TAM-EDA: Multivariate t distribution, archive and mutation based estimation of distribution algorithm

B. Gao[1] I. A. Wood[2]

(Received 8 November 2012; revised 26 December 2013)

Abstract

We present a novel estimation of a distribution algorithm (EDA), TAM-EDA, which uses a multivariate t distribution model, an archive population and a mutation operation to escape local minima, avoid premature convergence and utilize a record of the best solutions. Earlier EDAs used multivariate normal distributions to model low-cost regions of the search space. The multivariate t distribution has heavier tails and so is more likely to maintain diversity, while still allowing convergence to occur. The current population of potential solutions has limited ability to represent all the best regions of the search space explored so far. The archive allows storage of a larger population of promising solutions, which are then used in model building. However, the EDA

http://journal.austms.org.au/ojs/index.php/ANZIAMJ/article/view/6365
 Published January 14, 2014, as part of the Proceedings of the 16th Biennial Computational Techniques and Applications Conference. ISSN 1446-8735. (Print two pages per sheet of paper.)

model and archive may still become stuck at suboptimal solutions, so to combat this we introduce a decomposition mutation operation which retains most of the attributes of a current solution but attempts large changes in others. A comparison with generic EDA, genetic algorithms and the Nelder–Mead method shows that TAM-EDA is an effective optimization algorithm for a range of test problems.
Keywords: Optimization; Estimation of Distribution Algorithms; Decomposition Mutation

Contents

1 Introduction **C722**

2 Background **C722**

3 The proposed algorithm: TAM-EDA **C726**
3.1 TAM-EDA framework C726
3.2 Probability and penalization C727
3.3 Decomposition mutation operation C728

4 Experiments **C729**
4.1 Test functions C733
4.2 Model estimation of Lorenz system C738
4.3 Transistor design problem C740

5 Conclusion **C742**

References **C742**

1 Introduction

Estimation of distribution algorithms (EDAs) are a class of derivative-free metaheuristic algorithms for optimization, based on ideas from genetic algorithms (GAs). At each generation of an EDA, a statistical model is estimated using the solutions with lower objective values, and the next generation of solutions is sampled from this model. The TAM-EDA proposed in this article improves on generic EDAs for continuous optimization problems in three ways:

T (Multivariate t distribution) Generic EDAs for continuous optimization use multivariate normal distributions as the statistical models. However, the tails of a normal distribution fall away quickly, so points are rarely generated far beyond the first or 99th percentiles. Multivariate t distributions have heavier tails and so are more likely to produce distant sample solutions for exploring other regions of the search space.

A (Archive population) Exploration of all promising regions of the search space is important for finding the global optimum in multi-modal problems. However, the current population is limited in its ability to represent the best regions found so far. We retain a larger archive population of the best solutions found over all generations and utilize it in model estimation.

M (Mutation operation) The mutation step of a GA aims to aid exploration, preserve diversity and avoid premature convergence. There is no direct analogue in an EDA and so we introduced a mutation operation to be applied to a fraction of the solutions at each iteration.

2 Background

Without loss of generality, optimization is treated as minimization where the aim is to find x^* such that $f(x^*) \leqslant f(x)$ for all x within the space of

interest and which meet any imposed constraints. Here we consider only simple boundary constraints on each dimension.

An early EDA for continuous optimization is the univariate marginal distribution algorithm (UMDA) [14] which uses a univariate normal distribution for sampling each element of the individual. This was extended to model the dependencies among elements in the estimation of a multivariate normal algorithm (EMNA) [8] which fits a multivariate normal distribution to the elite solutions, and hence is parameterised via a mean vector and covariance matrix. An EDA utilizing the multivariate Cauchy distribution (t distribution with one degree of freedom) was proposed by Posik [16]. This produced promising results, but was fitted heuristically rather than via maximum likelihood estimation.

Another example of stochastic model-based algorithms that are used for optimization purposes are cross entropy (CE) methods [20]. The CE method for optimization is the same as EDAs but with truncation selection. These two approaches were developed independently at around the same time. The CE methods generally aim to estimate rare event probabilities and the EDA methods derive from genetic algorithms, and aim to solve optimization problems [2, 13, 19]. Important choices in the implementation of these methods include methods of initialisation, choice of distributions, selection method and parameters, and constraint handling. A reasonable amount of theory for such algorithms was developed by both the EDA and CE optimization communities [21, 12, 3, 11].

Here we model solutions using a multivariate t distribution. This provides wider tails than the normal distribution, and offers finite mean and covariance terms which avoid generating extreme solutions.

At each generation of TAM-EDA, an elite population is sampled from the archive population using a roulette wheel sampler (RWS). The multivariate t distribution is estimated from the elite population. The next generation of solutions is sampled from this distribution, mutations are applied to a fraction of these and the archive is then updated using the new solutions.

The TAM-EDA models the solutions sampled from the archive via a multivariate t distribution, which is a multivariate generalization of the t distribution. In D dimensions it has the probability distrubution function

$$f(\mathbf{x}) = \frac{\Gamma\left[(\nu + D)/2\right]}{\Gamma\left(\frac{\nu}{2}\right) \nu^{\frac{D}{2}} \pi^{\frac{D}{2}} |\mathbf{\Sigma}|^{\frac{1}{2}} \left[1 + \frac{1}{\nu}(\mathbf{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu})\right]^{(\nu+D)/2}}, \tag{1}$$

where ν, the degrees of freedom, is a positive scalar, $\boldsymbol{\mu}$ is the $1 \times D$ mean vector, $\mathbf{x}$ is a $1 \times D$ vector, $\Gamma(\cdot)$ is the gamma function and Σ is a $D \times D$ symmetric positive semidefinite matrix. The covariance matrix for this distribution is $\nu\Sigma/(\nu - 2)$ for $\nu > 2$ and undefined otherwise.

It is possible to estimate the degrees of freedom for the multivariate t distribution. However, we did not pursue this for the following reasons.

- With the limited population size (e.g. 100) and the often large number of dimensions, the accuracy of any such estimation will be poor. Lange et. al [7], in their description of methods to fit the parameters of the multivariate t distribution, claimed that for small samples, a priori fixing the degrees of freedom at four worked well in a number of applications.
- Estimation of the degrees of freedom for the t distribution can be fairly computationally intensive [10].
- There is little difference between the main part of the normal distribution and most t distributions—the difference is most pronounced in the tails. Regardless of the shape implied by data, we use a wider-tailed distribution to encourage further exploration. This departs somewhat from standard EDA or CE methodology.

