Study of a Highly Accurate and Fast Protein-Ligand Docking Algorithm Based on Molecular Dynamics

M. Taufer[1,2,3], M. Crowley[2], D. Price[2], A.A. Chien [1], C.L. Brooks III[2,3]

[1] Dept. of Computer Science and Engineering
University of California at San Diego
La Jolla, California 92093, U.S.
achien@ucsd.edu

[2] Dept. of Molecular Biology (TPC6)
The Scripps Research Institute
La Jolla, California 92037, U.S.
taufer,crowley,priced,brooks@scripps.edu

[3] Center for Theoretical Biological Physics
La Jolla, California 92093, U.S.

Abstract

Few methods use molecular dynamics simulations based on atomically detailed force fields to study the protein-ligand docking process because they are considered too time demanding despite their accuracy. In this paper we present a docking algorithm based on molecular dynamics simulations which has a highly flexible computational granularity. We compare the accuracy and the time required with well-known, commonly used docking methods like AutoDock, DOCK, FlexX, ICM, and GOLD. We show that our algorithm is accurate, fast and, because of its flexibility, applicable even to loosely coupled distributed systems like desktop grids for docking.

Keywords: *Force field based methods, docking accuracy, desktop grid computing.*

1 Introduction

A vast number of the essential roles that proteins play require small molecules to bind to specific spots in the protein structure. For instance, the small molecules can act as switches to turn on or off a protein function, or are the substrates for the particular chemical reaction that a protein enzyme catalyzes. Obtaining the atomic level details of the protein-ligand interactions is a valuable tool in the development of novel pharmaceuticals. Conventional experimental techniques for obtaining detailed structural information about protein-ligand complexes are time and resource intensive. As a result, much research effort has been focused on computational methods for the prediction of this difficult-to-obtain structural information. In general, this process is called docking.

Current docking algorithms typically use a fast, simplified scoring function to direct the conformational search and select the best structures. However, recent work has demonstrated that there are significant inaccuracies associated with these algorithms [1]. Furthermore, there are indications that inaccuracies can be reduced by using algorithms that use more sophisticated physics. For example, CDOCKER, a docking algorithm based on molecular dynamics (MD) and a conventional molecular mechanics force field, indeed provides better accuracy than other methods. However, it is still among the more compute-intensive methods. Our aim is to adapt the CDOCKER method to improve its performance without sacrificing the accuracy. In particular, we would like to take advantage of advances in computer technologies and the establishment of new distributed architectures, such as desktop grids.

Desktop grids provide a viable and inexpensive solution to the hitherto uncompetitive computational cost and time for the force field methods. New algorithms with finer computational granularity need to be developed, especially algorithms more suitable for the highly volatile nodes of desktop grids. In this paper we present an algorithm for the docking process based on MD simulations as in [2], but characterized by a higher flexibility that makes it adaptable to any computing platform, even

to very challenging desktop grids.

Docking many ligands to the same protein followed by scoring them for their relative strength of interaction has been proposed as a procedure to identify candidates for drug development. Screening large databases of compounds in this manner can potentially provide an alternative to conventional high-throughput screening, but it is not cost-effective unless the docking algorithm is fast and accurate.

Most of the well-known, commonly used docking methods that are not based on MD were compared and analyzed in detail in [3]. We validate the accuracy of our algorithm by applying the tests defined in [3] to our method, and compare to the published results for the following other methods: AutoDock [4], DOCK [5], FlexX [6], ICM [7], and GOLD [8]. We show that our algorithm is indeed more accurate than all other methods except ICM. We reach a docking success rate of over 70%, confirming the accuracy reported in [2]. The time required for running a complete docking attempt is longer but comparable with the time of the other methods. The fine computational granularity of our algorithm is trivially parallel and each simulation attempt is decomposable into independent sub-jobs. This flexibility makes our accurate docking simulations fast when there are many independent compute nodes, and thus, applicable to a wide range of platforms from traditional supercomputers to loosely coupled distributed systems like desktop grids.

