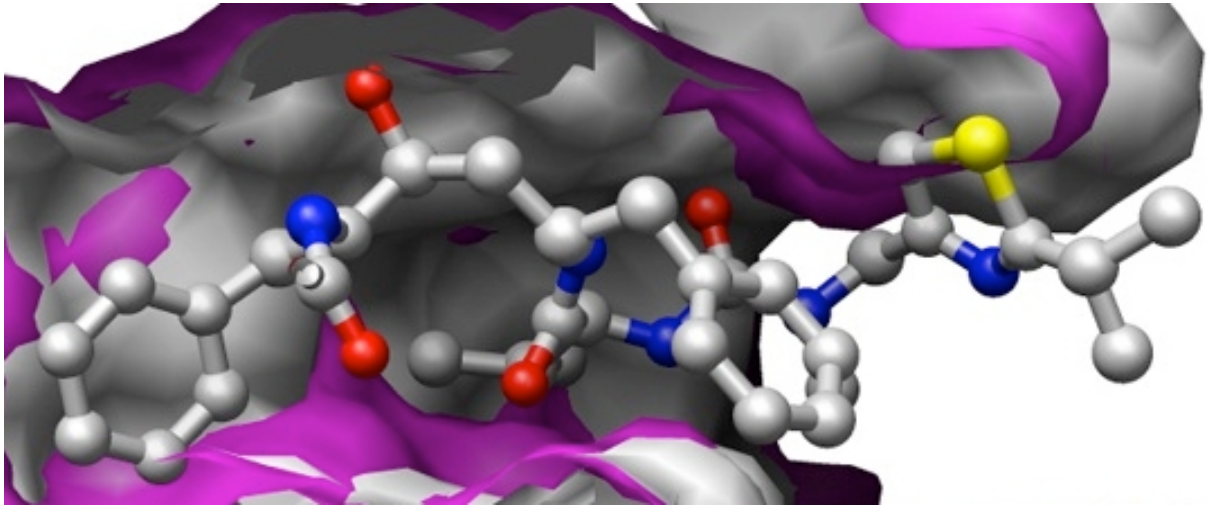


DAPLDS: A Dynamically Adaptive Protein-Ligand Docking System Based on Multi-Scale Modeling

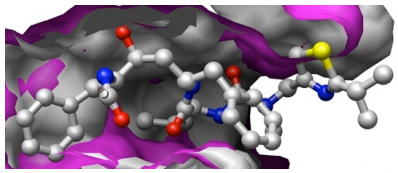
PI: Michela Taufer

Co-PIs:

Martine Ceberio and Patricia J. Teller (UTEP)

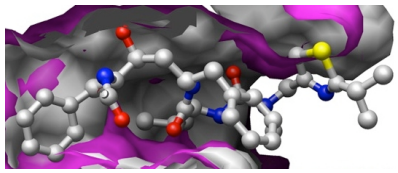
David Anderson (UC Berkeley)

Charles L. Brooks III (TSRI)



DAPLDS Project

- DAPLDS, Dynamically Adaptive Protein-Ligand Docking System, supports scientists in understanding the atomic details of specific protein-ligand interactions
- DAPLDS focuses on high-throughput docking by adapting the docking model
  - Multi-scale modeling based on computational scales
  - Molecular Dynamics based docking models
- Exploring large multi-scale spaces is resource demanding
  - Harness immense computing power of volunteers' computers
- DAPLDS deploys multi-scale computational modeling to balance:
  - Resource demand that guarantees a certain amount of docking accuracy (DA)
  - Resource availability that guarantees a short time to solution



DAPLDS Overview

Using Volunteer Computing resources, sample the protein-ligand space simulating the docking process:
attempt to dock a transformation of ligand, orienting it differently on each attempt, at potential docking sites of protein.

Potential energy function and solvent treatment

Model i+1

Model i

Initial model

Sampling strategy

Protein ligand P-L flexibility

Docked structures for P-L complex$_h$

Rank structures based on energy selecting those with lowest energy

Energy, docked structure, and execution time

Model $_{i+1}$

P-L complex$_h$ from class$_I$

LPDB ligands

Ligands

PDB proteins

LPDB proteins

X-ray crystal structure

Select docking parameters for model i+1 trading off accuracy and execution time

RMSD

For P-L complex$_h$

Time

Model

Identify P-L classes well-docked by model and those missdocked

Good model for P-L class

Bad results, then refine the model

"New" well-docked complexes drug leads

Experimental Laboratory

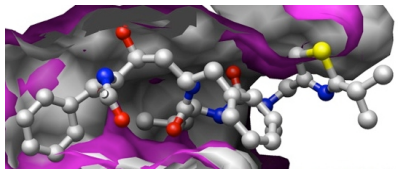
Volunteer Computing

Validate

Compute RMSD and docking accuracy of docked structure – compare to RMSD of X-ray crystal structure

