From greedy to branch & bound and back: Assessing optimization strategies for incremental construction molecular docking tools

Axel Griewel
Matthias Rarey

Center for Bioinformatics · University of Hamburg · Bundesstr. 43 · 20146 Hamburg
{griewel | rarey}@zbh.uni-hamburg.de

Abstract:

A branch & bound approach for the assembly-phase of the incremental construction algorithm of the software package FlexX is presented. For this a local bound for partial solutions has been implemented which estimates the best score achievable for the considered solutions. This estimation is based on scoring values which single components can achieve in certain regions of the active site as well as distance constraints deduced from the composition of the considered partial solution. Furthermore, a time- and space-bounded search strategy specific to the addressed problem has been developed.

The implemented algorithm was tested on a dataset containing 169 complexes. In 116 of these cases the calculation was finished in a reasonably defined time frame while the calculation for the remaining complexes was not completed in this period of time. For all calculated complexes, the best solution shows a score better or equal to the solution of standard-FlexX. This, however, is not always associated with an improvement of the RMSD between the calculated placement of the ligand and the crystal structure. The presented algorithm is applicable for thorough virtual screening of small sets of ligands comprising up to nine rotatable, acyclic bonds. Furthermore, the method gives important insights to the k-greedy method and can be used for scientific assessment of new scoring functions within FlexX.

1 Introduction

Virtual Screening has proven to be a potent tool in current drug design [AT01, Lyn02]. At this the goal is to set up a ranking of several ligands contained in a library according to binding affinity with respect to a certain protein. If the protein structure is available, it is possible to make such a prediction by calculating the binding mode of a ligand within the active site of the protein and establishing a ranking according to a scoring function estimating the binding affinity. Several approaches for addressing this problem are known [Rar02] of which FlexX [RKLK96] utilizes an incremental construction technique. This technique consists of three phases: base fragment selection, base placement and incremental assembly.

Due to the k-greedy approach which is utilized in the incremental assembly phase, it is possible that the incremental construction algorithm discards the best solution which can be found. This is caused by the deletion of less favorable placements which is undertaken after each step in the incremental assembly phase. In this paper we report a replacement of this approach by a branch & bound method [LW66]. In a branch & bound method the best solution is tracked down using a bound of the best score a partial solution is able to achieve as well as the score of the best complete solution found so far.

Two main tasks have to be addressed for the replacement of the k-greedy approach by the new method. Firstly, it is essential to define a local score-bound for partial solutions which determines the best score a partial solution is able to achieve. Here it is of crucial importance that the local bound yields a score estimation which is always better or equal to the actual score which can be achieved. Secondly, a search strategy is to be defined which allows for the expansion of promising partial solutions while taking physical limitations of computer resources into account.

In the following, we describe a branch & bound approach for FlexX. At the beginning of the methods section an introduction to basic technologies used in standard FlexX can be found. After this the determination of the local bound is described followed by a specification of the implemented search strategy. Due to the problem complexity and composition, it was not possible to define a sound local bound. In practical test, however, we can show that the presented local bound yields correct answers for nearly all cases. In the results section an assessment of the new method as well as a comparison in terms of RMSD, score and time to standard-FlexX is performed. Since the k-greedy method used before is heuristic, the implementation of the branch & bound method for the first time allows to estimate the error introduced by the k-greedy method.

2 Methods

2.1 FlexX basic technology

The chemical model underlying FlexX consists of the description of conformations, interactions and a scoring function. Ligand conformations in FlexX are based on torsion angle potentials taken from the MIMUMBA torsion angle library [KM94] while ring conformations are precalculated using the program CORINA [GCS90, SGK94]. The interaction model as well as the additive, empirical scoring function in FlexX are based on LUDI [Böh92b, Böh92a, Böh94]. Basically four types of protein ligand interactions are distinguished in FlexX: hydrogen bonds, metal interactions and geometrically restricted and unrestricted hydrophobic contacts. For chemical groups which are able to form hydrophilic interactions or geometrically restricted contacts, an interaction geometry is set up. This comprises an interaction center as well as an interaction surface. The surface defines a space in which a counter group yields a favorable interaction and is either shaped like a sphere, a spherical rectangle, or a spherical cone. For reasons of efficiency, interaction surfaces on the protein side are approximated using discrete interaction points.