In Section 2 we present our docking protocol as well as some well-known and commonly used docking algorithms that we will compare with our method. In Section 3 we define the metrics of accuracy and time that we use to validate our method while in Section 4 we compare our method with the other docking algorithms based on those metrics. Finally, in Section 5, we discuss the applicability of our docking protocol to desktop grids, and future work coming out of this research.

2 The MD-based Docking Algorithm

2.1 The CHARMM Scientific Computation Code

We use CHARMM to perform MD simulations and investigate the protein-ligand docking process. CHARMM is a program for simulating biologically relevant macromolecules (proteins, DNA, RNA) and complexes thereof [9]. It allows the investigation of the structure and dynamics of large molecules in solvent or crystals. CHARMM can be used to calculate free energy differences upon mutations or ligand binding [10]. One of the most common applications of CHARMM is MD, in which the Newtonian equations of motion are discretized and solved numerically by an integration procedure such as the Verlet algorithm. The force on the atoms is the negative gradient of the CHARMM potential energy function [11].

2.2 Modeling Protein-Ligand Interactions

Advances in energy calculation techniques make it viable to use a grid-based representation of the protein-ligand potential interactions to calculate our scoring function. A grid potential allows us to represent a rigid protein binding site as a potential field magnitude at each grid point. Protein interactions with the ligand in the binding site are interpolated from the interaction strength of the grid points near each atom of the ligand, rather than from computing the interactions of all ligand atoms with all protein atoms individually, resulting in orders of magnitude fewer floating-point computations than in the traditional molecular mechanics method.

In a preliminary phase of the docking simulations, we calculate three dimensional grid maps for each of the 20 potential atom types composing the ligands under investigation. Each grid map consists of a three dimensional lattice of regularly spaced points surrounding and centered on the active site of a protein. Each point within the grid map stores the potential energy of a 'probe' atom due to its interaction with the macromolecule. For example, in a carbon grid map, the value associated with a grid point represents the potential energy of a carbon atom at that location due to its interactions with all atoms of the protein receptor. We have chosen a grid spacing of 1Å based on previous work that showed no significant differences in docking accuracy for grid spacings between 0.25 Å and 1 Å [2].

To facilitate the penetration of small ligands into the protein sites and allow larger configurational changes, van der Waals (vdW) and electrostatic potentials with soft core repulsions [12] were utilized instead of the traditional potentials. A soft core repulsion reduces the potential barrier at vanishing interatomic distances to a finite limit. In this case, ligands can pass between conformational minima with a relatively small potential barrier that would normally be infinite and impassible with an unmodified potential.

2.3 MD Docking Protocol

As in most of the existing methods, we model the protein-ligand complex as composed of a rigid protein structure and a flexible ligand. A flexible ligand has three translational degrees of freedom, three rotational degrees of freedom and one dihedral rotation for each

rotable bond. The docking search is computed over a $6+n$ dimensional space where n is the number of rotable bonds in the ligand. Figure 1 shows the MD-based algorithm used for our docking simulations. One loop constitutes a docking trial. Given a protein and a ligand to dock into its binding site (a so-called protein-ligand complex), a docking attempt consists of a sequence of m independent trials. For each trial, a random configuration for the ligand is generated by running 1000 steps of MD at the constant temperature of 1000K in vacuum, starting from a reasonable structure with random initial velocities on each ligand atom. We have analyzed the distribution of torsional angles generated by this method, and found that they indeed vary randomly over the physically reasonable range for each rotable bond. We are confident that our initial configurations randomly sample the available configurational space of the ligand.

Starting from the new ligand configuration, a set of 10 different orientations are randomly generated and docked into the receptor, that is, moved into the center of the grid. Once the randomized ligand has been docked into the active protein site, we run a MD simulation consisting of a heating phase from 300K to 700K, followed by a cooling phase back to 300K. Finally, we refine the simulation result by running a short energy minimization. In the end, we use the energy of binding as the scoring function to rank the docked ligands and return the lowest energy structure as the solution to the docking trial. Twenty trials were run for each complex to ascertain the optimal number of trials that should constitute an attempt at docking.

