The Impact of Local Search on Protein-Ligand Docking Optimization

Jorge Tavares, Alexandru-Adrian Tantar, Nouredine Melab and El-Ghazali Talbi
INRIA Lille - Nord Europe Research Centre
Parc Scientifique de la Haute Borne
59650 Villeneuve d'Ascq, France
jorge.tavares@inria.fr, {alexandru-adrian.tantar, nouredine.melab, el-ghazali.talbi}@lifl.fr

Abstract

Common evolutionary approaches to protein-ligand docking optimization use mutation operators based on Gaussian and Cauchy distributions, with local search hybrids. The choice of a local search method is important for an efficient algorithm. We investigate the impact of local search with mutation operators by performing a locality analysis. High locality means that small variations in the genotype imply small variations in the phenotype. Results show that local search hybrids reduce locality and act as local optimizers with the solution as a starting point.

1 Introduction

The protein-ligand docking problem consists in finding the best ligand conformation and orientation relative to the active site of a target protein [8]. By considering the relative orientation and conformations of the two molecules the problem becomes very hard. Typically, the protein is fixed in a three-dimensional coordinate system while the ligand can be repositioned and rotated. The problem difficulty increases when both protein and ligand are allowed to be flexible. Therefore, the problem is classified, by increasing complexity, into the ensuing categories: rigid-structure docking (both molecules are rigid); rigid protein and flexible ligand; flexible protein and rigid ligand; and, both molecules are flexible. Having both molecules flexible, generally the active site of the protein and the ligand, the problem becomes harder. A higher degree of flexibility implies a considerable increase of the search space size.

Several docking methods have been proposed for the past years, e.g., incremental construction algorithms, stochastic algorithms and molecular dynamics. We refer the reader to several review studies [6, 4, 14] for more details. Evolutionary and swarm algorithms have recently become one of the most common search techniques applied, and proving to be successful [14, 3]. Although several applications exist, no comprehensive set of studies could be found *to understand why* these algorithms are successful. The only attempt made, to the best of our knowledge, was [13] where several parameters (e.g., population size) and some genetic operators are empirically investigated. For an efficient evolutionary algorithm, it is important to understand its components influence and interplays.

The aim of this paper is to study the impact of local search methods on the evolutionary algorithm model [5, 14] usually adopted for protein-ligand docking optimization. Locality measures for the analysis are used from the framework proposed by [9]. In this framework, local search hybrids are used in conjuction with mutation (mutation is the most frequent operator considered in locality studies). Locality is an important requisite to ensure the efficiency of search and it has been widely studied by the evolutionary computation community [11, 10, 9]. This property indicates that small variations in the genotype space imply small variations in the phenotype space [11]. 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 algorithm is inefficient and, in a worse case scenario, tends to resemble random search. In this paper, we concentrate on two methods applied in bio-inspired approaches for protein-ligand docking and continue the investigation initiated in [12]. In our previous work, we studied the degree of locality induced by different mutation operators. Results show that a Gaussian-based operators offer stronger locality than Cauchy-based ones. In addition, the locality analysis explains why Gaussian-based operators present better optimization results.

In this work, the empirical analysis show the effect of local search hybrids on the degree of locality induced by mutation. The methods act as local optimizers and have a disruptive effect on locality. Moreover, the first method's behavior is constant regardless of the operator. The second local search method shows a different effect according to the operator used. Understanding the role played by each

algorithm's component may provide useful insights for future applications of evolutionary algorithms to this problem.

The rest of the paper is structured as follows. Section 2 contains an overview of the evolutionary algorithm's components used in our experimentation. In section 3 we present our experimentation and respective discussion. Finally, section 4 contains the main conclusions.

