Estimation Multivariate Normal Algorithm with Thresheld Convergence

Dania Tamayo-Vera
School of Mathematics and Computer Science
University of Havana
Havana, Cuba
d.tamayo@lab.matcom.uh.cu

Antonio Bolufé-Röhler
School of Mathematics and Computer Science
University of Havana
Havana, Cuba
bolufe@matcom.uh.cu

Stephen Chen
School of Information Technology
York University
Toronto, Canada
sychen@yorku.ca

***Abstract*—Estimation of Distribution Algorithms (EDAs) use a subset of solutions from the current population to build a distribution function from which the next generation of solutions is created. If there is poor diversity in the current population, then there is poor diversity in the subset of solutions selected from it and in the next generation that is created from it. Like many metaheuristics, EDAs can suffer from an autocatalytic process in which convergence begets more convergence. In Thresheld Convergence, convergence is "held" back by a threshold function, and this new technique has been successfully applied to other metaheuristics to prevent autocatalytic convergence from cascading into premature convergence. In this paper, Thresheld Convergence is applied to Estimation Multivariate Normal Algorithm with a key difference: convergence is controlled in the parameter space instead of the search space. Computational results show that significant improvements can be achieved across a broad range of multi-modal functions.**

Keywords—estimation multivariate normal algorithm; Thresheld Convergence; multi-modal search spaces.

I. INTRODUCTION

An attraction basin represents the region of a search space within which greedy local search will always converge to the same local optimum. The fitness of an attraction basin can be defined as the fitness of this optimum. A multi-modal search space is thus divided into multiple attraction basins, and the effectiveness of an applied metaheuristic strongly depends on its ability to first detect the fittest basins (exploration) before then converging to their corresponding local optima (exploitation).

Many heuristic optimization techniques frequently used for multi-modal search spaces had their search mechanisms initially developed/conceptualized for unimodal search spaces. For example, Particle Swarm Optimization (PSO) begins with a cornfield vector (see Section 3.2 in [1]) and Differential Evolution (DE) builds its foundation from a simple unimodal cost function (see Figure 1 in [2]). As a consequence, the primary search mechanisms of these metaheuristics are biased towards converging on local optima rather than searching for the best attraction basins. However, finding the fittest solutions for multi-modal optimization problems depends on the efficacy of exploration and its prevalence over exploitation [3].

Estimation of Distribution Algorithms (EDAs) [4] also face this challenge. The efficacy of exploration in EDAs relies strongly on the diversity of their populations. However, by selecting the best solutions to estimate the parameters of the distribution, EDAs rapidly focus on the best found attraction basins. Once the algorithm starts to converge, the distribution functions start showing a smaller deviation with respect to their mean. With a smaller variance, new candidate solutions are generated closer to each other. This further decreases the deviation of the distribution which produces even closer solutions in the next generation. The autocatalytic nature of this process leads to a rapid convergence which may occur before the intended stopping condition is reached, i.e. the algorithm may converge prematurely.

Thresheld Convergence (TC) is a recently developed diversification technique which attempts to separate the processes of exploration and exploitation [5]. TC uses a threshold function to establish a minimum search step, and managing this step makes it possible to influence the transition from exploration to exploitation. The goal of a controlled transition is to avoid an early concentration of the population around a few attraction basins and thus avoid the loss of diversity which can cause premature convergence. Thresheld Convergence has been successfully applied to several population-based metaheuristics such as Particle Swarm Optimization [6], Differential Evolution [7], Evolution Strategies (ES) [8], and Minimum Population Search (MPS) [9]. This paper applies TC to the Estimation Multivariate Normal Algorithm for continuous search spaces (EMNA) [4].

EMNA is an Estimation Distribution Algorithm which uses a multivariate normal distribution to generate the new solutions. The parameters of this distribution are estimated from the best solutions of a given generation, and the next generation is obtained by simulating the distribution with these parameters. We hypothesize that by regulating these parameters, it will be possible to (indirectly) control the distance (minimum search step) between the newly generated solutions, and this will allow us to reduce the risk of premature convergence for EMNA on multimodal functions.

The remainder of this paper is organized as follows. The next section gives a background on Thresheld Convergence and the Estimation Multivariate Normal Algorithm. Section III

978-1-5090-0623-6/16/$31.00 ©2016 IEEE

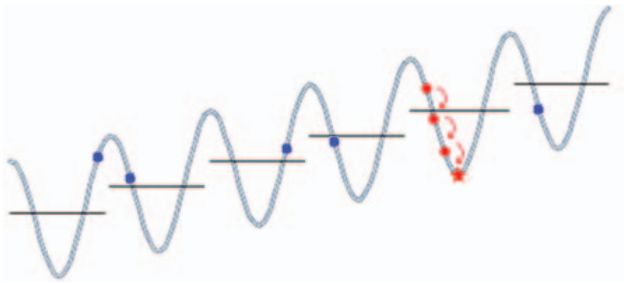

Fig. 1. A solution found through exploitation (red star) may appear more promising than random solutions (blue circles) from better attraction basin.

presents an analysis of EMNA's convergence on multimodal functions and describes the integration of TC into the algorithm. Results on the CEC'13 benchmark functions are presented in Section IV. Finally, a discussion about the new metaheuristic is carried out in Section V before the paper is summarized in Section VI.

