Genetic Heuristic for Search Space Exploration

Manuel Clergue and **Philippe Collard**
I3S Laboratory, University of Nice - Sophia Antipolis
2000 Rte des Lucioles, les Algorithmes/Bat. Euclide
F-06410 Biot - Sophia Antipolis
FRANCE

Abstract

This paper deals with the way dual genetic algorithms (DGA), an extension of the standard ones, explore the search space. After a brief introduction presenting genetic algorithms and dualism, the fitness distance correlation is discussed in the context of dualism. From this discussion, a conjecture is made about the genetic heuristic used by dual genetic algorithms to explore the search space. This conjecture is reinforced by the visualization of the population centroid trajectories in the plane fitness distance. These trajectories help to point out "leg-up" behaviors, which allow the dual genetic algorithm to reach the global optimum from walks on deceptive paths.

1 Introduction

1.1 Genetic algorithms

Genetic algorithms are adaptive systems inspired from Darwin's theories on natural evolution. Their fundamental bases were defined twenty five years ago by Holland ([Holland, 1975]) in his book on natural and artificial adaptation. Much works have been done since, in particular to improve their function optimization abilities ([Goldberg, 1989]).

The basic algorithm is simple: an initial population of N individuals is randomly drawn and evaluated given a *fitness* function; according to this fitness value, the individuals are allowed to reproduce in such a way that the number of expected children of each individual is proportional to the fitness; children are mated together, mutated, in order to obtain a new population of size N; and the algorithm loops until a stop condition is reached. Individuals may be problem solutions, function parameters, and so on.

Improvements consist in modifying the evaluation step, the reproduction step, the mating and mutation step, or even the algorithm itself. But the main problem is always to choose an appropriate representation for the individuals. Dualism was introduced in order to bypass this problem by having an algorithm less sensitive to representation shortcomings.

1.2 Dualism

Dual genetic algorithms (DGA) are an extension of GA (see [Collard and Escazut, 1995] for more details). Their principles are the same as the standard GA (SGA) ones. They only differ in the representation of the individuals. Within DGA, individuals are enhanced with an additional bit, termed *meta bit*, which controls their interpretation.

Let Ω be the binary space where the problem is originally defined, that is the *basic search space*. The space $\langle\Omega\rangle = \{0,1\} \times \Omega$ is called the *mirror space*. Strings from $\langle\Omega\rangle$ are interpreted as follows: 0ω is intended as ω while 1ω is intended as $\overline{\omega}$, the binary complement of ω. Apart from this interpretation step, this is a classical GA which is applied on the mirror space.

Elements from 0Ω are known as *chromosomes*, and elements from 1Ω as *anti-chromosomes*. A chromosome and an anti-chromosome corresponding to the same element of Ω, form a *dual pair*. For instance, 00100 and 11011, both represent the same individual from Ω, 0100.

There are many studies on application of DGA in several contexts: time dependent optimization, multi-objective optimization and classifier systems.

2 Fitness Distance Correlation

Some researchers have established a correspondence between evolutionary algorithms and heuristic state space search ([Jones and Forrest, 1995; Tackett, 1995]). This correspondence is based on the fact that both approaches can be seen as searching labeled graphs. Evolutionary landscape matches state space, individual potential solution and Fitness function Heuristic function. In the landscape model, the label on a vertex is a measure of the worth of that vertex as a solution. Whereas for many heuristic searches, as Means-Ends Analysis or the A* algorithm, the label is interpreted as a measure of the distance to a goal. There are many results that show that the better an heuristic is as an estimate to the function which gives the exact distance to the goal, the better it will perform. From these results, Jones ([Jones and Forrest, 1995]) suggests that the ideal fitness function for evolutionary algorithms must provide a measure of how far away a good solution is. So he proposes to use the relationship between fitness and distance to the goal as a measure of difficulty for a genetic algorithm.

The definition of the fitness distance correlation (FDC) is quite simple. Given a set $F = \{f_1, f_2, \ldots, f_n\}$ of fitness values from n individuals and the corresponding set $D = \{d_1, d_2, \ldots, d_n\}$ of distances from them to the nearest optimum, the correlation coefficient is calculated as:

$$r = \frac{c_{FD}}{\sigma_F \sigma_D}, \text{ where } c_{FD} = \frac{1}{n} \sum_{i=1}^{n} (f_i - \overline{f})(d_i - \overline{d})$$

is the covariance of F and D, and σ_F, σ_D, $\overline{f}$ and $\overline{d}$ are the standard deviations and means of F and D.

