Alternative Random Initialization in Genetic Algorithms

Leila Kallel
CMAP – URA CNRS 756
Ecole Polytechnique
Palaiseau 91128, France
Leila.Kallel@polytechnique.fr

Marc Schoenauer
CMAP – URA CNRS 756
Ecole Polytechnique
Palaiseau 91128, France
Marc.Schoenauer@polytechnique.fr

Abstract

Though unanimously recognized as a crucial step in Evolutionary Algorithms, initialization procedures have not been paid much attention so far. In bitstring Genetic Algorithms, for instance, the standard 0/1 equiprobable choice for every bit is rarely discussed, as the resulting distribution probability over the whole bitstring space is uniform. However, uniformity is relative to a measure on the search space. First, considering the measure given by the density of 1's, the *Uniform Covering* initialization procedure is naturally designed. Second, taking into account the probability of appearance of sequences of identical bits leads to design another alternative initialization procedure, the *Homogeneous Block* procedure. These procedures are compared with the standard initialization procedure on several problems. A priori comparison is achieved using Fitness-Distance Correlation. Actual experiments demonstrate the accuracy of these FDC-based comparisons, and emphasize the usefulness of the two proposed procedure.

1 Introduction

Initialization is recognized to be a crucial issue in all Evolutionary Algorithms (EAs) in general, and in Genetic Algorithms (GAs) in particular. Even if all theoretical convergence results on Evolutionary Algorithms are proved whatever the initial population (see e.g. (Rudolph 1997)), all EA practitioners have experienced that a bad initialization procedure can, in the best case modify the online performance (i.e. increase the time-to-solution), and, in the worst case, prevent the convergence toward the global optimum.

Very few works actually address this critical issue. In particular, the bitstring-uniform standard initialization procedure of GAs is hardly discussed, due to its optimality with respect to the criteria of both uniformity over the search space, and genetic diversity.

The situation can be quite different for some real world problems. For instance in the context of inverse problems in Structural Mechanics, where each individual represents a partition of a 2-dimensional domain into two different materials (Schoenauer 1996; Schoenauer, Jouve, & Kallel 1997), the solution can be known to contain far more 0's than 1's (small inclusions of one material into another), and is likely to contain contiguous blocks of 0's and 1's. In that specific case, the standard uniform sampling procedure can be improved by incorporating this domain knowledge into the initialization procedure (Kallel & Schoenauer 1996).

This paper argues that such alternative criteria can be useful in the general binary framework, to measure the uniformity and diversity of a random population, leading to define alternative initialization procedures. The *Uniform Covering* procedure is based on considerations on the density of 1's and 0's in the bitstring. The *Homogeneous Block* procedure is designed to increase the probability of occurrence of sequences of identical bits in the bitstrings.

An important issue is then the choice of an initialization procedure for a given problem. The proposed solution is based on the Fitness Distance Correlation (FDC) introduced in (Jones & Forrest 1995). This framework is extended to the (more realistic) case where the global optimum of the problem is unknown. This allows to define an a priori estimate of the validity of any random initialization procedure for a given problem. The accuracy of this estimate is studied both from the heuristical point of view, and experimentally on standard problems from the GA literature.

The paper is organized as follows: First, some related works are reviewed and discussed in section 2. Section 3 then discusses some drawbacks of the standard bitstring-uniform initialization; it introduces two other randomness criteria and the corresponding initialization procedures. Section 4 recalls the principle of FDC, then extends this framework to problems for which the global solution is unknown. Section 5 uses the extended FDC to study the impact of the different initialization procedures on various problems of GA literature, and estimate their a priori validity. These estimates are confronted to actual results of intensive GAs runs, demonstrating both the interest of the alternative initialization procedures proposed, and the accuracy of the FDC predictions.

2 Related work

Early work in (Grefenstette 1987) demonstrates the potential strength of incorporating domain specific knowledge into GAs. This can be achieved by seeding the initial population with fit individuals, or lying in regions of the search space known to be of some interest. Such ideas have been later implemented in the *SAMUEL* (Schultz & Grefenstette 1990). These idea however heavily rely on the user, and its expertise in the problem domain.

In (de Garis 1990), an iterative scheme is proposed, involving a sequence of fitness functions. A randomly initialized population is evolved by a GA using the first fitness, the resulting population is used as the initial population for a GA using the second fitness, and so on. (Schoenauer & Xanthakis 1993) used that idea for constrained optimization problems. In these works, the idea is to purposely bias the initial population of the GA, but the tool is here another GA. However, this approach is highly problem specific, and requires from the user the design of the appropriate sequence of fitness functions.

