2nd International Conference on Engineering Optimization
September 6-9, 2010, Lisbon, Portugal

Hybrid Optimization using Evolutionary and Approximate Gradient Search for Expensive Functions

Jeff S. Baggett and Brian E. Skahill

University of Wisconsin - La Crosse, La Crosse, WI, USA, baggett.jeff@uwlax.edu
US Army Engineer Research and Development Center, Vicksburg, MS, USA, Brian.E.Skahill@usace.army.mil

Abstract

Optimization of expensive functions is more feasible with algorithms that require fewer evaluations of the objective function. In that spirit, this paper proposes a hybrid evolution strategy. As in [11, 9, 10], the proposed algorithm combines an evolution strategy (ES) with a gradient search technique. The proposed algorithm differs in that it uses a local radial basis function approximation to the objective function to compute approximate first and second derivatives to the objective function surface. The derivative information is used to propagate a gradient individual alongside the evolving population. The gradient individual is included for possible selection each generation. Tests on a small suite of standard test functions and a hydrologic application show that this hybrid approach can greatly accelerate the covariance matrix adaptation evolution strategy (CMAES). This hybrid approach is flexible and requires little modification of an existing evolution strategy; thus, it does not seem to alter negatively affect convergence when an objective function does not have sufficient smoothness for derivatives to yield useful descent information.
Keywords: Hybrid algorithms; Evolution strategies; Quasi-Newton method

1. Introduction

For multi-modal fitness functions evolution strategies are known to be reliable, but slow for approximating global optima. 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.

The algorithm proposed in this paper follows the hybrid evolution strategy first outlined in [11], but differs in two respects: first and most importantly, a more accurate and more flexible method is utilized to approximate the derivatives required for the local search, and second, the evolution strategy is the covariance matrix adaptation evolution strategy (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 [1].

The derivative estimation method will be explained first and it will be shown how the approximated derivatives are used to propagate the gradient individual between 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 suite of test functions in 10 dimensions. Finally, we briefly demonstrate an application in hydrological modeling.

2. Gradient Individual using Radial Basis Functions

The foundation of this optimization algorithm is the usual evolution strategy in which new offspring are produced at each generation by recombination and mutation (see Figure 1). 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 gradient individual [9], is propagated by a different mechanism each generation. The gradient individual, x_g^t, is either the fittest offspring of the current generation (individual with lowest function value) or the gradient 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_g^t and used to perform an update of the gradient 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. To update the gradient individual we use a k-nearest neighbor local

function approximation of the objective function using the k nearest neighbors (Euclidean distance) of x_g^t 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_g^t in the n-dimensional search space, p is in Π_2^n (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 [8].
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 $c = (c_1, \ldots, c_{\hat{m}})^T \in \mathbb{R}^{\hat{m}}$. Powell [7] 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 gradient individual, x_g^t:

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

For the Hessian matrix, H, evaluated at the gradient individual, x_g^t:

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

Similar techniques for derivative approximation are routinely used in the solution of partial differential equations using so-called meshless methods. Moreover, such approximations can be spectrally accurate [3].
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 gradient individual x_g^t. Finally, a new gradient individual is found by the standard update:

$$x_g^{t+1} = x_g^t - H^{-1} g. \tag{7}$$

The new gradient 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/or at least the update does not lead to a minimum. It's worth noting that the addition of this gradient individual to the offspring pool for selection seems to never be harmful to the 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 approximated by centered finite differences),

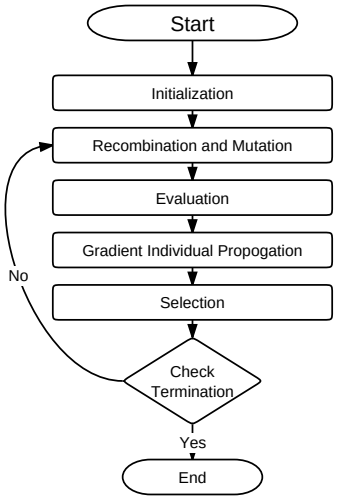

Figure 1: Flowchart for the hybrid evolution strategy algorithm

and the Hessian matrix is updated iteratively with information from new function and gradient evaluations. In fact, the original gradient individual/evolution strategy hybrid approaches of Tahk, et al [9, 10] used the quasi-Newton method to advance the 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 and should be more accurate than traditional quasi-Newton Hessian approximations.