Facilitate Drug Discovery

Use model i, which effectively well-docks P-L class *I*, to identify "new" well-docked complexes from class *i*



Multi-Scale Modeling

- Implement multi-scale docking models with different computational complexity and accuracy levels:

  $model_i = f(protein\text{-}ligand\ representation,\ potential\ energy\ function\ and\ solvent\ treatment,\ sampling\ strategy)$

- Cluster protein-ligand complexes in classes based on characteristics:

  $class_l = \{complex_h\}\ with\ h = 1, \ldots, N\ and\ N>>1$

- Define adaptive techniques based on simple heuristics and machine learning techniques to match models to classes dynamically:

  $model_{0\ /\ DA\ >\ p} \rightarrow \{class_a, \ldots\}$

  …

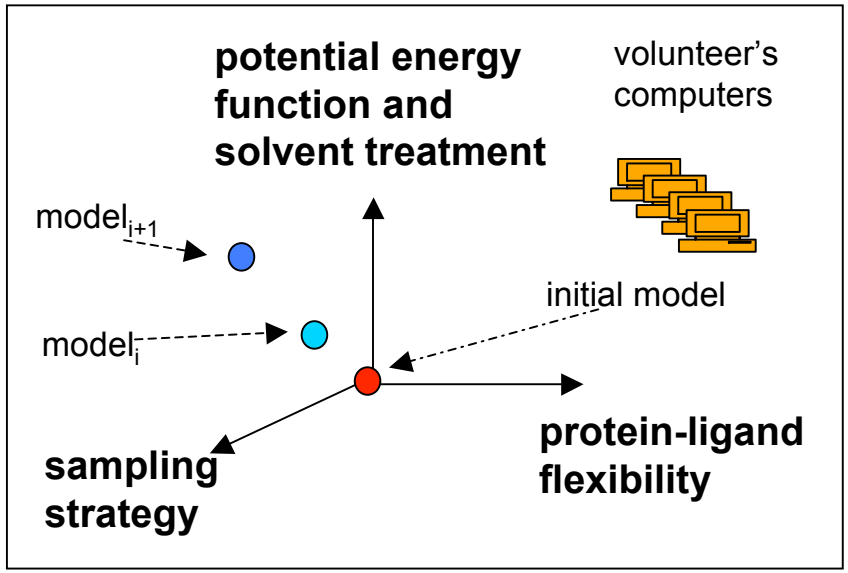
  $model_{i-1\ /\ DA\ >\ p} \rightarrow \{class_a, class_b, \ldots\}$

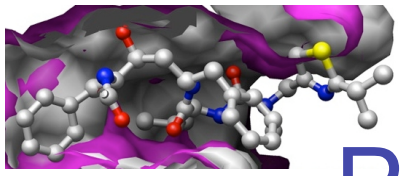
  $model_{i\ /\ DA\ >\ p} \rightarrow \{class_b, class_d, \ldots\}$

- Matching based on

  quantitative values, e.g.,

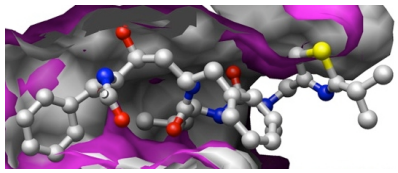
  free energy of binding and RMSDs





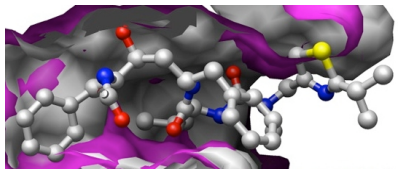
Protein-Ligand Representation

- Spanning scale from rigid to flexible representation of protein-ligand interactions
  - Coarse grid (spaced 1Å) with standard or soft Lennard-Jones potential
  - Finer grid (spaced 0.25Å) with standard or soft Lennard-Jones potential
  - All-atom representation of the protein-ligand interaction
  - Multiple protein structures of the same receptor considering *small side-chain movements*
  - Multiple protein structures of the same receptor considering *large protein movements*



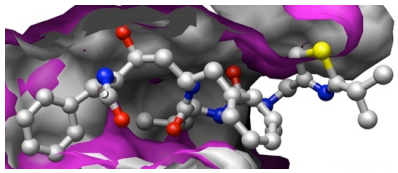
Energy and Solvent

- Spanning scale from less accurate to more accurate modeling of solvent treatment
  - Constant dielectric coefficient
  - Distance-dependent dielectric coefficient
  - Implicit representation of solvent using a Generalized Born model
  - Representation of the solvent via the Poisson-Boltzmann equation