From the correlation coefficient r, problems may be divided in three classes:

1. **deceptive problems** ($r \geq 0.15$), for which the fitness increases with the distance to optimum.
2. **hard problems** ($-0.15 < r < 0.15$), for which there is no correlation between fitness and distance.
3. **easy problems** ($r \leq -0.15$), for which the fitness increases when the distance decreases.

Hard problems are in fact hard to predict, since in this case, the FDC brings little information. So, Jones recommends to examine the scatter plot distance vs. fitness, especially when r is near zero. This allows to distinguish between a problem such as *a needle in a haystack* (NIAH) from a problem with a symmetrical scatter plot, for example. They both have a null coefficient, but NIAH is a hard problem, while there are symmetrical problems which are easy.

Jones recommends to use distances related to genetic operators. However, it is not sure that such distances exist, and if so, that they are easily computable. Jones claims that Hamming distance allows a first approximation of a FDC more related to the operators.

In order to validate the accuracy of the FDC, Jones uses it on problems with known difficulty. Results show undoubtedly the predictive ability of the FDC, even calculated with Hamming distance.

These results are surprising, since the distances actually implied by operators are not taken into account. The only operator taken into account is the mutation, as the Hamming distance between two individuals is directly related to the number of bit flips to pass from one individual to the other.

Altenberg ([Altenberg, 1997]) goes further. He says the fact that the FDC is only a statistical and static measure, based on a distance which is apparently only bound to mutation, implies two assumptions: either Hamming distance is connected to the way genetic algorithms work; or this relation exists in a fortuitous way among the test set chosen by Jones. In which case, counter examples exist for which this relation does not hold, and which, therefore, deceive the FDC.

Since there seems that there is no relation between recombination operators and Hamming distance, and that mutation is supposed to play a marginal role in genetic algorithms, Altenberg claims that it is possible to construct a counter example. The counter example he constructs is GA-easy, but the correlation between distance and fitness to optimum is null by construction. Further, the observation of the scatter plot gives no more informations.

This counter example deceives the Jones' conjecture which claims that if the FDC is close to 0 and if the scatter plot exhibits no particular structure, then the problem is GA-difficult. Moreover, Quick *et al.* ([Quick *et al.*, 1998a]) construct a class of problems, called *ridge functions*, which are GA-easy with a high positive correlation.

While the Altenberg's counter example is prone to discussion, in particular on the definition of the GA-easiness, the counter example of Quick *et al.* is clear: there are functions that the FDC predicts misleading and which are in fact easy. Nevertheless, these two counter examples exploit known weaknesses of the FDC: its nullity for the symmetrical functions and the low contribution of a particular path in the global calculation. Besides, Quick *et al.* ([Quick *et al.*, 1998b]) recognize that the FDC calculated with the points actually sampled by the GA gives better results.

In a more general way, Naudts and Kallel ([Naudts and Kallel, 1998]) show that a FDC drawback is its great sensibility to non-linear scaling. Kallel and Schoenauer ([Kallel and Shoenauer, 1997]) show in a more formal frame the irrelevance of FDC as performance predictor for GA, in an article judiciously entitled: "no universal statistical measure but a set of hints".

Nevertheless, the success of the FDC on a large number of functions remains an unsolved question. Collard *et al.* ([Collard *et al.*, 98]) bring some elements of response, exhibiting a correlation between Hamming distance and instability implied by crossover.

3 FDC and Dualism

This section deals with the relations that can be established between FDC theory and dualism. In the first part, we define the dual distance that should be used to calculate the FDC in the dual case. In the second part, a conjecture on dynamical behavior of DGA is proposed.

3.1 Dual FDC

The dualism implies two ways to transform an individual $x \in \Omega$ to another individual $y \in \Omega$. The first one, called direct, is equivalent to mute the $d_H(x, y)$ different bits between x and y. This corresponds to a walk on the boolean hypercube and the associated distance is simply the Hamming distance. The second one is equivalent to get x and y each in a different mirror subspace and to mute the $\lambda - d_H(x, y)$ identical bits, where λ is the chromosome length. This corresponds to a step through the boolean hypercube followed by a walk on it. We called the associated distance the *codistance*, d_c. It can be established that :

$$d_c(x, y) = 1 + \lambda - d_H(x, y)$$

The 1 is for the step through, and $\lambda - d_H(x, y)$ is the length of the walk on the hypercube.