II. Background

A. Thresheld Convergence

Thresheld Convergence is a technique designed to promote diversification and an unbiased exploration of the search space. TC is built upon the hypothesis that a concurrent execution of exploration and exploitation may bias the exploration process. Consider a multi-modal function where all of the attraction basins have similar shapes and sizes – e.g. the popular Rastrigin function. If exactly one random solution could be sampled from each attraction basin, then there is a reasonable expectation that the fittest solutions will represent the fittest attraction basins. The process of detecting the fittest attraction basins (exploration) would then consist of selecting the fittest sample solutions. However, if some attraction basins are over-sampled (e.g. due to exploitation), then the chances that worse attraction basins are represented by fitter solutions will increase, and this can bias the exploration towards the initially found (less fit) regions of the search space (see Fig. 1).

The design of many metaheuristics does not take this biasing of exploration into account. Specifically, large explorative and small exploitative steps are often indistinguishably made during the early (explorative) stage of the search process. By mistaking exploitation for exploration, inferior attraction basins can become represented by superior solutions, and this can cause a metaheuristic to over estimate the promise of the represented attraction basins. Thresheld Convergence aims to address this weakness by controlling the distance (search step) between a parent and its offspring solution. If the size of the search step is large enough, then the new sample solutions are more likely to represent different attraction basins [5].

The minimum search step (threshold) is decreased as the search progresses – convergence is "held" back until the last stages of the search process. The threshold is initially set to a fraction of the search space diagonal, and it is updated over the execution of a metaheuristic by following a decay rule (1). In (1), d is the main diagonal of the search space, *totalFEs* is the total number of function evaluations, and *FEs* is the amount of evaluations performed so far. The parameter α determines the initial threshold and γ controls the decay rate.

$$threshold = \alpha * d * ([totalFEs - FEs] / totalFEs)^{\gamma} \tag{1}$$

B. Estimation Multivariate Normal Algorithm

Estimation of distribution algorithms (EDA) [4] are stochastic optimization algorithms which explore the search space by sampling an explicit probabilistic model. The model is constructed from promising, previously found solutions, and it can be adapted to learn the structure of the problem. Thus, EDAs transform the optimization problem into a search over probability distributions. As the search proceeds, the parameters of the probability distributions are updated according to the best individuals of each generation.

The Estimation Multivariate Normal Algorithm (EMNA) uses a multivariate normal distribution to sample new solutions. The first population is randomly initialized with a uniform distribution over the entire search space. During each generation, the best solutions are selected from the current population and their mean μ and covariance matrix Σ are calculated. Using these estimators, the inverse multivariate Gaussian function is generated and sampled to obtain the next population. A general outline is presented in Algorithm 1.

The EMNA used in this paper is a standard implementation using a multivariate Gaussian density function and truncation selection. The population size is $p = 50n$ where n is the dimension of the search space (as recommended in [4]). The selection coefficient is $c = 0.3$, i.e. the best *30%* of the solutions are selected from each generation to learn the probabilistic model. A maximum number of function evaluations is used as the stopping criteria.

III. Estimation Multivariate Normal Algorithm with Thresheld Convergence

In this section, we analyze how EMNA can be affected by premature convergence and how TC can be used to address this weakness. For this initial analysis, we focus on the well-known multi-modal Rastrigin function (2) which is separable, symmetric, well-conditioned, and which has attraction basins of similar size and shape. We specifically use the rotated Rastrigin implementation provided in the CEC 2013 benchmark (function No. 12) [10]. Results are reported for n=30 dimensions and, as suggested by the benchmark, the

Algorithm 1 EMNA

```
Initialize population P_0 randomly
Evaluate P_0
t=0
Repeat
   Select set of best solutions S_t from P_t
   [μ, Σ] = EstimateParameters(S_t)
   P_{t+1} = SampleNormalMultivariate(μ, Σ)
   Evaluate P_{t+1}
   t=t+1
Until Stopping Criteria
```

stopping criteria is to have a maximum number of evaluations *totalFEs=10,000n*.

$$f(x) = 10n + \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i)] \qquad (2)$$

A. EMNA's Convergence

After each generation of EMNA, the best *30%* of the current population is selected to estimate the parameters μ and Σ from a normal multivariate distribution. The new population is sampled from this distribution. Since the best solutions will be located near to each other in a globally convex search space such as Rastrigin, the standard deviation of the estimated distribution becomes smaller with each generation. Sampled from a distribution with a smaller deviation, the new set of solutions will become even closer together. These close solutions will have an even smaller deviation from the mean, and this leads to even closer solutions in the next generation. This autocatalytic process, where concentration begets more concentration, is an important mechanism which allows EMNA to converge. However, it can also lead to a "cascading convergence" which may cause premature convergence.

This effect can be observed in two different spaces: the parameter space (where the norm of Σ becomes smaller with each generation) and the search space (where solutions become closer to each other). Fig. 2a shows how the Euclidian norm decreases as the search progresses. It can be noticed that the rate of decrease is exponential, and that near generation 80 (from a total of 200 allotted generations) its value is already close to zero. As a consequence, the average distance from the newly sampled solutions to their mean μ also decreases rapidly, see Fig. 2b.

These results present evidence that EMNA converges before consuming less than *50%* of the available function evaluations. This premature convergence reduces the effectiveness of EMNA. Fig. 2c shows the error of the best solution with respect to the global optimum.

B. EMNA's Convergence with Thresheld Convergence

Thresheld Convergence aims to influence the rate of convergence by implementing a minimum search step between the parents and offspring solutions. In algorithms such as Differential Evolution, this minimum step can be established directly in the search space, e.g. by "pushing" outwards the *new* solutions which are too close to their *base* solutions [7]. However, in EDAs, there is no parent-offspring relation between pairs of solutions. Further, arbitrarily pushing some solutions may alter the distribution of the new solutions with respect to the expected distribution of the sampled probability model.

