correct solution. Increasing the amount of perturbation may not always be the most effective use of resources: in cases where  $f^0$  and  $f^1$  are closely related (e.g., in terms of location of their global minimizers), less perturbation may lead to better results, with larger perturbations unnecessarily searching the domain in unpromising areas.

The ratios of function evaluations to successes per run (last four columns) show that although **HOPE** has a high success rate, it is expensive. For example, for  $p_{max} = 8$  and  $m = 8$ , the number of successful runs was 95, 98, 100, 100 for  $c_{max} = 2, 4, 8, 16$ , respectively, suggesting that larger ensemble sizes lead to more successful runs. An important question then is whether this increase in success justifies the corresponding

| $p_{max}$ | $c_{max}$ | Success (%)<br>when $m =$ |    |    |     | $N_f$ per Success<br>when $m =$ |        |        |       |
|-----------|-----------|---------------------------|----|----|-----|---------------------------------|--------|--------|-------|
|           |           | 1                         | 2  | 4  | 8   | 1                               | 2      | 4      | 8     |
| 1         | 2         | 7                         | 1  | 1  | 3   | 3245                            | 230915 | 157891 | 64000 |
|           | 4         | 5                         | 0  | 1  | 3   | 5365                            | —      | 257884 | 79734 |
|           | 8         | 6                         | 1  | 4  | 8   | 3959                            | 234786 | 122038 | 81714 |
|           | 16        | 8                         | 0  | 5  | 8   | 3724                            | —      | 96013  | 73627 |
| 2         | 2         | 12                        | 13 | 24 | 36  | 1897                            | 15710  | 6034   | 6244  |
|           | 4         | 7                         | 19 | 32 | 41  | 2320                            | 12370  | 7688   | 7971  |
|           | 8         | 9                         | 24 | 54 | 65  | 2388                            | 9177   | 7754   | 10150 |
|           | 16        | 4                         | 23 | 56 | 72  | 6265                            | 9104   | 7471   | 17395 |
| 4         | 2         | 9                         | 27 | 48 | 60  | 2418                            | 7239   | 2544   | 4538  |
|           | 4         | 5                         | 27 | 52 | 74  | 6337                            | 7709   | 4081   | 5039  |
|           | 8         | 7                         | 25 | 75 | 95  | 2978                            | 8310   | 5174   | 6652  |
|           | 16        | 8                         | 21 | 77 | 98  | 2901                            | 10253  | 4712   | 11717 |
| 8         | 2         | 22                        | 48 | 73 | 95  | 779                             | 3991   | 1598   | 2987  |
|           | 4         | 15                        | 49 | 79 | 98  | 1788                            | 4075   | 2368   | 5252  |
|           | 8         | 20                        | 54 | 94 | 100 | 1267                            | 3574   | 3355   | 8081  |
|           | 16        | 17                        | 59 | 95 | 100 | 995                             | 3153   | 2681   | 15442 |

TABLE 4.2

*Results of HOPE applied to Freu for different amounts of perturbation, maximum ensemble sizes and steps in  $\lambda$ .*

increase in the amount of computational effort. In this case, as  $c_{max}$  increases, there is only a gradual increase in success rate corresponding to a more dramatic increase in number of function evaluations performed per success. Specifically, for  $c_{max} = 8$ , HOPE had the same number of successes (100%) as for  $c_{max} = 16$ , but at half the computational effort (8081 to 15442  $N_f$  per success, respectively).

Another presentation of the results for  $p_{max} = 8$  is given in Figure 4.1, where the number of successful predictions of the global minimizer is plotted against the average number of calls to QNewton, the average total number of iterations of QNewton for all steps in  $\lambda$ , and the average number of function evaluations for each run. The four points for the different number of steps in  $\lambda$  correspond to the results for  $c_{max} = 1, 2, 4, 8$ . Again, the general trend is that HOPE is more successful when more steps in  $\lambda$  are taken. Also, as the maximum ensemble size increases, more computation was performed and this led to slightly better results. However, the increase in the amount of computation did not lead to significant increases in the number of successful predictions. This suggests that the number of steps in  $\lambda$  may impact performance of HOPE more than ensemble size.

