*Journal of Global Optimization* **21:** 27–37, 2001.



# A Locally-Biased form of the DIRECT Algorithm ⋆

J.M. GABLONSKY and C.T. KELLEY
*North Carolina State University, Center for Research in Scientific Computation and Department of Mathematics, Box 8205, Raleigh, N.C. 27695-8205 (e-mail: Tim_Kelley@ncsu.edu)*

(Received 22 December 2000; accepted in revised form 27 March 2001)

**Abstract.** In this paper we propose a form of the DIRECT algorithm that is strongly biased toward local search. This form should do well for small problems with a single global minimizer and only a few local minimizers. We motivate our formulation with some results on how the original formulation of the DIRECT algorithm clusters its search near a global minimizer. We report on the performance of our algorithm on a suite of test problems and observe that the algorithm performs particularly well when termination is based on a budget of function evaluations.

**Key words:** DIRECT, Locally-biased formulation, Local clustering

## 1. Introduction

The DIRECT (DIviding RECTangles) algorithm [13, 14] is a pattern search method (in the sense of [17]) that balances local and global search in a attempt to efficiently find a global optimizer. Other deterministic sampling methods, such as implicit filtering [9, 15], MDS [6], Hooke-Jeeves [10], or Nelder-Mead [16], drive an approximate gradient to zero and are not designed for global search. DIRECT, on the other hand, is designed to completely explore the variable open space, even after one or more local minima have been identified. This feature has even been exploited to generate initial iterates for other sampling methods [3].

In this paper we propose a form of the DIRECT algorithm that is more biased toward local search. This form should do well for small problems with a single global minimizer and only a few local minimizers. We motivate our formulation with some results on how the original formulation of the DIRECT algorithm clusters its search near a global minimizer. We then illustrate the performance of the new approach on a set of test problems.

We consider bound-constrained global optimization problems

$$\min_{x\in\Omega} f(x) \tag{1.1}$$

⋆ This research was supported by National Science Foundation grants #DMS-0070641 and #DMS-9714811. Computing activity was partially supported by an allocation from the North Carolina Supercomputing Center.

where $\Omega \subset R^N$ is a hyperrectangle. DIRECT's performance is independent of the scaling of $\Omega$ and problems are typically scaled so that

$$\Omega = \{x | x_i \in [0, 1]\}, \tag{1.2}$$

where $x_i$ is the $i$th component of the vector $x$.

The algorithm begins with a single hyperrectangle $\Omega$ and, at each sweep, updates a set $\mathcal{S}$ of hyperrectangles by dividing some of its members. Each hyperrectangle $\mathcal{S} \in \mathcal{S}$ has side length $3^{-l}$ (long directions) or $3^{-l-1}$ (short directions) for some $l \geqslant 0$ and $f$ has been evaluated at the center. The decision to divide $\mathcal{S}$ is based on its size and the value of $f$ at the center. When $\mathcal{S}$ is divided, $f$ is first evaluated at points midway (along the long coordinate directions) between the center $c$ and the boundary of $S$. $S$ is divided into three parts along the long coordinate direction corresponding to the second smallest function value in the stencil. The subrectangle containing $c$ is divided again along the long coordinate direction corresponding to the second smallest function value in the stencil. This process continues until each point on the stencil is the center of a new hyperrectangle. The rule for tie-breaking is not important and we use the order of the coordinates.

The formulation in [13, 14] begins by evaluating $f$ at the center of $\Omega$ and dividing $\Omega$ according to the rule described above. The new set $\mathcal{S}$ consists of the small hyperrectangles that came from that division. The new hyperrectangles are identified with their centers. Any or all of the hyperrectangles may be divided again if the value of $f$ at the center is sufficiently small relative to the size of the hyperrectangle. The algorithm continues in this way, always dividing based on the function value and the size of a hyperrectabgle, until a given budget of function evaluations in exhausted.