We wish to increase the probability with which rare events happen. There is approximately a one in 10,000 probability of producing a value more than 3.7 standard deviations from the mean with a standard normal distribution. The t distribution with four degrees of freedom produces observations further than 3.7 standard deviations from the mean with approximately 1% probability. We believed that this probability is high enough to allow occasional

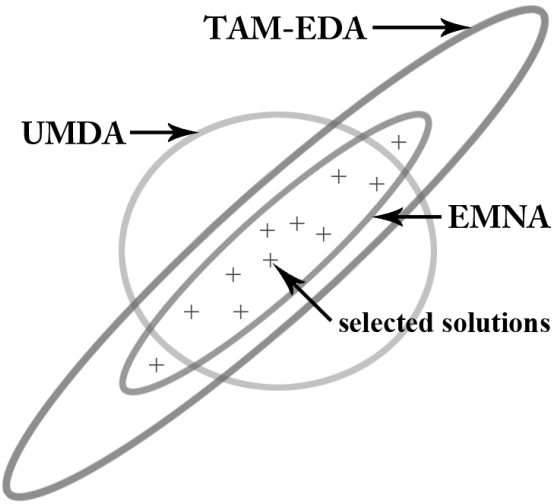

Figure 1: The contour plots of models built on the selected solutions.

exploration away from the current set of solutions. We do not use a distribution with wider tails such as the Cauchy because this might overly slow convergence. Therefore, the degrees of freedom $\nu = 4$ is used in TAM-EDA.

Figure 1 illustrates constant density contour lines including more than 95% of the probability for the models used in UMDA, EMNA and TAM-EDA when fitted to the solutions shown. UMDA's lack of covariance produces ellipsoidal contours whose axes coincide with those of the space. Both the EMNA and TAM-EDA models include covariance terms, so their ellipsoidal axes follow the principal components of the solutions from the previous step. The multivariate t distribution has heavier tails than the multivariate normal, so the contours required to enclose most of the probability are further away from the mean, allowing more exploration at each step.

3 The proposed algorithm: TAM-EDA

3.1 TAM-EDA framework

The framework of TAM-EDA is described as follows.

1. Initialization: Sample S_{cp} (size of current population) solutions uniformly from the domain. Let these comprise the current population P_c and let the generation index $t = 0$.
2. Evaluation: Evaluate the objective values of the population P_c with the penalization method described in Section 3.2.
3. Update archive: Combine the current population P_c with the archive population P_A and retain in the archive at most the S_{Amax} (maximum archive size) solutions with the lowest objective values. The archive size is now S_A.
4. Sample probabilities: Calculate the rank and probability of being drawn for each solution in the archive P_A.
5. Re-sampling: Sample S_{cp} solutions from the archive population P_A via RWS using their fitness values as proportions to obtain an elite population P_e.
6. Model fitting: Fit the multivariate t distribution with four degrees of freedom to the elite population P_e.
7. Sample next generation of solutions: For a mutation rate of γ, randomly select $S_{cp} \cdot \gamma$ solutions from P_e and apply a random mutation operation to each, as in Section 3.3. Sample $S_{cp} \cdot (1 - \gamma)$ solutions from the multivariate t distribution. Combine these solutions to form the next current population P_c.
8. Repeat steps 2–7 for T generations.

3.2 Probability and penalization

At each iteration, solutions are drawn from the archive via a RWS, and we assign a probability for each solution based on the rank of its cost value. While this preferentially selects lower cost solutions, every state in the archive has a reasonable probability of being selected, so some population diversity will be maintained. The probability of the ith solution being drawn is

$$p_i = r_i^{-\frac{1}{2}} \Big/ \sum_{j=1}^{S_A} r_j^{-\frac{1}{2}}, \quad i = 1, 2, \ldots, S_A, \tag{2}$$

where r_i is the rank of the ith solution, with the lowest cost solution ranked one.

Given the high dimensionality of the problem and the infinite support of the generation distribution, the probability of a solution being generated beyond the constraint boundaries is high, at least initially. Hence it is better to allow such solutions, but penalise them so that the algorithm moves into the feasible region over time.

The penalisation function $f_p(\mathbf{x})$ assigns an objective value to generated points $\mathbf{x}$ based on their distance beyond each of the constraint boundaries. We chose

$$f_p(\mathbf{x}) = \begin{cases} f_{max} \cdot \left[1 + \sum_{d=1}^{D} \frac{\max(b_d^{min} - x_d, 0) + \max(x_d - b_d^{max}, 0)}{b_d^{max} - b_d^{min}}\right] \\ \qquad \text{if } \sum_{d=1}^{D} [\mathbb{I}_{\mathbb{R}^+}(b_d^{min} - x_d) + \mathbb{I}_{\mathbb{R}^+}(x_d - b_d^{max})] > 0, \\ f(\mathbf{x}) \quad \text{otherwise,} \end{cases} \tag{3}$$

where b_d^{min} and b_d^{max} are the boundaries of the dth dimension of the search space, f_{max} is the largest objective value of the solutions within the boundaries found so far and $\mathbb{I}_{\mathbb{R}^+}$ is the indicator function for the set of positive real numbers.

The initial generation of solutions is sampled uniformly within the boundaries of the search space and the initial value of f_{max} is the maximum objective value among these.

3.3 Decomposition mutation operation

One concern for standard EDAs is that they may converge to a local minimum or flat region and be unable to escape. While the multivariate t_4 distribution (t distribution with four degrees of freedom) of TAM-EDA generates some distant solutions, this will become less common as the algorithm converges. A mutation operation with non-converging variance could continue to offer diversity. In constructing a mutation operation, we considered computational complexity and the idea that retaining many aspects of a current solution could lead to better new solutions, particularly for problems which are fully or partly separable. This approach is related to decomposition into subproblems utilised by Liu and Rubin [9], and to Gibbs sampling [18].

We propose a decomposition mutation operation (DMO) which at each generation randomly selects a one or two dimensional subspace in which to mutate some of the solutions. At each generation, a fraction γ of the elite population solutions is chosen for mutation, and with equal probabilities (0.5), either a one or two dimensional DMO is applied, as illustrated in Figures 2 and 3 for a Schwefel function in two dimensions.

For a selected dimension d, we add a mutation scalar S_m sampled from a t_4 distribution, scaled by σ_d, that is, $S_m/\sigma_d \sim t_4$.

For the dth dimension, σ_d is calculated as a fraction of the length of the feasible region, and so across all dimensions, written in vector form,

$$\boldsymbol{\sigma} = 0.6\boldsymbol{\rho}e^{-\tau/10}, \quad \boldsymbol{\rho} = \mathbf{b}^{\max} - \mathbf{b}^{\min}, \quad \tau = t \bmod 100, \tag{4}$$

where $\mathbf{b}^{\min}$ and $\mathbf{b}^{\max}$ are the lower and upper boundary vectors of the search space, respectively, and $\boldsymbol{\rho}$ is a vector of feasible region lengths, t is the generation number and τ is a local time parameter with a period of 100 generations. Since the most appropriate scale for mutations is unknown, we allow a range via a repeating geometric sequence of values, inspired by repetitions of simulated annealing [5], see Figure 4.

