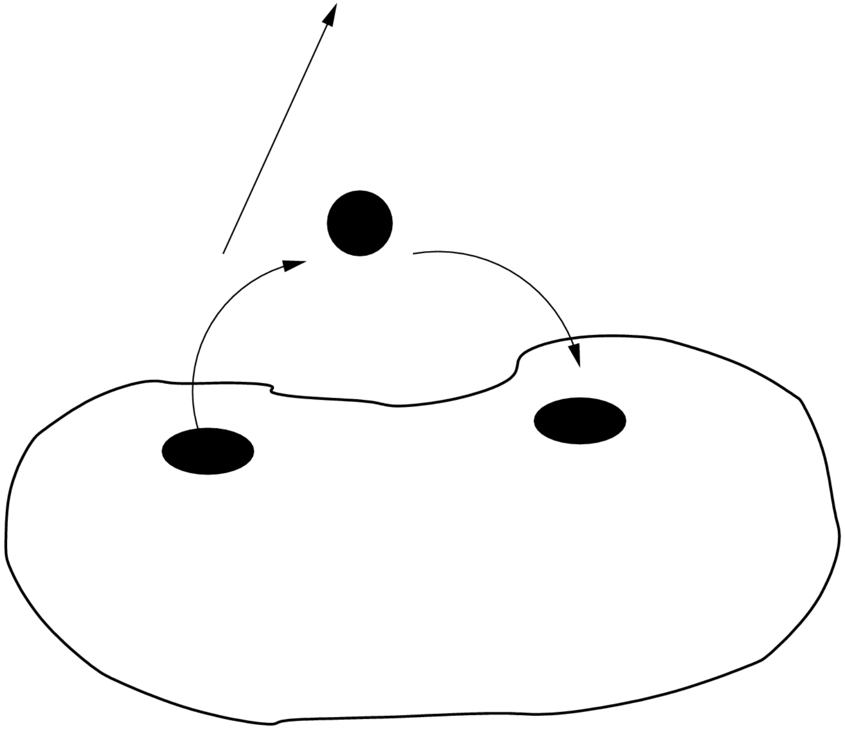
Browndye: Software for Brownian Dynamics

