

# Paired Comparison-based Interactive Differential Evolution

Takagi, Hideyuki  
Faculty of Design, Kyushu University

Pallez, Denis  
Nice Sophia-Antipolis University

<http://hdl.handle.net/2324/1808446>

---

出版情報 : World Congress on Nature and Biologically Inspired Computing (NaBIC). 2009, pp.475-480, 2009-12-09

バージョン :

権利関係 :



# Paired Comparison-based Interactive Differential Evolution

Hideyuki Takagi

Kyushu University  
Faculty of Design  
Fukuoka, Japan

<http://www.design.kyushu-u.ac.jp/~takagi>

Denis Pallez

Nice Sophia-Antipolis University  
I3S Laboratory – CNRS UMR 6070  
Nice, France

<http://www.i3s.unice.fr/~dpallez/>

*Abstract*—We propose a system of **Interactive Differential Evolution (IDE)** based on paired comparisons for reducing user fatigue and evaluate its convergence speed in comparison with **Interactive Genetic Algorithms (IGA)** and tournament IGA. User interface and convergence performance are central to reducing **Interactive Evolutionary Computation (IEC)** user fatigue. Unlike IGA and conventional IDE, users of the proposed IDE and tournament IGA do not need to compare whole individuals with each other but rather only to compare pairs of individuals, which largely decreases user fatigue. In this paper, we design a pseudo-IEC user and evaluate another factor, IEC convergence performance, using IEC simulators and show that our proposed IDE converges significantly faster than IGA and tournament IGA, i.e. our proposed method is superior to others from both user interface and convergence performance points of view.

*Evolutionary Algorithms; Differential Evolution; Interactive Evolutionary Computation, Paired Comparison, Gaussian Mixture Model*

## I. INTRODUCTION

There are many optimization tasks for which it is difficult or almost impossible to design scales for evaluating target systems quantitatively, but which are accommodating to subjective evaluations. These tasks include, for example, drawing montages based on a witness's memory, fitting a hearing-aid to achieve a satisfactory sounds, designing cute or lovely motions for home robots. Interactive Evolutionary Computation (IEC) has been applied to these tasks in a wide variety of application areas [1].

The biggest drawback with IEC is IEC user fatigue caused by human cooperation with a tireless computer. IEC users become exhausted and bored with evaluating many generated individuals. The population size and number of evolving generations are limited by this fatigue; 10 - 20 individuals and 10 - 20 generations are frequently used in IEC. Because significantly fewer individuals are used than in normal EC searches, convergence is slower. The slower convergence is another factor in IEC user fatigue.

Several trials have been performed to address the fatigue problem [1]. Improving IEC user interface is one avenue for making improvements. Some changes that can be made to

the interface include: improving the display interface to help users compare individuals easily; improving the input interface by reducing the number of evaluation levels, for example 5 evaluation levels instead of 100; and improving the data visualization by projecting the distribution of individuals in the  $n$ -D search space onto a 2-D or 3-D space.

Predicting the IEC user's evaluations using trained prediction models is other approach. The model is used as a fitness function in a normal EC search and is combined with IEC to accelerate IEC search. To train these models using the IEC user's evaluations from past generations, a distance-based model [1], genetic programming [2], neural networks [1], and Support Vector Machine [3, 4] have been used. References [5, 6] propose using an eye-tracker to avoid the input of fitness values or even to avoid requiring the user to select best individuals: this can instead be achieved by correctly interpreting cognitive store data; for instance, by considering time spent evaluating individuals.

Another possible solution is to use tournament IEC [7], which is based on paired comparisons rather than comparing all individuals. With this technique, it is easier for the IEC user because he or she need only evaluate a pair of individuals rather than all individuals at once. However, a drawback is there is less information from which to calculate each individual's fitness value because not all individuals are compared, which means that the fitness includes more noise and convergence may be slower. In simulated breeding IEC, the IEC user just chooses the better individuals from the total population, a one-click selection using a pointing device is an easy IEC user interface. Although it compares all individuals unlike in tournament IEC, the 1 bit evaluation includes more quantization noise in fitness than any other evaluation method and may slow IEC converge [8].