The description given above needs to be expanded for our purposes. Any hyperrectangle will be divided along a side of maximum length. Hence if the longest side has length $3^{-l}$, the shortest side will be at least of length $3{-}l - 1$. A rectangle has *level* $l$ if the length of the longest side is $3^{-l}$. A rectangle of level $l$ is at *stage* $0 \leq p \leq N - 1$ (i.e. is the result of $p$ subdivisions of a cube of side length $3^{-l}$) if $N - p$ sides have length $3^{-l}$ and $p$ have length $3^{-l-1}$.

The formulation in [13, 14] groups rectangles by level and stage by grouping all hyperrectangles having the same diameter. The $\ell^2$ diameter of a hyperrectangle with level $l$ and stage $p$ is

$$d(l, p) = \left( \sum_{k=1}^{p} 3^{-2l-2} + \sum_{k=p+1}^{N} 3^{-2l} \right)^{1/2} = 3^{-l}(N - 8p/0)^{1/2}.$$

So, after $L$ sweeps of DIRECT, hyperrectangles with at most $NL$ different diameters have been created.

The implementation in [8] groups hyperrectangles by the $\ell^\infty$ diameter, i.e. the length of the longest side. This grouping has fewer groups and, therefore, biases

the search more toward local exploration near good points rather than global search in regions of $\Omega$ that have been sampled sparsely. The purpose of this paper is to motivate that variation of DIRECT and illustrate its performance by numerical examples.

The results in this paper apply to both methods of categorizing hyperrectangles. If we group by $\ell^2$ diameter we define the size of a hyperrectangle with level and stage $l$ and $p$ as

$$\sigma_2 = d(l, p)/2, \tag{1.3}$$

and if we group by $\ell^\infty$ we define the size as

$$\sigma_\infty = 3^{-l}/2. \tag{1.4}$$

The two modes of grouping can be described as grouping by size.

A hyperrectangle $S$ with center $c = c(S)$ and size $\sigma = \sigma(S)$ is *potentially optimal* if there is a value of the Lipschitz constant of $f$ that is consistent with the optimal point being inside the hyperrectangle and the potential improvement is nontrivial.

The first condition means that there is $\tilde{K}$ such that

$$f(c) - \tilde{K}\sigma \leq \min_{\hat{S}} \left( f(c(\hat{S})) - \tilde{K}\sigma(\hat{S}) \right), \tag{1.5}$$

where the minimum is taken over all hyperrectangles $\hat{\mathrm{S}}$. There is a simple and efficient way to test for (5) [14]. For the purposes of this paper it suffices to point out that if a hyperrectangle with center $c$ and size $\sigma$ satisfies (5), then no rectangle with the same size can have a lower function value at the center i.e.

$$f(c) = \min_{\{S|\sigma(S)=\sigma\}} f(c(S)).$$

The potential improvement is nontrivial if

$$f(c) - \tilde{K}\sigma \leqslant f_{min} - \epsilon|f_{min}|, \tag{1.6}$$

where $\tilde{K}$ alo satisfies (5). In (6) $\epsilon$ is a parameter in the algorithm and $f_{min}$ is the minimum value of $f$ found so far in the iteration

$$f_{min} = \min_S f(c(S)).$$

The role of the parameter $\epsilon$ is to avoid oversampling near points with low function values and bias the sampling toward global search. Such a bias is important if there are many local minima or $N$ is large. Values of $\epsilon \in [10^{-7},10^{-3}]$ are reported to work well in [14] and

$$\epsilon = \max(10^{-4}|f_{min}, 10^{-8}) \tag{1.7}$$

is recommended in [13].

A sweep of DIRECT identifies the potentially optimal hyperrectangles from the previous sweep and then divides each of them once. Then a new set of potentially optimal hyperrectangles is identified. Both the DIRECT algorithm from [13, 14] and the locally biased version are sequences of these sweeps.

There is little convergence theory for DIRECT beyond the observation from [14] that the search will eventually sample arbitrarily near every point in $\Omega$. The method has been applied to optimal design of gas pipe lines [3–5] and aerospace engineering [1, 2] and seems to perform well, especially in the early stages of an optimization.

