Designing Efficient Evolutionary Algorithms for Cluster Optimization: A Study on Locality

Francisco B. Pereira[1,3], Jorge M. C. Marques[2], Tiago Leitão[3], and Jorge Tavares[3]

[1] Instituto Superior de Engenharia de Coimbra, Quinta da Nora, 3030-199 Coimbra, Portugal xico@dei.uc.pt
[2] Departamento de Química, Universidade de Coimbra, 3004-535 Coimbra, Portugal qtmarque@ci.uc.pt
[3] Centro de Informática e Sistemas da Universidade de Coimbra, 3030 Coimbra, Portugal {tleitao,jast}@dei.uc.pt

Summary. Cluster geometry optimization is an important problem from the Chemistry area. Hybrid approaches combining evolutionary algorithms and gradient-driven local search methods are one of the most efficient techniques to perform a meaningful exploration of the solution space to ensure the discovery of low energy geometries. Here we perform a comprehensive study on the locality properties of this approach to gain insight on the algorithm's strengths and weaknesses. The analysis is accomplished through the application of several static measures to randomly generated solutions in order to establish the main properties of an extended set of mutation and crossover operators. Locality analysis is complemented with additional results obtained from optimization runs. The combination of the outcomes allows us to propose a robust hybrid algorithm that is able to quickly discover the arrangement of the cluster's particles that correspond to optimal or near-optimal solutions.

Key words: Cluster Geometry Optimization, Hybrid Evolutionary Algorithms, Locality, Potential Energy

1 Introduction

A cluster is an aggregate of between a few and millions of atoms or molecules, which may present distinct physical properties from those of a single molecule or bulk matter. The interactions among those atoms (or molecules) may be described by a multi-dimensional function, designated as Potential Energy Surface (PES), whose knowledge is mandatory in the theoretical study of the properties of a given chemical system. The arrangement of particles corresponding to the lowest potential energy (*i.e.*, the global minimum on the PES) is an important piece of information to understand the properties of real

clusters. Usually, for systems with many particles (such as clusters), the PES is approximately written in an analytical form as a sum of all pair-potentials (*i.e.*, functions that depend on the distance connecting each pair of atoms or molecules). Due to their simplicity, both Lennard-Jones [1, 2] and Morse [3] potentials are among the mostly applied as pair-wise models in the study of clusters. In particular, Morse functions may be used to describe either long-range interactions, such as in the alkali metal clusters, or the short-range potentials arising between, *e.g.*, C_{60} molecules. From the point of view of global optimization, Morse clusters (especially the short-range ones) are considered to be more challenging than those described by the Lennard-Jones potential [4, 5]. Indeed, short ranged Morse clusters tend to present a rough energy landscape due to the great number of local minima and their PESs are more likely to have a multiple-funnel topography [4].

Since the early 1990's Evolutionary Algorithms (EAs) have been increasingly applied to several global optimization problems from the Chemistry/Biochemistry area. Cluster geometry optimization is a particular example of one of these problems [5, 6, 7, 8, 9, 10, 11, 12]. Nearly all the existing approaches rely on hybrid algorithms combining EAs with local search methods that use first order derivative information to guide search into the nearest local optimum. State of the art EAs adopt real-valued representations codifying the Cartesian coordinates of the atoms that compose the cluster [12]. Performance of evolutionary methods can be dramatically increased if local optimization is used to improve each individual that is generated. Hybrid approaches for this problem were first proposed by Deaven and Ho [6] and, since then, have been used in nearly all cluster optimization situations. Typically, local methods perform a gradient-driven local minimization of the cluster potential, allowing the hybrid algorithm to efficiently discover the nearest local optimum.

Locality is an essential requirement to ensure the efficiency of search and has been widely studied by the evolutionary computation community [13, 14, 15, 16, 17, 18]. Locality indicates that small variations in the genotype space imply small variations in the phenotype one. A locally strong search algorithm is able to efficiently explore the neighborhood of the current solutions. When this condition is not satisfied, the exploration performed by the EA is inefficient and tends to resemble random search.

The goal of our analysis is to perform an empirical study on the locality properties of the hybrid algorithm that is usually adopted for cluster optimization. The analysis adopts the framework proposed by Raidl and Gottlieb [15]. In this model a set of inexpensive static measures is used to characterize the interactions between representation and genetic operators and access how do they influence the interplay among the genotype/phenotype space. We extend this framework to deal with an optimization situation where the joint efforts of an EA and a gradient driven local method are combined during the exploration of the search space. The study considers a broad set of genetic

operators, suitable for a real valued representation. Furthermore, two distance measures are defined and used: fitness based and structural distance.

Mutation is the most frequent operator considered in locality studies. In a previous paper we already presented a detailed analysis concerning its properties when applied in this evolutionary framework [19]. Here, we briefly review the main conclusions and extend the work to consider crossover. We believe that to obtain a complete characterization of the hybrid EA search competence, crossover must also be taken into account. In what concerns this operator, locality should measure its ability to generate descendants by preserving and combining useful features of both parents.

Results allow us to gain insight about the degree of locality induced by genetic operators. In what concerns mutation, results establish a clear hierarchy in the locality strength of different types of mutation. As for crossover, the analysis shows that one of the operators is able to maintain/promote diversity, even if similar individuals compose the population. The other two crossover operators considered in this study require mutation to maintain diversity. To the best of our knowledge, this is the first time that a comprehensive locality analysis is used to study hybrid algorithms for cluster geometry optimization. Results help to provide a better understanding of the role played by each one of the components of the algorithm, which may be important for future applications of EAs to similar problems from the Chemistry area. For the sake of completeness, the empirical locality study is complemented with additional results obtained from real optimization experiments. The outcomes confirm the main conclusions of the static analysis.

The structure of the chapter is the following: in Sect. 2 we briefly describe Morse clusters. In Sect. 3 we present the main components of the hybrid algorithm used in the experiments. A brief report of some optimization results is presented in Sect. 4. Section 5 comprises a detailed analysis on the locality properties of the algorithm. In Sect. 6 we present the outcomes of the optimization of a large cluster to confirm the results of the locality analysis and, finally, Sect. 7 gathers the main conclusions.

2 Morse Clusters

Morse clusters are considered a benchmark for testing the performance of new methods for cluster structure optimization. The energy of such a cluster is represented by the N-particle pair-wise additive potential [3] defined as

$$V_{Morse} = \epsilon \sum_{i}^{N-1} \sum_{j>i}^{N} \{\exp[-2\beta(r_{ij} - r_0)] - 2\exp[-\beta(r_{ij} - r_0)]\} \qquad (1)$$

where the variable r_{ij} is the distance between atoms i and j in the cluster structure. The bond dissociation energy ϵ, the equilibrium bond length r_0 and

the range exponent of the potential β are parameters defined for each individual pair-wise Morse interaction. Usually, these are assumed to be constant for all interactions in a cluster formed by only one type of atoms. The potential of (1) is a scaled version [20] of the Morse function with non-atom-specific interactions, where ϵ and r_0 have been both set to 1 and β has been fixed at 14, which corresponds to a short-range interaction. Global optimization is particularly challenging for short-range Morse clusters, since they have many local minima and a "noisy" PES [4]. This simplified potential has already been studied by other authors [5, 9, 11, 20], and the minima are well established for many values of N [21].

The application of local minimization methods, as described below, requires the specification of the analytical gradient of the function to be optimized. In Cartesian coordinates, the generic element n of the gradient of the Morse cluster potential may be given by

$$g_n = -2\beta\epsilon \sum_{i \neq n}^{N} \left(\frac{x_{ni}}{r_{ni}} \right) \{\exp[-2\beta(r_{ni} - r_0)] - \exp[-\beta(r_{ni} - r_0)]\} \qquad (2)$$

where $x_{ni} = x_n - x_i$. Similar expressions apply for the y and z directions.

3 EAs for Morse Clusters Optimization

EAs have been used since 1993 for cluster geometry optimization. A comprehensive review of these efforts, including an outline of state-of-the art applications, can be found at [8]. In what concerns the application of EAs to Morse clusters, the most important works are from Johnston and collaborators [5, 11]. In our analysis we adopt an experimental model similar to the one used by these researchers. Its main components have been proposed and evaluated by different teams [5, 6, 7, 12].

3.1 Chromosome Representation and Evaluation

An individual must specify the location of the atoms that compose the cluster. For aggregates with N atoms, a solution is composed by $3 \times N$ real values specifying the Cartesian coordinates of each one of the particles. The scheme presented in Fig. 1 illustrates the chromosome format. Zeiri proposed this representation in 1995 [12] and, since then, it has become the most widely used in this context [5, 7]. The coordinate values range between 0 and λ. We set λ to $N^{1/3}$, as this interval ensures that the aggregate volume scales correctly with N [11].

There is another parameter δ that specifies the minimum distance that must exist between atoms belonging to the same cluster. It is useful to prevent the generation of aggregates with particles that are excessively close to each

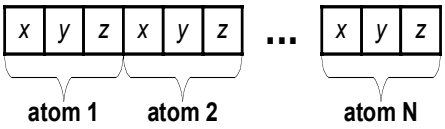

Fig. 1. Structure of a chromosome

other. This avoids possible numerical problems (if two particles are too close, then the pair-wise potential will tend to infinity) and reduces the size of the search space. This parameter is used in the generation of the initial population and during the application of genetic operators.

