CROATICA CHEMICA ACTA CCACAA **69** (3) 775–791 (1996)

ISSN-0011-1643
CCA–2340

Original Scientific Paper

A New Algorithm for Global Minimization Based on the Combination of Adaptive Random Search and Simplex Algorithm of Nelder and Mead

Miljenko Huzak[a] *and Željko Bajzer*[b]

[a]*Department of Mathematics, University of Zagreb, Bijenička 30, 10000 Zagreb, Croatia*

[b]*Department of Biochemistry and Molecular Biology, Mathematical Methods Core Facility, Mayo Foundation, Rochester MN, 55905*

Received December 21, 1995; revised February 19, 1996; accepted February 21, 1996

We propose a new general-purpose algorithm for locating global minima of differentiable and nondifferentiable multivariable functions. The algorithm is based on combination of the adaptive random search approach and the Nelder-Mead simplex minimization. We show that the new hybrid algorithm satisfies the conditions of the theorem for convergence (in probability) to global minimum. By using test functions we demonstrate that the proposed algorithm is far more efficient than the pure adaptive random search algorithm. Some of the considered test functions are related to membership set estimation method for model parameter determination which was successfully applied to kinetic problems in chemistry and biology.

INTRODUCTION

A large class of global optimization problems can be rather successfully handled by random search techniques.[1,2] This is especially true for functions which have many local minima or functions with discrete range. Although the convergence to the global minimum is in probabilistic sense guaranteed, the rate of convergence may be rather slow. On the other hand the Nelder-Mead Simplex algorithm,[3] which can be equally applied to any function (discontinuous, discrete), converges to a local minimum faster. However, for this algorithm there is no demonstrated probabilistic assurances that the global

minimum will be found. This situation motived us to combine the so called »Adaptive Random Search« (ARS) algorithm[1,4] with the Simplex algorithm. The resultant hybrid algorithm is expected to provide faster convergence then the pure ARS technique and is expected to retain the same nice property of probabilistically guaranteed convergence to global minimum. We prove the latter assertion rigorously in the Appendix of the paper.

The ARS algorithm has been compared to other algorithms for global minimization on a number of standard continuous functions.[4] Its combination with the quasi-Newton algorithm for local minimization appeared favorable[4] in comparison with other techniques. In this paper we compare pure ARS algorithm and our proposed hybrid to random-cost algorithm recently developed by Berg[5] and successfully applied to ground state conformations search for the protein metenkephalin. The comparison is performed on the illustrative function proposed by Berg with the results showing advantage of the ARS algorithm and still more remarkable advantage of our hybrid algorithm.

Recent years had witnessed a tremendous development of evolution and genetic algorithms for global optimization (see *e.g.* Ref. 6). In this paper we compare the performance of proposed hybrid algorithm to performances of various genetic algorithms on 10-dimensional Griewangk's function[7] and 20-dimensional Rastrigin's function.[8] The hybrid algorithm performed significantly better than all considered genetic algorithms on Griewangk's function but could not match the performance of most genetic algorithms for the Rastrigin's function.

The special interest for application of the proposed hybrid algorithm is the membership estimation method for parameters of nonlinear models.[9–12] This method offers comprehensive insight in parameter uncertainty regions and thus makes possible to characterize the nonuniqueness of parameter estimation. However, the method requires minimization of discrete criterion function which cannot be performed by standard minimization algorithms such as the Levenberg-Marquardt algorithm (designed only for functions with continuous range and continuous derivatives). Therefore, to make the membership estimation method workable E. Walter and his collaborators[9–12] applied the ARS algorithm described in Ref. 4. In this paper we demonstrate by numerical experiments that the proposed hybrid algorithm converges to the global minimum of criterion function much faster than the pure ARS algorithm, which in consequence makes the membership estimation method more practical.

For minimization of membership estimation criterion function we have chosen examples relevant to mathematical models in chemistry. Our first example is related to Hill's function which describes the velocity-substrate dependence in enzyme kinetics and which is the generalization of the well known Michaelis-Menten kinetic equation. The second example is the two-exponential decay model with four parameters which is ubiquitous in chemistry, biology and physics.

DESCRIPTION OF ALGORITHMS

ARS Algorithm

The Adaptive Random Search algorithm for finding the global minimum of a function can be applied to a large class of functions $f : S \to R$ where S is a subset of the d-dimensional Euclidean space R^d and the function is assumed to be sufficiently regular so that point(s) of the global minimum of f exists in S. In the following we describe the algorithm in some detail. More formal and rigorous description together with the theorem of convergence is given in the Appendix.

