GARLig: A Fully Automated Tool for Subset Selection of Large Fragment Spaces via a Self-Adaptive Genetic Algorithm

Patrick Pfeffer,[†] Thomas Fober,[‡] Eyke Hüllermeier,[‡] and Gerhard Klebe*[,†]

Department of Pharmaceutical Chemistry, Philipps-University, Marbacher Weg 6, 35032 Marburg, Germany, and, Department of Mathematics and Computer Science, Philipps-University, Hans-Meerwein-Strasse, 35032 Marburg, Germany

Received September 1, 2009

In combinatorial chemistry, molecules are assembled according to combinatorial principles by linking suitable reagents or decorating a given scaffold with appropriate substituents from a large chemical space of starting materials. Often the number of possible combinations greatly exceeds the number feasible to handle by an in-depth in silico approach or even more if it should be experimentally synthesized. Therefore, powerful tools to efficiently enumerate large chemical spaces are required. They can be provided by genetic algorithms, which mimic Darwinian evolution. GARLig (genetic algorithm using reagents to compose ligands) has been developed to perform subset selection in large chemical compound spaces subject to target-specific 3D-scoring criteria. GARLig uses different scoring schemes, such as AutoDock4 Score, GOLDScore, and DrugScore^{CSD}, as fitness functions. Its genetic parameters have been optimized to characterize combinatorial libraries with respect to the binding to various targets of pharmaceutical interest. A large tripeptide library of 20^3 members has been used to profile amino acid frequencies in putative substrates for trypsin, thrombin, factor Xa, and plasmin. A peptidomimetic scaffold assembled from a selection of a 25^3 building block was used to test the performance of the evolutionary algorithm in suggesting potent inhibitors of the enzyme cathepsin D. In a final case study, our program was used to characterize and rank a combinatorial drug-like library comprising 33 750 potential thrombin inhibitors. These case studies demonstrate that GARLig finds experimentally confirmed potent leads by processing a significantly smaller subset of the fully enumerated combinatorial library. Furthermore, the profiles of amino acids computed by the genetic algorithm match the observed amino acid frequencies found by screening peptide libraries in substrate cleavage assays.

INTRODUCTION

A major goal in computer-aided drug design is the automated generation of suitable ligands binding to the target protein under consideration. Because of the rapidly increasing number of novel, structurally characterized proteins, identified as putative targets for drug therapy, there is a demand for faster and more efficient approaches to suggest the most promising drug-like candidates that can easily be synthesized by medicinal chemists. Although there are a vast number of software tools available to design individual ligands or medium-sized compound libraries in a target-specific fashion following combinatorial principles, there is a definite need for efficient tools to virtually screen large combinatorial libraries covering a particular part of chemical space fairly comprehensively.

Several computer-aided ligand design methods have been reported.[1] The most popular de novo design programs from the early 90s are CAVEAT,[2] SPROUT,[3] and LUDI.[4] Since then, many new algorithms have been developed, particularly in the field of combinatorial docking. CombiDOCK[5] extends the well-known program DOCK[6] by linking scaffolds and fragments combinatorially. Boehm et al. extended LUDI using combinatorial principles, and reported on the discovery of nanomolar thrombin inhibitors.[7] FlexX^{C}[8] is an extension of the FlexX program series using an incremental built-up procedure in combinatorial fashion. FlexNovo[9] uses a sequential growth strategy to link chemical fragments taken from a large compound space of starting materials. In this program, the build-up procedure is based on a set of synthesis rules, physicochemical property filters, and the FlexX scoring function. KNOBLE[10] designs novel small molecules by linking molecular fragments to a given core skeleton using simple and feasible chemistry at all levels. Potential candidates of the considered fragments are retrieved from subpockets of proteins exhibiting similar pharmacophoric patterns, as identified by the Cavbase approach.[11] SQUIRREL[12] is a shape-based alignment method that decomposes small molecules into building blocks and compares them to a predefined query structure. The alignment is performed by means of a subgraph matching routine, and the similarity is calculated using a fuzzy pharmacophore function.

Reliable and discriminative subset selection strategies have been proposed using iterative and deterministic strategies[13,14] to avoid combinatorial explosion. Le Bailly de Tilleghem et al. suggested a probabilistic exchange of fragments that proceeds in an iterative manner.[15] The probability of a fragment being exchanged depends on a fitness score that is optimized during the design process.

* To whom correspondence should be addressed. E-mail: klebe@staff.uni-marburg.de.
[†] Department of Pharmaceutical Chemistry.
[‡] Department of Mathematics and Computer Science.

10.1021/ci9003305
Published on Web 08/26/2010

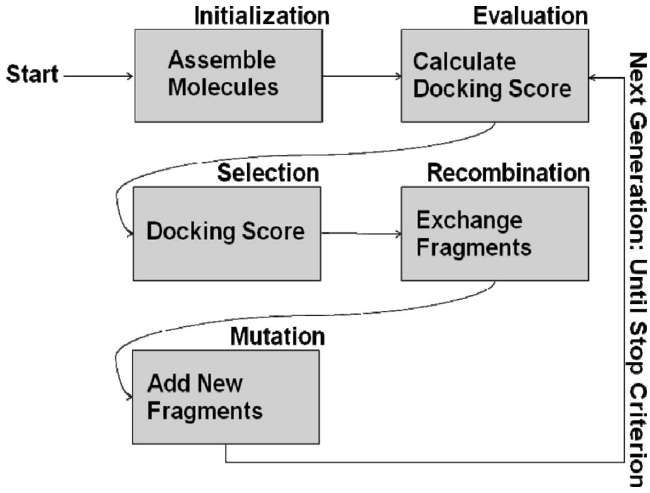

Figure 1. Schematic overview of our genetic algorithm showing the different stages of small molecule generation and evaluation.

Besides the deterministic procedures mentioned above, heuristic optimization algorithms were considered in many ligand design approaches. One of the well-known algorithms to escape local minima on rugged energy landscapes is simulated annealing (SA). SAGE,[16] HARPick,[17] and Focus-2D[18,19] use SA strategies to virtually assemble molecular fragments into complete ligands. The similarity to a given reference molecule serves as the objective function. PICCOLO performs an SA-driven subset selection of compounds with a multiobjective fitness function.[20]

Many methods construct small molecules or specific libraries using evolution-inspired algorithms. TOPAS,[21] Flux,[22] and a new method developed by Schuller et al.[23] are well-known examples. A disadvantage of these approaches is that structural information about a template ligand must be available.

Genetic algorithms (GA) also belong to the class of evolutionary algorithms and have been widely applied in the field of de novo design, for example, for the construction of small molecules, their subsequent structure optimization, or the design of compound libraries.[24−31] However, there are only a few methods, such as SYNOPSIS, ENPDA, ADAPT, and a multiobjective graph evolution method recently developed by Pattichis et al.,[32−36] that use docking scores as fitness criteria in the selection and optimization of putative drug candidate molecules. EAISFD is a method that combines the EA-inventor de novo design engine with the molecular docking tool Surflex-Dock. It has been successfully applied to the p38 MAP kinase to suggest novel ligand scaffolds or optimized lead structures.[36]

In this contribution, we present GARLig (genetic algorithm using reagents to compose ligands), a self-adaptive genetic algorithm that has been tailored to meet the demands of library design. The underlying intention is to combine molecular fragments to assemble substrates or drug-like inhibitors as candidate molecules that are handled as populations in a genetic algorithm. This process is iterated over multiple cycles whereby successive generations of candidate molecules with improved average fitness score are generated. A schematic overview presenting the applied GA is given in Figure 1. In more detail, GARLig optimizes side-chain substituents at a predefined molecular scaffold. The substituents to be attached are represented in terms of SMILES line notation. Special flags in the SMILES string denote attachment points for predefined chemical linking reactions. After assembly of the initial ligand population, their drug-likeness can be estimated according to Lipinski's Rules-of-Five.[37] Subsequently, 3D coordinates are generated using the program CORINA.[38] GARLig is then interfaced to two popular docking engines, AutoDock4[39] and GOLD3.2.[40] The whole workflow can be parametrized by use of a GA configuration file. Essential for the performance of GARLig is a reliable scoring scheme that considers, apart from the implemented scoring functions in AutoDock4 Score and GOLDScore, the scoring function DrugScore^{CSD}.[41] These ranking systems are used to calculate the fitness value during the evolutionary process. Using such a docking score as the fitness criterion keeps the approach independent of a priori knowledge about the possible binding of any ligand to the target protein under investigation. Apart from standard mutation, crossover or selection operators such as the Roulette Wheel Selection or Tournament Selection,[42] an operator named Simulated Binary Crossover[43] has been implemented, providing the genetic algorithm with a self-adaptive feature known from evolutionary strategies.

To understand how GARLig performs under the regime of different objective functions, a parametrization analysis has been performed using the Sequential Parameter Optimization Toolbox (SPOT),[44] implemented in MATLAB (2007a, the MathWorks, Natick, MA). Therefore, an interface between GARLig and MATLAB has been created. This interface supported the crucial parametrization step of the genetic algorithm for different scoring schemes applied to the different target proteins.

Enumerations of combinatorial libraries have been performed using several proteins of particular pharmaceutical interest: the serine proteases trypsin, thrombin, factor Xa, and plasmin, and the aspartyl protease cathepsin D. The obtained results were compared to similar protocols using random search and Monte Carlo sampling algorithms. In the case of the serine proteases, a large tripeptide library of 20^3 entries has been selected to identify those sequences which are the most likely proteinogenic substrates of the different enzymes according to our evolutionary learning process. These results could be compared to experimentally recorded profiles. To our knowledge, this was the first time that a GA docking methodology was successfully applied to profile serine proteases to identify their preferred substrate sequences in a fully automated manner. For the enzyme cathepsin D, the most promising inhibitors have been selected from a peptidomimetic library comprising 25^3 theoretical members.[45] For this example, the performance of GARLig can be compared to the program ADAPT, which also uses DOCKing Scores as a fitness function. Furthermore, experimental data for some members of this library have been reported, which allows us to evaluate the relevance of our computational approach. As a final scenario, to characterize the scope of GARLig, a large drug-like library of 33750 entries has been analyzed with respect to thrombin inhibition. This third example is to some degree special as the compiled library clusters into groups of candidate molecules with very similar substitution pattern. The special composition of this library clearly shows where the limits of a computational design strategy exploiting docking scores as fitness function are found. For all three libraries some experimentally determined

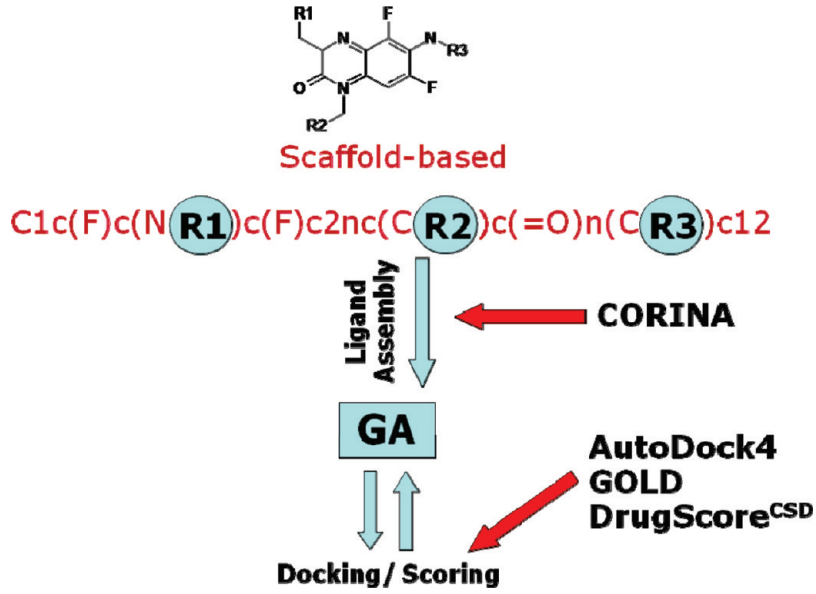

Figure 2. Workflow of GARLig. Scaffold and reagents are provided by the user in SMILES line notation. The ligand assembly phase denotes the stage where a user-defined number of molecules are randomly generated which then act as an initial population for the genetic algorithm. Fitness evaluation is performed via calculation of a docking score. The evaluated compounds are fed back into the core of the workflow, which is a genetic algorithm and mimics Darwinian evolution by applying mutation and crossover operators. Red arrows denote where external tools support our workflow.

reference data have been reported. Therefore in all test examples, enrichments with respect to known substrates or inhibitors could be compared with the most promising candidates found in the final generation of our computational approach.

