Chapter 8

Experimental Results in Function Optimization with EDAs in Continuous Domain

E. Bengoetxea T. Miquélez
Department of Computer Architecture and Technology
University of the Basque Country
{endika, teresa}@si.ehu.es

P. Larrañaga J.A. Lozano
Department of Computer Science and Artificial Intelligence
University of the Basque Country
{ccplamup, lozano}@si.ehu.es

Abstract This chapter shows experimental results of applying continuous Estimation of Distribution Algorithms to some well known optimization problems. For this UMDA$_c$, MIMIC$_c$, EGNA$_{BIC}$, EGNA$_{BGe}$, EGNA$_{ee}$, EMNA$_{global}$, and EMNA$_a$ algorithms were implemented. Their performance was compared to such of Evolution Strategies (Schwefel, 1995). The optimization problems of choice were Summation cancellation, Griewangk, Sphere model, Rosenbrock generalized, and Ackley.

Keywords: Estimation of Distribution Algorithms, Gaussian networks, function optimization, continuous domain

1. Introduction

The aim of this chapter is to show the results of applying Estimation of Distribution Algorithms (EDAs) in continuous domain on some well known optimization problems. Evolution Strategies (ESs) (Schwefel, 1995) were also applied to the same functions in order to compare the performance of continuous EDAs.

P. Larrañaga et al. (eds.), *Estimation of Distribution Algorithms*

The outline of this chapter is as follows: Section 2 describes the optimization problems that will be used and Section 3 explains which algorithms will be applied. Section 4 is a brief description of the experiments, and Section 5 shows the results obtained. Finally, Section 6 is the conclusion.

2. Description of the optimization problems

In order to test the performance of continuous EDAs and ESs, some standard functions broadly used on the literature for optimization techniques comparison have been chosen. The functions are as follows:

Summation cancellation: This is a maximization problem introduced in Baluja and Davies (1997). For any individual $\boldsymbol{x}= (x_1, \ldots, x_n)$, the range of the components is $-0.16 \leq x_i \leq 0.16$, $i = 1, \ldots, n$. The fitness function is computed as follows:

$$F(\boldsymbol{x}) = \frac{1}{(10^{-5} + \sum_{i=1}^{n} |y_i|)} \tag{8.1}$$

where $y_1 = x_1$, $y_i = x_i + y_{i-1}$, $i = 2, \ldots, n$. The fittest individual is the one whose variables have all components equal to 0, and this corresponds to a fitness value of 100000.

Griewangk: This is a minimization problem proposed in Törn and Žilinskas (1989). The fitness function is as follows:

$$F(\boldsymbol{x}) = 1 + \sum_{i=1}^{n} \frac{x_i^2}{4000} - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right). \tag{8.2}$$

The range of all the components of the individual is $-600 \leq x_i \leq 600$, $i = 1, \ldots, n$, and the fittest individual corresponds to a value of 0, that only can be obtained when all the components of the individual are 0.

Sphere model: This other problem is a broadly known simple minimization one. It is also defined so that $-600 \leq x_i \leq 600$, $i = 1, \ldots, n$, and the fitness value for each individual is as follows:

$$F(\boldsymbol{x}) = \sum_{i=1}^{n} x_i^2. \tag{8.3}$$

The reader can easily appreciate that the fittest individual is the one whose all components are 0, which corresponds to the fitness value 0.

Rosenbrock generalized: This problem proposed in Rosenbrock (1960) is a minimization one. The originally proposed problem was thought for only 2 dimensions, and this problem has been generalized to n dimensions (Salomon, 1998). It is defined as:

$$F(\boldsymbol{x}) = \sum_{i=1}^{n-1} \left[100 \cdot (x_{i+1} - x_i^2)^2 + (1 - x_i)^2\right]. \tag{8.4}$$

The optimum value is 0, which is obtained again for the individual whose components are all set to 0. The range of values is of $-10 \leq x_i \leq 10$, $i = 1, \ldots, n$.

Ackley: This optimization problem was proposed in Ackley (1987), and it is again a minimization problem which best value is 0. This fitness value is obtained by the individual whose components are all set to 0. Originally this problem was defined for two dimensions, but the problem has been generalized to n dimensions (Bäck, 1996). The fitness function is computed as follows:

$$F(\boldsymbol{x}) = -20 \cdot \exp\left(-0.2\sqrt{\frac{1}{n} \cdot \sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n} \cdot \sum_{i=1}^{n} \cos(2\pi x_i)\right) + 20 + \exp(1). \quad (8.5)$$

Figure 8.1 shows a graphic of each of the problems explained so far for the particular case of $n = 2$.