The first step to be carried out for a docking process in FlexX is the selection of *base fragments* [RKL97]. For this and further purposes the ligand is split into *components* by cutting at each acyclic rotatable bond. Base fragments which consist of up to three components are then chosen with a preference for rather rigid parts of the molecule which comprise a large number of characteristic functional groups. Each base fragment is fitted into the manually defined active site using line and triangle matching and pose clustering yielding several base placements [RWL96].

During the incremental assembly phase of FlexX the remaining components are added subsequently. Starting from a base fragment, it is possible to set up a rooted tree describing a *fragmentation* and thereby defining the sequence in which the residual components are added. Every time a placement of a part of the ligand is expanded by a component, the component is added in all considered conformations and checks for substantial clashes and new interactions between ligand and protein are carried out. After each expansion an optimization of the ligand position with respect to the newly determined clashes and interactions is performed.

Since the solution space in the incremental assembly phase is growing exponentially with the number of components and their conformation, a full exploration of this space is not feasible. In order to focus on promising placements, a k-greedy approach is used in standard FlexX. As a starting point the base placements are utilized. Then the next component is added to each partial solution in the set in all given conformations. The set of partial solutions consists here of either base placements or - in later iterations - placements of the partial ligand. Expanded solutions which seriously clash are deleted immediately while all other partial solutions are subject to optimization and clustering. The clustered set of partial solutions is further diminished by selecting only the best i solutions for each fragmentation plus the best j solutions of the remaining partial solutions regardless of their fragmentation. All partial solutions of this set are used in the next iteration for the addition of the next fragment. This process is iterated until all retained partial solutions are completed.

2.2 Evaluation of partial solutions

A local bound for partial solutions has been developed for the presented branch & bound method. The goal here is to estimate the best score a partial solution is able to achieve. In the branch and bound method this value is compared to the best score achieved so far. If the local bound is worse than the best score found as yet, it is decided to delete the considered solution; if it is better, the partial solution is kept for further exploration. The local bound will make use of the additivity of the scoring function with respect to the ligand components, distance constrains and scoring values components of the ligand can achieve.

The local bound is composed of three values: The first value e_{fix} is the score of the already placed part of the ligand. This value can be easily computed by applying the standard scoring function to the fixed, partial placement. The second value estimates

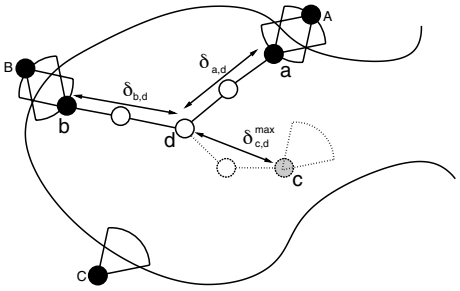
the best possible score contribution of unplaced, non-interacting components. These are components which are unable to form interactions or geometrically restricted contacts as e. g. CH_2-groups. The estimated score contribution for these components is defined as $e_{cont}(c) = \sum_{a \in c} \max(f_{\overline{H}}(a) - 5; 0) \cdot \Delta G_{lipo}$ where c is the considered component, a references an atom of c, $f_{\overline{H}}(a)$ determines the number of non-hydrogen atoms bound to a and ΔG_{lipo} is the maximal score-contribution of a geometrically not restricted, hydrophobic contacts. The third term of the local bound e_{strong} estimates score contributions for components which are not yet processed and able to form interactions or geometrically restricted contact. The score contributions of these strongly interacting components are of major interest since interactions of these components usually dominate the score of complete solutions. The algorithm for the estimation of e_{strong} is described now.

A new phase termed annotation phase has been introduced to the FlexX algorithm. Prior to all other phases it is attempted to find the best score a certain component is able to achieve if it is interacting with a certain protein interaction surface. Note that the total score the component achieves is calculated here using the standard scoring function including interactions with other atoms.