**4.3. Influence of homotopy function.** Several experiments were performed on the Nmod function

$$f^1(x) = \sin(x) + \sin(Nx),$$

of Figure 4.2a, using the related function

$$f_a^0(x) = -\sin(x) + \sin(Nx)$$

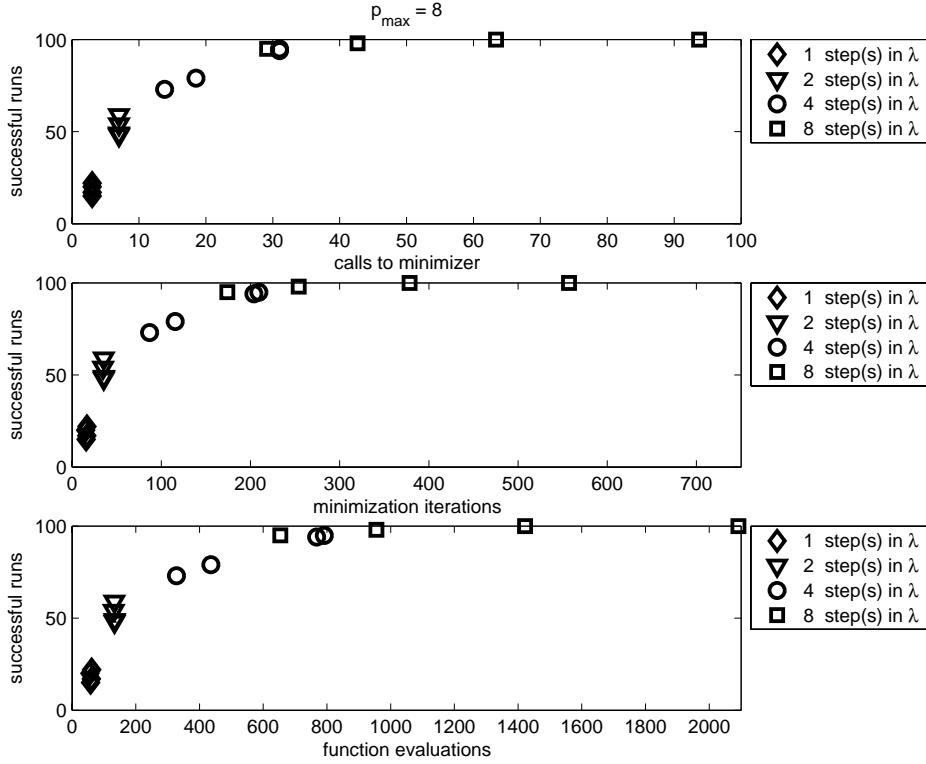


FIG. 4.1. Results of HOPE applied to Freu for  $p_{\max} = 8$ . The four markers for each value of  $m$  correspond to the four values of  $c_{\max} = 2, 4, 8, 16$ .

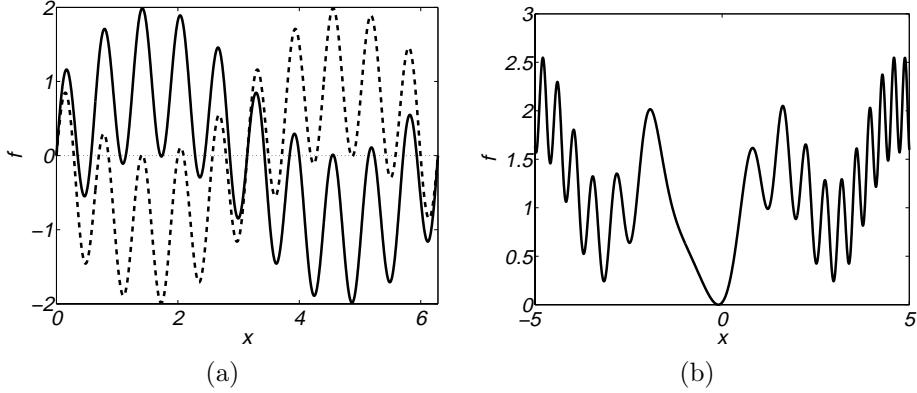


FIG. 4.2. (a) Plots of  $f_a^0$  (dashed) and  $f_a^1$  (solid) used in the Nmod function experiments. (b) Plot of Pint in one dimension with  $x^* = -0.1$ .

or the quadratic function

$$f_b^0(x) = \frac{1}{2}(x - \pi)^2. \quad (4.4)$$

The experiments consisted of 1000 runs of HOPE using  $\hat{c} = 1$ ,  $c_{\max} = 8$ , and

$m = kN/5$ ,  $k = 1, \dots, 5$ . QNewton was used for performing local minimization in HOPE with **MaxIter** = 10. Perturbations in HOPE were performed using  $\xi_{hr}$  with maximum perturbation lengths of  $p_{max} = 3\pi/N$ . Thus, perturbations of point  $x$  lay in the same basin of attraction as  $x$  or in one of the two adjacent basins (see Figure 4.2a).