2.4 The Other Docking Methods

Common search techniques for predicting binding affinities and geometries are based on genetic algorithms, chemistry, geometry of atoms, Monte Carlo or MD. Selection of best docked structures is performed using scoring functions belonging to three different categories: explicit force field scoring functions (as in our case), empirical scoring functions, or knowledge-based scoring functions. AutoDock, DOCK, FlexX, ICM and GOLD are well-known, commonly used programs which use a variety of search methods and scoring functions to address the study of protein-ligand docking. **AutoDock** [4] uses the Lamarckian genetic algorithm (LGA) by alternating local search with selection and crossover. The ligands are ranked using an energy-based scoring function and, to speedup the score calculation, a grid-based protein-ligand interaction is used. **GOLD** [8], like AutoDock, deploys a genetic algorithm and uses a scoring function which is the sum of energy

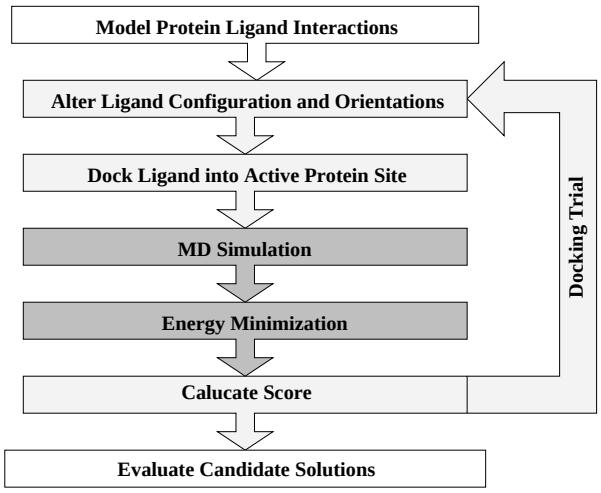

Figure 1. Our MD-based protein-ligand docking algorithm.

terms, some of which reflect the short-range vdW interaction between protein and ligand as well as the ligand internal energy. The search in **DOCK** [5] is driven by the geometry of the ligand in the active site. Different scoring functions can be employed: (1) geometric alignment and shape constraints, (2) the electrostatic potential of the protein-ligand complex using the program DELPHI, or (3) the energy of the protein-ligand complex under the AMBER force field. **FlexX** [6] is also driven by the geometry of the ligand in the active site like DOCK. In FlexX, the scoring uses a variation of the Böhm scoring function with terms for several kinds of interactions and penalty functions for the deviations from ideal interaction geometries. **ICM** [7] uses a Monte Carlo minimization on the internal coordinates to find the global minimum of the scoring function. The scoring function used to rank placements of ligands relative to one another takes into account the force field energy of the ligand and the protein-ligand interaction energy.

3 Metrics

3.1 Accuracy

The accuracy of any given docking attempt is measured by the root-mean-square-deviation (RMSD) of all non-hydrogen ligand atoms between the lowest-energy structure from the docking attempt and the ligand's position in the crystal structure. For many of the ligands studied here, a dihedral rotation can result in a ligand conformation that is geometrically and chemically indistinguishable, but with a different RMSD relative to the experi-

Protein	Protein-Ligand Complex PDB Entry
Trypsin	3ptb(3), 1tng(2), 1tnj(3), 1tnk(4), 1tni(5), 1tpp(7), 1pph(11)
Cytochrome $P450_{cam}$	1phf(1), 1phg(5), 2cpp(3)
Neuraminidase	1nsc(12), 1nsd(11), 1nnb(11)
Carboxypeptidase	1cbx(5), 3cpa(8), 6cpa(16)
L-Arabinose	1abe(4), 1abf(5), 5abp(6)
e-Thrombin	1etr(15), 1ets(13), 1ett(11)
Thermolysin	3tmn(10), 5tln(14), 6tmn(20)
Penicillopepsin	1apt(30), 1apu(29)
Intestinal FABP	2ifb(15)
Carbonic Anhydrase II	1cil(6), 1okl(5), 1cnx(13)

Table 1. Data set of the 31 protein-ligand complexes used for our experiments.The number of rotable bonds for each ligand is reported beside the complex name.

mentally determined structure. That is, the RMSD between a docking attempt and the crystal structure using a one-to-one mapping of atoms may or may not accurately measure the quality of the docking attempt. Consequently, we have exhaustively calculated the RMSD of all degenerate conformers related by the rotation of all symmetry-conferring dihedral angles. The lowest RMSD obtained from this search is guaranteed to be the correct RMSD for the structure.