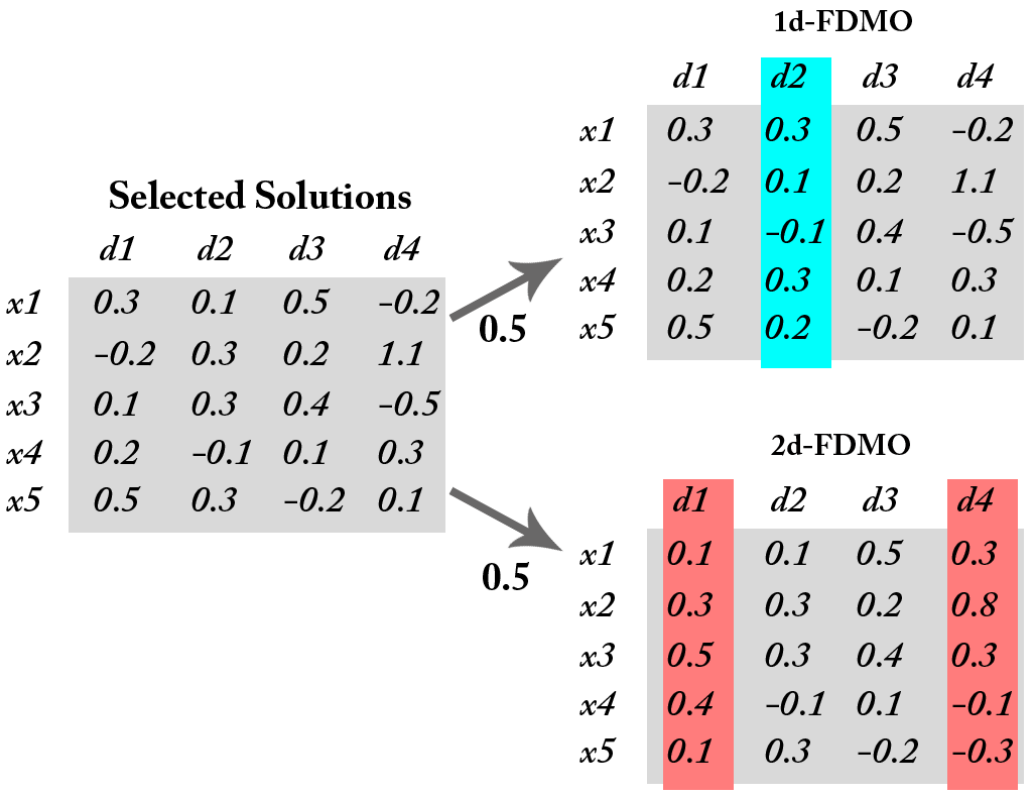

Figure 2: Decomposition mutation operation: numerical operations.

4 Experiments

TAM-EDA, UMDA, EMNA, GA and the Nelder–Mead method [6] were tested on three 100 dimensional optimization test functions and two applications. Standard Matlab functions from its Optimization Toolbox were employed for the GA and the Nelder–Mead method, with the domain provided but default options used otherwise. All algorithms were initialised from a uniform distribution over the problem domain.

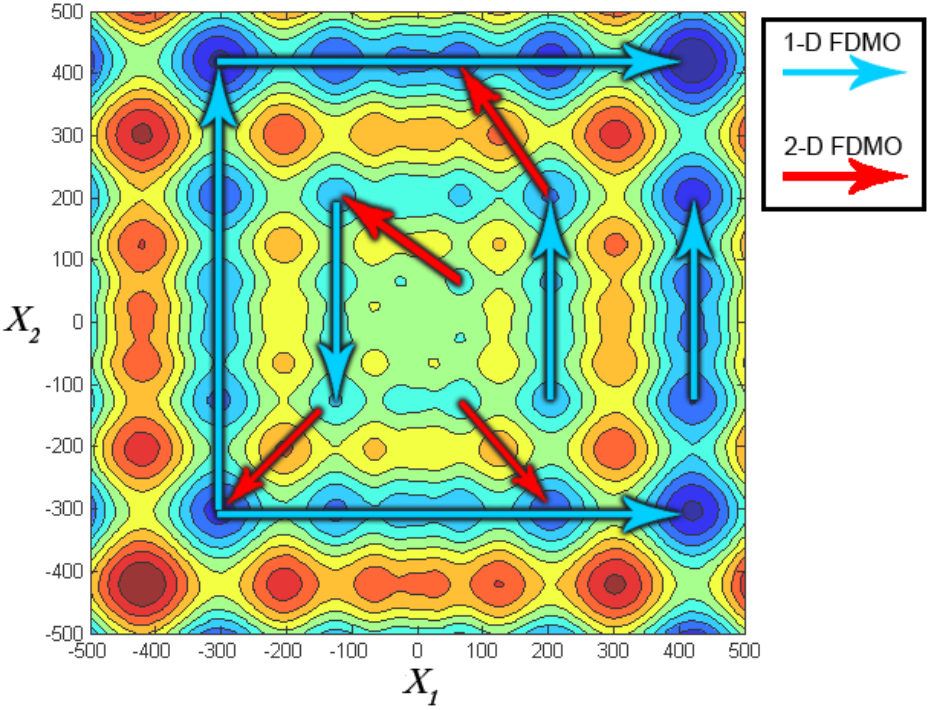

Figure 3: Decomposition mutation operation: illustration on 2D Schwefel problem.

Since this is an iterative algorithm and we are primarily counting the number of cost function evaluations (since these could be expensive), it is quite possible that the algorithm will make the same progress in five steps with population size N or one step with population size 5N. The presence of the archive mutes the effect of the population size since, for the purposes of storage and model generation, the archive contains the population of interest. The members of the current population are only placed in the archive if they improve upon its current members.

The current population is generated by constructing a model of the current archive and sampling from it. Once we fill the archive, the ideal population

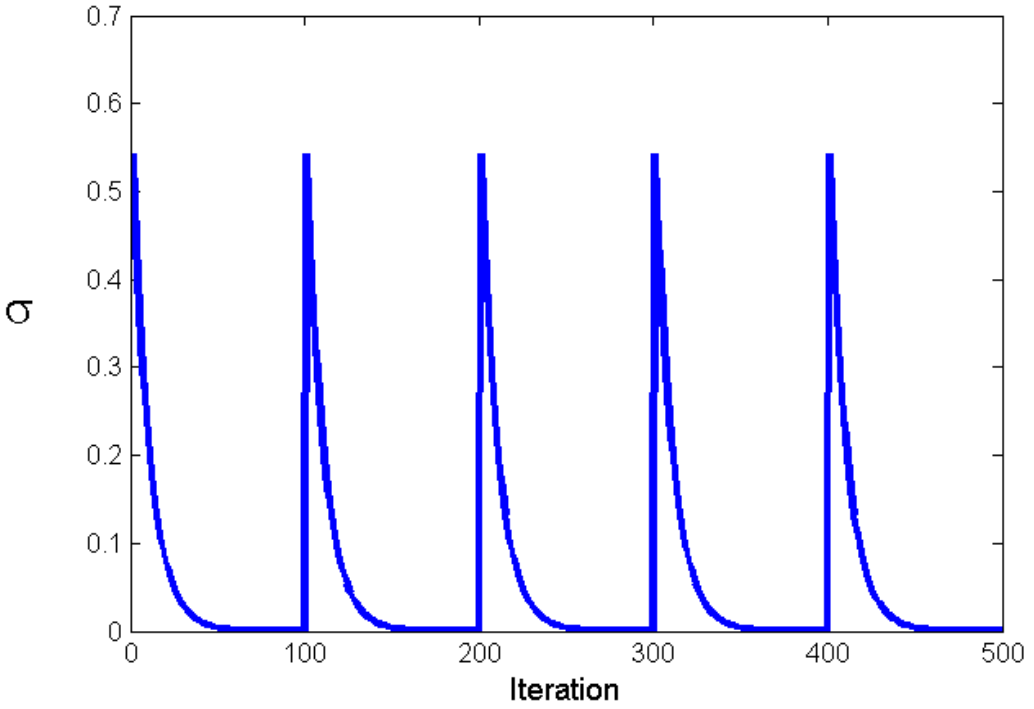

Figure 4: Decomposition mutation operation: repeating geometric sequence for scale.

size is a single point, then the archive is updated and a new model constructed. However, this process is too slow computationally, so we instead choose the population size to be small, but reasonable, and the exact number is not so important.