3.Covariance Matrix Adaptation Evolution Strategy with Gradient Individual

The basic outline of the hybrid evolution strategy algorithm is illustrated in the flowchart in Figure 1. To study the efficacy of this version of the hybrid gradient individual / evolution strategy 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. We want to see if the addition of the gradient individual to the offspring at each generation will negatively affect the convergence of the CMAES. In particular, we utilize the Matlab version of CMAES (version 2.54) provided publically by Hansen [6].

CMAES is an evolution strategy that adapts a full covariance matrix of a normal search distribution [5]. 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:

$$\mathbf{x}_{k=1:\lambda}^{(t+1)} = \langle \mathbf{x} \rangle_W^{(t)} + \sigma^{(t)} \mathbf{B}^{(t)} \mathbf{D}^{(t)} \mathbf{z}_{k=1:\lambda} \tag{8}$$

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 [4].

Table 1: Pseudo-code of the hybrid algorithm.

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

Pseudo-code for the CMAES-RBFGI 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 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_g is measured as $\|(BD)^{-1}(x - x_g)\|_2$. (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 test suite

A small suite of test problems has been selected to compare the performance of our hybrid CMAES gradient individual algorithm (CMAES-RBFGI) with ordinary CMAES. For comparison, we have also implemented the quasi-Newton Hessian-approximation gradient individual approach of Tahk, et al. [10] 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 Eq.8. 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 symmetrical 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. Functions f_1, the classical sphere function, and f_2 are simple unimodal functions. f_3 is a simple unimodal function with random noise. f_4 is a simple unimodal function but is not differentiable at the minimum (or at many other points). f_5 is the classical Rosenbrock function which has two minima and a long flat valley. f_6, f_7, and f_8 are all multimodal. The Rastrigin function, f_7 presents some difficulty for CMAES, while Ackley function has the feature that it is not differentiable at the global minimum. All of the functions have their global minimum value of zero at zero.

Table 2: Test Functions for Hybrid Optimization Strategy

Name	Definition	Search Domain
Sphere	$f_1(x) = \sum_{i=1}^{n} x_i^2$	$[-40, 60]^n$
Schwefel 1.2	$f_2(x) = \sum_{i=1}^{n} \left(\sum_{j=1}^{i} x_j \right)^2$	$[-40, 60]^n$
Schwefel 1.2 with noise	$f_3(x) = f_2(x)(1 + 0.4z)$ where z is $N(0,1)^n$	$[-40, 60]^n$
Schwefel 1.5	$f_4(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	$[-40, 60]^n$
Rosenbrock	$f_5(x) = \sum_{i=1}^{n-1} \left[100 \left(x_{i+1} - x_i^2 \right)^2 + (x_i - 1)^2 \right]$	$[-40, 60]^n$
Griewank	$f_6(x) = 1 + \sum_{i=1}^{n} \frac{x_i^2}{4000} - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}} \right)$	$[-600, 600]^n$
Rastrigin	$f_7(x) = 10n + \sum_{i=1}^{n} \sum_{i=1}^{n} \left(x_i^2 - 10 \cos(2\pi x_i) \right)$	$[-40, 60]^n$
Ackley	$f_8(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^{n} x_i^2} \right) - \exp\left(\frac{1}{n} \sum_{i=1}^{n} \cos(2\pi x_i) \right) + 20 + e$	$[-32, 32]^n$

Figure 2 shows convergence graphs for each of the test functions for each of the three algorithms: CMAES, CMAES-RBFGI, CMAES-QNGI. For each function the dimension is set at $n = 10$. The population size is $\lambda = 30$ with $\mu = 15$ parents being selected at each generation. 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-RBFGI algorithm, the local RBF approximation is constructed using the k nearest neighbors with $k = \lceil 1.5(n+1)(n+2)/2 \rceil = 99$ which is just 50% larger than 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 $N = 2k = 198$ individuals that have been evaluated. For each algorithm, 30 trials are conducted for each test function. 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$= 5 \times 10^{-10}$.