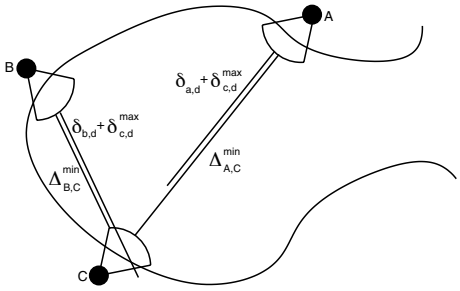
The single interaction scan method (M. Rarey, in preparation) is utilized in the annotation phase: Given a pair of a ligand interaction center ℓ and protein interaction surface s, this method finds favorable positions of a fragment of the ligand containing ℓ with respect to s. For this purpose, vectors are set up pointing from the interaction centers of s and ℓ to each associated interaction point. These vectors are iteratively contrary superimposed and for each superimposition, the considered fragment of the ligand is rotated in discrete steps about the axis given by the superimposed vectors. This procedure is carried out for each compatible tuple of interaction surface of ligand and active site. Note that only the component associated to ℓ is considered here yielding the best score the component is able to achieve if it is interacting with a certain protein interaction surface. This phase yields a list of optimal score values for each compatible component at each protein interaction surface.

The best score contribution an unplaced component is able to yield in a partial solution is determined using the aforementioned annotation values as well as distance constraints. Our goal is to calculate which protein interaction surfaces an unplaced component is able to reach under consideration of distance constraints derived from the partial solution. For this purpose two tables `lig_tab` and `rec_tab` are precalculated. The first mentioned table contains maximal distances between points of the ligand and is calculated according to [HRBL02]. The table `rec_tab` provides information on minimal distances between protein interaction surfaces. Using these tables, a set of reachable protein interaction surfaces is defined for strongly interacting, unplaced components.

An interaction surface X is classified as (not) reachable for an unplaced component u based on the interactions already formed between the placed part of the ligand and the protein as exemplary depicted in Figure 1. For each unplaced, strongly interacting component u of the ligand, the fragment ℓ to which the unplaced component is to be added is determined. The maximal distance $\delta_{u,\ell}$ between u and ℓ is then read out from the precalculated table `lig_tab`. The euclidian distance $\delta_{i,\ell}$ between each ligand atom i forming an interaction



(a) Schematic partial placement of a ligand in an active site. Black circles inside the indicated active site represent placed fragments of the ligand, while grey circles represent unplaced fragments. Structures containing an interaction surface are represented by filled circles with a circle-sector of bigger radius denoting the interaction surface. The distances δ used for calculating reachability are shown as well.



(b) Check if C is reachable for c in the partial placement displayed in (a): Comparison of the minimal distances Δ between receptor interaction surfaces to the maximal distance δ a fragment of the ligand is able to stretch out from the fragment interacting. In this example $\Delta^{\mathsf{min}}_{\mathsf{B,C}} < \delta_{\mathsf{b,d}} + \delta^{\mathsf{max}}_{\mathsf{c,d}}$ and $\Delta^{\mathsf{min}}_{\mathsf{A,C}} > \delta_{\mathsf{a,d}} + \delta^{\mathsf{max}}_{\mathsf{c,d}}$.

Figure 1: A partial placement of a ligand comprising two interactions is shown in (a). The distance measures $\delta_{\mathsf{a,d}} + \delta^{\mathsf{max}}_{\mathsf{c,d}}$ and $\delta_{\mathsf{b,d}} + \delta^{\mathsf{max}}_{\mathsf{c,d}}$ state how far the unplaced part of the ligand can stretch out from the considered interaction surface A and B. These distances are compared to the corresponding precalculated minimal distances Δ between receptor interaction surfaces as shown in (b). In the considered example it is not possible that an interaction between C and the unplaced part of the ligand takes place, because the unplaced part of the ligand can not stretch out far enough from interaction surface A.

with the receptor and ℓ is determined on the fly. Afterwards the maximal distance the unplaced component u is able to stretch out from the interacting atom i is calculated as $\delta_{i,u} = \delta_{i,\ell} + \delta_{\ell,u}$.