The archive size is potentially more important, since this is the basis of model construction. We made this as large as possible while maintaining a reasonable algorithm computation time. Larger archive sizes represent more diversity but in lower dimensions this may offer little benefit given that the model is a fairly simple unimodal symmetric distribution. Larger archives may be more useful with higher dimensional problems, but they also impose

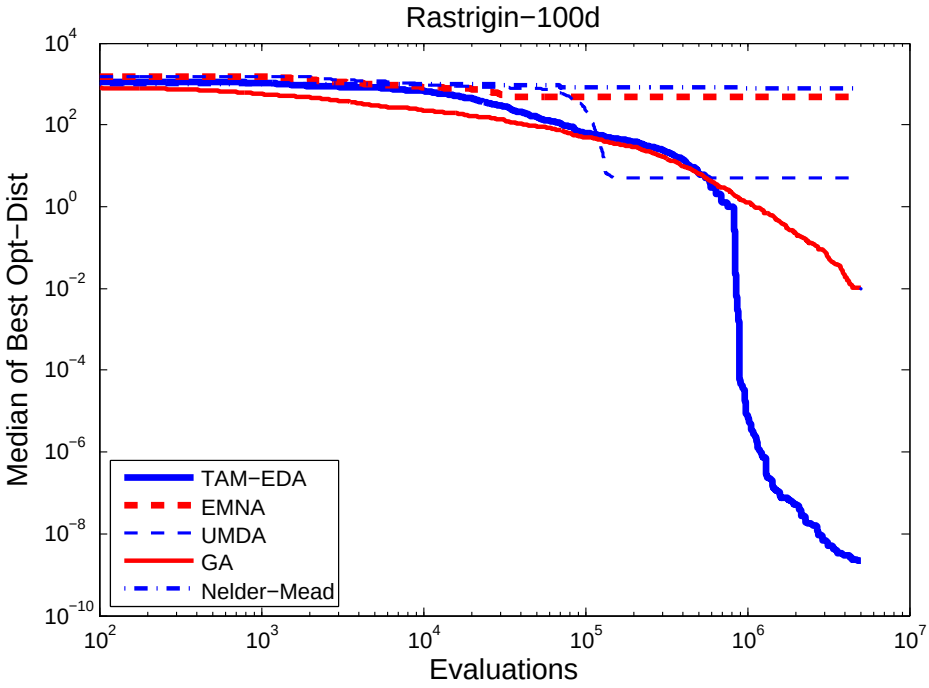

Figure 5: Log-log plot of median opt-dist on 100 dimensional Rastrigin test function across 31 runs.

a serious computational burden and tradeoffs must be made.

To summarise, we make the current population as small as possible and the archive population as large as possible, tempered by considerations of computation time.

For all experiments, TAM-EDA and GA used a population size of $S_{cp} = 100$. The archive size for the TAM-EDA was $S_{Amax} = 500$. The mutation rate of DMO in TAM-EDA was set at 0.3. The crossover and mutation rates of the GA were set at 0.8 and 0.2, respectively.

For UMDA and EMNA, the population size was 500, and the selection step

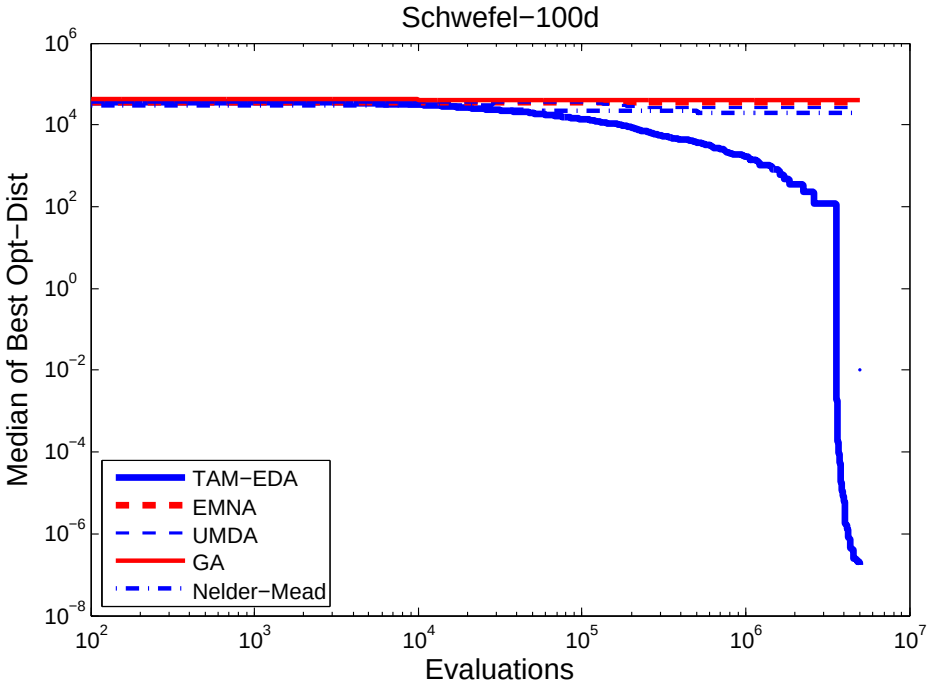

Figure 6: Log-log plot of median opt-dist on 100 dimensional Schwefel test function across 31 runs.

retained the best 50% of the solutions to estimate the model at each generation. Experimental results suggest that a larger population size, such as 500, improves the efficiencies of UMDA and EMNA. The Nelder–Mead method was restarted from a new random position once the one-step improvement in the cost function fell below 10^{-30}.

