Engineering Optimization
Vol. 00, No. 00, Month 200x, 1–18

RESEARCH ARTICLE

Hybrid Optimization using an Evolutionary Strategy and Surrogate Assisted Local Search

Jeff S. Baggett[a*] and Brian E. Skahill[b]

[a] *University of Wisconsin - La Crosse, La Crosse, WI, USA*; [b] *US Army Engineer Research and Development Center, Vicksburg, MS, USA*

(*October 15, 2010*)

We present a new derivative-free hybrid algorithm for global optimization of expensive black box functions. The algorithm uses an evolution strategy for global search. Convergence toward local minima is accelerated by including a local search individual in each generation. The local search individual is computed by extracting derivative information from a radial basis function approximation to the objective function interpolated from previously evaluated points in the evolutionary strategy. This hybrid approach does not require artificial or user-defined switching between global and local search. Numerical results are presented on mathematical test problems from the optimization literature and for a small dimensional conceptual watershed model from hydrology.

Keywords: Hybrid algorithms; Evolution strategies; Quasi-Newton method

1. Introduction

Evolution strategies are known to be reliable but expensive for approximating global optima particularly for multi-modal fitness functions. A large population size is required for the strategy to explore the real parameter space, but this slows local convergence. In contrast, classical gradient-based algorithms are exploitative in nature and converge quickly to local minima, but they are not good at finding the global minimum.

Many algorithms attempt to accelerate the convergence of the evolutionary strategy, or other population-based search method, by either switching to a local, usually gradient-based, search at some user-defined threshold or by applying some local search operator at every generation so that the global and local searches are interwoven. Examples of the former strategy include switching from particle swarm optimization to sequential quadratic programming at a user defined threshold (Min *et al.* 2007) and similarly switching from a genetic algorithm to a Levenberg-Marquardt algorithm (Peters *et al.* 2010). Examples

*Corresponding author. Email baggett.jeff@uwlax.edu

ISSN: 0305-215X print/ISSN 1029-0273 online

DOI: 10.1080/0305215YYxxxxxxx
http://www.informaworld.com

of interwoven strategies include incorporation of an extra individual at each generation of an evolution strategy calculated from an approximated Newton step (Woo *et al.* 2004, Tahk *et al.* 2007, 2009), the use of a discrete gradient operator to improve the best individual in each generation of an evolution strategy (Abbas *et al.* 2003), and a hybrid genetic algorithm that uses a quasi-Newton step to attempt to improve the fitness of every individual at each iteration (Renders and Flasse 1996).

Another approach taken to lessen the number of expensive objective function evaluations in evolutionary strategies and other population-based algorithms is to use function approximation models as surrogates for the objective function. Typically, estimated function values are used to screen offspring and the more expensive objective function is evaluated only at the most promising offspring (Regis and Shoemaker 2004, Kern *et al.* 2004). In this paper we propose to use similar function approximation models to approximate derivative information for a local search that will be interwoven with the global evolutionary strategy. In fact, the same function approximation model can be used to approximate derivatives for the local search and as a surrogate to screen objective function values for the global search, though we do not report on that here.

We follow essentially the same framework as the hybrid evolutionary strategy first outlined in (Woo *et al.* 2004) and subsequently improved in (Tahk *et al.* 2007, 2009). In that algorithm, a standard evolutionary strategy is used to advance the population at each generation. Additionally, an individual called the gradient individual is propogated alongside the evolving population. The gradient individual is calculated by making a Newton update from the gradient individual of the previous generation or from the fittest point of the current generation. The gradient is estimated from a least squares finite difference approximation and the Hessian is iteratively approximated by one-dimensional finite difference updates using the previously evaluated points in the population. The gradient individual hybrid approach has been shown to improve convergence of the standard evolutionary strategy on a small number of mathematical test problems. One drawback to the algorithm is that the individuals in the evolving population are selected symmetrically about the estimated minimum of the current generation: $x_{\min} \pm \Delta x$. Selecting the population symmetrically increases the accuracy of the finite difference discretizations of the derivatives, however the symmetric population seems to interfere (Baggett and Skahill 2010) with the convergence of the covariance matrix adaptation variation of the evolutionary strategy (CMAES due to Hansen and Ostermeier 1996). While we have not determined exactly how the symmetrized population interferes with adaptation in CMAES, we think it is likely related to the covariance matrix updating and/or the step length updating within that algorithm. Another drawback to the symmetrized finite difference hybrid algorithm is that the minimum population size for the evolutionary strategy is $2n$, where n is the dimension of the search space.

Our proposed algorithm, however, differs from the gradient individual hybrid evolution strategy (Woo *et al.* 2004, Tahk *et al.* 2007, 2009) in two important respects: first, instead of using finite differences to approximate local derivative information we fit a local function approximation model to previously evaluated points and differentiate the model analytically, and second, the evolution strategy is the covariance matrix adaptation evolution strategy (CMAES) (Hansen and Ostermeier 1996). The use of the surrogate model to approximate the derivatives allows the estimation of derivatives without making any modifications to the basic structure of the evolutionary strategy. In fact, this approach allows for hybridization of virtually any population-based search algorithm. In this article we will focus on hybridization of CMAES which is known to be particularly effective at approximating global optima. CMAES is especially effective at locating global optima

when used in conjunction with a population doubling strategy as described in Auger and Hansen (2005). More recently, a two population restart strategy in which one population grows exponentially larger while the other is maintained at a small size has shown promise (Hansen 2009), but is not explored here.

The derivative estimation method will be explained first and it will be shown how the approximated derivatives are used to perform local search by calculating a "local search individual" in each generation. Next the hybrid algorithm will be summarized. We have implemented the hybrid algorithm in CMAES, and it will be shown how the new hybrid algorithm performs on a small suite of test functions in 10 and 30 dimensions. Finally, we briefly demonstrate the calibration of a conceptual watershed model using the hybrid algorithm.

2. Local Search using Radial Basis Functions

The foundation of this optimization algorithm is an evolution strategy in which new offspring are produced at each generation by recombination and mutation (see Figure 2). The objective function is evaluated at each of these offspring and the fittest offspring are selected as parents for the next generation. In the hybrid approach, an additional individual, called the local search individual, is propagated by a different mechanism each generation. Tahk *et al.* (2007) refer to this additional individual as the gradient individual, but we call it the local search individual since in practice it could be the result of *any* local search that is a function of the previously evaluated points. The current local search individual, x^t_{ls}, is either the fittest offspring of the current generation (individual with lowest function value) or the local search individual from the previous generation. From information gathered during the evolution of the population the first and second derivatives of the objective function are estimated at x^t_{ls} and used to perform an update of the local search individual which is hopefully moved closer to a stationary point.

As an evolution strategy proceeds, it typically does not use the previously evaluated points beyond the current generation; however, in our hybrid strategy we store the last N points and their evaluated functions values in a database. In practice, if the objective function is very expensive to evaluate, we might use *all* of the previously evaluated points. To update the local search individual we use a k-nearest neighbor local function approximation of the objective function using the k nearest neighbors (Euclidean distance) of x^t_{ls} in the database to construct a cubic radial basis function (RBF) approximation:

$$s(x) = \sum_{i=1}^{k} w_i \phi\left(\|x - x_i\|_2\right) + p(x), \quad x \in \mathbb{R}^n \tag{1}$$

where $x_i, i = 1, 2, \ldots, k$ are the k nearest neighbors of x^t_{ls} in the n-dimensional search space, p is in Π^n_2 (the linear space of polynomials in n variables of degree less than or equal to 2), and $\phi(r) = r^3$. Cubic radial basis functions were selected not only for their simplicity and differentiability, but also because they have been used successfully as surrogate models for pre-evaluating function values to lessen the number of function evaluations required by an evolution strategy (Regis and Shoemaker 2004).