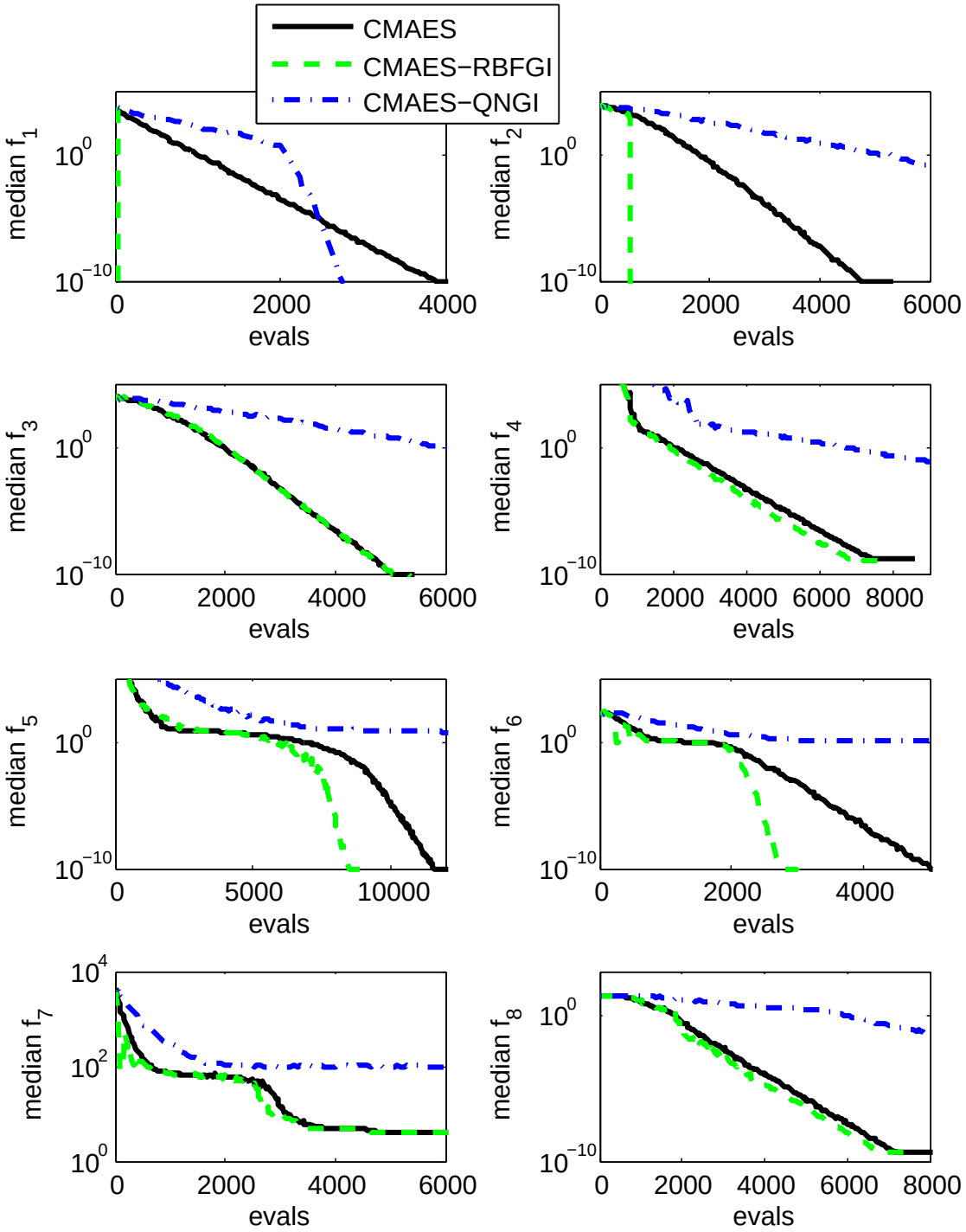

Figure 2: Convergence of the median objective value over 30 trials for 10 dimensional test functions.