A more general approach to improve the initialization procedure of a GA is that of (Bramlette 1991): the initial population is built by taking the best of n randomly chosen individuals. Another general approach is that of (Maresky *et al.* 1995), whose re-start operator is called to re-initialize the population whenever premature convergence is detected. These latter ideas, however, do not discuss which random initialization procedure is used during either the initial deterministic selection or for the re-initialization.

A more recent work addressing precisely the initialization procedure is (Iba 1996), where a new initialization procedure is proposed in the context of Genetic Programming enhancing the diversity of the initial population regarding some specific criteria.

The work presented here goes along the same line: the initialization procedure of bitstring GAs has hardly been discussed, because it is optimal with respect to the usual criterion of uniformity in the search space. However, several degrees of randomness have been defined (e.g. in (Knuth 1981), Chapter 3.5: *What is a random sequence?*), among which uniformity is not the sole criterion. Next section will observe uniformity and diversity from points of view that differ from the usual bit-wise consideration.

3 Bitstring initialization procedures

From now on, the search space will be the binary space $\{0,1\}^n$ of fixed-length bitstrings, and the underlying EA some standard binary GA (Holland 1975; Goldberg 1989) (see section 5.1).

This section first discusses "uniformity" of the standard random procedure generally used during the initialization phase of Genetic Algorithms: Other points of view on uniformity show some drawbacks of that standard procedure, and lead to design other initialization procedures.

3.1 The Bitstring-Uniform procedure

Almost all GAs use as initialization procedure the *Bitstring-Uniform* procedure (BU): it assigns value 0 or 1 with probability 0.5 to every bit.

The argument favoring that procedure is its uniformity in the binary space: using the BU procedure, all points of $\{0,1\}^n$ have equal probability $1/2^n$ to be drawn. Moreover, the *bit-wise* diversity is maximal, too: at every bit position, there will be in mean as many 0's than 1's.

Consider now the point of view of the number of 1's in each bitstring. Denote $\sharp 1(b)$ the number of 1's in bitstring b. The probability distribution of the sum of n independent random boolean variables having probability p to draw 1 follows the well-known binomial law:

$$P[\sharp 1 = k] = C_n^k p^k (1-p)^{n-k}$$

For large values of n (e.g. $n \geq 30$), the binomial law can be approximated by a Gaussian law of mean np and variance $np(1-p)$. Hence for the Bitstring-Uniform initialization procedure ($p = 0.5$), $\sharp 1$ lies in a very restricted range: For instance, for $n = 900$, 99%

of all bitstrings generated by the standard Uniform procedure have a number of 1's in $[411, 489]$. Hence, from the point of view of the number of 1's in the bitstrings, the BU initialization procedure is certainly not uniform.

3.2 Uniform Covering procedure

To address the issue raised in the preceding subsection, the *Uniform Covering* initialization procedure (UC) is designed: for each bitstring, select a density of 1's, uniformly in $[0, 1]$, then choose each bit to be 1 with that specific probability.

The UC procedure of course leads to a uniform probability distribution of $\sharp 1$. But, on the other hand, the probability for a bit at a given position l in a random bitstring to be 1 is 0.5, and the expectation of the total number of 1's in the population is exactly 1/2 of the total number of bits: the UC procedure still achieves maximal bit-wise diversity.

But let us now consider another alternative point of view, considering the size of sequences of identical bits that are likely to occur in a bitstring.

When using the Bitstring Uniform procedure, sequences of k 1's happen at a given locus with probability[1] $1/2^k$. With the Uniform Covering procedure, this probability is $1/(k+1) (= \int_0^1 r^k dr)$.

Depending on the pre-supposed regularity of the solution (in term of sequences of identical bits), it might be useful to further bias the Uniform Covering procedure to favor the emergence of homogeneous sequences.

3.3 Homogeneous Block procedure

An homogeneous block of bits is characterized by its length l and the position of its center c: "adding" the block of 1's of characteristics (c, l) to a bitstring $(b_i)_{i \in [1,n]}$ amounts to set to 1 all bits b_j with $max(1, c - l/2) \leq j < min(n, c + (l+1)/2)$.

The basic idea of the *Homogeneous Block* initialization procedure (HB) is to start from a bitstring initialized to a default value (e.g. 0), and to gradually "add" homogeneous blocks of the other value (1). The critical issue then becomes the number of such blocks, and the way their characteristics are randomly drawn.