Define the matrix $\Phi \in \mathbb{R}^{k\times k}$ by

$$(\Phi)_{ij} := \phi\left(\|x_i - x_j\|_2\right), \quad i, j = 1, \ldots k. \tag{2}$$

Let $\hat{n} = (n+1)(n+2)/2$ be the dimension of Π_2^n, let $p_1, \ldots, p_{\hat{n}}$ be a basis of this linear space, and define the matrix $P \in \mathbb{R}^{k \times \hat{n}}$ as follows:

$$P_{ij} := p_j(x_i), \quad i = 1, \ldots, k; j = 1, \ldots, \hat{n}. \tag{3}$$

In this model, the RBF that interpolates the points $(x_1, f(x_1)), \ldots (x_k, f(x_k))$ is obtained by solving the system

$$\begin{pmatrix} \Phi & P \\ P^T & 0 \end{pmatrix} \begin{pmatrix} w \\ c \end{pmatrix} = \begin{pmatrix} F \\ 0_{\hat{n}} \end{pmatrix} \tag{4}$$

where $F = (f(x_1), \ldots, f(x_k))^T, w = (w_1, \ldots, w_k) \in \mathbb{R}^k$ and $c = (c_1, \ldots, c_{\hat{n}})^T \in \mathbb{R}^{\hat{n}}$. Powell (1992) gives sufficient and necessary conditions for the system above to be uniquely solvable, but in practice the real issue can be that the coefficient matrix above becomes ill-conditioned. However, we have found that simply rescaling and shifting the points $x_1, \ldots, x_k$ so that they all lie in $[-1, 1]^n$ is usually sufficient to address this issue.

Once the RBF, $s(x)$, has been determined by Eq.(1), then $s(x)$ is differentiated analytically to determine approximations to the gradient and Hessian of the objective function, $f(x)$. For the gradient vector, g, evaluated at the local search individual, x_{ls}^t:

$$g_i = \left(\nabla f(x_{\text{ls}}^t)\right)_i = \frac{\partial}{\partial x_i} f(x_{\text{ls}}^t) \approx \left(\nabla s(x_{\text{ls}}^t)\right)_i = \frac{\partial}{\partial x_i} s(x_{\text{ls}}^t), \quad i = 1, \ldots, n \tag{5}$$

For the Hessian matrix, H, evaluated at the gradient individual, x_{ls}^t:

$$H_{ij} = \left(H(x_{\text{ls}}^t)_{ij}\right) = \frac{\partial^2}{\partial x_i \partial x_j} f(x_{\text{ls}}^t) \approx \frac{\partial^2}{\partial x_i \partial x_j} s(x_{\text{ls}}^t), \quad i, j = 1, \ldots, n \tag{6}$$

Similar techniques for derivative approximation are routinely used in the numerical solution of partial differential equations using so-called meshless methods. Moreover, such approximations can be spectrally accurate (faster than polynomial in the grid size) depending on the selection of interpolation points (Fornberg *et al.* 2009).

Once the offspring and their function values from the current generation have been appended to the database, we construct the RBF as above and determine the gradient and Hessian approximated at the current local search individual x_{ls}^t. Finally, a new local search individual is found by the standard update:

$$x_{\text{ls}}^{t+1} = x_{\text{ls}}^t - H^{-1} g. \tag{7}$$

The new local search individual is then added to the current generation of offspring for possible selection. When the population, and hence the points in the database, are sufficiently close to a minimum then the gradient and Hessian can be accurate and can yield fast local convergence to the minimum. However, when the evolving population is far from a local minimum, then the gradient and Hessian tend to be inaccurate and the update does not lead to a minimum. In such cases, x_{ls}^{t+1}, is typically not selected as an offspring by the evolutionary strategy and thus does no harm to convergence of the gloal search.

This method of gradient-based search is similar to a quasi-Newton method, but in quasi-Newton methods the first derivatives of the function are usually available (or ap-

proximated by centered finite differences), and the Hessian matrix is updated iteratively with information from new function and gradient evaluations. In fact, the original evolution strategy hybrid approaches of Tahk *et al.* (2007, 2009) used the quasi-Newton method to advance their gradient individual. In contrast, in this proposed approach, the Hessian matrix is completely recomputed at each generation from a local approximation to the objective function. In the next subsection we give a synthetic numerical comparison to demonstrate that the numerical accuracy of our method and that of Tahk *et al.* (2009) are comparable. The main difference in the methods is that, due to the use of the function approximation model for derivative calculation, our algorithm does not require symmetrizing the population and thus can be used to estimate derivatives and add local search to any population-based search algorithm.

The local search technique proposed here could easily be modified so that the local search individual is not simply the approximate Newton point. For instance, because we have the local function approximation available, the local search individual could be the result of a trust region search of the radial basis function along the Newton update direction similar to the approach described by Wild *et al.* (2008). In fact, any kind of local search algorithm that utilized primarily the previously collected points could be used to find the local search individual.

2.1. *A numerical comparison*

To demonstrate the capability of the radial basis function approach to approximating derivatives the 10-dimensional generalized Rosenbrock function (see Table 2) is used as a test-case. To facilitate a direct comparision with the Hessian approximation approach described by Tahk *et al.* (2009) the following synthetic numerical experiment was conducted.

The global minimizer of the Rosenbrock function is the point $\mathbf{1}$ consisting of all ones. A synthetic sequence of points was selected that converges toward the global minimizer at a constant rate:

$$x^{(i)} = \mathbf{1} + 0.01\frac{100-i}{100}(1,2,3,4,5,6,7,8,9,10), \qquad i = 1,\ldots,100 \tag{8}$$

At each iteration a population, from a normal distribution as in an evolution strategy, of $\lambda = 80$ points is generated around $x^{(i)}$ in two steps. In the first step, $\mu = 40$ points are generated by:

$$x = x^{(i)} + 0.01z, \tag{9}$$

where z is selected from the 10-dimensional standard normal distribution with mean 0 and variance 1. In the second step, 40 additional points are generated by reflecting the first 40 points through the center point $x^{(i)}$ - this center point will serve as the linearization point for each iteration and the method proposed by Tahk *et al.* (2009) utilizes the reflected points to produce higher order approximations to the gradient and Hessian. The Rosenbrock function is evaluated at the 81 points and the points and function values are stored in a database for use by our radial basis function approach.

We do not present the details of the approach of Tahk *et al.* (2009) here, but at each iteration the gradient is approximated by a least squares fit of the centered finite differences of the symmetric pairs through the linearization point. Moreover, the inverse Hessian is sequentially updated with the centered difference second derivative information

from each symmetric pair of points. Utilizing these approximation techniques, at each iteration we obtain an approximation to the gradient and inverse Hessian: g_{Tahk} and H^{-1}_{Tahk}, respectively.