Taking these considerations into account, we decided to apply Thresheld Convergence not on the search space but on the parameter space instead. The value provided by the threshold function (1) is not used to enforce a minimum distance among solutions, but it is instead used to fix a "minimum measure" in the parameter space. More precisely, it is used to set the (Euclidian) norm of the matrix Σ. As it can be seen in Fig. 2, the norm of the covariance matrix holds a direct relation to the distance among newly generated solutions with respect to their mean, and subsequently to the convergence rate of the overall algorithm.

The addition of Thresheld Convergence to EMNA (EMNA-TC) follows a simple logic. In each generation, after calculating μ and Σ, we modify Σ to adjust its norm to the threshold value provided by (1). First we normalize Σ (dividing it by its norm) and then we multiply Σ by the threshold value. To ensure that the initial threshold value matches the norm of the first covariance matrix, the $\alpha \times$*diagonal* value is replaced in (1) by the norm of the first covariance matrix Σ.

Fig. 3 shows the performance of EMNA-TC for the Rastrigin function. As it can be seen in Fig. 3a, the norm of Σ decays according to the threshold function (1), and it reaches zero only during the final generations. The distance among the newly sampled solutions also decays in a controlled manner, and this helps prevent premature convergence, see Fig. 3b. As a result of this controlled convergence, EMNA-TC achieves a much smaller error (mean error of 6.6e-2) than EMNA (mean error of 4.2e+1), see Fig. 2c and Fig. 3c.

C. An Adaptive Threshold Function

One goal of controlling the convergence rate is to increase the amount of exploration that occurs when the search step reaches an "ideal search scale". Specifically, the ideal case for Thresheld Convergence is to have one sample solution from each attraction basin. This "ideal search scale" is reached when the threshold size is similar to the distance among attraction basins. If the distance among the different attraction basins could be known *a priori*, then the threshold could be held longer at this "ideal search scale". Allocating more function evaluations to this threshold level should allow a better exploration of different basins and increase the chances of finding the best regions before performing a more intense/local search [11].

Algorithm 2 EMNA-TC

```
Initialize population P_0 randomly
Evaluate P_0
t=0
repeat
  Select set of best solutions S_t from P_t
  [μ, Σ] = EstimateParameters(S_t)
  if t==0
    threshold = norm(Σ)
  else
    if best solution has improved
      γ = 0.95 * γ
    else
      γ = 1.05 * γ
    endif
    threshold = UpdateThreshold (γ)          % Eq. (1)
  endif
  Σ = threshold * ( Σ / norm(Σ) )
  P_t+1 = SampleNormalMultivariate(μ, Σ)
  Evaluate P_t+1
  t=t+1
until Stopping Criteria
```

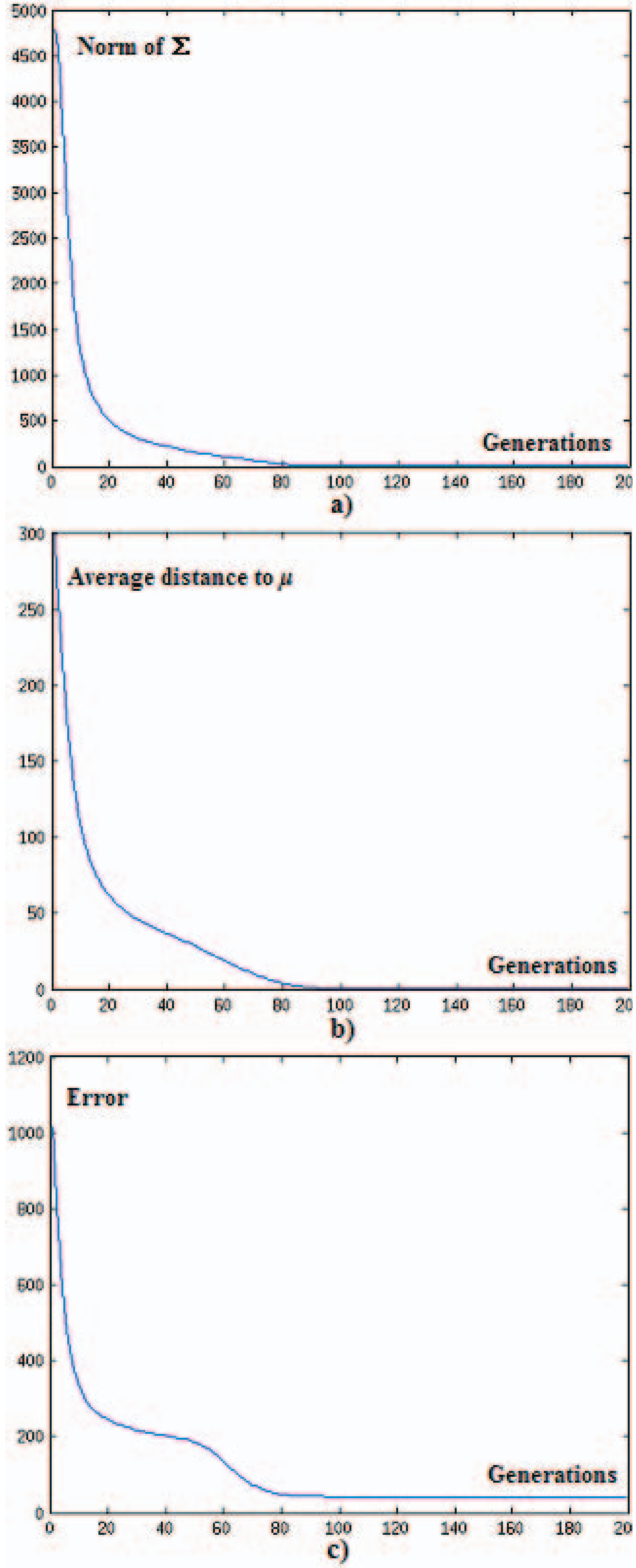

Fig. 2 Convergence of EMNA on the Rastrigin function.