4.1 Test functions

The definition of the test functions and the search domains are given in Table 1. For all the tested algorithms, the total number of cost function evaluations

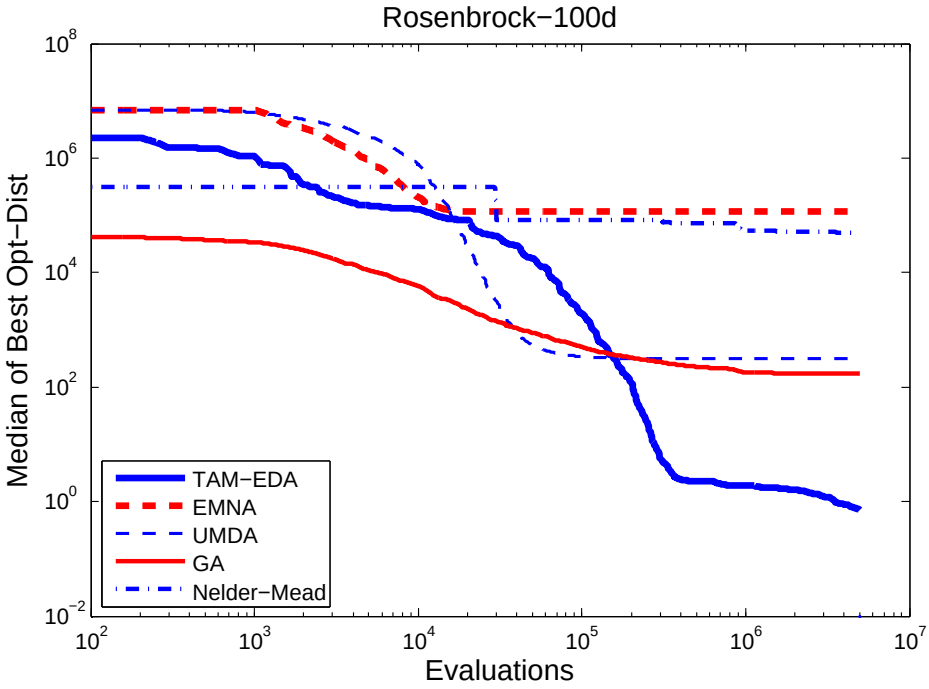

Figure 7: Log-log plot of median opt-dist on 100 dimensional Rosenbrock test function across 31 runs.

Table 1: Test functions

problem	objective function	domain
Rastrigin	$f(x) = 10D + \sum_{i=1}^{D} \left[x_i^2 - 10\cos(2\pi x_i)\right]$	$[-5.12, 5.12]^D$
Schwefel	$f(x) = 418.9829D + \sum_{i=1}^{D} \left[-x_i \sin\sqrt{\lvert x_i\rvert}\right]$	$[-500, 500]^D$
Rosenbrock	$f(x) = \sum_{i=1}^{D-1} \left[100\left(x_{i+1} - x_i^2\right)^2 + (1 - x_i)^2\right]$	$[-5, 10]^D$

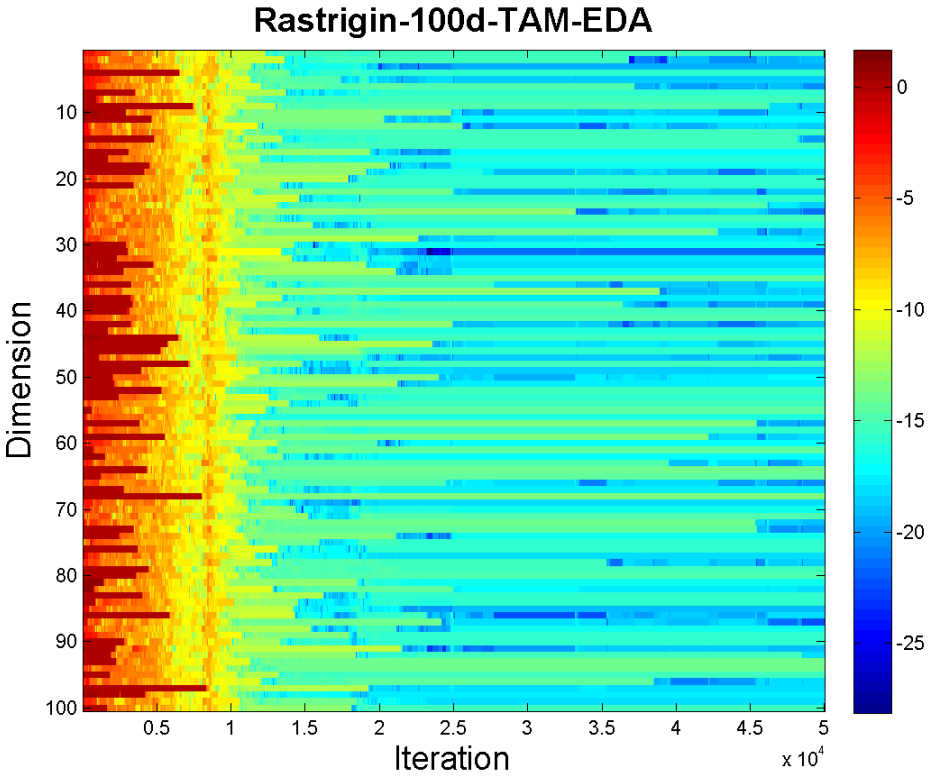

Figure 8: Heatmap of distance to the global minimum as a power of 10 in all dimensions for median TAM-EDA runs on the Rastrigin test function.

was 5×10^6 per run, with the best cost found so far recorded throughout each algorithm run. We define the opt-dist as the difference between the best objective value found so far and the global optimum objective value. Log-log plots of the median opt-dist over 31 runs versus generation number are reported in Figures 5–7 . TAM-EDA produced the best results for 10^6 or more evaluations on all three problems and was the only algorithm to reliably find the global minimum of the Schwefel function within the given number of function evaluations. Although it was difficult for all the test algorithms to

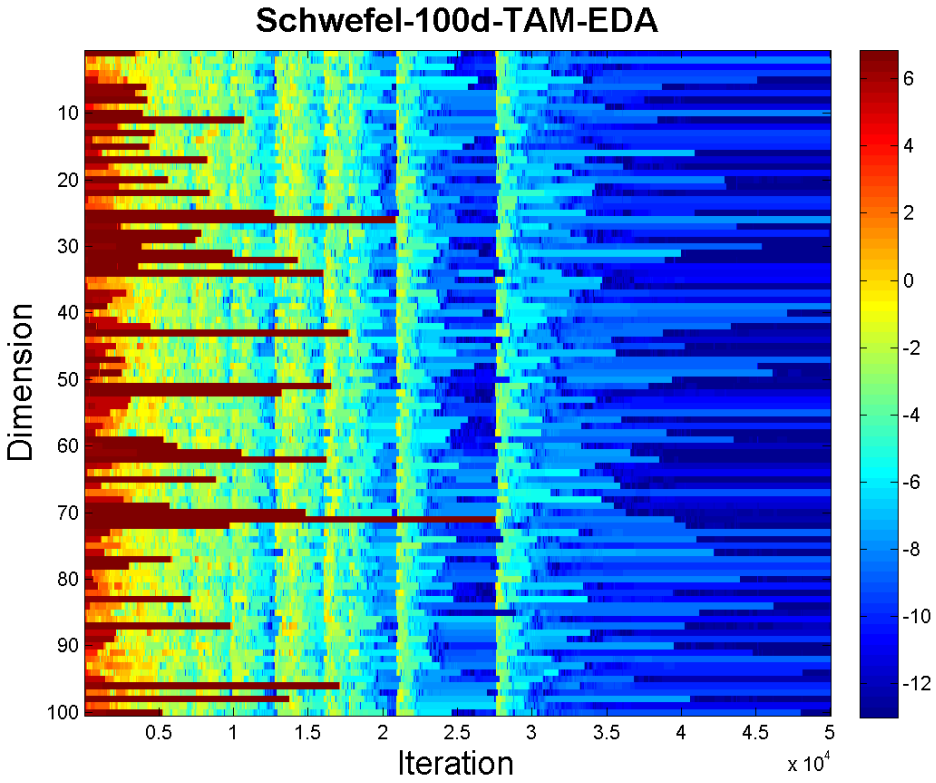

Figure 9: (Heatmap of distance to the global minimum as a power of 10 in all dimensions for median TAM-EDA runs on the Schwefel test function.

find the global minimum of the Rosenbrock test function, TAM-EDA obtained the solution closest to the global minimum.