The convergence graphs shown in Figure 2 illustrate several things. First, the approach of Tahk, et al. [9, 10] appears to interfere with the convergence of 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. Rather the symmetrization of the population at each generation seems to interfere with the finely tuned covariance matrix adaptation and step size adaptation algorithm in this version of CMAES. The RBF gradient individual approach does not have this difficulty since it does not modify the existing population of the evolution strategy in any way, but simply appends an extra individual to the population. Secondly, for simple, smooth convex functions the CMAES-RBFGI

algorithm is greatly accelerated relative to CMAES as can be seen for functions f_1 : Sphere and f_2 : Schwefel 1.2. When there is strong noise in the objective function we would not expect a gradient-based search to perform well; as can be seen in with f_3, there is no acceleration with CMAES-RBFGI. Functions f_4 : Schwefel 1.5 (unimodal) and f_8 : Ackley (multimodal) are not differentiable at the global minimum (both have a sharp points). As can be seen in the figure, little acceleration is achieved by CMAES-RBFGI. (In a test not shown here, the Ackley function was squared to remove the singularity in the first derivative and the result appeared more like that of f_6.) For the Rosenbrock function, f_5 and the Griewank function, f_6 the convergence graphs show that the convergence of CMAES-RBFGI accelerates dramatically as the population approaches the minimizer. Finally, for the Rastrigin function, f_7 there is very little difference between CMAES and CMAES-RBFGI because both algorithms have difficulty locating the global minimum. Instead they converge to a variety of local minima.
As for reliability, the algorithms are nearly indistiguishable. For functions $f_1 - -f_4$ both CMAES and CMAES-RBFGI found the global optimum for all trials. For the Rosenbrock function, f_5 both algorithms located the global optimum in 28 of the 30 trials. For the Griewank function, f_6 CMAES found the global optimum 21 times while CMAES-RBF-GI found it 23 times. Neither algorithm ever found the global optimum for the Rastrigin function, f_7. For the Ackley function, f_8 CMAES only failed to find the global optimum once, while CMAES-RBFGI found it for all 30 trials. These results are promising as they show that the addition of the gradient individual does not impede the global search capability of the evolution strategy.