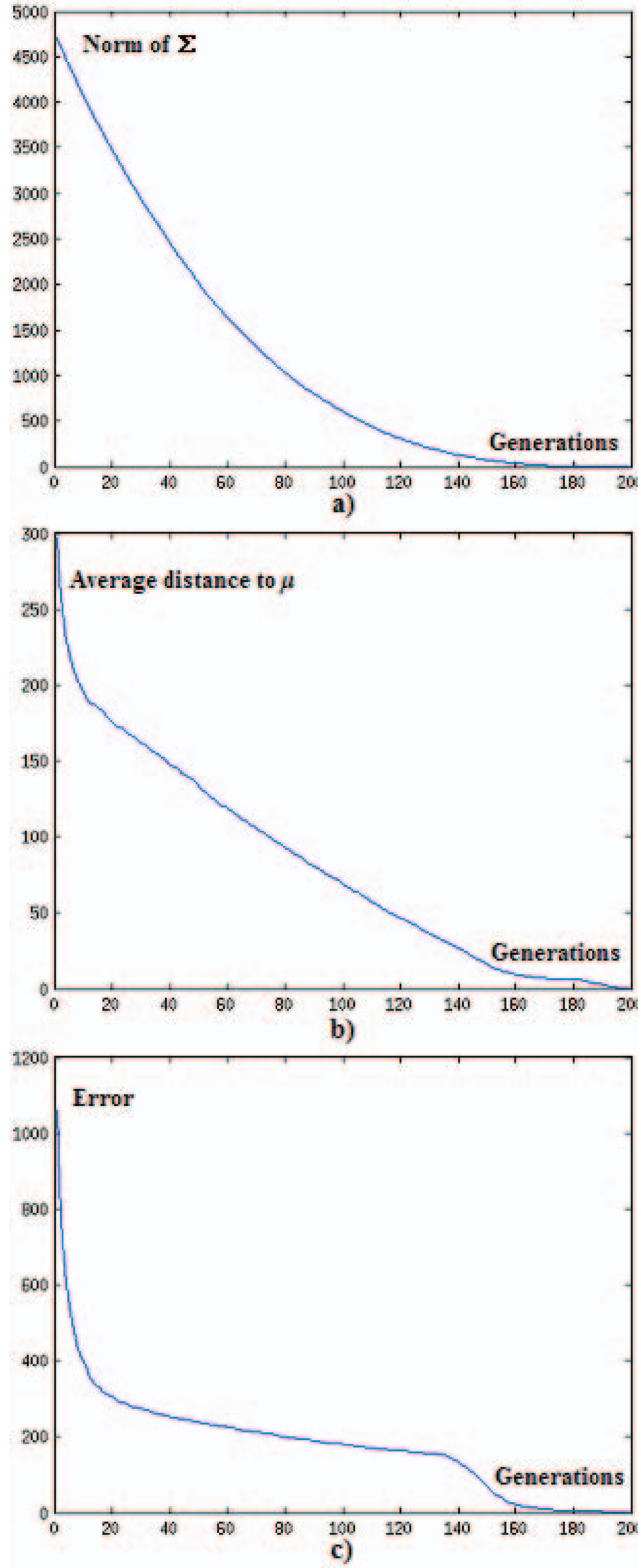

Fig. 3 Convergence of EMNA with Thresheld Convergence on the Rastrigin function

Determining the "ideal search scale" is a difficult task. Previous work in Differential Evolution [7] and Minimum Population Search [12] has focused on the proportion of new solutions which survive to form the new population as an indicator of whether exploration is operating near the "ideal search scale". Since in EMNA the population is entirely replaced by the new sampled solutions, a different criterion needs to be defined. We use a simple strategy to determine whether the convergence rate needs to be increased or decreased: if the best found solution improves from the previous generation, it suggests that exploration is paying off and convergence should be delayed; otherwise, the convergence is accelerated to promote exploitation.

In the threshold function (1), the γ parameter controls the convergence rate. It is possible to control the convergence speed by adaptively adjusting γ. Thus, if the best solution is improved, then γ is decreased by multiplying it with 0.95 (which leads slower convergence); otherwise, γ is increased by multiplying it with 1.05. A general outline of EMNA-TC with adaptive threshold is presented in Algorithm 2.

Fig. 3 compares the decay rate of the fixed threshold of Eq. (1) vs. the adaptive threshold presented in this section. The adaptive threshold is shown for Rastrigin (F12) and the Weierstrass (F9) functions. All thresholds start with a similar initial value, which is the norm of the covariance matrix of the random initial population. For the adaptive threshold the value decreases faster or slower according to the differing topologies of the current search space.