At each iteration two different approximations to the stationary point are computed. The first is the from the approximation due to Tahk *et al.* (2009):

$$x^{(i)}_{\text{Tahk}} = x^{(i)} + h_{\text{Tahk}} = x^{(i)} - H^{-1}_{\text{Tahk}} g_{\text{Tahk}}. \tag{10}$$

The second approximation uses the nearest $k = 132$ points from among the last $N = 264$ points stored in the database to compute the radial basis function approximations to the gradient and Hessian approximations described in equations (5) and (6); we label these g_{rbf} and H_{rbf}, respectively.

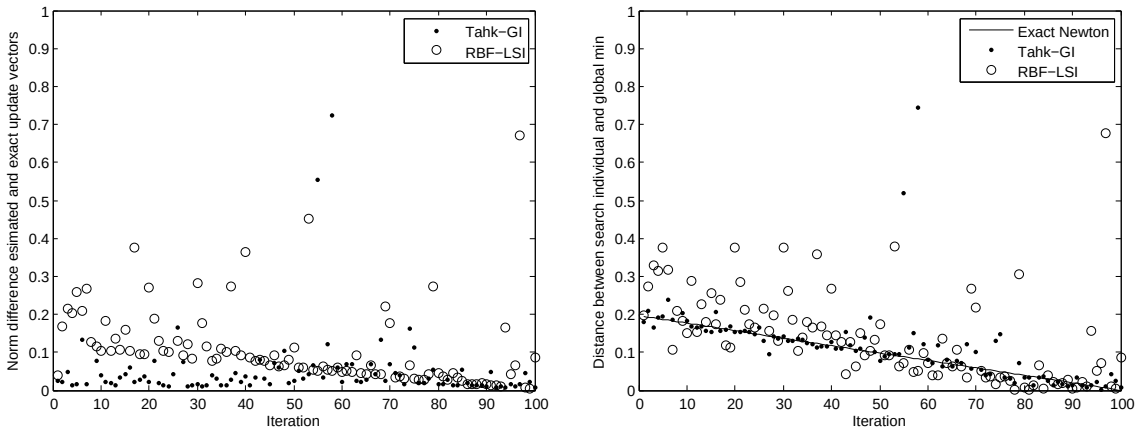
$$x^{(i)}_{\text{rbf}} = x^{(i)} + h_{\text{rbf}} = x^{(i)} - H^{-1}_{\text{rbf}} g_{\text{rbf}}. \tag{11}$$

We compare the update vectors h_{Tahk} and h_{rbf} to the analytically computed update vector in Figure 1(a). For the first 50 iterations the h_{Tahk} is a better approximation to the analytically computed update vector than computed by radial basis functions, but for iterations 51–100 the approximations are comparable. Even though h_{rbf} is not always the most accurate approximation to the update vector, it turns out that it does provide a good search direction. Figure 1(b) shows the distance from each of the approximated stationary points, $x^{(i)}_{\text{Tahk}}$ and $x^{(i)}_{\text{rbf}}$, as well as the analytically computed Newton point, to the global minimum of the 10-dimensional Rosenbrock function. It can be seen that the radial basis function approximation local search individuals and the Tahk *et al.* (2009) gradient individuals give comparable approximations to the global minimizer in spite of the fact that h_{rbf} is sometimes a less accurate approximation to the true update vector (from the analytic derivatives) than that of h_{Tahk}. In both plots, it can be seen that the radial basis function local search individiual is occasionally quite a bad approximation to the global minimum. This appears to happen because the problem of radial basis function interpolation and derivative interpolation is ill-conditioned and very sensitive to the choice of interpolation points. In most generations our simple choice of nearest neigbor points is adequate, but not always. This will be the subject of future work.

The radial basis function approach to computing the local search individual is the more expensive algorithm, but can be adapted to be used with any population based algorithm as it requires no modification of the underlying sampling algorithm. This makes the radial basis function approach suitable to be used with the covariance matrix adaptation evolution strategy as will be described in the next section.

3. Covariance Matrix Adaptation Evolution Strategy with Local Search Individual

The basic outline of the hybrid evolution strategy algorithm is illustrated by the flowchart in Figure 2. To study the efficacy of this version of our hybridized approach we have implemented the algorithm in the context of CMAES. While we have also done this in the context of the standard evolution strategy, we use CMAES here because it seems to have better global convergence properties (Hansen and Kern 2004, Hansen 2009). In particular, we utilize the Matlab version of CMAES (version 2.54) provided publically



(a) (b)

Figure 1. Comparison of rbf local search and Tahk local search for 10-dimensional Rosenbrock function. Figure (a) compares the update vectors for the two methods; Figure (b) shows that the distance from the approximated stationary point to the global minimizer is comparable for the two methods.

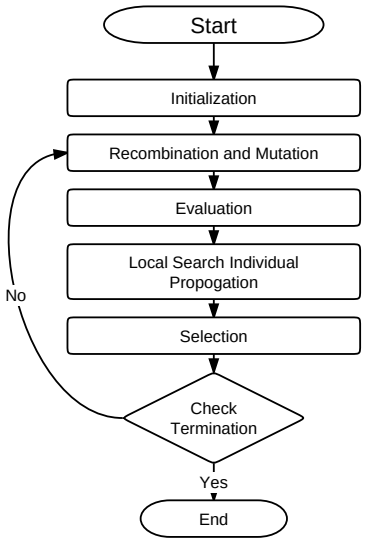


Figure 2. Flowchart for the hybrid evolution strategy algorithm

by Hansen (2004).

CMAES is an evolution strategy that adapts a full covariance matrix of a normal search distribution (Hansen and Ostermeier 1996). The strategy begins with an initial population of λ individuals $\mathbf{x}_{k=1:\lambda}^{(0)}$. After evaluating the objective function, the best μ individuals are selected as parents and their centroid is computed by using a weighted average: $\langle \mathbf{x} \rangle_W^{(0)} = \sum_{k=1}^{\mu} w_k \mathbf{x}_{k:\lambda}^{(0)}$, where the weights, w_i, are positive reals and sum to one. The notation $\mathbf{x}_{k:\lambda}$ is called selection notation and represents the point with the k^{th} lowest corresponding objective fitness value. While many weighting schemes have been proposed, here we use the super-linear weights: $w_i = \ln(\mu) - \ln(i), i = 1, \ldots, \mu$, wherein the individuals with lowest fitness values get the highest recombination weights.

After selection and recombination a new population is created by:

$$\mathbf{x}_{k=1:\lambda}^{(t+1)} = \langle \mathbf{x} \rangle_W^{(t)} + \sigma^{(t)} \mathbf{B}^{(t)} \mathbf{D}^{(t)} \mathbf{z}_{k=1:\lambda} \tag{12}$$

where $\mathbf{z}_k \sim N(0, \mathbf{I})$ are independent realizations of an n-dimensional standard normal distribution with mean zero and covariance matrix $\mathbf{I}$. The base points, $\mathbf{z}_k$, are rotated and scaled by the eigenvectors, $\mathbf{B}^{(t)}$, and the square root of the eigenvalues ,$\mathbf{D}^{(t)}$, of the covariance matrix ,$\mathbf{C}^{(t)}$. The covariance matrix, $\mathbf{C}^{(t)}$, and global step size, $\sigma^{(t)}$, are updated after each generation. This approach yields a strategy that is invariant to any linear transformation of the search space. Equations for initializing and updating the strategy parameters are given in (Hansen and Kern 2004). For complete details on the CMAES algorithm the tutorial (Hansen 2010) is a definitive source.