To assign fitness to an individual we just have to calculate its potential energy by using (1).

3.2 Population Model and Genetic Operators

A generational model is adopted and the standard set of variation operators is used. In what concerns crossover, three different operators are analyzed in this research study: uniform, cut and splice and generalized cut and splice crossover. The purpose of all of them is to exchange sub-clusters between parents when generating descendants. In this context, a sub-cluster is defined as a subset of the atoms that compose the cluster. In uniform crossover, the atoms that will compose the offspring are randomly selected from those of the parents. More specifically, the parent chromosomes are scanned from left to right (atoms 1 through N) and, in each position, the child inherits the atom from one of the parents with equal probability. When combining particles from two parents to create a descendant, uniform crossover does not consider the spatial distribution of the atoms. It just cares about the ordering of atoms in the chromosome, which is not related to their positioning in the 3D space.

Cut and splice crossover (C&S crossover), proposed by Deaven and Ho in 1995 [6], was specifically designed for cluster geometry optimization. Unlike generic operators, such as uniform crossover, C&S is sensitive to the semantic properties of the structures being manipulated and therefore it is able to do a more suitable combination of the parents' features. Since its proposal, it became widely used and several authors confirm that it enhances the performance of the algorithm [8, 11]. When generating two descendants $D1$ and $D2$ from parents $P1$ and $P2$, C&S determines the sub-clusters to be exchanged in the following way:

1. Apply random rotations to $P1$ and $P2$;
2. Define a random horizontal cutting plane (parallel to the xy plane) for $P1$. This plane splits $P1$ in two complementary parts (X atoms below the plane and $N - X$ atoms above it);

3. Define a horizontal cutting plane (parallel to the xy plane) for $P2$, in such a way that X atoms stay below the plane and $N - X$ are above it;
4. Generate $D1$ and $D2$ by combining complementary parts of each one of the parents.

Special precautions are taken when merging sections from different parents to ensure that the distance between two atoms is never smaller than δ.

Unlike uniform crossover, C&S ensures that the contribution of each parent is formed by a set of atoms that are closed together (they are above or below a randomly determined plane). The sub-clusters will tend to have a low-energy[1], therefore increasing the likelihood of combining useful building blocks that compose good quality solutions. In addition to these two existing operators, here we propose and analyze a generalization of C&S. This will help us to perform a more detailed study concerning the locality properties of crossover operators used in evolutionary cluster optimization. The new operator, which we will identify as generalized cut and splice (GenC&S), acts in a way that resembles standard C&S crossover. The most relevant difference is related to the way it determines the sub-clusters to be exchanged. With GenC&S, subsets of atoms that are close together in the parent clusters will form the building blocks used to create descendants. The constraint that exists in the original cut and splice operator (atoms above/below the plane) is removed and Euclidian distance is the only criterium used to select atoms.

More specifically, GenC&S creates a descendant $D1$ from parents $P1$ and $P2$ in the following way (the other descendant $D2$ is created swapping the role played by the parents):

1. Select a random atom CP from $P1$;
2. Select a random number $X \in [1,\, N-2]$, where N is the number of atoms that compose the cluster;
3. From $P1$, copy CP and the X atoms closer to it, to $D1$;
4. Select $N - (X + 1)$ atoms from $P2$ to complete $D1$. Give preference to atoms that, in the 3D space, are closer to the original location of CP. Skip atoms that are too close (*i.e.*, at a distance smaller than δ) to particles already belonging to $D1$.

Depending on the atom distribution, in a small number of situations it might be impossible to select enough particles from $P2$ to complete $D1$. This can happen because too many atoms are skipped due to the distance constraint. If this situation occurs, $D1$ is completed with atoms randomly placed.

There is another difference between C&S and GenC&S: in the second operator no random rotations are applied to the parents before the genetic material is mixed. This action is not necessary in the generalized version because we removed the constraint that forces the cutting plane defining the sub-clusters to be parallel to the xy plane and therefore there is not a bias associated with this operator.

[1] Potential energy is directly related to the distance between pairs of atoms.

Two mutation operators were tested in this work: Sigma mutation and Flip mutation. We consider that mutation is performed on atoms, i.e., when applied it modifies the value of the three coordinates that specify the position of a particle in the 3D space. The first operator is an evolutionary strategy (ES) like mutation and acts in the following way: when undergoing mutation, the new value v_{new} for each one of the three coordinates of an atom (x, y, z) is obtained from the old value v_{old} through the expression:

$$v_{new} = v_{old} + \sigma N(0, 1) \tag{3}$$

where $N(0, 1)$ represents a random value sampled from a standard Normal distribution and σ is a parameter from the algorithm. The new value must be between 0 and λ.

Flip mutation works in the following way: when applied to an atom, it assigns new random values to each one of its coordinates, *i.e.*, it moves this atom to a random location.

3.3 Local Optimization

Local optimization is performed with the Broyden-Fletcher-Goldfarb-Shanno limited memory quasi-Newton method (L-BFGS) of Liu and Nocedal [22, 23]. L-BFGS is a powerful optimization technique that aims to combine the modest storage and computational requirements of conjugate gradient methods with the superlinear convergence exhibited by full memory quasi-Newton methods (when sufficiently close to a solution, Newton methods are quadratically convergent). In this limited memory algorithm, the function to be minimized and its gradient must be supplied, but knowledge about the corresponding Hessian matrix is not required *a priori*.

L-BFGS is applied to every generated individual. During local search, the maximum number of iterations that can be performed is specified by a parameter of the algorithm, the Local Search Length (LSL). However, L-BFGS stops as soon as it finds a local optimum, so the effective number of iterations can be smaller than the value specified by LSL.

4 Optimization Results

The main goal of the research reported here is to study the locality of different genetic operators. This will be carried out in the next section. Nevertheless, to establish an appropriate background, we first present some experimental results.

Aggregates ranging from 19 to 50 atoms compose the standard instances used when Morse clusters are adopted as a benchmark for accessing the efficiency of evolutionary algorithms. The original research conducted by Johnston *et al.* [11] revealed that the hybrid algorithm is efficient and reliable, as it

was able to find nearly all known best solutions. The only exception was the cluster with 30 atoms, where the current putative optimum was only reported in a subsequent paper by the same authors [5]. The algorithm used in the experiments relied on C&S crossover and flip mutation as genetic operators. Other details concerning the optimization can be found in [5, 11].

In 2006, we developed a hybrid algorithm to be used in the locality analysis. In order to confirm its search competence, we repeated the experiments of searching for the optimal geometry of Morse clusters ranging from 19 to 50 atoms. C&S crossover and the two mutation operators previously described were used in the tests. Achieved results confirmed the efficiency of the hybrid approach, as it was able to find all known best solutions. Regardless of this situation, a more detailed analysis of the outcomes revealed that there are some differences in what concerns results achieved by different mutation operators. Whilst experiments performed with Sigma mutation achieve good results for all instances, tests done with Flip mutation reveal a less consistent behavior. For small clusters (up to 30 atoms), the achieved results are analogous to the ones obtained by Sigma mutation. As the clusters grow in size, the algorithm shows signs of poor scalability and its performance starts to deteriorate. For clusters with more than 36 atoms, in most times it fails to find the optimum. A detailed description and analysis of the results and a complete specification of the parameter settings used in the experiments may be found in [19].

5 Locality Analysis

The cluster with 50 atoms (the largest instance that was considered in the previous optimization experiments) was selected to perform all the tests concerning the locality properties of the different genetic operators. When appropriate, the empirical analysis is complemented with experimental results.

5.1 Related Work

Many approaches were proposed to estimate the behavior of EAs when applied to a given problem. Some of these techniques adopt measures that are, to some extent, similar to the locality property adopted in this work. In this section we highlight the most relevant ones.

The concept of fitness landscapes, originally proposed by Wright [24], establishes a connection between solution candidates and their fitness values and it has been widely used to predict EAs performance. Several measures for fitness landscapes were defined for this task. Jones and Forrest proposed fitness distance correlation as a way to determine the relation between fitness value and distance to the optimum [25]. If fitness values increase as distance to the optimum decreases, then search is expected to be easy for an EA [26].

An alternative way to analyze the fitness landscape is to determine its ruggedness. Some autocorrelation measures help to determine how rugged a

landscape is. Weinberger [27] proposed the adoption of autocorrelation functions to measure the correlation of all points in the search space at a given distance. Another possibility to investigate the correlation structure of a landscape is to perform some random walks. The value obtained with the random walk correlation function can then be used to determine the correlation length, a value that directly reflects the ruggedness of the landscape [24]. In general, smoother landscapes are highly correlated, making the search for an EA easier. More rugged landscapes are harder to explore in a meaningful way.

Sendhoff et al. studied the conditions for strong causality on EAs [18]. A search process is said to be locally strongly causal if small variations in the genotype space imply small variations in the phenotype space. In the abovementioned work, variations in genotypes are caused by mutation (crossover is not applied). Fitness variation is used to access distances in the phenotype space. A probabilistic causality condition is proposed and studied in two situations: optimization of a continuous mathematical function and optimization of the structure of a neural network. They conclude that strong causality is essential, as it allows for controlled small steps in the phenotype space that are provoked by small steps in the genotype space.