Figures 8–10 show heatmaps of the best solutions obtained using TAM-EDA versus the generation number for the run with median final performance of each problem. The colours (with the scale shown in the side bar) indicate the distance in the solution space between the best solution so far and the position of the global minimum as a power of 10 in each dimension. These show that the time to converge to the global optimum varied widely among

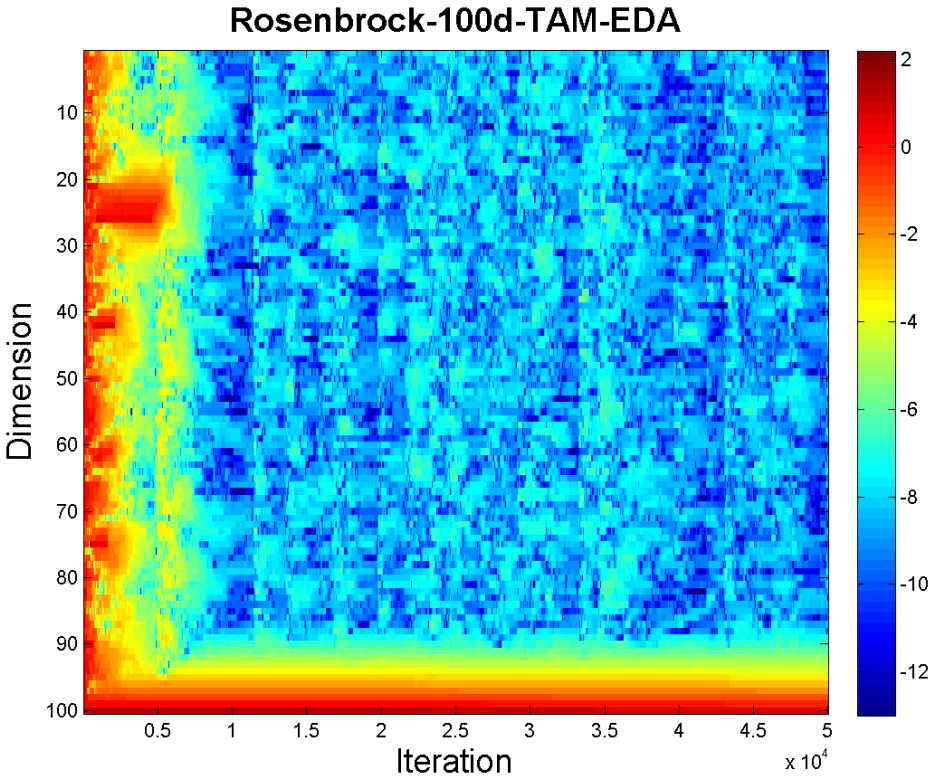

Figure 10: Heatmap of distance to the global minimum as a power of 10 in all dimensions for median TAM-EDA runs on the Rosenbrock test function.

dimensions. The Schwefel results, Figure 9, also show the algorithm becoming stuck in various local minima before escaping, possibly due to mutations. Each escape includes new best solutions closer to the global minimum, which is likely due to the large-scale structure of the problem.

4.2 Model estimation of Lorenz system

The Lorenz system was developed to model types of hydrodynamical flow and is known to have chaotic dynamics for some combinations of parameter values and initial conditions. The ordinary differential equations to describe the system are

$$\begin{aligned} \dot{x} &= \sigma(y - x)\,, \\ \dot{y} &= x(\rho - z) - y\,, \\ \dot{z} &= xy - \beta z\,, \end{aligned} \tag{5}$$

where x, y and z are the system variables, σ, ρ and β are the system parameters and dots are used to show derivatives with respect to time t.

Here we consider the optimization problem of estimating the parameters of this system based on a set of n noiseless observations, as previously studied by Alfi [1]. The aim is to find parameter estimates to minimize the sum of squared errors over the observations:

$$\text{err} = \sum_{i=1}^{n} \sum_{d=1}^{D} \left(P_d^i - Q_d^i\right)^2, \tag{6}$$

where $P = (P^1, \ldots, P^n)$ is the data generated using predefined parameters, $Q = (Q^1, \ldots, Q^n)$ is the corresponding data calculated using the estimated parameters, and each observation has dimensionality $D = 3$.

The parameters used for generating data were $x_0 = 0$, $y_0 = 1$, $z_0 = 0$, $\sigma = 3.0$, $\rho = 26.5$, $\beta = 1.0$, $t_0 = 0$, $t_e = 3$, where t_0 and t_e are the initial and end times, respectively, and 1000 equally spaced observations were taken in this time interval. The initial state of the system was known. The number of cost function evaluations used was 10^5 for each algorithm. Each evaluation requires the solution of the differential equations and calculation of the sum of squared errors.