Another approach is to introduce a new type of EC to accelerate IEC convergence. Particle Swarm Optimization (PSO) can be used as an EC part in IEC. As PSO is sensitive to quantization noise in IEC fitness, better performance of Interactive PSO than Interactive Genetic Algorithms (IGA: Interactive GA) is achieved by combining some methods reducing the effect of the quantization noise in fitness with Interactive PSO [8].

Differential Evolution (DE) [9] is an optimization technique which has been used frequently this decade. DE has two possible advantages that are relevant to IEC: first is the use of comparisons between only two individuals (*paired comparisons*) and second one is its potential for faster convergence. The objective of this paper is to evaluate Interactive DE (IDE) [10-12] in comparison with conventional IEC approaches to demonstrate its potential. Unfortunately, the IDE of references [10-12] did not make use of the first feature, paired comparisons. Through IEC simulation, we show that our proposed paired comparison-based IDE method has better convergence performance than conventional IEC methods. Since the advantage of paired comparison over the many comparisons of conventional IGA are obvious, the proposed IDE is considered best if it is faster than tournament IGA or faster than or equal to IGA.

We explain the EC algorithms used in our study including the GA and DE in section II and how individuals are evaluated in the interactive frameworks in section III. Section IV evaluates how the IDE converges in comparison with the three conventional IEC approaches.

## II. EC ALGORITHMS

We compare four EC algorithms (DE, Genetic Algorithm (GA), tournament1-GA, and tournament2-GA) with and without an IEC framework. Let us first present what tournament-GA and DE are.

### A. Tournament Genetic Algorithms (TGA)

Reference [13] first proposed *competitive fitness*, a fitness that uses relative evaluations for the fitness function instead of absolute values. Reference [7] was the first to apply tournament fitness to IEC. It also proposed a tournament IEC that uses not only which is the better of two paired individuals but also how much better its fitness values are.

*Tournament1-GA* is a GA in which individuals are evaluated according to a method of competitive fitness called *single-elimination tournament* in [13, 14]. Individuals are paired at random, and one game played per pair. Losers of games are eliminated from the tournament... This process continues until only the champion remains. The fitness of an individual is the number of games played. In the interactive case, e.g. tournament1-IGA, the IEC user just chooses one of the two displayed individuals.

*Tournament2-GA* is a GA in which individuals are evaluated according to another method of competitive fitness based on the single-elimination tournament first proposed in [7]. The fitness is computed based on not only the number of games played but also the distance between paired individuals. For instance, we start by giving a fitness of 10 to the champion of the tournament. An individual which has lost against the champion is given the champion's fitness, i.e. 10, minus the difference between it and the champion, and so on for all individuals. In the interactive case, the difference between the individuals is supposed to be given by the user and previous fitness values are also made discrete in  $n$ -evaluation levels. An individual that has fought against the tournament's champion in the first game, i.e. it lost the

tournament in the first round, will have a better fitness in tournament2 than in tournament1.

### B. Differential Evolution (DE)

The salient relevance of DE is that *comparing two vectors* is the only evaluation used in DE. We believed that this *paired comparison* is the big advantage DE can offer IEC. However, work done in [10-12], which seems to be first work with Interactive DE, did not exploit this big potential to reduce IEC user fatigue; the IEC user was asked to choose better individuals among all eight shown individuals.

DE is a population based, stochastic and continuous function optimizer [9] where distance and direction information from the current population is used to guide the search process [15]. DE is known to be able to handle non-differentiable, nonlinear and multimodal cost functions, to be parallelized to cope with computation intensive cost functions, easy to use, and well suited for rapid convergence, i.e. consistent convergence to the global is achieved in consecutive independent trials.

Essentially, for each individual of the population (*parent* or *target vector*), first a *mutant vector* is generated by adding a weighted difference (*difference vector*) between two randomly chosen vectors (*parameter vectors*) to the third chosen vector (*base vector*). Secondly, the *trial vector* is obtained from the mutant vector and the target vector using binomial or exponential crossover. Finally, the target vector is replaced with the better vector of either the trial or target. There are some variations in how to determine the base vector. See references [9, 15] for detailed explanation of the DE algorithm.