Figure 4.3 shows plots of the percentage of successful runs versus  $m$ , the number of steps in  $\lambda$ , for  $N = 10, 20, 40, 60$ . The solid and dashed lines in the figure are the results using  $f_a^0$  and  $f_b^0$ , respectively. For smaller values of  $N$ , there is little difference in the performance of HOPE using the different homotopies. As  $N$  increases, there is a definite advantage in using the quadratic function,  $f_b^0$ .

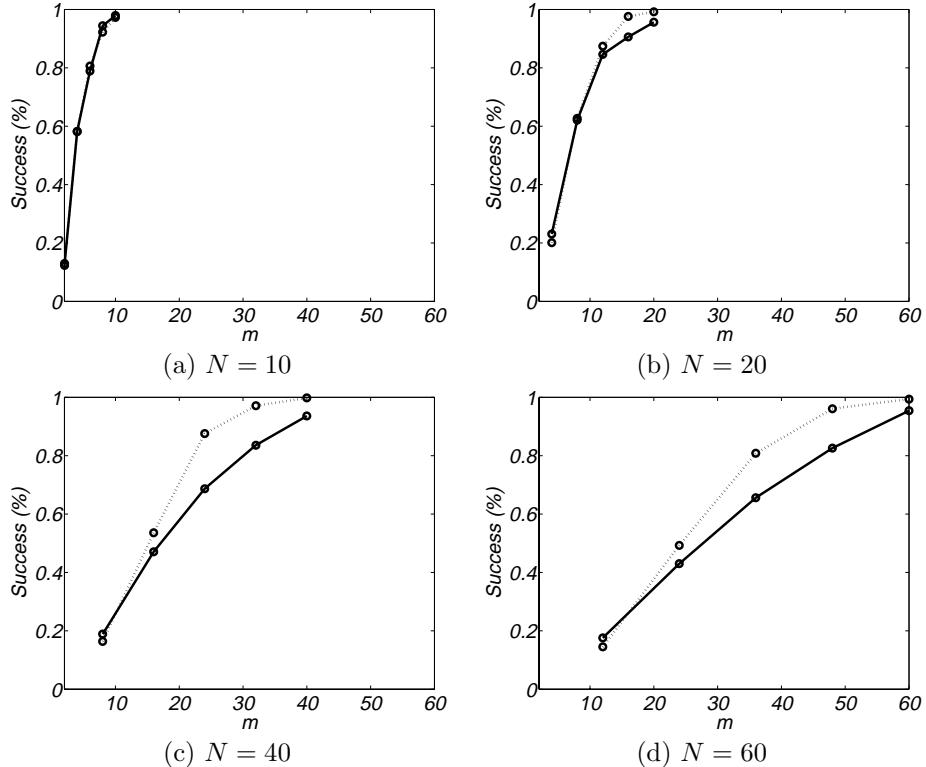


FIG. 4.3. Results of 1000 runs of HOPE applied to Nmod function  $f^1$  using  $f_a^0(x) = -\sin(x) + \sin(Nx)$  (solid lines) and  $f_b^0(x) = \frac{1}{2}(x - \pi)^2$  (dashed lines).

These results suggest that the choice of homotopy affects the performance of HOPE. In these experiments, the generic homotopy performed better;  $f_a^0$  provided a very poor homotopy since its global minimizer is far from  $x^*$  and there are many intermediate local minimizers. In general, we find that a well-designed customized homotopy function improves performance [16].

**4.4. Comparison of methods on a problem of varying size.** In [34], Pintér advocates using “randomized test functions that will have a randomly selected unique global solution” in testing global optimization methods. He argues that this reduces the ability to tune a method’s performance to the test problems. We follow this advice and test HOPE and HOM on the functions described by him in that work. The general

form of the test problem, denoted as **Pint**, is

$$f(x) = s \sum_{i=1}^n (x_i - x_i^*)^2 + \sum_{k=1}^{k_{max}} a_k \sin^2[f_k P_k(x - x^*)], \quad (4.5)$$

where  $x^*$  is the unique global solution,  $s > 0$  is a scaling factor,  $a_k > 0$  are amplitude scaling factors,  $f_k$  are (integer) frequency multipliers, and  $P_k(\cdot)$  are polynomial noise terms that vanish at the zero vector. Note that this defines a class of functions where each instance is specified by the choice of  $x^*$ . Thus, we can create a set of random test functions by randomly choosing the elements of  $x^*$ .

