Palmi E, Computer Science, University of Minnesota
Devaurs D, Department of Computer Science, Rice University
Kavraki L, Department of Computer Science, Rice University

Evaluation of DINC 2.0: An Incremental Docking Protocol for Large Ligands

Introduction

The ability to computationally predict the binding structure of protein-ligand complexes is critical in the field of drug development. Computational docking tools exist that are able to accurately predict the binding modes of protein-ligand complexes but accuracy decreases as the size of the ligand increases. This problem is of particular concern in the development of individualized immunotherapy treatments for cancer patients, which rely on predicting the structure of the complex resulting from human leukocyte antigen (HLA) proteins binding to peptides produced by tumor cells. These peptides are large in size and existing docking tools fail to predict the peptide-HLA complex binding structure [1].

The Kavraki Lab has developed a novel computational docking tool, known as DINC 2.0, to better predict the binding modes of these challenging complexes. DINC 2.0 relies on a docking protocol that allows existing docking tools to dock fragments of a large peptide incrementally. An array of docking tools exists; our lab has used both Autodock4 and Vina and has determined that Vina generally yields better results. We have used Vina as the docking tool in the experiments described in this report. Initially, the ligand is split into a user-specified number of fragments. In each round of docking the fragment is expanded to include the atoms from the subsequent fragment. At each round only a select number of bonds are considered flexible. Vina will find multiple binding modes and the best n-modes are selected to use as the basis for the next round of docking (n corresponds to the DINC parameter “number of docking tasks”). The docking rounds proceed until the whole ligand has been reassembled and docked.

Our research aims at evaluating DINC 2.0 on two datasets of protein-peptide complexes from the literature. DINC is evaluated on how well it is able to reproduce the structure of the known bound complex. Our goal is to identify a protocol for parameters that consistently generates accurate binding structures for an arbitrary complex. In particular the experiments described in this report focus on evaluating the effects of different values for the following parameters: exhaustiveness, number of docking tasks and fragment size. Achieving consistency is a critical step towards the capability of modeling the structure of unknown peptide-HLA complexes.

Methods

We selected two datasets from the literature to evaluate in our benchmarking experiments (known as the Renard dataset and the LEADS dataset). These datasets were selected because they are known to contain complexes that are particularly challenging for docking software to model.

We conducted several experiments in order to identify the effects of different parameter values. The parameters of the program that we examined included: exhaustiveness, the number of docking tasks and the size of the molecular fragment at each step of docking. Exhaustiveness is a parameter tied to the underlying docking software, Vina, which expresses how thoroughly Vina explores the conformation search space while docking a molecular fragment. The number of docking tasks refers to how many molecular conformations are produced at each round of docking. The greater the number of docking tasks, the greater the chance that a good result is produced. Finally, the size of the molecular fragment used in docking is related to the incremental aspect of DINC. Since DINC is an incremental docking protocol, at each step of docking only a fragment of the whole ligand is docked using the molecular docking tool (Vina in this case).

Our first experiment explored the effects of setting the exhaustiveness parameter at 8 (the recommended best value from Vina) and 100 (an arbitrarily high value) in order to determine which would provide the best results and what the computational cost of increasing exhaustiveness would be. For the second experiment we compared the results obtained using 3, 6, 12, 24 and 36 docking tasks, but without involving the incremental process (i.e., by having a fragment size larger than the ligand size). For the third experiment we compared the results obtained using fragment sizes starting at 6 degrees of freedom and increasing in increments of 6 up to the degrees of freedom of the largest ligand in the dataset (e.g. 6, 12, 18, 24…) [1]. We performed each of these experiments on both datasets.

While the ultimate goal is to be able to use DINC to consistently model the structure of unknown protein-ligand complexes, in order to evaluate how well a protocol works we ran DINC on known protein-ligand complexes

and calculated the root-mean-squared-deviation (RMSD) between the computationally identified binding mode and the known binding mode. The RMSD was measured in angstroms (Å).