The empirical framework to study locality that we adopt in our research was proposed by Raidl and Gottlieb [15]. The model is useful to study how the adopted representation and the genetic operators are related and how does this interplay influence the performance of the search algorithm. The analysis is based on static measures applied to randomly generated individuals that help to quantify the distance between solutions in the search space and how it is linked to the similarity among corresponding phenotypes. This model allows the calculation of three features, which are essential for good performance: locality, heritability and heuristic bias. Locality was already defined in the introduction. Heritability refers to the ability of crossover operators to create children that combine meaningful features of both parents. Heuristic bias concerns the genotype-phenotype mapping function. Some functions that favor the mapping towards phenotypes with higher fitness might help to increase performance. This effect is called heuristic bias. The authors refer that these properties can be studied either in a static fashion or be dynamically analyzed during actual optimization runs. They also claim that the achieved results provide a reliable basis for accessing the efficiency of representations and genetic operators. In the above mentioned work, this framework is used compare different representations for the multidimensional knapsack problem.

5.2 Definitions

When performing studies with an evolutionary framework it is usual to consider two spaces: the genotype space Φ_g and the phenotype space Φ_p [28]. Genetic operators work on Φ_g, whereas the fitness function f is applied to solutions from Φ_p: f: $\Phi_p \rightarrow \mathbb{R}$. A direct representation is adopted in this paper. Since there is not a maturation or decoder function, genetic operators

are directly applied to phenotypes. This way, it is not necessary to perform an explicit distinction between the two spaces and, from now on, we will refer to individuals or phenotypes to designate points from the search space.

To calculate the similarity between two individuals from Φ_p, a phenotypic distance has to be defined. This measure captures the semantic difference between two solutions and is directly related to the problem being solved. We determine phenotypic distance in the two following ways.

Fitness based distance

Determining the fitness distance between two phenotypes A, B is straightforward:

$$d_{fit}(A, B) = |f(A) - f(B)| \tag{4}$$

In cluster optimization, it calculates the difference between the potential energy of the two solutions.

Structural distance

According to (1), the basic features that influence the quality of a N-atom cluster are the $N \times (N-1)/2$ interactions occurring between particles forming the aggregate. The interaction between atoms i and j depends only on the distance r_{ij} between them. We implement a simple method to approximate the structural shape of a cluster. First, all the $N \times (N-1)/2$ distances between atoms are calculated. Then, they are separated into several sets according to its values. We consider 10 sets S_i. The limits for each S_i, $i = 1, \ldots, 10$, are defined as follows:

$$\left[\frac{i-1}{10} \times \mu, \frac{i}{10} \times \mu\right[, i = 1, \ldots, 10 \tag{5}$$

where μ is the maximum distance between two atoms. Considering the parameter λ, μ is equal to $\sqrt{3\lambda^2}$. Structural distance captures the dissimilarity between two clusters A and B in what concerns the distances between all pairs of atoms. It is measured in the following way:

$$d_{struct}(A, B) = \frac{1}{10} \sum_{i=1}^{10} |\#S_i(A) - \#S_i(B)| \tag{6}$$

where $\#S_i(A)$ (likewise, $\#S_i(B)$) is the cardinality of subset S_i for cluster A (likewise, for cluster B).

5.3 Mutation Innovation

To analyze the effect of mutation on locality we adopt the innovation measure proposed by Raidl and Gottlieb [13]. The distance between the individuals

involved in a mutation is used to predict the effect of the application this operator. Let X be a solution and X^m the result of applying mutation to X. The mutation innovation MI is measured as follows:

$$MI = dist(X, X^m) \tag{7}$$

Distance can be calculated using either fitness based or structural distance. MI illustrates how much innovation the mutation operator introduces, i.e., it aims to determine how much this operator modifies the semantic properties of an individual. Locality is directly related to this measure. The application of a locally strong operator implies a small modification in the phenotype of an individual (i.e., there will be a small phenotypic distance between the two involved solutions). Conversely, operators with weak locality allow large jumps in the search space, complicating the task of the search algorithm. To determine the MI, 1000 random individuals were generated and then, a sequence of mutations was applied to each one of them. In each one of the 1000 mutation series, distance is measured between the original individual and the solution created after $k \in \{1, 2, 3, 4, 5, 10, 25, 50, 100\}$ successive mutation steps. In conformity to the adopted optimization framework, local search is considered as part of this genetic operator, i.e., L-BFGS is applied after each mutation and distance is measured using the solution that results from this operation.

We will study the locality properties of two mutation operators: Sigma, Flip. In what concerns the first operator, three values for σ are tested: $\{0.01 \times \lambda, 0.1 \times \lambda, 0.25 \times \lambda\}$[2]. We expect that the variation in the value of σ will provide insight on the effect of this parameter on the performance of the algorithm.

Fitness Based Distance

To simplify the analysis, distances between the original solution and the successive mutants are grouped in different sets. Given a d_{fit} fitness distance between two solutions, set $\mathbf{G_i}$ to which d_{fit} is assigned, is determined in the following way: $\{\mathbf{G0} : 0 \leq d_{fit} < 1$; $\mathbf{G1} : 1 \leq d_{fit} < 5$; $\mathbf{G2} : 5 \leq d_{fit} < 10$; $\mathbf{G3} : 10 \leq d_{fit} < 20$; $\mathbf{G4} : 20 \leq d_{fit} < 30$; $\mathbf{G5} : 30 \leq d_{fit} < 50$; $\mathbf{G6} : 50 \leq d_{fit} < 100$; $\mathbf{G7} : 100 \leq d_{fit} < 250$; $\mathbf{G8} : 250 \leq d_{fit} < 500$; $\mathbf{G9} : 500 \leq d_{fit}\}$.

The specific values that were selected to determine intervals are arbitrary. The relevant information to obtain here is the distribution of the fitness distances through the sets. Situations where values tend to be assigned to higher order sets (i.e., large variations), suggest that the locality is low. In Figs. 2

[2] The σ value used in Sigma mutation is proportional to λ to ensure that its effect is comparable for clusters of different size. Nevertheless, in the text we will adopt the simplified notation Sigma0.01 instead of Sigma0.01$\times\lambda$ (alike for other values of σ).

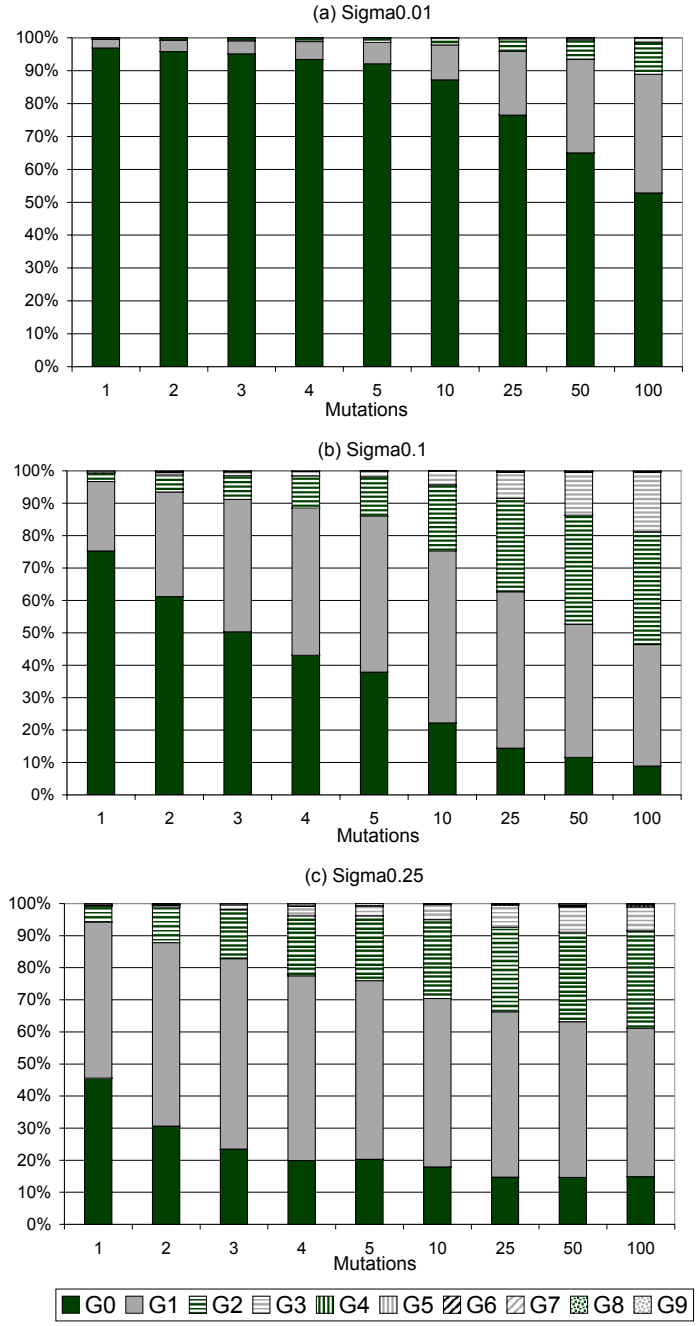

Fig. 2. Distribution of fitness distances between the original solutions and the mutants iteratively generated for Sigma mutation: **(a)** Sigma0.01; **(b)** Sigma0.1; **(c)** Sigma0.25