When assessing the reachability of a protein interaction surface Y for unplaced components u the precalculated table `rec_tab` can be utilized to determine the minimal distance $\Delta_{\mathsf{X,Y}}$ between the interaction surface Y under investigation and each surface X interacting with the already placed part of the ligand. If the equation $\Delta_{\mathsf{X,Y}} < \delta_{i,u}$ where i is a ligand component interacting with X and u is an unplaced component compatible to receptor interaction surface Y holds, then this surface is considered as reachable for u. From all reachable interaction surfaces for u the best score which is achievable according to the annotation values is then taken as estimation of the score value e_{strong} for u in the considered placement.

Recapitulating, the local bound b of a partial solution p is built up of the sum of three values which belong to three disjunct sets of components as described in equation 1. Here $F(p)$ references all placed components in p while $K(p)$ comprises all unplaced components which are not able to form interactions or geometric restricted contacts in p leaving all

components which are unplaced and able to form interactions in $S(p)$.

$$b(p) = \sum_{f \in F(p)} e_{fix}(p, f) + \sum_{k \in K(p)} e_{cont}(k) + \sum_{s \in S(p)} e_{strong}(p, s) \tag{1}$$

2.3 Search Strategy

Another essential part of a branch & bound method is the applied search strategy, i. e. the way of exploring the solution space. Here it was chosen to implement a time- and space bounded hybrid algorithm which combines features of depth- (DFS) and best-first search (BeFS). Time references in this case the number of non-clashing (partial) solutions which have successfully been assessed and is thus correlated to computing time while not giving an accurate temporal measure. To implement this algorithm the partial solutions currently present in computer memory are split into sets by the number of placed components. Each set is represented by a priority queue which is sorted by ascending local bound. This yields easy access to the best partial solution for each set which shall be termed front runner. Using this representation, search priorities according to DFS and BeFS are set up for each front runner and according to a time- and a space-dependent criterion it is decided which front runner is to be expanded.

The search priority of a partial solution according to DFS σ_{DFS} is defined as the percentage of the already placed components with respect to the total number of components in the ligand. The search priority according to BeFS σ_{BeFS} is calculated using the score of each front runner. A percentage of the score of each front runner relative to the best score present in the set of front runners is calculated. This yields two search priorities σ_{DFS} and σ_{BeFS} each in range $[0; 100]$. The partial solution which will be expanded next is determined by the two introduced priorities as well as the number of partial solutions currently represented in computer memory and the number of explored partial solutions. The latter two properties are included by determining a variable $\alpha \in [0; 1]$ using piecewise linear functions. The larger α, the more the search priority according to DFS counts, the closer α to zero, the more the search priority according to BeFS counts. The computer space as well as the time available can be configured. The variable α is set to zero as long as certain time- and space thresholds are not reached. As soon as one of the thresholds is passed, the value of α rises linearly to one.

Finally the combined search priority $\varrho = \alpha \cdot \sigma_{DFS} + (1 - \alpha) \cdot \sigma_{BeFS}$ is determined for each front runner. The front runner with maximal ϱ will be expanded next. The determination of ϱ allows to combine the attributes ”most promising” (BeFS) and ”next to be finished” (DFS) for a partial solution and thereby potentiates a sensible decision which partial solution should be expanded next.