* Basic step. Generate n_2 random points

$$x^j = x_{sp} + \xi^j \in S, \quad j \in \{1, \ldots, n_2\} \tag{1}$$

where d components of ξ^j are independent and normally distributed with zero mean and variances $v_{j1}, \ldots v_{jd}$. In other words, ξ^j has d-dimensional normal distribution with zero mean and covariance $v = \text{diag}(v_{j1}, \ldots v_{jd})$. x_{sp} is a chosen starting point. If $f(x^j) \le f(x^{j-1})$ then $x_{\min} = x^j$, otherwise the previous $x_{\min}$ is retained. If a generated point x^j does not belong to admissible domain S, it is replaced by a border point in S, *i.e.* if the k-th coordinate $x_k^j < r_{k,\,min}$, than x_k^j is replaced by $r_{k,\min}$ or if $x_k^j > r_{k,\max}$ it is replaced by $r_{k,\max}$. The interval $[r_{k,\min}, r_{k,\max}]$ representd the given admissible range for k-th coordinate.

* Step 1. Repeath the basic step n_1 times, each time usin a smaller covariance $v_i = \text{diag}(v_{i1}, \ldots v_{id})$, $i = 1, \ldots n_1$ (in a sense that variances by coordinates are smaller; see Figure 1). x_{sp} is fixed. The number n_2 of generated points per covariance diminishes with covariance according to $n_2 = [n_3/i]$. The most successful covariance v_{opt} is the one corresponding to the final $x_{\min}$ after all points of this step were generated.

* Step 2. Repeat the basic step n_4 times with $n_2 = 1$ and covariance v_{opt}, each time choosing the current $x_{\min}$ as starting point x_{sp} (Figure 2).

* Repeat steps 1 and 2 until the minimal covariance v_{n1} was selected as the optimal covariance for n_5 times successively, or until n_6 such repetitions are completed. The final $x_{\min}$ is assumed to be the global minimum.

Clearly, positive integers n_1, n_3, n_4, n_5, n_6 are to be set in advance. Results in Ref. 4 show that certain combination of their values provides efficient solution for a wide class of optimization problems. However, numerical experiments in this study indicate that values of these parameters should be increased as dimension d of the problem increases.

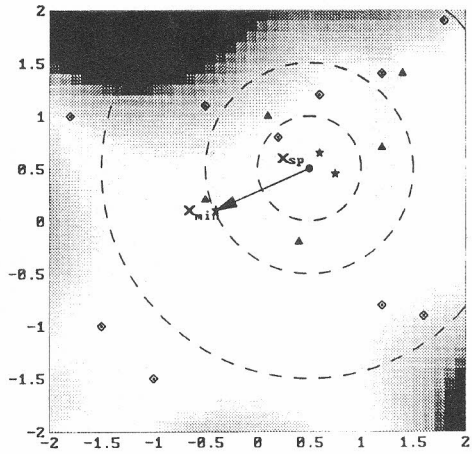

Figure 1. The first step of ARS with the parameters $n_1 = 3$, $n_3 = 10$. The picture represents the density plot of the function which should be minimized. Brighter areas represent areas with the smaller function value. Circles shows the magnitude of covariances $v_1, \ldots, v_{n_1}$ and $x_{\min}$ is the best obtained point of minimum at that step. Randomly generated points with the same covariance are represented by the same symbol.

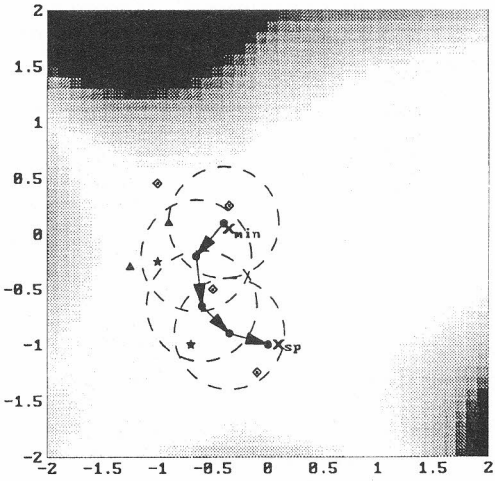

Figure 2. The second step of ARS with parameter $n_4 = 4$. Arrows show the way how the starting point change from $x_{\min}$, found at the first step of ARS, to the best point found by the second step. This point will became either the starting point x_{sp} for the first step of ARS in next iteration or the final point of the global minimum. Circles illustrates the magnitude of the fixed covariance v_{opt} used for generating random points. The same symbol represents random points generated from the same starting point.

In the actual implementation of the algorithm the initial starting point $x_{sp} = x^0$ is conveniently chosen at the central point of the domain S.[4] More precisely

$$x_k^0 = (r_{k,\max} - r_{k,\min}) / 2 , \quad k = 1, \ldots, d .$$

For the functions which have the global minimum at the central point of the domain the starting point was chosen as

$$x_k^0 = \alpha r_{k,\min} + (1 - \alpha) r_{k,\max} , \quad k = 1, \ldots, d$$

where $\alpha \in [0, 1]$ denotes uniformly distributed random number. The diminishing covariance matrices are chosen[4] to decrease in norm by factor 10^2, *i.e.*, $v_i = 0.1^2 v_{i-1}$ with $v_1 = \mathrm{diag}(r_{1,\max} - r_{1,\min}, \ldots, r_{d,\max} - r_{d,\min})^2$.