3. Algorithms to test

For each of the optimization problems in continuous domains introduced in the previous section, the behavior of 8 evolutionary computation algorithms has been compared to each other. From these 8 algorithms, 7 correspond to examples of continuous EDAs, while the other is an ES.

In particular, the 8 algorithms tested have been the following:

- Evolutionary strategy, $(\mu + \lambda)$-strategy with recombination.
- UMDA_c, in which the factorization of the joint density function is computed as the product of univariate normal densities.
- MIMIC_c, where in this case the joint density is factorized by means of a chain-like model using statistics of first and second order.
- EGNA_{BIC}, in which the factorization is given by a Gaussian network. The search of a model in every generation is based on the penalized maximum-likelihood criterion, and the heuristic that looks for the best structure is reduced to a local search that is initialized as the model obtained in the previous generation.
- EGNA_{BGe}, its characteristics are similar to the previous algorithm except for the evaluation criterion, which in this case we make use of a Bayesian metric.
- EGNA_{ee}, induces Gaussian network models starting from the results of hypothesis tests applied to each of the arcs in the structure of a Gaussian network.

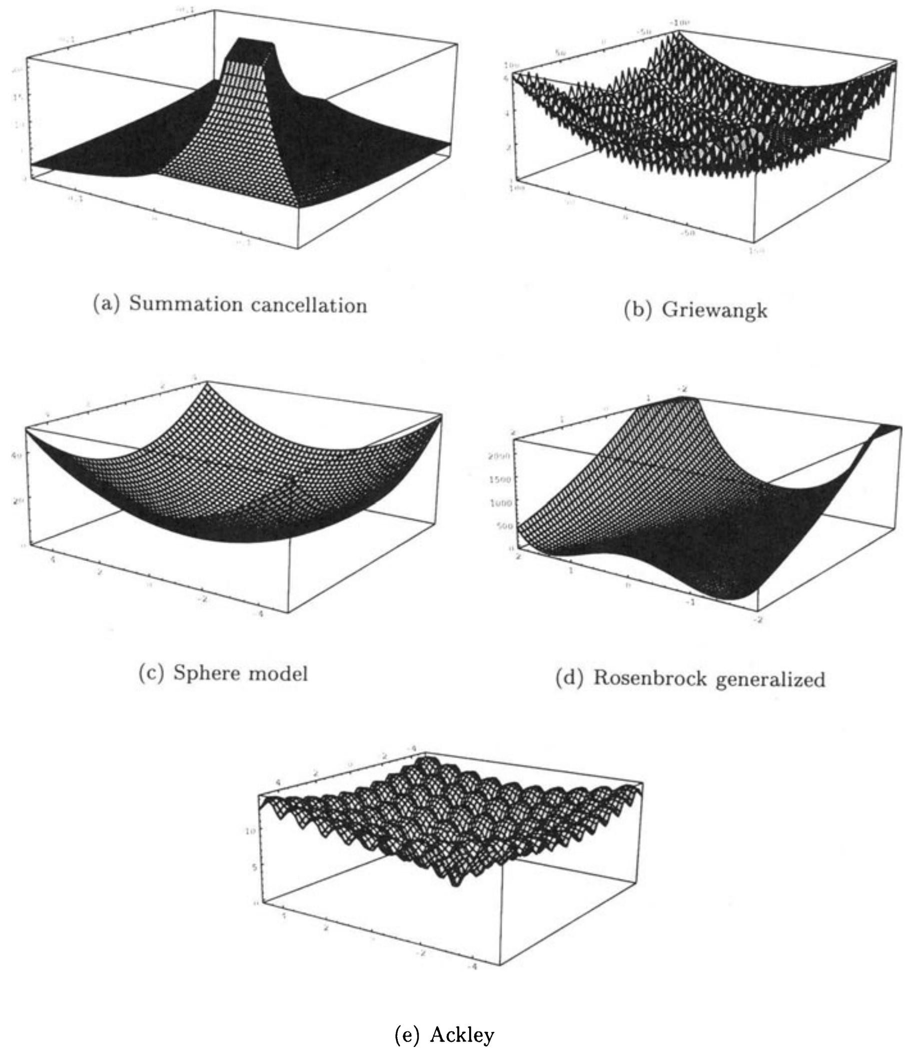

Figure 8.1 Plots of the problems to be optimized with continuous EDAs and ES techniques ($n = 2$).