## III. EVALUATION TASK

### A. Pseudo-IEC User

Humans cannot conduct thousands of evaluations under completely identical conditions and it would not be reliable to evaluate the convergence of IDE by comparing it with those of conventional IEC methods. We should therefore evaluate them using an IEC simulation by designing a pseudo-IEC user, although we evaluate our proposed paired comparison-based IDE with human IEC user later [8].

There are three IEC features that we must realize in the pseudo-IEC user:

- (1) evaluation characteristics have lower complexity,
- (2) relative fitness is produced each generation, and
- (3) fitness is expressed discretely in  $n$ -evaluation levels.

Furthermore, the evaluation characteristics of the pseudo-IEC user should be controlled parametrically.

We realized (1) using the Gaussian Mixture Model described in section B. (2) is realized thanks to the *competitive* fitness function [7, 13, 14] explained in section II.A, and (3) is realized by dividing the range of the best fitness and the worst fitness obtained with the Gaussian Mixture Model into  $n$ -evaluation levels in each generation.

We use the Gaussian Mixture Models consisting of four Gaussian Mixture functions to emulate the evaluation landscape of the human mind. An IEC task may not be unimodal, but nor is it greatly complex because IEC users

can reach satisfactory solutions with lower population sizes and fewer generations (see Figure 1).

### B. Gaussian Mixture Model

Our evaluation tasks are 4 different multidimensional Gaussian Mixture Models in 3-D, 5-D, 7-D, and 10-D. We design them to make all their function characteristics the same so we can control our experimental complexities by changing only the dimensionality. They are expressed as:

$$f(x_1, \dots, x_n) = \sum_{i=1}^k a_i \exp\left(-\sum_{j=1}^n \frac{(x_{ij} - \mu_{ij})^2}{2\sigma_{ij}^2}\right) \quad (1)$$

where  $k$  and  $n$  are respectively the number of Gaussian functions and the dimensionality;  $k=4$  and  $n = 3, 5, 7,$  and  $10$  in this paper.  $a_i$ ,  $\sigma_{ij}$  and  $\mu_{ij}$  represent the height, the standard deviation and the central position of the  $i$ -th Gaussian function, respectively. Parameter values that we used for our model are as follows:

$$\sigma = \begin{pmatrix} 1.5 & 1.5 & 1.5 & 1.5 & 1.5 & 1.5 & 1.5 & 1.5 & 1.5 & 1.5 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \end{pmatrix}, \quad a = \begin{pmatrix} 3.1 \\ 3.4 \\ 4.1 \\ 3 \end{pmatrix} \quad \text{and}$$

$$\mu = \begin{pmatrix} -1 & 1.5 & -2 & -2.5 & -1 & 1.5 & -2 & -2.5 & -1 & 1.5 \\ 0 & -2 & 3 & 1 & 0 & -2 & 3 & 1 & 0 & -2 \\ -2.5 & -2 & 1.5 & 3.5 & -2.5 & -2 & 1.5 & 3.5 & -2.5 & -2 \\ -2 & 1 & -1 & 3 & -2 & 1 & -1 & 3 & -2 & 1 \end{pmatrix}. \quad \text{These values}$$

have been chosen such that the Gaussian Mixture Model landscape has four peaks more or less overlapped to emulate human decision making as shown in Figure 1. In such a case, boundaries between classes of evaluation are not precise.

## IV. EXPERIMENTAL RESULTS

### A. Experimental conditions

The genotype is a vector of floating point values; each float value is represented as an array of 12 bits. The vector's dimension is equal to the Gaussian Mixture Model's dimension (3, 5, 7 or 10). The crossover operator is a multipoint crossover with a 100% rate. Mutation rate is 5%. We use a tournament selection operator with 2 individuals for each tournament. For all experiments, we do 100 runs for 100 generations. For the IEC frameworks, fitness is discretized into 5 evaluation levels (as explained in section III.A).

Our experiments are also conducted with population sizes of 16 and 128. The former corresponds to the population size when a human IEC user runs IEC experiments, while the latter corresponds to that for a normal EC search and is conducted as the reference for comparing with the convergence characteristics of IEC. Population size are also constrained by the implementation of single-elimination tournament (cf. section II.A); size needs to be a *power of two*, otherwise some individuals will not have the same number of wins.

As mentioned in section II.B, different strategies exist for DE; we used the standard DE/best/1/bin algorithm.