2 Evolutionary Algorithms and Protein-Ligand Docking

Evolutionary algorithms applied to molecular docking can be found since 1993 [1]. In [5] is presented one of the most important evolutionary algorithms proposed. Commonly referred to as *AutoDock*, this approach is a conformational search method which uses an approximate physical model to evaluate possible protein-ligand conformations. It incorporates flexibility by allowing the ligand to change its conformation during the docking simulation. In addition, it pre-calculates the pairwise interactions between atoms, considerably speeding up the docking simulation. To search the space of possible protein-ligand conformations, an evolutionary algorithm is used with local search. The application of this method results in the genotype of the individuals being replaced with the new best solution found. This process is usually referred to as Lamarckian evolution.

In our analysis, we adopt an experimental model which uses the main components from [5], because it serves as a basis for the large majority of evolutionary-inspired approaches (e.g.,[14, 3]).

2.1 Encoding

An individual represents only the ligand. The protein remains rigid whilst the ligand is flexible during the docking process. A genotype of a candidate solution is encoded by a vector of real-valued numbers which represent the ligand's orientation, translation and torsion angles[5]. Cartesian coordinates represent the ligand translation, three variables in the vector, whereas four variables defining a quaternion represent the ligand orientation. A quaternion can be considered to be a vector (x, y, z) which specifies an axis of rotation with an angle θ of rotation for this axis. For each flexible torsion angle one variable is used.

The genotype is translated into a phenotype composed of the atomic coordinates that represent the three-dimensional structure of the ligand. Therefore, we work with an indirect representation. The atomic structure of the ligand is built from the translation and orientation coordinates in the ligand crystal structure with the application of the flexible torsion angles.

2.2 Evaluation

An energy evaluation function is used to evaluate each individual. The fitness for each candidate solution is given by the sum of the intermolecular interaction energy between the ligand and the protein, and the intramolecular energy that arises from the ligand itself [5]. An empirical free energy potential composed of five terms is used by *AutoDock*. The first three terms are pairwise interatomic potentials that account for weal long-range attractive forces and short-range electrostatic repulsive forces. The fourth term measures the unfavorable entropy of a ligand binding due to the restriction of conformational degrees of freedom. The fifth and last term uses a desolvation measure. Further details of the energy terms and how the potential is derived can be found in [5].

2.3 Genetic Operators

Mutation is performed by using evolutionary strategies based operators. The genetic operator acts in the following way: when undergoing mutation, the new value for a gene x' is obtained from the old value x by adding a random real number sampled from a distribution: $x' = x + \sigma \times U(0, 1)$. The common distribution used for $U(0, 1)$ is the standard Gaussian distribution. In spite of that, the *AutoDock* approach replaces the Gaussian distribution with a Cauchy distribution:

$$C(x, \alpha, \beta) = \frac{\beta}{\pi\beta^2 + (x - \alpha)^2} \tag{1}$$

where $\alpha \leq 0, \beta > 0, -\infty < x < +\infty$ (α and β are parameters that control the mean and spread of the distribution). The Cauchy distribution has a bias toward small variations. However, unlike the Gaussian distribution, it has thick tails which allows larger variations more frequently. To control the variance we apply the annealing scheme $\sigma(t) = \frac{1}{\sqrt{1+t}}$. Results show it presents good results [13].

2.4 Local Search

In this paper we study the impact of two local search methods. The first technique is the algorithm used in [5]: the Solis-Wets method. The Solis-Wets algorithm is a direct search method with an adaptive step size. which performs a randomized local minimization of a given candidate solution. The process starts with a candidate solution $x \in \Re^n$ and for each step a deviate $\epsilon \in \Re$ is chosen from a normal distribution. In the case a better solution is found, by adding or removing ϵ from x, the current solution is replaced with the new one. A *success* is recorded otherwise it is a *failure*. If several successes occur in a row, the variance of the normal distribution is adapted for the search to move more

quickly. If the opposite occurs, the variance is adapted to focus the search. This is accomplished through a parameter, ρ. Moreover, a bias term is applied to drive the search in successful directions. The method ends when a certain certain lower-bound threshold for ρ is passed or a maximum number of steps is reached.

