Molecular Tetris: Crowdsourcing Molecular Docking Using Path-Planning and Haptic Devices

[1]Torin Adamson, [1]John Baxter, [1]Kasra Manavi, [1]April Suknot, [2,1]Bruna Jacobson, [1]Patrick Gage Kelley, [1]Lydia Tapia
{toriadam,johnbaxter,kazaz,asuknot,bjacobson,pgk,tapia}@cs.unm.edu

[1]Department of Computer Science, University of New Mexico
[2]Department of Physics and Astronomy, University of New Mexico

Abstract

Many biological processes, including immune recognition, enzyme catalysis, and molecular signaling, are associated to the binding of two or more proteins. This molecular docking is still an open problem in biological sciences. We present Molecular Tetris, a game in which a player can explore the binding between a protein receptor and ligand. This exploration is similar to the game Tetris with atomic forces guiding best fits between shapes. This game will be utilized for crowdsourced haptic-guided motion planning. Haptic touch devices enable users to feel the interactions of two molecules as they move the ligand into an appropriate binding site on the receptor. We demonstrate the method on a critical piece of human immune response, ligand binding to a Major Histocompatibility Complex (MHC) molecule. Through multiple runs by our users, we construct a global roadmap that finds low energy paths to molecular docking sites. These paths are comparable to a highly-biased roadmap generated by Gaussian sampling around the known bound state. Our users are able to find low energy paths with both a specialized force-feedback device and a commodity game console controller.

CR Categories: J.3 [Computer Applications]: LIFE AND MEDICAL SCIENCES—Biology and genetics

Keywords: probabilistic roadmap methods, molecular docking, haptics, crowdsourcing

1 Introduction

Many biological processes rely on the interactions between small-molecule ligands and large protein receptors. Ligand-receptor binding prediction is therefore critical to biochemical engineering. Ligands bound in a low energy configuration are said to be *docked*. Binding affinities are controlled by the energetic feasibility of transition from undocked to docked configurations. Motion planning techniques can predict interactions between molecules efficiently without falling into local minima traps [Cortés et al. 2005] while screening techniques consider the validity of final configurations.

Molecular docking methods are more efficient than molecular dynamics (MD) methods for predicting ligand-receptor binding [Jones et al. 1997]. Efficiency from reduced-complexity models make interactive components (e.g., haptics) feasible. Molecule docking tools can keep molecules rigid during simulation [Morris et al. 2009]. While such simplifications may prevent identification of properly docked configurations, they enable interactivity.

MIG '14, November 06 - 08 2014, Playa Vista, CA, USA

ACM 978-1-4503-2623-0/14/11 ... $15.00
http://dx.doi.org/10.1145/2668064.2668086

Haptic-based molecular docking simulators can be efficient tools for studying molecular interactions and docking. In this work we develop a method to generate docking pathways using motion planning techniques integrated with realtime haptic simulations, similar to previous realtime haptic adaptations running on commodity hardware [Bolopion et al. 2010]. Probabilistic Roadmap Methods (PRMs) have been applied to motion planning problems with high-dimensional configuration space (e.g., protein folding, docking) [Al-Bluwi et al. 2012]. We construct a roadmap of feasible motions by sampling the configuration space and connecting configurations with weighted edges that indicate the feasibility of moving between configurations. Configurations with the ligand bound to the receptor are low in potential energy, allowing for the use of potential energy to test for feasibility.

Crowdsourcing utilizes the public to generate a large set of data, and in some applications as users become more familiar with the software, they can improve their performance. Finding solutions to biological problems using crowdsourcing has been successful. Popular programs such as Foldit [Khatib et al. 2011] and Folding@home [Beberg et al. 2009] have been applied to protein folding. To our knowledge, crowdsourcing has not been applied to ligand binding.

In this study, we construct pathways of ligands moving into binding sites generated using two different methods of construction. The first method utilizes the Molecular Tetris game (a preliminary version appears in [Adamson et al. 2014]) where players operate a haptic device with force-feedback or a game controller with vibrational feedback to explore the configuration space. Multiple player inputs are combined to generate a roadmap that can be traversed to identify feasible paths. The second method uses a Gaussian distribution sampling to generate configurations, centered around a known docked configuration of the ligand. As this method is biased towards a pre-identified, e.g., native, binding site, it provides us with a "best-case scenario" result that we can use to compare to the outcomes from Molecular Tetris.