THEORY

Genetic Algorithms belong to the class of stochastic, population-based search algorithms that mimic Darwinian evolution. The procedure starts with an initial population of individuals proposed to be a solution for a given problem. In the present case, this population is a set of small molecules. The goal of the approach is to find the best suited compounds in their most likely bioactive conformation adopted at the binding site of the target protein under investigation. An iterative process is initiated to detect better fitting individuals. These will more likely survive and succeed in subsequent mutations and crossovers. The iterative GA continues until a predefined termination criterion is met. In the following section, algorithmic details of GARLig will be presented.

Compound Representation. Figure 2 shows a schematic overview of the principal steps performed by GARLig. The user must supply an initial core scaffold in SMILES line notation, along with putative substituents to be attached. They must be given in a canonical way, matching with the branching functionality of the SMILES notation. After the initial population has been established (i.e., 100 molecules randomly assembled from combinatorial solution space), a 3D conformer is generated for each library entry as input for the subsequently applied docking programs.

Fitness Evaluation. GARLig has implemented interfaces to the well-established docking programs GOLD3.2 and AutoDock4. The fitness value to be optimized during a GARLig run is a docking score, which estimates the expected binding affinity of a ligand toward the considered receptor site. Adjustable parameters for the docking step can be defined in the GARLig configuration file. During the GA steps, the obtained docking geometries are evaluated by the scoring functions AutoDock4 Score and GOLDScore and can be rescored by DrugScore^{CSD}, developed in our group.[41]

Breeding. After the initial docking cycle, a new generation of candidate ligands is assembled using the genetic operations *crossover* and *mutation*. GARLig takes advantage of the elitism functionality, also known as the "survival of the fittest" principle. It allows direct copying of the fittest individual into the next generation without subjection to further recombinatorial events. Two selection strategies known as "Tournament Selection" and "Roulette Wheel Selection" have been implemented.[42] Both are based on the assumption that better fitting candidates are more likely to experience a crossover, an event characterized by bond breaking and reformation to assemble the newly selected individuals. Therefore, uniform single- and two-point crossover operators have been implemented to perform reagent exchange between selected molecule partners.[46]

Furthermore, an operator called "Simulated Binary Crossover" (SBX)[43] has been implemented with the idea of allowing for self-adaption in the evolutionary process. Therefore, the reagents are sorted with respect to their chemical similarity based on molecular descriptors appropriate for small reagent probes, such as molecular weight or number of H-bond donors/acceptors.

The crossover operator SBX works as follows:

```
For each Selected Pair of Individuals p_m
and p_n:
  For each Residue Position r:
    μ_m = p_m(r_k), μ_n = p_n(r_k)
    σ = δ(μ_m, μ_n)
    p_m_new(r_k) = random.gauss(μ_m, σ), p_n_new(r_k)
    = random.gauss(μ_n, σ)
```

where *random.gauss* returns a random Gaussian distributed number, μ is the value of a parent's residue r at position k, δ denotes the distance (chemical similarity) between μ_m

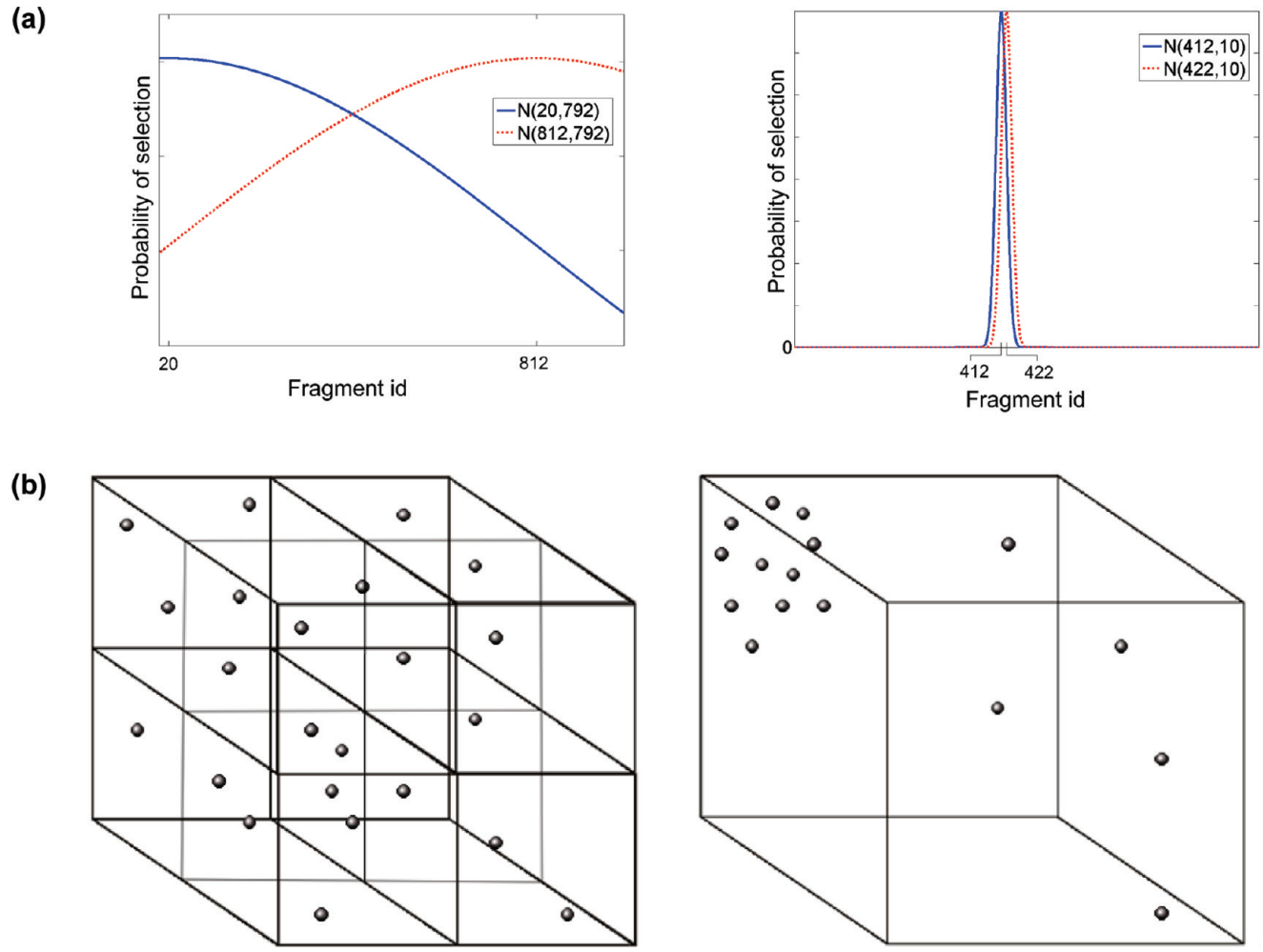

Figure 3. (a) Depiction of the Simulated Binary Crossover (SBX) parameter. Two library decorations are selected for deriving a new decoration depending on the distance δ of the selected decorations. The fragments are ordered according to their chemical similarity, that is, decoration 20 and 812 seem to be dissimilar. A new decoration can be chosen randomly according to two Gaussian probability distributions, where the means are the integers of the selected decorations and standard deviation equals δ. On the left side, the genetic algorithm has not converged yet as a large number of fragments **have high probability** of being selected. On the right side, the algorithm seems to converge to a chemical subspace. New decorations will be more similar than in the first case. In both examples, contracting crossover has the same probability as expanding crossover of avoiding a prematurely converging GA. (b) LHS (left) and simple sampling (right) for three parameters. LHS divides the range of all parameters (here 3 parameters) into i equal sized intervals (here $i = 2$) resulting in a lattice. Points are drawn uniformly from all cubes defined by the lattice. If parameters require integer values, a rounding will be applied afterward. This approach allows a good coverage of the parameter space. For a simple sampling approach, the drawn points might not be optimally distributed, for example, if a region of the parameter space is not covered.

and μ_n, and σ is the standard deviation. This procedure produces more diverse compounds if chemically diverse parents have been selected for mating. However, if parents with chemically closely related substituents are selected, the algorithm will be more likely to converge to a chemically rather uniform subspace (Figure 3a).

Uniform mutations can occur at each gene position of all chromosomes after crossover, whereby the frequency of mutating genes is adjusted by a probability parameter.

Parameter Optimization. The described algorithm has a few adjustable parameters which have to be optimized, as the results obtained from the genetic algorithm will strongly depend on this parameter selection. A variety of methods can be applied to optimize external parameters. In the present case, a strategy for multiple parameter optimizations is required. We have chosen the SPOT method.[44] SPOT follows a semiautomatic method that tries to keep the computational costs for determining an improved parametrization low by internally fitting a model Y that is able to predict the quality of the algorithm with respect to the docking scores for a certain so far untested parametrization of the algorithm. Thus, this method belongs to the group of respond-surface approaches that have to fit a response. For this, a regression model with polynomials of order 2 and a Gaussian correlation function will be used. Assuming that the algorithm has k parameters, this response requires fitting of $q = 0.5(k^2 + 3k + 2)$ variables.[42] Therefore, it is necessary to have at least q parametrizations of the algorithm with the dependent output values defined by the docking score. In a first step, the q variables have to be defined by the user as a set of preselected intervals, each specifying the allowed values for a parameter (e.g., [1, 50] for the population size) called the *region of interest* (ROI). After defining the ROI, a set of parameters is generated to be used in the genetic algorithm. In the beginning of this procedure, *latin hypercube sampling* (LHS) is performed to generate the required q parametrizations. For this the hypercube, defined by ROI, is divided into a set of subhypercubes and for each a predefined number of parametrizations is sampled. This has the advantage that the selected variables are uniformly scattered over

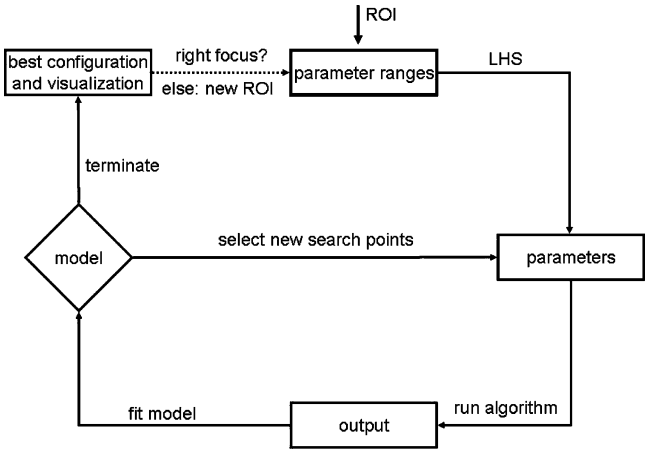

Figure 4. Parameter optimization performed by SPOT demands a region of interest (ROI) to be given by the user. Here, boundaries must be specified for the parameters being optimized. Subsequently, k parametrizations are generated and improved in a recursive procedure. At the end of the procedure, the best parametrization for a scoring function under investigation is suggested.

the whole ROI, unlike a simple sampling scheme (Figure 3b). After creation of the initial parametrizations, the SPOT-loop illustrated in Figure 4 is accomplished. In each iteration, the selected parametrizations are evaluated by running the GA with this set of values to obtain the fitness scores. Subsequently, SPOT derives a regression model using the optimized parametrizations and corresponding fitness values. The subsequently obtained model is further used to generate novel parametrizations x for the following purposes: (1) improving the regression model and (2) finding better parametrizations.