IV. Experimental Results

To compare the improvement achieved by adding Thresheld Convergence to EMNA, both algorithms are tested on the standard CEC 2013 benchmark [10]. This benchmark consists of a set of 28 unimodal and multi-modal functions with various characteristics. The functions are divided in three sets: unimodal functions (1 to 5), basic multi-modal functions (6 to 20) and composite multi-modal functions (21 to 28). Since Thresheld Convergence has been specifically

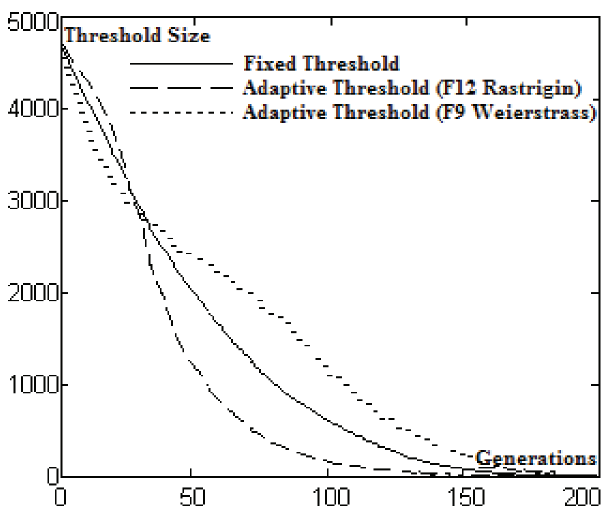

Fig. 3 Fixed and adaptive threshold functions.

TABLE I
Comparison between EMNA and EMNA-TC

	EMNA		EMNA-TC			*t* -
Fun.	mean	std. dev.	mean	std. dev.	%-diff	test
F_1	6.83e+03	1.31e+03	7.00e−05	1.46e−05	**99.9%**	**0.00**
F_2	5.07e+06	3.68e+06	9.62e+06	6.40e+06	**47.3%**	**0.00**
F_3	3.15e+12	4.49e+12	1.04e+13	1.74e+13	-69.9%	**0.01**
F_4	8.56e+03	2.36e+03	7.74e−05	2.31e−05	**99.9%**	**0.00**
F_5	2.84e+02	1.08e+02	4.69e+01	1.08e+02	**83.5%**	**0.00**
F_1-F_5					33.2%	
F_6	9.01e+02	1.78e+02	2.30e+01	7.67e−01	**97.5%**	**0.00**
F_7	4.93e+02	3.93e+02	1.37e+03	1.22e+03	-64.1%	**0.00**
F_8	2.09e+01	4.52e−02	2.10e+01	4.97e−02	-0.1%	0.24
F_9	1.07e+01	1.71e+00	4.04e+00	1.50e+00	**62.2%**	**0.00**
F_{10}	6.51e+02	1.86e+02	5.68e−04	6.19e−05	**99.9%**	**0.00**
F_{11}	4.18e+01	1.09e+01	7.00e−01	6.96e−01	**98.3%**	**0.00**
F_{12}	4.22e+01	9.51e+00	3.99e−02	1.97e−01	**99.9%**	**0.00**
F_{13}	9.76e+01	2.11e+01	9.96e−02	3.02e−01	**99.9%**	**0.00**
F_{14}	8.51e+02	3.64e+02	6.47e+02	3.19e+02	**24.0%**	**0.01**
F_{15}	7.09e+02	3.05e+02	4.44e+02	1.99e+02	**37.4%**	**0.00**
F_{16}	2.45e+00	2.48e−01	8.16e−02	1.68e−01	**96.7%**	**0.00**
F_{17}	3.81e+01	9.72e+00	3.79e+01	9.34e+00	0.7%	0.89
F_{18}	1.46e+02	3.02e+01	4.53e+01	1.52e+01	**69.0%**	**0.00**
F_{19}	2.52e+02	1.91e+02	2.64e+00	5.72e−01	**98.9%**	**0.00**
F_{20}	1.39e+01	1.26e+00	1.49e+01	5.98e−01	-6.4%	**0.00**
F_6-F_{20}					54.2%	
F_{21}	6.82e+02	1.25e+02	2.88e+02	3.28e+01	**57.8%**	**0.00**
F_{22}	7.82e+02	2.92e+02	4.18e+02	1.82e+02	**46.6%**	**0.00**
F_{23}	7.41e+02	3.13e+02	5.46e+02	2.55e+02	**26.3%**	**0.00**
F_{24}	2.41e+02	7.12e+00	2.03e+02	4.90e−01	**15.9%**	**0.00**
F_{25}	2.97e+02	9.34e+00	2.01e+02	2.03e−01	**32.2%**	**0.00**
F_{26}	2.93e+02	1.87e+01	2.87e+02	1.93e+01	1.9%	0.16
F_{27}	5.06e+02	3.39e+01	3.29e+02	4.69e+00	**35.1%**	**0.00**
F_{28}	1.63e+03	1.34e+02	3.00e+02	4.19e−02	**81.6%**	**0.00**
F_{21}-F_{28}					37.1%	
F_1-F_{28}					45.6%	

Mean error, standard deviation, relative improvement and p-value from 51 independent trials on the set of CEC'13 benchmark functions.

designed for optimizing multi-modal functions, the second group contains the most interesting functions in terms of landscape characteristics. Results are presented for all the functions according to the experimental setup proposed for the benchmark, i.e. *51* independent trials on each function in $n=30$ dimensions with a maximum of *300,000* function evaluations.

Table I shows a comparison between EMNA and EMNA-TC. In addition to the mean error and standard deviation, the relative difference in the performance of the algorithms is also reported: $(p_1 - p_2)/ max(p_1, p_2)$ where p_1 is the performance of the original EMNA, and p_2 is the performance of EMNA-TC. Positive values indicate by what amount (percent) EMNA-TC outperforms EMNA – negative values indicate the opposite. A standard *t*-test is also performed to measure statistical significance. Results with statistically significant improvements (*p-value* < 0.05) of EMNA-TC over the baseline EMNA are highlighted in bold. It can be seen in Table I that EMNA-TC achieves a significant improvement over the original version of EMNA on 22 of the 28 functions.