- EMNA_{global}, in this algorithm the model induced corresponds to a multivariate normal distribution.

- EMNA_a, in which, similarly as in the previous algorithm, the model is a multivariate normal distribution. In this case, the distribution is adapted every step following the same philosophy as a steady state genetic algorithm (Whitley and Kauth, 1988).

This last algorithm has only been applied to a dimension of 10, but the rest of the algorithms have been tested to dimensions of both 10 and 50.

For more details about ESs the reader is referred to Schwefel (1995). Algorithms UMDA_c, MIMIC_c, EGNA_{BGe} and EGNA_{ee} are described in Larrañaga et al. (2000), and algorithms EGNA_{BIC}, EMNA_{global}, and EMNA_a are further described in Larrañaga et al. (2001).

4. Brief description of the experiments

This section describes the experiments and the results obtained. Continuous EDAs were implemented in ANSI C++ language, and the ES techniques were obtained from Schwefel (1995). Some of the selected problems were not implemented in the ESs source code, and therefore some changes were made in order to include them in the ANSI C program.

The initial population for both continuous EDAs and ESs was generated using the same random generation procedure based on a uniform distribution.

In EDAs, the following parameters were used: a population of 2000 individuals ($M = 2000$), from which a subset of the best 1000 are selected ($N = 1000$) to estimate the density function, and the elitist approach was chosen (that is, always the best individual is included for the next population and 1999 individuals are simulated).

The ES chosen was taken from Schwefel (1995). This is a standard $(\mu + \lambda)$-strategy with discrete recombination in points of the search space and intermediary recombination for strategy parameters. The value of the parameters was set to $\mu = 15$, $\lambda = 100$, lower bound to step sizes (absolute of 1E-30 and relative of 9.8E-07), parameter in convergence test (absolute of 1E-30 and relative of 9.8E-07).

For all the continuous EDAs the ending criterion was set to reach 301850 evaluations (i.e. number of individuals generated) or when the result obtained was closer than 1E-06 from the optimum solution to be found.

These experiments were all executed in a two processor Ultra 80 Sun computer under Solaris version 7 with 1 Gb of RAM.

4.1 Experimental results

Results such as the best individual obtained and the number of evaluations to reach the final solution were recorded for each of the experiments.

Each algorithm was executed 10 times, and the null hypothesis of the same distribution densities was tested. The results are shown in Tables 8.1 to 8.5. In order to check whether the difference behaviour between the algorithms is statistically significant the non-parametric tests of Kruskal-Wallis and Mann-Whitney were used. This task was carried out with the statistical package S.P.S.S. release 9.00.

Table 8.1 Mean values of experimental results after 10 executions for the problem Summation cancellation with a dimension of 10 and 50 (optimum fitness value = 1.0E+5).

Dimension	*Algorithm*	*Best fitness value*	*Number of evaluations*
10	$UMDA_c$	6.23390E+04 ± 1.87E+04	301850 ± 0.0
	$MIMIC_c$	5.79875E+04 ± 2.30E+04	301850 ± 0.0
	$EGNA_{BIC}$	5.50837E+04 ± 1.70E+04	301850 ± 0.0
	$EGNA_{BGe}$	9.99991E+04 ± 1.25E-01	190305 ± 1836.9
	$EGNA_{ee}$	9.99992E+04 ± 1.73E-01	195903 ± 1632.2
	$EMNA_{global}$	9.99991E+04 ± 1.03E-01	192904 ± 1056.6
	$EMNA_a$	6.40978E+00 ± 6.78E-01	301850 ± 0.0
	ES	3.57871E-03 ± 6.31E-03	43000 ± 8232.7
50	$UMDA_c$	6.89860E-01 ± 2.92E-02	301850 ± 0.0
	$MIMIC_c$	6.91292E-01 ± 4.73E-02	301850 ± 0.0
	$EGNA_{BIC}$	7.23125E-01 ± 3.28E-02	301850 ± 0.0
	$EGNA_{BGe}$	9.17252E+03 ± 2.89E+04	278861.5 ± 17761.2
	$EGNA_{ee}$	8.62138E+04 ± 8.91E+03	301850 ± 0.0
	$EMNA_{global}$	8.61907E+04 ± 1.31E+04	301850 ± 0.0
	ES	5.31544E-08 ± 1.05E-08	193000 ± 27507.6

Table 8.2 Mean values of experimental results after 10 executions for the problem Griewangk with a dimension of 10 and 50 (optimum fitness value = 0).