ARS + SIMPLEX Hybrid algorithm

It has been pointed out by Pronzato *et al.* that the efficiency of the ARS algorithm for differentiable functions can be greatly improved by applying quasi-Newton method for local minimizations.[4] In the context of the ARS approach the local minimization occurs when the smallest covariance v_{n_1} (in the sense described above) was chosen as v_{opt}.

We will partially adopt this strategy by introducing the Nelder-Mead simplex (NMS) algorithm[2] (sometimes also called the polytope algorithm) as a mean for faster local minimization. The NMS algorithm converges to the local minimum by advancement of simplex (or polytope) – a geometric figure in d-dimensional space which have $d + 1$ vertices. In two dimensions the simplex is a triangle. The advancement of simplex towards the region of minimum is obtained by its reflection, expansion, contraction and shrinkage.[2,13]

Generally, the NMS algorithm is not as fast as for example the quasi-Newton or the conjugate gradient algorithm, but it can be applied to non-differentiable functions. In addition, the strategy of the NMS algorithm offers the possibility of accelerating the overall global minimum search, not just an improvement of the local minimization. On the other hand the application of the pure NMS algorithm for global minimization may suffer to often of finding only the local minimum. That is the reason why we believe that the NMS minimization have to be combined with the ARS minimization. After some numerical experimentation we found an efficient combination which can be shown to converge in the probabilistic sense to a global minimum. The proof of convergence is given in the Appendix and the hybrid algorithm is described below.

The structure of the hybrid algorithm remains the same as that of the ARS algorithm. The only modification occurs in Step 2 where NMS minimization is included. More precisely the new Step 2 is:

* Step 2a. Repath the basic step n_4 times, each time generating $n_2 = d + 1$ random points with the covariance v_{opt} and x_{sp} set at the current x_{min}. Each time use the generated $d + 1$ random points to form the initial simplex and start the minimization according to the NMS algorithm.

Steps 1 and 2a are again repeated until the minimal covariance v_{n_1} was selected as the optimal variance for n_5 times in succession, or until n_6 such repetitions are completed. The final x_{min} is assumed to be the global minimum.

It is significant that the initial simplex in step 2a is defined by randomly chosen points. In this way both smaller and larger size simplexes are created enhancing the chance for NMS to find the global minimum. For the NMS algorithm we have essentially used its implementation from »Numerical Recipes«[10] adding some features regarding the stopping criterion and provision how to treat the points of the current simplex which fall outside the admissible domain. We employ the stopping criterion based on the quantities R_f and R_x.

$$R_f = 2\frac{|f(x_h) - f(x_l)|}{u_\varepsilon(|f(x_h)| + |f(x_l)|)}\,; \quad u_\varepsilon(y) = y,\ y > \varepsilon \geq 0,\ u_\varepsilon(y) = 1,\ y \leq \varepsilon \tag{2}$$

where x_h and x_l are the »highest« and the »lowest« points of the current simplex, *i.e.* $f(x_h) = \max_{1 \leq i \leq d+1} f(x_i)$ and $f(x_l) = \min_{1 \leq i \leq d+1} f(x_i)$. In the actual implementation $\varepsilon = 10^{-20}$. The quantity R_x is given by

$$R_x = \max_{1 \leq k \leq d;\ 1 \leq i,j \leq d+1} \frac{|x_{ki} - x_{kj}|}{u_0(|x_{ki}| + |x_{kj}|)} \tag{3}$$

where the function u_0 is given by (2) and x_{kj} is the k-th coordinate of j-th point of the current simplex. For the stopping criterion we first require that the function values in vertices of simplex do not mutually differ more than some prescribed tolerance ε_f, *i.e.* we require $R_f \leq \varepsilon_f$. If this is satisfied we require that the size of the simplex is sufficiently small, *i.e.* $R_x \leq \varepsilon_x$ where ε_x is a given tolerance ($\varepsilon_f < \varepsilon_x$). In other words we require that the simplex is sufficiently localized in the neighborhood of the local minimum. If such criterion cannot be met before $R_f < \varepsilon_f/10$ the procedue is terminated. The procedure is also terminated if the number of function evaluations exceeds certain maximum number.

When the NMS algorithm creates a new simplex vertex we require this point to be within the admissible domain. Therefore, if $x_{ki} < r_{k,\min}$ then we replace x_{ki} by

$$r_{k,\min} + \eta(r_{k,\max} - r_{k,\min})/10^3 \ .$$

Similarly, if $x_{ki} > r_{k,\max}$ then we replace x_{ki} by

$$r_{k,\max} - \eta(r_{k,\max} - r_{k,\min})/10^3 \ .$$

Here $\eta \in [0,1]$ denotes uniformly distributed random number, different each time when the vertex point escapes from the admissible domain.