In this paper we quantify how the subdivisions cluster near a global minimizer and use this result to motivate an alternative version of DIRECT [8], which is different from that in [13, 14] in that the $\ell^\infty$ norm is used to form the groups and at most one hyperrectangle from each group is subdivided, even if there are more than one potentially optimal hyperrectangle in a group.

## 2. Local Clustering

### 2.1. ELEMENTARY PROPERTIES OF DIRECT

The *state* $\mathcal{S}$ of DIRECT is the set of all hyperrectangles at a given sweep of the optimization. If $\mathcal{S}$ is the state of DIRECT, two special classes of hyperrectangles are guaranteed to be potentially optimal.

LEMMA 1. *If $S \in \mathcal{S}$ has maximal size and $f(c(S))$ has the smallest value among centers of hyperrectangles in $\mathcal{S}$ of maximal size, i.e.*

$$\sigma(S) = \max_{S \in \mathcal{S}} \sigma(\hat{S} \text{ and } f(c(S)) = \min_{\hat{S} \in \mathcal{S}, \sigma(\hat{S}) = \sigma(S)} f(C(\hat{S})) \tag{2.8}$$

*then $S$ satisfies (5).*

*Proof.* Let $\sigma = \sigma(s)$ and $c = c(S)$. We will find $\tilde{K}$ such that

$$f(c) - \tilde{K}\sigma \leqslant f(c(\hat{S})) - \tilde{K}\sigma(\hat{S}) \tag{2.9}$$

for all $\hat{S} \in \mathcal{S}$. This will imply (5) for $\tilde{K}$ sufficiently large. Increasing $\tilde{K}$ if needed, will imply (6), and hence potential optimality.

Now, let $\hat{S} \in \mathcal{S}$, $\hat{\sigma} = \sigma(\hat{S})$, and $\hat{c} = c(\hat{S})$; by assumption $\hat{\sigma} \leqslant \sigma$.

We consider three cases. If $\hat{\sigma} = \sigma(\hat{S}) = \sigma$ then (8) implies that $f(c) \leqslant f(c(\hat{\sigma}))$. Hence (5) holds for any $\tilde{K}$.

If $\hat{\sigma} < \sigma$ and $f(c) \leq f(\hat{c})$ then (5) also holds for any $\tilde{K}$.

Finally, if $\hat{\sigma} < \sigma$ and $f(c) > f(\hat{c})$, then (5) holds for any

$$\tilde{K} \geqslant \max \frac{f(c) - f(\hat{c})}{\sigma - \hat{\sigma}},$$

where the maximum is taken over all $\hat{S} \in \mathcal{S}$ such that $\hat{\sigma} < \sigma$ and $f(c) > f(\hat{c})$.

Hence (9) holds for sufficiently large $K$ and the proof is complete.

If $\epsilon$ is too large, the search will be strongly biased toward global exploration. This will delay refinement near a global minimum. Lemma 2.2 quantifies this statement by giving an upper bound on $\epsilon$ that will guarantee good local exploration. This upper bound (13) combines the two conditions for potential optimality in a way that includes the largest rectangles having the minimum value at the center. (13) is not a significant restriction in practice.

LEMMA 2. *If $f(c(S)) = f_{min}$, the smallest value at all the centers, and $\sigma(S)$ is the largest among all rectangles having the same value at the center, i.e.*

$$f(c(S)) = \min_{s \in \mathcal{S}} f(c(\hat{S})) \text{ and } \sigma(S) = \max_{\hat{S} \in \mathcal{S},\, f(c(\hat{S})) = f_{min}} \sigma(\hat{S}) \tag{2.10}$$

*then $S$ is potentially optimal if $\epsilon$ is sufficiently small.*

*Proof.* As before we let $\sigma = \sigma(S)$ and $c = c(S)$. Since $f(c) = f_{min}$, (6) holds if

$$\epsilon \leqslant \frac{\tilde{K}\sigma}{f_{min}}. \tag{2.11}$$

We now verify that (5) holds.