The *dual distance* between two individuals from the basic space x and y, is naturally defined as follows :

$$d_{dual}(x,y) = \min(d_H(0x,0y), d_H(0x,1\overline{y}), d_H(1\overline{x},0y), d_H(1\overline{x},1\overline{y}))$$

where d_H is the Hamming distance, $0x \in \langle\Omega\rangle$ and $1\overline{x} \in \langle\Omega\rangle$, are the two dual individuals corresponding to x. Since :

$$\begin{aligned} d_H(\overline{x},\overline{y}) &= d_H(x,y) \\ d_H(x,\overline{y}) &= \lambda - d_H(x,y) \end{aligned}$$

where λ, is the length of x and y, we get:

$$\begin{aligned} d_{dual}(x,y) &= \min(d_H(0x,0y), d_H(0x,1\overline{y})) \\ &= \min(d_H(0x,0y), \lambda' - d_H(0x,0y)) \\ &= \min(d_H(x,y), \lambda' - d_H(x,y)) \end{aligned}$$

λ' being the length of strings over $\langle\Omega\rangle$ ($\lambda' = \lambda + 1$). This result expresses that the dual distance between x and y is the minimum of the Hamming distance and the codistance.

The dual distance remains a distance over the basic space Ω. But, it can be easily extended to the mirror space $\langle\Omega\rangle$. We note $\langle d\rangle$ the distance over the mirror space $\langle\Omega\rangle$ to the nearest optimum. It is easily shown that :

$$\langle d\rangle(x) = d_{dual}(\dot{x}, x^\star)$$

where $\dot{x}$ is the individual of Ω corresponding to x, and $x^\star$ is the optimum over Ω. Thus, there is a correspondence between a distance over $\langle\Omega\rangle$ appropriate to calculate a FDC and the dual distance defined over Ω.

Now, we are going to study the predictive capacity of the dual FDC on a well known function family, the *Trap* functions ([Deb and Goldberg, 1993]). They are defined over the unitation of chromosomes, that is the number of genes set to '1'. They admit two optima, a global one and a local one, one being the logical complement of the other. They are parametrized by two values x_b and r. The first one, x_b allows to set the width of the attractive basin for each optima, and r sets their relative importance. The function f_T is so defined by :

$$f_T(x) = \begin{cases} 1.0 - \frac{Un(x)}{x_b} & \text{if } Un(x) \leq x_b \\ \frac{r(Un(x)-x_b)}{1.0-x_b} & \text{elsewhere} \end{cases}$$

where $Un(x)$ is the unitation divided by λ, the length of chromosomes (in the experiments, $\lambda = 20$).

We calculated the exact value of FDC (that is with all the points of Ω), in function of x_b and r. Results are shown on figure 1. They are roughly identical to the Deb & Goldberg [Deb and Goldberg, 1993] ones, who used another theoretical framework to obtain the conditions on x_b and r for which the *Trap* function is deceptive. These results are corroborated by the performance obtained with a "real" GA.

We define performance as the number of runs for which the global optimum is found in less than a given number of generations (200), divided by the total number of runs (40). The figure 2 presents such performance measures obtained with a GA on the *Trap* functions. It is shown that the experimental deceptive area (where performance is less than 0.05) corresponds approximately to the theoretical deceptive area found with the FDC.

For DGA, the predicted deceptive area is less important (figure 3). It can be shown, from figure 4, that there is no couple (x_b, r) for which the global optimum is never found. This reinforces the hypothesis made elsewhere ([Collard and Aurand, 1994]) for which there is no totally deceptive problem for DGA.

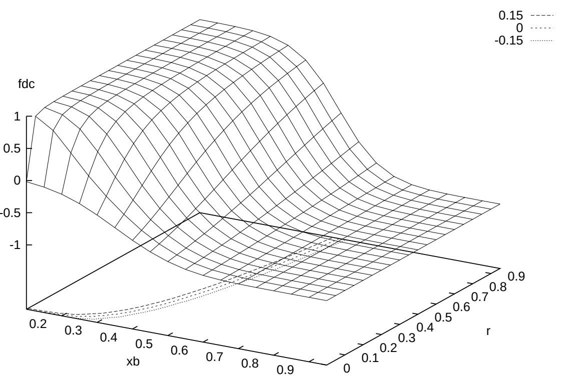

Figure 1: Standard FDC for *Trap* functions. Lines delimit areas in the plane (x_b, r) for respectively deceptive, hard and easy *Trap* functions, according to the Jones' classification.