COMPARISON OF ALGORITHMS ON TEST CASES

In the absence of any rigorous theoretical argument why our hybrid algorithm should converge faster to global minimum than the ARS algorithm we present some numerical examples which clearly demonstrate faster convergence.

Differentiable functions

We will first search the global minimum of the function

$$f(x) = \sum_{k=1}^{d} [A(x_k^2 - B)^2 + Cx_k], \quad x = (x_1, \ldots, x_d), \quad x_k \in [-1,1] \tag{4}$$

with $A = 10$, $B = 0.25$, $C = 0.1$, which was used by Berg for illustrating his random-cost approach for optimization.[5] For the chosen parameters there are 2^d local minima in a given domain.

In assessing the efficiency of ARS and hybrid algorithms we have performed 50 minimizations for a given combination of parameters n_1, n_3, n_4, n_5, n_6 and a given algorithm. These minimizations differ only by the choice of a seed for the random number generator, *i.e.* generated points in each minimization are different. The efficiency of the algorithm is measured by i) the average number (for 50 trials) of function evaluations N with standard deviation s; ii) the median number M of function evaluations; iii) the percentage P of successful findings of the global minimum, and iv) by the standard deviation from the value of the function at the »exact« global minimum x_{exact} (found by analytical methods): $s_f = \sqrt{\sum_{i=1}^{l} [f(x_g^i) - f(x_{\text{exact}})]^2 / l}$, $l \leq 50$ where x_g^i are points of the global minimum found by the algorithm. We have

TABLE I

Results of 50 trials in minimization of function (4). Achieved efficiency was $P = 100\%$, except for the case marked with * where $P = 96\%$. For the hybrid algorithm tolerances were set to $\varepsilon_x = 10^{-3}$, $\varepsilon_f = 10^{-7}$.

d	Algorithm(n_1, n_3, n_4, n_5, n_6)	M	N	s	s_f
2	ARS(6,85,25,5,40)	9281	9281	0	$5 \cdot 10^{-16}$
2	Hybrid(3,30,20,1,1)	1607	1663	78	$9 \cdot 10^{-11}$
3	ARS(6,300,300,5,150)	127901	127901	0	$8 \cdot 10^{-11}$
3	Hybrid(3,75,25,1,1)	3648	3686	119	$3 \cdot 10^{-10}$
4	ARS*(6,900,900,5,450)	1086751	1086751	0	10^{-11}
4	Hybrid(3,75,70,1,1)	16418	16267	439	$4 \cdot 10^{-10}$

chosen parameters n_1, n_3, n_4, n_5, n_6 so that $P = 100\%$ or somewhat less and that s_f is smaller than 10^{-9}.

The results of our numerical experiments for $d = 2, 3, 4$ are shown in Table I. The efficiency of the hybrid algorithm is remarkably greater than the efficiency of the ARS algorithm. Furthermore the difference in efficiency becomes larger as the dimension of the problem increases. Comparing medians for the number of function ealuations we see that in case $d = 2$ the hybrid algorithm is roughly 6 times more efficient; for $d = 3$ it is over 30 times more efficient and for $d = 4$ it is over 30 times more efficient. Here it should be noted that computing time for computations other than the function evaluation in NMS algorithm is relatively small so that the number of function ealuations is a good measure of the actual computing time.

From the results of Table I it is clear that for the hybrid algorithm there is no necessity to set n_3 and n_6 larger than 1. Essentially, only n_3 and/or n_4 have to be increased as the dimension of the problem increases. For the ARS algorithm the number of iterations n_6 had to be increased as well. The accuracy in determining the minimum value of the function in hybrid algorithm is easily controlled by tolerance ε_f. By decresing ε_f a substantial improvement can be achieved with relatively small increase in the number of function evaluations.

Berg[5] has applied his random-cost algorithm to function (4) only for $d = 2$. In average his algorithm required roughly 300,000 function evaluations to determine the global minimum, which greatly exceeds the number of function evaluations required by the ARS algorithm for approximately the same accuracy. One can imagine that the large number required by the random-cost algorithm can be somewhat reduced by better tunning of the procedure, but on the basis of presented results it is difficult to believe that this procedure might become more efficient than the ARS algorithm.

In the next example we consider two differentiable functions for which the search of global minimum is much more difficult. The Griewangk's function:[7]

$$f_G(x) = \sum_{k=1}^{10} \frac{x_k^2}{4000} - \prod_{k=1}^{10} \cos\left(\frac{x_k}{\sqrt{k}}\right) + 1, \quad x_k \in [-512,512] \tag{5}$$

where $x = (x_1, \ldots, x_{10})$, achieves the global minimum $f_G(x_0) = 0$ at $x_0 = (0, \ldots, 0)$ and exhibits a large number of local minima. Four of those are close to the global minimum: $f(x) \approx 0.0074$ at $x \approx (\pm\pi, \pm\pi\sqrt{2}, 0, \ldots, 0)$.