Dimension	*Algorithm*	*Best fitness value*	*Number of evaluations*
10	$UMDA_c$	6.0783E-02 ± 1.93E-02	301850 ± 0.0
	$MIMIC_c$	7.3994E-02 ± 2.86E-02	301850 ± 0.0
	$EGNA_{BIC}$	3.9271E-02 ± 2.43E-02	301850 ± 0.0
	$EGNA_{BGe}$	7.6389E-02 ± 2.93E-02	301850 ± 0.0
	$EGNA_{ee}$	5.6840E-02 ± 3.82E-02	301850 ± 0.0
	$EMNA_{global}$	5.1166E-02 ± 1.67E-02	301850 ± 0.0
	$EMNA_a$	12.9407 ± 3.43	301850 ± 0.0
	ES	3.496E-02 ± 1.81E-02	25000 ± 1699.7
50	$UMDA_c$	8.9869E-06 ± 9.36E-07	177912 ± 942.3
	$MIMIC_c$	9.0557E-06 ± 8.82E-07	177912 ± 942.3
	$EGNA_{BIC}$	1.7075E-04 ± 6.78E-05	250475 ± 18658.5
	$EGNA_{BGe}$	8.6503E-06 ± 7.71E-07	173514.2 ± 1264.3
	$EGNA_{ee}$	9.1834E-06 ± 5.91E-07	175313.3 ± 965.6
	$EMNA_{global}$	8.7673E-06 ± 1.03E-06	216292 ± 842.8
	ES	1.479E-03 ± 3.12E-03	109000 ± 13703.2

Table 8.3 Mean values of experimental results after 10 executions for the problem Sphere model with a dimension of 10 and 50 (optimum fitness value = 0).

Dimension	*Algorithm*	*Best fitness value*	*Number of evaluations*
10	$UMDA_c$	6.7360E-06 ± 1.26E-06	74163.9 ± 1750.3
	$MIMIC_c$	7.2681E-06 ± 2.05E-06	74963.5 ± 1053.5
	$EGNA_{BIC}$	2.5913E-05 ± 3.71E-05	77162.4 ± 6335.4
	$EGNA_{BGe}$	7.1938E-06 ± 1.78E-06	74763.6 ± 1032.2
	$EGNA_{ee}$	7.3713E-06 ± 1.98E-06	73964 ± 1632.1
	$EMNA_{global}$	7.3350E-06 ± 2.24E-06	94353.8 ± 842.8
	$EMNA_a$	4.8107E+04 ± 1.32E+04	301000 ± 0.0
	ES	0 ± 0.0	48200 ± 1135.2924
50	$UMDA_c$	8.9113E-06 ± 8.41E-07	211495.2 ± 1264.2
	$MIMIC_c$	8.9236E-06 ± 9.66E-07	211695.1 ± 1474.9
	$EGNA_{BIC}$	1.2126E-03 ± 7.69E-04	263869 ± 29977.5
	$EGNA_{BGe}$	8.7097E-06 ± 1.30E-06	204298.8 ± 1264.2
	$EGNA_{ee}$	8.3450E-06 ± 1.04E-06	209496.2 ± 1576.8
	$EMNA_{global}$	8.5225E-06 ± 1.35E-06	247477.2 ± 1264.2
	ES	1.541E-45 ± 4.43E-46	173000 ± 4830.4

Table 8.4 Mean values of experimental results after 10 executions for the problem Rosenbrock generalized with a dimension of 10 and 50 (optimum fitness value = 0).