Results

Experiment 1: Varying Exhaustiveness

Table 1. Effect of Exhaustiveness- Renard and LEADS Dataset

Average RMSD (Å)		
Exhaustiveness	Renard	LEADS
8	4.55	4.95
100	4.39	4.26

Over the entire Renard dataset the average RMSD for all complexes using exhaustiveness 8 was 4.55 Å. With exhaustiveness 100 that number improved to 4.39 Å. This is a 0.16 Å decrease in RMSD, which required 4 to 5 times more computation. Over the entire LEADS dataset the average RMSD for all complexes using exhaustiveness 8 was 4.95 Å. With exhaustiveness 100 that number improved to 4.26 Å. This is a 0.69 Å decrease in RMSD.

Experiment 2: Varying Number of Docking Tasks

Figure 1. Number of Docking Tasks- Renard Dataset

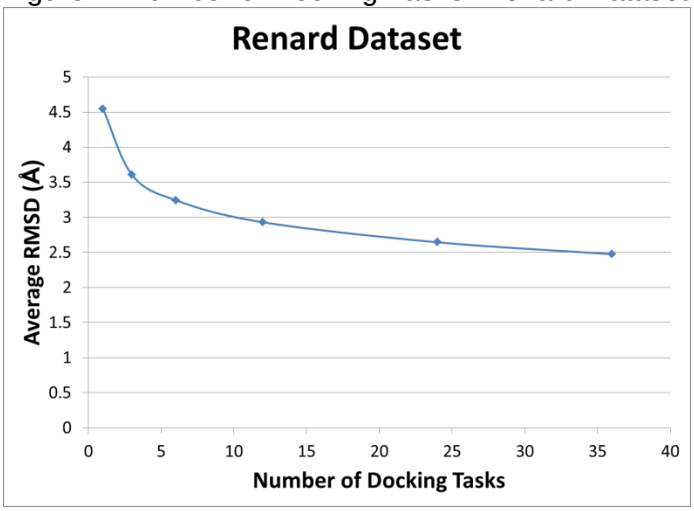
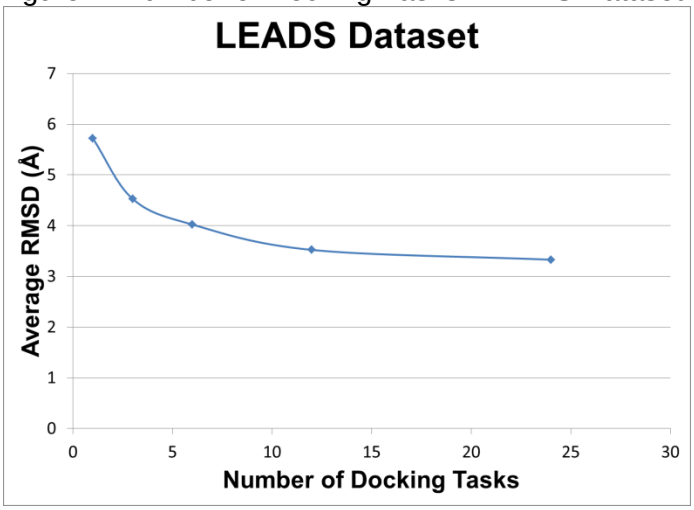