Reference [3] provides an additional measure describing the frequencies where high-quality docking solutions are found. For many docking attempts, the docking accuracy (DA) can be defined as follows:

$$DA = f_{RMSD \leq 2} + 0.5(f_{RMSD \leq 3} - f_{RMSD \leq 2}) \quad (1)$$

where $f_{RMSD \leq a}$ is the fraction of docking attempts that produce structures with an RMSD relative to the experimental structure of a Angstroms.

3.2 Computational Time

In order to compare the performance of our docking algorithm with the other methods and study its applicability to different compute platforms, we look at the CPU time required for completing a docking attempt on a single node. In the case of the docking methods reported in [3], the length of a docking attempt was controlled by the default or recommended parameter settings of the specific docking algorithm. For our MD-based method, we consider the time to complete a set of docking trials and report CPU time for sets of 1, 10 and 20 trials.

4 Simulation Results

4.1 Testbed Characterization

All the docking simulations for the methods presented in Section 2.4 were performed on an SGI R10000 equipped with a single 195 MHz IP2 processor and 128MB memory. We use the same machine for the measurement and comparison of the time required for completing a single attempt. For the investigation of the accuracy and, in particular, for the investigation of the optimal number of trials per attempts to reach an acceptable DA, we run our several simulations on a cluster of 64 dual-processor nodes at the San Diego Supercomputer Center (SDSC) at UCSD equipped with 930MHz Pentium III processors.

4.2 Characterization of our Docking Simulations

We run our docking simulations on a data set of 31 protein-ligand complexes, all of the complexes used in [3] that are present in the Ligand Protein DataBase (LPDB) [13]. The criteria for choosing the protein-ligand complexes in [3] are that the proteins under investigation have at least two entries with different ligands in the PDB (with the exception of the Intestinal FABP), and that no protein-ligand covalent bonds are present. Table 1 shows the list of the ten proteins and their ligands. The ligands have different structures and numbers of rotable bonds, ranging from 1 to 30. The number of ligand rotable bonds is reported next to each complex in the table. We consider four different cases each with a different number of MD steps for the heating and cool-

ing phases. Table 2 shows the four cases and the associated number of 1 fsec MD steps. Figure 2 shows the DA of our MD-based method with different number of trials per attempt and different lengths for the MD simulation (each case is reported in Table 2). In the figure we label each attempt with *Ti* where *i* is the number of independent trials per protein-ligand docking attempt (*i* ranges from one to twenty). By looking at the data reported in Figure 2, we conclude that we need about 10 trials per attempt to reach a docking accuracy of 70%.

Case	Heating Phase # MD steps	Cooling Phase # MD steps
Case A - 1K2.5K	1000	2500
Case B - 2K5K	2000	5000
Case C - 4K10K	4000	10000
Case D - 8K20K	8000	20000

Table 2. The four different MD simulations, each with a different number of MD steps for the heating and cooling phases.