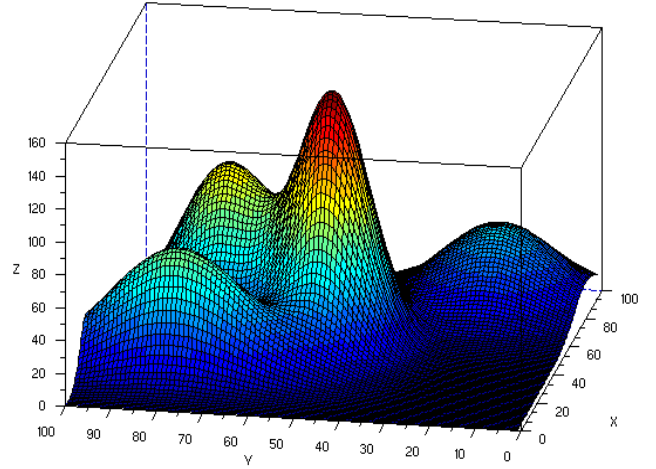


Figure 1. 3D view of a 4D Gaussian Mixture Model used in our experiments.

The following remarks apply to the best fitness, not the average of fitness.

### B. Comparison of DE with three other EC algorithms for reference

In general :

- EC with smaller population size is harder than with greater population size,
- higher dimensional tasks are more difficult,
- tournament GAs use less information for selection than normal GAs, which selects better parents by comparing all individuals, implying that normal GA converges faster than the tournament GA in general, and
- observations in EC research this decade indicate that DE seems to converge faster than normal GA.

Figure 3 to Figure 10 reflect all these general observations. For this reason, we may believe that our four EC algorithms ran correctly though we have not applied sign tests to these results.

### C. Comparison of IDE and three other IEC algorithms

From the practical IEC point of view, we should note that a realistic number of generations would be 10 - 20 (maybe at most 30) and the population size would be up to 20. However, convergence curves until 100 generations in the graphs are useful for observing the general characteristics of IEC.

From the convergences up until 20-30 generations in Figure 11 to Figure 14, which are practical IEC conditions, it seems that there are no significant differences among the 4 EC methods except with the simplest task of the 3-D Gaussian Mixture Model. We should apply a sign test or Wilcoxon sign test and confirm whether this view is correct; we will do this in the next session.

Comparison of Figure 11 to Figure 14 and Figure 15 to Figure 18 shows that:

- normal IGA with a large population size works well in early generations, and
- IDE runs better in later generations, i.e. IDE is a slow starter compared to IGA. Although we cannot use IEC

with a big population size and Figure 15 to Figure 18 are not realistic, analysis of the reason for the IDE's slow start may give us a hint that can be applied under practical IEC conditions with lower population size and fewer generations, improving IDE under realistic conditions.

#### D. Results

Unlike DE and GA, the two tournament GAs use rank order fitness, i.e. relative fitness. All IDE, IGA, and tournament IGA use relative fitness. If we make a graph of relative fitness versus generations, of course no convergence will be observed. Normalization of fitness is also unable to solve this limitation. However, to make comparisons, we must observe their convergence in a searching space using the absolute function values of the evolved individuals based on relative fitness.

#### V. DISCUSSION

We statistically tested whether DE or IDE were significantly better or worse than the best of the others at each generation. Results are shown in Figure 2. Our observations are that:

- DE is always the best or at least it is not significantly poorer than the others in the earlier generations. For three different task complexities with small (16) and large(128) population size, it achieved the best fitness after 10-20 generations.
- IDE with a population size of 16 shows a similar tendency. However, IDE becomes significantly better than the others after about the 10<sup>th</sup> generation for a simple task (3-D Gaussian mixture model) and the 40<sup>th</sup> generation for complex tasks (5-D, 7-D, and 10-D models).
- IDE with a population size of 128 is not realistic as an IEC, and its results are therefore not important.