The second method used is the simplex local search algorithm described by Nelder and Mead (NMS) for nonlinear, continuous function optimization [7]. A simplex is a polytype of $n+1$ vertices in an *n*-dimensional space. The NMS algorithm transforms the points of a given starting simplex by using a set of operations: reflection, expansion and contraction. These operations are applied until the fractional range (from the highest to the lowest point in the simples) is less than a tolerance value or a maximum allowed number of function evaluations occurs. The NMS method was used as the local search method in [3].

3 Locality Analysis

To perform our locality tests, we selected the HIV-1 protease/XK263 (1hvr) protein-ligand complex from the *AutoDock* test suit [5]. This complex has 10 rotatable bonds with 8 torsional degrees of freedom. This implies a total of 17 degrees of freedom and is one of the largest complexes in the test suit. In addition, we made experiments with other instances and found the same patterns. Due to space constraints, we only present the results for the 1hvr complex.

3.1 Related Work

Several techniques have been proposed to estimate and study the behavior of evolutionary algorithms and their components. Some of these methods adopt measures that are, to some extent, similar to the locality property. We highlight the most relevant ones.

The concept of fitness landscapes, originally proposed by [16], establishes a connection between candidate solutions and their fitness. Moreover, [2] proposed fitness distance correlation as a way to determine the relation between fitness and distance to the optimum. If fitness values increase as the distance to the optimum decreases, then search is expected to be easy. An alternative way to analyze the fitness landscape is to determine its ruggedness. In [15], it is proposed the adoption of autocorrelation functions to measure the correlation of all points in the search space at a given distance. In [11], it is studied the conditions for strong causality. A search process is said to be locally strong causal if small variations in the genotype space imply small variations in the phenotype space. Variations in genotypes are caused by mutation.

3.2 Definitions

Investigations with an evolutionary framework usually means considering two spaces: the genotype space Φ_g and the phenotype space Φ_p. Genetic operators are applied on Φ_g while the fitness function, f, is applied to solutions from the phenotype space: $f : \Phi_p \rightarrow \Re$. To establish 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. The phenotypic distance can be determined with a structural distance measure. To evaluate a final ligand conformation we compare it with the experimental structures using the standard Cartesian root-mean-square deviation (RMSD):

$$RMSD_{lig} = \sqrt{\frac{\sum_{i=1}^{n} dx_i^2 + dy_i^2 + dz_i^2}{n}} \qquad (2)$$

where n is the number of atoms in the comparison and dx_i^2, dy_i^2 and dz_i^2 are the deviations between the crystallographic structure and the corresponding coordinates from the predicted structure lig on Cartesian coordinate i. RMSD values below or near 1.0Å can be considered to be a success criterion. Thus, lower values mean that the observed and the predicted structures are similar. Therefore, our structural distance measure determines the difference between RMSD values of two phenotypes:

$$d_{struct}(A, B) = |RMSD_A - RMSD_B| \qquad (3)$$

We adopt the innovation measure proposed by Raidl and Gottlieb [9] to study the effect of mutation on locality. To predict the effect of applying this operator we use the distance between individuals in a mutation step. Let X be a solution and X^m the result of applying m mutation steps to X, then the Mutation Innovation (MI) is given by:

$$MI = dist(X, X^m) \qquad (4)$$

The distance measure used can be either fitness-based or structural. 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. The distance between the two solutions is small. On the other hand, operators with weak locality allow large jumps on the search space. To evaluate MI, 1000 random individuals are generated. Afterwards, a sequence of mutation steps is applied to each one of them and the distance between the original individual and the new solution is measured. In our experimentation, we start by applying a single mutation step. Later, we repeat the experiment with k successive mutation steps, with $k \in \{1, 2, 4, 8, 16, 32, 64, 128, 256, 512\}$.