After 30 trials with $(n_1, n_3, n_4, n_5, n_6) = (5, 600, 400, 5, 100)$, $\varepsilon_x = \varepsilon_f = 10^{-6}$, the hybrid algorithm recovered the global minimum 11 times. For each trial the algorithm was stopped when approximately 4×10^5 function evaluations were performed. (Due to simplex minimization which was not immediately stopped the actual number of function evaluations was $4 \times 10^5 + n$, $0 \leq n \leq 2850$).

This result comparres favorably with the results of nine genetic algorithms tested on the same function by Gordon and Whitley.[8] The most successful genetic algorithm recovered the global algorithm 7 times and other eight algorithms 6, 3, 3, 3, 1, 1, 0 and 0 times respectively. In each trial the search was terminated after 1000 generations with population of size fixed to 400 leading to 4×10^5 function evaluations.

Highly multimodal Rastrigin's function:[8]

$$f_R(x) = \sum_{k=1}^{20} x_k^2 - 10\cos(2\pi x_k) + 200, \quad x_k \in [-5.12,5.12] \tag{6}$$

where $x = (x_1, \ldots, x_{20})$ has the global minimum $f_R(x_o) = 0$ at $x_0 = (0, \ldots 0)$. When we tested the hybrid algorithm on that function (with the same parameters as for the Griewangk's function) the result of zero success in 30 trials showed the algorithm's weakness. Three of nine genetic algorithms tested by Gordon and Whitley[8] were also unsuccessful, but six algorithms succeeded to recover the global minimum 2, 10, 13, 23, 23 and 24 times respectively.

Nondifferentiable functions

Here we consider the class of functions for which the search of global minimum is much more difficult task than for the differentiable functions and obviously none of the classical methods based on knowledge of gradient can be used. The functions we consider are not mathematical peculiarities but do appear in problems of model parameter esimation based on the so

called membership estimation method.[9] The aim of this method is to find all model functions (defined by certain number of parameters) which are consistent with noisy data in a sense that the curve associated with each of these functions »passes through data error bars«. The following example explains the essence of the method.

Reaction velocity as dependent on substrate concentration in a simple enzymatic reaction kinetics can be described by Hill's model function:

$$v = h(S; x) = \frac{v_{\max} S^c}{K^c + S^c}, \quad x = (v_{\max}, K, c) \tag{7}$$

where $v > 0$ is the reaction velocity with its maximal value $v_{\max} > 0$, $S > 0$ is the substrate concentration, $K > 0$ is the half-saturation concentration and $c \geq 1$ is the degree of sigmoidicity related to possible cooperativity. Reaction velocity is measured as a function of substrate concentration and these data are used to determine the values of parameter vector x. For the sake of this example we will assume the following »experimental data set«;

$$s_i = 10^{-1+0.2(i-1)}, \quad i = 1, \ldots, \nu, \quad \nu = 11$$

$$h_i = h(s_i; x) + \sigma\rho_i, \quad x = (1, 2, 1.5), \quad \sigma = 0.25$$

where ρ_i is uniformly distributed random variable on [–1,1] and σ determines the »error bars«. The parameter vector is determined by minimizing the following criterion function:

$$C(x) = -\frac{m(x)}{\nu} \tag{8}$$

where $m(x)$ is the number of s_i for which $h_i - \sigma < h(s_i; x) < h_i + \sigma$. More formally $m(x) = \text{card}\{s_i | h(s_i; x) \in (h_i - \sigma, h_i + \sigma)\}$. The essence of the membership estimation method is to find the set of all admissible parameter vectors x for which the criterion function $C(x)$ achieves its global minimum. Thus, the method requires search for global minimum of $C(x)$ which cannot be smaller than –1 nor it can be larger than zero. the values of $C(x)$ are obviously rational numbers and the function is not continuous.*

---

* For that reason we had to change the stopping criterion in NMS (see above). Quantity R_f in (2) is now replaced by $R_f = |f(x_h) - f(x_l)|$ and the stopping criterion is as follows: The procedure terminates when $R_f = 0$ or the maximum number of function calls is achieved. In case $R_f = 0$ and $f(x_l) \neq 0$ the additional requirement for termination is as before $R_x \leq \varepsilon_x$ or if the number of such cases exceeds certain maximal number n_0. In case of $R_f = f(x_l) = 0$ the procedure is terminated when the number of such cases exceeds n_0. Finally the procedure is terminated instantly when the value of the function reaches –1.

The ARS algorithm has been applied for minimization of discrete criterion (8) extensively by E. Walter and his collaborators[9–12] for various model functions. Unfortunately in these papers we could not find any reference on efficiency of the ARS algorithm. Here we compare the efficiency of the ARS and the hybrid algorithm for minimization of $C(x)$ with Hill's model function. In Table II we displayed the number N_s of successful findings of the global minimum (–1) for 50 different realizations of data set. We also displayed the total number N_c, of criterion function evaluation for those class when both algorithms have found the global minimum. The advantage of the hybrid algorithm is remarkable both in the number of successful findings and in number of criterion function evaluations.