Let $\hat{S} \in S, \hat{\sigma} = \sigma(\hat{S})$, and $\hat{c} = c(\hat{S})$. If $\hat{\sigma} \leqslant \sigma$, then (5) holds for all $\tilde{K} \geqslant 0$ since $f(c) = f_{min}$. If $f(\hat{c}) = f_{min}$ then (10) implies that (5) holds for all $\tilde{K} \geqslant 0$. The remaining case is $\hat{\sigma} > \sigma$ and $f(\hat{c}) > f(c) = f_{min}$. In that case (5) holds for

$$\tilde{K} \leqslant \min \frac{f(\hat{c}) - f_{min}}{\hat{\sigma} - \sigma} \tag{2.12}$$

where the minimum is taken over all $\hat{S} \in \mathcal{S}$ such that $\hat{\sigma} > \sigma$ and $f(c) < f(\hat{c})$.

Combining (11) and (12) we see that the bound on $\epsilon$ is

$$\epsilon \leqslant \frac{\sigma}{f_{min}} \min \frac{f(\hat{c}) - f_{min}}{\hat{\sigma} - \sigma}. \tag{2.13}$$

### 2.1.1. *Local Clustering Theory*

In this section we discuss some consequences of Lemma 2.2 that show how, if there is a single global minimizer, the subdivisions become refined near that minimizer as the sweeps progress. We let $\mathcal{S}_n$ be the state of DIRECT after $n$ sweeps. We will quantify the progress to optimality in terms of the minimum value of $f$ in the $n$th sweep

$$f_{min}^n = \min_{S \in \mathcal{S}_n} f(c(S)).$$

The minimum is attained on the non-empty set

$$\mathcal{S}^n_{min} = \{S \in \mathcal{S}^n | f(c(S)) = f^n_{min}\}.$$

Let

$$v_n = \max_{S \in \mathcal{S}^n_{min}} \sigma(S). \tag{2.14}$$

Let

$$\mathcal{T}^n = \{S \in \mathcal{S}^n_{min} | \sigma(S) = v_n\}. \tag{2.15}$$

The hyperrectangles in $\mathcal{T}^n$ are the ones that Lemma 2.2 identifies as potentially optimal.

The hyperrectangles and their centers can cluster in one of three ways. The number of hyperrectangles in $\mathcal{T}^n$ can increase, $v_n$ can decrease, or the $f^n_{min}$ can decrease. However, these three do not need to happen simultaneously in a single sweep. Our clustering result Theorem 2.1 characterizes how at least one of the three modes of clustering must occur in each sweep.

THEOREM 1. *For all n every $S \in \mathcal{T}^n$ is potentially optimal while (11) holds. Moreover, at least one of*

$$f^{n+1}_{min} < f^n_{min}, \tag{2.16}$$

$$V_{N+1} \leqslant v_n/3, \tag{2.17}$$

*or*

$$|S^{n+1}_{min}| > |S^n_{min}|, \tag{2.18}$$

*holds.*

*Proof.* $v_{n+1} > v_n/3$ implies that $f^{n+1}_{min}$ is attained in a hyperrectangle that is not the result of a subdivision of any member of $\mathcal{T}^n$. Hence either $f^{n+1}_{min} < f^n_{min}$ or the minimum value is the same and a new hyperrectangle with the minimum value $f^{n+1}_{min} = f^n_{min}$ at the center has been created. This completes the proof.

## 2.2. LOCALLY-BIASED FORMULATION

As one can see from the discussion above, the density of the subdivisions will increase near a global minimizer. However, the cost of a sweep can be dominated by global search.

If one knows that there are only a few local minima, then biasing the search more toward local improvement can reduce the cost of a sweep and more rapidly

identify the global minimum. Such a modification is likely to be more useful for small $N$, as for larger $N$ more work will be needed in the global search even to explore the design space at a coarse level [1, 2].

A globally-biased version of direct, called *aggressive DIRECT*, was proposed in [1]. In that approach the potential optimality condition is abandoned and the hyperrectangles with the lowest function value in each group are all subdivided.