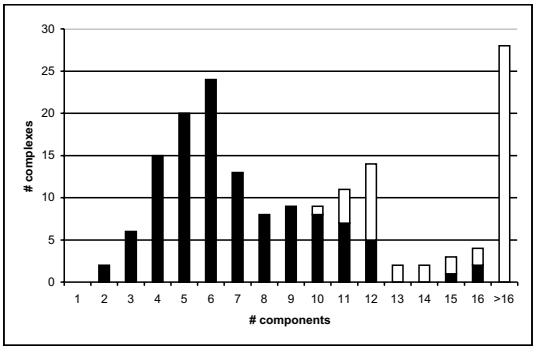


Figure 2: Number of completed (black) and not finished (white) calculations separated by the number of components of the ligand

3 Results and Discussion

A first test of the new algorithm was carried out using 169 complexes from an earlier benchmark set of 200 complexes [KRL99]. Each of these complexes was taken from the PDB and split into ligand and protein. Afterwards the active site of the protein has been defined and the structure of the ligand has been minimized. The neglected 31 complexes comprise examples in which no incremental assembly phase is performed due to the size of the ligand.

Using the presented method it was possible to complete the calculation for 116 of the considered complexes in a time frame of 1,000,000 expansions. The calculation for complexes containing more flexible ligands failed because the considerable size of the solution space could not be narrowed down appropriately using the described algorithm. In Figure 2 the number of complexes for which the calculation was (not) completed after 1,000,000 expansions is related to the number of components of the ligand. The diagram shows that all 97 calculations for ligands containing up to nine components were completed within the given time frame while none of the 28 calculations for ligands with more than 16 components could be finished. For all complexes with intermediately flexible ligands the feasibility of the calculation depends on the nature of the complex, i. e. the number of interaction centers and the shape of the active site.

An assessment of the difference in placement of the highest scoring solution determined by the branch & bound procedure and the standard version of FlexX in different configurations was carried out. The values i and j [see FlexX basic technology] which configure the width in which the solution space is explored in the k-greedy approach were chosen as $i = 2^w \cdot 25$ and $j = 2^w \cdot 100$ with $w \in \{0; 1; 2; 3\}$ while the time frame for the branch & bound method was fixed at 1,000,000 expansions. For each complex and configuration w the RMSD has been calculated between the best scoring solution of the k-greedy vs. the branch & bound method. The results are displayed in the form of four histograms in Figure 3 with an extra line added showing the number of complexes with very similar

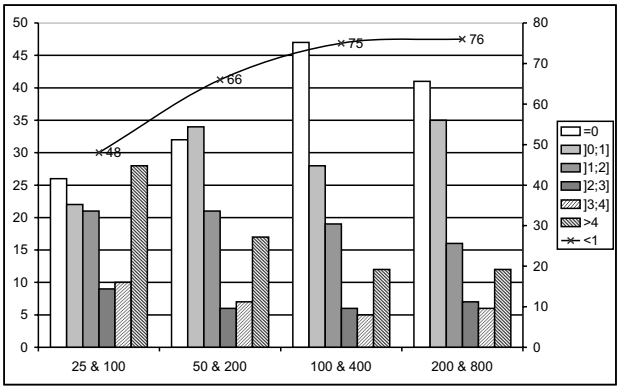


Figure 3: Four histograms each displaying the distribution of RMSD-values between the best solution found with the standard- and the branch & bound-method are shown. At this each histogram depicts the distribution for a certain configuration of i & j of the k-greedy approach (abscissa). In each histogram six columns can be found which display how many complexes deviate by a certain RMSD as defined in the legend. While all columns are scaled according to the left ordinate, the line which depicts the number of complexes with an RMSD < 1 is scaled according to the right ordinate.

results.

From the diagram in Figure 3 it can be seen that the larger i and j are chosen, the better the chance to find the best solution according to the given base placements. This can be deduced from the fact that the number of complexes with an RMSD of > 5 shrinks and that the number of complexes which yield an RMSD < 1 rises the higher the values for i and j are. The best tradeoff between the width in which the solution space is to be searched and the number of complexes with RMSD < 1 is reached for $i = 100$ and $j = 400$. This is indicated by the number of solutions having an RMSD < 1 which is nearly the same for the configurations $i = 100; j = 400$ and $i = 200; j = 800$. Furthermore, the number of solutions with an RMSD of 0 decreases from the second last to the last configuration. This can be explained with the mixed numeric-, combinatorial approach of standard-FlexX which includes clustering and optimizations. Thus certain assumptions which hold for combinatorial considerations can only partially be adopted for this problem. As an example the clustering of FlexX shall be mentioned. Here it is possible that the partial solution which would lead to the best scoring complete solution is not chosen as representative for a cluster because of inferior placement of the processed components.