This novel approach incorporates a reduced polygon model for visualization efficiency, intuitive docking, and small data storage, while computing energy and forces from the all-atom structures. These simplifications along with multi-threading allow for fast real-time use on commodity hardware and an intuitive, gaming experience. The force feedback devices used in this work are the Novint Falcon®, and an Xbox 360 controller.

2 Related Work

2.1 Molecular Docking

Molecular simulations and physical experiments are costly and time-consuming, so fast accurate methods for sampling and scoring ligand-receptor binding candidates help prioritize limited scientific resources. A wide array of molecular docking algorithms and tools have been developed (see [Huang and Zou 2010]). Early dock-

ing tools used ligand-receptor cavity geometry complementarity to both guide sampling and score feasibility [Kuntz et al. 1982]. Later tools incorporate atomic force fields into scoring [Moustakas et al. 2006]. Methods to discover and measure ligand/cavity complementarity have also become more sophisticated [Chen and Honig 2010]. Relative to rigid body approximations, accounting for ligand and receptor flexibility greatly increases docking problem dimensionality. Techniques applied to the problem of docking with flexible molecules include incremental construction by tree search [Rarey et al. 1996], genetic algorithms [Jones et al. 1997], and Monte Carlo energy minimization [Meiler and Baker 2006]. Receptor flexibility may also be handled as ligand binding to rigid receptor ensembles [ClauBen et al. 2001]. Most docking tools automate configuration sampling; in contrast, our method utilizes user guidance to discover feasible ligand trajectories. The force field used to score samples additionally provides haptic feedback informing the user's search. We present a system that merges configurations discovered by separate users to produce better quality trajectories.

2.2 Molecular Docking With Haptics

It has been shown that haptic devices enhance the operator's intuition and understanding of molecular binding processes [Bivall et al. 2011]. They can enable the user to guide automated docking algorithms with *hints* from user ligand configurations combined with existing configurations into a single roadmap [Bayazit et al. 2001]. Haptic feedback has also been incorporated into docking simulations with adaptive user control of the flexibility/performance tradeoff [Bolopion et al. 2010]. Haptic controls directly manipulating ligand positions have been compared against force-based control schemes [Bolopion et al. 2010]. These devices have also been applied to the control of probe objects such as water molecules to discover solvent accessible locations [Stocks et al. 2009].

Molecular docking tools developed for different device capabilities include 3-DOF force feedback and 6-DOF force/torque feedback [Hou and Sourina 2011]. Collaborating users manipulating separate molecules has been considered [Hou et al. 2014]; our method, by contrast, synthesizes configuration sampling performed independently by multiple users into ligand binding trajectories.

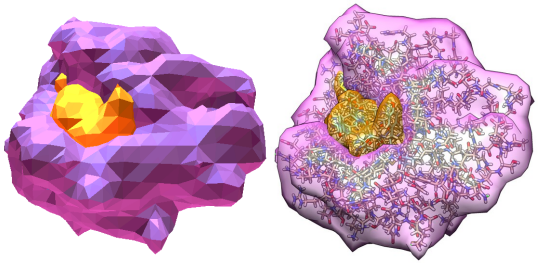
We implement haptic force-feedback as the gradient of potential energy. We scale, and time-smoothe forces to prevent unstable haptic feedback. Unlike [Hou et al. 2014], there are no dead zones; haptic feedback always reflects the potential energy gradient. Responsiveness is maintained on commodity hardware by computing potentials and haptic-feedback on separate threads. A complex energy model is handled similarly in realtime in [Daunay et al. 2007], but was not performed on commodity hardware.

2.3 Motion Planning With Haptics

Haptics are well suited for integration with motion planning problems and have been used to give hints to planners [Bayazit et al. 2001; He and Chen 2009]. These devices can also be used to guide or train users [Vázquez et al. 2010; Lawitzky et al. 2012]. In [Vázquez et al. 2010], Kautham path planning (configuration space cellular decomposition/classification) is used to generate a local channel (path). This local channel is used to generate haptic force feedback to aid the user during execution of a task. In our method force on the ligand is felt since the potential energy gradient generates the haptic force-feedback. The user makes the decisions for ligand movement based on this force-feedback, taking advantage of their intuition during docking. Sampled ligand configurations in this process are used in PRM construction.