The second (and most important) purpose is fulfilled once the improvement 1 is maximized for x:

$$I(x) = \begin{matrix} f(x) - f^* & \text{if} & f^* < f(x) \\ 0 & & \text{else} \end{matrix} \quad (1)$$

where f^* denotes the best value found so far. However, the exact value $I(x)$ will not be known a priori, so SPOT has to use the expected improvement based on the model Y.

Standard Random Search. This method is independent of the computed generation cycles and randomly assembles a user-defined number of candidate ligands and evaluates them via docking.

Monte Carlo Sampling Algorithm. A stochastic search without using crossover or mutation events has been applied for comparison. The algorithm runs for a predefined number of generations and randomly assembles molecules. The best evaluated individual from each generation is kept and is only replaced once a better fitting molecule is produced.

PROGRAMS, DATA SETS, AND MATERIALS

Characterization of a Tripeptide Library with 20^3 Entries for Trypsin, Thrombin, Factor Xa, and Plasmin. GARLig was used to suggest the preferred tripeptide amino acid profiles as potential substrates to be cleaved in the respective serine proteases. These sequences are suggested in the final generation of our GA. The obtained profiles can be mapped against experimentally collected data obtained from an enzymatic assay recording peptide cleavage by use of an attached fluorescence probe.[47] It is assumed that prior to efficient cleavage, selective and potent binding of the peptide substrate has to be achieved. Two main objectives were addressed in this application: (1) Is the GA able to identify only those members of a fully enumerated substrate library that are known to be preferentially cleaved by the individual serine proteases? (2) Is a docking score sufficient to discriminate between substrates and nonsubstrates?

The results obtained by GARLig were compared with those produced by a Monte Carlo Sampling and a standard random search. All three methods were intended to enumerate only 7.5% of a 20^3-entry tripeptide library (out of 8000 possible substrates, only 600 had their fitness evaluated by our GA). The study has been performed using the geometries from crystal structures of trypsin, thrombin, factor Xa, and plasmin (PDB codes 1k1p, 1ype, 2w26, and 1bui) and a tripeptide scaffold enumerating all 20 proteinogenic amino acid residues at each position P1−P3. To investigate the sampling properties of our GA and the reliability of the applied scoring functions with respect to the targets, an exemplary parametrization study has been carried out on trypsin using the MATLAB implementation of SPOT. Five parameters of the GA were selected for optimization within the following regions of interest (ROI):

- population size of 50−150 individuals per generation
- mutation probability ranging from 0 to 1
- crossover probability ranging from 0 to 1
- crossover type is one-point- and two-point crossover, SBX
- selection type is Roulette Wheel and Tournament Selection

In the following, $q = 0.5(5^2 + 3 \times 5 + 2) = 21$ parameter variations were applied to determine the initial regression model, which was improved in four sequential steps each involving three further parametrizations. The parameters suggested by SPOT complemented the setting of GARLig and docking parameters in the input regarding the best performing parameters (Bst) with respect to the selected scoring function.

Prior to the docking, the assembled tripeptides had to be converted from SMILES to 3D coordinates in mol2 format[48] (CORINA) or pdbqt format (AutoDockTools).[49] Docking geometries were generated using AutoDock4 and GOLD; DrugScore^{CSD} was applied to rescore the GOLD solutions. The set of "fast" docking parameters suggested by CCDC[50] has been applied in all GOLD docking runs. For each assembled tripeptide, 10 geometries were computed assuming standard protonation states (i.e., carboxylate groups deprotonated, aliphatic amines and amidino/guanidino groups protonated). The protonation states of the protein residues at pH 8.0 were taken as predicted by MOE.[51]

Similarly, parameters suggested for high-throughput docking with AutoDock4 were applied, and 10 geometries were computed for each ligand. A total of 150 000 energy evaluations were considered using a population size of 150 individuals and 27 000 generations in AutoDock's Lamarckian GA.

To efficiently compute the docking, the individual runs were distributed on a 50-node compute-cluster by the Condor Queuing System.[52]

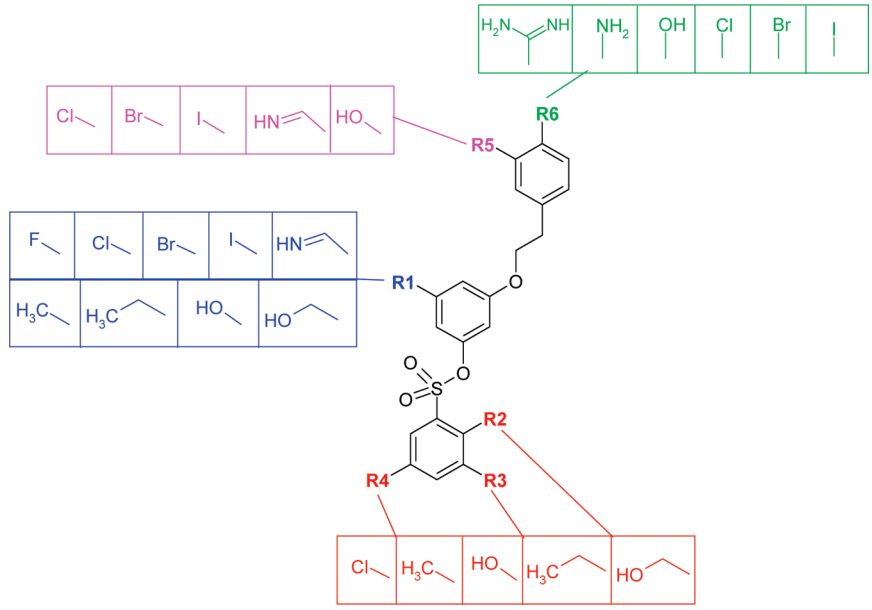

Figure 5. Scaffold of sulfonic acid ester derivatives as thrombin inhibitors. Different chemical groups have been attached to the six corresponding substitution sites.

Enumeration of a Peptidomimetic Library of 25^3 Inhibitors for Cathepsin D. In a second scenario, GARLig was tested on a peptidomimetic library comprising 15625 entries intended to bind to the aspartyl protease cathepsin D (PDB code 1lyb). This data set has already been evaluated using the program ADAPT.[35] GARLig has been applied similarly, thus allowing for a direct comparison of the GAs implemented in the two programs, as both use docking scores as objective function. Since for some entries experimentally determined affinity data have been published, enrichment rates with respect to known actives could be computed in the final generation of the optimization procedure. Also here GARLig results are compared to a Monte Carlo Sampling and a random search. As the reliability of scoring functions varies with the target protein, a SPOT parametrization study has been carried out. To allow for a direct comparison with ADAPT, we parametrized GARLig comparably to the protocol used in the ADAPT study. The following region of interest has been applied:

- population size of 10−50 individuals per generation
- mutation probability ranging from 0 to 1
- crossover probability ranging from 0 to 1
- crossover type is one-point- and two-point crossover, SBX
- selection type is Roulette Wheel- and Tournament Selection

In total, only 3.2% of the conceivable product space (only 600 compounds) was evaluated using each of the scoring functions under investigation. With respect to docking, AutoDock4 was applied with a higher number of energy evaluations (10^6) to improve its accuracy. GOLD was applied with standard settings as no specific parametrization has been suggested for aspartic protease application. Protonation states were assigned as described above.

Enumeration of a Library of 33 750 Sulfonic Acid Esters toward Thrombin. In a third scenario, GARLig was tested on a compound library of 33 750 drug-like inhibitors with a sulfonic acid ester scaffold with respect to thrombin binding[52] (Figure 5). Since experimental binding data are available for twelve library entries,[52] a crude enrichment of active compounds in the final generation of the optimization procedure could be computed. All GA parameters previously optimized for the docking of the tripeptide substrates to thrombin were also applied in this study.

Again, only 3.2% of the fully enumerable product space (only 1080 compounds) was evaluated. GOLDScore was used as the sole objective function because it showed the best performance in the tripeptide study. GOLD was used in the "fast" docking setup and the protonation state assignment was handled as mentioned above.

RESULTS AND DISCUSSION

GARLig Applied to a Tripeptide Substrate Library toward Trypsin, Thrombin, Factor Xa, and Plasmin. Figure 6 shows the results obtained by SPOT, which was used to study the influence of different fitness functions and parameters on our GA. It depicts the mean square error found by the regression model with respect to the combination of two optimized parameters. Small mean square errors of the regression model denote high reliability of the suggested parameter set. Closer inspection of the iso-surfaces suggests that minima of mean square errors are located at similar positions with respect to $\text{DrugScore}^{\text{CSD}}$ and GOLDScore. This overall agreement is achieved for the minima in the crossover type/crossover probability, selection type/crossover type and mutation probability/population size diagram. The pronounced agreement in the regression models suggests similar parametrizing of the GA for the different scoring functions. The only exception was detected for a GARLig run using AutoDock4 Score as objective function. Here, the parameters to be optimized appear to be mutually independent and thus prevent any optimization. This behavior can probably be explained by the reduced number of energy evaluations considered in the docking setup.

Table 1 lists the best parametrizations for each scoring scheme. It is remarkable that similar GARLig parametrizations are suggested using GOLDScore and $\text{DrugScore}^{\text{CSD}}$.