In the same vein as for the UC procedure, the number of blocks is adjusted depending on a uniformly drawn target density of 1's: Blocks are added to an initial bitstring of all 1's, until that target density value is reached. The block characteristics are chosen as follows: A local length L_{loc} is randomly chosen, uniformly in $[0, L_{max}]$, where L_{max} is user-defined (typically $n/10$). The center c and the length l of each block are chosen uniformly, respectively in $[0, n]$ and $[0, L_{loc}]$. Finally, to avoid too long loops in the case of high target density and small local length L_{loc}, a maximum number of blocks is authorized (typically n).

Due to the tight control over the density of 1's, the bitwise diversity should be close to optimal. However, because of the limit on the number of blocks, that procedure is slightly biased toward bitstrings having more 0's than 1's. A way to counter-balance that bias is to draw half of the bitstrings using 0 as the default value, and the other half by reversing 0 and 1.

3.4 Comparing initialization procedures

The most simple and common way to compare some parameters settings of Evolutionary Algorithms is to run numerous experiments for each one of the settings, and compare averages of on-line or off-line fitnesses. This experimental method is indeed rather time consuming. However, the lack of efficient and robust measures of difficulty and/or adequation of parameters for a given problem has made this experimental comparative method very popular since the early days of GAs (see e.g. (Schaffer *et al.* 1989)).

Nevertheless, some studies have tried to propose heuristical approaches to the a priori estimation of problem difficulty (Radcliffe 1991; Grefenstette 1995; Kargupta 1995). Among these, the Fitness Distance Correlation (Jones & Forrest 1995) seems suited to comparing initialization procedures: it allows to compare the behavior of a GA on different problems, on the basis of a random sample of individuals built using the initialization procedure.

4 Fitness Distance Correlation

This section first recalls the definition of the Fitness Distance Correlations (FDC) to measure problem difficulty, as introduced in (Jones & Forrest 1995). It then extends this definition to the case where the global optimum is unknown, and finally assesses FDC as a tool to compare initialization procedures.

4.1 FDC coefficient and plots

Consider the problem of optimizing a real fitness function $\mathcal{F}$ defined on a metric search space $\mathcal{E}$ and suppose

[1] This is **not** the probability that a given bitstring contains a sequence of k 1's. The expectation of that probability must be computed, e.g. by recurrence on the bitstring length.

that there exists one global optimum of $\mathcal{F}$. Given a sample of n individuals x_i, with associated fitnesses (f_i) and distances (d_i) to global optimum, the *FDC coefficient* τ is defined as:

$$\tau = \frac{\frac{1}{n}\sum_{i=1}^{n}(f_i - \bar{f})(d_i - \bar{d})}{\sigma_F \sigma_d}, \tag{1}$$

where $\bar{f}, \bar{d}, \sigma_F$ and σ_d are respectively the means and standard deviations of fitnesses and distances over the sample.

A high correlation between the fitness of an individual and its distance to that global optimum characterizes optimization problems that should be easy to solve. For maximization problems for instance, values close to -1 indicate easy problem, while values close to 1 show that the problem is somehow misleading.

However, a single correlation coefficient cannot catch all the flavor of a problem difficulty. Additional information is given by the shape and position of the samples in the fitness $\times$ distance-to-optimum space, (see e.g. section 5.2). Such scatter plots of (fitness,distance) points will be termed *FDC plots* in the following.

4.2 Local FDC

One major limitation of FDC for real problems is that the computations of FDC coefficient and plots requires to already know the solution of the problem at hand. Though FDC based on global optimum allows one to explain and understand previous results on well-known problems of GA literature (Jones & Forrest 1995), it does not apply to real unknown problems.

A "Local Fitness Distance Correlation" has been introduced and advocated in (Kallel & Schoenauer 1996), relaxing this unrealistic prerequisite. The Local FDC computes the correlation coefficient between a sample and the best individual *in that sample*. Such a coefficient, termed τ_{loc}, is computed as in Equation (1), with the difference that all distances are now distances to the "local" optimum (the optimum in the sample). Of course, this coefficient will likely depend on the choice of that local optimum, and its stability with respect to different samplings should be checked carefully. However, results in (Kallel & Schoenauer 1996) demonstrate that, provided this stability toward the local optimum, Local FDC and original – global – FDC results agree for the inclusion problem.

4.3 FDC to compare initialization procedures

One of the main limitations of FDC analyzes is that it does not take directly into account the Evolutionary Algorithm used thereafter. Hence, it appears that FDC-based conclusions regarding problem difficulty are mainly valid for the first generations of the Evolutionary Algorithm (Kallel & Schoenauer 1996).