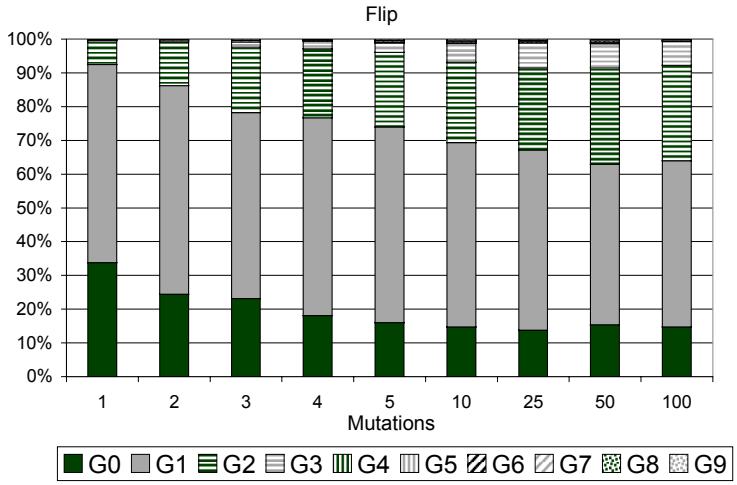

Fig. 3. Distribution of fitness distances between the original solutions and the mutants iteratively generated for Flip mutation

and 3 we present, for the considered number of mutations, the distances between the original solutions and the mutants iteratively generated. The three charts from Fig. 2 are from experiments performed with Sigma, whereas the chart from Fig. 3 gathers results obtained with Flip mutation. In each column (corresponding to a given mutation step), we show the distribution of the 1000 distances for the 10 $\mathbf{G_i}$ sets. It is clear that there are important differences in what concerns the locality of mutation. Experiments combining Sigma0.01 with L-BFGS exhibit the highest locality and, even after 100 steps, nearly all fitness distances belong to sets **G0**-**G2** (more than 50% are in cluster **G0**). This shows that there is still a clear relation, maybe even excessive, to the departure point. On the contrary, experiments with Sigma0.25 and Flip mutation evidence lower locality. The modification they induce is substantial (large displacement of one atom strongly modifies the fitness of the solution) and local search might not be able to do an appropriate repair. Sigma0.1 is between these two scenarios. In the beginning it shows signs of reasonable locality, but after some steps it approaches the distribution exhibited by Sigma0.25 and Flip. This is a behavior that is more in accordance to what one should expect from a mutation operator. Individuals that are just a few steps away should have similar phenotypic properties. Moreover, distance should gradually increase, as more mutations are applied to one of them.

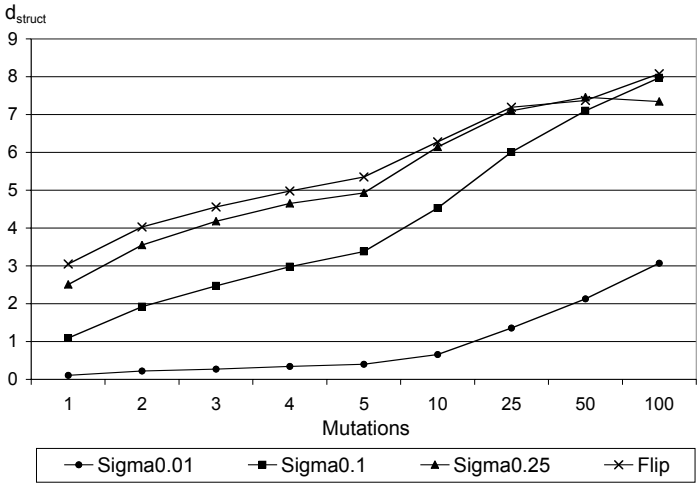

Fig. 4. Average structural distances between the original solution and the mutants iteratively generated

Structural Distance

Analysis of MI with structural distance confirms the conclusions from the previous section. In Fig. 4 we show the structural distance for the same operators and settings. Once again, it is clear that Sigma0.01 has a high locality, Sigma0.25 and Flip have both low locality and Sigma0.1 is between the two extremes. Results with both phenotypic distances reveal that, as σ increases, the effect on the locality of Sigma mutation tends approach that of Flip mutation.

5.4 Crossover Innovation

Crossover innovation measures the ability of this operator to create descendants that are different from their parents. Let C be a child resulting from the application of crossover to parents $P1$ and $P2$. Following the definition proposed by Raidl and Gottlieb [15], crossover innovation CI can be measured as follows:

$$CI = min\{dist(C, P1), dist(C, P2)\} \tag{8}$$

According to (8), CI measures the phenotypic distance between a child and its phenotypically closer parent. In general, we expect CI to be directly related to the distance that exists between parents involved in crossover. Similar parents tend to create descendants that are also close to both of them. On the contrary, dissimilar parents tend to originate larger crossover innovations.

Nevertheless, under the same circumstances (i.e., when applied to the same pair of individuals), different crossover operators might induce distinct levels of innovation. This disparity reflects diverse attitudes on how the genetic material is combined. When exploring the search space, it is important to rely on crossover operators that maintain a moderately high value of CI. This will help to preserve population diversity and ensure that an appropriate exploration of the space is performed. Anyway, it is also important that CI is not too high because this might prevent the preservation and combination of useful features that are inherited from the parents. It is a well-known fact that mixing properties from the parents in a meaningful way is one of the most important roles of crossover [29].

The CI of the different crossover operators was empirically analyzed. To study how the parental distance affects this measure, we adopted the following procedure: parent $P1$ was randomly generated and then kept unchanged throughout the experiments whilst parent $P2$ was obtained from $P1$ through the application of a sequence of mutations. In the experiments performed, we measured CI after $k \in \{1, 2, 3, 4, 5, 10, 25, 50, 100\}$ successive mutation steps. Local optimization is applied to the original solution and also after each mutation. Sigma0.1 was the operator chosen to generate the sequence of mutated individuals that act as parent $P2$. As it can be confirmed from Figs. 2 and 4, in the beginning of the sequence (i.e., when the number of mutations is small), the parents will be similar. When the number of successive mutations increases, difference between parents tends to increase steadily ($P1$ remains unchanged whilst $P2$ accumulates mutations). In conformity with the adopted optimization framework, the child that is generated is locally improved before CI is determined.

CI was used to study the locality properties of three crossover operators: {Uniform, C&S, GenC&S}. The described procedure was repeated 1000 times for each one of the operators. Results obtained with the two distance measures will be analyzed separately.

Fitness Based Distance

The 1000 fitness distances were grouped in the same 10 sets **G0**-**G9** previously described. In the charts from Fig. 5 we present, for the three crossover operators, the distances between a child and its phenotypically closer parent. Each column corresponds to a given distance between the two parents: in the first column from the left, parents are just one mutation away, whilst in the last one they are 100 mutations away. In every one of these columns we show the distribution of the 1000 distances for the 10 $\mathbf{G_i}$ sets.

It is clear from the charts that the combined application of crossover and subsequent local search establishes a process with fairly high locality. Even with parents that are 100 mutations away, the child maintains a clear relation, in what concerns the potential energy of the cluster, with at least one of its