TABLE II

(Hill's model) Results for 50 simulated data sets. The tolerance for the hybrid algorithm was $\varepsilon_x = 10^{-5}$ and the termination number was set to $n_0 = 2$.

Algorithm(n_1, n_3, n_4, n_5, n_6)	N_s	N_c
ARS(5,100,100,50,100)	8	59427
Hybrid(5,100,100,50,100)	25	10296

Next example requires minimization in 4-dimensional space. The model function is the two-exponential decay model:

$$f(t; x) = x_1 e^{x_2 t} + x_3 e^{-x_4 t}, \quad x = (x_1, x_2, x_3, x_4) \ . \tag{9}$$

»Experimental data set« is given by

$$t_i = 10^{-1+0.2(i-1)}, \quad i = 1, \ldots, v, \quad v = 15$$

$$f_i = f(t_i; x) + \sigma\rho_i, \quad x = (0.1, 2, 0.9, 3), \quad \sigma = 0.25$$

where ρ_i is the same as above. The minimization of criterion function given by (8) (with $h(s_i; x)$ replaced by $f(t_i; x)$ and h; by f_i) for 50 data set realizations, yielded results displayed in Table III. Again the hybrid algorithm is significantly more efficient although not as much as in the case of Hill's three dimensional model.

TABLE III

(Two-exponential decay model) Results for 50 simulated data sets. The tolerance for the hybrid algorithm was $\varepsilon_x = 10^{-4}$ and the termination number was set $n_0 = 2$.

Algorithm(n_1, n_3, n_4, n_5, n_6)	N_s	N_c
ARS(8,400,200,50,80)	4	115972
Hybrid(8,400,200,50,80)	10	93663

CONCLUSION

We have shown that the proposed hybrid algorithm based on combination of the adaptive random search and the Nelder-Mead simplex procedure is much more efficient in search for the global minimum than the pure adaptive random search. At the same time the hybrid algorithm shares the property of convergence to global minimum (in probabilistic sense) with the adaptive random search algorithm. We performed the number of tests on differentiable and nondifferentiable functions with the results clearly in favor of the hybrid algorithm. Based on our tests the hybrid algorithm can be certainly recommended for minimizations of discrete functions involved in membership estimation method. Preliminary comparison with genetic algorithms indicated that the hybrid algorithm could be considered as a good alternative.

APPENDIX

Convergence Theorem

Let S be a subset of the Euclidean space R^d and $f : S \to R$ be a measurable function (with respect to Lebesgues' measure λ[14]) defined on S. We suppose that S is enough regular so that f has point(s) of the global minimum in S. The aim is to find an algorithm which gives an acceptable approximation of any point of the global minimum.

Random search algorithm for optimization is defined as any algorithm which has the following form:

choose x^0
set $k \leftarrow 0$
while stopping criterion is not satisfied **do**
    generate the random value ξ_k from the distribution μ_k
    set $x^{k+1} = G(x^k, \xi_k)$
    choose μ_k
    set $k \leftarrow k + 1$
end

where $(\mu_k; k \in N_0)$ is a sequence of probabilities on measurable space $(R^d, \mathscr{B}(R^d))$ ($\mathscr{B}(R^d)$ is a σ-algebra of Borel sets in S (Ref. 14) and $G : S \times R^d \to S$ is a measurable function with the property that $f(G(x, \xi)) \leq f(x)$ for all $x \in S, \xi \in R^d$ and $f(G(x, \xi)) \leq f(\xi)$ for all $x, \xi \in S$. The sequence of points $(x^k; k \in N_0)$ in S is a sequence of approximative global minimum points if that algorithm converges to the global minimum.

If S is not discrete space and f is not function with discrete range, the family of distributions $(\mu_k; k \in N_0)$ should not be discrete too. Otherwise, the point of global minimum will never be reached by the random search algorithm. On the other hand, if these conditions are fulfilled but f is almost continuous functions with discountinuity at the point of the global minimum x_0 with property that there exists $\lim_{x \to x_0} f(x) = L$ and $L > f(x_0)$, $\inf f$ will never be reached. The conclusion is that any random search algorithm tries to calculate not the global minimum but the essential infimum essinf f of the function f:

$$\operatorname{essinf} f := \inf\{t : \lambda(\{x \in S : f(x) < t\}) > 0\} .$$

Let, for $\varepsilon > 0$, $R_\varepsilon := \{x \in S : f(x) < \operatorname{essinf} f + \varepsilon\}$. The random search algorithm is said to converge to the global minimum (in probability) if for any $\varepsilon > 0$, $\lim_{k \to \infty} P\{x^k \in R_\varepsilon\} = 1$.[3]

Solis and Wets[3] proved the following theorem.