The formulation from [8] differs from that in [13, 14] in that the hyperrectangles are grouped by $\sigma_\infty$ and at most one hyperrectangle from each group is subdivided, even if there are more than one potentially optimal hyperrectangle in some of the groups. The first of these differences reduces the number of groups and the second reduces the number of divisions within a group. The idea is that the overall number of divisions will be reduced and that most of this reduction will be in the large hyperrectangles that are not near the global optimum. We will refer to this method as DIRECT-1.

We will illustrate the performance advantages of DIRECT-1 for small problems with only a few global minima in $\mathcal{S}3$. We close this section by showing that the conclusions of Theorem 2.1 also hold for DIRECT-1.

COROLLARY 1. *For all $n$ every $S \in \mathcal{T}^n$ is potentially optimal while (11) holds. Moreover, either at least one of (16), (17), or (18) holds, or*

$$\mathcal{T}^{n+1} \subset \mathcal{T}^n \text{ and } |\mathcal{T}^{n+1}| = |\mathcal{T}^n| - 1. \tag{2.19}$$

*Proof.* In DIRECT-1 only one $S \in \mathcal{T}^n$ is selected for subdivision. Hence, if $\mathcal{T}^n$ has more than one element, only one will be divided. If one of the new, smaller, rectangles is not potentially optimal, then (19) will hold. This is the only difference from the proof of Theorem 2.1.

## 3. Numerical Results

We compare two formulations of DIRECT, the original formulation from [14] and a strongly locally-biased form. The differences in the algorithms are summarized in Table 1.

*Table I.* Two formulations of DIRECT

| Formulation | DIRECT | DIRECT-1 |
|---|---|---|
| $\sigma$ | $\sigma_2$ | $\sigma_\infty$ |
| $\epsilon$ | $10^{-3}$ | $10^{-3}$ |
| Division | all potentially optimal hyperrectangles | one potentially optimal hyperrectangle at each level |
| Subdivision Order | [14], pg 169 | [14], pg 169 |

In Table 2 we list the test functions we used in our numerical experiments. We give the name of the function, the dimension of the problem, the domain over which the function is defined, the number of global minima, and the global minimal value.

*Table II.* Test Problems

| # | Name | $N$ | $\Omega$ | Global minima | |
|---|---|---|---|---|---|
| | | | | number | function value |
| 1 | Branin | 2 | $[-5,10]\times[0,15]$ | 3 | 0.398 |
| 2 | Shekel-5 | 4 | $[0, 10]^4$ | 1 | $-10.153$ |
| 3 | Shekel-7 | 4 | $[0, 10]^4$ | 1 | $-10.403$ |
| 4 | Shekel-10 | 4 | $[0, 10]^4$ | 1 | $-10.536$ |
| 5 | Hartman-3 | 3 | $[0,1]^3$ | 1 | $-3.863$ |
| 6 | Hartman-6 | 6 | $[0, 1]^6$ | 1 | $-3.322$ |
| 7 | Goldprice | 2 | $[-2, 2]^2$ | 1 | 3.000 |
| 8 | Sixhump | 2 | $[-3, 3]\times[-2, 2]$ | 2 | $-1.032$ |
| 9 | Shubert | 2 | $[-10, 10]^2$ | 18 | $-186.831$ |

The first seven problems were taken from [7]. These problems have been widely used to compare global optimization algorithms [11, 12, 14]. Problems eight and nine are from [18]. These nine test problems were used in [14] to test the original implementation of DIRECT.

We report comparisons of DIRECT and DIRECT-1 in Tables 3 and 4. Our test problems are small and the computer time for a function evaluation is, for most of the problems, not a factor in the computer time for the entire optimization. The is different from the case for many practical problems, where functions evaluations are expensive. We and others [1, 2, 14] think that counting function evaluations is a more effective way to compare methods.

### 3.1. TERMINATION BASED ON THE GLOBAL MINIMUM

In Table 3 we report on the performance of the two variants of DIRECT using the termination criterion from [14], which uses knowledge of the global minimum. Let $f_{global}$ be the known global minimal function value and denote by $f_{min}$ the best function value found by DIRECT. We can then define the percent error $p$ as

$$p = 100 \begin{cases} \frac{f_{min} - f_{global}}{|f_{global}|}, & f_{global} \neq 0, \\ (f_{min} - f_{global}), & f_{global} = 0, \end{cases}$$

and terminate the iterations once $p$ is lower than $10^{-2}$ or over 20 000 function evaluations have been completed at the end of a sweep.