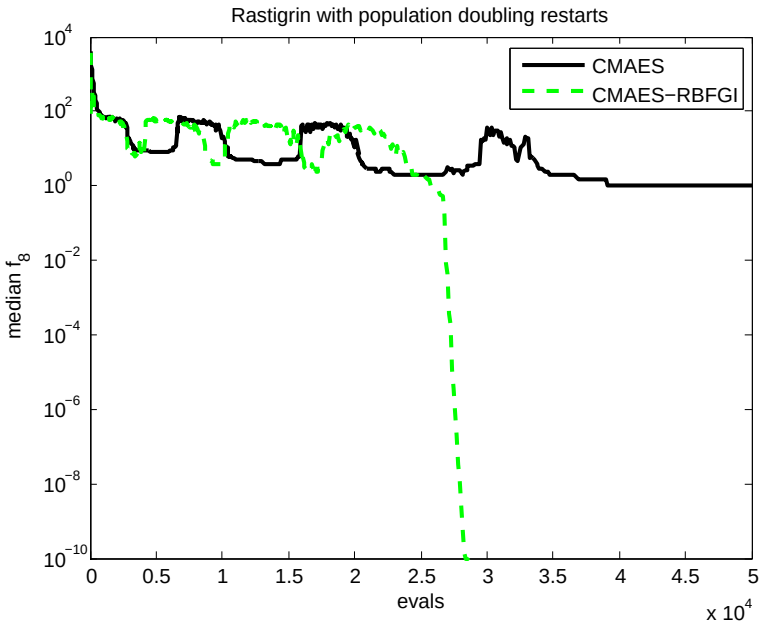

Figure 3: Convergence comparison for Rastrigin function with restarts

For the Rastrigin function we conduct a new experiment in which we run the evolution strategy using a population doubling restart strategy [1]. Each variation of CMAES is run until convergence as above with same $\lambda = 30$ as above and then the population size λ is doubled while $\mu = \lambda/2$ and the algorithm restarted. For CMAES-RBFGI the database of points is maintained between restarts. A total of 4 restarts are allowed to a maximum population size of 480. The maximum number of function evaluations remains capped at 50,000. The CMAES population size increasing strategy has been shown to be one of the most successful global optimization methods presently known for some benchmark problems [1]. The convergence graph for the median function value over 30 trials is shown in Figure 3. Most significantly, CMAES-RBFGI demonstrates far greater reliability. CMAES correctly locates, within the alloted 50,000 function evaluations, the global minimum at zero in 5 of the 30 trials, while CMAES-RBFGI finds the global minimum in 22 of the 30 trials.

5. Application to calibration of a watershed model.

To demonstrate the utility of the CMAES-RBFGI algorithm in a practical setting we used the algorithm to calibrate HYMOD, a five-parameter conceptual rainfall-runoff model (see Figure 5), introduced in [2].

In short, given time series of daily precipitation (P) and evapotranspiration (ET) data the objective is 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.

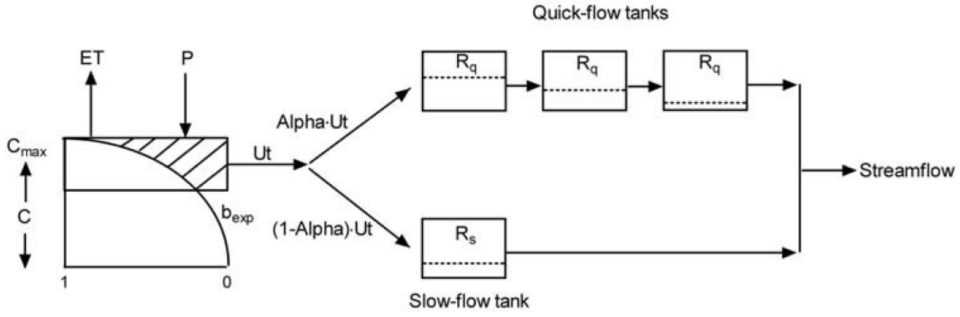

Figure 4: Schematic representation of the HYMOD model; from [12]

The HYMOD model is a simple rainfall excess model (details in [Moore 1985]), connected with two series of linear reservoirs: 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.
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-RBFGI 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. As for the 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-RBFGI 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.

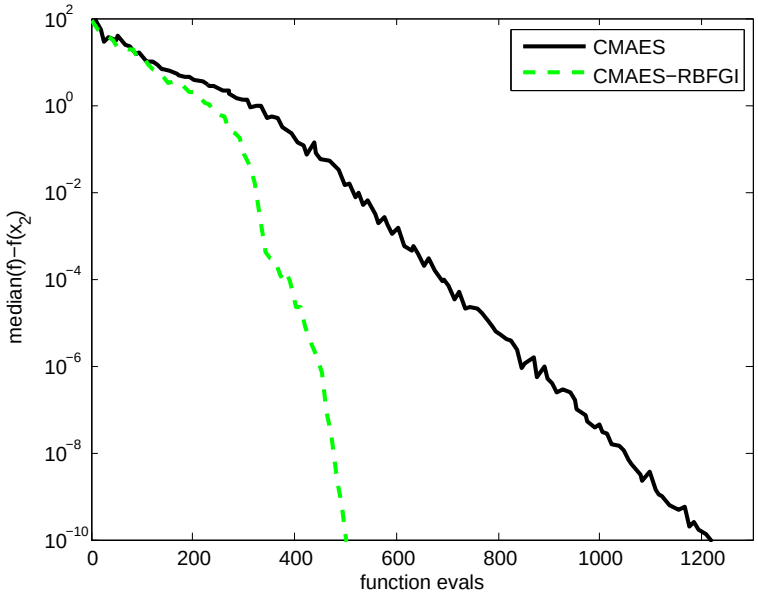

Figure 5: Convergence of CMAES and CMAES-RBFGI for the HYMOD model.