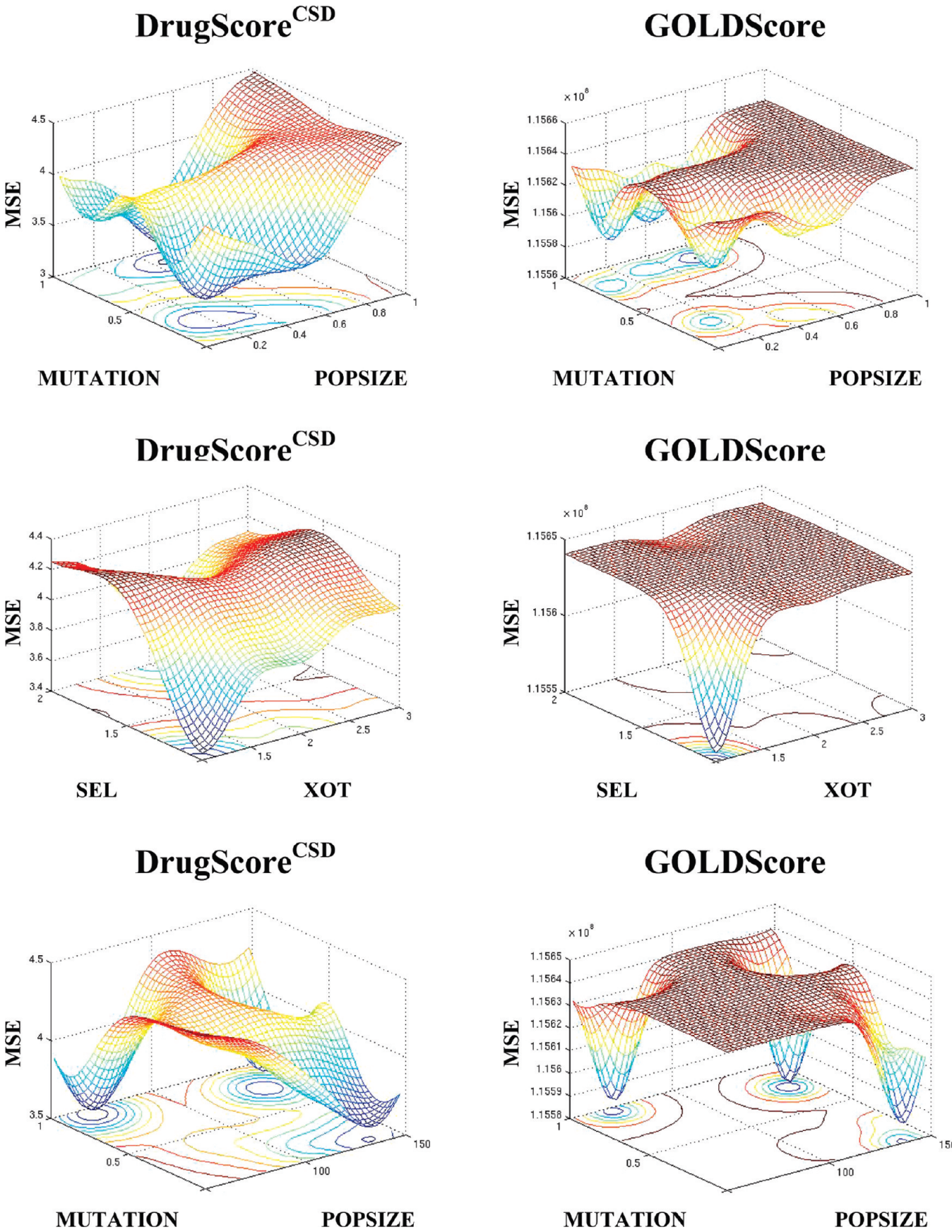

Figure 6. Energy landscapes as a result of the SPOT parameter determination. Results are shown for the two scoring functions GOLDScore and DrugScore^{CSD}. The plots show the mean square error (MSE) of the regression model computed by SPOT as a function of different variables such as mutation probability (MUTATION), population size (POPSIZE), selection type (SEL), crossover type (XOT), and crossover probability (XO). Convincing similarity can be detected between the landscapes of the different scoring functions as both of them have the same error minima of the regression model. This leads to the conclusion that a similar parametrization can be chosen for different fitness functions applied to the GA.

Table 1. Best Parametrization for Each Scoring Scheme[a]

scoring function	best score	POPSIZE[b]	MUTATION[c]	XO[d]	XOT[e]	SEL[f]
GOLDScore	48.98	**60**	**0.05**	**0.71**	**1**	**2**
DrugScore^{CSD}	−147788	**60**	**0.05**	**0.71**	**1**	**2**
AutoDock4 Score	−7.09	68	0.06	0.77	2	2

[a] GOLDScore and DrugScore^{CSD} can be equally parameterized in the serine proteases study. [b] Population size (POPSIZE). [c] Mutation probability (MUTATION). [d] Crossover probability (XO). [e] Crossover type (XOT). [f] Selection type (SEL) for each scoring function in use. A GARLig run can be started with a mutation probability ≤20% and a crossover probability ≥59% in all cases.

Interestingly, mutation and crossover probability parameters must be set to ≤20% and ≥59%, respectively, for all applied scoring functions. These probability thresholds are consistent with values suggested in GA literature.[42]

Figure 7 shows the convergence of the GARLig runs using different scoring functions as fitness criteria. In each case, the best parametrization setup suggested by SPOT was applied. The results of the GA runs were then compared to a standard random search and a Monte Carlo Sampling procedure. As expected, GARLig outperforms the other two strategies, in particular resulting in a much faster convergence to better fitness values.

Table 2. Results for Trypsin Using GARLig Together with an Optimal Set of Parameters for Different Scoring Schemes

scoring function	substrate	scoring rank final generation
DrugScore^{CSD}	P−H−R	2
GOLDScore	K−H−R	1
AutoDock4 Score	Q−A−R	4

Table 2 shows the top-scored tripeptide sequences of substrates for trypsin in the final generation of the GA with respect to the different scoring schemes. For each scoring function, amino acids that are experimentally known to be preferred at the P1−P3 positions are suggested among the four top-scored solutions. This result is obtained by setting the docking parameters to "faster and less accurate". Obviously, it still identifies sequences known to be favorable trypsin substrates. Since GOLDScore was able to place the known trypsin substrate K−H−R on rank 1,[47] this function was considered in the subsequent serine protease studies.

Figure 8 shows that GARLig suggests in the final generation amino acids at the different positions P1, P2, and P3 that are actually known to occur in productive substrate sequences of the distinct proteases. The computed amino acid profiles are in good agreement with the experimentally determined results.[47] The position P1 shows high preference

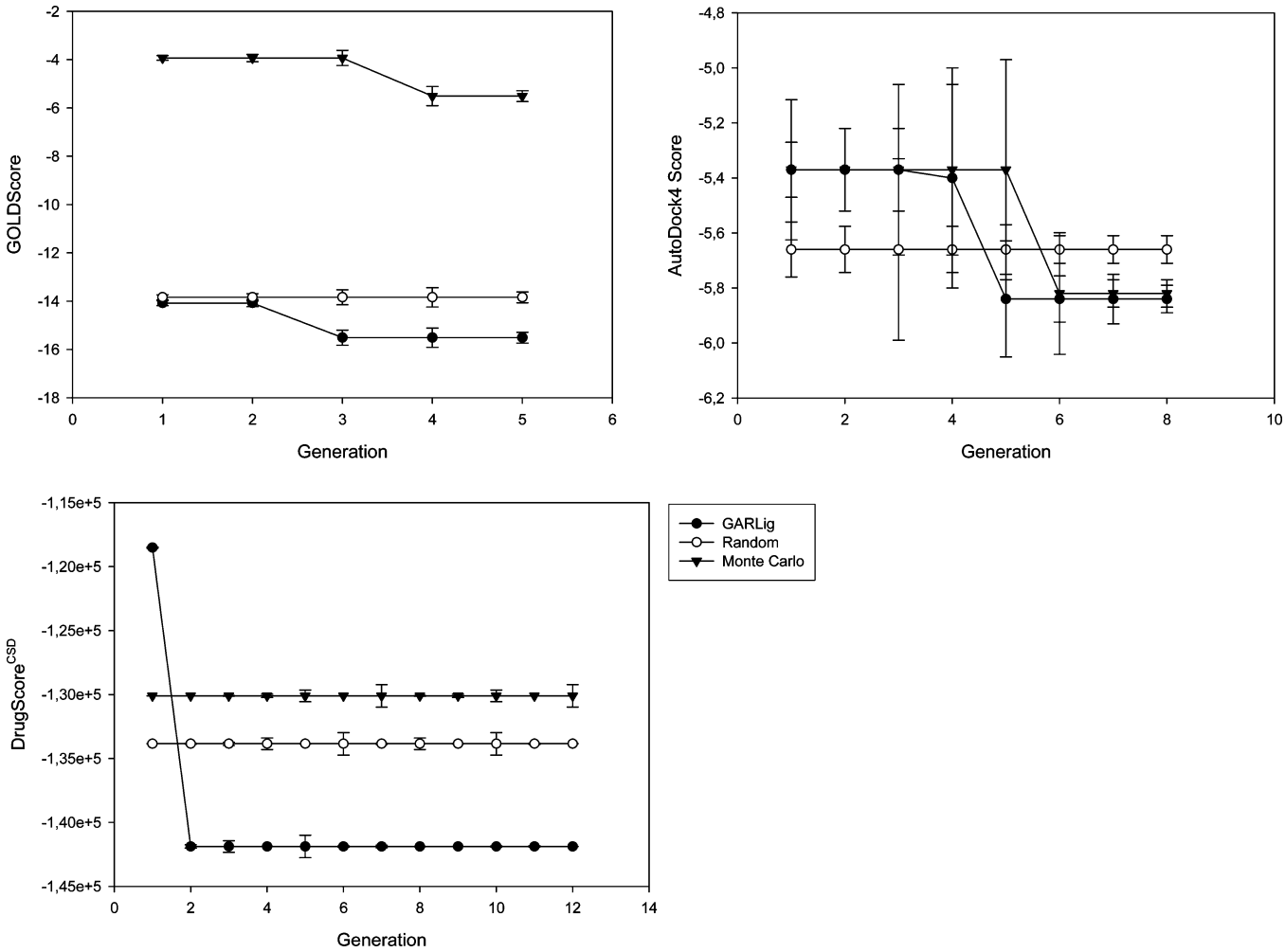

Figure 7. Plots show the docking score (either DrugScore^{CSD} or AutoDock4 Score or GOLDScore in their respective units) as a function of the generation (the score of the best individual of a generation is depicted) of the genetic algorithm. GARLig has been run three times on trypsin with the best parameters determined by SPOT. The GA results are compared to a Monte Carlo Sampling and a standard random search. The error bars show the standard deviation of the fitness values computed in the three runs and the points show the fitness value as a function of the generation cycle.