3 Implementation and Methods

3.1 Molecular Simulation



(a) *Isosurface* **(b)** *All Atoms*

Figure 1: *(a-b) Ligand (orange) with receptor (purple). The docking site can be seen as a cavity in the receptor's isosurface.*

In our simulation the receptor and ligand are represented as rigid bodies with static atoms. This rigid body representation reduces complexity for runtime performance, but prevents Molecular Tetris from finding docked configurations that require flexibility. The Tel1p and MHC chains have 3027 atoms combined. The receptor is fixed in place while the ligand is free for the user to move. The underlying set of atoms (Figure 1b) are used for the potential energy approximation, but the molecules are only shown to the user as isosurfaces (Figure 1a). Drawing only the isosurface representation decreases the time spent drawing the scene and simplifies the problem visually for the user. The colors chosen for the molecules are arbitrary and only used to visually differentiate between them. The isosurface models were generated from PDB files using Chimera with a resolution setting of 2 for the ligand and 3 for the receptor [Pettersen et al. 2004]. The ligand atoms translate and rotate as a rigid body when the user moves the ligand. The structure of the human class I MHC molecule (receptor) bound to Tel1p (ligand) was taken from the RCSB Protein Data Bank (PDB ID 3H9S) [Borbulevych et al. 2009]. MHC is used due to its diverse binding and importance in immune system activation.

3.2 Energy Approximation Function

We calculate intermolecular potential energy U_{inter} (1) between receptor R and ligand L as the sum of all pairwise electrostatic U_{es} (2) and Lennard-Jones U_{vdw} (3) atomic interactions of receptor atoms i and ligand atoms j:

$$U_{\text{inter}}(R,L) = \sum_{i}^{R}\sum_{j}^{L} U_{\text{es}}(i,j) + U_{\text{vdw}}(i,j), \tag{1}$$

$$U_{\text{es}}(i,j) = C\frac{q_i q_j}{r_{ij}}, \tag{2}$$

$$U_{\text{vdw}}(i,j) = \sqrt{\epsilon_i \epsilon_j}\left[\left(\frac{\rho_i+\rho_j}{r_{ij}}\right)^{12} - 2\left(\frac{\rho_i+\rho_j}{r_{ij}}\right)^{6}\right]. \tag{3}$$

In the above equations, r_{ij} is interatomic distance and C the electrostatic constant. Our current implementation uses values for partial charges q_i, Lennard-Jones well depths ϵ_i, and Lennard-Jones minimal distances $\rho_i = 2^{\frac{1}{6}}\sigma_i$ from the AMBER force field [Duan et al. 2003].

The intermolecular potential energy is used directly to rank ligand-receptor configurations. Because we make the rigid body assumption for both molecules, intramolecular interactions are not calculated, therefore $E = U_{inter}$. Users are encouraged to manipulate the ligand to discover local and global potential minima.

The force approximation used for feedback is calculated from the gradient of the potential approximation. For torque, the cross product between each ligand atom's displacement vector (from the center of mass) and the force from the interaction between the ligand atom and each receptor atom can be used, similar to [Hou and Sourina 2011]. However, torque and force are not handled at the same time due to the limitations of the particular three axis device. The operator can hold one button down for translation movement and force feedback, or a different button for angular movement and torque feedback. These calculations are done using an all-atom cloud model between ligand and receptor (see Figure 1b).

3.3 Force Feedback

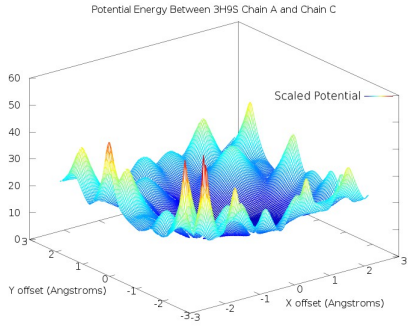


Figure 2: *Scaled potential energy approximation near the native bound configuration of human class I MHC receptor bound to ligand Tel1p. The lowest energy points are at the bottom of the frame as seen in dark blue.*