Returning now to the main point of this paper, the above arguments clearly indicate that FDC might be a good tool for comparing different initialization procedures for the same problem. Initialization is the heart of FDC analyzes, and hence have a direct influence on FDC results. Hence, FDC-based conclusions, known to be accurate for the early generations, should be an efficient tool to compare initialization procedures.

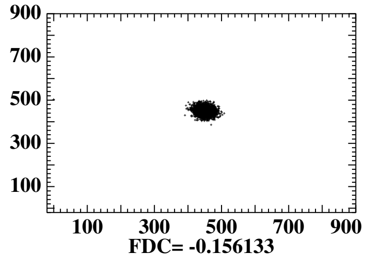
5 Experimental Validation

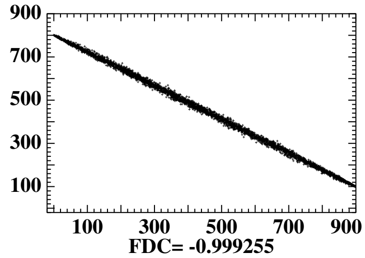
This section experimentally demonstrates that FDC plots can indeed be used to chose between the three initialization procedures presented in section 3: on different test functions, the FDC plots based on the three procedures will be discussed and compared. Moreover, results of actual runs of the same GA starting from population drawn using these procedures will validate the FDC-based approach.

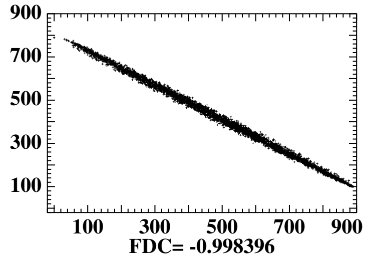
5.1 Experimental conditions

All problems below – except the Long Path problem – are set for bitstrings of length 900. The FDC coefficients and plots are computed on samples of 6000 individuals (see (Kallel & Schoenauer 1996) for an adaptive adjustment of the size of the sample in the case of continuous variables).

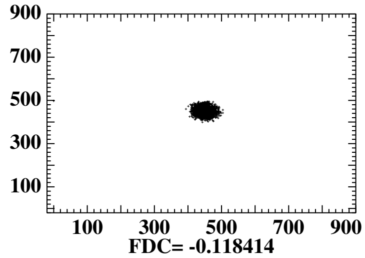
In all what follows, the Evolutionary Algorithm used to both validate the alternative initialization procedures and check the validity of FDC analysis is based on standards: population size 100, linear ranked-based selection, 1-point crossover with probability 0.6, mutation rate with probability of $1/n$ per bit, elitist generational replacement. The only exception is the Gray-coded Baluja function (section 5.4) that uses a (10,100)-ES scheme (10 parents give birth to 100 offspring, among which the best 10 become the parents of next generation), 1-point crossover with probability 0.2, and mutation rate with probability of $4/n$ per bit. No tentative to tune these parameters was performed, as the objective was to compare the initialization procedures, not the absolute GA results per se.

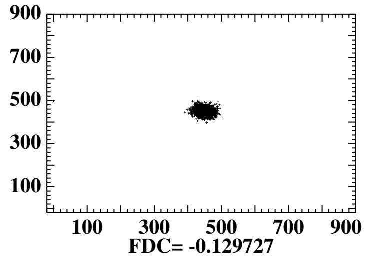


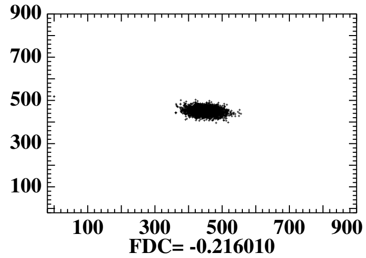




(a) $(800, 100)$ - BU - $\sigma = 0.0100$ (b) $(800, 100)$ - UC - $\sigma = 1.2610^{-5}$ (c) $(800, 100)$ - HB - $\sigma = 1.910^{-5}$

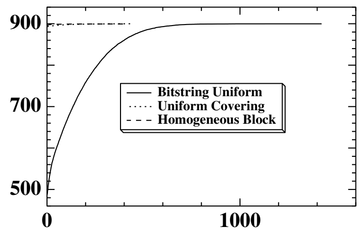


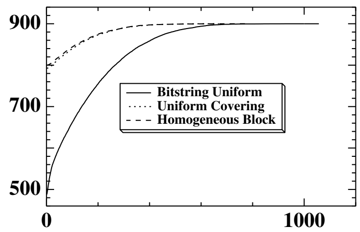


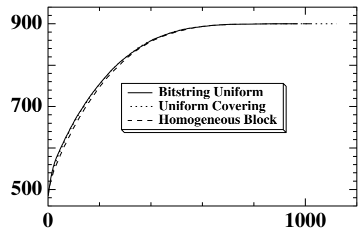


(d) $(450, 450)$ - BU - $\sigma = 0.0169$ (e) $(450, 450)$ - UC - $\sigma = 0.0401$ (f) $(450, 450)$ - HB - $\sigma = 0.091$