Dimension	*Algorithm*	*Best fitness value*	*Number of evaluations*
10	$UMDA_c$	8.7204 ± 3.82E-02	301850 ± 0.0
	$MIMIC_c$	8.7141 ± 1.64E-02	301850 ± 0.0
	$EGNA_{BIC}$	8.8217 ± 0.16	268066.9 ± 69557.3
	$EGNA_{BGe}$	8.6807 ± 5.87E-02	164518.7 ± 24374.5
	$EGNA_{ee}$	8.7366 ± 2.23E-02	301850 ± 0.0
	$EMNA_{global}$	8.7201 ± 4.33E-02	289056.4 ± 40456.9
	$EMNA_a$	3263.0010 ± 1216.75	301000 ± 0.0
	ES	–	–
50	$UMDA_c$	48.8949 ± 4.04E-03	301850 ± 0.0
	$MIMIC_c$	48.8894 ± 1.11E-02	301850 ± 0.0
	$EGNA_{BIC}$	50.4995 ± 2.30	301850 ± 0.0
	$EGNA_{BGe}$	48.8234 ± 0.118	275663.1 ± 1750.3
	$EGNA_{ee}$	48.8893 ± 1.11E-02	301850 ± 0.0
	$EMNA_{global}$	49.7588 ± 0.52	296252.8 ± 7287.1
	ES	–	–

Table 8.5 Mean values of experimental results after 10 executions for the problem Ackley with a dimension of 10 and 50 (optimum fitness value = 0).

Dimension	*Algorithm*	*Best fitness value*	*Number of evaluations*
	$UMDA_c$	7.8784E-06 ± 1.17E-06	114943.5 ± 1413.5
	$MIMIC_c$	8.8351E-06 ± 9.01E-07	114743.6 ± 1032.3
	$EGNA_{BIC}$	5.2294 ± 4.49	229086.4 ± 81778.4
10	$EGNA_{BGe}$	7.9046E-06 ± 1.39E-06	113944 ± 1632.2
	$EGNA_{ee}$	7.4998E-06 ± 1.72E-06	118541.7 ± 2317.8
	$EMNA_{global}$	8.9265E-06 ± 6.89E-07	119141.4 ± 1032.3
	$EMNA_a$	10.8849 ± 1.19	301000 ± 0.0
	ES	20 ± 0.0	18000 ± 7180.2
	$UMDA_c$	9.0848E-06 ± 3.11E-07	296852.5 ± 1053.5
	$MIMIC_c$	9.6313E-06 ± 3.83E-07	295653.1 ± 632.1
	$EGNA_{BIC}$	1.9702E-02 ± 7.50E-03	288256.8 ± 29209.4
50	$EGNA_{BGe}$	8.6503E-06 ± 3.79E-07	282059.9 ± 632.1
	$EGNA_{ee}$	6.8198 ± 0.27	301850 ± 0.0
	$EMNA_{global}$	9.5926E-06 ± 2.39E-07	291255.3 ± 1349.2
	ES	20 ± 0.0	88000 ± 19888.6

4.2 Comments on the results

The experimental results shown in Tables 8.1 to 8.5 contain important differences between the algorithms depending on the optimization problem. Next, each of the problems will be analyzed separately, showing for each case which appeared to be the most suited algorithms and testing whether the different performance is statistically significant.

Summation cancellation: For the Summation cancellation example the algorithms that arrived to the optimum solution for the 10 dimension case were $EMNA_{global}$, $EGNA_{BGe}$ and $EGNA_{ee}$. When applying the non parametric tests to these algorithms we obtain $p = 0.249$ for the best fitness value and $p < 0.001$ for the number of evaluation required. This means that for this example there are statistically significant differences in the number of evaluations required to reach the best solution, but the best result obtained is not statistically significant for these three algorithms.

For the 50 dimension case, the algorithms that performed best (that arrived to the best solution) were $EMNA_{global}$ and $EGNA_{ee}$, as $EGNA_{BGe}$ shows a worse results when increasing the complexity of the problem. In both algorithms the final results obtained were close to the optimum, but in all the cases they arrived to the maximum of evaluations and their execution was stopped (they satisfied the ending criterion before reaching the best solution). If we apply the non parametric test to these two algorithms we obtain $p = 0.191$ for the best fitness value and $p < 0.001$ fór the number of evaluations required, which means that the fitness values obtained with these two best algorithms are not statistically significant, but difference in number of evaluations required for convergence is statistically significant.