Table 1. Pseudo-code of the hybrid algorithm.

```
Generate and evaluate initial population x_{k=1:λ}^{(0)}
Append these points to the database
Choose best individual and set x_ls^{(0)}
Set t = 0, g_0 = 0_{n×1}, H_0 = C^{(0)} = I_{n×n}
repeat
   Compute nearest neighbor RBF and find g_t and H_t
   Update the local search individual x_ls^{(t+1)} = x_ls − (H_t)^{-1} g_t
   Evaluate x_ls^{(t+1)} and append to the database if feasible
   Select the best μ individuals from the population and x_ls^{(t+1)}
   Generate new population x_{k=1:λ}^{(t+1)} by recombination and mutation
   Evaluate new population and append to database.
   if min_i f(x_i) < f(x_ls^{(t)}) then
      swap individual with lowest function value and x_ls^{(t)}
   end if
   t = t + 1
until Stopping criteria are satisfied
```

Pseudo-code for the CMAES-RBFLSI algorithm is shown in Table 1. One additional feature that has not been mentioned is that a weighted norm is used to compute nearest neighbors for determining the support of the local radial basis function approximation. We use the current covariance matrix of the CMAES since it should reflect the shape of the search distribution and the objective function surface. The eigenvector/eigenvalue decomposition of the current covariance matrix is $C = BD^2B^T$. The distance between a point $x \in \mathbb{R}^n$ and the current gradient individual x_{ls} is measured as $\|(BD)^{-1}(x - x_{\text{ls}})\|_2$. (For instance, a unit ball in this norm will be elliptically shaped to fit in a long narrow valley in the search space.) The nearest neighbors in this norm should be ideal points for constructing an approximation to the Hessian matrix.

4. Hybrid Algorithm applied to mathematical test suite

A small suite of test problems has been selected to compare the performance of our hybrid CMAES local search individual algorithm (CMAES-RBFLSI) with ordinary CMAES.

Our aim is to establish that the hybridized approach does not change the global convergence properties of CMAES while accelerating the local convergence rate. For comparison, we have also implemented the quasi-Newton Hessian-approximation gradient individual approach of Tahk *et al.* (2009) in the context of CMAES (their implementation was in the standard evolution strategy); we refer to this implementation as CMAES-QNGI.

In CMAES-QNGI, after recombination (finding the centroid of the selected parents), only the first half of the new population of individuals is generated by equation (12). After evaluating the objective function for these individuals, the current gradient individual is swapped for an individual with lower objective function value, if one exists, to ensure that the gradient individual is the current best point. The remainder of the current population is formed by reflecting the first half of the population symmetrically through the gradient individual in $\mathbb{R}^n$. This symmetrically selected population reduces the order of the discretization errors in forming the first and second derivative approximations at the gradient individual.

A summary of the selected test functions is shown in Table 2. Function f_1 is the quadratic Schwefel 1.2 function and is a test case for CMAES because its elliptical contours test the ability of the algorithm to adapt the shape of the search distribution. f_2 is the cone function selected for its lack of differentiability at the minimum. f_3 is the classical generalized Rosenbrock function which has two minima for $n \geq 4$ and the long narrow valley which slows convergence for many algorithms. f_4 is the Schwefel 1.5 function which, while unimodal, is not differentiable if any $x_i = 0, i = 1, \ldots, n$. The Griewank function f_5 is multimodal, but is not particularly challenging for CMAES and is smooth. The Rastrigin function, f_6, is multimodal and smooth and is a difficult problem for CMAES (Auger and Hansen 2005). Finally, the Ackley function, f_7, is multimodal and smooth except at the minimum and is somewhat challenging in larger dimensions for CMAES.

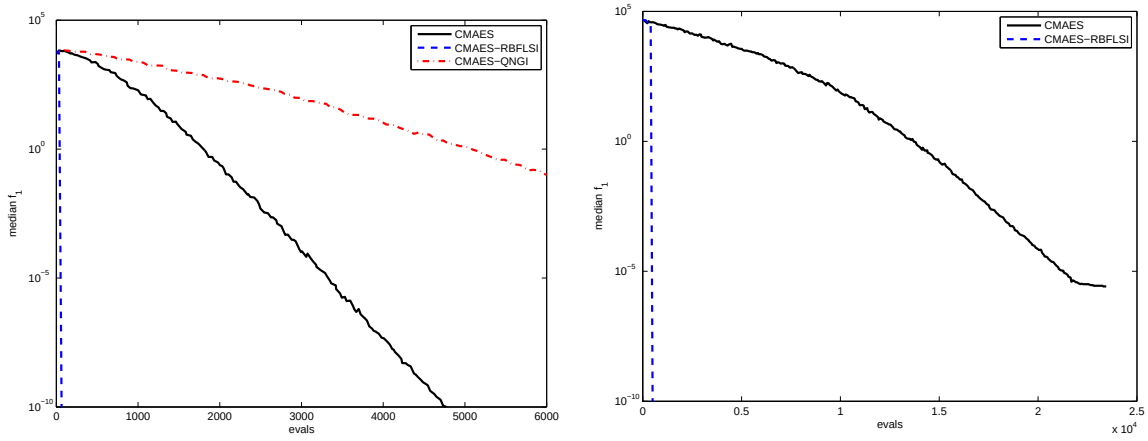
Table 2. Test Functions for Hybrid Optimization Strategy

Name	Definition	Search Domain
Schwefel 1.2	$f_1(x) = \sum_{i=1}^{n} \left(\sum_{j=1}^{i} x_j \right)^2$	$[-40, 60]^n$
Cone	$f_2(x) = \left(\sum_{i=1}^{n} x_i^2 \right)^{1/2}$	$[-40, 60]^n$
Rosenbrock	$f_3(x) = \sum_{i=1}^{n-1} \left[100 \left(x_{i+1} - x_i^2 \right)^2 + (x_i - 1)^2 \right]$	$[-40, 60]^n$
Schwefel 1.5	$f_4(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	$[-40, 60]^n$
Griewank	$f_5(x) = 1 + \sum_{i=1}^{n} \frac{x_i^2}{4000} - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}} \right)$	$[-600, 600]^n$
Rastrigin	$f_6(x) = 10n + \sum_{i=1}^{n} \sum_{i=1}^{n} \left(x_i^2 - 10 \cos(2\pi x_i) \right)$	$[-40, 60]^n$
Ackley	$f_7(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^{n} x_i^2} \right) - \exp\left(\frac{1}{n} \sum_{i=1}^{n} \cos(2\pi x_i) \right) + 20 + e$	$[-32, 32]^n$

We examine only the convergence graphs to compare the algorithms. The CMAES-RBFLSI algorithm is expensive since it constructs the local radial basis function approximation at each generation based on $O(n^2)$ points using a naive linear solver that requires $O(n^6)$ operations. The solution time of (4) can be improved by using a null space method or iterative methods, but the presumption here is that the objective function evaluations are very expensive. Thus local search algorithms, even expensive ones, can and should be used to improve the efficiency of the overall search.