The energy potential approximation is highly sensitive to the position of the atoms due to the nature of the Lennard-Jones potential. Because of these large differences between low and high energy approximation values, a logarithmic scaling function (4) is used to bring the values into a smaller range for force-feedback (Figure 2):

$$E_s(E) = \begin{cases} \ln(E), & \text{if } E \geq 1 \\ -\ln(2-E), & \text{if } E < 1 \end{cases}, \tag{4}$$

where E is the original energy value and E_s is energy rescaled. To prevent sudden device tremors and confusion with force-feedback, the feedback vector is time-smoothed according to:

$$\vec{v_t} = S(\vec{v_i} - \vec{v_{t0}}), \tag{5}$$

$\vec{v_t}$ is the new force-feedback direction, $\vec{v_i}$ is the eventual target feedback direction, and $\vec{v_{t0}}$ is the current force-feedback direction. S is an adjustable factor that increases or decreases the speed at which $\vec{v_t}$ reaches $\vec{v_i}$. This proportional time delay applied to each frame of the program maintains the need for quick changes in feedback for extreme energy differences, as well as smooths out the peaks in the scaled energy approximation. Higher values of S "tighten" the force-feedback with $S = 1$ resulting in no time delay. A value of $S = \frac{1}{2}$ was used in the study to balance stability with responsiveness.

After scaling and proportional time delay, the resulting vector is passed to the haptic device for output. With all of these combined, a maximum threshold for force output through the haptic device is unnecessary.

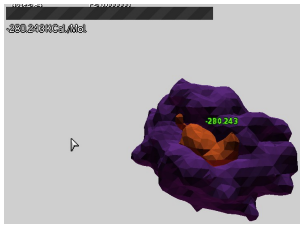
Since force-feedback devices may not be available to all users, we have also used a game controller with vibrational feedback. For the controller used, the motor is a scalar output and the force feedback was adapted by using the magnitude of force as the vibration feedback strength. This enables the user to feel the atomic forces through the strength of the vibration.

3.4 Roadmaps

Roadmaps are constructed similar to previous PRM methods [Amato and Song 2002]. However, in this paper, users generate the ligand configurations that are used for the roadmap. First, configurations are sampled from Molecular Tetris by recording user ligand configurations where each configuration is no more than 0.1Å apart from the next. Ligand configurations of an energy greater than a high potential energy threshold $E_{\max}$ are not recorded.

An edge between two ligand configurations (c_1, c_2) is weighted by a function of the difference between the maximum potential energy among interpolated ligand configurations between the start and end configuration, $c_1 = s_0, s_1, ..., s_n = c_2$, and the initial potential energy $E(c_1)$. The edge weight, $W_{i,j}$, is $\ln(\Delta E + 1)$ where the difference in energy, ΔE, is $max(E(s_0), ..., E(s_n)) - E(c_1)$. Therefore, edges of decreasing potential energy are given a weight of 0, otherwise the weight reflects an energetic traversal cost. This is needed to identify shortest paths using Dijkstra's algorithm. Edges are calculated for every pair of configurations in both directions. New roadmaps are built from existing roadmaps by appending them with new user sets using the incremental roadmap generation method [Xie et al. 2008].

3.5 Game Mechanics



(a) *Gameplay*

(b) *Leaderboards*

(c) *Force Feedback Device*

(d) *Game Controller*

Figure 3: *(a-b) Screenshots of Molecular Tetris, (c) users operating the force feedback device, and (d) game controller and laptop.*

While playing Molecular Tetris, the user will have the ability to move the ligand in space and explore the receptor surface to look for low-energy binding configurations. Example players, hardware setups, and game screens are shown in Figure 3. A scoring system is incorporated based on the lowest energy found by a user while

playing. As the player encounters configurations with lower energy than the user's current score, the score is updated. The score improvement is also reflected on the screen with an animation of the new score above the molecules (Figure 3a). Leaderboards have been constructed for players to see how their personal scores compare to scores achieved by other players (Figure 3b). Since low energy states correspond to good docking configurations, players who find the lowest energy configurations will be at the top of the leaderboard. This study presents results with an Xbox 360 controller (Figure 3d) and the Novint Falcon® (Figure 3c); two systems that utilize haptic feedback, are commonly availiable, facilitate user interactions, and have not been previously used for molecular docking.