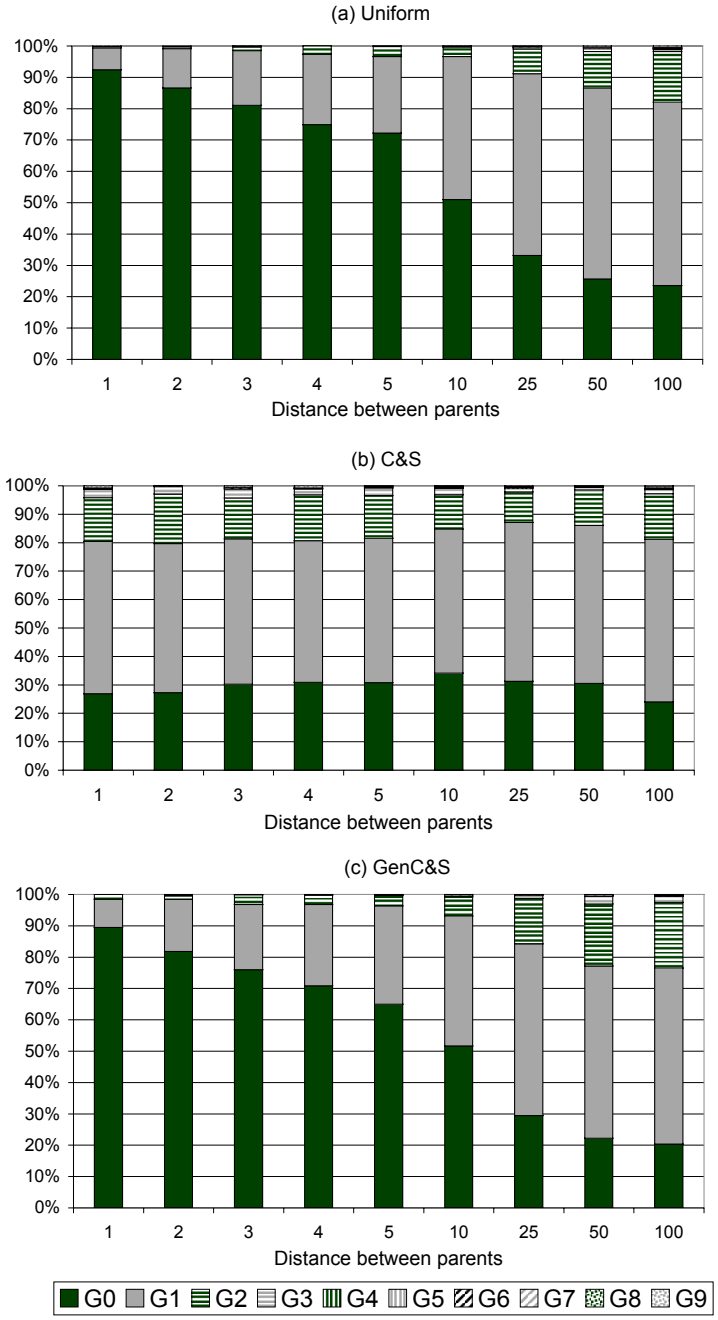

Fig. 5. Distribution of fitness distances between the child and the phenotypically closer parent: **(a)** Uniform; **(b)** C&S; **(c)** GenC&S

parents. Nearly all fitness distances lie in sets **G0**, **G1** and **G2**, meaning that the variation in potential energy does not exceed 10.

Another relevant outcome is the noteworthy difference between the results achieved with C&S and the results obtained with the other two crossover operators. C&S seems to be insensitive to the difference that exists between parents. The distribution of the fitness distances is similar, whether the parents are almost identical or have large dissimilarities. This result suggests that the diversity level of the population is irrelevant when C&S is applied. The justification for the results might be related to the way C&S acts. Before cutting parents, it applies a random rotation to each one of them. As the rotations are independent, even if the two parents are identical the descendants might be distinct. This is an unusual behavior because nearly all crossover operators are unable to introduce any novelty into the population when they are applied to an identical pair of solutions. On the contrary, C&S is capable of adding diversity to the population. As a consequence, existing diversity might not be as relevant as it is in other situations. Finally, this outcome also suggests that mutation might be less important in experiments with C&S than in experiments that rely on other crossover operators.

Charts displaying the CI distribution of uniform and GenC&S operators present a pattern that is more in compliance with the expected behavior of crossover. When parents are similar, the innovation is small. As the distance between parents increases, average CI also increases.

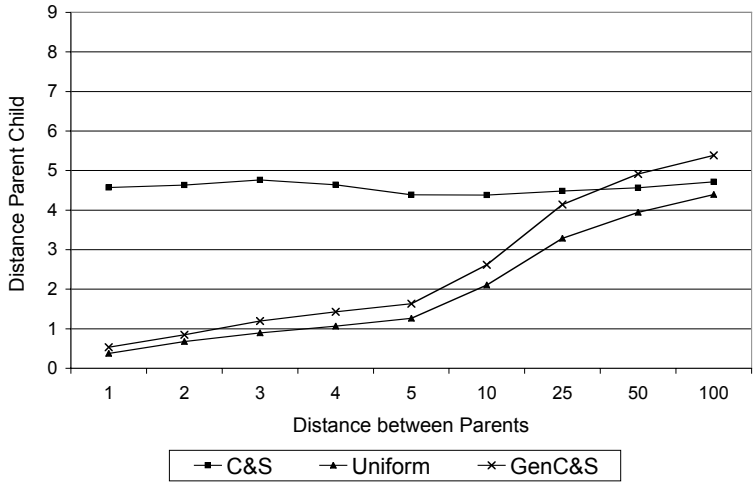

Fig. 6. Average structural distances between the child and the phenotypically closer parent

Structural Distance

Results achieved with structural distance are presented in the chart from Fig. 6. They are in agreement with the information provided by fitness based distance. Whilst C&S crossover is insensitive to the distance that exists between parents, both uniform and GenC&S tend to generate more innovative children as this distance increases.

5.5 Additional Tests

To complement our study, and to verify if the empirical study is confirmed by experimental results, we performed an extended set of optimization experiments using the same Morse instance with 50 atoms that was already selected for the locality analysis. The study focused on the behavior of the search algorithm when using different genetic operators. The settings are the following: Evaluations: 3,000,000; Population Size: 100; Elitist Strategy; Tournament Selection with tourney size 10; Crossover operators: {Uniform, C&S, GenC&S}; Crossover rate: 0.7; Mutation operators: {Sigma, Flip}; $\sigma = \{0.01 \times \lambda, 0.1 \times \lambda, 0.25 \times \lambda\}$; Mutation rate: {0.0, 0.01, 0.05, 0.1, 0.2, 0.3}; LSL: 200; δ: 0.5.

Each iteration performed by L-BFGS counts as one evaluation. The initial population is randomly generated and for every set of parameters we performed 30 runs. When appropriate, statistical significance of the results is accessed with a t-test (level of significance 0.01).

In tables 1 to 3 (respectively for uniform, C&S and GenC&S), we present an overview of the achieved results. For each one of the settings we present the average of the best fitness over the 30 runs (MBF). In brackets we also present the Gap, defined as the distance between the MBF and the putative optimum value for the potential energy of a Morse cluster with 50 atoms (gap value expressed in percentage).

If we combine the information obtained with the locality analysis and the optimization results it is possible to infer some conclusions concerning the behavior of the search algorithm. Experiments performed with uniform crossover obtain results of inferior quality than those achieved by the other two operators. This is true for all settings adopted during the tests. The analysis presented in the previous section showed that the locality properties of uniform crossover are comparable to those of GenC&S. In contrast, optimization results suggest that knowing the locality of an operator is not sufficient to predict its efficiency. The difference in performance between uniform crossover and the other two operators shows that, when solving difficult optimization problems, it is essential to rely on specific operators to explore the search space. Specific operators are sensitive to the structure of individuals being manipulated and, therefore, increase the probability of exchanging genetic material in a meaningful way.

Results in bold in tables 1 to 3 identify the best crossover operator for each specific setting. There is a clear separation between the settings where C&S

Table 1. Optimization results of the 50-atom Morse cluster using uniform crossover

	Mutation rate				
Mutation	0.01	0.05	0.1	0.2	0.3
Sigma 0.01	-188.247 (5.1)	-189.247 (4.6)	-190.449 (4.0)	-191.197 (3.7)	-191.894 (3.3)
Sigma 0.1	-192.831 (2.8)	-193.831 (2.3)	-194.086 (2.2)	-193.561 (2.5)	-194.348 (2.1)
Sigma 0.25	-194.433 (2.0)	-194.745 (1.9)	-193.548 (2.5)	-187.483 (5.5)	-183.937 (7.3)
Without mutation	-182.427 (8.1)				

Table 2. Optimization results of the 50-atom Morse cluster using C&S crossover

	Mutation rate				
Mutation	0.01	0.05	0.1	0.2	0.3
Sigma 0.01	**-195.656** (1.4)	**-195.066** (1.7)	**-195.246** (1.6)	**-195.223** (1.6)	**-195.547** (1.5)
Sigma 0.1	-194.505 (2.0)	**-195.329** (1.6)	**-194.971** (1.8)	-194.132 (2.2)	-194.455 (2.0)
Sigma 0.25	-193.995 (2.2)	-193.464 (2.5)	-191.784 (3.4)	-187.481 (5.5)	-183.347 (7.6)
Without mutation	**-194.816** (1.8)				

had the best performance and the settings where GenC&S was better. When Sigma0.01 mutation is used, experiments performed with C&S always achieve the best results. On the contrary, when Sigma0.25 is adopted, experiments performed with the new crossover operator always exhibit the best performance. When Sigma0.1 is used, differences in performance between these two operators are small (with the exception of the experiments performed with a mutation rate of 0.3). Table 4 reviews the statistical analysis performed. The symbol '$\star$' identifies settings where there are significant differences between the results achieved by experiments performed with C&S and GenC&S. Results show that they occur in two situations:

Table 3. Optimization results of the 50-atom Morse cluster using GenC&S crossover

	Mutation rate				
Mutation	0.01	0.05	0.1	0.2	0.3
Sigma 0.01	-193.708 (2.4)	-194.313 (2.1)	-193.967 (2.3)	-194.054 (2.2)	-194.760 (1.9)
Sigma 0.1	**-194.626** (1.9)	-194.789 (1.8)	-194.634 (1.9)	**-195.225** (1.6)	**-196.177** (1.2)
Sigma 0.25	**-195.254** (1.6)	**-195.272** (1.6)	**-195.275** (1.6)	**-190.498** (4.0)	**-186.093** (6.2)
Without mutation	-192.913 (2.8)				

Table 4. C&S *vs* GenC&S: Significant differences (50-atom Morse cluster)

	Mutation rate				
Mutation	0.01	0.05	0.1	0.2	0.3
Sigma 0.01	⋆				
Sigma 0.1					⋆
Sigma 0.25		⋆	⋆	⋆	⋆
Without mutation	⋆				

i) They are visible when the effect of mutation is almost irrelevant. This happens in the test performed without mutation and also in the experiment using Sigma0.01 with rate 0.01. In these scenarios C&S crossover is clearly more efficient than GenC&S. Locality analysis results help to explain why this happens. C&S does not require distinct parents to generate original descendants. It is therefore able to maintain and promote the diversity level of the population. The addition of a mutation operator with the ability to perform considerable changes in the individuals might lead to too large disruptions in the solutions preventing an efficient exploration of the search space.

ii) Significant differences also occur when mutation plays a major role in the optimization process. More specifically, differences appear in nearly all experiments performed with Sigma0.25 (the only exception being the situation with a mutation rate of 0.01) and also in the test performed with Sigma0.1 with rate 0.3. In all these situations, GenC&S was more efficient than the

original C&S operator. This result is also in accordance with the locality analysis. GenC&S is sensitive to the diversity level of the population and therefore it requires different parents to create children with a substantial level of innovation. Just like experimental results show, its performance is enhanced if the mutation operator helps to maintain an appropriate level of diversity. In the experiments that are between these two extremes, optimization results achieved by the two crossover operators can be considered similar.