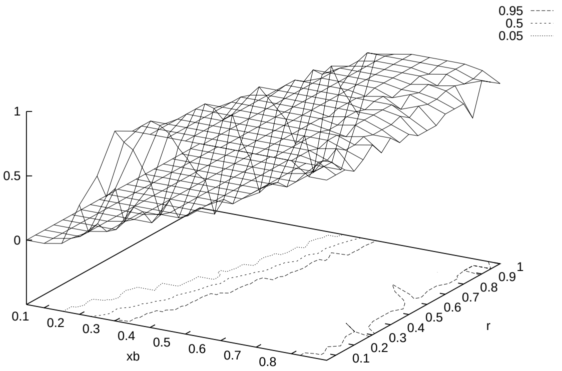

Figure 2: Performances of the SGA on *Trap* functions.

3.2 Fitness Driven Distance

Now, we are concerned in studying the dynamical behaviors of DGA and the relation that can be made with FDC. During evolution, the DGA has to make choices between direct path and indirect path, that is, for each dual pair, it has to choose between the chromosome and the

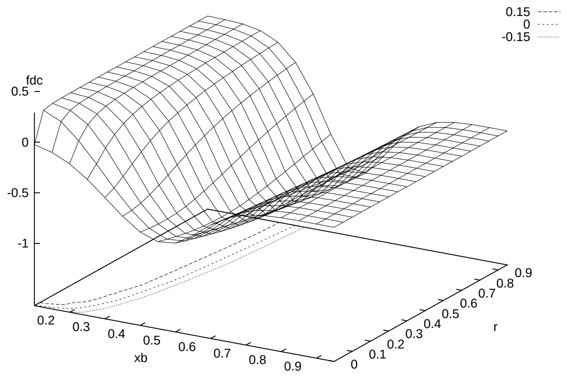

Figure 3: Dual FDC for *Trap* functions.

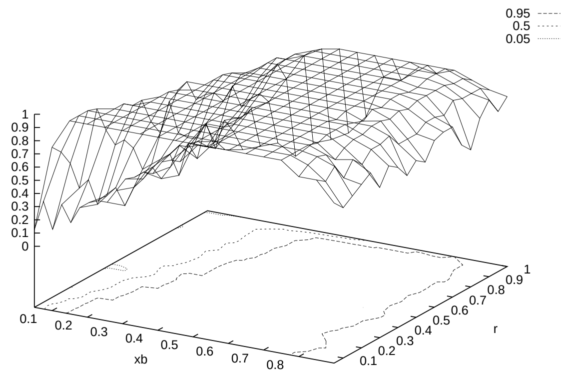

Figure 4: Performances of the DGA on *Trap* functions.

anti-chromosome. Indeed, keeping both of them implies too much instability. Such a choice cannot be influenced directly by fitness, since dual individuals have, by definition, the same fitness. So, there is no direct explanation on the way these choices are made.

In the context of FDC, this choice may be expressed by the definition of a distance over Ω to the optimum. If the optimum is considered in one of the two mirror subspaces, say 0Ω, choosing an individual in the same subspace is equivalent to consider the Hamming distance between it and the considered optimum; and choosing an individual in the other subspace is equivalent to consider the codistance. Thus, there is a strong relation between a strategy of choice for the DGA and the effective distance over Ω used to calculate the FDC.

In the next, we are going to show that the best strategy, with regard to the FDC, is that which consists in choosing the individual in the same mirror subspace than the optimum if its fitness is above average, and in the other subspace else. This strategy is instantiated by the so-called *fitness driven distance* (f_d), defined as follows :

$$d_f(x) = \begin{cases} \min(d_H(x), d_c(x)) & \text{if } f(x) < \overline{f} \\ \max(d_H(x), d_c(x)) & \text{elsewhere} \end{cases}$$

It may be obviously established that:

Property 1 *The fitness-distance correlation coefficient calculated with the fitness driven distance is always negative.*

Moreover, the following property is verified:

Property 2 *The fitness driven distance is the best strategy of choice between dual individuals with regard to the fitness-distance correlation.*