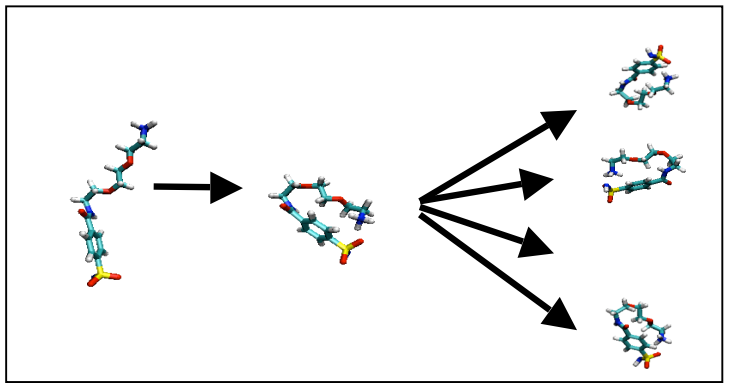
Sampling

- Spanning scale from fixed to adaptive sampling of the protein-ligand docking space
  - Fixed number of trials per attempt (initial random conformations) and for each trial a fixed number of orientations per conformation
  - Change the number of trials per attempt as well as the number of orientations per trial
- Different lengths for the heating and cooling phases as well as minimization in MD simulation