Results from the tables also confirm the robustness of the hybrid EA. In most cases (particularly in experiments performed with crossover operators that are sensitive to the structures being manipulated), the gap to the putative global optimum is small, ranging between 1.5 and 2.0%.

6 Optimization of a Larger Cluster

We performed a final set of tests pertaining the optimization of a Morse cluster with 80 atoms. Our aim is twofold: first, we want to verify if the results achieved in a difficult optimization situation confirm our locality analysis. Also, we intend to collect some statistics during the runs to measure the diversity of the population throughout the optimization. In the previous section, the locality of genetic operators was studied separately. Now, by collecting these statistics on the fly, we expect to gain insight on how the combination of different genetic operators with other algorithmic components influence search dynamics. We will also verify whether these results confirm the static empirical analysis that was performed earlier.

6.1 Optimization Results

The settings of the experiments performed are as follows: Number of runs: 30; Evaluations: 3,000,000; Population size: 100; Elitist strategy; Tournament selection with tourney size 10; Crossover operators: {C&S, GenC&S} with rate 0.7; Sigma0.1 mutation with rate {0.0, 0.01, 0.05, 0.1, 0.2}; LSL: 200; δ: 0.5.

Here, we do not aim to conduct an all-inclusive study. Our goal is just to obtain an additional verification whether the locality analysis can find support in optimization results. That is why we maintain the settings selected for the optimization of the 50-atom cluster, even though we are aware that the number of evaluations should be increased to enable an appropriate exploration of the search space. Nevertheless, 3 million evaluations should be enough to provide some hints concerning the search performance of different genetic operators. Also, we selected just a subset of the genetic operators previously considered. On the one hand, we did not perform experiments with uniform crossover. Results achieved on the optimization of the 50-atom cluster show that it is clearly less efficient than the other two crossover operators and so

it was not considered in this last step of the research. As for mutation, we selected Sigma0.1, as it proved to be the most balanced operator.

In table 5 we show an overview of the achieved results. For each one of the selected settings we present the mean best fitness calculated over the 30 runs and the gap to the putative global optimum expressed in percentage (value in brackets). The first row presents results from experiments performed with C&S and the second pertains results achieved by tests done with GenC&S. Values in bold highlight settings where the MBF is significantly better than that achieved by the other test performed with the same mutation rate.

Table 5. Optimization results of the 80-atom Morse cluster

	Mutation rate				
	0.0	0.01	0.05	0.1	0.2
C&S	**-334.318** (1.9)	-332.984 (2.3)	-332.783 (2.4)	-331.433 (2.7)	-328.197 (3.7)
GenC&S	-329.520 (3.3)	-334.318 (1.9)	-336.083 (1.4)	**-335.755** (1.5)	**-337.043** (1.1)

A brief overview of the optimization results shows that experiments performed with GenC&S achieved better results than those that used original C&S crossover. The only exception is when mutation is absent. Here, the MFB is better in the experiment performed with the original C&S operator. Actually, results show that this is the best setting for C&S crossover. As soon as mutation is added, the efficiency of the operator decreases. This is true even for small rates, even though the increase of the mutation rate amplifies the effect. This situation was already visible in tests performed with the 50-atom cluster, but here it is more evident.

Experiments performed with GenC&S crossover achieve better results when mutation is used. It is also clear that fairly high mutation rates are required to enhance the performance of the algorithm when this type of crossover is used. The only experiment that was able to discover the putative global optimum for this instance was the one that combined GenC&S crossover and Sigma0.1 mutation with a rate of 0.2. Even though the relevance of this finding should be handled with care (just like we mentioned in the beginning of this section, the number of evaluations might be too small to allow a proper exploration of the search space), it nevertheless presents a sign concerning the efficiency of the different operators. Anyway, the gaps that exist between the MFB and the global optimum are more relevant to access the efficiency of the search algorithm. When using GenC&S crossover and Sigma mutation, gaps range between 1.1 and 1.9%. These low values demonstrate the competence

of the optimization method, showing that promising areas of the search space can be discovered even with a limited number of evaluations.

Table 6. C&S significant differences (80-atom Morse cluster)

	0.0	0.01	0.05	0.1	0.2
0.0					⋆
0.01					⋆
0.05					⋆
0.1					
0.2					

Table 7. GenC&S significant differences (80-atom Morse cluster)

	0.0	0.01	0.05	0.1	0.2
0.0		⋆	⋆	⋆	⋆
0.01					
0.05					
0.1					
0.2					

We performed a brief statistical analysis to confirm the validity of our conclusions. Values in bold in table 5 show that there is a significant difference between the MBF attained by C&S and GenC&S when no mutation is used. When a moderate mutation rate is adopted (0.01 and 0.05), GenC&S outperforms C&S even though differences are not statistically significant. As the mutation rate increases, differences in MBF become more evident. When 0.1 or 0.2 are used, there is already a significant difference between the results achieved by the two crossover operators. We also studied whether there are significant differences between experiments that used the same crossover operator and different mutation rates. Tables 6 and 7 summarize these results. Cells marked with '⋆' identify a situation where a significant difference exist.

The results are once again in compliance with our previous analysis. When using C&S crossover (table 6), all experiments performed with a mutation rate of 0.2 achieve significantly poorer results than experiments performed with other mutation rates. The only exception is when experiments with 0.1 and 0.2 rates are compared (in this situation, the difference is not significant). As for GenC&S (table 7), significant differences are observed when comparing experiments with and without mutation. All MBFs obtained in tests performed without mutation are significantly worse that those achieved by experiments that rely on mutation to promote diversity.

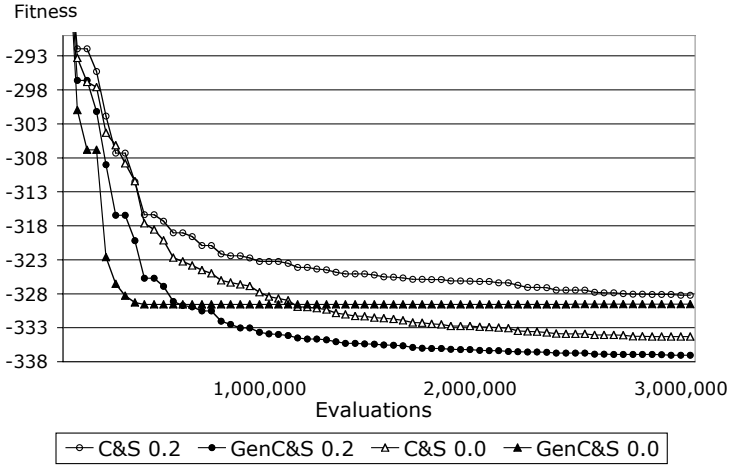

Fig. 7. Evolution of MBF in the optimization of the 80-atom cluster

A final chart concerning the optimization of the 80-atom cluster is presented in Fig. 7. It shows the evolution of the best solution (averaged over 30 runs) during the optimization for 4 selected settings: two experiments without mutation and two experiments using both operators (mutation rate 0.2). For the sake of clarity, lines obtained from experiments performed with other mutation rates (0.01, 0.05 and 0.1) are not shown. However, results presented are representative of the behavior of the optimization algorithm. When C&S crossover is used, there is a clear advantage if the mutation operator is not present. After the initial stage, when both lines exhibit a similar pattern, the experiment combining C&S and Sigma mutation starts to stagnate, suggesting that it is unable to identify promising areas of the search space. When GenC&S is used, two completely different patterns emerge. If this operator is used in combination with mutation, there is a continuous improvement of the best solution found. If mutation is not considered, then the algorithm quickly converges to a sub-optimal solution and it is completely unable to escape from

there. This is an expected result, given the locality analysis of GenC&S that was presented in Sect. 5.

6.2 Search Dynamics

The optimization results for the 80-atom cluster are in accordance with the locality analysis that was performed before. This agreement is another sign suggesting that locality analysis is a useful tool to predict the performance of evolutionary algorithms when solving difficult optimization problems.