#### VI. CONCLUSION

Better IEC user interface and faster convergence are necessary to reduce IEC user fatigue. We proposed a paired comparison-based IDE that can greatly reduce IEC user

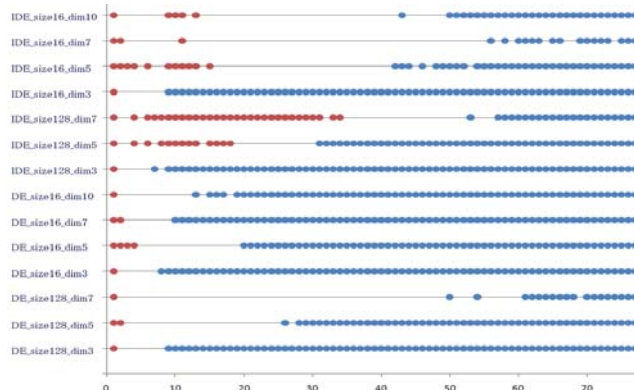


Figure 2. Sign test for DE (IDE); blue dots mean that DE (or IDE) is significantly better than others in the (1) case, and the red dots mean that DE (IDE) is significantly poorer than at least one of others.

fatigue over methods that compare all individuals. It is especially effective for IEC tasks that display individuals time-sequentially, i.e. sounds or movies. Since this advantage is obvious, we evaluated another key point, convergence speed, and demonstrated the superiority of IDE to IGA and two tournament IGAs. With these two advantages of our proposed IDE established, we can say it is better than conventional IDE, IGA, and tournament IGAs.

The next step of this research is to evaluate whether the advantages of our IDE found though IEC simulations are effective in practice with human IEC. We plan to evaluate the proposed IDE using real human users.

#### VII. REFERENCES

- [1] H. Takagi, "Interactive Evolutionary Computation: Fusion of the Capacities of EC Optimization and Human Evaluation," in IEEE, 2001, pp. 1275-1296.
- [2] D. Costelloe and C. Ryan, "Genetic Programming for Subjective Fitness Function Identification," in Genetic Programming, LNCS, Ed., 2004, pp. 259-268.
- [3] X. Llorà and K. Sastry, "Combating User Fatigue in iGAs : Partial Ordering , Support Vector Machines and Synthetic Fitness," in GECCO, Washington, D.C., USA, 2005, pp. 1363 - 1370.
- [4] X. Llorà, K. Sastry, and F. Alias, "Analyzing Active Interactive Genetic Algorithms using Visual Analytics," in Genetic and Evolutionary Computation (GECCO), Seattle, Washington, USA, 2006, pp. 1417-1418.
- [5] T. Holmes and J. Zanker, "Eye on the Prize: Using Overt Visual Attention to Drive Fitness for Interactive Evolutionary Computation," in GECCO (Genetic and Evolutionary Computation COnference), Atlanta, Georgia, USA, 2008, pp. 1531-1538.
- [6] D. Pallez, P. Collard, T. Baccino, and L. Dumercy, "Eye-Tracking Evolutionary Algorithm to minimize user fatigue in IEC applied to Interactive One-Max problem," in GECCO (Genetic and Evolutionary Computation COnference), University College, London, 2007, pp. 2883-2886.
- [7] B. Johanson, "Automated Fitness Raters for the GP-Music System," University of Birmingham, Masters Degree Final Project 1997.
- [8] Y. Nakano and H. Takagi, "Influence of Quantization Noise in Fitness on the Performance of Interactive PSO," in Congress on Evolutionary Computation, Trondheim, Norway, 2009.
- [9] R. Storn and K. Price, "Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces," Journal of Global Optimization, vol. 11, pp. 341–359, 1997.
- [10] T. Akbal, G. N. Demir, A. E. Kanlikiliçer, M. C. Kus, and F. H. Ulu, "Interactive Nature-Inspired Heuristics for Automatic Facial Composite Generation," in Genetic and Evolutionary Computation Conference (GECCO), Undergraduate Student Workshop, Seattle, WA, USA, 2006.
- [11] A. E. Kanlikiliçer, "Interactive Differential Evolution for Facial Composite Generation," in Genetic and Evolutionary Computation (GECCO), Seattle, USA, 2006.
- [12] B. Kurt, A. S. Etaner-Uyar, T. Akbal, N. Demir, A. E. Kanlikilicer, M. C. Kus, and F. H. Ulu, "Active Appearance Model-Based Facial Composite Generation with Interactive Nature-Inspired Heuristics " in Multimedia Content Representation, Classification and Security, Lecture Notes in Computer Science. vol. 4105, 2006, pp. 183-190.
- [13] P. J. Angeline and J. B. Pollack, "Competitive Environments Evolve Better Solutions for Complex Tasks," in 5th International Conference on Genetic Algorithms, Urbana/Champaign IL, USA, 1993, pp. 264 - 270.
- [14] L. Panait and S. Luke, "A Comparison Of Two Competitive Fitness Functions," in GECCO (Genetic and Evolutionary Computation COnference), 2002, pp. 503 - 511.
- [15] A. P. Engelbrecht, Computational Intelligence: An Introduction, Second Edition ed.: John Wiley and Sons, 2007.