Figure 4.2b shows an example of a **Pint** function in 1-dimension with global minimizer  $x^* = -0.1$ . **Pint** functions have the property that the basin of attraction of the global minimizer is relatively large compared to the basins of other local minimizers. Also, far from the global minimizer, high frequency oscillation terms create local minimizers with deep narrow basins of attraction. Minimization methods that rely on local methods for searching for a global minimizer will likely converge to a local minimizer unless the method starts at a good approximation of the solution.

| Parameter      | Value   |
|----------------|---|
| $s$            | 0.025n  |
| $k_{max}$      | 2   |
| $a_k$          | 1   |
| $f_k$          | 1   |
| $P_1(x - x^*)$ | $\sum_{i=1}^n (x_i - x_i^*) + \sum_{i=1}^n (x_i - x_i^*)^2$ |
| $P_2(x - x^*)$ | $\sum_{i=1}^n (x_i - x_i^*)$                                |

TABLE 4.3  
Parameters and functions used to define the **Pint** functions.

Values of the parameters and the functions in  $f(x)$  used in the experiments are defined in Table 4.3. Note that the elements of the global solutions,  $x_i^*$  ( $i = 1, \dots, n$ ), for each problem instance were chosen from a uniform distribution on  $[-5, 5]$ . We applied HOPE to **Pint** functions of dimension  $n = 1, \dots, 10$ . The **Pint** function for  $n = 10$  is defined using

$$\begin{aligned} x^* = & (-3.0173, -4.4483, 4.6930, -4.7538, 1.5104, \\ & -3.9100, -4.3961, -1.4326, -0.3789, 1.4885)^T. \end{aligned} \quad (4.6)$$

These values were samples from a uniform distribution on  $[-5, 5]$ .

The homotopy function for HOPE used the function

$$f^0(x) = \frac{1}{2}(x - x^0)^2, \quad (4.7)$$

with

$$\begin{aligned} x^0 = & (-1.4127, 4.3035, -4.1816, -0.8379, 3.5322, \\ & 3.1757, 2.9291, 0.1542, 3.2336, 3.0290)^T. \end{aligned} \quad (4.8)$$

The **Pint** and homotopy functions for problems with  $n < 10$  are defined using the first  $n$  elements of each of  $x^*$  and  $x^0$  above.

In the first set of experiments, **HOPE** was compared to another global optimization method in cases where both methods perform the same amount of computation. We also focused on the the interplay between the number of steps taken in  $\lambda$  and the dimension of the function being minimized in the experiments of **HOPE**.

The global optimization method used in these experiments was a two-phase stochastic search method [38] consisting of *Multistart* [37] for the global phase and **QNewton** for local refinement, which we denote as **MS-QNewton**. The elements of each of the starting points used were chosen from a uniform distribution on  $[-5, 5]$ , and local minimization using **QNewton** was then performed from each of these points. Note that a single run of **MS-QNewton** consisted of this procedure being repeated until the total number of function evaluations performed,  $N_f$ , reached a specified maximum,  $\tilde{N}_f$ . The approximation to the global minimizer for each run was taken to be the point with lowest function value of all points found during the run.

In these experiments, **HOPE** was performed using **QNewton** for local minimization with **MaxIter** = 10; the **HOPE** parameters used were  $\hat{c} = 1$ ,  $c_{max} = 2^m$  (i.e., no pruning of duplicate points), and  $m = 1, 2, 4, 8$ . Perturbations of ensemble points in **HOPE** were generated using  $\xi_{pct}$  with a maximum of 10% perturbation ( $\tilde{p}_{max} = 0.10$ ). As for **MS-QNewton**, a single run of **HOPE** consisted of repeated calls to **HOPE** from different starting points (generated as for **MS-QNewton** above) until a maximum number of function evaluations were performed.

To compare the two methods, **HOPE** and **MS-QNewton** were run 20 times for each value of  $n$ , using several values for the maximum number of function evaluations allowed in each run. Table 4.4 presents the results of 20 runs of each method using two values of  $\tilde{N}_f$ . Column 1 shows the value of  $n$ , the dimension of the **Pint** function being minimized. Columns 2–6 present the percentages of successful runs for **MS-QNewton** and **HOPE** with  $m = 1, 2, 4, 8$ , respectively, using  $\tilde{N}_f = 1000$ ; columns 7–11 show the corresponding percentages for runs using  $\tilde{N}_f = 10000$ .

In both cases, **HOPE** with  $m = 1$  and **MS-QNewton** performed very comparably. This result was expected as the only difference between the two methods is that in each run of **HOPE** two local minimizations are performed—one from the starting point and one from a perturbed version of the starting point. For the experiments using  $\tilde{N}_f = 1000$ , there is a trend of decline in the success of **HOPE** as  $m$  increases. Note that none of the runs of **HOPE** with  $m = 8$  completed with fewer than 1000 function evaluations, and thus no runs were considered successful. This highlights the need for careful planning for the choices of parameters in **HOPE** when computational resources are limited.