Figure 8. GARLig results in the tripeptide substrate study on trypsin, factor Xa, thrombin, and plasmin. For each protein target, the preferred profile of experimentally observed proteinogenic amino acids is recorded across the positions P1−P3 (data from ref 46). Experimentally obtained amino acid frequencies (red) are next to the computed ones (blue). Red numbers denote the number of amino acids which remained at the positions P1, P2, and P3 in the last generation of the GA.

for lysine and arginine in the case of trypsin. Additionally, tyrosine is proposed by our calculations to be a potential S1 anchor group. At position P2, the experimentally detected aspartate, valine and glutamine are not identified by our simulation, but frequently occurring residues such as tyrosine and proline are well captured. Considering P3, there is a fair agreement between experimental and computational results except for proline, which is frequently detected by our calculations but not preferred in the cleavage experiments. For factor Xa, our study shows a good match between the computational and experimental profiles, particularly for the P1 and P2 positions. At P1, arginine and lysine are correctly predicted as preferred; at P2, glycine, phenylalanine, tyrosine, and serine match with the experimentally most frequently found amino acids. In the case of thrombin, good agreement between experimental and computational results is obtained for P1 and P2. Interestingly, the amino acid phenylalanine

Table 3. Top Scoring Substrate Sequences Are Shown for All Studied Serine Proteases Using GOLDScore as a Fitness Function

target protein	substrate
thrombin	DF−P−R
thrombin	W−L−K
factor Xa	W−G−K
plasmin	H−F−W

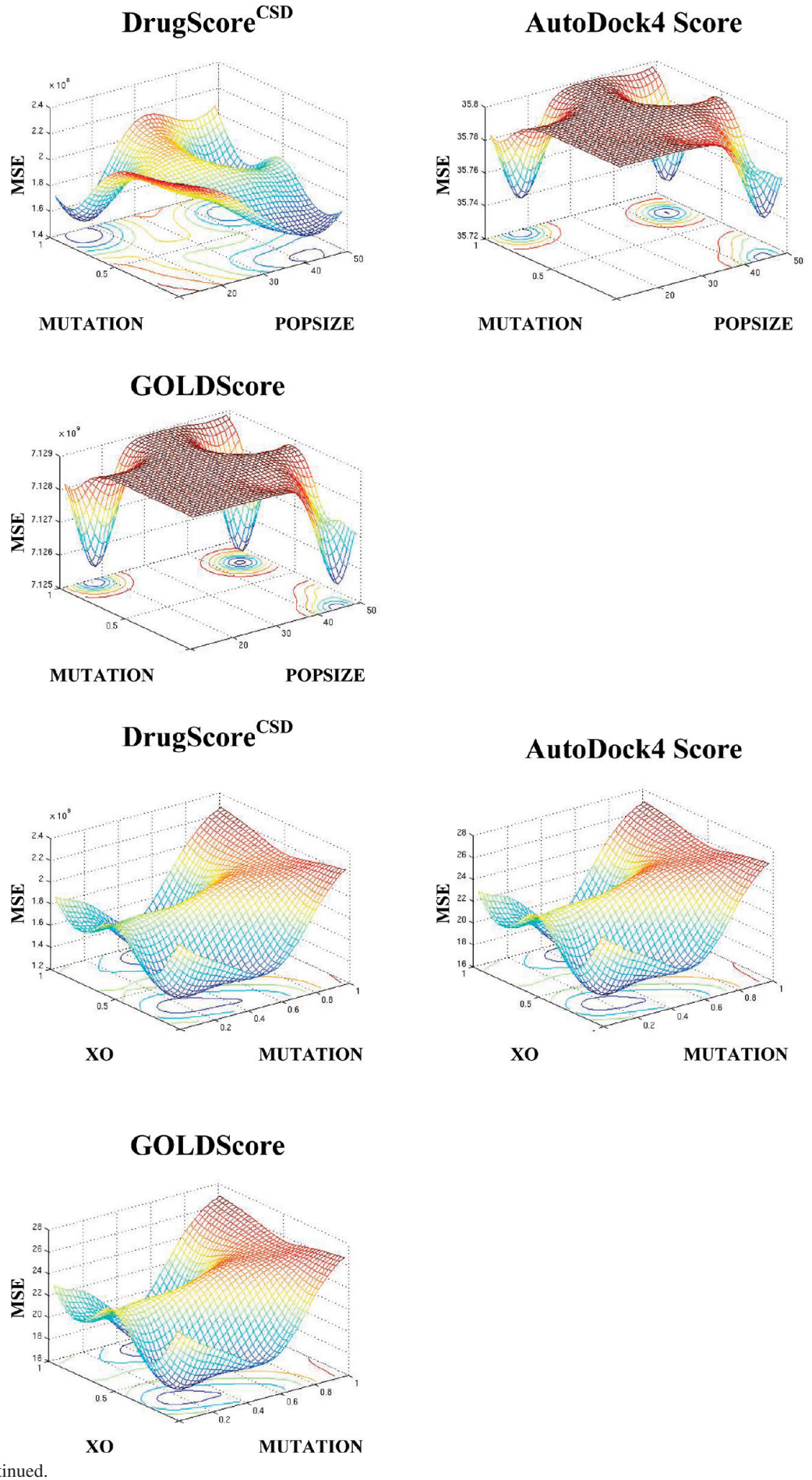

Figure 9. Continued.

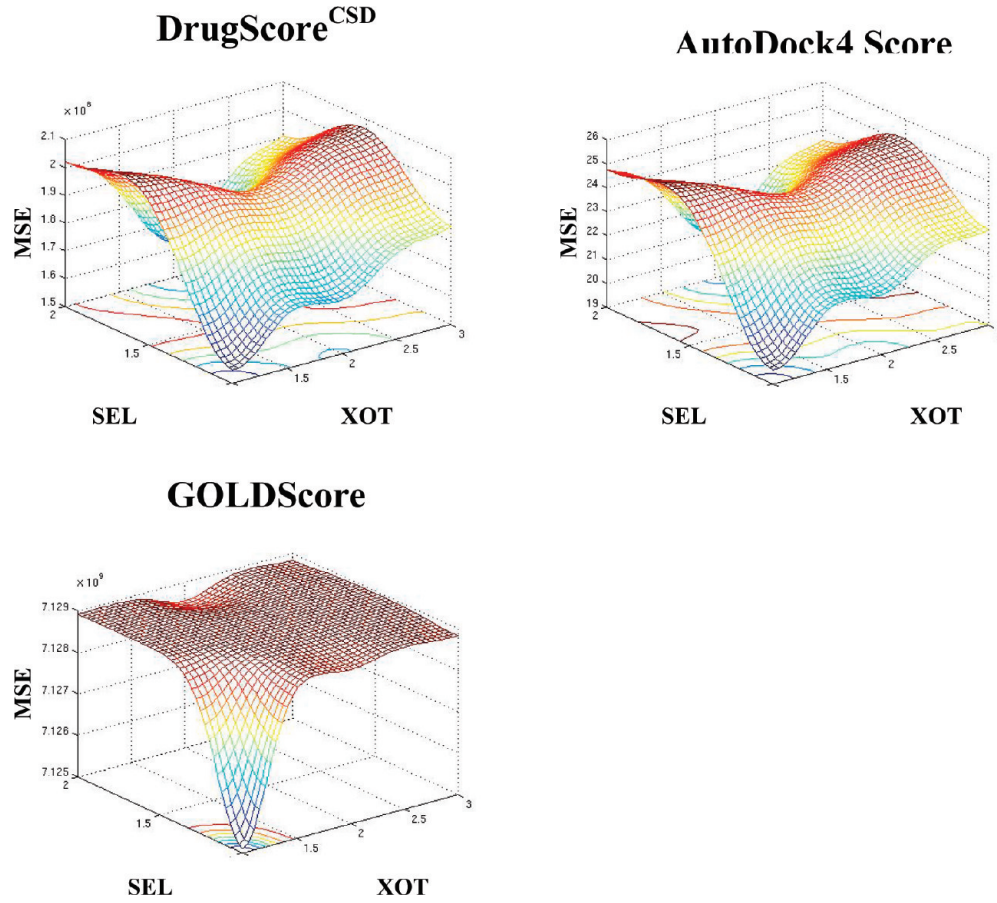

Figure 9. Energy landscapes as a result of the SPOT parameter determination in the cathepsin D study. The plots show the mean square error (MSE) of the regression model computed by SPOT as a function of the mutation probability (MUTATION), population size (POPSIZE), selection type (SEL), crossover type (XOT), and crossover probability (XO). Again, a convincing agreement can be seen considering the landscapes of the different scoring functions as all of them have the same error minima of the regression model. This leads to the conclusion that a similar parametrization can be chosen for different fitness functions applied to the GA.

is not recognized at P3 in our results. As the substrate Phe-Pro-Arg is a preferred ligand for this protease, if the Phe residue is supplied in D-configuration, we decided to perform an additional calculation. However, this time, the P3 amino acid was docked into the protein using the latter configuration. Remarkably, the prominent D-Phe-Pro-Arg substrate, which was used as one of the first peptide-like references for the development of thrombin inhibitors, is found on the first scoring rank. A cleavage of this substrate was proven experimentally.[54] Substrates with a D-phenylalanine are able to interact with Trp215, Ile174, and Leu99 in the P3-subpocket of thrombin, whereas this residue remains solvent-exposed when applied in its L-configuration.[53,55] The results collected from the plasmin study are overall convincing. The most frequent P1 amino acids such as lysine and arginine are recognized by our approach with high priority. P2 amino acids like phenylalanine, tryptophan, and tyrosine and P3 residues like glutamine are in good agreement with the experimental cleavage preferences. Finally, Table 3 shows the substrate sequences found on rank 1 for the four serine proteases in the final generation of our GA.

At the P1 position, all calculated tripeptide substrates show preferred accumulation of lysine and arginine. These residues are known to be good cleavage sequences for trypsin-like serine proteases which all exhibit an aspartic acid residue in the S1 pocket. Furthermore, the algorithm suggests histidine, tyrosine and tryptophan at P1, which were not reported to occur in the best substrates.[47] However, several crystal structures have been reported which show that such residues can be accommodated in the S1 pocket of these proteases.[56−58] Even though our approach does not detect all experimentally observed residues, we believe it can help to prioritize a fraction of a fully enumerated library for experimental evaluation. Particularly in the field of protease inhibitor design, strategies starting with peptide-like mimetics as first leads are still very popular and GARLig can be very supportive in selecting promising initial sequences for this approach.

Cathepsin D Library. Analysis of the optimized parametrization (Figure 9) suggests again an overall agreement of the estimated minimal errors among the applied scoring functions. This time, even for the AutoDock4 Score function, the parameters optimized show mutual interdependence, possibly due to an increased number of energy evaluations in the docking setup of AutoDock4.

Table 4 shows that also in this case, similar parameters can be used for the GA. Among the different scoring schemes, GARLig performed better using the Tournament Selection and the self-adaptive Simulated Binary Crossover parameter. Either GOLDScore or DrugScore^{CSD} can be applied with the same set of parameters. Mutation and crossover probabilities must be generally set to $\leq$16% and $\geq$52%, respectively. Figure 10 shows the results of the different GARLig runs compared to a standard random search and a Monte Carlo Sampling. Although experimental binding data are only available for 9 highly potent entries out of the total library of 15 625 compounds (Figure 11a), the GA is able to identify some of these compounds as

Table 4. Best Parametrization for Each Scoring Scheme[a]

scoring function	best score	POPSIZE[b]	MUTATION[c]	XO[d]	XOT[e]	SEL[f]
GOLDScore	82.14	**29**	**0.09**	**0.98**	3	**1**
DrugScore^{CSD}	−201447	**29**	**0.09**	**0.98**	**3**	**1**
AutoDock4 Score	−12.22	35	0.11	0.63	2	1

[a] In the cathepsin D study, GARLig can be parameterized equally for GOLDScore and DrugScore^{CSD}. [b] Population size (POPSIZE). [c] Mutation probability (MUTATION). [d] Crossover probability (XO). [e] Crossover type (XOT). [f] Selection type (SELTYPE). All GARLig runs can be started with a mutation probability ≤16% and a crossover probability ≥52%. Furthermore, there is an agreement in using Tournament Selection and the self-adaptive Simulated Binary Crossover parameter.

hits in the last generation. In all scenarios, the chemical variability of substituents suggested by the last cycle is dramatically reduced compared to the initial vast chemical space (Table 5). Our best performing setup was a GOLDScore run, which converges to two known binders and 10 additional entries from the total combinatorial library of 15 625 entries. Figure 11b shows the substituents which remained in the last generation and Figure 11c depicts the interaction scheme observed for the compounds placed on rank 1 and 2. The top scoring compound (**cat7**) is known to inhibit cathepsin D at 5.8 nM, whereas, unfortunately, affinity data are not published for the second compound. However, comparing the key interactions performed by both compounds suggests that the second best ranked compound should also be a high affinity binder. The GA runs using DrugScore^{CSD} also converged to a small subset of 12 library entries; however, they did not render one of the experimentally confirmed binders. As mentioned, unfortunately experimental data are reported for only 9 high affinity binders, thus there might be a number of reasonably potent ligands among the suggested candidates. Using AutoDock4 Score as an objective function, 4 of the experimentally characterized binders are listed among the 64 entries suggested in the final GA generation.