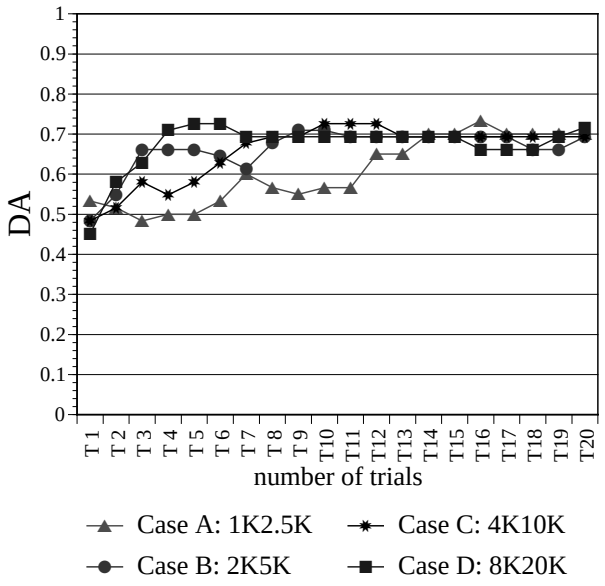

Figure 2. Docking Accuracy (DA) for different number of trials per docking attempt and with different number of MD steps per simulation.

Figure 3 shows the average time in seconds per trial and with different number of MD steps per simulation as reported in Table 2. As expected, the increase of number of MD steps during the heating and cooling phases causes an almost linear increase of the simulation time.

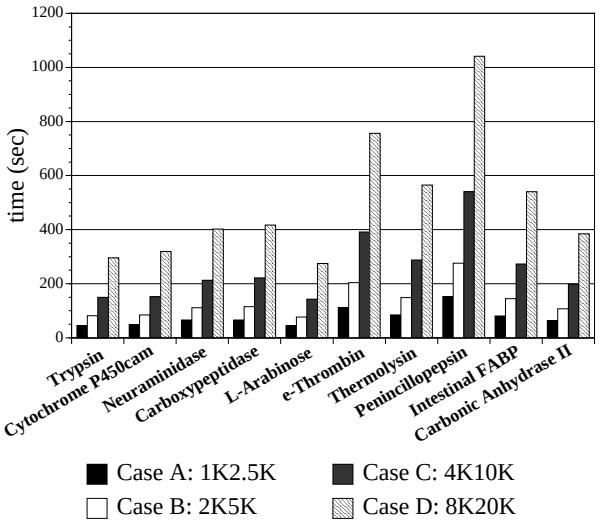

Figure 3. Average time in seconds per trial and with different number of MD steps per simulation.

We have run several experiments with 10 and 20 trials and have confirmed that the results shown in Figure 2 and Figure 3 are repeatable (data not shown). For our comparison in the rest of the paper we use *Case B* as a reference case for which we run 2000 MD steps during the heating phase, and 5000 MD steps during the cooling phase. Each MD step consists of 1 fsec time step. The DA, RMSD and time values for the other methods reported in Section 2.4 and used in our comparisons in the rest of this paper are from the previous work of our group [3].

4.3 Comparison of the Docking Accuracy (DA)

Figure 4 compares the DA of the well-known methods with the DA of our MD-based method for *Case B* in which each MD simulation consists of 2000 heating MD steps and 5000 cooling MD steps. By looking at the data reported in Figure 4, we observe that our method provides better DA than all the other methods, except ICM. ICM employs an algorithm which improves convergence by using an analytical gradient minimizer and running multiple Monte Carlo minimizations from several starting configurations. We plan to make a more detailed study of MD and Monte Carlo simulations for the docking process in the near future.