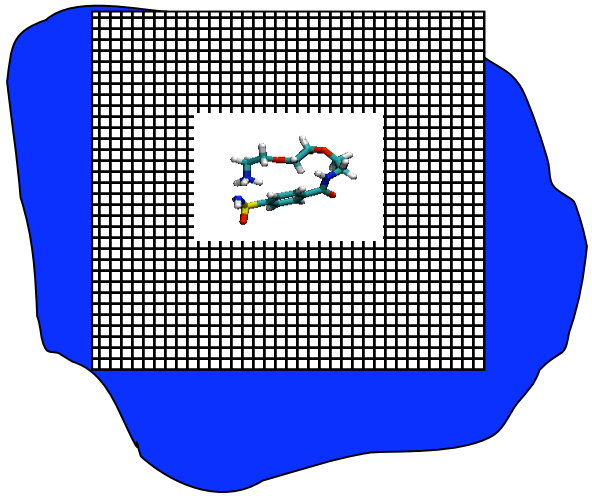


Docking Algorithms

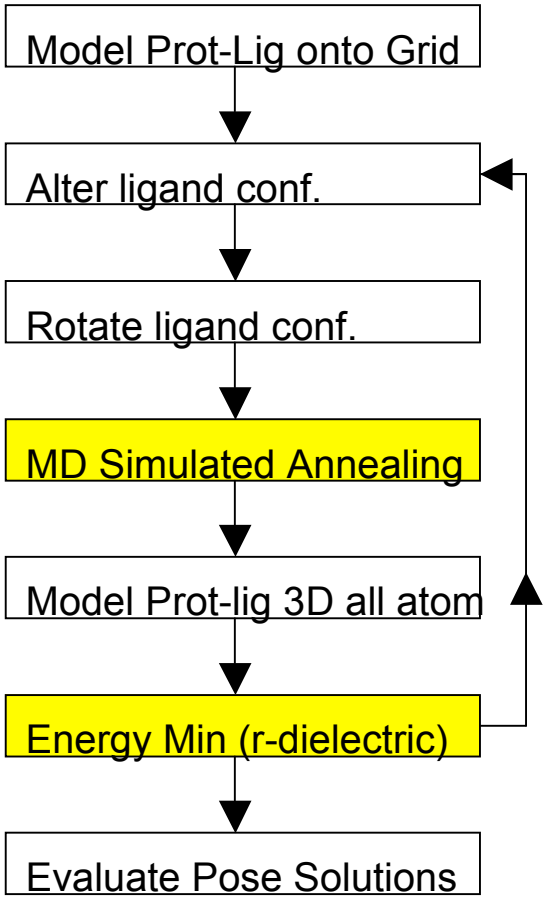
Flexible Ligand



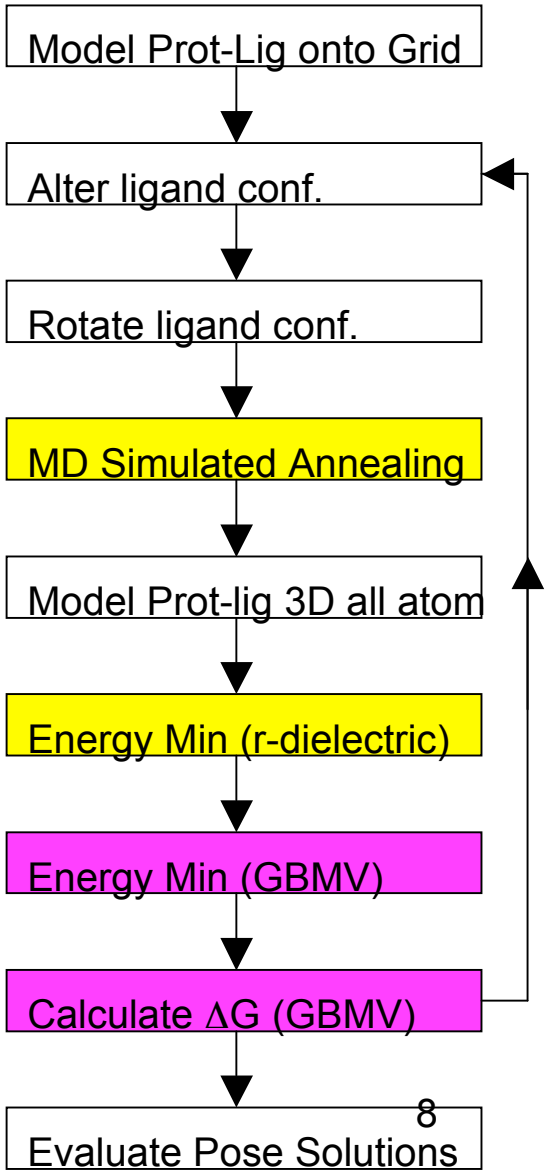
Rigid Protein

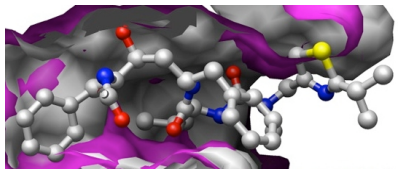


Score with Grid Energy



Score with ΔG (GBMV)





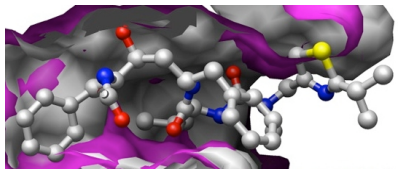
Challenges in Docking

1. Create reliable potential functions for new ligands (***MATCH***)
   (a) Read new ligand geometries
   (b) Match known connections between atoms to atom types
   (c) Build potential function from bond increment rules.
   (d) Charges, VDW parameters, torsions, angles

2. Validate these protein-ligand potential functions for docking
   (a) Docking test sets: accuracy and binding free energy
   (b) Small virtual screens: binding free energy

3. Incorporate protein flexibility into the docking method
   (a) Cross-docking: experimentally determined structures
   (b) Develop models for protein flexibility
   (c) Compare performance of models to cross-docking

- Harnessing computing power by using volunteer computing resources
  - Computers connected to the Internet and owned by the public
- Docking@Home has been in alpha test since September 7, 2006
  - http://docking.utep.edu
- Volunteer distributed computing for high-throughput protein-ligand docking simulations:
  - Built around BOINC (Berkeley Open Infrastructure for Network Computing)
  - CHARMM-based molecular docking
- Initial scientific goals aimed at validating existing docking methods and developing and validating new methods
  - Run-time selection of docking models and computing resources

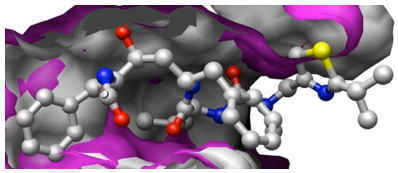


Challenges in Computation

1. Implement robust docking simulations
   (a) Across heterogeneous machines: homogenous redundancy
   (b) Across volatile machines: checkpointing
   (c) Across error-prone machines: work-unit replication

2. Explore adaptive scheduling policies
   (a) Need for reliable simulation environments for testing
   (b) Deal with different levels of resource availability and reliability
   (c) Prevent starving machines and reduce redundant computation

3. Implement multi-scale algorithmic adaptations
   (a) Accommodate adaptations in cyber-infrastructures
   (b) Characterize resources, p-l complexes, and docking models
   (c) Design techniques for selection of models and resources at run-time



Acknowledgments

- Trilce Estrada, Richard Zamudio, David Flores, Karina Escapita, Guillermo Lopez, Andre Kerstens, Martine Ceberio, and Pat Teller (UTEP)
- Charles L. Brooks III and Roger Armen (TSRI)
- David Anderson (UC Berkeley)
- Kevin Reed (World Community Grid, IBM)
- Mark Somers (University of Leiden, NL)
- The BOINC community and the Docking@Home volunteers
- OCI - NSF