Second-order rate association rate constants

Site-to-site transfer probabilities



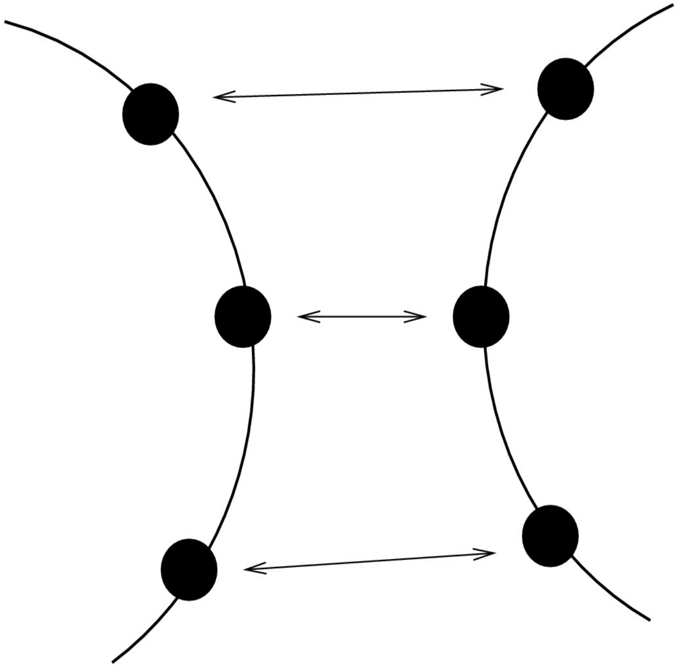
Trajectories for visualization

Inputs

- Atom positions and charges, both molecules
  Crystal structure
  PDB2PQR

- Electric Field
  APBS
  Can have nested grids

- Reaction Criteria
  Atom-atom distances



Software Structure

About 35 preprocessing programs

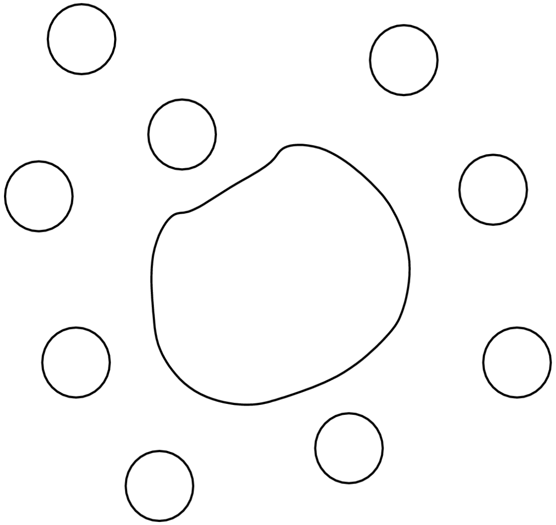
   Written mostly in Ocaml

Two simulation programs

   Written in C++11

   Multithreaded

      Run many simulations at once while sharing the same data

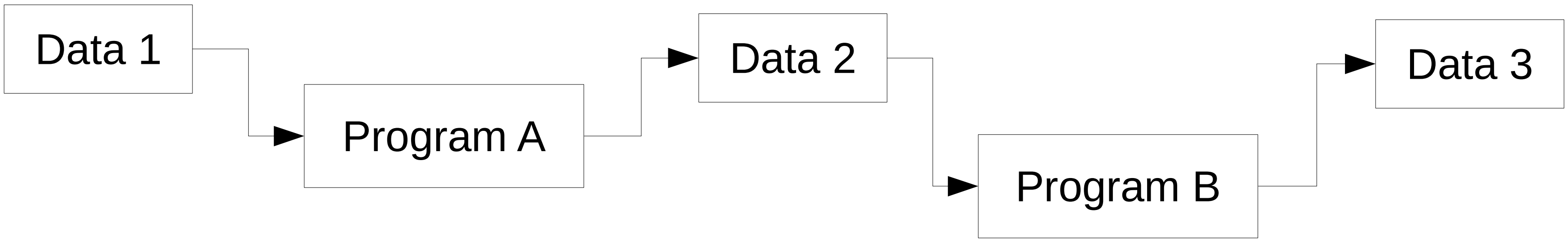


Software Structure (cont.)

- Orchestrator Program (bd_top)

  Makes sure that preprocessing programs are called in proper order on input and intermediate files

  If a file changes, reruns programs that depend on it



If a change is made to Data 2, only Program B is rerun

XML Files

```
<roottag>
  <residue>
    <atom>
      <x>1.0</x>
      <y>2.0</y>
      <z>3.0</z>
      <radius>4.0</radius>
      <charge>1.0</charge>
    </atom>
    ….
  <residue>
  ….
</roottag>
```

- Easy to read and edit

- Easy to generate

- Easy to parse, lots of free software for doing so

- Can add additional stuff without breaking software

Software Structure (cont)

- Runs on Linux and Mac
  - can use Cygwin on Windows
- Key algorithms are coded in a generic manner
  - C++ templates
  - User can use own data structures
  - User just has to write a little glue code in order to use in own software package

Brownian Dynamics

- Assume rigid bodies (working on flexible parts, though)
- Can be derived from Newton's Laws of Motion
- Assumption: time scale of solute motions >> time scale of solvent molecule motions

$$d\,\vec{x} = \frac{1}{k_B T}\boldsymbol{D}\cdot\vec{F}\,dt + \sqrt{2\,dt}\,\boldsymbol{S}\cdot\vec{w}$$

$\vec{w}$ - Vector of random, Gaussian numbers (mean 0, sdev 1)

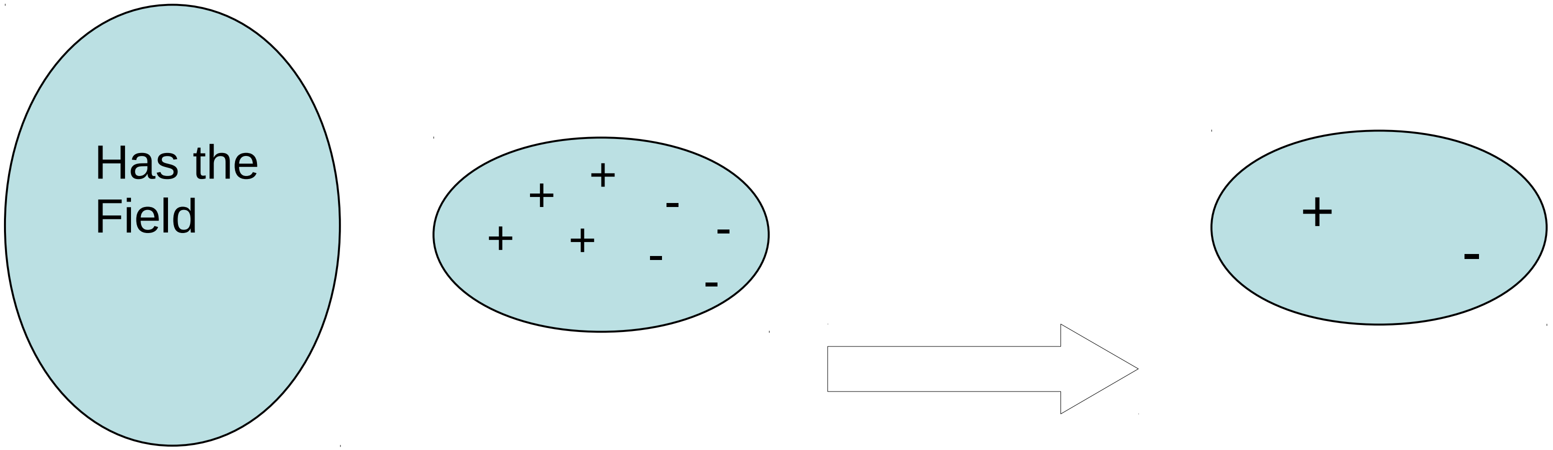
Valid only if F does not vary much during a time step