Figure 1: *FDC plots for two modified onemax problems and the three initialization procedures.*







(a) $(900 - 0)$-onemax problem (b) $(800, 100)$-onemax problem (c) $(450, 450)$-onemax problem

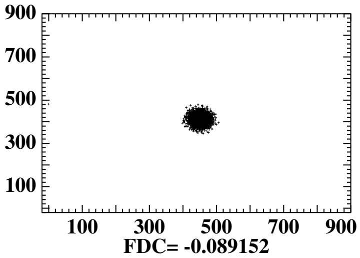
Figure 2: *Average on-line results of GA on three onemax problems: influence of the initialization procedure.*

5.2 Onemax problems

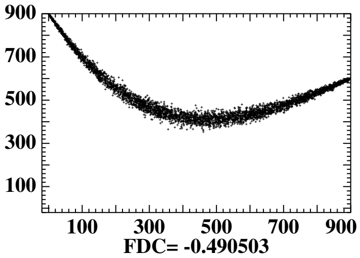
In the well known *onemax* problem, the fitness function is simply the number of 1's in the bitstring. Hence, the fitness is equal to n minus the distance to the global optimum: the FDC coefficient is -1, and the – global – FDC plot is a straight line (Jones & Forrest 1995). As might be expected, both the Uniform Covering and the Homogeneous Block procedures find bitstrings close to the global optimum in the initial population, and hence achieve better convergence.

In order to avoid the bias favoring the two latter procedures, modified versions of the onemax problem were designed: The fitness function is defined as the Hamming distance to a fixed bitstring. That bitstring is chosen to have a certain number O of 1's, randomly placed in the bitstring. Such problem is termed the *(O,n-O)-onemax* problem. Different values for O were experimented, and Figure 1 presents the Local-FDC plots for the $(800, 100)$ and $(450, 450)$ problems, for BU, UC and HB initialization procedures. The standard deviation for 20 different local maxima is given below each plot, witnessing that the stability criterion discussed in section 4.2 is fulfilled.

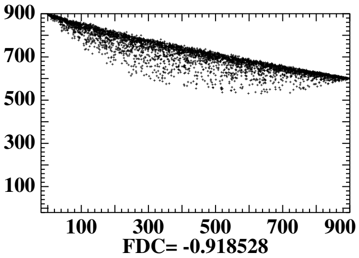
For the $(800, 100)$ problem, there are still high differences between the different procedures: whereas the BU procedure (Figure 1-a) shows a round-shaped cloud with FDC coefficient close to 0, both other procedures (Figures 1-b and 1-c) resemble the ideal onemax global FDC plots. Further, not only the FDC coefficient are close to optimal -1, but also the shapes of the clouds give clear indications: The diversity of the sample is larger for both criteria of fitness and genetic distance for the UC and HB procedures than for the BU procedure: In spite of the optimal bit-wise diversity for all three procedures (see section 3.1), the genetic diversity seems quite smaller for the standard BU procedure when the Hamming distance among genotypes is considered.