Protein-ligand Complex	# rotable bonds	AutoDock	DOCK	FlexX	ICM	GOLD	T10	T20
3ptb	3	0.80	0.59	1.11	0.49	1.09	0.56	**0.54**
1tng	2	**0.62**	0.86	1.08	0.71	1.89	0.70	0.69
1tnj	3	**1.21**	1.56	1.73	2.17	1.90	1.42	1.50
1tnk	4	1.69	1.87	1.70	2.53	3.08	1.16	**1.14**
1tni	5	2.61	5.26	2.73	3.40	4.93	**2.22**	**2.22**
1tpp	7	1.80	3.25	1.95	**1.71**	2.33	2.43	2.53
1pph	11	5.14	3.91	3.27	1.44	4.23	4.00	**0.53**
1phf	1	2.09	2.39	4.68	1.23	4.42	**1.20**	**1.20**
1phg	5	3.52	5.57	4.87	**0.46**	4.20	1.07	1.08
2cpp	3	3.40	2.48	**0.44**	2.53	3.49	3.26	3.27
1nsc	12	1.40	4.86	6.00	1.80	**1.02**	1.47	1.40
1nsd	11	1.20	4.51	1.56	**1.04**	0.96	1.85	1.85
1nnb	11	0.92	4.51	0.92	1.09	**0.84**	1.67	3.97
1cbx	5	1.33	3.13	1.32	0.82	1.87	**0.62**	**0.62**
3cpa	8	2.22	6.48	1.51	**0.77**	1.87	2.22	2.22
6cpa	16	8.30	8.30	9.83	**1.60**	4.96	4.00	4.00
1abe	4	**0.16**	1.87	0.55	0.36	0.18	0.56	0.56
1abf	5	**0.48**	3.25	0.76	0.61	0.50	0.68	0.70
5abp	6	**0.48**	3.89	4.68	0.88	0.59	**0.48**	0.51
1etr	15	4.61	6.66	7.26	**0.87**	5.99	1.09	1.09
1ets	13	5.06	3.93	2.11	6.22	2.39	**1.97**	**1.97**
1ett	11	8.12	1.33	6.24	0.99	1.30	**0.82**	**0.82**
3tmn	10	4.51	7.09	5.30	**1.36**	3.96	3.65	3.65
5tln	14	5.34	1.39	6.33	1.42	1.60	**1.21**	**1.21**
6tmn	20	8.72	7.78	4.51	2.60	8.54	**2.21**	**2.21**
1apt	30	1.89	8.06	5.95	**0.88**	8.82	5.72	4.79
1apu	29	9.10	7.58	8.43	2.02	10.70	**1.32**	**1.32**
2ifb	15	3.09	1.43	8.94	**1.04**	2.61	2.09	5.19
1cil	6	5.81	2.78	3.52	2.00	6.04	**1.86**	**1.86**
1okl	5	8.54	5.65	4.22	3.03	3.55	**2.84**	**2.84**
1cnx	13	10.9	7.35	6.83	**2.09**	6.32	6.20	6.20

Table 3. Comparison of best RMSD for different docking methods. The best RMSD is the RMSD of the predicted ligand from the Xray structure. For each protein-ligand complex, the best RMSD found is reported in bold.

4.4 Comparison of RMSD for the Different Docking Methods

The RMSD's reported in Figure 3 are the root mean square deviations of the heavy atoms of the predicted ligands from the corresponding ligands in their published complex crystal structures. For our MD-based docking, we present results of attempts with different numbers of trials: $T10$ with 10 trials per attempt and $T20$ with 20 trials per attempt. In general, we observe that for both $T10$ and $T20$, we get, on the average, lower RMSD than the other methods.

4.5 Comparison of Simulation Time for the Different Docking Methods

The main question we want to address in Table 4 is whether the high level of accuracy is also supported by competitive execution time when compared with the execution times of the other docking methods. Table 4 shows the average CPU time to complete a protein-ligand docking for the ten proteins in Table 1 and for

Protein	AutoDock	DOCK	FlexX	ICM	GOLD	T1	T10	T20
Trypsin	391	51	26	65	165	81	805	1610
Cytochrome $P450_{cam}$	291	29	82	40	273	84	846	1693
Neuraminidase	620	98	72	99	269	111	1110	2220
Carboxypeptidase	624	88	92	147	437	115	1156	2313
L-Arabinose	353	37	31	39	288	76	766	1533
e-Thrombin	1174	421	83	336	676	203	2036	4073
Thermolysin	789	170	65	238	500	148	1483	2966
Penicillopepsin	1122	412	77	645	840	276	2760	5520
Intestinal FABP	560	138	29	234	489	145	1450	2900
Carbonic Anhydrase II	519	55	88	92	388	107	1070	2140

Table 4. Comparison of average time simulations for different proteins and different docking methods.