	Gaussian	Cauchy	Gaussian+Solis-Wets	Cauchy+Solis-Wets	Gaussian+NMS	Cauchy+NMS
$P(MI=0)[\%]$	9.1	7.4	0	0	0	0
$E(MI\|MI>0)$	0.03	0.11	3.54	3.5	1.15	1.1
$S(MI\|MI>0)$	0.07	0.39	2.52	2.51	0.86	0.84
$Max(MI)$	0.75	5.54	12.1	12.94	6.66	7.22

Table 1. Characteristic values for the mutation innovation MI with $k = 1$.

3.3 Experimentation and Discussion

Table 1 shows the characteristic values for MI with a single mutation ($k = 1$). $P(MI = 0)$ represent the percentage of cases for which $MI = 0$. $E(MI|MI > 0)$ and $\sigma(MI|MI > 0)$ show the mean value and the standard deviation of MI, for $MI > 0$. They act as estimations for the expected values. $Max(MI)$ gives the maximum value.

We start by considering the case where mutation does not affect the phenotype, $MI = 0$ (occurring with probability $P(MI = 0)$). A large value of $P(MI = 0)$ indicates that mutation does not make often moves in the search space. In alternative, it may also be an evidence of redundancy or strong heuristic bias since many elements could map to the same phenotype. As expected, adding a local search method prevents the possibility of occurring such event. Table 1 shows a zero percentage of $MI = 0$ for both operators and local search methods. Local search methods ensures that different phenotypes are generated. This is not the case when these methods are not present. Mutation alone produces phenotypes which are not different from the original individual. However, it must be noticed it is a very small probability.

Moving on to $E(MI|MI > 0)$, $\sigma(MI|MI > 0)$ and $Max(MI)$, in general, small values indicate high locality. A single mutation changes the phenotype only a little and thus, should be aspired [9]. Although lower values are good signs for a *good* locality, it should be noted that larger values for the standard deviation and for Max of MI may not necessarily be a bad indication. From table 1 we can see that both local search methods increase the locality measures considerably. A direct comparison between the operators with and without local search, on both methods, we see a significant difference. Gaussian and Cauchy mutation present low values for $E(MI|MI > 0)$ (0.03 and 0.11) and $\sigma(MI|MI > 0)$ (0.07 and 0.39). By adding the Solis-Wets method to the operators makes these estimations values increase to 3.54 and 3.5 (for $E(MI|MI > 0)$), and to 2.52 and 2.51 (for $\sigma(MI|MI > 0)$). This indicates that for a single mutation step, the degree of innovation introduced by the local search method is very large. In fact, by looking at the phenotypes, the semantic relationship between the original individual and the new phenotype is very low. This shows that the local search method is acting more like an optimizer. In this case, the individual serves as a starting point for the Solis-Wets algorithm. This degree of locality and innovation induced by the combination of a mutation operator and this method, would be more expected in the case when several mutation steps of an operator are applied. The same effect can be reported for NMS method. Values are also very large when comparing to mutation without local search: 1.15 and 1.1 for $E(MI|MI > 0)$, 0.86 and 0.84 for $\sigma(MI|MI > 0)$ (Gaussian and Cauchy respectively). The difference of behavior is when comparing to Solis-Wets method. Although the values are larger, they are considerably smaller than the ones given by the application of Solis-Wets. The NMS method is able to preserve more locality in spite of also acting as a local optimizer. To establish if these differences are statistically significant, we performed the Wilcoxon rank sum test with significance value