4 Results

4.1 Performance

In order to quantify the computation time for the interactive system, we captured runtimes that reflected the potential calculation and the impact of model rendering and resolution. Our method uses two threads to improve overall performance and provide smooth force-feedback. One thread repeatedly updates potential energy, averaging about 12 calculations per second. Force-feedback and scene drawing are handled on another thread, using the most recent results of the potential energy thread to calculate force. The scene is drawn using the isosurface representation for higher performance.

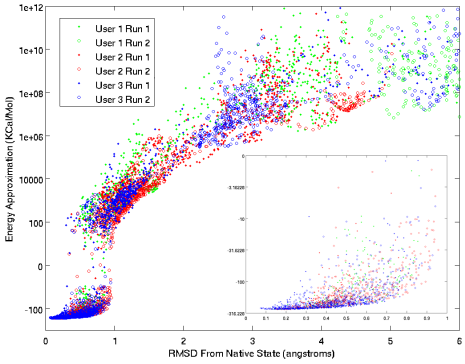
Table 1: *Runtime Performance of Main Thread With Different Chimera Resolution Settings*

Resolution	Polygons in Isosurfaces	Time per Frame (ms)
-	0 (No Drawing)	18
3	3160	21
2	7840	23
1	60184	51

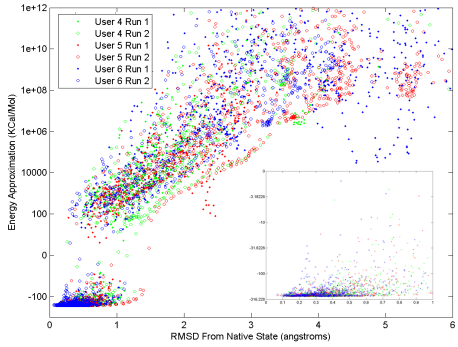
The polygon count in the isosurface vs. main thread performance can be seen in Table 1. Recall that model resolution can be adjusted without affecting the energy and force calculations. In Table 1, the first entry is the baseline performance, force feedback and program overhead, when drawing no models. We also studied the impact of model resolution. In Table 1 resolution corresponds to the isosurface model resolution setting from Chimera. Polygon triangles are more efficient for GPUs to render than realistic atomic spheres, of which Tel1P and MHC combined have 3027 atoms. Together, this multi-threaded environment produces a real-time sensation of the atomic forces modeled in the energy approximation on commodity hardware with a visual and haptic touch feedback frame rate of about 42 frames per second (about 23 milliseconds per visual and haptic touch feedback frame) using 61.4MB memory on a commodity laptop with an AMD A6-5200 APU chipset with a 4-core CPU 2GHz clock rate and Radeon HD 4800 GPU. User runs lasted between 5 to 15 minutes depending on the user speed.

4.2 Haptic-Guided User Sampled Configurations

In order to capture configurations for roadmap generation, data from six users with two runs per user were recorded. Each user run contributed 1000 configurations (Figure 4). The first three operated the force feedback device (Figure 4a) while the second three used a game controller (Figure 4b) with vibration feedback. It is interesting to notice that different strategies are implemented by users, represented by distinct colors (Figure 4). The inset shows details around negative energy configurations. Root-mean-squared deviation (RMSD) is the distance between two configurations, and we show the RMSD to the known native bound state. There are particular locations where users would focus exploration before investigating other locations, such as User 3 (blue) between 2 and 4 Å RMSD and the cluster generated by several users near the native configuration (Figure 4a). Also, the area around the native configuration had been densely explored despite the fact that users were not explicitly aware of the native configuration location.



(a) *Force Feedback Device*



(b) *Game Controller*

Figure 4: *Potential energies and RMSD for (a) force feedback device and (b) game controller. Points are configurations generated by users as they play the game. The game controller (b) shows a larger spread of data as compared to (a), which shows well-defined clusters. The insets show detail around the low energy RMSD regions, i.e., near the bound native state. The energy is shown on a logarithmic scale.*

The lowest RMSD from the known native state found amongst the configurations in each set are similar between the two devices, shown in Table 2. Low RMSD configurations, 0.075Å and 0.068Å, were found with both force-feedback device and game controller inputs, respectively. This shows that users were able to find configurations extremely close to the experimentally determined bound state with only haptic and visual guidance.

4.3 Multi-user Roadmaps

Once we obtain configurations from all user runs they can be combined to generate roadmaps, as explained in Section 3.4. The samples from Section 4.2 were used to build the roadmap. Each subsequent user extended the roadmap iteratively. Table 3 shows the number of configurations and edges created for each run.