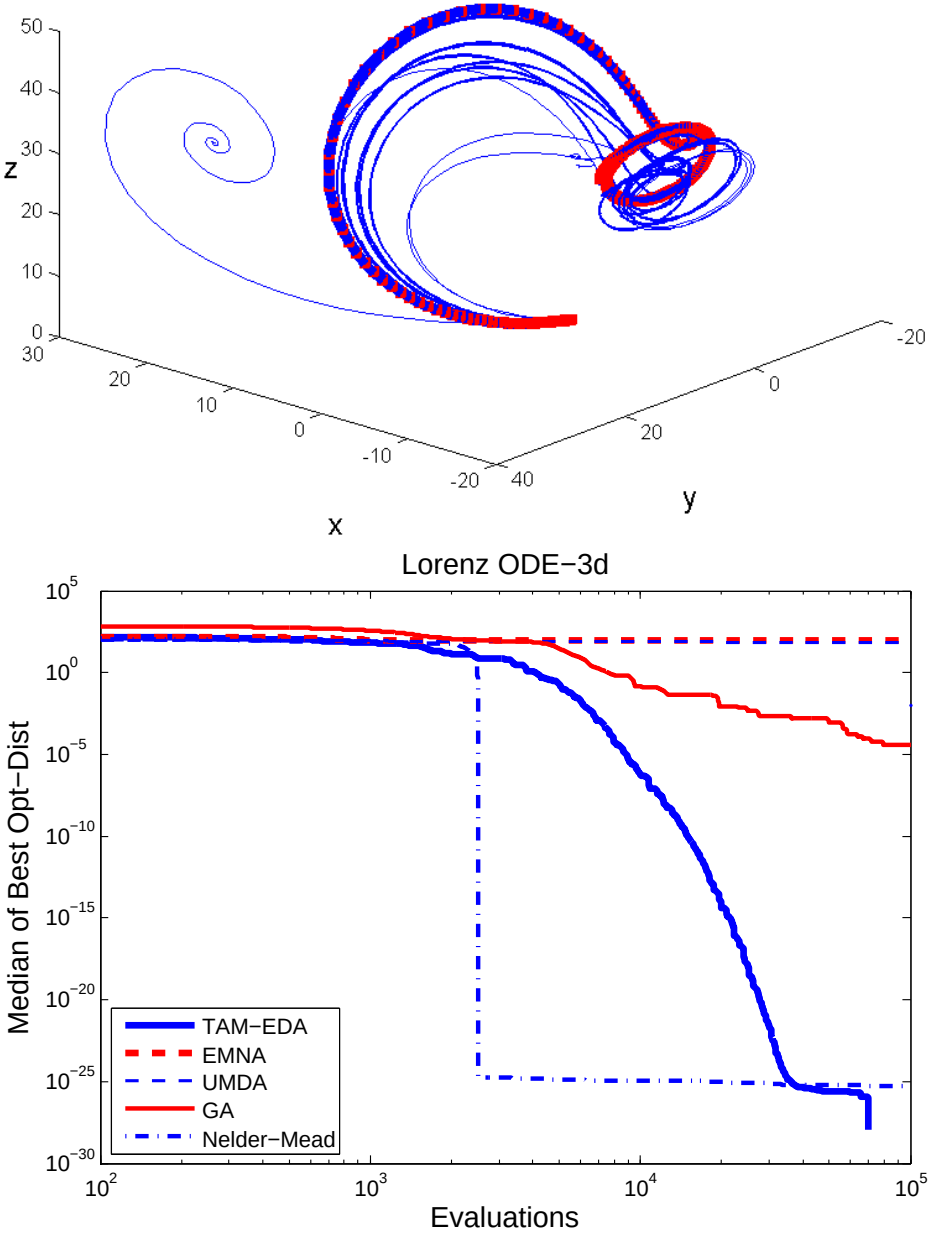

Figure 11: Lorenz model estimation.

The $t = 0$ to $t = 3$ trajectory of the best-fitting model obtained using TAM-EDA is plotted in Figure 11 (top). Progress in opt-dist is plotted versus number of cost function evaluations for all algorithms in Figure 11 (bottom). After 8×10^4 evaluations, the median TAM-EDA opt-dist value was zero and hence could no longer be plotted on a log scale. The true parameters were found to the limits of Matlab's double precision, that is, 15–16 decimal digits.

The Nelder–Mead method achieved excellent results sooner than the other methods: after about 2,500 function evaluations. The GAs made steady, but slower progress, but UMDA and EMNA failed to make progress on this problem.

4.3 Transistor design problem

Dimmer and Cutteridge [4] described a a transistor modelling problem which was studied by many others [15, 17]. This problem reduces to minimising a sum of squares involving a set of nine dimensional nonlinear equations. The objective function to be minimized is

$$f(\mathbf{x}) = \delta^2 + \sum_{k=1}^{4} (\alpha_k^2 + \beta_k^2)\,, \tag{7}$$

where

$$\begin{aligned}
\alpha_k &= (1 - x_1 x_2) x_3 \exp\left[x_5(g_{1k} - 10^{-3} g_{3k} x_7 - 10^{-3} g_{5k} x_8) - 1\right] - g_{5k} + g_{4k} x_2\,,\\
\beta_k &= (1 - x_1 x_2) x_4 \exp\left[x_6(g_{1k} - g_{2k} - 10^{-3} g_{3k} x_7 + 10^{-3} g_{4k} x_9) - 1\right]\\
&\quad - g_{5k} x_1 + g_{4k}\,,\\
\delta &= x_1 x_3 - x_2 x_4\,,
\end{aligned}$$

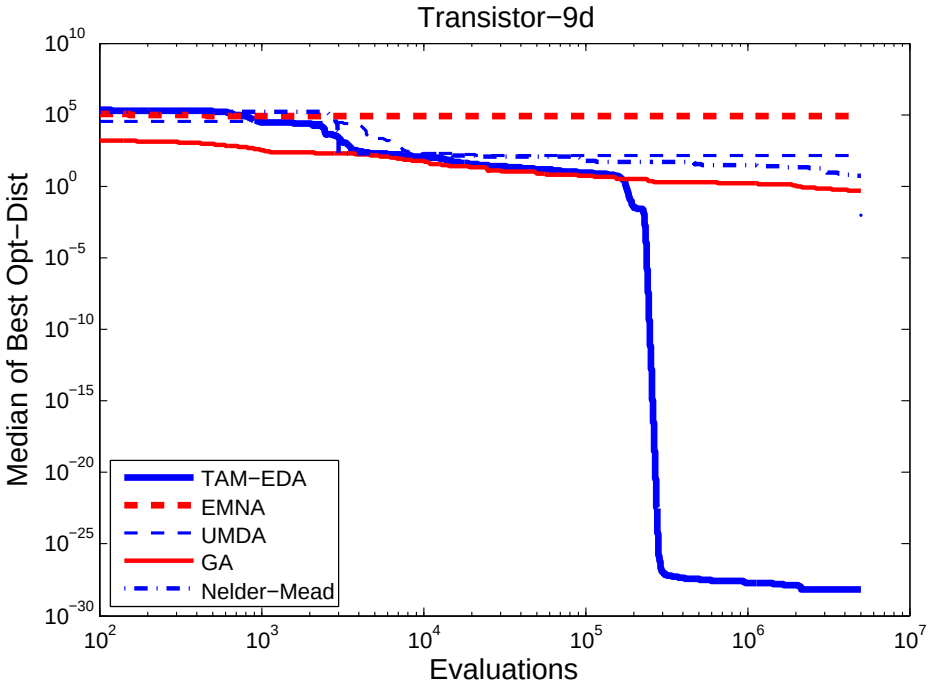

Figure 12: Transistor design problem.

and constant matrix

$$g = \begin{bmatrix} 0.49 & 0.75 & 0.87 & 0.98 \\ 0.37 & 1.25 & 0.70 & 1.46 \\ 5.21 & 10.07 & 22.93 & 20.22 \\ 23.30 & 101.78 & 111.46 & 191.27 \\ 28.51 & 111.85 & 134.39 & 211.48 \end{bmatrix}.$$

The number of evaluations used was 5×10^6 for all the algorithms. The opt-dist is plotted in Figure 12. The best solution is obtained by TAM-EDA and has an objective value of 5.4×10^{-29}, and the median objective value in 31 runs was 6.5×10^{-29}. The best solution found by Price [17] with a

controlled random search was 3.9×10^{-4} in 3×10^4 function evaluations, and Pant et al. [15] were unable to match this using a particle swarm method. The GA produced a median objective value of 0.19, while UMDA and EMNA were unable to find a solution with a function value lower than 10.