Another decisive parameter for the assessment of the two algorithms is the comparison of the score values of the best solution found. In Figure 4 the improvement in score for single complexes is shown. Here the best score achieved with the the k-greedy approach ($i = 100; j = 400$) has been subtracted from the best score found with the branch & bound procedure. A regression line for the whole dataset and one for all dots in the abscissa-interval of [2;9] was included in this diagram.

The introduction of the shorter regression line into Figure 4 was motivated by the data un-

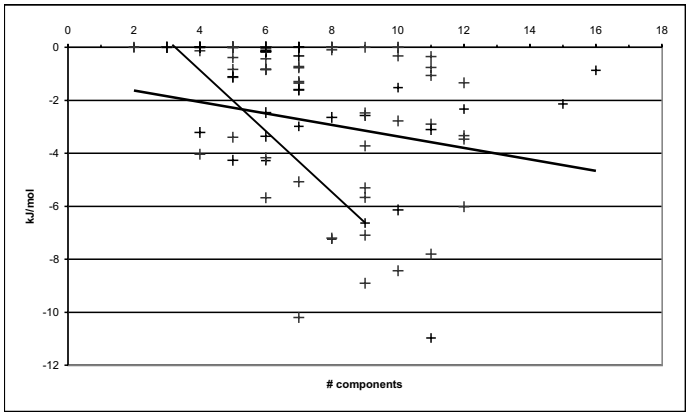


Figure 4: Difference in score of the best solution found in the branch & bound procedure and with the standard version of FlexX. For each completed calculation a dot regarding the number of components of that complex (abscissa) and the improvement in score when performing a branch & bound calculation (ordinate) is plotted. Furthermore a regression line for the intervals [2;9] and [2;16] is shown.

derlying Figure 2. Here it is shown that for ligands containing more than nine components the calculation with the branch & bound procedure does not finish for all cases. This suggests that the solution space of those complexes which have been completely calculated is rather narrow and that these complexes are biassed on an efficient calculation using standard FlexX. This is reflected in the comparison of the slope of the long regression line which is -0.22 to the slope for the short regression line which is -1. On this account only the rough conclusion shall be drawn that for a ligand with n components the score of the best solution found in the k-greedy approach is on average $-(n-3)\ {}^{\mathrm{kJ}}/_{\mathrm{mol}} \mid n \in \{3,..,9\}$ worse than the best achievable score. For ligands which are made up of more than nine components no explicit prediction can be made. It is expected, however, that this trend is counter played by the score which does not rise linearly with the number of components of a ligand.

It has also been assessed whether the improvement in score yields solutions which are more similar to the crystal structure. For this purpose the RMSD-value between the best solution and the crystal structure (RMSD_c) has been calculated for both approaches. Afterwards the difference between the RMSD_c values for each complex has been calculated yielding a measure describing if an improvement in position has occurred. Here it was found that the improvement in score which has been achieved using the branch & bound method is not unambiguously related to an improvement in RMSD_c. There are 52 cases for which no improvement in RMSD_c was found. For 32 complexes the RMSD_c shrinks by less than 1 Å while there are also 22 complexes in which the RMSD rises by less than 1 Å. Furthermore five improvements and five deteriorations with a change in RMSD_c bigger than 1 Å have been observed. This shows that the branch & bound approach does not in every case lead to preferable placements. For the five large deteriorations weaknesses of the scoring function can be held responsible. These result from e. g. the missing consid-

eration of desolvation effects and a tendency of FlexX to overestimate charged ineractions resulting in placements at the border of the defined active site.