For comparison against haptic-guided roadmaps, a roadmap was built with 6000 Gaussian distributed rigid-body ligand configura-

Table 2: *Lowest RMSD from native state found for each user run.*

Type	(User,Run)	Lowest RMSD
Force-Feedback Device	(1,1)	0.128Å
	(1,2)	4.143Å
	(2,1)	0.143Å
	(2,2)	0.126Å
	(3,1)	0.079Å
	(3,2)	0.075Å
Game Controller	(4,1)	0.100Å
	(4,2)	0.106Å
	(5,1)	0.088Å
	(5,2)	0.077Å
	(6,1)	0.157Å
	(6,2)	0.068Å

Table 3: *Roadmaps from haptic-guided configurations and Gaussian distributed configurations.*

Data type	Cumulative sets (User,Run)	Configuration Count	Edge Count
Force-Feedback Device	(1,1)	1000	12516
	(1,1),(2,1)	2000	25246
	(1,1),(2,1),(1,2)	3000	37424
	(1,1),(2,1),(1,2),(3,1)	4000	50240
	(1,1),(2,1),(1,2),(3,1),(2,2)	5000	63126
	(1,1),(2,1),(1,2),(3,1),(2,2),(3,2)	6000	75750
Game Controller	(4,1)	1000	1998
	(4,1),(5,1)	2000	15926
	(4,1),(5,1),(4,2)	3000	29498
	(4,1),(5,1),(4,2),(6,1)	4000	44086
	(4,1),(5,1),(4,2),(6,1),(5,2)	5000	59892
	(4,1),(5,1),(4,2),(6,1),(5,2),(6,2)	6000	75326
Gaussian	-	6000	88758

tions centered where the ligand is in a docked configuration, with 5Å in translational and 5° in rotational standard deviations. In total 88,758 weighted edges were created.

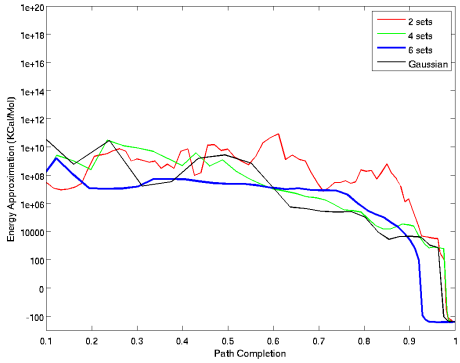
All roadmap configurations were connected by nearest neighbors ($K = 10$) using a Euclidean distance metric. Then a K-pair = 10 component connection method was used. Roadmaps of user sampled configurations are created by incremental construction [Xie et al. 2008]. Building roadmaps incrementally is online and requires less computation, enabling the roadmap constructor to receive new user sets during runtime. After the first roadmap of 1000 samples is built, we create successive roadmaps by importing another set of user configurations and connecting.

Finally, queries were performed from a start ligand configuration about 5.02Å RMSD distance from the known native configuration of 3H9S using Dijkstra's algorithm. As user sets were combined into larger roadmaps, the resulting query path became smoother with less peaks or energy barriers to overcome as seen in Figure 5. The query path results from force feedback device data in Figure 5a are similar to the results from the game controller data in Figure 5b. The path resulting from 6 user sets, shown in blue, also has the least potential energy as it approaches the native configuration.

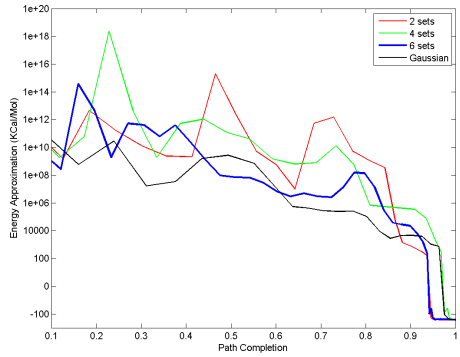
The query from the Gaussian roadmap, displayed in black, contains less pronounced energy peaks. However, recall that the Gaussian ligand configuration samples were generated with a mean centered around the known native configuration while the users did not know the precise native configuration and had only the force feedback to guide them. However, the Gaussian sampler could not be applied to new ligand receptor pairs of unknown native configuration.