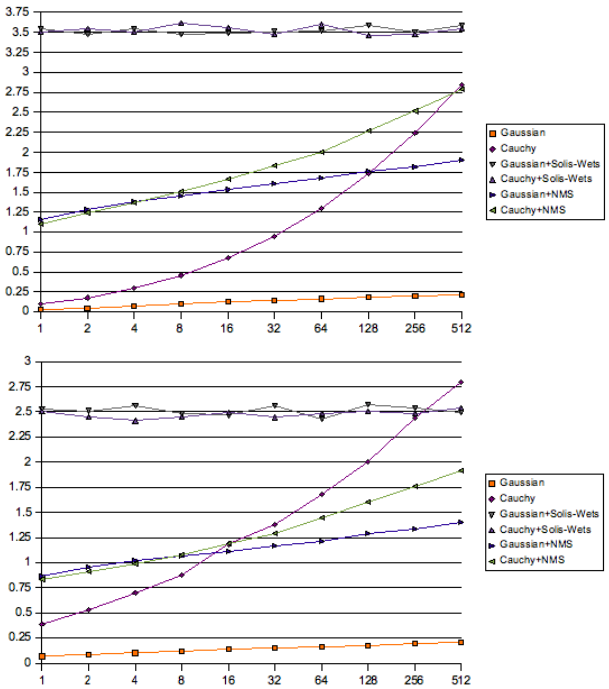

Figure 1. $E(MI|MI > 0)$ and $\sigma(MI|MI > 0)$, top and bottom respectively, over the number of $k \geq 1$ mutations.

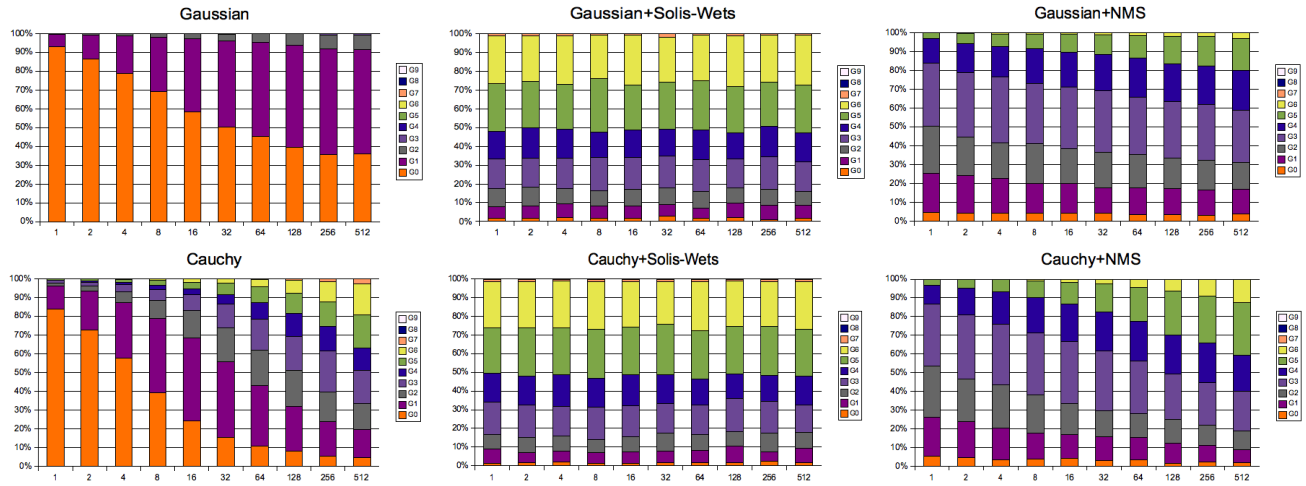

Figure 2. Distribution of structural distances for $k \geq 1$ mutations.

$\alpha = 0.01$. We found significant differences between both local search methods. Differences between operators with the same distribution were not found.

How does the distribution of innovation changes when considering $k > 1$ mutations? We will now consider the case for $k \in \{1, 2, 4, 8, 16, 32, 64, 128, 256, 512\}$. Figure 1 plots the empirically obtained mean values $E(MI|MI > 0)$ and $\sigma(MI|MI > 0)$ over the number of mutations k.

Gaussian mutation without local search shows lower mean and standard deviations than all the other variants. The contrast is evident since the values are always low, with a very slight increase with the number of mutations. In the other opposite we find the operators with the Solis-Wets method. Both operators, Gaussian and Cauchy with Solis-Wets, present very high values for the mean and standard deviations. However, increasing the number of mutation steps does not change the behavior. For both measures, the Solis-Wets operators remain constant around the 3.5 value. When looking at $E(MI|MI > 0)$ and $\sigma(MI|MI > 0)$ over the number of mutations clearly shows the strong locality given by Gaussian Mutation and the low locality properties induced by the Solis-Wets method, on both types of operators. The remaining genetic operators present a different behavior.