5 Conclusion

We proposed a new type of derivative-free EDA, namely TAM-EDA, which uses a multivariate t distribution model, maintains an archive of the best solutions seen so far, and utilises mutation operations to allow exploration throughout all stages of algorithm convergence. This article also introduced a mutation operation (DMO) specifically for TAM-EDA to expand the searching range. Experiments were conducted to compare the proposed algorithm with generic EDAs to two widely used derivative-free alternatives: genetic algorithms and the Nelder–Mead method. TAM-EDA produced the best median results on each of five test problems, although the number of iterations required for this varied widely. Some insights into the behaviour of the algorithm were obtained through the use of heatmaps to track the distance from the global optimum in each dimension.

References

[1] Alfi, A. Particle Swarm Optimization Algorithm with Dynamic Inertia Weight for Online Parameter Identification Applied to Lorenz Chaotic System, *International Journal of Innovative Computing, Information and Control* 8:1191–1203, 2012.
http://www.ijicic.org/ijicic-10-04102.pdf C738

[2] Baluja, S. Population-Based Incremental Learning: A Method for Integrating Genetic Search Based Function Optimization and

Competitive Learning, Technical Report CMU-CS-94-163, Carnegie Mellon University, 1994. http://www.ri.cmu.edu/pub_files/pub1/baluja_shumeet_1994_2/baluja_shumeet_1994_2.pdf C723

[3] Costa, A., Jones, O. D. and Kroese, D. Convergence properties of the cross-entropy method for discrete optimization. *Operations Research Letters* 35.5: 573-580, 2007. doi:10.1016/j.orl.2006.11.005 C723

[4] Dimmer, P. R. and Cutteridge, O. P. D. Second derivative Gauss-Newton-based methods for solving nonlinear simultaneous equations. *Electronics Letters* 10:182–184, 1980. doi:10.1049/ip-g-1.1980.0047 C740

[5] Kirkpatrick, S., Gelatt, C. D. and Vecchi, M. P. Optimization by Simulated Annealing. *Science* 220:671–680, 1983. doi:10.1126/science.220.4598.671 C728

[6] Lagarias, J. C., Reeds, J. A., Wright, M. H., and Wright, P. E. Convergence Properties of the Nelder–Mead Simplex Method in Low Dimensions. *SIAM Journal of Optimization* 9(1):112–147, 1998. doi:10.1137/S1052623496303470 C729

[7] Lange, K. L., Little, R. J. A. and Taylor, J. M. G. Robust statistical modeling using the t distribution. *Journal of the American Statistical Association*, 84(408):881–896, 1989. http://www.jstor.org/stable/2290063 C724

[8] Larrañaga, P., Lozano, J. A. and Bengoetxea, E. Estimation of Distribution Algorithms Based on Multivariate Normal and Gaussian Networks. Techincal Report KZZA-IK-1-01, Department of Computer Science and Artificial Intelligence, University of the Basque Country, Spain, 2001. http://www.sc.ehu.es/acwbecae/ikerkuntza/these/Ch4.pdf C723

[9] Lee, S. and Wright, S. J. Decomposition algorithm for training large-scale semiparametric support vector machines. In European

Conference on Machine Learning Proceedings Part II, *Lecture Notes in Computer Science* 5782 pp. 1–14, Springer, Berlin, 2009. doi:10.1007/978-3-642-04174-7_1 C728

[10] Liu, C. and Rubin, D. B. ML estimation of the t distribution using EM and its extensions, ECM and ECME. *Statistica Sinica*, 5(1):19–39, 1995. http://www3.stat.sinica.edu.tw/statistica/oldpdf/A5n12.pdf C724

[11] Margolin, L. On the convergence of the cross-entropy method. *Annals of Operations Research* 134(1):201–214, 2005. doi:10.1007/s10479-005-5731-0 C723

[12] Martí, L., García, J., Berlanga, A., Coello Coello, C. A. and Molina, J. M. On current model-building methods for multi-objective estimation of distribution algorithms: Shortcomings and directions for improvement. Department of Informatics, Universidad Carlos III de Madrid, Madrid, Spain, Tech. Rep. GIAA2010E001,2010. http://www.giaa.inf.uc3m.es/miembros/lmarti/_media/papers%3Bmarti-et-al--2010--model-building-tech-rep.pdf C723

[13] Mühlenbein, H. and Paass, G. From recombination of genes to the estimation of distributions I. Binary parameters. *Parallel Problem Solving from Nature-PPSN IV*, 178–187, Springer-Verlag, 1996. doi:10.1007/3-540-61723-X_982 C723

[14] Mühlenbein, H. The equation for response to selection and its use for prediction, *Evolutionary Computation* 5:303–346, 1997. doi:10.1162/evco.1997.5.3.303 C723

[15] Pant, M., Thangaraj, R. and Abraham, A. Particle Swarm Based Meta-Heuristics for Function Optimization and Engineering Applications. *Proceedings of the 7th International Conference on Computer Information Systems and Industrial Management Applications* pp. 84–90, IEEE, Piscataway, NJ, 2008. doi:10.1109/CISIM.2008.33 C740, C742

[16] Posik, P. BBOB-benchmarking a simple estimation of distribution algorithm with Cauchy distribution. In Rothlauf, F. ed., *Proceedings of the Genetic and Evolutionary Computation Conference GECCO 2009* pp. 2309–2314, ACM, New York, 2009. doi:10.1145/1570256.1570322 C723

[17] Price, W. L. A Controlled Random Search Procedure for Global Optimization *The Computer Journal* 20:367–370, 1977. doi:10.1093/comjnl/20.4.367 C740, C741

[18] Robert, C. and Casella, G. *Monte Carlo Statistical Methods*, 2nd ed., Springer, New York, 2005. http://www.springer.com/statistics/statistical+theory+and+methods/book/978-0-387-21239-5 C728

[19] Rubinstein, R. Y. Optimization of Computer simulation Models with Rare Events, *European Journal of Operations Research*, 99:89–112, 1997. doi:10.1016/S0377-2217(96)00385-2 C723

[20] Rubinstein, R. Y. and Kroese, D. P. *The cross-entropy method: a unified approach to combinatorial optimization, Monte-Carlo simulation and machine learning*. Springer, 2004. http://www.springer.com/computer/theoretical+computer+science/book/978-0-387-21240-1 C723

[21] Zhang, Q. and Muhlenbein, H. On the convergence of a class of estimation of distribution algorithms. IEEE Transactions on Evolutionary Computation, 8(2):127–136, 2004. doi:10.1109/TEVC.2003.820663 C723

Author addresses

1. **B. Gao**, School of Mathematics and Physics, University of Queensland, Australia.
mailto:bo.gao@uqconnect.edu.au

2. **I. A. Wood**, School of Mathematics and Physics, University of Queensland, Australia.
mailto:i.wood1@uq.edu.au