All of our GA runs converge much faster and create significantly smaller libraries in the final generation compared to the previously developed ADAPT program. In the cathepsin D experiment, we used the same peptidomimetic library and an identical number of GA evaluations. The GOLDScore run converged within 17 generations to the final 12 entries comprising two known highly potent binders. The AutoDock4 run converged within 10 generations to 64 entries, comprising four of the known binders. To select the same number of known hits, ADAPT converges only after 50 generations, showing them in a subset of 392 compounds ($8 \times 7 \times 7$).

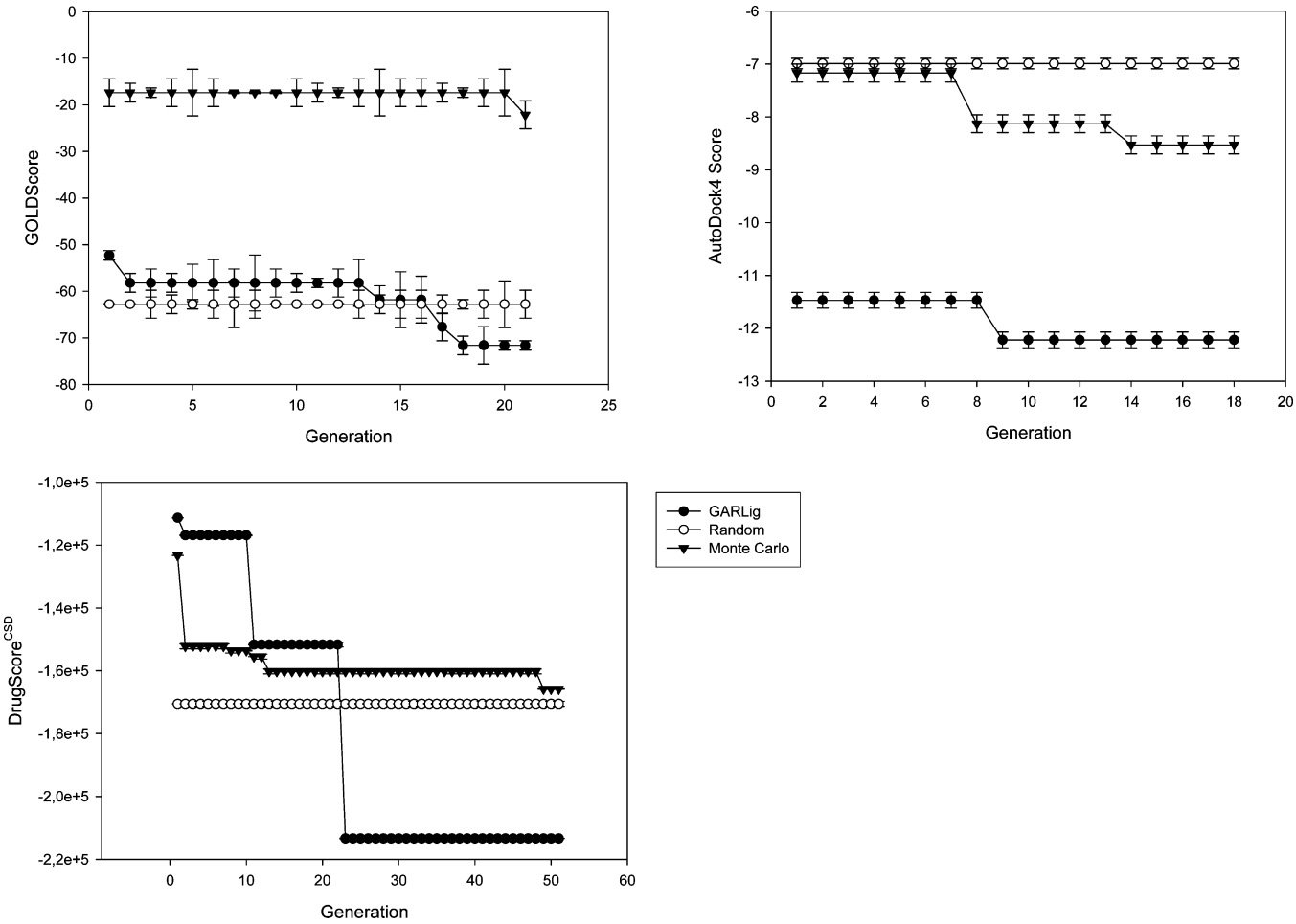

Figure 10. Plots show the docking score (either DrugScore^{CSD} or AutoDock4 Score or GOLDScore in their respective units) as a function of the generation (the score of the best individual of a generation is depicted) of the genetic algorithm. GARLig has been run three times on cathepsin D with the best parameters determined by SPOT. The GA results are compared to a Monte Carlo Sampling and a standard random search. The error bars show the standard deviation of the fitness values computed in the three runs and the points show the fitness value as a function of the generation cycle.

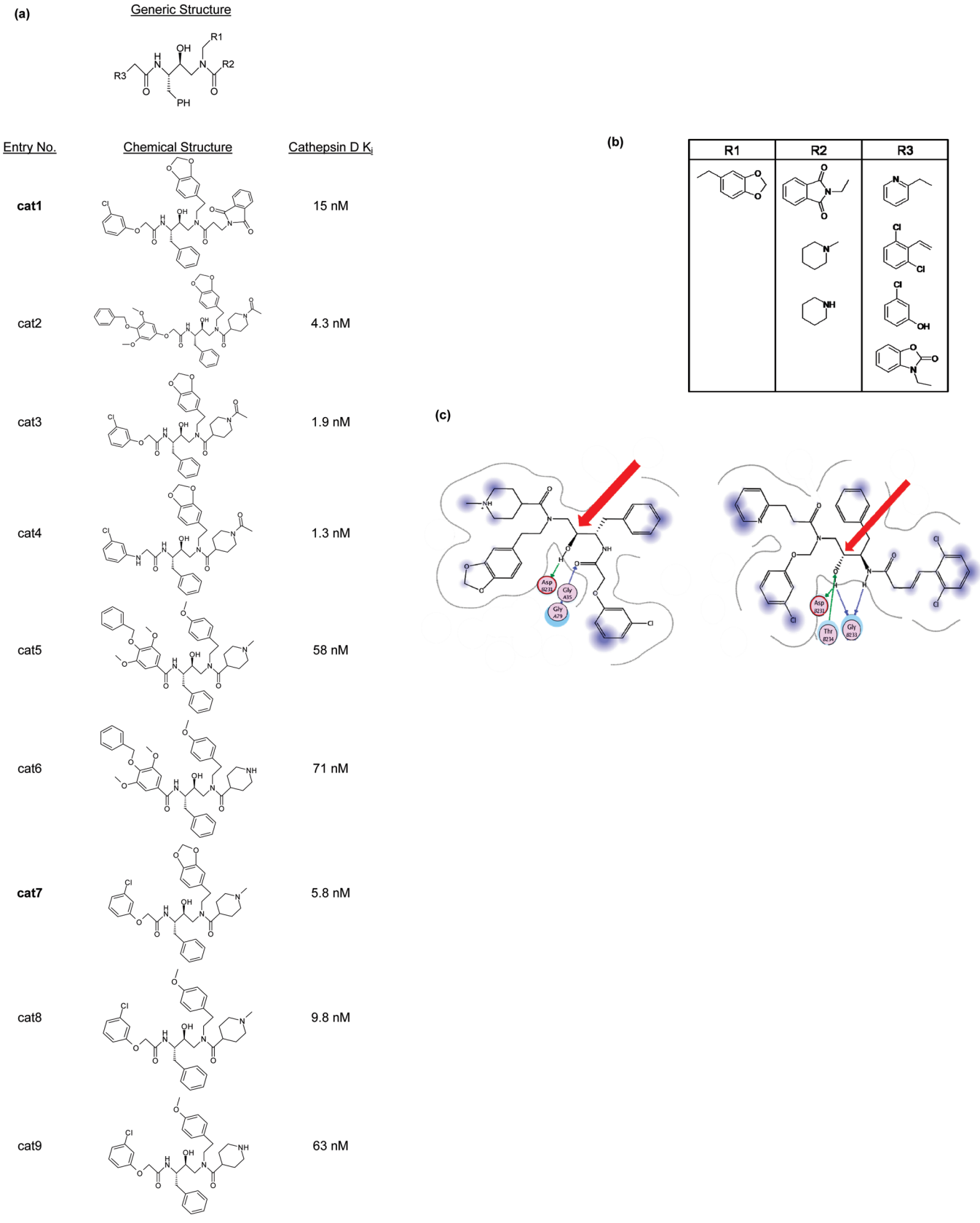

Figure 11. (a) Experimental binding data and chemical structures of 9 highly potent members of the cathepsin D library. (b) The fragment collection which remained in the last generation of the GARLig run at positions R1−R3. The remaining chemical space shown here (3 × 1 × 4) contains two known cathepsin D binders (**cat1** and **cat7**, marked in bold in figure 11a). (c) The top scoring compound proposed by GARLig (left) is a cathepsin D binder known to inhibit at 5.8 nM (**cat7**). The image on the right shows the second ranked compound. The key interaction between the hydroxyl group of this inhibitor and the Asp 231 residue of the protein (marked by the red arrow) and rather hydrophobic interactions (blue spheres) performed by the side chains appear in both results.