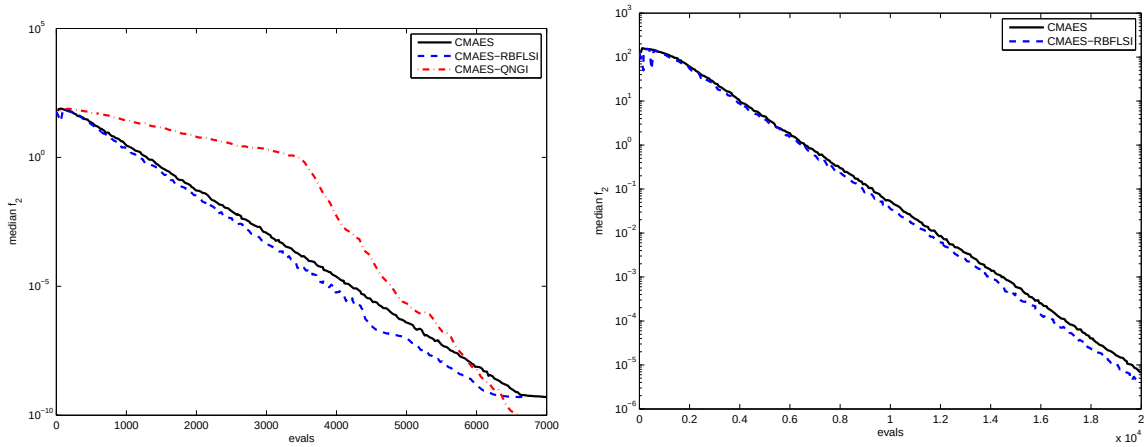
Figures 3-9 shows median convergence graphs based on a set of 30 trials for each of the test functions for each of the three algorithms: CMAES, CMAES-RBFGI, CMAES-QNGI. In each figure the results for dimension $n = 10$ are shown in plot (a), and the results for dimension $n = 30$ are shown in plot (b). We do not show the results of CMAES-QNGI at $n = 30$ since that algorithm does not perform well as will be discussed further below. For $n = 10$, the population size $\lambda = 30$ with $\mu = 15$ parents being selected at each generation, while with $n = 30$ we use $\lambda = 80$ and $\mu = 40$.. The initial global step size, σ is set to 30% of the total length of the search domain in each dimension. The initial population is sampled from a uniform distribution, and the same samples are used to initialize each of the three algorithms. For the CMAES-RBFLSI algorithm, the local RBF approximation is constructed using the k nearest neighbors with $k = (n + 1)(n + 2)$ which is twice the minimum number of points necessary to construct a quadratic polynomial interpolant in $\mathbb{R}^n$. The k nearest neighbors are selected from among the last $N = 2k$ individuals that have been evaluated. The algorithm is stopped when a minimum objective function value of 10^{-10} is reached or when the best objective function value does not change for the last $10 + 30n/\lambda$ generations or when the ratio of the range of the current function values to the maximum current function value is below TolFun$= 5 \times 10^{-10}$. The maximum number of function evaluations was set to be $n \times 10^4$.

Several points can be made upon inspection of the convergence graphs in Figures 3-9. First, the approach of Tahk *et al.* (2009) appears to interfere with the convergence of



(a) (b)

Figure 3. Convergence graphs for, f_1, the Schwefel 1.2 quadratic, unimodal function in 10 (a) and 30 (b) dimensions. Convergence is nearly instantaneous for CMAES-RBFLSI once enough points are obtained for an interpolant.

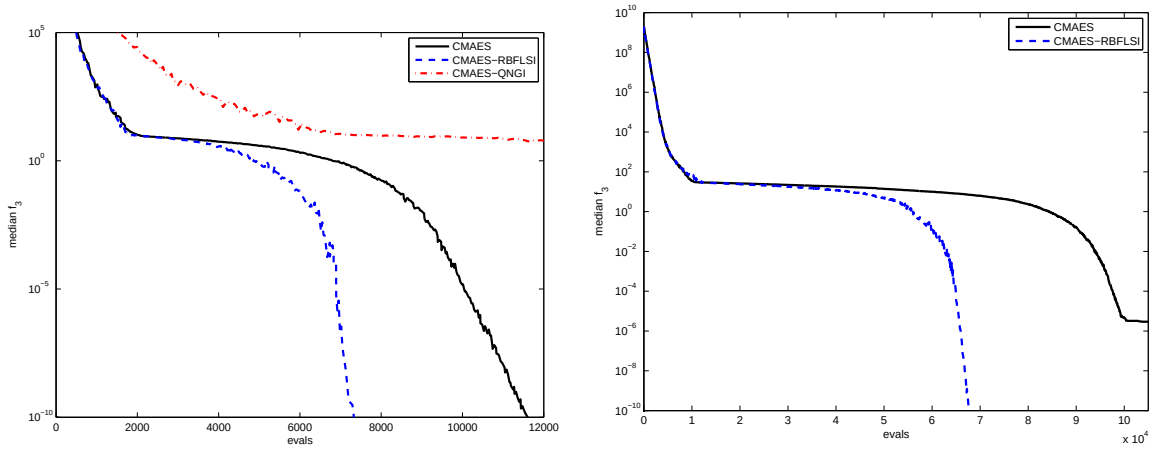


(a) (b)

Figure 4. Convergence graphs for, f_2, the cone function in 10 (a) and 30 (b) dimensions. The cone function is unimodal, but not differntiable at the minimum. Speedup is neglible.

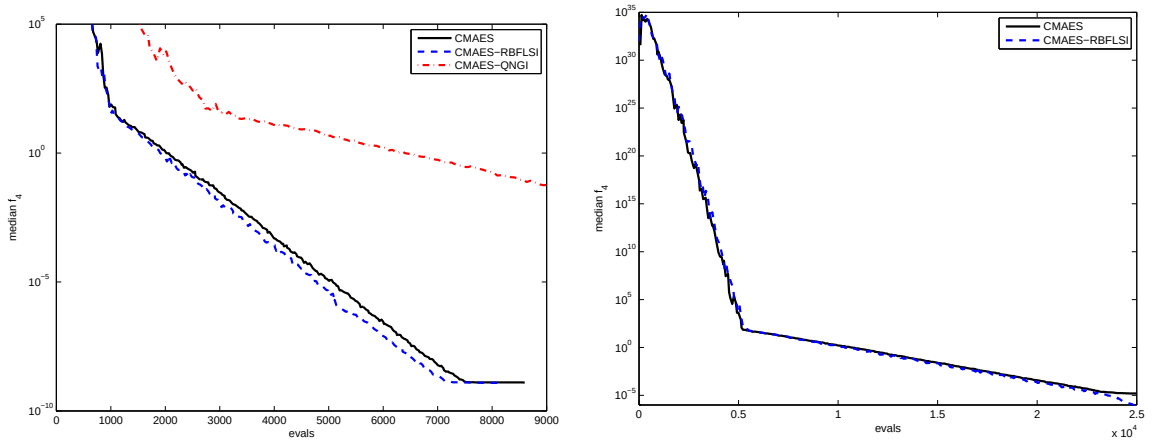
CMAES. This does not seem to be a case of simple error in implementing their strategy as we have been able to successfully implement and test it in the context of a standard evolution strategy; results not shown here. Rather the symmetrization of the population at each generation seems to interfere with the covariance matrix adaptation and/or step size adaptation algorithm in CMAES. It may be possible to develop a new version of CMAES which can accomodate the symmetrized population used in CMAES-QNGI, but that is beyond the scope of the current paper. The RBF local search individual approach does not have this difficulty since it does not modify the existing population of the evolution strategy other than simply appending an extra individual to the population.