For the experiments using  $\tilde{N}_f = 10000$ , the success rates of **MS-QNewton** and **HOPE** for all values of  $m$  are more consistent and significantly better for those using  $\tilde{N}_f = 1000$ . Again, there is an slight overall downward trend in performance of **HOPE** as  $m$  increases—for  $m = 1, 2, 4, 8$ , **HOPE** failed to find the global minimizer in 1, 2, 5, 16 runs, respectively, out of the total 200 runs for all values of  $n$  combined. This is compared to 4 failures in all 200 runs of **MS-QNewton**. Note that in these experiments, the average number of starting points per run over all  $n$  was 729.4 for **MS-QNewton** and 346.1, 121.0, 24.5, 1.2 for **HOPE** with  $m = 1, 2, 4, 8$ , respectively. The relatively consistent success rates for **HOPE** over these values of  $m$  suggest that for some problems, **HOPE** can be parameterized to use a few (or just one) starting points to produce acceptable results, even when the overall amount of computation

| $n$ | $N_f = 1000$ |                 |    |    | $N_f = 10000$  |                 |     |     |     |   |
|-----|--------------|-----------------|----|----|----------------|-----------------|-----|-----|-----|---|
|     | MS-QNewton   | HOPE when $m =$ |    |    | MS-QNewton     | HOPE when $m =$ |     |     |     |   |
|     |              | 1               | 2  | 4  | 8 <sup>†</sup> |                 | 1   | 2   | 4   | 8 |
| 1   | 100          | 100             | 90 | 50 | —              | 100             | 100 | 100 | 85  |   |
| 2   | 65           | 55              | 40 | 25 | —              | 100             | 100 | 100 | 80  |   |
| 3   | 40           | 50              | 40 | 20 | —              | 100             | 100 | 100 | 70  |   |
| 4   | 55           | 45              | 50 | 15 | —              | 100             | 100 | 100 | 90  |   |
| 5   | 45           | 45              | 40 | 15 | —              | 100             | 100 | 95  | 100 |   |
| 6   | 40           | 55              | 40 | 20 | —              | 100             | 95  | 100 | 100 |   |
| 7   | 20           | 20              | 35 | 25 | —              | 100             | 100 | 100 | 95  |   |
| 8   | 40           | 50              | 40 | 15 | —              | 95              | 100 | 95  | 100 |   |
| 9   | 35           | 35              | 30 | 15 | —              | 90              | 100 | 90  | 100 |   |
| 10  | 25           | 40              | 45 | 30 | —              | 95              | 100 | 100 | 100 |   |

<sup>†</sup>:  $N_f > 1000$  for each run of HOPE when  $m = 8$ .

TABLE 4.4

Percentages of successful runs of MS-QNewton and HOPE applied to Pint problems of dimensions  $n = 1, \dots, 10$ .  $N_f$  is the maximum number of function evaluations allowed for each run.

is constrained.

We conclude that on these problems, HOPE using local perturbations and a generic homotopy function performs as well as MS-QNewton, which uses global perturbations. Note that the perturbations used in MS-QNewton in these experiments were confined to the interval  $[-5, 5]$ ; in general such tight bounds around the global minimizer may not be known, and MS-QNewton may require more starting points (and thus more function evaluations) to perform as well in those cases. In contrast, we expect that using larger perturbations in HOPE would lead to higher success rates, as was demonstrated in the previous sections.

In a final experiment, MS-QNewton, HOM, and HOPE were used to solve the Pint problem from [34] with  $n = 100$ . For these methods, 100 runs were performed starting at random points (generated as in the experiments above). MS-QNewton and HOPE used the same parameters as above, except  $c_{max} = 4$ . Perturbations in HOPE were performed as above, using  $\xi_{pct}$  with  $\tilde{p}_{max} = 0.10$ . HOM was run using the same parameters as HOPE.

Table 4.5 presents the results of these experiments. The columns of the table show the method used, number of steps in  $\lambda$ , percentage of successful runs, average number of function evaluations per run ( $\bar{N}_f$ ), ratio of function evaluations performed to number of successes, and average function value of the predicted points ( $\bar{f^1(x^1)}$ ), respectively. These results present evidence that HOPE outperforms HOM and that HOM outperforms MS-QNewton. For the same number of starting points, HOPE clearly outperforms the other two methods, as shown by the percentage of successful runs. However the increase in success requires more computation per run. The ratio of function evaluations performed to number of successes allows for more direct comparison of the three methods. When this ratio is in the range of 16–18, HOPE is about twice as effective as the other methods (HOPE has 44 successes when the ratio is 16, HOM has 22 when the ratio is 17, and MS-QNewton has 20 when the ratio is 18). Results when the ratios are at 28 for HOPE and HOM also show increased success for HOPE: 70 successes for HOPE to HOM's 43 (approximately 1.63 times more).