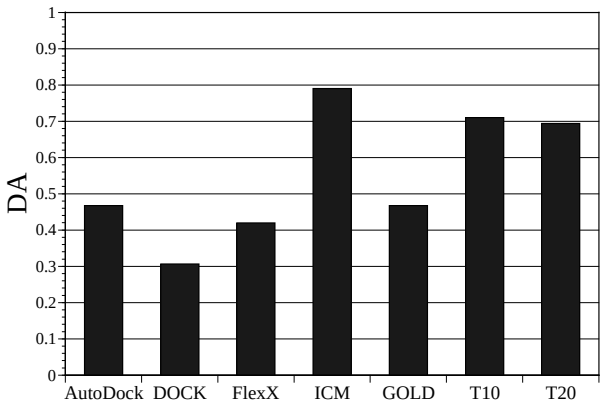

Figure 4. Comparison of docking accuracy (DA). The docking accuracy is the weighted sum of the fraction of docked attempts with acceptable accuracy (lower or equal to 2Å and 3Å).

the different methods under investigation. For our MD-based docking method, we consider the average time of a single trial as well as the time for an attempt of 10 and 20 trials. Again we consider the *Case B* in Table 2 as a reference case. We observe that an attempt of 10 trials is completed in less than one hour even for complex protein-ligand docking with a large number of rotable degrees of freedom. In addition, each trial of each attempt is independent, and therefore, the 10 trials can run at the same time on different processors in parallel. If enough processors are available, the time for completing a protein-ligand docking becomes the time for a single trial, making our algorithm highly competitive with the other methods.

5 Computational Platforms for our MD-based Docking

MD simulations are time-consuming but are also accurate general techniques for the study of protein-ligand docking. The time needed by MD-based algorithms to screen large sets of ligands (of the order of 10,000 molecules) makes this approach prohibitive even on expensive supercomputers. On non-dedicated systems, even the docking of a single protein-ligand complex might result in a time-to-solution on the order of hours due to computing resource contentions. The motivation to port existing applications to more cost-effective distributed systems like desktop grids is not strong for such applications unless more time-effective algorithms are designed and implemented.

The need for new algorithms that are more flexible and suitable for desktop grids, but still accurate, is the motivation behind our search for the docking algorithm presented in this paper. Docking attempts of our MD-based algorithm consist of sequences of independent trials. We have observed and measured that attempts for even complex ligands with a large number of rotable bonds are characterized by short simulation times, much shorter than 1 hour. By decomposing each attempt into sets of independent trials, we can further increase the computational granularity of the algorithm. Using available desktop PC's simultaneously to process each trial, proportionally decreases the time to solution. Long computation tasks, which are more probable to be interrupted by annoyed desktop users, should also be avoided. Our result shows that we can ensure the time to solution to be equal to the time for a single trial when a large number of

desktop PCs is available. An acceptable accuracy can be ensured by sending out more trials than are needed for the desired accuracy, and using the first trials to complete.

Therefore, we conclude that our docking algorithm is well-suited for Intranet desktop grid platforms (e.g., Entropia DCGrid [14], Infuzion [15]) and on the Internet (e.g., XtremWeb [16], BOINC [17]). The combination of our algorithm with such platforms, which might allow us to perform fast and accurate screening of very large ligand databases, is currently under our development and investigation.

6 Conclusion

In this paper we present a MD and detailed force field protein-ligand docking algorithm based on a grid representation of the protein-ligand interactions and soft-core potential. We prove that our docking method provides better docking accuracy than most of the other well-known and commonly used docking techniques, displaying a successful docking rate of 70%.

Based on our time comparisons, we claim that the computational time is no longer a justified reason to avoid using detailed force field based docking techniques. Even for complex ligands, the completion time for a protein-ligand docking attempt of 10 trials is modest (less than one hour on a 930MHz processor for ligands with large numbers of rotable bonds). Desktop grid platforms are well-suited for our accurate, fine-grained parallel algorithm for which each docking trial is short and independent.