There are many local minima, but CMAES and CMAES-RBFGI 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-RBFGI 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-RBF to the local minimum of the HYMOD model is demonstrated in Figure 5. 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 5 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 RBFGI method would still accelerate the convergence significantly.

6. Conclusions

The gradient individual hybridization approach for evolution strategies has been shown to be effective for significantly accelerating the convergence of the covariance matrix adaptation evolution strategy. 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 gradient individual is added at each generation. In the gradient individual approach of Tahk, et al 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 [9, 10] 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 RBFGI method and the gain due to fewer function evaluations. For expensive objective functions the cost of adding a gradient 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 RBFGI approach should be incorporated into evolution strategies for expensive functions as it can sometimes greatly increase converge speed and reliability with little downside.

Acknowledgements

The first author gratefully acknowledges the support of ARO Grant W911NF-06-0306.

References

[**1**] A. Auger and N. Hansen, A restart CMA evolution strategy with increasing population size, *Proc. IEEE Congr. Evol. Comput.,* Edinburgh, U.K., vol. 1, 1777-1784, 2005.

[**2**] D.P. Boyle, Multicriteria calibration of hydrological models, Ph.D. disserations, Univ. Ariz., Tuscon, 200.

[**3**] B. Fornberg, N. Flyer, and J.M. Russell, Comparisons between pseudospectral and radial basis function derivative approximations, *IMA Journal of Numerical Analysis*, 30 (1), 149-172, 2009.

[**4**] N. Hansen and S. Kern, Evaluating the CMA strategy on multimodal test functions, *Lecture Notes in Computer Science, X*, Yao, et al Eds., Springer-Verlag, vol. 3242, Parallel Problem Solving from Nature – PPSN VIII, 282-291, 2004.

[**5**] N. Hansen and A. Ostermeier, Completely arbitrary normal mutation distributions in evolution strategies, *Proc. IEEE Congr. Evol. Comput.,* Nagoya, Japan, 312-317, 1996.

[**6**] N. Hansen, Source Code for CMA-ES, `http://www.lri.fr/~hansen/cmaes_inmatlab.html`.

[**7**] M.J.D. Powell, The theory of radial basis function approximation in 1990, *Advances in Numerical Analysis, Volume 2: Wavelets, Subdivision Algorithms and Radial Basis Functions*, W. Light Ed., London, U.K., Oxford Univ. Press, 105-210, 1992.

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

[**9**] M.-J. Tahk, H.W. Woo, and M.S. Park, A hybrid optimization method of evolutionary and gradient search, *Engineering Optimization*, 39 (1), 87-104, 2007.

[**10**] M.-J. Tahk, M.S. Park, H.W. Woo, H.J. Kim, Hessian approximation algorithms for hybrid optimization methods, *Engineering Optimization*, 41 (7), 609-633, 2009.

[**11**] H.W. Woo, H.H. Kwon, and M.-J. Tahk, A hybrid method of evolutionary algorithms and gradient search, *Proceedings of the 2nd International Conference on Autonomous Robots and Agents*, S.C. Mukhopadhyay and G. Sen Gupta Eds., Palmerston North, New Zealand, 2004 (ISBN 0-476-00994-4).

[**12**] J.A. Vrugt, H.V. Gupta, W. Bouten, and S. Sorooshian, A shuffled complex evolution Metropolis algorithm for optimization and uncertainty assessment of hydrologic model parameters, *Water Resources Research*, 39 (8), doi:10.1029/2002WR001642, 2003.