A second observation is that for simple functions which are sufficiently smooth near their minima, such as the Schewefel 1.2 quadratic (f_1; Figure 3) and Rosenbrock (f_3; Figure 5) functions the CMAES-RBFLSI algorithm requires significantly fewer function



(a) (b)

Figure 5. Convergence graphs for, f_3, the generalized Rosenbrock function in 10 (a) and 30 (b) dimensions. It has two minima and is smooth, but has a long, flat and narrow valley that makes optimization slow for most algorithms. CMAES clearly benefits from local search as the minimum is approached.

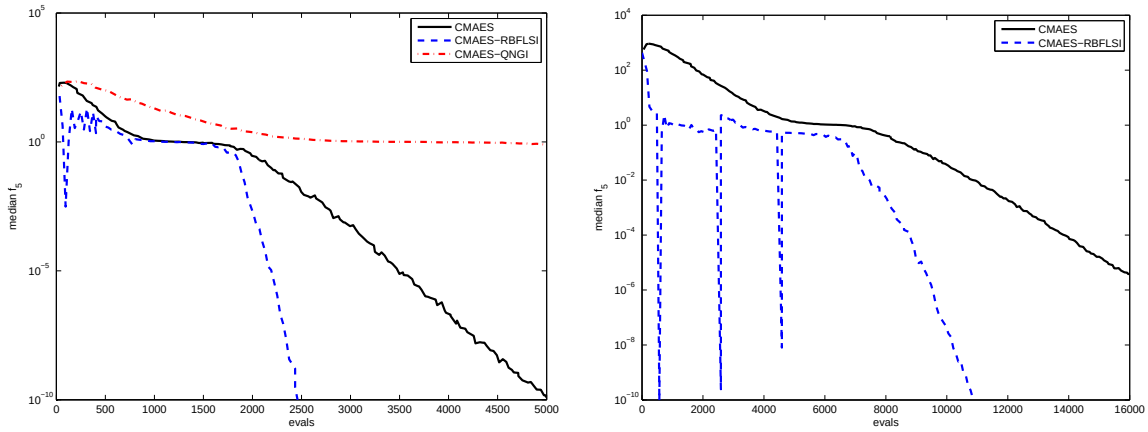


(a) (b)

Figure 6. Convergence graphs for, f_4, the Schwefel 1.5 function in 10 (a) and 30 (b) dimensions. It is unimodal with the minimum at $x = 0$, but is not differentiable if any $x_i = 0, i = 1, \ldots, n$.

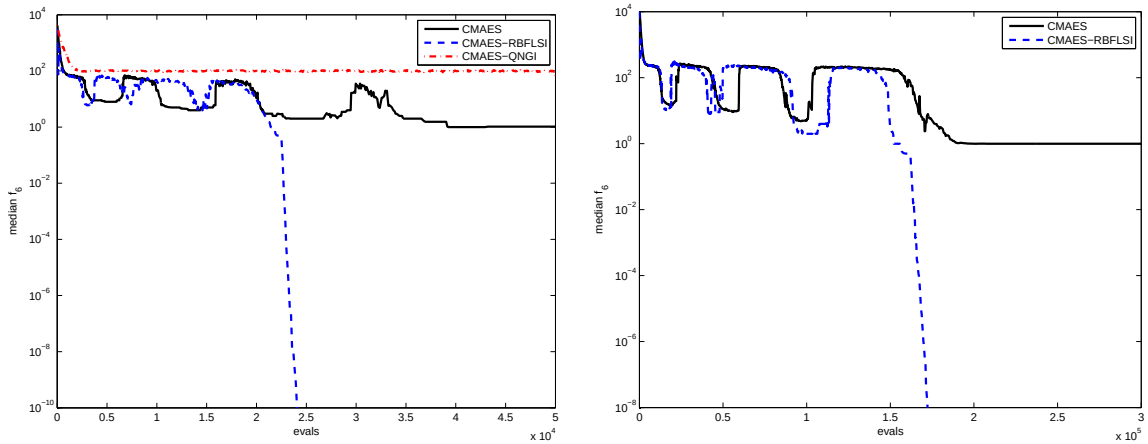
evaluations than standard CMAES. However, when the function is not differentiable at the minima the speedup due to the proposed algorithm is quite small. The latter is evident in the cone functon (f_2; Figure 4), Schwefel 1.5 function(f_4; Figure 6), and even the multimodal Ackley function (f_7; Figure 9).

The hybrid algorithm performs well for multimodal functions as well. For instance, in 10 and 30 dimensions there are significant reductions in the number of function evaluations for the Griewank function (f_5; Figure 7). The Griewank function is a relatively easy function for CMAES to optimize, but the CMAES-RBFLSI is able converge more quickly in the vicinity of the smooth global minimum. In fact, in many of the trials, the radial basis function local search individual is very close to the global minimum very early in the run, giving rise to the dips in the convergence curves for CMAES-RBFLSI in Figure



(a) (b)

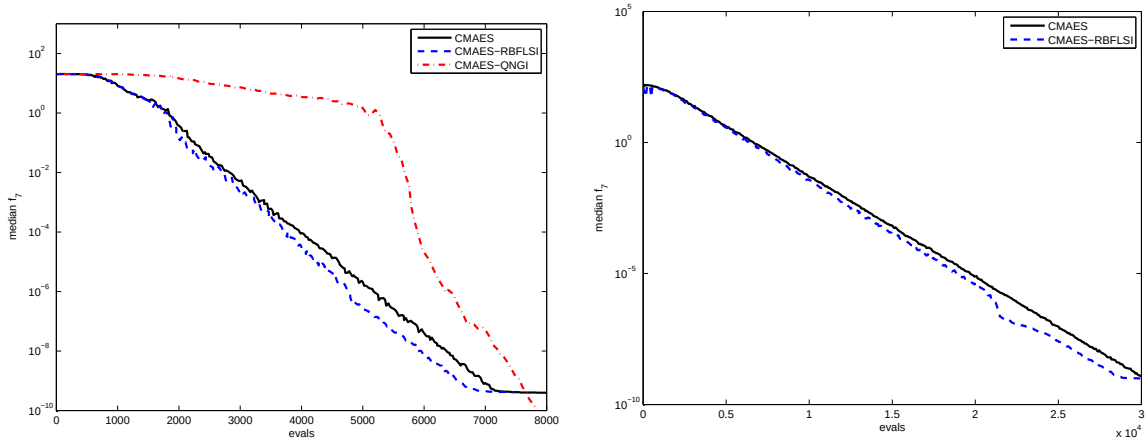
Figure 7. Convergence graphs for, f_5, the Griewank function in 10 (a) and 30 (b) dimensions. It is multimodal and smooth. While the Griewank function is not particularly challenging for CMAES, it nevertheless is greatly accelerated by inclusion of the local search individual. The radial basis function search individual often nearly finds the global optimum early in the optimization run giving rise to the dips in these convergence graphs.



(a) (b)

Figure 8. Convergence graphs for, f_6, the Rastrigin function in 10 (a) and 30 (b) dimensions. It is multimodal and smooth. For this function restarts and population doubling are used; see the text. This function is difficult for CMAES which rarely finds the global minimum, but CMAES-RBFLSI is much more reliable.