Before concluding the chapter, we present a last set of results that hopefully will contribute to a full clarification of the behavior of the algorithm when exploring the coordinate space. Locality analysis is performed in static environments, where a single operator is applied at a time. In a real optimization situation, interactions that exist between the genetic operators and selection might influence the behavior of the algorithm. We will now present several measures that illustrate how different configurations of the search algorithm, in what concerns the genetic operators used and parameter settings adopted, are able to promote and maintain diversity in a population.

When the algorithm was searching for good solutions for the 80-atom cluster, we collected in several predetermined generations, {1, 5, 10, 20, 30, 50, 100, 150}, the average distance between all pairs of individuals from the population. The two distance measures previously defined were used. In Figs. 8 and 9 we show the distribution of the fitness distances obtained with different settings. The three charts from Fig. 8 display results achieved in experiments performed with C&S crossover, whereas charts from Fig. 9 present the outcomes from tests performed with GenC&S. We show results only from optimization experiments performed with the following mutation rates: {0, 0.01, 0.1}. Experiments performed with other mutation rates follow the same pattern.

Two clear distinct configurations emerge. When C&S crossover is used, the diversity of the population is similar throughout the optimization. Also, the mutation rate (and even the absence of mutation at all) does not influence the diversity level. Real optimization results thus confirm the static analysis. C&S crossover is able to maintain a high level of diversity, independently of the difference that exists between parents.

In experiments performed with GenC&S there is a clear distinction whether mutation is used or not. If crossover acts alone then the algorithm quickly converges to a situation where approximately 80% of the individuals are identical and the other 20% are distinct[3]. This result confirms the locality study that showed how GenC&S is unable to inject diversity in the population. As soon

[3] The 20% of descendants that have a considerable different potential energy from that of its parents is a consequence of GenC&S way of acting: when two mates are nearly identical, it might be impossible to select enough atoms from the parents in such a way that the minimum distance constraint is satisfied. If this happens the descendant is completed with atoms placed at random locations.

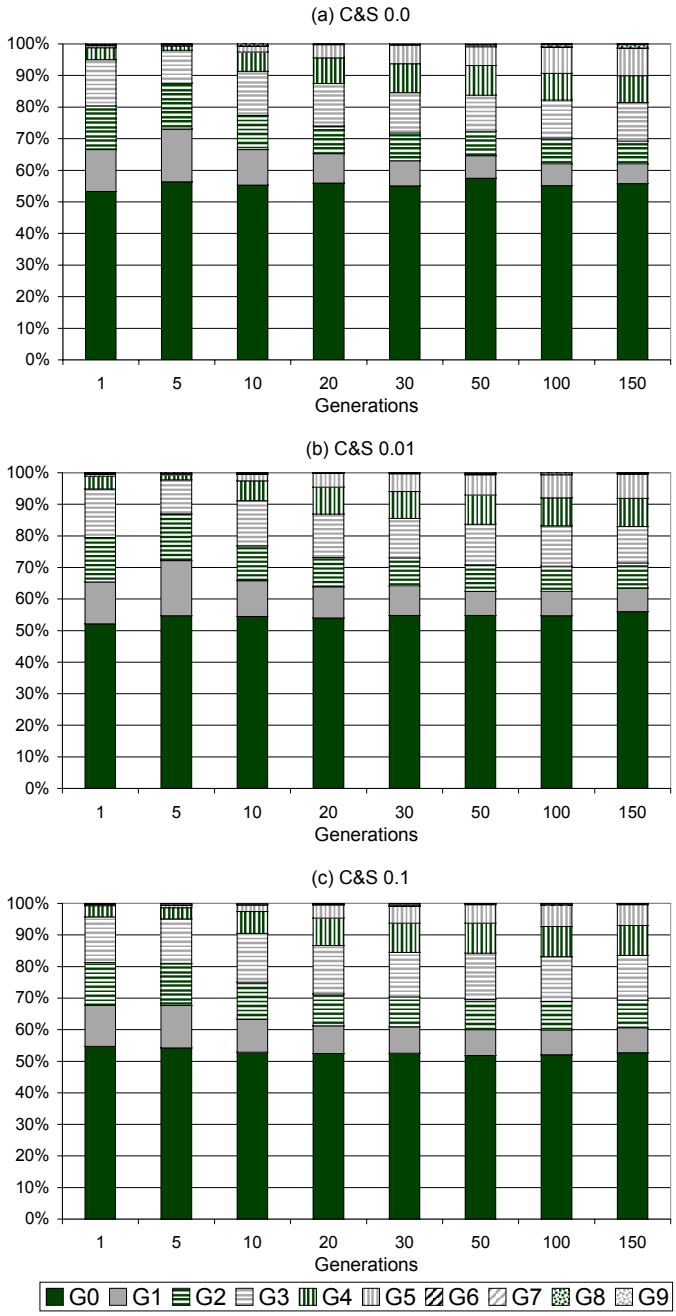

Fig. 8. Distribution of fitness distances in the population throughout the optimization of the 80-atom cluster: **(a)** C&S 0.0; **(b)** C&S 0.01; **(c)** C&S 0.1

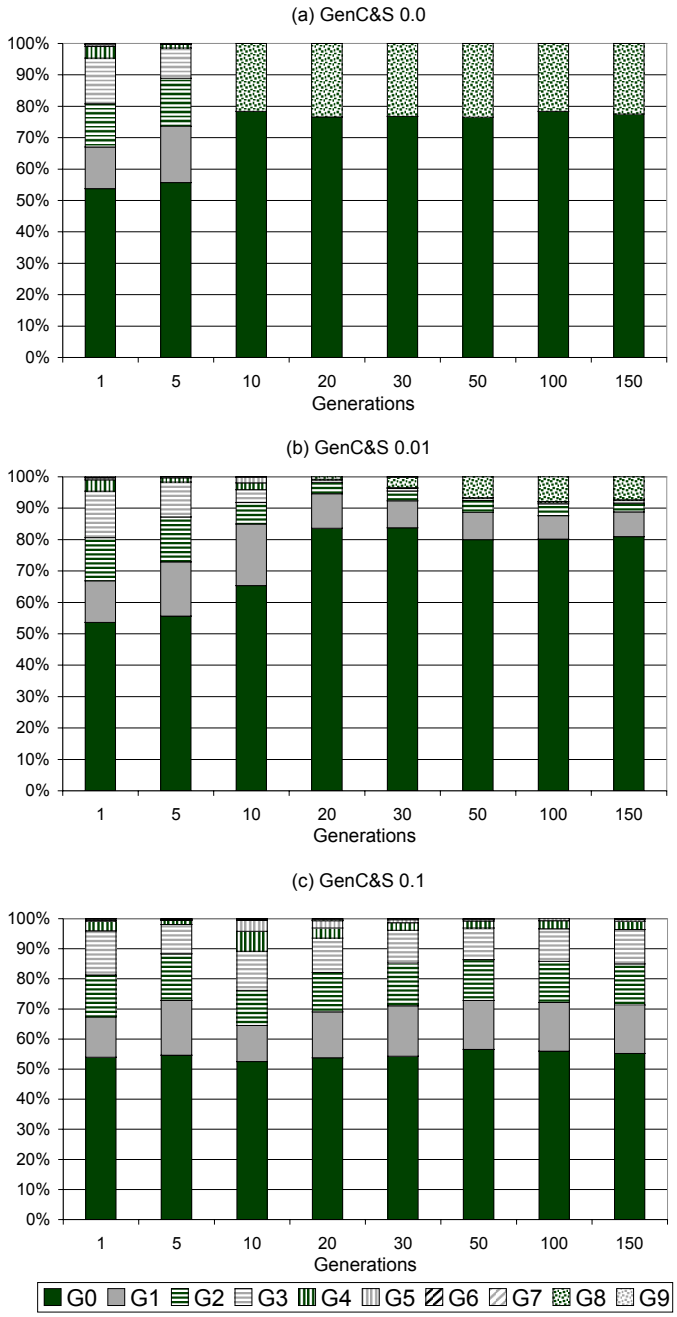

Fig. 9. Distribution of fitness distances in the population throughout the optimization of the 80-atom cluster: **(a)** GenC&S 0.0; **(b)** GenC&S 0.01; **(c)** GenC&S 0.1

as mutation is added, the problem is reduced. When a mutation rate of 0.01 is adopted, a small level of diversity is already visible, even though it is still clearly below the one that is verified in experiments performed with C&S. If the mutation rate is raised to 0.1, the diversity is comparable to the one that is visible in tests performed with the other crossover operator. This dependence on mutation to promote the diversity of the population was already predicted during the locality analysis of GenC&S.

The chart from Fig. 10 presents the results obtained by structural distance in the same optimization experiments. They confirm all the analysis. Actually, they are even more evident as they perfectly rank the diversity level achieved by different combinations of crossover and mutation.