Demonstration: Let d be a distance over Ω representing a choice strategy for each individual between Hamming distance and codistance to the optimum. It is obvious that:

$$\sigma_d = \sigma_c = \sigma_H$$

Then, each term $c(i) = (f(i) - \overline{f})(d(i) - \overline{d})$ involved in the covariance calculation is

$$c_d(i) = \begin{cases} c_H(i) & \text{or} \\ c_c(i) \end{cases}$$

Since $c_f = \min(c_H(i), c_c(i))$, it is easily established that the covariance calculated using the fitness driven distance is inferior or equal to the covariance using any other distance d.
So, the correlation coefficient calculated with the fitness driven distance is inferior or equal to the one calculated with any other distance d. ■

We conjecture that the DGA tends to follow the strategy expressed by the fitness driven distance.

4 FDC and Population Dynamics

In this section, we give more insight in the DGA dynamical behavior, by plotting the trajectories of the population centroid on the fitness-distance plane. In particular, we point out the "leg-up" behavior, which allows the algorithms to lean on a deceptive attractor to reach the global optimum.

4.1 Fitness distance trajectories

Representing the population dynamics is a difficult task. Indeed, there is a trade-off to do between the information being presented and the clarity of the representation. The first step is to choose a relevant plane. The fitness distance space seems to be an interesting one. The second step consists in choosing how the population is to be represented in this plane. To do so, it is natural to plot the position at each generation of the centroid of the population, that is, the point corresponding to the mean distance to the optimum and the mean fitness of the population.

4.2 "Leg-up" : a way to converge

In order to exhibit the choices made by a DGA during evolution, we are going to consider two functions, the previously introduced *Trap* function ($\lambda = 50$, $x_b = 10$, $r = 0.8$), and a modified *Trap* function called *bi-Trap*.

The *Trap* function

The trajectory of the population centroid for the SGA (see figure 5) is not surprising. The population follows the deceptive branch of the landscape toward the local optimum.

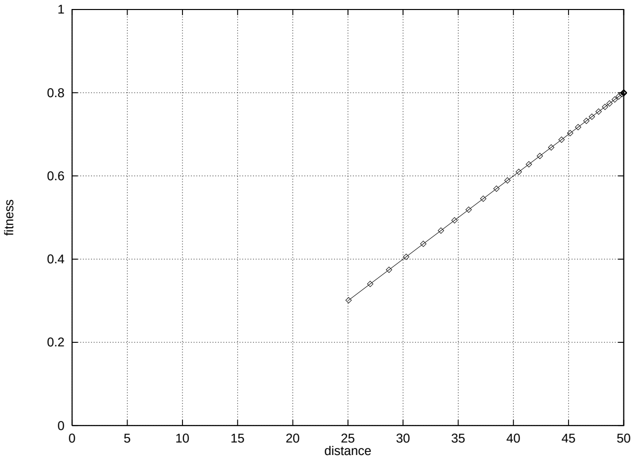

Figure 5: SGA: trajectory for the *Trap* function.

The figure 6 represents the trajectory for a DGA considering distances in the basic space. First, as for the SGA, the population converges toward the deceptive optimum. But after a while, the population crosses the space to converge on the global optimum. This is the "leg-up" phenomenon: the algorithm leans on the local optimum to reach the global one.

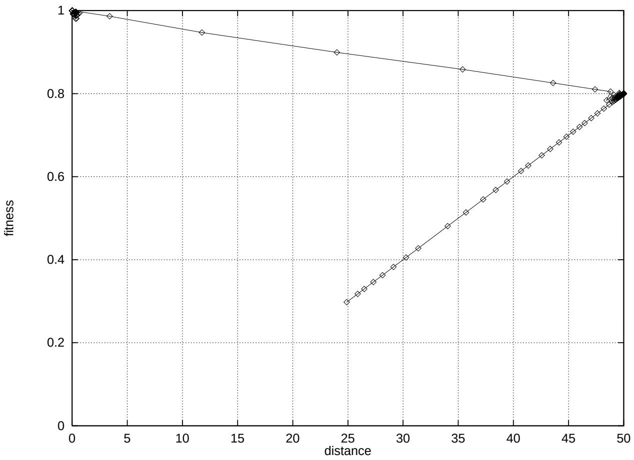

Figure 6: DGA: trajectory for the *Trap* function in the basic space.