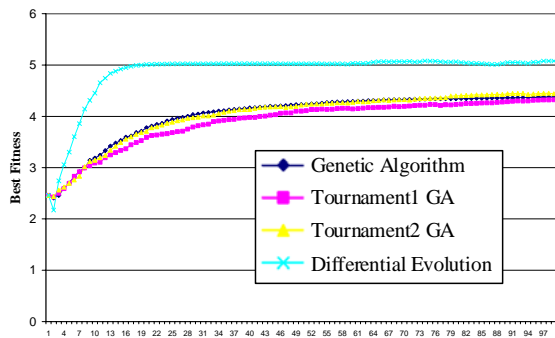


Figure 3. EC Task is 3-D Gaussian Mixture Model (16 individuals).

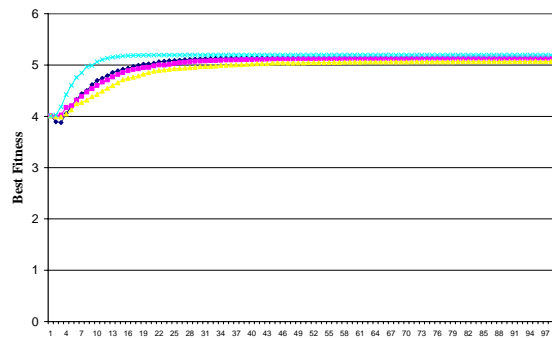


Figure 7. EC Task is 3-D Gaussian Mixture Model (128 individuals).

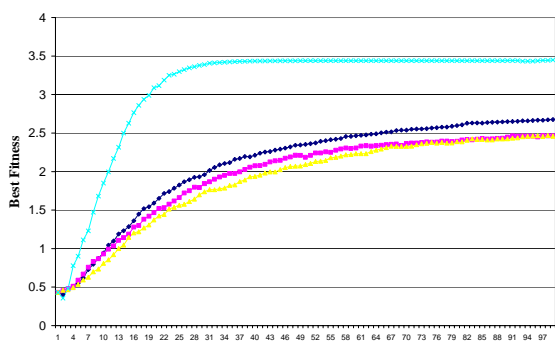


Figure 4. EC Task is 5-D Gaussian Mixture Model (16 individuals).

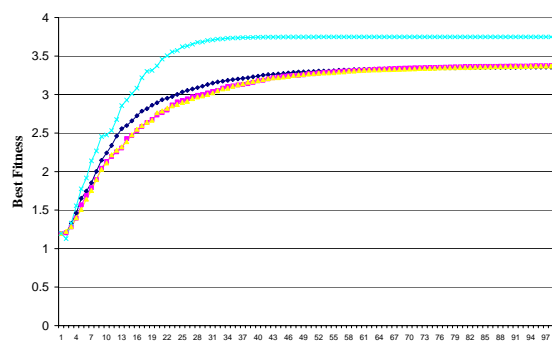


Figure 8. EC Task is 5-D Gaussian Mixture Model (128 individuals).

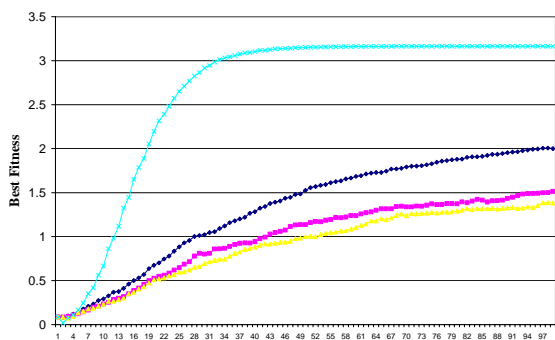


Figure 5. EC Task is 7-D Gaussian Mixture Model (16 individuals).