| Method     | $m$ | Success (%) | $\bar{N}_f$ | $N_f$ per Success | $f^1(x^1)$ |
|------------|-----|-------------|-------------|-------------------|------------|
| MS-QNewton | —   | 20          | 35          | 18                | 1.431      |
| HOM        | 1   | 22          | 37          | 17                | 1.028      |
|            | 2   | 30          | 64          | 21                | 1.021      |
|            | 4   | 43          | 119         | 28                | 0.283      |
|            | 8   | 36          | 227         | 63                | 0.221      |
| HOPE       | 1   | 44          | 71          | 16                | 0.436      |
|            | 2   | 70          | 196         | 28                | 0.085      |
|            | 4   | 94          | 695         | 73                | 0.012      |
|            | 8   | 98          | 1739        | 177               | $10^{-11}$ |

TABLE 4.5

*Results of MS-QNewton, HOM, and HOPE applied to the Pint function with  $n = 100$ .*

**5. Discussion and conclusions.** We have presented two new methods for optimization. HOM differs from previous homotopy and continuation methods in that its aim is to find a minimizer for every step in the homotopy parameter, and it does not necessarily follow a path of minimizers. HOPE allows HOM to follow an ensemble of points obtained by perturbation of previous ones and is related to standard methods such as simulated annealing and stochastic search.

We have demonstrated that HOPE and HOM are more reliable than a quasi-Newton method and a stochastic search method in solving general unconstrained minimization problems. Results of several experiments suggest that as more steps in  $\lambda$  are taken and larger perturbations are used, the performance of HOPE improves. By taking more steps in  $\lambda$ , the function  $f^0$  is deformed more gradually into the function  $f^1$ . Such gradual change may be necessary for problems where these two functions behave very differently. The use of perturbations allows searching of the function domain in areas that may not be reachable by following curves of minimizers of the homotopy function. We suspect that larger perturbations will be more useful when little is known about the relationship between  $f^0$  and  $f^1$  and a generic homotopy function is used.

Both methods allow the homotopy to be tuned to a particular application, but we have shown that both are successful even when no domain-specific information is included.

For most of the experiments, HOPE performed considerably more computation than other methods, but the increase in cost was coupled with an increase in performance. An important feature of HOPE is that it is inherently parallelizable and communication between processors would be minimal.

Some refinements to our algorithm are possible:

- Several of the parameters used in HOPE could be determined adaptively, leading to more efficient use of computational resources and faster convergence for some problems. The value of  $\Delta^{(k)}$  could be determined, for example, as a function of the number of local minimization iterations required in the previous step, or based on the changes in homotopy function values from one step to the next. This may allow for larger (and thus fewer) steps in  $\lambda$  with no significant decrease in the success rate. Investigation into the use of adaptive steps in  $\lambda$  (as described in [3]), as well as into adaptively determining the amount of perturbation, number of perturbations of ensemble members to generate, and the maximum ensemble size may lead to more optimal imple-

mentations of HOPE.

- In this work, the ensemble members carried forward from the previous iteration were used as starting points for local minimization. However, we could extrapolate from these starting points using derivative information to produce starting points that may lead to less work in the local minimization [3].

These will be the subject of future work.