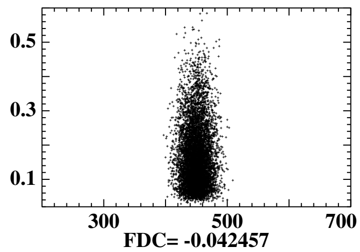
(a) Original ugly - BU



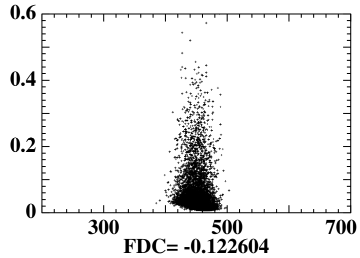
(b) Original ugly - UC



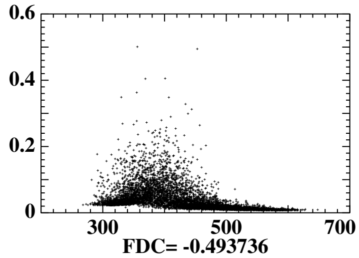
(c) Original ugly - HB



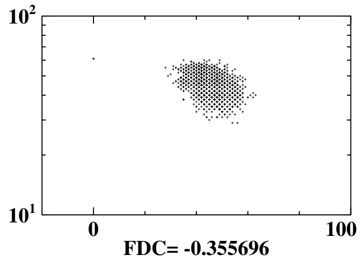
(d) F1g - BU



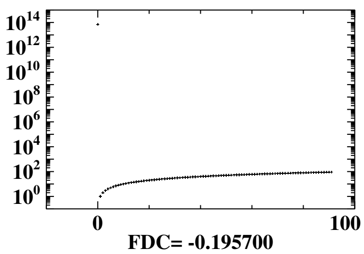
(e) F1g - UC



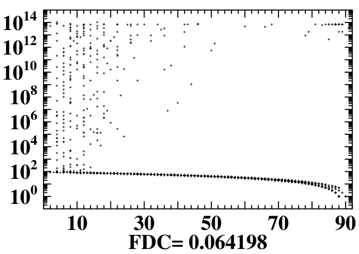
(f) F1g - HB



(g) LongPath - BU

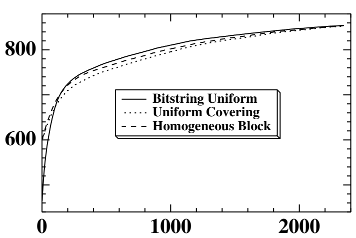


(h) LongPath - UC

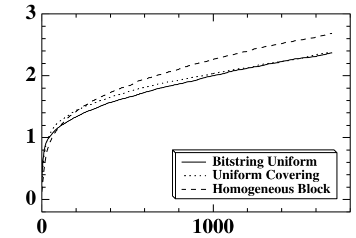


(i) LongPath - HB

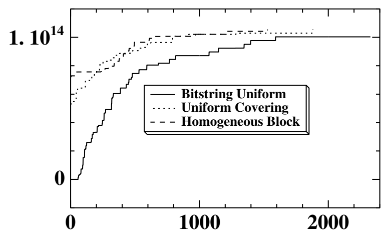
Figure 3: *FDC plots for the Ugly, F1g and LongPath problems using the three initialization procedures.*



(a) Modified ugly problem



(b) F1g problem



(c) LongPath problem

Figure 4: *Average on-line performances of 21 GA runs on Modified ugly, F1g and LongPath problems: influence of the initialization procedure.*

For the $(450, 450)$ problem, BU and UC (Figures 1-d and 1-e) exhibit almost exactly the same plots and coefficients, while the genetic diversity is slightly larger in the HB plot (Figure 1-f) (the "cloud" is wider on the x-axis), with a slightly better coefficient. However, in all cases, these results suggest that no direction will be preferred by the GA (coefficient close to 0, compact round-shaped clouds).

These a priori considerations are verified when it comes to actual runs of the GA: Figure 2 shows the plots of the on-line performance of GA for the original and modified onemax problems. These plots are the usual average best fitness along generations of 21 independent runs. On the $(800, 100)$ problem (Figure 2-b), the T–test with 99 percent confidence states that the BU plot is different from both others up to generation 900. Last, on the extreme case of the $(450, 450)$ problem (Figure 2-c), all three plots are almost indistinguishable: but at least using the biased procedures UC and HB does not degrade the results.

5.3 The Ugly problem

The Ugly problem (Whitley 1991) is defined from an elementary deceptive 3-bit problem (F(x) = 3 if x = 111; F(x) = 2 for x in $0 \star \star$, and F(x) = 0 otherwise). The – deceptive – full problem is composed of 300 concatenated elementary deceptive problems.

The situation resembles here that of the onemax problem (the maximum is the all 1's bitstring). However, the FDC plots on the other hand are much more interesting: whereas the BU-plot has again a round regular shape (Figure 3-a), a clear tendency toward the optimum is shown by the UC-plot (Figure 3-b), and even more by the HB-plot (Figure 3-c). Moreover, both latter plots show a higher range of both fitnesses and distances.

For the original ugly problem, the average performance of GA runs are very similar to those of the $(900, 0)$ onemax problem (see Figure 2-a). Again, a modified version of the ugly problem was designed: A reference bitstring is randomly generated using the BU procedure. The fitness of an individual is then the the Ugly-fitness of its XOR with that reference bitstring. The correlation plots of the three procedures for that modified Ugly problem (not presented here) look very much alike: a round shape with a correlation coefficient close to zero. The average performance of GA runs (Figure 4-a), reflects that situation.