As expected, the largest improvements are achieved on the set of basic and composite multi-modal functions with improvements of 54.2% and 37.1% respectively. Interestingly, the integration of TC also leads to an overall improvement of 33.2% on the set of unimodal functions. The

TABLE II
COMPARISON BETWEEN EMNA-TC AND OTHER METAHEURISTICS

Fun.	**EMNA-TC**	**CMA-ES**			**DE**			**PSO**		
	(mean error)	(mean error)	(%-diff)	(*t*-test)	(mean error)	(%-diff)	(*t*-test)	(mean error)	(%-diff)	(*t*-test)
F_1	7.00e−05	**0.00e+00**	-99.9%	0.00	4.17e−07	-99.4%	0.00	0.00e+00	-99.9%	0.00
F_2	9.62e+06	**0.00e+00**	-99.9%	0.00	3.91e+06	-59.3%	0.00	1.87e+06	-80.6%	0.00
F_3	1.04e+13	**2.07e+00**	-99.9%	0.00	2.14e+06	-99.9%	0.00	9.02e+07	-99.9%	0.00
F_4	**7.74e−05**	2.57e+03	99.9%	0.02	2.36e+04	99.9%	0.00	1.67e+04	99.9%	0.00
F_5	4.69e+01	**0.00e+00**	-99.9%	0.00	7.79e−05	-99.9%	0.00	0.00e+00	-99.9%	0.00
F_1-F_5			**-59.9%**			**-51.7%**			**-56.1%**	
F_6	2.30e+01	**7.68e+00**	-66.5%	0.00	1.38e+01	-40.1%	0.00	1.53e+01	-33.5%	0.00
F_7	1.37e+03	3.14e+01	-97.7%	0.00	**2.13e+01**	-98.5%	0.00	6.49e+01	-95.3%	0.00
F_8	2.10e+01	2.15e+01	2.4%	0.00	2.10e+01	0.0%	0.90	**2.09e+01**	-0.2%	0.00
F_9	**4.04e+00**	2.01e+01	79.9%	0.00	3.80e+01	89.4%	0.00	2.84e+01	85.8%	0.00
F_{10}	**5.68e−04**	1.38e−02	95.9%	0.00	5.89e−01	99.9%	0.00	1.19e−01	99.5%	0.00
F_{11}	**7.00e−01**	5.43e+01	98.7%	0.00	6.12e+01	98.9%	0.00	6.32e+01	98.9%	0.00
F_{12}	**3.99e−02**	5.28e+01	99.9%	0.00	2.19e+02	99.9%	0.00	7.97e+01	99.9%	0.00
F_{13}	**9.96e−02**	1.24e+02	99.9%	0.00	2.17e+02	99.9%	0.00	1.36e+02	99.9%	0.00
F_{14}	**6.47e+02**	3.80e+03	83.0%	0.00	2.68e+03	75.9%	0.00	2.60e+03	75.2%	0.00
F_{15}	**4.44e+02**	4.38e+03	89.9%	0.00	7.26e+03	93.9%	0.00	4.12e+03	89.2%	0.00
F_{16}	**8.16e−02**	8.91e+00	99.1%	0.00	2.46e+00	96.7%	0.00	1.63e+00	95.0%	0.00
F_{17}	**3.79e+01**	1.67e+02	77.3%	0.00	1.13e+02	66.4%	0.00	9.82e+01	61.4%	0.00
F_{18}	**4.53e+01**	2.85e+02	84.1%	0.00	2.46e+02	81.6%	0.00	1.72e+02	73.6%	0.00
F_{19}	**2.64e+00**	3.25e+00	18.6%	0.00	1.36e+01	80.5%	0.00	5.72e+00	53.8%	0.00
F_{20}	1.49e+01	1.50e+01	0.8%	0.16	1.29e+01	-13.5%	0.00	**1.17e+01**	-21.3%	0.00
F_6-F_{20}			**51.0%**			**55.4%**			**52.1%**	
F_{21}	2.88e+02	3.19e+02	9.5%	0.07	2.95e+02	2.4%	0.50	**2.31e+02**	-19.9%	0.00
F_{22}	**4.18e+02**	4.16e+03	89.9%	0.00	2.27e+03	81.6%	0.00	3.06e+03	86.4%	0.00
F_{23}	**5.46e+02**	4.81e+03	88.7%	0.00	7.30e+03	92.5%	0.00	4.65e+03	88.3%	0.00
F_{24}	**2.03e+02**	2.41e+02	15.9%	0.00	2.38e+02	14.7%	0.00	2.76e+02	26.7%	0.00
F_{25}	**2.01e+02**	2.66e+02	24.2%	0.00	2.50e+02	19.5%	0.00	2.91e+02	30.8%	0.00
F_{26}	2.87e+02	3.11e+02	7.8%	0.00	2.03e+02	-29.3%	0.00	**2.10e+02**	-26.7%	0.00
F_{27}	**3.29e+02**	7.36e+02	55.4%	0.00	9.77e+02	66.4%	0.00	1.04e+03	68.5%	0.00
F_{28}	3.00e+02	3.81e+02	21.2%	0.07	3.00e+02	-0.1%	0.00	**2.92e+02**	-2.8%	0.15
F_{21}-F_{28}			**39.1%**			**30.9%**			**31.4%**	
F_1-F_{28}			**27.8%**			**29.2%**			**26.8%**	

Mean error from known optimum and relative improvements of EMNA-TC over CMA-ES, DE, and PSO.

fast convergence of EMNA and its center bias (caused by the use of a normal distribution) make it difficult to find the optimum on shifted functions like the Sphere (F1) where the optimum is located far away from the origin. In such cases, slowing convergence can also help on unimodal functions. Over the entire benchmark, EMNA-TC improves EMNA's performance by an average of 45.6%.