*Table III.* Numerical results with percentage termination criteria

| Problem | $N$ | DIRECT | | | DIRECT-1 | | |
|---|---|---|---|---|---|---|---|
| | | f-eval. | Time | $p$ | f-eval. | Time | $p$ |
| 1 | 2 | 195 | 0.023 | 0.98E-03 | 159 | 0.024 | 0.98E-03 |
| 2 | 4 | 155 | 0.024 | 0.84E-02 | 147 | 0.024 | 0.84E-02 |
| 3 | 4 | 145 | 0.024 | 0.93E-02 | 141 | 0.024 | 0.93E-02 |
| 4 | 4 | 145 | 0.025 | 0.97E-02 | 139 | 0.024 | 0.97E-02 |
| 5 | 3 | 199 | 0.024 | 0.85E-02 | 111 | 0.024 | 0.85E-02 |
| 6 | 6 | 571 | 0.028 | 0.89E-02 | 295 | 0.026 | 0.89E-02 |
| 7 | 2 | 191 | 0.024 | 0.30E-02 | 115 | 0.023 | 0.30E-02 |
| 8 | 2 | 285 | 0.024 | 0.48E-03 | 191 | 0.024 | 0.48E-03 |
| 9 | 2 | 2967 | 0.060 | 0.50E-02 | 2043 | 0.055 | 0.50E-02 |

Both the original DIRECT and our modification find an acceptable solution for all of the problems. Note that our modification always needs less function evaluations, significantly less for problems 5 through 9.

These results emphasize our earlier observations. Our modification should be used for lower dimensional problems, which do not have too many local and global minima. We believe that the original DIRECT or even the more aggressive version from [1] is the better choice for higher dimensional problems.

### 3.2. TERMINATION ON A BUDGET

In Table 4 we show the results when we give both methods a budget of 100 function evaluations. We implement this by examination of the number of function evaluations completed after each sweep and terminate the optimization when the budget has been exhausted. Since sweeps are not stopped before completion, the number of function evaluations will exceed the budget. This kind of termination criterion is what would be used in practice.

The results of both versions of DIRECT are nearly identical for problems 2–4, and 9. This is consistent with the results with the percentage termination criteria shown in Table 3. The original DIRECT algorithm finds a much better point than DIRECT-1 on problem 8.

For problems 1 and 5–7, DIRECT-1 finds significantly better points with about the same number of function evaluations as the original DIRECT algorithm. To summarize, for all problems other than problem 8 DIRECT-1 finds a better solution than the original DIRECT algorithm with about the same number of function evaluations.

*Table IV.* Numerical results with a budget of 100 function evaluations

| Problem | $N$ | DIRECT | | | DIRECT-1 | | |
|---|---|---|---|---|---|---|---|
| | | f-eval. | Time | $p$ | f-eval. | Time | $p$ |
| 1 | 2 | 117 | 0.024 | 0.84E-01 | 103 | 0.023 | 0.39E-01 |
| 2 | 4 | 103 | 0.024 | 0.59E+00 | 107 | 0.024 | 0.59E+00 |
| 3 | 4 | 107 | 0.024 | 0.58E+00 | 101 | 0.024 | 0.58E+00 |
| 4 | 4 | 107 | 0.024 | 0.56E+00 | 117 | 0.024 | 0.41E+00 |
| 5 | 3 | 113 | 0.024 | 0.15E+00 | 111 | 0.024 | 0.85E-02 |
| 6 | 6 | 101 | 0.024 | 0.27E+02 | 109 | 0.024 | 0.23E+01 |
| 7 | 2 | 101 | 0.023 | 0.25E+00 | 101 | 0.023 | 0.27E-01 |
| 8 | 2 | 113 | 0.023 | 0.79E+00 | 111 | 0.023 | 0.16E+01 |
| 9 | 2 | 101 | 0.024 | 0.83E+02 | 103 | 0.024 | 0.82E+02 |

The numerical experiments were done on a Sun Ultra 10 workstation with a UltraSPARC-IIi processor with 440 MHz and 256 MByte of RAM, running SOLARIS version 5.6. The software was compiled using Sun WorkShop Compiler FORTRAN 77 5.0. The run times in the tables are given in seconds.