Griewangk: The Griewangk problem is a very complex problem to optimize due to the many local minima it presents, as it can be seen in Figure 8.1b. In the 10 dimension case all the mean fitness values of the Table 8.2 appear quite similar for all but the EMNA_a algorithm. When performing the non parametric test for all the results we obtain though that differences are significant for both the fitness value and number of evaluations. The algorithm that shows the best behaviour is ESs, closely followed by EGNA_{BIC}, EMNA_{global}, EGNA_{ee}, MIMIC_c and UMDA_c algorithms. When comparing ESs and these five continuous EDAs we do not obtain statistically significant differences in the best fitness value ($p = 0.101$). On the other hand, the differences are significant in the number of evaluations required for these algorithms ($p < 0.001$).

In the 50 dimension case the differences are bigger, and therefore the performance of the different algorithms can be seen more clearly. In this case ESs arrived to the ending criterion and stopped the search without reaching a solution as well as the obtained with any of the EDAs. Differences among all the continuous EDAs are statistically significant looking at the results ($p < 0.001$), and the algorithms that quicker arrived to these results were EGNA_{BGe}, EGNA_{ee}, MIMIC_c and UMDA_c, although the difference in fitness value of these was non significant ($p = 0.505$). The fact that EGNA_{BIC} did not converge as quick as the rest shows that this algorithm is more dependent on the complexity of the problem than the others.

Sphere model: This problem does not have any local minima, and its optimum fitness value is 0. This is again a very suitable optimization problem for the ES method, that obtained the optimum result in practically all the executions and for both the 10 and 50 dimensions.

If we do not consider the EMNA_a, differences for the 10 dimension case were not significant in the best results obtained ($p = 0.197$). The main differences are in the number of evaluations, which is statistically significant when performing the test for all the algorithms ($p < 0.001$), but when performing the same test excluding the EMNA_{global} we obtain $p = 0.125$ for the fitness values and $p = 0.671$ for the number of evaluations, which shows clearly that all the non-EMNA algorithms do not require a significantly different number of evaluations to reach similar solutions.

In the 50 dimension these differences appear to be more important, and for this case again the $EGNA_{BIC}$ algorithm shows a worse performance when increasing the dimension of the problem from 10 to 50. As a result, if excluding the latter algorithm from the non parametric test we obtain that differences are not statistically significant between all the rest of continuous EDAs ($p = 0.719$). The number of evaluations required for these algorithms (all continuous EDAs except for $EGNA_{BIC}$) is still significantly different ($p < 0.001$).

Rosenbrock generalized: Rosenbrock is a problem illustrated in Figure 8.1d which does not contain many local minima nor maxima. For this reason, the ES method is very suitable and shows the best results for both the 10 and 50 dimension cases.

Looking at the performance of the continuous EDAs, excluding $EMNA_a$, there are significant differences between all of them for the 10 dimension case ($p = 0.002$). If we perform the test for the algorithms $MIMIC_c$, $EMNA_{global}$, $UMDA_c$, $EGNA_{BGe}$ and $EGNA_{ee}$ we obtain that the difference in the best fitness value obtained are not significant ($p = 0.074$). From all of them, the $EGNA_{BGe}$ appears to require significantly less number of evaluations to converge.

In the 50 dimension case, the differences in the best fitness value obtained are significant between all the continuous EDAs, and the best fitness values are obtained with $EGNA_{BGe}$, $EGNA_{ee}$, $MIMIC_c$ and $UMDA_c$. The results of these algorithms were not statistically significant ($p = 0.375$), but when computing the non parametric test also with $EMNA_{global}$ differences appear to be important ($p < 0.001$). The faster convergence was achieved with the $EGNA_{BGe}$ algorithm.

Ackley: The Ackley problem has also several local minima as it can appreciated in Figure 8.1e. The ES method performed quite worse in both 10 and 50 dimensions than the rest of the algorithms, that showed much closer fitness results to the optimum value of 0.