Regarding $E(MI|MI > 0)$ values, Cauchy mutation without local search presents higher values as the number of k mutation steps increases. The rise is very accentuated: for $k = 1$ MI is close to zero, by $k = 32$ is almost 1 and with $k = 512$, MI is higher than 2.75. For the operators with NMS local search, MI values also rise with the number of mutation steps. Nonetheless, the behavior is a little more stable than Cauchy mutation without local search. Gaussian mutation with NMS presents a stable curve, starting from values around 1.15 and finishing close to 2.0. For Cauchy mutation with NMS, the pattern is similar although it shows higher MI values as the number of mutation steps increase. By $k = 512$ the innovation mean is around 2.75.

Observing $\sigma(MI|MI > 0)$ values, we find the same behavior: Cauchy mutation presents a steep rise of values; operators with NMS are more stable with the Cauchy variant presenting final higher values. Cauchy mutation without local search shows some *good* locality but with a high number of mutations steps, locality properties start to be lost. The type of curves indicate that for a higher number of mutations, this operator might become inefficient. Adding the NMS method as local search introduce higher innovation values, although lower than Solis-Wets method. In spite for a low number of mutation steps these operators don't induce a strong locality, in the long term, they could induce some necessary innovation to help an algorithm escape local optima. This is especially true when comparing to Gaussian mutation without local search. This method presents a very strong locality but it could prove excessive [12]. Comparing the local search methods, this analysis demonstrates that they act as local optimizers. This is more evident in the case of the Solis-Wets method.

Grouping the distances between the original solution and the successive mutants allow us to observe the different types of changes operated by mutation for $E(MI|MI > 0)$. Given a structural distance d_{struct} between two phenotypes, the set G_i to which d_{struct} is assigned is determined the following way: $\{G0 : 0 \leq d_{struct} < 0.1; G1 : 0.1 \leq d_{struct} < 0.5; G2 : 0.5 \leq d_{struct} < 1; G3 : 1 \leq d_{struct} < 2; G4 : 2 \leq d_{struct} < 3; G5 : 3 \leq d_{struct} < 5; G6 : 5 \leq d_{struct} < 10; G7 : 10 \leq d_{struct} < 25; G8 : 25 \leq d_{struct} < 50; G9 : 50 \leq d_{struct}\}$. The specific values that were selected to determine intervals are arbitrary. The relevant information is the distribution of the structural

distances through the sets. Low order sets (i.e., small variations) suggest that locality is strong.

The charts in figure 2 show the distribution of structural distances for 1000 individuals, for all operators, with each column representing a mutation step. Important differences can be observed. Operators without local search exhibit high locality: Gaussian mutation has the values distributed on the first groups. Cauchy mutation also presents a large number of values in the first groups. For a large number of k the loss of locality properties is shown by the number of groups. The pattern given by the Solis-Wets method shows what we already know: the method acts a local optimizer with no locality. The groups distribution of the NMS method displays a weaker locality for low values of k. For larger values, the values distribution suggest a higher locality than simple Cauchy mutation but weaker than simple Gaussian mutation. The correct use of the NMS method prove helpful. However, this study indicates that more suitable local search methods for this problem can be studied and developed.

4 Conclusions

The majority of evolutionary algorithms applied to molecular docking use local search methods, mainly Solis-Wets, and Cauchy-based mutation operators. Because of this, it is important to understand their behavior and related operators. However, no studies were performed to conclude about its efficiency and performance with the exception of [13], where several parameters (e.g., population size) and some genetic operators are empirically investigated.