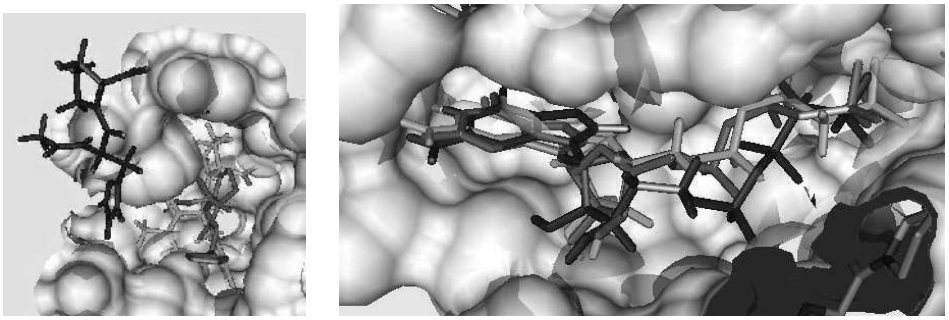
As expected, the new branch & bound method needs substantially more computing time. The runtime for completely calculated complexes spans from 1 s to 2 h while 90% of the complexes were completed in less than 14 min. These calculations were carried out on Linux machines having intel dual xeon processors with 3,06 GHz, 1 MB cache and 4 GB memory. At this only one processor was used for one complex at a time. This shows that the presented method is considerably slower than the standard version but still applicable to the prediction of complexes containing less flexible ligands. Furthermore the approach can be utilized for the thorough assessment of scoring functions.

As an example the results for the docking of N-Acetyl-L-His-D-Pro-Oh with an Immunoglobulin λ Light Chain Dimer (Mcg) which can be found under PDB-code *1mcr* [EHF+93] shall be described here. In Figure 5(a) the docking results are displayed. At this the red structure shows the position of the ligand determined in the X-ray experiment, the green structure displays the best solution found with the branch & bound procedure while the blue structure displays the best solution found with standard FlexX. Here an RMSD_c of 10.8 Å is achieved in the standard version of FlexX while the best solution found in the branch & bound procedure is localized in the same area as the ligand in the crystal structure with an RMSD_c of 4.6 Å. The smallest RMSD_c in the set of solutions equals 4.16 Å which shows that the standard FlexX scoring function is not well suited for the calculation of this complex. Caused by the extreme difference in RMSD_c the scores differ by $-7.2\ {}^{\text{kJ}}/_{\text{mol}}$. This, however, is paid for with an increase in calculation time from 12.6 s to 15.5 min.

As a second example the docking of the G-binding protein H-Ras P21 and Guanosine-5'-[B, G-Methylene] Triphosphate which is filed under PDB-code *121p* [WKJ+91] shall be described. The RMSD_c for the best solution achieved with the branch & bound procedure is 0.89 Å while the one achieved with standard FlexX is 1.19 Å. At this the two solutions differ by 1.71 Å. In Figure 5(b) the two solutions are displayed using the introduced coloring scheme. It can be seen that the two calculated solutions overlap very well with the crystal structure in the guanosine region, while this is not the case for the phosphate chain. Since the phosphate chain is quite hydrophilic this yields a considerable difference of $-10.97\ {}^{\text{kJ}}/_{\text{mol}}$ in score. Nevertheless the improvement in RMSD_c caused by this score-improvement is relatively small. The time used for the incremental assembly phase lies at 16 min for the branch & bound procedure while standard FlexX finishes this calculation in about one minute.

4 Conclusion

In this work, we presented a branch & bound algorithm for the thorough calculation of the incremental assembly phase of FlexX. It was shown that for many complexes considerable differences in score were achieved which could make a difference in screening experiments. Nevertheless it was also found that the difference in score does not clearly relate to an improvement in placement with respect to the crystal structure. This, however,



(a) Immunoglobulin λ Light Chain Dimer (Mcg) Complex With N-Acetyl-L-His-D-Pro-Oh (PDB-Code: *1mcr*)

(b) H-Ras P21 Protein Complex With Guanosine-5'-[B, G-Methylene] Triphosphate (PDB-Code: *121p*)

Figure 5: Docking results - The crystal structure is shown in red, the solution calculated with standard FlexX in blue and the solution calculated with the presented method in green.

is caused by the scoring function, which neglects certain terms and thus does not always favor the placement closest to the crystal structure. Furthermore the presented approach is - caused by its thoroughness - considerably slower than the k-greedy approach used in standard FlexX. This was expected and accepted because the developed tool allows for the accurate calculation of the binding mode of a complex. This again can be used for a detailed analysis of small sets of ligands or the assessment of other modules of the software package as e. g. the scoring function.