Force Calculations

- Electrostatic (Coulomb's Law)
- Short Ranged
- Desolvation
- Hydrodynamic

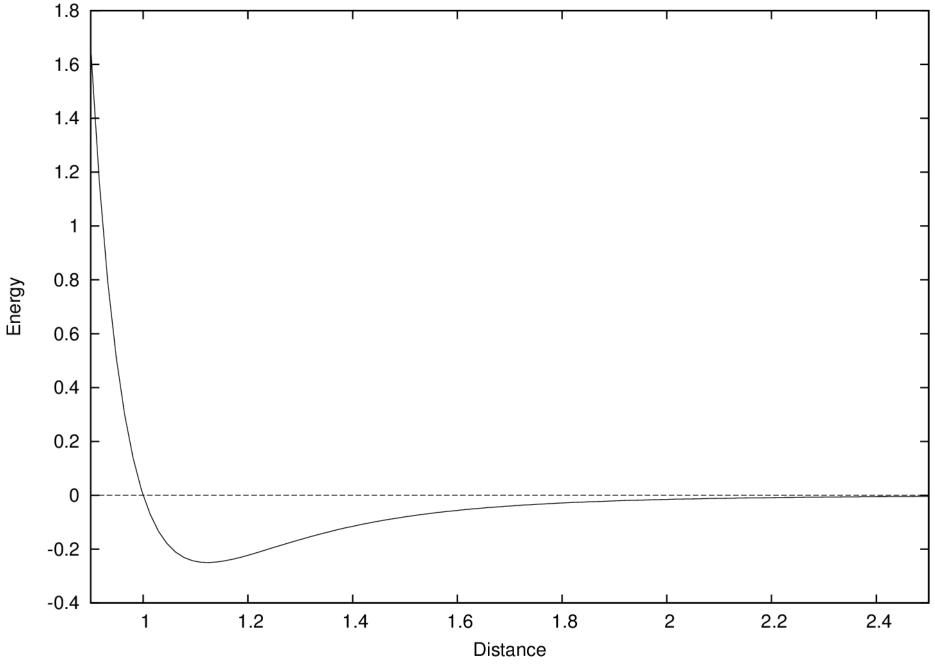
Electrostatic Forces

- Charges of one molecule feel electric field of other molecule
- Lump charges together to save time

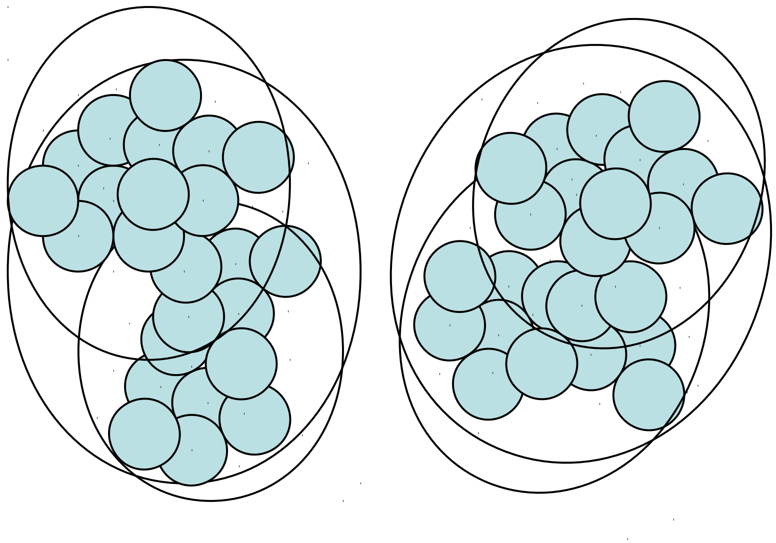


Short-Ranged Forces

- Steric (simply forbid collisions)
- Dispersion (Lennard-Jones)
  - Attract when farther out
  - Strongly repel when too close