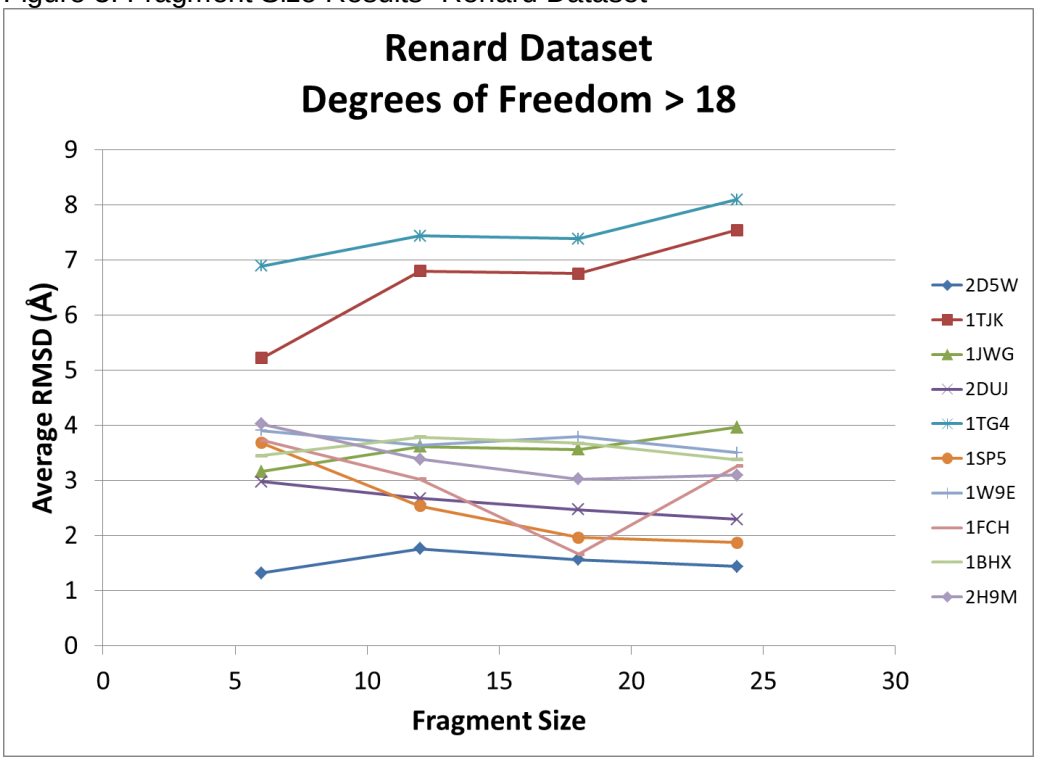
Figure 2. Number of Docking Tasks- LEADS Dataset



In Figure 1 and Figure 2 we see that as we increase the number of docking tasks used by DINC, the average RMSD over the entire dataset decreases. This behavior is exhibited by both the Renard and the LEADS dataset. However, especially for the LEADS Dataset, we see that there is not much improvement in RMSD moving from 12 docking tasks to 24 docking tasks. The Renard Dataset does not experience as dramatic a plateau between 12 docking tasks and 24 docking tasks.