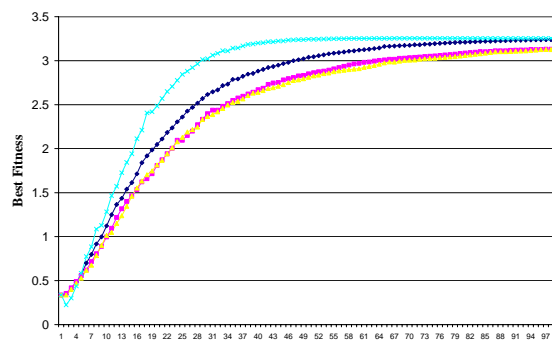


Figure 9. EC Task is 7-D Gaussian Mixture Model (128 individuals).

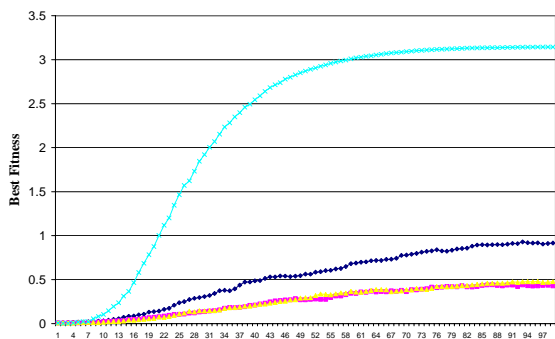


Figure 6. EC Task is 10-D Gaussian Mixture Model (16 individuals).

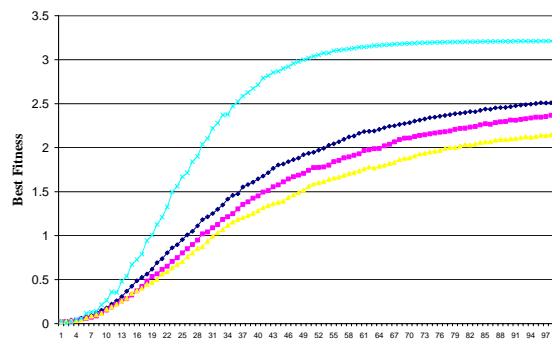


Figure 10. EC Task is 10-D Gaussian Mixture Model (128 individuals).

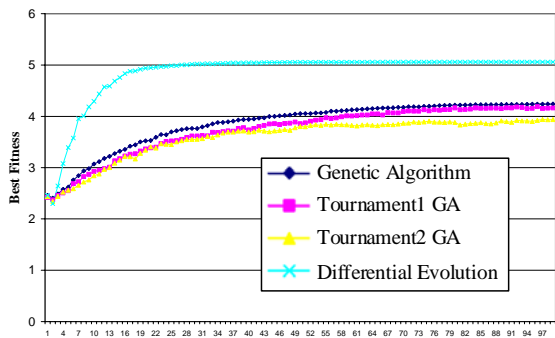


Figure 11. IEC Task is 3-D Gaussian Mixture Model (16 individuals).

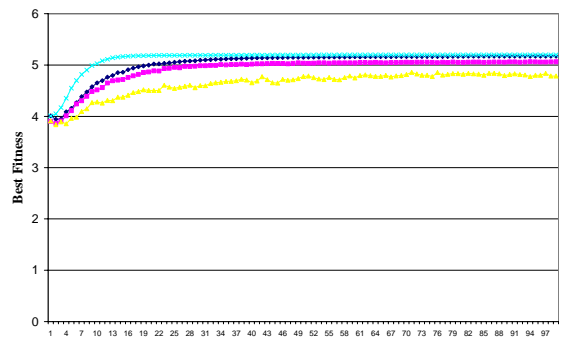


Figure 15. IEC Task is 3-D Gaussian Mixture Model (128 individuals).