We investigated the impact of local search methods on the degree of locality induced by different mutation operators, when applied to protein-ligand docking optimization. Results confirm common local search methods have a high influence on locality. The outcome from this work is useful in two maners: 1) it explains in terms of locality the effect of local search and operators under investigation; 2) it provides hints on how future mutation methods can be developed. Is important for an operator to induce strong locality to obtain good optimization results. This result is sustained by the study described in [13, 12]. Gaussian mutation provides locally strong operators. This is an indication that more fine-tuning of the conformations is allowed. On the other hand, local search induced a lower degree of locality. This could be useful if used correctly, i.e., on a small part of the population and/or for briefs periods of time. Although results from optimization runs show this effect [13], there are other components which also have a direct influence on the search process. As such, it is necessary to extend our research to them, e.g., crossover, and perform additional optimization runs. We will also complete it with additional local search methods, such as scatter search and gradient methods.

References

[1] J. S. Dixon. Flexible docking of ligands to receptor sites using genetic algorithms. In *Proc. of the 9th European Symposium on Structure-Activity Relationships*, pages 412–413, The Netherlands, 16-21 July 1993. ESCOM Science.

[2] T. Jones. *Evolutionary Algorithms, Fitness Landscapes and Search*. PhD thesis, University of New Mexico, Albuquerque, New Mexico, May 1995.

[3] O. Korb, T. Stützle, and T. Exner. An ant colony optimization approach to flexible protein-ligand docking. *Swarm Intelligence*, 1(2):115–134, December 2007.

[4] N. Moitessier, P. Englebienne, D. Lee, J. Lawandi, and C. Corbeil. Towards the development of universal, fast and highly accurate docking/scoring methods: a long way to go. *British Journal of Pharmacology*, 153:1–20, 2007.

[5] G. M. Morris, D. S. Goodsell, R. S. Halliday, R. Huey, W. E. Hart, R. K. Belew, and A. J. Olson. Automated docking using a lamarckian genetic algorithm and and empirical binding free energy function. *Journal of Computational Chemistry*, 19(14):1639–1662, 1998.

[6] G. M. Morris, A. J. Olson, and D. S. Goodsell. Protein-ligand docking. In D. E. Clark, editor, *Evolutionary Algorithms in Molecular Design*, chapter 3, pages 31–48. Wiley-VCH, 2000.

[7] J. A. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal*, 7:308–313, 1965.

[8] A. Neumaier. Molecular modeling of proteins and mathematical prediction of protein structure. *SIAM Review*, 39(3):407–460, 1997.

[9] G. R. Raidl and J. Gottlieb. Empirical analysis of locality heriability and heuristic bias in evolutionary algorithms: A case study for the multidimensional knapsack problem. *Evolutionary Computation Journal*, 13(4):441–475, December 2005.

[10] F. Rothlauf. On the locality of representations. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2003)*, pages 1608–1609, 2003.

[11] B. Sendhoff, M. Kreutz, and W. V. Seelen. A condition for the genotype-phenotype mapping: Casualty. In *7th Int. Conf. on Genetic Algorithms*, pages 73–80, 1997.

[12] J. Tavares, A.-A. Tantar, N. Melab, and E.-G. Talbi. The influence of mutation on protein-ligand docking optimization: a locality analysis. In *Proceedings of the 10th International Conference on Parallel Problem Solving From Nature*, 13-17 September 2008.

[13] R. Thomsen. Flexible ligand docking using evolutionary algorithms: investigating the effects of variation operators and local search hybrids. *Biosystems*, 72(1-2):57–73, 2003.

[14] R. Thomsen. Protein-ligand docking with evolutionary algorithms. In G. B. Fogel, D. W. Corne, and Y. Pan, editors, *Computational Intelligence in Bioinformatics*, chapter 8, pages 169–195. Wiley-IEEE Press, 2008.

[15] E. D. Weinberger. Correlated and uncorrelated fitness landscapes and how to tell the difference. *Biological Cybernetics*, 63:325–336, 1990.

[16] S. Wright. The roles of mutation, inbreeding, crossbreeding and selection in evolution. In *Proceedings of the VI International Conference on Genetics, Vol. 1*, pages 356–366, 1932.