7. The Rastrign function (f_6; Figure 8) is more interesting as the population size needs to be quite large to locate the global minimum. For this function a population doubling restart scheme (Auger and Hansen 2005) was used in which the algorithm was restarted iteratively with population size $\lambda_k = 2^k\lambda, k = 1, 2, 3, 4$, and $\mu = \lambda_k/2$. In 10 dimensions CMAES locates the global minimum in only 5 of the 30 trials, while in 30 dimensions only 4 of the 30 trials. While with CMAES-RBFLSI, the global minimum is located in 22 or 30 trials in 10 dimensions and in 26 of 30 trials in 30 dimensions. Evidently, the proposed algorithm is able to more quickly converge to local minima in each restart and is thus able to use the computational budget more efficiently to find the global minimum.



(a) (b)

Figure 9. Convergence graphs for, f_7, the Ackley function in 10 (a) and 30 (b) dimensions. It is multimodal and not differentiable at the global minimum. For this function restarts and population doubling are used; see the text.

5. Application to calibration of a watershed model.

To demonstrate the utility of the CMAES-RBFLSI algorithm in a practical setting we used the algorithm to calibrate HYMOD, a five-parameter conceptual rainfall-runoff model (see Figure 11), introduced by Boyle (2000). In short, given time series of daily precipitation (P) and evapotranspiration (ET) data the objective is to tune the parameters so that the least squares error between the model predicted stream flow time series and the observed stream flow time series is minimized. Such problems are usually characterized by multiple minima, sometimes unidentifiable parameters and even discontinuities in the objective function (Duan *et al.* 1992).

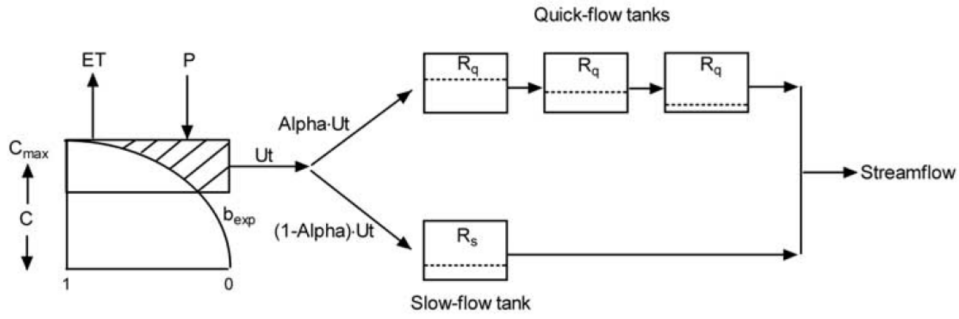


Figure 10. Schematic representation of the HYMOD model; from (Vrugt *et al.* 2003)

In attempts to parsimoniously represent the salient features of the precipitation-runoff process in a watershed system, the HYMOD model model structure (Moore 1985) is characterized by two series of linear reservoirs; in particular, three identical quick and a single slow response reservoir. The five parameters to be calibrated for the model stream flow to match the observed stream flow data are: the maximum storage capacity of the catchement, $C_{\max}$; the degree of spatial variability of the soil moisture capacity, $b_{\exp}$; the factor distributing flow between the two series of reservoirs, Alpha; and the residence

time of the linear quick and slow reservoirs, R_q and R_s, respectively. The hydrologic model parameters are inferred by adjusting their values until an acceptable level of agreement is achieved between a set of historical observations of the real world system that the model represents and their simulated counterparts. In this case, the objective function is simply the sum of the squared differences between the observed and simulated daily stream flows.

Three years, October 1, 1948, to September 30, 1951, of daily hydrologic data from the Leaf River watershed were used for model calibration. This humid 1944 km^2 watershed is located north of Collins, Mississippi. The data, obtained from the National Weather Service Hydrology Labratory, consist of mean areal precipitation (mm/d), potential evapotranspiration (mm/d), and stream flow (m^3/s).

Table 3. Uncertainty ranges of HYMOD model parameters

	Minimum	Maximum	Unit
$C_{\max}$	1.000	500.000	mm
$b_{\exp}$	0.100	2.000	
Alpha	0.100	0.990	
R_s	0.000	0.300	day
R_q	0.000	0.990	day

The CMAES and CMAES-RBFLSI algorithms are each applied to the optimization or calibration of the HYMOD model for 30 trials. The initial ranges of the parameter values are shown in Table 3; similar ranges of parameter values are used in model calibration study in (Vrugt *et al.* 2003). As for the mathematical test functions discussed above, the algorithms are initialized with the same uniform initial distributions. We set $\lambda = 10$ and $\mu = 5$. Each algorithm stops when TolFun$= 5e-4$, as described previously. For CMAES-RBFLSI the number of nearest neighbors used for the local RBF is $k = \lceil 1.5(n+1)(n+2)/2 \rceil = 23$ chosen from the last $N = 2k = 46$ evaluated points.

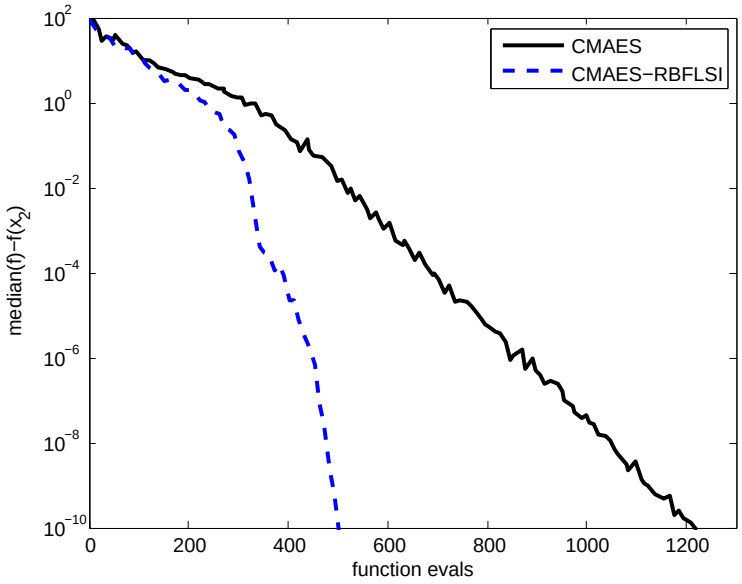


Figure 11. Convergence of CMAES and CMAES-RBFLSI for the HYMOD model.

There are many local minima, but CMAES and CMAES-RBFLSI nearly always converge to one of two minima $\mathbf{x}_1 = (157.0796, 0.5440, 0.2376, 0.2624, 0.8178)$ where

$f_{\text{HYMOD}}(\mathbf{x}_1) = 128.5346$ or $\mathbf{x}_2 = (146.9868, 0.7165, 0.2416, 0.2619, 0.8313)$ where $f_{\text{HYMOD}}(\mathbf{x}_2) = 128.6374$. The global minimum appears to be at $\mathbf{x}_1$ but CMAES converges to $\mathbf{x_2}$ in 28 of 30 trials, while CMAES-RBFLSI converges to the same minimum in 27 of 30 trials. Evidently, the basin of attraction for the global minimum, $\mathbf{x}_1$, is quite small as both algorithms have trouble finding it. The accelerated convergence of CMAES-RBFLSI to the local minimum of the HYMOD model is demonstrated in Figure 11. For each trial the best function value is saved at each generation. The median function value over the thirty trials minus the value at the local minimum , $f(\mathbf{x_2})$, is plotted versus the number of function evaluations. As Fiure 11 demonstrates, the increase in convergence speed is quite dramatic: CMAES-RBFGI typically converges with fewer than half of the objective function evaluations. Though neither algorithm reliably locates the global minimum, both algorithms give good approximations to the global minimum that produce adequate approximations to the daily stream flows. To locate the global minimum reliably, a restart strategy could be used as with the Rastrigin function above. The RBFLSI method would still accelerate the convergence significantly.