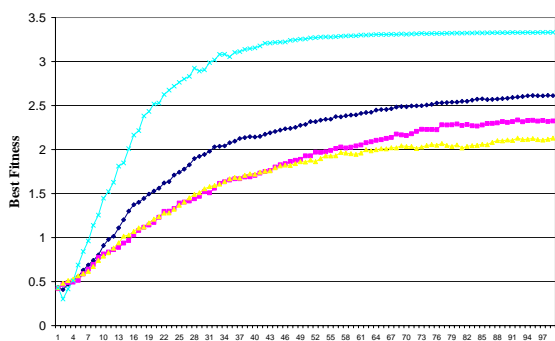


Figure 12. IEC Task is 5-D Gaussian Mixture Model (16 individuals).

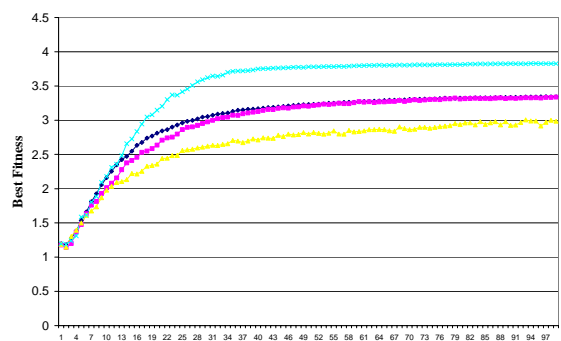


Figure 16. IEC Task is 5-D Gaussian Mixture Model (128 individuals).

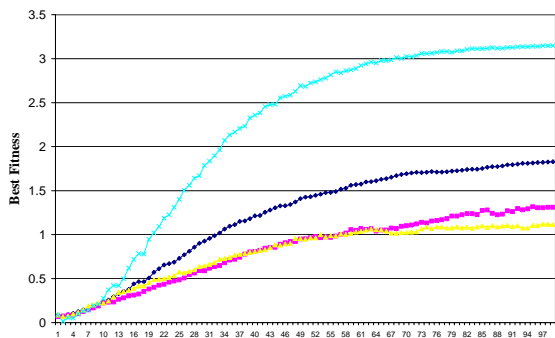


Figure 13. IEC Task is 7-D Gaussian Mixture Model (16 individuals).

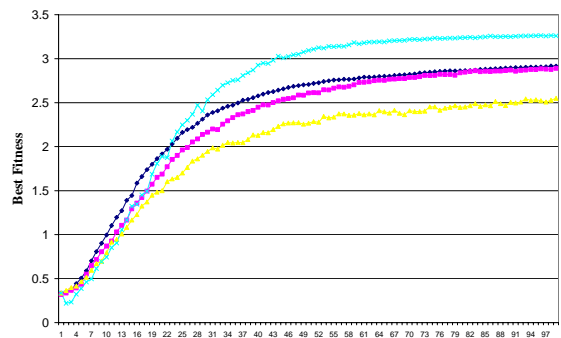


Figure 17. IEC Task is 7-D Gaussian Mixture Model (128 individuals).

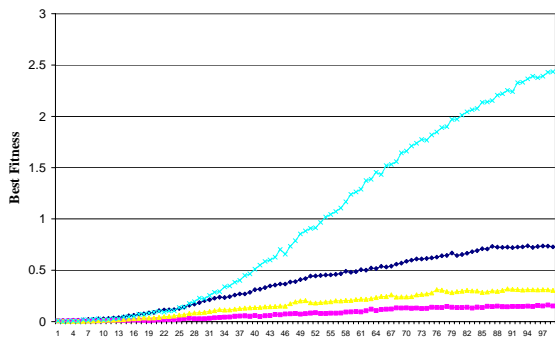


Figure 14. IEC Task is 10-D Gaussian Mixture Model (16 individuals).

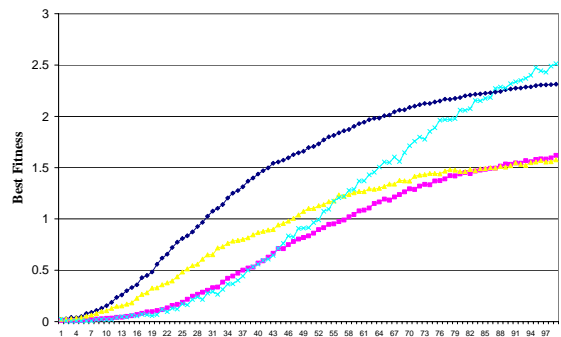


Figure 18. IEC Task is 10-D Gaussian Mixture Model (128 individuals).