Table 5. Results on Cathepsin D Using GARLig Together with an Optimal Set of Parameters for Different Scoring Schemes

scoring function	library size last generation (R1−R3)	known binders in library
GOLDScore	3 × 1 × 4	2
DrugScore^{CSD}	3 × 2 × 2	0
AutoDock4 Score	4 × 4 × 4	4

Enumeration of a Library of Sulfonic Acid Ester Inhibitors for Serine Proteases. For the last example, the results of our GA are less convincing and clearly show the limitations of such an approach. The initial library of 33 750 entries could only be reduced to about a third, still comprising 11250 compounds. At least the 12 binders reported as potent thrombin inhibitors in the literature are detected among them (Figure 12a). Seeking for an explanation for the limited

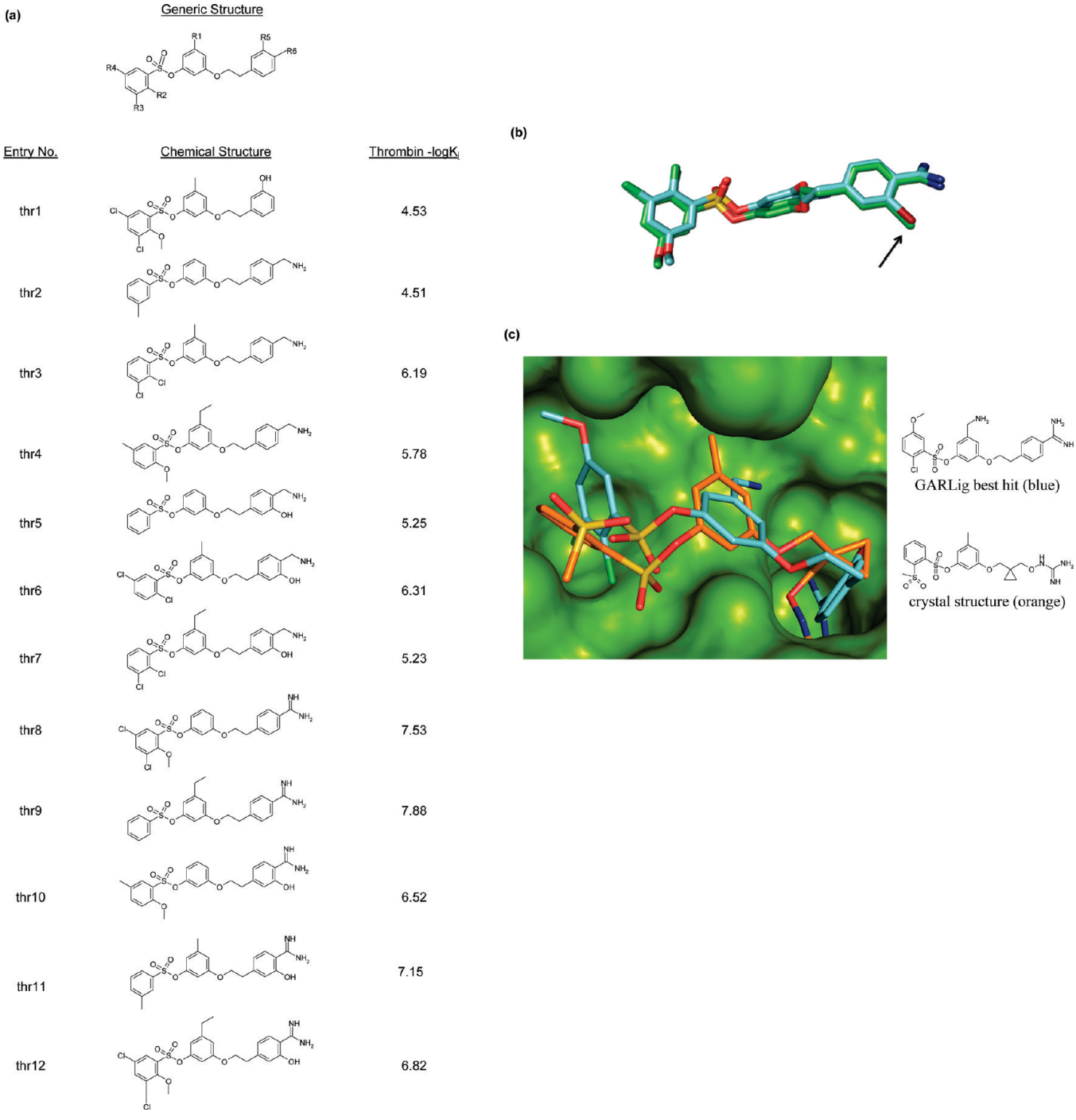

Figure 12. (a) Experimental binding data and chemical structures of 12 highly potent thrombin inhibitors in the sulfonic acid ester library, which were also retrieved by GARLig. (b) Superposition of the GOLD docking solutions of two very similar library entries as generated in a GARLig run. The arrow points to a position where the substituents differ among the superimposed inhibitors. The compound with carbon atoms colored in green (GOLDScore: 66.69) contains a chlorine atom whereas the compound with carbon atoms in cyan (GOLDScore = 66.47) contains a bromine atom. The high similarity of the docking geometries leads to decision problems in the selection step of the genetic algorithm. (c) Superposition of the best scored library member found by GARLig (carbons in blue) with a similar sulfonic acid ester inhibitor cocrystallized with thrombin (carbons in orange, PDB code 1t4u).

ability to reduce the total size of the initial library with our GA, we detected that all generated docking solutions obtained rather similar GOLDscores. Thus, a sufficient discrimination among the various candidate ligands is not successfully achieved. On the one hand, this could indicate the relevance and reliability of the docking solutions and a rather target-tailored choice of the building blocks to assemble the thrombin library. On the other hand, GA parameters such as, for example, Tournament Selection, run into decision problems if docking scores are not sufficiently discriminating among a generation of individual compounds. The computed per-atom contribution to the docking score of a bromine or chlorine atom placed at closely related positions, for example, differ only slightly, much too little to sufficiently guide the GA with respect to scoring differences. Therefore, the selection between the two derivatives by the GA will remain arbitrary and must be rather inefficient (Figure 12b). Consequently, both alternative substituents in the library will be progressed to the last generation of the GA, producing a library with no or only a slight reduction in chemical diversity. This explains why we still find a huge number of equally scored library entries after the last optimization step and our initial library is only reduced in size to one-third. In conclusion, many chemical groups selected as putative substituents at the central scaffold are appropriate as potential interaction partners with the target protein. Figure 12c shows the best ranked compound generated by GARLig superimposed with the crystal structure of a related sulfonic acid ester derivative (PDB code 1t4u). The docking geometry of this best-ranked library candidate (no experimental inhibition data available) is in fair agreement with the experimentally determined binding mode of the related compound.

CONCLUSIONS

GARLig, a genetic algorithm to support the design of combinatorial libraries with self-adaptive features is presented. It evaluates AutoDock4 Score, GOLDScore, and DrugScore^{CSD} as fitness functions. The use of docking scores as fitness criteria can be considered controversial with respect to library design. As a major advantage, such calculations can be initiated without requiring a priori information about experimentally characterized ligands previously reported to bind to the target protein under consideration. As major disadvantage, the computational complexity of the multiple docking steps has to be regarded in the context of the known limitations of our currently applied scoring functions.

The program has been applied to several proteases of pharmaceutical interest: trypsin, thrombin, factor Xa, plasmin, and cathepsin D. In all enumerations, parameter optimization using the tool SPOT was performed prior to the actual GARLig runs. Most importantly, similar parameters were found to be optimal in each case, independent of the applied fitness function and the considered biological target. In all cases, GARLig performs better compared to random search or simple Monte Carlo Sampling. Based on the optimized parameters, GARLig was able to predict tripeptide substrate profiles that correctly render the experimentally observed preferred amino acids at the different subsites with respect to four distinct serine proteases. These results suggest GARLig as very promising tool to efficiently enumerate both proteinogenic and nonproteinogenic substrate profiles for other protease targets of pharmacological relevance. Such profiles can be extremely valuable as first ideas for the design of substrate−analogue inhibitors, a concept that has been applied quite successfully over the last decades in protease inhibitor research.

Our GA succeeds in retrieving from large putative libraries the correctly substituted entries that are experimentally characterized as potent binders of the serine protease thrombin or the aspartic protease cathepsin D. They are found as hits on high scoring ranks among a strongly reduced subset of only 7.5% and 3.2% of all possible entries of the total library. Compared to ADAPT, GARLig shows much faster convergence and better enrichments of known binders in the final generation. We think this faster convergence leading to a smaller chemical space can be explained by the combination of an optimized GA parametrization along with the implemented self-adaptive feature.

For a large combinational library of thrombin inhibitors, containing substituents with rather similar properties, all 12 known binders were included in the final library. However, this library was only reduced to about a third of the initial chemical space. Most of the substituents in the initial population are known to interact with thrombin at their respective positions, and accordingly the obtained docking scores do not discriminate the various library members sufficiently. Therefore, it cannot be expected that a dramatic reduction of the chemical space is achieved for this example.

Additional studies will be required to evaluate at best even larger data sets. Unfortunately, however, there are only a very small number of libraries in the public domain, for which a significant fraction of its members have been experimentally characterized in terms of structure and binding affinity.

ACKNOWLEDGMENT

We are grateful to Andreas Spitzmüller (University of Marburg, Germany) and Simon Cottrell (CCDC, Cambridge, U.K.) for critically reading the manuscript.

REFERENCES AND NOTES

(1) Schneider, G.; Fechner, U. Computer-based de novo design of drug-like molecules. *Nat. Rev. Drug Discovery* **2005**, *4*, 649–63.
(2) Lauri, G.; Bartlett, P. CAVEAT: A program to facilitate the design of organic molecules. *J. Comput.-Aided Mol. Des.* **1994**, *8*, 51–66.
(3) Law, J. M. S.; Fung, D. Y. K.; Zsoldos, Z.; Simon, A.; Szabo, Z.; Csizmadia, I. G.; Johnson, A. P. Validation of the SPROUT de novo design program. *THEOCHEM* **2003**, 8463.
(4) Böhm, H. J. The computer program LUDI: a new method for the de novo design of enzyme inhibitors. *J. Comput.-Aided Mol. Des.* **1992**, *6*, 61–78.
(5) Sun, Y.; Ewing, T. J. A.; Skillman, A. G.; Kuntz, I. D. CombiDOCK: Structure-based combinatorial docking and library design. *J. Comput.-Aided Mol. Des.* **1998**, *12*, 597–604.
(6) Ewing, T. J.; Makino, S.; Skillman, A. G.; Kuntz, I. D. DOCK 4.0: search strategies for automated molecular docking of flexible molecule databases. *J. Comput.-Aided Mol. Des.* **2001**, *15*, 411–28.
(7) Böhm, H.-J.; Banner, D.; Weber, L. Combinatorial docking and combinatorial chemistry: Design of potent non-peptide thrombin inhibitors. *J. Comput.-Aided Mol. Des.* **1999**, *1*, 51–56.
(8) Gastreich, M.; Lilienthal, M.; Briem, H.; Claussen, H. Ultrafast de novo docking combining pharmacophores and combinatorics. *J. Comput.-Aided Mol. Des.* **2007**, *12*, 717–734.
(9) Degen, J.; Rarey, M. FlexNovo: Structure-based searching in large fragment spaces. *ChemMedChem* **2006**, *1*, 854–68.
(10) Gerlach, C.; Munzel, M.; Baum, B.; Gerber, H. D.; Craan, T.; Diederich, W. E.; Klebe, G. KNOBLE: A knowledge-based approach for the design and synthesis of readily accessible small-molecule chemical probes to test protein binding. *Angew. Chem., Int. Ed. Engl.* **2007**, *46*, 9105–9.