Theorem. Let f and S be as above and the random search algorithm be given with the sequence $(\mu_k; k \in N_0)$ of distributions which have the following property:

$$\forall\ B\ \in \mathscr{B}(S), \lambda(B) > 0 \Rightarrow \prod_{k=0}^{\infty} (1 - \mu_k(B)) = 0 .$$

Then the random search algorithm converges to the global minimum.

Algorithms

Let $S \subset R^d$ be a compact cube, $S = \prod_{j=1}^{d} [r_{j,\min}, r_{j,\max}]$, $v_i = \operatorname{diag}(v_{i1}, \ldots v_{id})$, $i = 1, \ldots, n_1$, be a sequence of diagonal matrices, (each representing a covariance matrix) such that $v_{ij} > v_{i+1,j}$ for all $j = 1, \ldots d$, $i = 1, \ldots n_1 - 1$ and $v_1 = \operatorname{diag}(r_{1,\max} - r_{1,\min}, \ldots, r_{d,\max} - r_{d,\min})^2$. Shortly, $v_1 > v_2 > \ldots > v_{n_1} = v_{\min}$. Let $\pi_S : R^d \to S$ be a coordinate projector onto S. The ARS algorithm[1,2] can be formulated as follows:

choose x^0

set $k \leftarrow 0$

put $f_{\min} \leftarrow f(x^0)$, $v_{\text{opt}} \leftarrow v_{\min}$

while max. no. iter. n_6 is not achived or $v_{\min}$ is not chosen n_5 times successively **do**

{The first step:}

$x_{sp} \leftarrow x^k$

for $i \leftarrow 1$ **to** n_1 **do**

$n_2 \leftarrow [n_3/i]$

for $j \leftarrow 1$ **to** n_2 **do**

generate ξ from $v_i(dx) = \dfrac{1}{\sqrt{(2\pi)^d \det v_i}} e^{-\frac{1}{2}(v_i^{-1}x|x)}\,dx$

$\xi^k = \pi_S(x_{sp} + \xi)$

if $f(\xi^k) < f_{\min}$ **then**

$x^{k+1} = \xi^k$

$f_{\min} \leftarrow f(\xi^k)$

$v_{opt} \leftarrow v_i$

else $x^{k+1} = x^k$

$k \leftarrow k + 1$

{The second step:}

for $j \leftarrow 1$ **to** n_4 **do**

generate point ξ from $v(dx) = \dfrac{1}{\sqrt{(2\pi)^d \det v_{opt}}} e^{-\frac{1}{2}(v_{opt}^{-1}x|x)}\,dx$

$\xi^k = \pi_S(x^k + \xi)$

if $f(\xi^k) < f_{\min}$ **then**

$x^{k+1} = \xi^k$

$f_{\min} \leftarrow f(\xi^k)$

else $x^{k+1} = x^k$

$k \leftarrow k + 1$

end

It should be noted that ξ^k has truncated normal distribution μ_k.

The form of the hybrid ARS + SIMPLEX algorithm can be formulated as follows:

choose x^0

set $k \leftarrow 0$

put $f_{\min} \leftarrow f(x^0)$, $v_{\text{opt}} \leftarrow v_{\min}$

while max. no. iter. n_6 is not achived or $v_{\min}$ is not chosen n_5 times successively **do**

{The first step:}

$x_{sp} \leftarrow x^k$

for $i \leftarrow 1$ **to** n_1 **do**

$n_2 \leftarrow [n_3/i]$

for $j \leftarrow 1$ **to** n_2 **do**

generate ξ from $v_i(dx) = \dfrac{1}{\sqrt{(2\pi)^d \det v_i}} e^{-\frac{1}{2}(v_i^{-1}x|x)}\,dx$

$\xi^k = \pi_S(x_{sp} + \xi)$

if $f(\xi^k) < f_{\min}$ **then**

$x^{k+1} = \xi^k$

$f_{\min} \leftarrow f(\xi^k)$

$v_{opt} \leftarrow v_i$

else $x^{k+1} = x^k$

$k \leftarrow k + 1$

{The second step:}

for $j \leftarrow 1$ **to** n_4 **do**

generate d + 1 points $\xi_l, \ldots \xi_{d+1}$ from $v(dx) = \dfrac{1}{\sqrt{(2\pi)^d \det v_{opt}}} e^{-\frac{1}{2}(v_{opt}^{-1}x|x)}\,dx$