The representation in the mirror space $\langle\Omega\rangle$ (see figure 7) gives another point of view of this behavior. In particular, it shows that the population begins to move on the mirror subspace 0Ω (solid line). Then it jumps on 1Ω to reach the global optimum (dashed line).

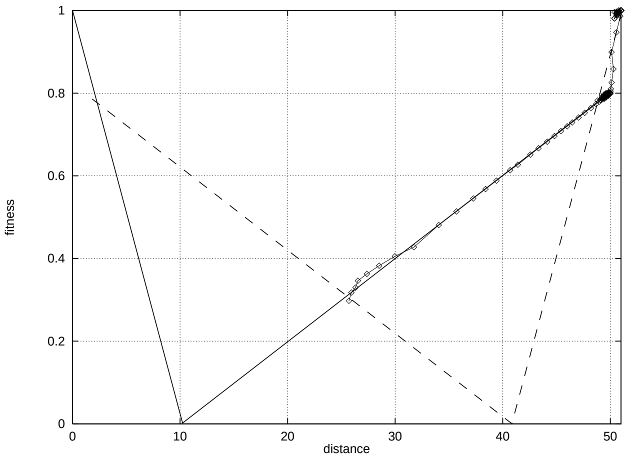

Figure 7: The *Trap* function in $\langle\Omega\rangle$. The solid line represents the 0Ω individuals, and the dashed line, the 1Ω ones. The points are the trajectory of the population.

The *bi-Trap* function

The *Trap* function involves an unique "leg-up". Indeed, although the space Ω is crossed from part to part, this corresponds to the change of only 1 bit in $\langle\Omega\rangle$. But this strategy may be used by DGA, even when the local optimum is not in the vicinity of the global optimum complement. Moreover, there may be several successive "leg-up" during evolution.

Now, we are going to introduce a function involving two "leg-up". The *Trap* function is modified in order to get many local optima (see figure 8) : The binary complement of the global one and the individuals situated at unitation 10.

According to the initial population, the SGA behaves half of the time as for the *Trap* function, that is it converges toward the complement of the global optimum. Any time else, it converges toward the other local optima, from where it occasionally reaches the global optimum.

With DGA, the trajectory of the population exhibits clearly two "leg-up" (see figure 8). Initially, the population follows the left branch. It jumps a first time to follow the right one and converge toward the deceptive optimum. Then it makes another jumps to reach the global optimum as with the standard *Trap*. These jumps in Ω correspond in fact to mirror subspace changes, as shown figure 9.

5 Conclusion

Dual genetic algorithms seem to be a promising improvement of genetic algorithms. However, it lacked explanation on why they are so efficient and on the way they behave. In this paper, we bring some element of response, using the fitness distance correlation, and plotting their dynamical behaviors in the fitness distance plane.

The trajectories plots in the distance fitness space exhibit the "leg-up" phenomenon. The choice of the *Trap* functions was influenced by our needs to get a function

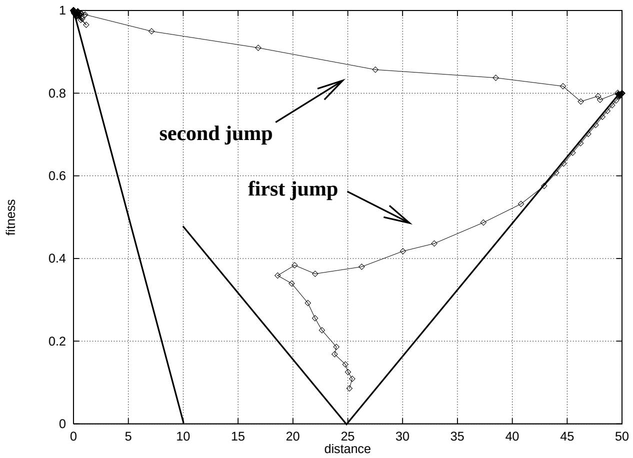

Figure 8: The *bi-Trap* function in Ω, along with the trajectory of a DGA.