6. Conclusions

The local search individual hybridization approach for evolution strategies has been shown to be effective for significantly accelerating the convergence of the covariance matrix adaptation evolution strategy for functions which exhibit sufficient smoothness near the minimum. Likewise, it also works with the standard evolution strategy, though the results are not shown here.

To develop a hybrid evolution strategy using local RBF approximation, as we have done here, requires very little modification of the actual evolution strategy. The evolving population itself is not modified, but the additional local search individual is added at each generation. In the approach of Tahk *et al.* (2009) the population is chosen symmetrically at each generation and, as seen here, this can interfere with the convergence of CMAES.

Another advantage to this approach is that no minimum population size is required. In (Tahk *et al.* 2007, 2009) the population size must be at least twice the dimension of the search space to estimate the gradient vector. The downside of RBFGI approach is that it is expensive to form the coefficient matrix in Eq. (4). Moreover, the size of that system scales as $O(n^2)$, so its solution by a direct method requires $O(n^6)$ operations per generation. Thus, there is a trade-off between the computational complexity of the RBFLSI method and the gain due to fewer function evaluations. For expensive objective functions the cost of adding a local search individual propagated by local radial basis function approximation is expected to be incidental and the increase in speed can be enormous. As a result, the RBFLSI approach should be incorporated into evolution strategies for expensive functions as it can sometimes greatly increase converge speed and reliability with little downside.

In a future work we will consider a local search based on a trust region search along the Newton update direction on the local radial basis function approximation, as in (Wild *et al.* 2008) to locate the local search individual. The integration of a more robust local search may improve the local convergence properties of this hybridization approach.

Acknowledgements

The first author gratefully acknowledges the support of ARO Grant W911NF-06-0306. Permission was granted by the Chief of Engineers to publish this information.

References

Abbas, H., Bagirov, A., and Zhang, J., 2003. The discrete gradient evolutionary strategy for global optimization. *In*: *Evolutionary Computation, CEC '03*, 435–442.

Auger, A. and Hansen, N., 2005. A restart CMA evolution strategy with increasing population size. *In*: *Proc. IEEE Congr. Evol. Comput.*, Vol. 1, Edinburgh, U.K.., 1777–1784.

Baggett, J. and Skahill, B., 2010. Hybrid Optimization using Evolutionary and Approximate Gradient Search for Expensive Functions. *In*: *Proc. 2nd Int. Conf. Eng. Opt.*

Boyle, D., 2000. Multicriteria calibration of hydrological models. Thesis (PhD). Univ. Arizona.

Duan, Q., Gupta, V., and Sorooshian, S., 1992. Effective and efficient global optimization for conceptual rainfall-runoff models. *Water Resour. Res.*, 28, 1015–1031.

Fornberg, B., Flyer, N., and Russell, J., 2009. Comparisons between pseuospectral and radial basis function derivative approximations. *IMA Journal of Numerical Analysis*, 30 (1), 149–172.

Hansen, N., 2004. *Source Code for CMA-ES* [online]. INRIA, Orsay, France: N. Hansen. Available from: `http://www.lri.fr/~hansen/cmaes_inmatlab.html`, [Accessed July 2007].

Hansen, N., 2009. Benchmarking a BI-Population CMA-ES on the BBOB-2009 Function Testbed. *In*: *Workshop Proc. GECCO Genetic Evol. Comp. Conf.*, Montreal, Canada., 2389–2395.

Hansen, N., 2010. *The CMA Evolution Strategy: A tutorial* [online]. INRIA, Orsay, France: N. Hansen. Available from: `http://www.lri.fr/~hansen/cmatutorial.pdf` [Accessed October, 2010].

Hansen, N. and Kern, S., 2004. Evaluating the CMA strategy on multimodal test functions. *In*: X. Yao, E. Burke, J. Lozano, J. Smith, J. Merelo-Guervos, J. Bullinaria, J. Rowe, P. Tino, A. Kaban and H.P. Schwefel, eds. *Parallel Problem Solving from Nature – PPSN VIII.*, Vol. 3242 of *Lecture Notes in Computer Science* Springer Verlag, 282–291.

Hansen, N. and Ostermeier, A., 1996. Completely arbitrary normal mutation distributions in evolution strategies. *In*: *Proc. IEEE Cong. Evol. Comput.*, Nagoya, Japan., 312–317.

Kern, S., Hansen, N., and Koumoutsakos, P., 2004. Local Meta-Models for Optimization Using Evolution Strategies. *In*: T. Runarsson, H.G. Beyer, E. Burke, J. Merelo-Guervs, L. Whitley and X. Yao, eds. *Parallel Problem Solving from Nature – PPSN IX.*, Vol. 4192 of *Lecture Notes in Computer Science* Springer Verlag, 282–291.

Min, B.M., *et al.*, 2007. Autopilot Design using Hybrid PSO-SQP Algorithm. *Advanced Intelligent Computing Theories and Applications. With Aspects of Contemporary Intelligent Computing Techniques.*, Vol. 2 of *Communications in Computer and Information Science* Springer Verlag, 596–604.

Moore, R., 1985. The probability-distributed principle and runoff production at point and basin scales. *Hydrol. Sci. J.*, 30 (2), 273–297.

Peters, F., Barra, L., and Lemaoge, A., 2010. Application of a Hybrid Optimization Method for Identification of Steel Reinforcement in Concrete by Electrical

Impedance Tomography. *In*: *Proc. 2nd Int. Conf. Eng. Opt.*

Powell, M., 1992. The theory of radial basis function approximation in 1990. *In*: W. Light, ed. *Wavelets, Subdivision Algorithms and Radial Basis Functions.*, Vol. 2 of *Advances in Numerical Analysis* Oxford Univ. Press, 105–210.

Regis, R. and Shoemaker, C., 2004. Local Function Approximation in Evolutionary Algorithms for the Optimization of Costly Functions. *IEEE Trans. Evol. Comput.*, 8 (5), 490–505.

Renders, J.M. and Flasse, S., 1996. Hybrid Methods using Genetic Algorithms for Global Optimization. *IEEE Trans. Sys. Man. Cyb.*, 26 (2), 243–258.

Tahk, M.J., *et al.*, 2009. Hessian approximation algorithms for hybrid optimization methods. *Engineering Optimization*, 41 (7), 609–633.

Tahk, M.J., Woo, H., and Park, M., 2007. A hybrid optimization method of evolutionary and gradient search. *Engineering Optimization*, 39 (1), 87–104.

Vrugt, J., *et al.*, 2003. A shuffled complex evolution Metropolis algorithm for optimization and uncertainty assessment of hydrologic model parameters. *Water Resources Research*, 39.

Wild, S.M., Regis, R.G., and Shoemaker, C.A., 2008. ORBIT: Optimization by Radial Basis Function Interpolation in Trust-Regions. *SIAM J. on Scientific Computing*, 30 (6), 3197–3219.

Woo, H., Kwon, H., and Tahk, M.J., 2004. A hybrid method of evolutionary algorithms and gradient search. *In*: S. Mukhopadhyay and G.S. Gupta, eds. *Proc. 2nd Int. Conf. Autonomous Robots and Agents*, Palmerston North, New Zealand.