Acknowledgments

Financial support from the National Institutes of Health Grant, GM37554, is greatly appreciated. Financial support through the NSF sponsored Center for Theoretical Biological Physics (grants# PHY-0216576 and 0225630) and the LJIS Fund are also acknowledged. Supported in part by the National Science Foundation under awards NSF EIA-99-75020 Grads and NSF Cooperative Agreement ANI-0225642 (OptIPuter), NSF CCR-0331645 (VGrADS), NSF ACI-0305390, and NSF Research Infrastructure Grant EIA-0303622. Support from Hewlett-Packard, BigBangwidth, Microsoft, and Intel is also gratefully acknowledged. We wish to thank the San Diego Supercomputer Center at UCSD for giving us access to the Meteor cluster.

References

[1] P. Ferrara, H. Gohlke, D. Price, G. Klebe, and C.L. Brooks III. Assessing Scoring Functions for Protein-Ligand Interactions. *J. Med. Chem.*, 2003. Submitted.

[2] G. Wu, D.H. Robertson, C.L. Brooks III, and M. Veith. Detailed Analysis of Grid-Based Molecular Docking: A Case Study of CDOCKER-A CHARMm-Based MD Docking Algorithm. *J. Comput. Chemistry*, (24):1549–1562, 2003.

[3] B.D. Bursulaya, M. Totrov, R. Abagyan, and C.L. Brooks II. Comparative Study of Several Algorithms for Felxible Ligand Dockign. *J. Comp. Aided Molecular Design*, 2003. in press.

[4] G.M. Morris, D.S. Goodsell, R.S. Halliday, R. Huey, W.E. Hart, R.K. Belew, and A.J. Olson. Automated Docking Using a Lamarkian Genetic Algorithm and and Empirical Binding Free Energy Function. *J. Comp. Chem.*, 19:1639–1662, 1998.

[5] T.J.A. Ewing and I.D. Kuntz. Critical Evaluation of Search Algorithms for Automated Molecular Docking and Database Screening. *J. Comp. Chem.*, 18:1176–1189, 1997.

[6] M. Rarey, B. Kramer, T. Lengauer, and G.A. Klebe. A Fast Flexible Docking Method using an Incremental Construction Algorithm. *J. Mol. Biol.*, 261:470–489, 1996.

[7] R. Abagyan, M. Totrov, and D. Kuznetsov. ICM - A New Method for Protein Modeling and Design: Applications to Docking and Structure Prediction from the Distorted Native Conformation. *J. Comp. Chem.*, 15:488–506, 1994.

[8] G. Jones, P. Willett, R.C. Glen, A.R. Leach, and R. Taylor. Development and Validation of a Genetic Algorithm for flexible Docking. *J. Mol. Biol.*, 267:727–748, 1997.

[9] B. R. Brooks, R. E. Bruccoleri, B. D. Olafson, D. J. States, S. Swaminathan, and M. Karplus. CHARMM: A Program for Macromolecular Energy, Minimization, and Dynamics Calculations. *J. Comput. Chem.*, 4:187–217, 1983.

[10] X. Kong and C.L. Brooks. lambda-dynamics: A new Approach to Free Energy Calculations. *J. Chem. Phys*, 105:2414–2423, 1996.

[11] A.D. MacKerell Jr. and et al. All-atom empirical Potential for Molecular Modeling and Dynamics Studies of Proteins. *J. Phys. Chem. B*, 102:3586–3616, 1998.

[12] M. Vieth, J.D. Hirst, A. Kolinski, and C.L. Brooks III. Assessing Energy Functions for Flexible Docking. *J. Comp. Chemistry*, 19(14):1612–1622, 1998.

[13] C.L. Brooks III and et al. LPDB: Ligand Protein DataBase. lpdb.scripps.edu/.

[14] Entropia. www.entropia.com.

[15] EnFuzion. www.axceleon.com.

[16] G. Fedak, C. Germain, Néri V., and F. Cappello. XtremWeb : A Generic Global Computing System. In *Proc. of CCGRID 2001, Workshop on Global Computing on Personal Devices*, May 2001.

[17] BOINC. boinc.berkeley.edu.