Following the results shown in Table 8.5 for the 10 dimension case, the best results were obtained with $EGNA_{ee}$, $UMDA_c$, $EGNA_{BGe}$, $MIMIC_c$ and $EMNA_{global}$. However, the non parametric test of Kruskal-Wallis showed that differences were not statistically significant for these algorithms in the fitness value ($p = 0.085$). In the 50 dimension case, from the 5 algorithms, $EGNA_{ee}$ performed quite worse than the rest. The only not statistically significant results in the fitness value is for the algorithms $UMDA_c$ and $EGNA_{BGe}$ ($p = 0.151$), although there are significant differences for both in the number of evaluations required ($p < 0.001$). This means that EGNA_{BGe} behaves better for this problem at a dimension of 50. If we perform the same hypothesis test with UMDA_c, EGNA_{BGe} and EMNA_{global}, the differences in fitness value become statistically significant ($p = 0.005$).

4.3 The evolution of the search

As an example to illustrate the difference in the way of reaching the final result for all the continuous EDAs, Figure 8.2 shows which is the behaviour of all the algorithms for the Summation cancellation problem with dimension of 10. In this figure appears clear that the algorithms that arrive quicker to the optimum solution are EMNA_{global}, EGNA_{BGe} and EGNA_{ee}, which arrive to convergence. The rest of the algorithms do not show such a good behaviour, and when the maximum number of evaluations is reached their execution was stopped. Another important aspect in the figure is that it shows clearly that the EMNA_{global} converges a bit faster than EGNA_{BGe} and EGNA_{ee}: if the execution had been stopped at about the 50^{th} generation this algorithm would have returned the best result. EGNA_{BGe} converges also very close, but results in Table 8.3 show that when the complexity of the problem increases from 10 to 50 its relative performance worsens. This fact has also been seen for most of the optimization problems in the experiments.

4.4 The computation time

The computation time is the CPU time of the process for each execution, and therefore it is not dependent on the multiprogramming level at the execution time. As an example of the difference in computation time for all the algorithms again the example of the Summation cancellation problem was used for both 10 and 50 dimensions. The results are shown in Table 8.6. This computation time is presented as a measure to illustrate the different computation complexity of all the algorithms. It is important also to note that all the operations for the estimation of the distribution, the simulation, and the evaluation of the new individuals are carried out through memory operations.

As expected, the CPU time of each algorithm is according to the complexity of the algorithm for the learning step in the EDA algorithm. Following this fact, the shortest algorithm in computation time are in order UMDA_c and MIMIC_c. All the EGNA type algorithms show a longer computation time due to the calculation of the structure that represents the learning, which has no restriction in the number of parents for each variable.

It is also worth mentioning the computation time of EMNA_{global}, which is a bit shorter that the EGNA type ones. As EMNA_{global} is based on the assumption of the complete dependence of all variables each other (the structure

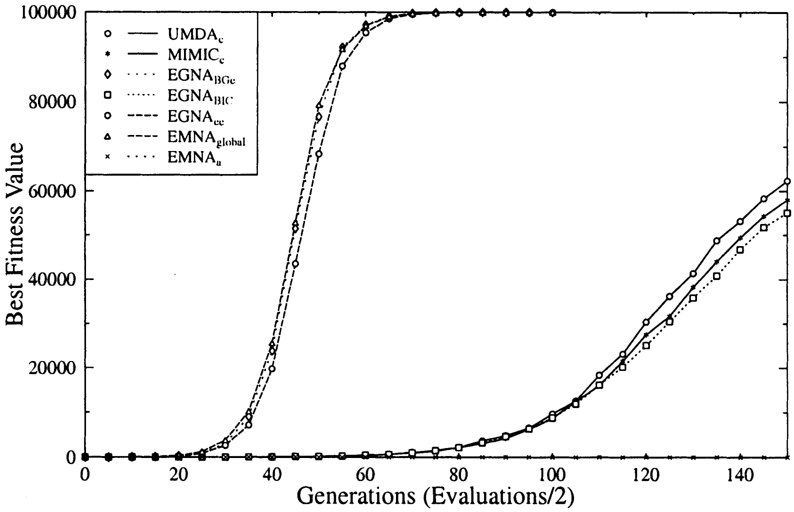

Figure 8.2 Evolution of the different continuous EDAs for the Summation cancellation problem with a dimension of 10.

Table 8.6 Mean values of the computation time after 10 executions for the problem Summation cancellation with a dimension of 10 and 50.