(11) Kuhn, D.; Weskamp, N.; Schmitt, S.; Hüllermeier, E.; Klebe, G. From the similarity analysis of protein cavities to the functional classification of protein families using cavbase. *J. Mol. Biol.* **2006**, *359*, 1023–44.
(12) Proschak, E.; Sander, K.; Zettl, H.; Tanrikulu, Y.; Rau, O.; Schneider, P.; Schubert-Zsilavecz, M.; Stark, H.; Schneider, G. From molecular shape to potent bioactive agents II: fragment-based de novo design. *ChemMedChem* **2009**, *4*, 45–8.
(13) Truchon, J. F.; Bayly, C. I. GLARE: A New Approach for Filtering Large Reagent Lists in Combinatorial Library Design Using Product Properties. *J. Chem. Inf. Model.* **2006**, *4*, 1536–48.
(14) Agrafiotis, D. K.; Lobanov, V. S. Ultrafast algorithm for designing focused combinational arrays. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 1030–8.
(15) Le Bailly de Tilleghem, C.; Beck, B.; Boulanger, B.; Govaerts, B. A fast exchange algorithm for designing focused libraries in lead optimization. *J Chem Inf Model* **2005**, *45*, 758–767.
(16) Zheng, W.; Cho, S. J.; Waller, C. L.; Tropsha, A. Rational combinatorial library design. 3. Simulated annealing guided evaluation (SAGE) of molecular diversity: a novel computational tool for universal library design and database mining. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 738–46.
(17) Good, A. C.; Lewis, R. A. New Methodology for Profiling Combinatorial Libraries and Screening Sets: Cleaning Up the Design Process with HARPick. *J. Med. Chem.* **1997**, *40*, 3926–3936.
(18) Tropsha, A. Rational principles of compound selection for combinatorial library design. *Comb. Chem. High Throughput Screening* **2002**, *5*, 111−123.
(19) Zheng, W.; Cho, S. J.; Tropsha, A. Rational combinatorial library design. 1. Focus-2D: A new approach to the design of targeted combinatorial chemical libraries. *J. Chem. Inf. Model.* **1998**, *38*, 251–258.
(20) Zheng, W.; Hung, S. T.; Saunders, J. T.; Seibel, G. L. PICCOLO: a tool for combinatorial library design via multicriterion optimization. *Pac. Symp. Biocomput.* **2000**, 588–599.
(21) Schneider, G.; Lee, M.-L.; Stahl, M.; Schneider, P. De novo design of molecular architectures by evolutionary assembly of drug-derived building blocks. *J. Comput.-Aided Mol. Des.* **2000**, *14*, 487–494.
(22) Fechner, U.; Schneider, G. Flux (1): A virtual synthesis scheme for fragment-based de novo design. *J. Chem. Inf. Model.* **2006**, *46*, 699–707.
(23) Schuller, A.; Schneider, G. Identification of hits and lead structure candidates with limited resources by adaptive optimization. *J. Chem. Inf. Model.* **2008**, *48*, 1473–91.
(24) Singh, J.; Ator, M. A.; Jaeger, E. P.; Allen, M. P.; Whipple, D. A.; Soloweij, J. E.; Chowdhary, S.; Treasurywala, A. M. Application of genetic algorithms to combinatorial synthesis: A computational approach to lead identification and lead optimization. *J. Am. Chem. Soc.* **1996**, *118*, 1669–1676.
(25) Brown, R. D.; Martin, Y. C. Designing Combinatorial Library Mixtures Using a Genetic Algorithm. *J. Med. Chem.* **1997**, *40*, 2304–2313.
(26) Sheridan, R. P.; SanFeliciano, S. G.; Kearsley, S. K. Designing targeted libraries with genetic algorithms. *J. Mol. Graph. Model.* **2000**, *18*, 320–34.
(27) Westhead, D. R.; Clark, D. E.; Frenkel, D.; Li, J.; Murray, C. W.; Robson, B.; Waszkowycz, B. PRO-LIGAND: An approach to de novo molecular design. 3. A genetic algorithm for structure refinement. *J. Comput.-Aided Mol. Des.* **1995**, *9*, 139–48.
(28) Gillet, V. J.; Khatib, W.; Willett, P.; Fleming, P. J.; Green, D. V. Combinatorial library design using a multiobjective genetic algorithm. *J. Chem. Inf. Comput. Sci.* **2002**, *42*, 375–85.
(29) Douguet, D.; Thoreau, E.; Grassy, G.Â. r. A genetic algorithm for the automated generation of small organic molecules: Drug design using an evolutionary algorithm. *J. Comput.-Aided Mol. Des.* **2000**, *14*, 449–466.
(30) Dey, F.; Caflisch, A. Fragment-based de novo ligand design by multiobjective evolutionary optimization. *J. Chem. Inf. Model.* **2008**, *48*, 679–90.
(31) Weber, L. Evolutionary combinatorial chemistry: application of genetic algorithms. *Angew. Chem., Int. Ed. Engl.* **1998**, *107*, 2.
(32) Nicolaou, C. A.; Apostolakis, J.; Pattichis, C. S. De novo drug design using multiobjective evolutionary graphs. *J. Chem. Inf. Model.* **2009**, *2*, 295–307.
(33) Vinkers, H. M.; de Jonge, M. R.; Daeyaert, F. F.; Heeres, J.; Koymans, L. M.; van Lenthe, J. H.; Lewi, P. J.; Timmerman, H.; Van Aken, K.; Janssen, P. A. SYNOPSIS: Synthesize and optimize system in silico. *J. Med. Chem.* **2003**, *46*, 2765–73.
(34) Belda, I.; Madurga, S.; Llora, X.; Martinell, M.; Tarrago, T.; Piqueras, M.; Nicolas, E.; Giralt, E. ENPDA: an evolutionary structure-based de novo peptide design algorithm. *J. Comput.-Aided Mol. Des.* **2005**, *19*, 585–601.
(35) Pegg, S. C.; Haresco, J. J.; Kuntz, I. D. A genetic algorithm for structure-based de novo design. *J. Comput.-Aided Mol. Des.* **2001**, *15*, 911–933.
(36) Liu, Q.; Masek, B.; Smith, K.; Smith, J. Tagged fragment method for evolutionary structure-based de novo lead generation and optimization. *J. Med. Chem.* **2007**, *50*, 5392–402.
(37) Lipinski, C. A.; Lombardo, F.; Dominy, B. W.; Feeney, P. J. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Adv. Drug Delivery Rev.* **2001**, *46*, 3–26.
(38) Sadowski, J., Schwab, C. H., Gasteiger, J. *CORINA, 3D Structure Generator*, version 2.0; Molecular Networks: Erlangen, Germany, 2009.
(39) Huey, R.; Morris, G. M.; Olson, A. J.; Goodsell, D. S. A semiempirical free energy force field with charge-based desolvation. *J. Comput. Chem.* **2007**, *28*, 1145–52.
(40) Verdonk, M. L.; Cole, J. C.; Hartshorn, M. J.; Murray, C. W.; Taylor, R. D. Improved protein−ligand docking using GOLD. *Proteins* **2003**, *52*, 609–23.
(41) Velec, H. F.; Gohlke, H.; Klebe, G. DrugScore(CSD)-knowledge-based scoring function derived from small molecule crystal data with superior recognition rate of near-native ligand poses and better affinity prediction. *J. Med. Chem.* **2005**, *48*, 6296–303.
(42) Back, T. *Evolutionary Algorithms in Theory and Practise*; Oxford University Press: Oxford, U. K., 1996.
(43) Deb, K.; Beyer, H. G. Self-adaptive genetic algorithms with simulated binary crossover. *Evol. Comput.* **2001**, *9*, 197–221.
(44) Bartz-Beielstein, T. *Experimental Research in Evolutionary Computation—The New Experimentalism*; Springer Verlag: Heidelberg, Germany, 2006.
(45) Kick, E. K.; Roe, D. C.; Skillman, A. G.; Liu, G.; Ewing, T. J.; Sun, Y.; Kuntz, I. D.; Ellman, J. A. Structure-based design and combinatorial chemistry yield low nanomolar inhibitors of cathepsin D. *Chem. Biol.* **1997**, *4*, 297–307.
(46) Goldberg, D. E. *Genetic Algorithms in Search, Optimization and Machine Learning*; Addison-Wesley: Bonn, 1989.
(47) Harris, J. L.; Backes, B. J.; Leonetti, F.; Mahrus, S.; Ellman, J. A.; Craik, C. S. Rapid and general profiling of protease specificity by using combinatorial fluorogenic substrate libraries. *Proc. Natl. Acad. Sci. U. S. A.* **2000**, *97*, 7754–9.
(48) *Tripos mol2 File Format*, 2009, www.tripos.com.
(49) *MGLTools v1.5.2*, 2009, http://mgltools.scripps.edu/.
(50) CCDC, 2009, http://ccdc.cam.ac.uk/products/life_sciences/gold/case_studies/gold_validation_virtual_screening.
(51) *Molecular Operating Environment*; Chemical Computing Group: Montreal, CA, 2009.
(52) Linusson, A.; Gottfries, J.; Olsson, T.; Ornskov, E.; Folestad, S.; Norden, B.; Wold, S. Statistical molecular design, parallel synthesis, and biological evaluation of a library of thrombin inhibitors. *J. Med. Chem.* **2001**, *44*, 3424–39.
(53) Bode, W.; Mayr, I.; Baumann, U.; Huber, R.; Stone, S. R.; Hofsteenge, J. The refined 1.9 Å crystal structure of human α-thrombin: interaction with D-Phe-Pro-Arg chloromethylketone and significance of the Tyr-Pro-Pro-Trp insertion segment. *EMBO J.* **1989**, *8*, 3467–75.
(54) Mathews, I. I.; Padmanabhan, K. P.; Ganesh, V.; Tulinsky, A.; Ishii, M.; Chen, J.; Turck, C. W.; Coughlin, S. R.; Fenton, J. W., 2nd. Crystallographic structures of thrombin complexed with thrombin receptor peptides: existence of expected and novel binding modes. *Biochemistry* **1994**, *33*, 3266–79.
(55) Sall, D. J.; Bastian, J. A.; Briggs, S. L.; Buben, J. A.; Chirgadze, N. Y.; Clawson, D. K.; Denney, M. L.; Giera, D. D.; Gifford-Moore, D. S.; Harper, R. W.; Hauser, K. L.; Klimkowski, V. J.; Kohn, T. J.; Lin, H. S.; McCowan, J. R.; Palkowitz, A. D.; Smith, G. F.; Takeuchi, K.; Thrasher, K. J.; Tinsley, J. M.; Utterback, B. G.; Yan, S. C.; Zhang, M. Dibasic benzo[*b*]thiophene derivatives as a novel class of active site-directed thrombin inhibitors. 1. Determination of the serine protease selectivity, structure-activity relationships, and binding orientation. *J. Med. Chem.* **1997**, *40*, 3489–93.
(56) Malikayil, J. A.; Burkhart, J. P.; Schreuder, H. A.; Broersma, R. J., Jr.; Tardif, C.; Kutcher, L. W., 3rd; Mehdi, S.; Schatzman, G. L.; Neises, B.; Peet, N. P. Molecular design and characterization of an α-thrombin inhibitor containing a novel P1 moiety. *Biochemistry* **1997**, *36*, 1034–40.
(57) Riester, D.; Wirsching, F.; Salinas, G.; Keller, M.; Gebinoga, M.; Kamphausen, S.; Merkwirth, C.; Goetz, R.; Wiesenfeldt, M.; Sturzebecher, J.; Bode, W.; Friedrich, R.; Thurk, M.; Schwienhorst, A. Thrombin inhibitors identified by computer-assisted multiparameter design. *Proc. Natl. Acad. Sci. U. S. A.* **2005**, *102*, 8597–602.
(58) van de Locht, A.; Lamba, D.; Bauer, M.; Huber, R.; Friedrich, T.; Kroger, B.; Hoffken, W.; Bode, W. Two heads are better than one: crystal structure of the insect derived double domain Kazal inhibitor rhodniin in complex with thrombin. *EMBO J.* **1995**, *14*, 5149–57.

CI9003305