To compare the performance of EMNA-TC with other population-based metaheuristics, Table II includes comparisons to Particle Swarm Optimization, Differential Evolution, and Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [13]. CMA-ES is a well-known state-of-the-art algorithm, which uses a multivariate normal distribution for generating new solutions [13]. The CMA-ES code is the MATLAB version 3.61.beta available in [14], the default set of parameters was used. The PSO implementation is a standard version with a ring topology [15], zero initial velocities [16], "Reflect-Z" for particles that exceed the boundaries of the search space [17], and a population size of 50 individuals. The DE method is the highly common and frequently effective variant labeled DE/rand/1/bin [2], and it also uses a population size of 50 individuals.

Results for which EMNA-TC achieves a statistically significant improvement over the other three algorithms are highlighted in bold. It can be seen that the performance of EMNA-TC is relatively poor on the set of unimodal functions, achieving the best performance on only one of the five functions. On the other hand, EMNA-TC performs very well on the sets of multi-modal functions, achieving the best results on 16 of the 22 functions. Especially good results are achieved on the basic multi-modal functions where EMNA-TC outperforms the other metaheuristics with an average improvement of over 50%.

To further extend the analysis on the set of basic multi-modal functions, Fig. 4 presents a boxplot of the recorded errors. The results from Fig. 4 show that EMNA-TC achieves a robust performance with a low standard deviation on most functions. Especially noticeable are the results on the functions Griewank (F_{10}), rotated Rastrigin (F_{12}), and non-continuous rotated Rastrigin (F_{13}), for which the median errors are respectively 5.5e-04, 5.7e-05, and 6.6e-05.

V. DISCUSSION

Estimating the probability distribution from the best solutions and using a multivariate Gaussian density functions to generate new solutions encourages the new solutions to crowd around the best found regions. This search mechanism guarantees the convergence of EMNA, but since it is uncorrelated to the stopping criteria, it may also lead to premature convergence (e.g. the ineffective use of the allotted function evaluations). A premature convergence reduces the exploration of the algorithm and its ability to detect the fittest attraction basins. The integration of TC into EMNA allows extending the exploration process as it promotes a distinct separation between exploration and exploitation.

Previous work has applied the threshold value provided by TC directly to the search space, e.g. by directly modifying the solution's position (e.g. pushing or reflecting the solutions towards the threshold edge). However, this approach may be counterproductive with EDAs since the new solutions may no longer correspond to the sampled distribution function. In this paper, TC is applied on the parameter space of the distribution function, and thus it indirectly influences the size of the search steps involved with generating new solutions. This approach, which has shown to be effective (reaching almost 50% in overall improvement versus the original version), may allow migrating the concept of Thresheld Convergence towards combinatorial optimization.

VI. SUMMARY

Thresheld Convergence provides a simple yet effective mechanism to prevent premature convergence in population-based heuristics. Its integration into different metaheuristic has led to significantly improved performance on multimodal functions. This paper presents the first integration of TC into an EDA. Adapting a metaheuristic to the concept of Thresheld Convergence is not a simple mechanical process, but it requires an analysis of the characteristics of the given metaheuristic. The key contribution of this paper is that for the first time, TC is applied to the parameter space instead of the search space. The paper also presents a new design for an adaptive threshold function. When paired with Thresheld Convergence, the performance of the Estimation Multivariate Normal Algorithm greatly improves across a broad range of multi-modal functions. These results make TC a promising technique for future research on Estimation Distribution Algorithms.

REFERENCES

[1] J. Kennedy and R. Eberhart, "Particle swarm optimization," in IEEE International Conference on Neural Networks, pp. 1942–1948, 1995.

[2] R. Storn and K. Price, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces," Journal of Global Optimization, vol. 11, no. 4, pp. 341–359, 1997.

[3] M. Crepinsek, S.-H. Liu, and M. Mernik, "Exploration and exploitation in evolutionary algorithms: A survey," ACM Computing Surveys, vol. 45, no. 3, pp. 35:1–35:33, 2013.

[4] P. Larrañaga and J. A. Lozano, "Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation," Kluwer Academic Publishers, pp. 181–193, 2011.

[5] S. Chen, J. Montgomery, A. Bolufé-Röhler and Y. González-Fernández, "A Review of Thresheld Convergence," GECONTEC, vol. 3 (1), pp. 1–13, 2015.

[6] S. Chen and J. Montgomery, "Particle Swarm Optimization with Thresheld Convergence," IEEE CEC, pp. 510–516, 2013.

[7] A. Bolufé-Röhler, S. Estévez-Velarde, A. Piad-Morffis, S. Chen, and J. Montgomery, "Differential Evolution with Thresheld Convergence," IEEE CEC, pp. 40–47, 2013.

[8] A. Piad-Morffis, A. Bolufé-Röhler, S. Estévez-Velarde, J. Montgomery and S. Chen, "Evolution Strategies with Thresheld Convergence," IEEE CEC, pp. 2097–2104, 2015.