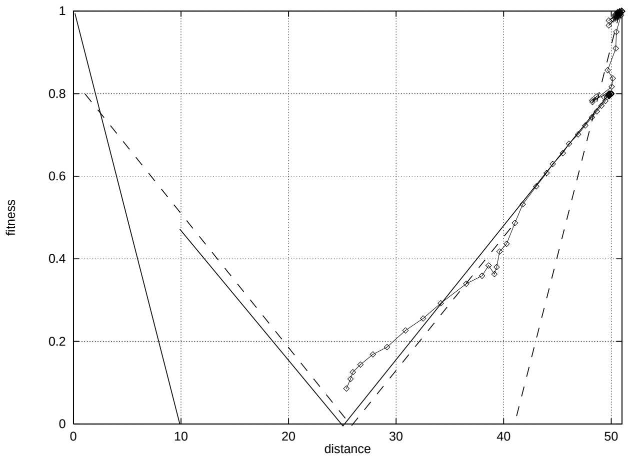

Figure 9: The *bi-Trap* function in $\langle\Omega\rangle$. The solid line represents the 0Ω individuals, and the dashed line, the 1Ω ones. The points are the trajectory of the population.

with a simple representation in this space. This way, the position of the centroid gives more informations about the distribution of chromosomes and anti chromosomes, than with functions with fuzzier representation. However, in the general case, it may be supposed that this jumping behavior is also present, while less observable.

This reinforces our conjecture, firstly made with the *fitness driven distance*, for which, during evolution, the DGA is confronted to choices between dual individuals, and that it tends to make such choices in a way which optimizes the correlation between fitness and distance to the global optimum.

References

[Altenberg, 1997] L. Altenberg. Fitness Distance Correlation Analysis: An Instructive Counterexample. In Thomas Bäck, editor, *Proceedings of the Seventh International Conference on Genetic Algorithms (ICGA97)*. Morgan Kaufmann, 1997.

[Collard and Aurand, 1994] P. Collard and J.P. Aurand. DGA: An efficient genetic algorithm. In A.G. Cohn, editor, *ECAI'94: European Conference on Artificial Intelligence*, pages 487–491. John Wiley & Sons, 1994.

[Collard and Escazut, 1995] P. Collard and C. Escazut. DCS: A promising classifier systems. In D.W. Pearson, N.C. Steele, and R.F. Albrecht, editors, *Proceedings on the second International Conference on Neural Networks and Genetic Algorithms*, pages 14–17, Wien, NY, 1995. Springer-Verlag.

[Collard *et al.*, 98] P. Collard, A. Gaspar, M. Clergue, and C. Escazut. Fitness distance correlation, as statistical measure of genetic algorithm difficulty, revisited. In *Proceedings of the European Conference on Artificial Intelligence*, pages 650–654. John Witley & Sons, Ltd, 98.

[Deb and Goldberg, 1993] K. Deb and D. E. Goldberg. Analysing deception in trap functions. In L. D. Whitley, editor, *Foundations of Genetic Algorithms 2*, pages 93–108. Morgan Kaufmann, 1993.

[Goldberg, 1989] D. E. Goldberg. *Genetic algorithms in search, optimization, and machine learning.* Reading, MA: Addison-Wesley, 1989.

[Holland, 1975] J. H. Holland. *Adaptation in natural and artificial systems.* Ann Arbor: University of Michigan Press, 1975. Second printing in 1993, The MIT Press.

[Jones and Forrest, 1995] T. Jones and S. Forrest. Fitness distance correlation as a measure of problem difficulty for genetic algorithms. In L. Eshelman, editor, *ICGA'95: Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 184–192. Morgan Kaufmann, 1995.

[Kallel and Shoenauer, 1997] L. Kallel and M. Shoenauer. A priori comparison of binary crossover: no universal statistical measure but a set of hints. In *Artificial Evolution*, pages 287–299. Springer, 1997.

[Naudts and Kallel, 1998] B. Naudts and L. Kallel. Some Facts about So Called GA-Hardness Measure. Technical report, University of Antwerp, 1998.

[Quick *et al.*, 1998a] R.J. Quick, V.J. Rayward-Smith, and G.D. Smith. Fitness Distance Correlation and Ridge Functions. In *Proceedings of the fifth Conference on Parralel Problems Solving from Nature*, 1998.

[Quick *et al.*, 1998b] R.J. Quick, V.J. Rayward-Smith, and G.D. Smith. Ridge Functions. Technical report, University of East Anglia, 1998.

[Tackett, 1995] W. A. Tackett. Greedy recombination and genetic search on the space of computer programs. In L. D. Whitley and M. D. Vose, editors, *Foundations of Genetic Algorithms 3*, pages 271–298. Morgan Kaufmann, 1995.