References

[AT01] R. Abagyan und M. Totrov. High-throughput docking for lead generation. *Curr Opin Chem Biol*, 5(4):375–82, Aug 2001.

[Böh92a] H. J. Böhm. The computer program LUDI: a new method for the de novo design of enzyme inhibitors. *J Comput Aided Mol Des*, 6(1):61–78, Feb 1992.

[Böh92b] H. J. Böhm. LUDI: rule-based automatic design of new substituents for enzyme inhibitor leads. *J Comput Aided Mol Des*, 6(6):593–606, Dec 1992.

[Böh94] H.-J. Böhm. The development of a simple empirical scoring function to estimate the binding constant for a protein-ligand complex of known thee-dimensional structure. *Journal of Computer-Aided Molecular Design*, 8:243–256, 1994.

[EHF+93] A. B. Edmundson, D. L. Harris, Z. C. Fan, L. W. Guddat, B. T. Schley, B. L. Hanson, G. Tribbick und H. M. Geysen. Principles and pitfalls in designing site-directed peptide ligands. *Proteins*, 16(3):246–67, Jul 1993.

[GCS90] J. Gasteiger, C.Rudolph und J. Sadowski. Automatic gerneration of 3D-atomic coordinates for organic molecules. *Tetrahedron Computer Methodology*, 3:537–547, 1990.

[HRBL02] Sally A Hindle, Matthias Rarey, Christian Buning und Thomas Lengaue. Flexible docking under pharmacophore type constraints. *J Comput Aided Mol Des*, 16(2):129–49, Feb 2002.

[KM94] G. Klebe und T. Mietzner. A fast and efficient method to generate biologically relevant conformations. *J Comput Aided Mol Des*, 8(5):583–606, Oct 1994.

[KRL99] B. Kramer, M. Rarey und T. Lengauer. Evaluation of the FLEXX incremental construction algorithm for protein-ligand docking. *Proteins*, 37(2):228–41, Nov 1999.

[LW66] E. L. Lawler und D. E. Wood. Branch and Bound methods: a survey. *Operations Research*, 14:699–719, 1966.

[Lyn02] Paul D Lyne. Structure-based virtual screening: an overview. *Drug Discov Today*, 7(20):1047–55, Oct 2002.

[Rar02] M. Rarey. Protein-Ligand Docking in Drug Design. In T. Lenauer, Hrsg., *Bioinformatics - From Genomes to Drugs*, Seiten 315–360. Wiley-VCH Verlag GmbH, 2002.

[RKL97] M. Rarey, B. Kramer und T. Lengauer. Multiple automatic base selection: protein-ligand docking based on incremental construction without manual intervention. *J Comput Aided Mol Des*, 11(4):369–84, Jul 1997.

[RKLK96] M. Rarey, B. Kramer, T. Lengauer und G. Klebe. A fast flexible docking method using an incremental construction algorithm. *J Mol Biol*, 261(3):470–89, Aug 1996.

[RWL96] M. Rarey, S. Wefing und T. Lengauer. Placement of medium-sized molecular fragments into active sites of proteins. *J Comput Aided Mol Des*, 10(1):41–54, Feb 1996.

[SGK94] J. Sadowski, J. Gasteiger und G. Klebe. Comparison of automatic three-dimensional model builders using 639 x-ray structures. *Journal of Chemical Information and Computer Science*, 34:1000–1008, 1994.

[WKJ+91] F. Wittinghofer, U. Krengel, J. John, W. Kabsch und E. F. Pai. Three-dimensional structure of p21 in the active conformation and analysis of an oncogenic mutant. *Environ Health Perspect*, 93:11–5, Jun 1991.