call *simplex*$(x^k + \xi_l; l = 1, \ldots, d + 1)$

if *simplex* finds a better point $x^k + \xi_l \in S$ **then**

$\xi^k = x^k + \xi_l$

$x^{k+1} = \xi^k$

$f_{\min} \leftarrow f(\xi^k)$

else $x^{k+1} = x^k$

$k \leftarrow k + 1$

end

Convergence of algorithms

Both algorithms fulfill the condition for global convergence noted in Theorem (Solis & Wets[3]). Namely, let $(\mu_{k_j}; j \in N)$ be a subsequence of the distributions $(\mu_k; k \in N_0)$ which were used in the first phase of the algorithm all with the same covariance v_1, but with different mean parameters. Let B be any Borel set in S with positive Lebesgues measure, $B^c = R^d \backslash B$. The function $F : S \to \mathrm{R}$, defined by $F(y) := \dfrac{1}{\sqrt{(2\pi)^d \det v_1}} \int_{B^c} e^{-\frac{1}{2}(v_1^{-1}(x-y)|x-y)}\, dx$ is continuous with compact domain. So, there exists the point of maximum y_0 of F in S. Therefore, for any $j \in N$, $\mu_{k_j}(S \backslash B) \le F(y_0) < 1$, which implies

$$0 \le \prod_{k=0}^{\infty} (1 - \mu_k(B)) \le \prod_{j=1}^{\infty} (1 - \mu_{k_j}(B)) = \prod_{j=1}^{\infty} \mu_{k_j}(S \backslash B) \le \lim_n F(y_0)^n = 0 \ .$$

Acknowledgment. – We are thankful to Dr. Frank G. Prendergast for his interest and enthusiastic support of our research in optimization algorithms. We are also grateful to Dr. Armando Manduca and to one of the referees for advices regarding genetic algorithms. Supported in part by GM34847.

REFERENCES

1. S. W. Masri, G. A. Bekey, and F. B. Safford, *A Global Optimization Algorithm Using Adaptive Random Search, Applied Mathematics and Computation* **7** (1980) 353–375.
2. F. J. Solis and R. J. B. Wets, *Minimization by Random Search Techniques, Mathematics of Operations Research*, **6** (1981) 19–30.
3. J. A. Nelder and R. Mead, *A Simplex Method for Function Minimization, Computer Journal* **7** (1965) 308–313.
4. L. Pronzato, E. Walter, A. Venot, and J-F. Lebruchec, *A General Purpose Global Optimizer: Implementation and Application, Mathematics and Computers in Simulation* **24** (1984) 412–422.
5. B. A. Berg, *Locating Global Minima in Optimization Problems by a Random-Cost Approach, Nature* **361** (1993) 708–710.
6. H. P. Schwefel, *Evolution and Optimum Seeking*, Wiley, New York, 1995.
7. A. O. Griewangk, *Generalized Descent for Global Optimization, JOTA*, **34** (1981) 11–39.
8. V. S. Gordon and D. Whitley, *Serial and Parallel Genetic Algorithms as Function Optimizers, The 5th Int. Conf. on Genetic Algorithms, Urbanna-Champaign 1993*, S. Forrest (Ed.), Morgan-Kaufmann, 1993, pp. 177–183.
9. H. Lahanier, E. Walter, and R. Gomeni, *OMNE: A New Robust Membership-Set Estimator of Nonlinear Models, Journal of Phamacokinetics and biopharmaceutics* **15** (1987) 203–219.

10. E. Walter and H. Piet-Lehanier, *Bounded-Error Alternative to Least Squares for Estimating the Parameters of a Nonlinear Model from Short Records, Biomedical Modelling and Simulation*, J. Eisenfeld, D. S. Levine (Eds.), Baltzer Pub. Co., 1989, pp. 95–99.
11. E. Walter, H. Piet-Lehanier, and J. Happel, *Estimation of Non-uniquely Identifiable Parameters via Exhaustive Modeling and Membership Set Theory, Mathematics and Computers in Simulation* **28** (1986) 479–490.
12. E. Walter and H. Piet-Lehanier, *Estimation of the Parameter Uncertainty Resulting from Bounded-Error Data, Mathematical Biosciences* **92** (1988) 55–74.
13. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes*, Cambridge Univ. Press, 1986.
14. D. L. Cohn, *Measure Theory*, Birkhäuser, Boston, 1980.

SAŽETAK

Novi algoritam za globalnu minimizaciju zasnovan na kombinaciji prilagodljivog slučajnog pretraživanja i simpleks algoritma Neldera i Meada

Miljenko Huzak i Željko Bajzer

Predlažemo novi algoritam opće namjene za određivanje globalnih minimuma derivabilnih i nederivabilnih funkcija s više varijabli. Algoritam se zasniva na kombinaciji prilagodljivog slučajnog pretraživanja i simplex minimizacije Neldera i Meada. Pokazujemo da novi hibridni algoritam zadovoljava teorem o konvergenciji (po vjerojatnosti) prema globalnom minimumu. Pomoću probnih funkcija pokazujemo da je predloženi algoritam znatno efikasniji od čistog prilagodljivog slučajnog pretraživanja. Neke od razmatranih probnih funkcija su povezane s metodom ocjene skupa članstva koja se koristi za određivanja modelskih parametara i koja se pokazala uspješnom u primjeni na kinetičke probleme u kemiji i biologiji.