## 4. Conclusion

DIRECT-1 is a modification of the DIRECT [14] algorithm that biases the search toward exploration near local minima. The algorithm was designed for low-dimensional problems with only a few global minima. Our experimental results show that DIRECT-1 performs well for such problems and particularly well when termination is based on a low budget of function evaluations.

## 5. Acknowledgements

The authors are grateful to Richard Carter, Evin Cramer, Don Jones, and Layne Watson for sharing their insights on the performance and implementation of DIRECT.

## References

1. Baker, C.A., Watson, L.T., Grossman, B., Haftka, R.T. and Mason, W.H. (1999a), Parallel Global Aircraft Configuration Design Space Exploration, (preprint).
2. Baker, C.A., Watson, L.T., Grossman, B., Mason, W.H. Cox, S.E. and Haftka, R.T. (1999b), Study of a Global Design Space Exploration Method for Aerospace Vehicles, (preprint).
3. Carter, R., Gablonsky, J.M., Patrick, A., Kelley, C.T. and Eslinger, O.J. (2000), Algorithm for Noisy Problems in Gas Transmission Pipeline Optimization. Technical Report CRSC-TR00-10, North Carolina State University, Center for Research in Scientific Computation.

4. Carter, R.G. (1998), Pipeline optimization: dynamic programming after 30 years. Proceedings of the Pipeline Simulation Interest Group, Denver Colorado, Paper number PSIG-9803.
5. Carter, R.G. (1999), Private communication.
6. Dennis, J.E. and Torczon, V. (1991), Direct Search Methods on Parallel Machines, *SIAM J. Optim.* 1: 448–474.
7. Dixon, L. and Szegö, G. (1978), The Global Optimisation Problem: An Introduction, In: L. Dixon and G. Szegö (eds.), *Towards Global Optimization 2*, Vol. 2, North-Holland Publishing Company, pp. 1–15.
8. Gablonsky, J. (1998), An Implementation of the DIRECT Algorithm. Technical Report CRSC-TR98-29, North Carolina State University, Center for Research in Scientific Computation.
9. Gilmore, P. and Kelley, C.T. (1995), An implicit filtering algorithm for optimization of functions with many local minima, *SIAM J. Optim.* 5: 269–285.
10. Hooke, R. and Jeeves, T.A. (1961), 'Direct search' solution of numerical and statistical problems, *Journal of the Association for Computing Machinery* 8: 212–229.
11. Huyer, W. and Neumaier, A. (1999), Global optimization by multilevel coordinate search, *J. Global Optim*. 14(4): 331–355.
12. Janka, E. (1999), Vergleich Stochastischer Verfahren zur Globalen Optimierung. Diplomarbeit, Universität Wien.
13. Jones. D.R. (1999), The DIRECT Global Optimization Algorithm, to appear in the *Encyclopedia of Optimization*.
14. Jones, D.R., Perttunen, C.C. and Stuckman, B.E. (1993), Lipschitzian Optimization without the Lipschitz Constant, *J. Optim. Theory Appl.* 79: 157–181.
15. Kelley, C.T. (1999), *Iterative Methods for Optimization*, No. 18 in Fromtiers in Applied Mathematics. SIAM, Philadelphia.
16. Nelder, J.A. and Mead, R. (1965), A simplex method for function minimization, *Comput. J.* 7: 308–313.
17. Torczon, V. (1997), On the convergence of pattern search algorithm, *SIAM J. Optim.* 7: 1–25.
18. Yao, Y. (1989), Dynamic Tunneling Algorithm for Global Optimization, *IEEE Transactions on Systems, Man, and Cybernetics* 19(5).