[9] A. Bolufé-Röhler and S. Chen, "Minimum population search - lessons from building a heuristic technique with two population members," IEEE CEC, pp. 2061–2068, 2013.

[10] J. Liang, B. Qu, P. Suganthan, and A. G. Hernandez-Dıaz, "Problem definitions and evaluation criteria for the cec 2013 special session on real-parameter optimization," Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Nanyang Technological University, Singapore, Technical Report, vol. 201212, 2013.

[11] Y. Gonzalez-Fernandez and S. Chen, "Identifying and exploiting the scale of a search space in particle swarm optimization," GECCO, pp. 17–24, 2014.

[12] A. Bolufé-Röhler, and S. Chen, "Extending Minimum Population Search towards Large Scale Global Optimization," IEEE CEC, pp. 845–852, 2014.

[13] N. Hansen, S. D. Muller, and P. Koumoutsakos, "Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES)," *Evolutionary Computation,* vol. 11(1), pp. 1–18, 2003.

[14] https://www.lri.fr/~hansen/cmaes_inmatlab.html

[15] D. Bratton and J. Kennedy, "Defining a standard for particle swarm optimization," IEEE SIS, pp. 120–127, 2007.

[16] A. Engelbrecht, "Particle swarm optimization: velocity initialization," in Proc. IEEE CEC, pp. 70–77, 2012.

[17] S. Helwig, J. Branke, and S. Mostaghim, "Experimental Analysis of Bound Handling Techniques in Particle Swarm Optimization," IEEE Trans. Evolutionary Computation, vol. 17, pp. 259–271, Apr. 2013.

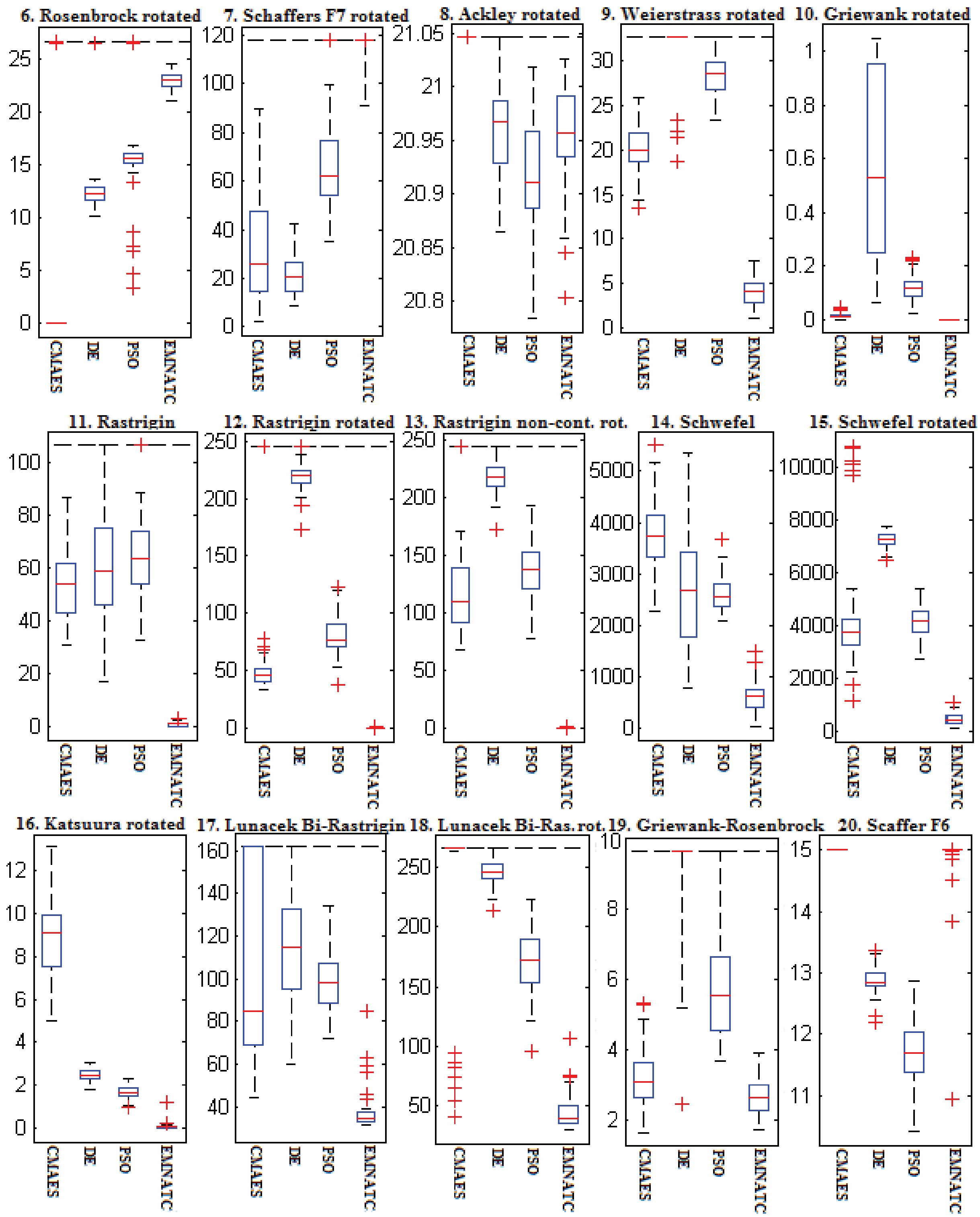

Figure 4.Boxplots for the basic multi-modal functions for the CEC'13 benchmark