5 Conclusion and Future Work

To our knowledge, this is the first work that investigates molecular docking by combining haptics and the potential to crowdsource



(a) *Force Feedback Device*



(b) *Game Controller*

Figure 5: *(a-b) Potential energy from the query to the docked configuration for constructed roadmaps. Path completion is the normalized amount of total RMSD traveled in query path. The energy is shown on a logarithmic scale.*

path planning. Based on our preliminary user test, we found that the force-feedback device more strongly restricted the space a user explored, but that both types of devices allowed users to identify low potential energy configurations and smooth trajectories. This result allows us to expand user-assisted molecular docking from specialty force-feedback devices to commodity hardware devices. Hardware devices such as the game controller allow us to expand our experimental results to a larger, crowdsourced user base.

Roadmaps built from a small number of haptic-guided user trajectories produced paths of low, smooth potential energy to the native configuration where an automated sampling method required full knowledge of the native configuration. The iterative nature of roadmap construction can be integrated easily with streams of new user configuration sets and resulting roadmap quality can be monitored over time. We are exploring kinematic linkage extensions that will allow us to represent molecular flexibility. This would enable the exploration of larger configuration spaces, possibly leading to improved path quality and lower energy configurations.

Acknowledgements

This work is supported in part by the National Institutes of Health (NIH) Grant P20GM110907 supporting the Center for Evolutionary and Theoretical Immunology.

References

ADAMSON, T., BAXTER, J., MANAVI, K., JACOBSON, B., AND TAPIA, L. 2014. Crowdsourced molecular docking using path-planning and haptic devices. In *Workshop on Robotics Methods for Structural and Dynamic Modeling of Molecular Systems, RSS 2014*.

AL-BLUWI, I., SIMÉON, T., AND CORTÉS, J. 2012. Motion planning algorithms for molecular simulations: A survey. *Computer Science Review 6*, 4, 125–143.

AMATO, N. M., AND SONG, G. 2002. Using motion planning to study protein folding pathways. *JCB 9*, 2, 149–168. Special issue of Int. Conf. Comput. Molecular Biology (RECOMB) 2001.

BAYAZIT, O. B., SONG, G., AND AMATO, N. M. 2001. Ligand binding with OBPRM and user input. In *IEEE Int. Conf. on Rob. and Auto.*, 954–959.

BEBERG, A., ENSIGN, D., JAYACHANDRAN, G., KHALIQ, S., AND PANDE, V. 2009. Folding@home: Lessons from eight years of volunteer distributed computing. In *Parallel Distributed Processing, 2009. IEEE International Symposium on*, 1–8.

BIVALL, P., AINSWORTH, S., AND TIBELL, L. A. 2011. Do haptic representations help complex molecular learning? *Wiley Periodicals, Inc. Sci. Ed. 95* (March), 700–719.

BOLOPION, A., CAGNEAU, B., REDON, S., AND RÉGNIER, S. 2010. Comparing position and force control for interactive molecular simulators with haptic feedback. *Journal of Molecular Graphics and Modelling 29*, 2, 280–289.

BORBULEVYCH, O., PIEPENBRINK, K., GLOOR, B., SCOTT, D., SOMMESE, R., COLE, D., SEWELL, A., AND BAKER, B. 2009. T cell receptor cross-reactivity directed by antigen-dependant tuning of peptide-MHC molecular flexibility. *Immunity 31*, 885–896.

CHEN, B. Y., AND HONIG, B. 2010. VASP: A volumetric analysis of surface properties yields insights into protein-ligand binding specificity. *PLoS Comput Biol 6*, 8 (8), e1000881.

CLAUBEN, H., BUNING, C., RAREY, M., AND LENGAUER, T. 2001. FlexE: Efficient molecular docking considering protein structure variations. *Journal of Molecular Biology 308*, 2, 377–395.

CORTÉS, J., SIMÉON, T., RUIZ DE ANGULO, V., GUIEYSSE, D., REMAUD-SIMÉON, M., AND TRAN, V. 2005. A path planning approach for computing large-amplitude motions of flexible molecules. *Bioinformatics (Oxford, England) 21 Suppl 1*, i116–i125.

DAUNAY, B., MICAELLI, A., AND REGNIER, S. 2007. Energy-field reconstruction for haptic-based molecular docking using energy minimization processes. *Int. Conf. on Intelligent Robots and Systems IEEE RSJ*, 2704–2709.