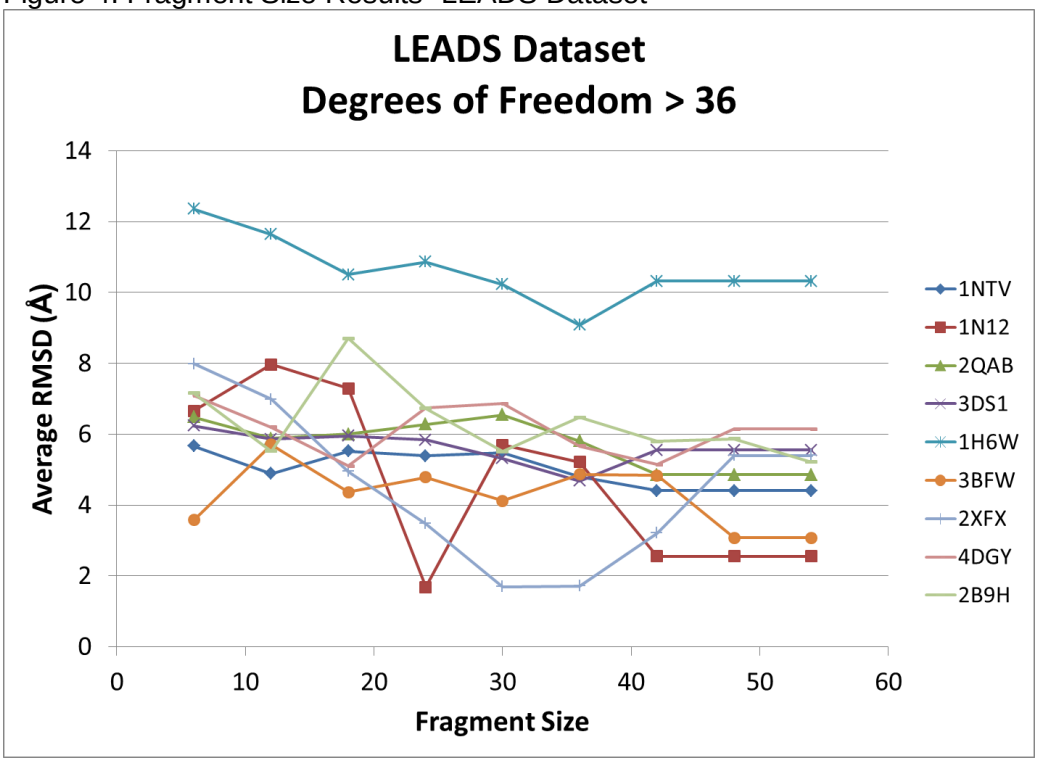
Experiment 3: Varying Fragment Size

Figure 3. Fragment Size Results- Renard Dataset



Complexes in the Renard Dataset with more than 18 degrees of freedom (the category with the largest ligands) can be categorized into two groups based on their behavior with different fragment sizes. The first category includes those complexes for which a fragment size of 6 is optimal. The second category includes complexes for which it is better to consider the ligand as a whole instead of in fragments (fragment size 24).

Figure 4. Fragment Size Results- LEADS Dataset



The LEADS Dataset contains large complexes than the Renard dataset contains. Figure 4 includes complexes with more than 36 degrees of freedom. The behaviors in Figure 4 appear more chaotic than those in Figure 3. This is not surprising, since results are a subset of conformations produced by the docking software. For the experiments reported in Figures 3 and 4 only 6 docking tasks were used, so variability is to be expected. Figure 4 shows that for most of the complexes with more than 36 degrees of freedom docking the ligand as a whole is better than using any smaller fragment size. However, there are several complexes for which the minimum is achieved at a fragment size somewhere between 6 degrees of freedom and 54 degrees of freedom.

Discussion

As demonstrated by the results summarized in Table 1, increasing exhaustiveness did not significantly improve the results found for either the Renard or the LEADS dataset. We determined that the slight improvements in RMSD results found using a high exhaustiveness are not worth the significant increase in computation cost.
In our second experiment we examined the effect of increasing the number of docking tasks on the results obtained for individual complexes. For many of the complexes increasing the number of docking tasks decreased the RMSD significantly. However, several of the complexes experienced little improvement in RMSD. These more challenging complexes were of particular interest going into our third experiment.
In the third experiment we examined the effect of fragment size on the success of the incremental approach in docking challenging complexes. The results did not allow us to identify one “magic bullet” number for fragment size that worked best for all complexes.

Conclusion

The results of the three experiments provided several important insights into how to optimize the parameters of DINC to consistently produce good results. Increasing the exhaustiveness to 100 did not yield a significant improvement in average RMSD. As a result, all further experiments were run with exhaustiveness 8 (the Vina recommended value). The more docking tasks that are used the better results we see, although the average RMSD results begin to experience diminishing returns around 12 docking tasks for the LEADS Dataset. We also determined that there is not one fragment size that will work best for all complexes. However, the results did guide us toward several future directions for further experimentation. One is for the development of several protocols, which when run as a battery will consistently yield good results for any complex. Another is to change the fragment size used based on the size of each individual complex.

References

[1] Dhanik A, McMurray JS, Kavraki LE. DINC: A new AutoDock-based protocol for docking large ligands. BMC Structural Biology. 2013 Nov [cited 2018 Jul 26]; Available from: http://kavrakilab.org/publications/dhanik-mcmurray2013dinc-new-autodock-based.pdf
[2] DINC – Docking INCrementally [Internet]. Houston: Rice University; c2018 [cited 2018 Jul 26]. Available from: http://dinc.kavrakilab.org

Funding Acknowledgement: This work is supported in part by the Distributed Research Experiences for Undergraduates (DREU) program, a joint project of the CRA Committee on the Status of Women in Computing Research (CRA-W) and the Coalition to Diversify Computing (CDC), which is funded in part by the NSF Broadening Participation in Computing program (NSF BPC-A #1246649). This research was indirectly sponsored (not funded) by the CPRIT Summer Undergraduate Program in Computational Cancer Biology, training grant award RP 170593 from the Cancer Prevention & Research Institute of Texas (CPRIT).