## REFERENCES

- [1] B. ADDIS AND S. LEYFFER, *A trust-region algorithm for global optimization*, Tech. Rep. MCS-P1190-0804, Argonne National Laboratory, 2004.
- [2] B. ADDIS, M. LOCATELLI, AND F. SCHOEN, *Local optima smoothing for global optimization*, Tech. Rep. DS15-2003, Dipartimento di Sistemi e Informatica, Universit degli Studi di Firenze, Firenze, Italy, 2003.
- [3] E. L. ALLGOWER AND K. GEORG, *Introduction to Numerical Continuation Methods*, SIAM, Philadelphia, PA, 2003.
- [4] T. ASSELMEYER, W. EBELING, AND H. ROSÉ, *Evolutionary strategies of optimization*, Phys. Rev. E, 56 (1997), pp. 1171–1180.
- [5] K. B. ATHREYA AND P. E. NEY, *Branching Processes*, Springer–Verlag, Berlin, 1972.
- [6] M. C. BIGGS, *Minimization algorithms making use of non-quadratic properties of the objective function*, J. Inst. Math. Appl., 8 (1971), pp. 315–327.
- [7] A. BONEH AND A. GOLAN, *Constraints' redundancy and feasible region boundedness by random feasible point generator (RFPG)*, 1979. Paper presented at EURO III, Amsterdam, The Netherlands.
- [8] A. DEKKERS AND E. AARTS, *Global optimization and simulated annealing*, Math. Program., 50 (1991), pp. 367–393.
- [9] A. DHOOGHE, W. GOVAERTS, AND Y. A. KUZNETSOV, *MATCONT: A Matlab package for numerical bifurcation analysis of ODEs*, ACM Trans. Math. Software, 29 (2003), pp. 141–164.
- [10] I. DIENER, *Trajectory methods in global optimization*, in *Handbook of Global Optimization*, R. Horst and P. M. Pardalos, eds., Kluwer Academic Publishers, The Netherlands, 1995, pp. 649–668.
- [11] L. C. W. DIXON AND G. P. SZEGÖ, eds., *Towards Global Optimisation, Volumes 1–2*, North-Holland, The Netherlands, 1975, 1978.
- [12] E. J. DOEDEL, *AUTO: A program for the automatic bifurcation analysis of autonomous systems*, in Proc. 10th Manitoba Conf. on Num. Math. and Comp., Univ. of Manitoba, Winnipeg, Canada, 1981, pp. 265–284.
- [13] E. J. DOEDEL, A. R. CHAMPNEYS, T. F. FAIRGRIEVE, Y. A. KUZNETSOV, B. SANDSTEDE, AND X.-J. WANG, *AUTO97: Continuation and bifurcation software for ordinary differential equations*, tech. rep., Department of Computer Science, Concordia University, Montreal, Canada, 1997.
- [14] E. J. DOEDEL, R. C. PAFFENROTH, A. R. CHAMPNEYS, T. F. FAIRGRIEVE, Y. A. KUZNETSOV, B. SANDSTEDE, AND X. WANG, *AUTO 2000: Continuation and bifurcation software for ordinary differential equations (with HomCont)*, tech. rep., Caltech, 2001.
- [15] D. M. DUNLAVY, *Homotopy Optimization Methods and Protein Structure Prediction*, PhD thesis, University of Maryland, 2005.
- [16] D. M. DUNLAVY, D. P. O'LEARY, D. KLIMOV, AND D. THIRUMALAI, *HOPE: A homotopy optimization method for protein structure prediction*, J. Comput. Biol., (to appear).
- [17] C. A. FLOUDAS AND P. M. PARDALOS, eds., *Recent Advances in Global Optimization*, Princeton University Press, Princeton, NJ, 1992.
- [18] F. FREUDENSTEIN AND B. ROTH, *Numerical solution of systems of nonlinear equations*, J. ACM, 10 (1963), pp. 550–556.
- [19] R. HORST AND P. M. PARDALOS, eds., *Handbook of Global Optimization*, Kluwer Academic Publishers, The Netherlands, 1995.
- [20] R. HORST AND H. TUY, *Global Optimization – Deterministic Approaches*, Springer–Verlag, Berlin, third ed., 1996.
- [21] L. INGBER, *Simulated annealing: Practice versus theory*, Math. Comput. Model., 18 (1993), pp. 29–57.
- [22] R. I. JENNICH AND P. F. SAMPSON, *Application of stepwise regression to nonlinear estimation*, Technometrics, 10 (1968), pp. 63–72.