DUAN, Y., WU, C., CHOWDHURY, S., LEE, M. C., XIONG, G., ZHANG, W., YANG, R., CIELPLAK, P., LUO, R., LEE, T., CALDWELL, J., WANG, J., AND KOLLMAN, P. 2003. A point-charge force field for molecular mechanics simulations of proteins based on condensed-phase quantum mechanical calculations. *Wiley Periodicals, Inc. J. Comput. Chem. 24* (July), 1999–2013.

HE, X., AND CHEN, Y. 2009. Haptic-aided robot path planning based on virtual tele-operation. *Robotics and Computer-Integrated Manufacturing 25*, 4–5, 792–803.

HOU, X., AND SOURINA, O. 2011. Six degree-of-freedom haptic rendering for biomolecular docking. In *Transactions on Computational Science XII*, vol. 6670. 98–117.

HOU, X., SOURINA, O., AND KLIMENKO, S. 2014. Visual haptic-based collaborative molecular docking. In *The 15th International Conference on Biomedical Engineering*, vol. 43. 360–363.

HUANG, S.-Y., AND ZOU, X. 2010. Advances and challenges in protein-ligand docking. *International Journal of Molecular Sciences 11*, 8, 3016–3034.

JONES, G., WILLETT, P., GLEN, R. C., LEACH, A. R., AND TAYLOR, R. 1997. Development and validation of a genetic algorithm for flexible docking. *Journal of Molecular Biology 267*, 3, 727–748.

KHATIB, F., COOPER, S., TYKA, M. D., XU, K., MAKEDON, I., POPOVIC, Z., AND BAKER, D. 2011. Algorithm discovery by protein folding game players. *Proceedings of the National Academy of Sciences*.

KUNTZ, I. D., BLANEY, J. M., OATLEY, S. J., LANGRIDGE, R., AND FERRIN, T. E. 1982. A geometric approach to macromolecule-ligand interactions. *Journal of Molecular Biology 161*, 2, 269–288.

LAWITZKY, M., MEDINA, J., LEE, D., AND HIRCHE, S. 2012. Feedback motion planning and learning from demonstration in physical robotic assistance: differences and synergies. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, 3646–3652.

MEILER, J., AND BAKER, D. 2006. Rosettaligand: Protein-small molecule docking with full side-chain flexibility. *Proteins: Structure, Function, and Bioinformatics 65*, 3, 538–548.

MORRIS, G. M., HUEY, R., LINDSTROM, W., SANNER, M. F., BELEW, R. K., GOODSELL, D. S., AND OLSON, A. J. 2009. AutoDock4 and AutoDockTools4: Automated docking with selective receptor flexibility. *Journal of Computational Chemistry 30*, 16, 2785–2791.

MOUSTAKAS, D. T., LANG, P. T., PEGG, S., PETTERSEN, E., KUNTZ, I. D., BROOIJMANS, N., AND RIZZO, R. C. 2006. Development and validation of a modular, extensible docking program: DOCK 5. *Journal of Computer-Aided Molecular Design 20*, 10–11, 601–619.

PETTERSEN, E., GODDARD, T., HUANG, C., COUCH, G., GREENBLATT, D., MENG, E., AND FERRIN, T. 2004. UCSF Chimera–a visualization system for exploratory research and analysis. *J. Comput. Chem. 25* (October), 1605–1612.

RAREY, M., KRAMER, B., LENGAUER, T., AND KLEBE, G. 1996. A fast flexible docking method using an incremental construction algorithm. *Journal of Molecular Biology 261*, 3, 470–489.

STOCKS, M. B., HAYWARD, S., AND LAYCOCK, S. D. 2009. Interacting with the biomolecular solvent accessible surface via a haptic feedback device. *BMC Structural Biology 9* (October).

VÁZQUEZ, C., ROSELL, J., CHIRINOS, L., AND DOMÍNGUEZ, O. 2010. Haptic primitives guidance based on the Kautham path planner. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, 4686–4691.

XIE, D., MORALES, M., PEARCE, R., THOMAS, S., LIEN, J.-M., AND AMATO, N. M. 2008. Incremental map generation (IMG). In *Algorithmic Foundation of Robotics VII*. Springer, Berlin/Heidelberg, 53–68.