5.4 Gray-coded Baluja function F1

Consider the function of k variables $(x_1, \ldots x_k)$:

$$F_1(\vec{x}) = \frac{100}{10^{-5}+\sum_{i=0}^{k-1}|y_i|}, \text{ with } -2.56 \leq x_i < 2.56,$$
$$y_0 = x_0 \text{ and } y_i = x_i + y_{i-1} \text{ for } i = 1, \ldots, k-1$$

Its maximum is 10^7 at point $(0, \ldots 0)$. As in (Baluja 1995), a binary version of F_1 with 100 variables will be considered, each variable being encoded on 9 bits using Gray coding (resulting in a 900 bits problem).

The correlation plots show similar repartitions for the BU and UC procedures. Both of them present a better fitness range than the HB procedure. This means they are likely to start from with fitter individuals than HB. Hence, the on-line results (Figure 4-b) show that BU and UC achieve better performances than HB during approximately the first 100 generations. However, after a longer period of evolution, HB clearly takes the lead (as confirmed by the T–test with 99 percent confidence), and its off-line results are far better. This phenomenon can be explained by a close look at the FDC results: BU and UC exhibit weaker correlation, witnessed by both the correlation coefficients and the general shape of the FDC plots (Figure 3-d, 3-e). On the other hand, the HB procedure, though starting from lower fitnesses, has a better correlation coefficient and better-shaped FDC plot (Figure 3-f): as individuals improve their fitness, they are likely to get closer to the optimum. On-line computation of FDC coefficient might help to get a better understanding of this situation.

5.5 The LongPath problem

The LongPath problem (Horn & Goldberg 1995) was purposely designed to discourage standard hill-climbers. The fitness landscape is composed of two different regions; the first one resembles a Zeromax landscape, and culminates at $(0, \ldots, 0)$; over that smooth hill, a path of exponential length, starting at $(0, \ldots, 0)$, climbs toward the global optimum by 1-bit steps. The Hamming distance between two non-adjacent points on the path is greater than 2: local hill-climber will have to climb all the way up the path, while global methods like GAs can take short-cuts to reach the optimum quickly. For the 91-bits LongPath used here, the highest value is 210^{14} at point $(1100 \ldots 00)$.

Due to the wide range of fitness values, and to the small size ot the path itself, the FDC plots for the LongPath problem are plotted in Log scale (note also the different scales of the BU plot of Figure 3-g) For the same reason, the FDC coefficients are rather meaningless, as most points of the sample lie outside the path. A simple look at the FDC plots for procedures UC (Figure 3-h) and HB (Figure 3-i) shows that the latter covers a much wider range of fitnesses - while both have the same range of distances. Moreover, on both plots, the initial onemax-like part of the fitness landscape is clearly visible. Indeed, the on-line performances of both procedures are almost identical, as there seems to be enough diversity to allow to find shortcuts of the path (Figure 4-c).

6 Conclusion

Two new binary initialization procedures have been introduced. The *Uniform Covering* procedure ensures that the obtained density of 0's and 1's per bitstring is uniform over $[0, 1]$. The *Homogeneous Block* procedure favors sequences of 0's or 1's in the bitstring, somehow betting on some regularity of the target optimal solution. Note that these procedure respect the bit-wise uniformity among the whole population.

Experiments on different problems of the GA literature have demonstrated the potential usefulness of these procedures: either better on-line results, or/and better off-line results can be obtained. Of course, the diversity of the panel of problems used during these tests is questionable, though these problems are usual benchmarks for GAs. However, the modified onemax and ugly problems have been used to address this issue: even in the worst case, both alternative procedures provide the same quality of results than the standard procedure.

Anyhow, we do not claim that the proposed procedures should be preferred to the standard procedure. Rather, we state that the alternative procedures should be considered together with the usual Bitstring Uniform procedure before starting the optimization of any binary problem.

Further, we have demonstrated that the measure of the – Local – Fitness-Distance Correlation has good predictive properties for the a priori choice of an initialization procedure, regardless of its accuracy as a general predictor of problem difficulty. This comparison of performances of different initialization procedures relies on the relative repartitions of the samples in the fitness $\times$ distance space, considered together with the correlation coefficients. Hence, at the cost of a single run of the evolutionary algorithm (a few thousand function evaluations), FDC coefficients and plots can be obtained, allowing in most cases to choose among different available initialization procedures. Note that this conclusion holds for any search space where more than one initialization procedure can be designed.

References

Baluja, S. 1995. An empirical comparizon of seven iterative and evolutionary function optimization heuristics. Tech. Rep. CMU-CS-95-193, Carnegie Mellon University.