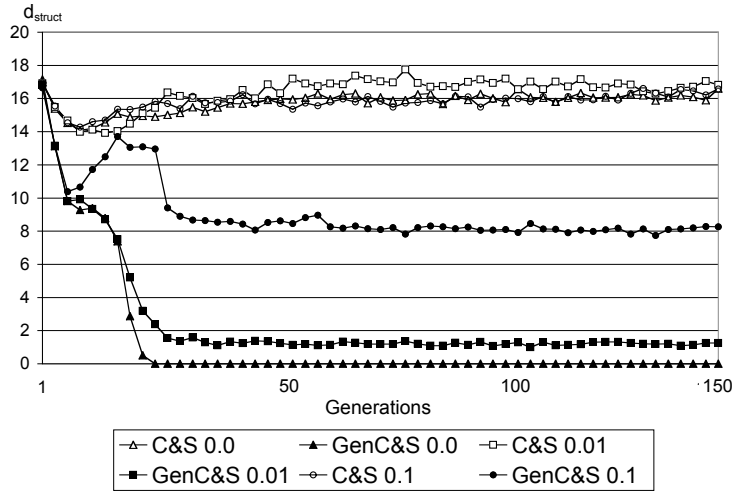

Fig. 10. Average structural distances in the population throughout the optimization of the 80-atom cluster

7 Conclusions

In this chapter we studied the locality properties of the hybrid evolutionary algorithm usually applied in cluster geometry optimization. Several Morse clusters instances, a well-known NP-hard benchmark model system, were selected for the analysis.

Two distance measures, required to determine the semantic difference between two solutions were used to conduct a comprehensive analysis concerning the locality strength of an extended set of mutation and crossover operators.

In what concerns mutation, the empirical study showed that there are important differences in the locality level induced by different operators. Sigma mutation is an appropriate variation operator, but a moderate standard deviation should be selected to ensure the preservation of a reasonable correlation between individuals before and after the application of this operator. Conversely, flip mutation has low locality. This operator tends to perform large jumps in the search space, complicating the task of the exploration algorithm.

As for crossover, interesting patterns emerge from the analysis. C&S, which is the most widely used operator for cluster optimization, showed a remarkable innovation capacity. This operator is able to generate original descendants, even when the diversity of the population is low. This can be considered as an interesting behavior, but it also suggests that C&S might have difficulties in performing a meaningful identification and combination of important features that exist in parents. The other two operators considered in the analysis, GenC&S and uniform crossover, exhibit a behavior that is more in compliance to what is expected from crossover. If the mating parents are similar, the descendants tend to be analogous to them. If the distance between parents is high, the probability of generating a child with distinct features increases.

To validate our analysis we performed several optimization experiments using different settings and distinct combinations of genetic operators. Experimental outcomes support the most relevant results of the locality analysis and confirm the role played by different genetic operators.

This study is part of an ongoing project concerning the application of EAs to optimization problems from the Chemistry area. In the near future, we plan to extend our model to consider heritability and heuristic bias, the other two features that compose the original framework proposed by Raidl and Gottlieb. Results obtained with this analysis will be valuable for the future development of enhanced methods to employ in optimization problems with properties similar to the ones addressed at this research.

Acknowledgments

This work was supported by Fundação para a Ciência e Tecnologia, Portugal, under grant POSC/EIA/55951/2004.

We are grateful to the John von Neumann Institüt für Computing, Jülich, for the provision of supercomputer time on the IBM Regatta p690+ (Project EPG01).

References

1. J. E. Jones. On the Determination of Molecular Fields. II. From the Equation of State of a Gas. Proc. Roy. Soc. A, 106, 463-477, 1924.
2. J. E. Lennard-Jones. Cohesion. Proc. Phys. Soc., 43, 461-482, 1931.

3. P. Morse. Diatomic Molecules According to the Wave Mechanics. II. Vibrational Levels. Phys. Rev., 34, 57–64, 1929.
4. J. P. K. Doye, R. Leary, M. Locatelli and F. Schoen. Global Optimization of Morse Clusters by Potential Energy Transformations, Informs Journal on Computing, 16, 371–379, 2004.
5. R. L. Johnston. Evolving Better Nanoparticles: Genetic Algorithms for Optimising Cluster Geometries, Dalton Transactions, 22, 4193–4207, 2003.
6. D. M. Deaven and K. Ho. Molecular Geometry Optimization with a Genetic Algorithm, Phys. Rev. Lett. 75, 288–291, 1995.
7. B. Hartke. Global Geometry Optimization of Atomic and molecular Clusters by Genetic Algorithms, In L. Spector et al. (Eds.), Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001), 1284–1291.
8. B. Hartke. Application of Evolutionary Algorithms to Global Cluster Geometry Optimization, In R. L. Johnston (Ed.), Applications of Evolutionary Computation in Chemistry, Structure and Bonding, 110, 33–53, 2004.
9. F. Manby, R. L. Johnston and C. Roberts. Predatory Genetic Algorithms. Commun. Math. Comput. Chem. 38, 111–122, 1998.
10. W. Pullan. An Unbiased Population-Based Search for the Geometry Optimization of Lennard-Jones Clusters: $2 \leq N \leq 372$. Journal of Computational Chemistry, Vol. 6, No. 9, pp. 899–906, 2005.
11. C. Roberts, R. L. Johnston and N. Wilson (2000). A Genetic Algorithm for the Structural Optimization of Morse Clusters. Theor. Chem. Acc., 104, 123–130, 2000.
12. Y. Zeiri. Prediction of the Lowest Energy Structure of Clusters Using a Genetic Algorithm, Phys. Rev., 51, 2769–2772, 1995.
13. J. Gottlieb and C. Eckert. A Comparison of Two Representations for the Fixed Charge Transportation Problem, In M. Schoenauer et al. (Eds.), Parallel Problem Solving from Nature (PPSN VI), 345–354, Spinger-Verlag LNCS, 2000.
14. J. Gottlieb and G. Raidl. Characterizing Locality in Decoder-Based EAs for the Multidimensional Knapsack Problem, In C. Fonlupt et al. (Eds.), Artificial Evolution: Fourth European Conference, 38–52, Springer-Verlag LNCS, 1999.
15. G. Raidl and J. Gottlieb. Empirical Analysis of Locality, Heritability and Heuristic Bias in Evolutionary Algorithms: A Case Study for the Multidimensional Knapsack Problem. Evolutionary Computation, Vol. 13, No. 4, 441–475, 2005.
16. F. Rothlauf and D. Goldberg, Prüfernumbers and Genetic Algorithms: A Lesson on Hoe the Low Locality on an Encoding Can Harm the Performance of Gas, In M. Schoeneauer et al. (Eds.), Parallel Problem Solving from Nature PPSN VI, 395–404, 2000.
17. F. Rothlauf. On the Locality of Representations, In E. Cantú-Paz et al. (Eds.), Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2003), Part II, 1608–1609, 2003.
18. B. Sendhoff, M. Kreutz and W. Seelen. A Condition for the Genotype-Phenotype Mapping: Causality. In T. Bäck (Ed.), Proceedings of the 7th International Conference on Genetic Algorithms (ICGA-97), 73–80, 1997.
19. F. B. Pereira, J. M. C. Marques, T. Leitão, J. Tavares. Analysis of Locality in Hybrid Evolutionary Cluster Optimization. In G. Yen et al. (Eds.), Proceedings of the IEEE Congress on Evolutionary Computation (CEC-2006), pp. 8049–8056, 2006.

20. J. P. K. Doye and D. J. Wales. Structural Consequences of the Range of the Interatomic Potential. A Menagerie of Clusters. J. Chem. Soc. Faraday Trans. 93, 4233–4243, 1997.
21. D. J. Wales et al. The Cambridge Cluster Database, URL: `http://www-wales.ch.cam.ac.uk/CCD.html`, accessed on January 2007.
22. D. C. Liu and J. Nocedal. On the Limited Memory Method for Large Scale Optimization, Mathematical Programming B, 45, 503–528, 1989.
23. J. Nocedal. Large Scale Unconstrained Optimization, In A. Watson and I. Duff (Eds.), The State of the Art in Numerical Analysis, 311–338, 1997.
24. S. Wright. The Roles of Mutation, Inbreeding, Crossbreeding and Selection in Evolution. In Proceedings of the 6th International Conference on Genetics, Vol. 1, 356–366, 1932.
25. T. Jones and S. Forrest. Fitness Distance Correlation as a Measure of Problem Difficulty for Genetic Algorithms. In L. Eshelman (Ed.), Proceedings of the 6th International Conference on Genetic Algorithms (ICGA-95), 184–192, 1995.
26. P. Merz. Memetic Algorithms for Combinatorial Optimization Problems: Fitness Landscapes and Effective Search Strategies. Ph.D. Thesis, Department of Electrical Engineering and Computer Science, University of Siegen, Germany, 2000.
27. E. D. Weinberger. Correlated and Uncorrelated Fitness Landscapes and How to Tell the Difference, Biological Cybernetics, Vol. 63, 325–336, 1990.
28. W. Hart, T. Kammeyer, R. Belew. The Role of Development in Genetic Algorithms. In D. Whitley and M. Vose, (Eds.), Foundations of Genetic Algorithms 3, Morgan Kaufmann, pp. 315–332, 1995.
29. D. Thierens, D. Goldberg. Mixing in Genetic Algorithms. In S. Forrest (Ed.), Proceedings of the Fifth International Conference on Genetic Algorithms (ICGA-93), Morgan Kaufmann, pp. 38–45, 1993.