- [23] S. KIRKPATRICK, C. D. GELATT JR., AND M. P. VECCHI, *Optimization by simulated annealing*, Science, 20 (1983), pp. 671–680.
- [24] A. KUNTSEVICH AND F. KAPPEL, *Moré set of test functions*. <http://www.uni-graz.at/imawww/kuntsevich/solvopt/results/moreset.html>.
- [25] Y. A. KUZNETSOV AND V. V. LEVITIN, *CONTENT: Integrated environment for analysis of dynamical systems*, tech. rep., CWI, Amsterdam, 1997.
- [26] M. LOCATELLI, *Simulated annealing algorithms for continuous global optimization*, in Handbook of Global Optimization, Volume 2, P. M. Pardalos and H. E. Romeijn, eds., Kluwer Academic Publishers, The Netherlands, 2002, pp. 179–229.
- [27] N. METROPOLIS, A. ROSENBLUTH, M. ROSENBLUTH, A. TELLER, AND E. TELLER, *Equations of state calculations by fast computing machines*, J. Chem. Phys., 21 (1953), pp. 1087–1090.
- [28] R. R. MEYER, *Theoretical and computational aspects of nonlinear regression*, in Nonlinear Programming, J. B. Rosen, O. L. Mangasarian, and K. Ritter, eds., New York, NY, 1970, Academic Press, pp. 465–496.
- [29] A. MIGDALAS, P. M. PARDALOS, AND P. VÄRBRAND, eds., *From Local to Global Optimization*, Kluwer Academic Publishers, The Netherlands, 2001.
- [30] J. J. MORÉ, B. S. GARBOW, AND K. E. HILLSTROM, *Testing unconstrained optimization software*, ACM Trans. Math. Software, 7 (1981), pp. 17–41.
- [31] A. NEUMAIER, *Global optimization*. <http://www.mat.univie.ac.at/~neum/glopt.html>.
- [32] ———, *Complete search in continuous global optimization and constraint satisfaction*, Acta Numerica, (2004), pp. 271–369.
- [33] P. M. PARDALOS AND H. E. ROMEIJN, eds., *Handbook of Global Optimization, Volume 2*, Kluwer Academic Publishers, The Netherlands, 2002.
- [34] J. D. PINTÉR, *Global optimization: Software, test problems, and applications*, in Handbook of Global Optimization, Volume 2, P. M. Pardalos and H. E. Romeijn, eds., Kluwer Academic Publishers, The Netherlands, 2002, pp. 515–569.
- [35] A. G. SALINGER, N. M. BOU-RABEE, R. P. PAWLOWSKI, E. D. WILKES, E. A. BURROUGHS, R. B. LEHOUCQ, AND L. A. ROMERO, *LOCA 1.1, a library of continuation algorithms: Theory and implementation manual*, tech. rep., Sandia National Laboratories, Albuquerque, NM, 2002.
- [36] S. SCHELSTRAETE, W. SCHEPENS, AND H. VERSCHELDE, *Energy minimization by smoothing techniques: A survey*, in Molecular Dynamics. From Classical to Quantum Methods, P. Balabuena and J. Seminario, eds., Elsevier, 1998, pp. 129–185.
- [37] F. SCHOEN, *Stochastic techniques for global optimization: a survey of recent advances*, J. Global Optim., 1 (1991), pp. 207–228.
- [38] ———, *Two-phase methods for global optimization*, in Handbook of Global Optimization, Volume 2, P. M. Pardalos and H. E. Romeijn, eds., Kluwer Academic Publishers, The Netherlands, 2002, pp. 151–177.
- [39] R. L. SMITH, *Monte Carlo procedures for generating random feasible solutions to mathematical programs*, 1980. Paper presented at ORSA/TIMS Conference, Washington, D.C.
- [40] E. SPEDICATO, *Computational experience with quasi-Newton algorithms for minimization problems*, Tech. Rep. CISE-N-175, Segrate, Milano, Italy, 1975.
- [41] A. C. SUN AND W. D. SEIDER, *Homotopy-continuation algorithm for global optimization*, in Recent Advances in Global Optimization, C. A. Floudas and P. M. Pardalos, eds., Princeton University Press, Princeton, NJ, 1992, pp. 561–592.
- [42] A. TÖRN AND A. ŽILINSKAS, *Global Optimization*, Springer–Verlag, Berlin, 1989.
- [43] P. J. M. VAN LAARHOVEN AND E. H. L. AARTS, *Simulated Annealing: Theory and Applications*, D. Reidel Publishing Company, Dordrecht, The Netherlands, 1987.
- [44] D. J. WALES AND H. A. SCHERAGA, *Global optimization of clusters, crystals, and biomolecules*, Science, 284 (1999), pp. 1368–1372.
- [45] L. T. WATSON, *Theory of globally convergent probability-one homotopies for nonlinear programming*, SIAM J. Optim., 11 (2000), pp. 761–780.
- [46] L. T. WATSON, S. C. BILLUPS, AND A. P. MORGAN, *Algorithm 653: HOMPACK: A suite of codes for globally convergent homotopy algorithms*, ACM Trans. Math. Software, 13 (1987), pp. 281–310.
- [47] L. T. WATSON AND R. T. HAFTKA, *Modern homotopy methods in optimization*, Comput. Methods Appl. Mech. Engrg., 74 (1989), pp. 289–304.
- [48] L. T. WATSON, M. SOSONKINA, R. C. MELVILLE, A. P. MORGAN, AND H. F. WALKER, *Algorithm 777: HOMPACK90: A suite of Fortran 90 codes for globally convergent homotopy algorithms*, ACM Trans. Math. Software, 23 (1997), pp. 514–549.
- [49] Z. B. ZABINSKY, R. L. SMITH, J. F. McDONALD, H. E. ROMEIJN, AND D. E. KAUFMAN, *Improving hit-and-run for global optimization*, J. Global Optim., 3 (1993), pp. 171–192.