Bramlette, M. F. 1991. Initialization, mutation and selection methods in genetic algorithmes for function optimization. In Belew, R. K., and Booker, L. B., eds., *Proc. of the 4^{th} International Conference on Genetic Algorithms*, 100–108. Morgan Kaufmann.

de Garis, H. 1990. Genetic programming : building artificial nervous systems using genetically programmed neural networks modules. In Porter, R., and Mooney, B., eds., *Proc. of the 7^{th} International Conference on Machine Learning*, 132–139. Morgan Kaufmann.

Goldberg, D. E. 1989. *Genetic algorithms in search, optimization and machine learning.* Addison Wesley.

Grefenstette, J. J. 1987. Incorporating problem specific knowledge in genetic algorithms. In L., D., ed., *Genetic Algorithms and Simulated Annealing*, 42–60. Morgan Kaufmann.

Grefenstette, J. J. 1995. Predictive models using fitness distributions of genetic operators. In Whitley, L. D., and Vose, M. D., eds., *Foundations of Genetic Algorithms 3*, 139–161. Morgan Kaufmann.

Holland, J. H. 1975. *Adaptation in natural and artificial systems.* University of Michigan Press, Ann Arbor.

Horn, J., and Goldberg, D. 1995. Genetic algorithms difficulty and the modality of fitness landscapes. In Whitley, L. D., and Vose, M. D., eds., *Foundations of Genetic Algorithms 3*, 243–269. Morgan Kaufmann.

Iba, H. 1996. Random tree generation for genetic programming. In Voigt, H.-M.; Ebeling, W.; Rechenberg, I.; and Schwefel, H.-P., eds., *Proc. of the 4^{th} Conference on Parallel Problems Solving from Nature*, volume 1141 of *LNCS*, 144–153. Springer Verlag.

Jones, T., and Forrest, S. 1995. Fitness distance correlation as a measure of problem difficulty for genetic algorithms. In Eshelman, L. J., ed., *Proc. of the 6^{th} International Conference on Genetic Algorithms*, 184–192. Morgan Kaufmann.

Kallel, L., and Schoenauer, M. 1996. Fitness distance correlation for variable length representations. Technical Report 363, CMAP, Ecole Polytechnique.

Kargupta, H. 1995. Signal-to-noise, crosstalk, and long range problem difficulty in genetic algorithms. In Eshelman, L. J., ed., *Proc. of the 6^{th} International Conference on Genetic Algorithms*, 193–200. Morgan Kaufmann.

Knuth, D. E. 1981. *The art of Computer Programming*, volume 2: Seminumerical Algorithms. Addison Wesley.

Maresky, J.; Davidor, Y.; Gitler, D.; G.Aharoni; and Barak, A. 1995. Selectively destructive re-start. In Eshelman, L. J., ed., *Proc. of the 6^{th} International Conference on Genetic Algorithms*, 144–150. Morgan Kaufmann.

Radcliffe, N. J. 1991. Equivalence class analysis of genetic algorithms. *Complex Systems* 5:183–20.

Rudolph, G. 1997. *Convergence Properties of Evolutionary Algorithms.* Hamburg: Kovac.

Schaffer, J. D.; Caruana, R. A.; Eshelman, L.; and Das, R. 1989. A study of control parameters affecting on-line performance of genetic algorithms for function optimization. In Schaffer, J. D., ed., *Proc. of the 3^{rd} International Conference on Genetic Algorithms*, 51–60. Morgan Kaufmann.

Schoenauer, M., and Xanthakis, S. 1993. Constrained GA optimization. In Forrest, S., ed., *Proc. of the 5^{th} International Conference on Genetic Algorithms*, 573–580. Morgan Kaufmann.

Schoenauer, M.; Jouve, F.; and Kallel, L. 1997. Identification of mechanical inclusions. In Dasgupta, D., and Michalewicz, Z., eds., *Evolutionary Computation in Engeneering*, 477–494. Springer Verlag.

Schoenauer, M. 1996. Shape representations and evolution schemes. In Fogel, L. J.; Angeline, P. J.; and Bäck, T., eds., *Proc. of the 5^{th} Annual Conference on Evolutionary Programming*, 121,129. MIT Press.

Schultz, A., and Grefenstette, J. 1990. Improving tactical plans with genetic algorithms. In *Proc. of IEEE Conference on Tools for AI*, 328–334. Morgan Kaufmann.

Whitley, D. 1991. Fundamental principles of deception in genetic search. In Rawlins, G. J. E., ed., *Foundations of Genetic Algorithms.* Morgan Kaufmann.