	Dimension 10	*Dimension 50*
UMDA$_c$	0:02:36 ± 0:00:00	0:03:23 ± 0:00:01
MIMIC$_c$	0:02:47 ± 0:00:01	0:04:12 ± 0:00:00
EGNA$_{BIC}$	0:07:15 ± 0:00:01	3:15:31 ± 0:00:04
EGNA$_{BGe}$	0:03:03 ± 0:00:02	4:03:13 ± 0:13:18
EGNA$_{ee}$	0:01:59 ± 0:00:01	3:19:37 ± 0:03:42
EMNA$_{global}$	0:01:55 ± 0:00:00	3:16:07 ± 0:00:10
EMNA$_a$	0:05:49 ± 0:00:04	–
ES	0:00:02 ± 0:00:00	0:00:29 ± 0:00:06

is a complete graph), no time is required in order to estimate the most suitable structure for the learning step. It is also important to note that the other EMNA type algorithm (EMNA$_a$) shows a much longer computation time than the rest of the algorithms, which made it not suitable for its use with the 50 dimension example. This fact happened as well in the rest of the optimization problems.

On the other hand, it is important to note that ESs show a very short computation time for this Summation cancellation problem. However, although its performance is quite good for some of these problems (e.g. the Sphere model), for the case of the Summation cancellation the ending criterion is reached too quick to obtain a good solution and its execution is aborted in advance.

5. Conclusions

At the light of the results obtained in the fitness values, we can conclude the following: generally speaking, for small dimension EMNA_{global}, and EGNA type algorithms perform better that the rest, but when increasing the dimension some of the algorithms show a poorer performance as a result of the higher complexity to take into account (e.g. the case of EGNA_{BIC}). The EMNA_a algorithm showed a very poor behaviour for all these optimization problems, and its additional computation effort made impossible to apply it to the 50 dimension cases.

An important aspect to take into account is that the EMNA_{global} algorithm appears to be the method that more quickly approaches to the best results, although these results are not always the optima. Nevertheless, once this algorithm is nearby the optimum solution it requires more time than algorithms as EGNA_{BGe} or EGNA_{ee} to satisfy the ending criterion. This is the reason why in Tables 8.1 to 8.5 this fact is not clear.

Depending on the problem the ES method showed better results than the continuous EDAs, but when the type of problem to optimize presents many local minima or maxima, continuous EDAs show a more appropriated behaviour. The main drawback for continuous EDAs in general is the computation time they require, but for some problems the results that can be obtained with them are not comparable to methods in the ESs category.

Acknowledgments

This article has been partially supported by the Spanish Ministry for Science and Education with the project HF1999-0107.

References

Ackley, D. H. (1987). *A connectionist machine for genetic hillclimbing.* Kluwer, Boston.

Bäck, T. (1996). *Evolutionary Algorithms in Theory and Practice.* Oxford University Press.

Baluja, S. and Davies, S. (1997). Using optimal dependency-trees for combinatorial optimization: Learning the structure of the search space. Technical report, Carnegie Mellon Report, CMU-CS-97-107.

Larrañaga, P., Etxeberria, R., Lozano, J., and Peña, J. (2000). Optimization in continuous domains by learning and simulation of Gaussian networks. In *Proceedings of the Workshop in Optimization by Building and using Probabilistic Models. A Workshop within the 2000 Genetic and Evolutionary Computation Conference, GECCO 2000*, pages 201–204, Las Vegas, Nevada, USA.

Larrañaga, P., Lozano, J. A., and Bengoetxea, E. (2001). Estimation of Distribution Algorithms based on multivariate normal and Gaussian networks. Technical Report KZZA-IK-1-01, Department of Computer Science and Artificial Intelligence, University of the Basque Country.

Rosenbrock, H. H. (1960). An automatic method for finding the greatest or least value of a function. *The Computer Journal*, 3:175–184.

Salomon, R. (1998). Evolutionary algorithms and gradient search: similarities and differences. *IEEE Transactions on Evolutionary Computation*, 2(2):45–55.

Schwefel, H.-P. (1995). *Evolution and Optimum Seeking*. Wiley InterScience.

Törn, A. and Žilinskas, A. (1989). *Global optimization*. Lecture Notes in Computer Science 350. Springer–Verlag, Berlin Heidelberg.

Whitley, D. and Kauth, J. (1988). GENITOR: A different genetic algorithm. In *Proceedings of the Rocky Mountain Conference on Artificial Intelligence*, volume II, pages 118–130.