Collision Detection and Short Ranged Forces



Done ahead of time:

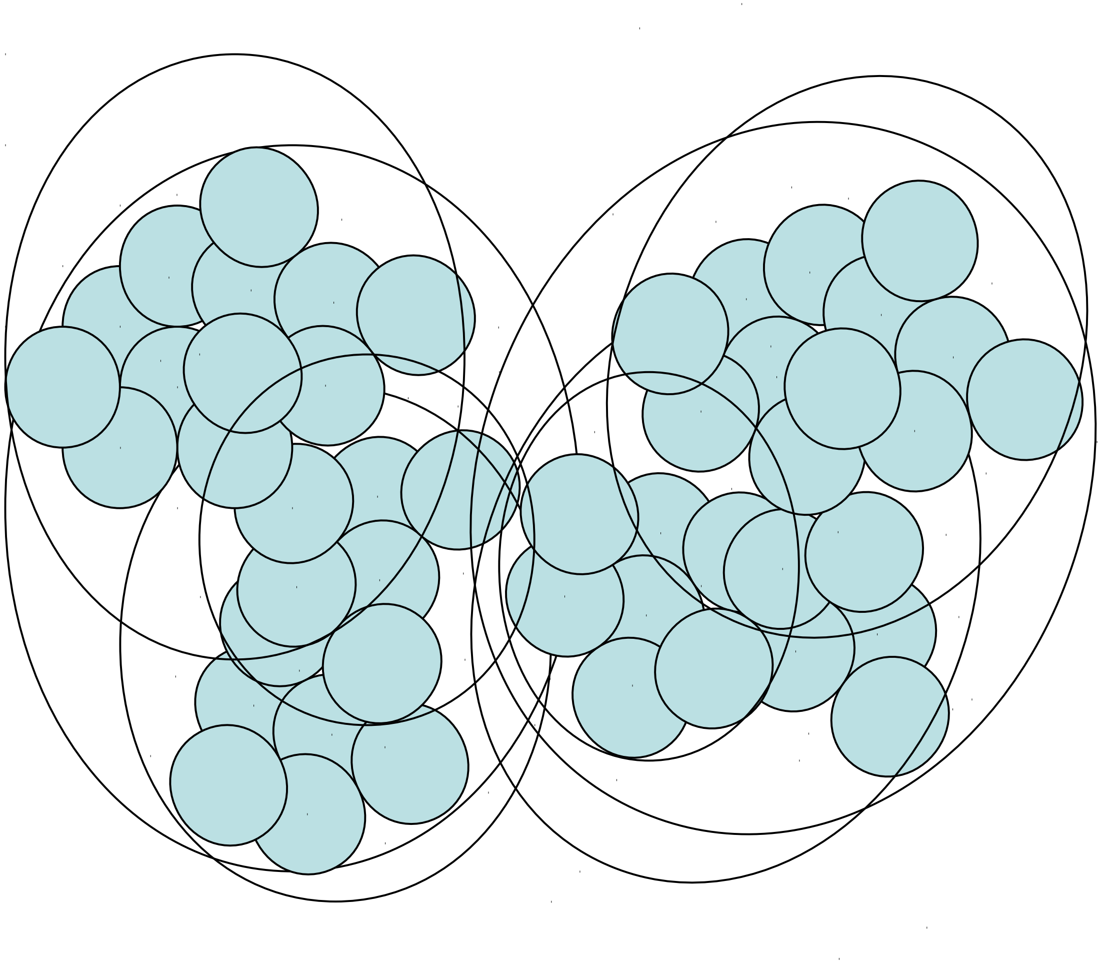
- Find smallest enclosing sphere for molecule
- Divide molecule in half, and find enclosing spheres for each half
- Repeat previous steps for halves and their children

-Stop when just a few atoms in each sphere

Collision test:

- Test distance between two outer spheres
- If not touch, then no collision
- If do touch, then recursively test distances between children spheres
- Test individual atoms at lowest level

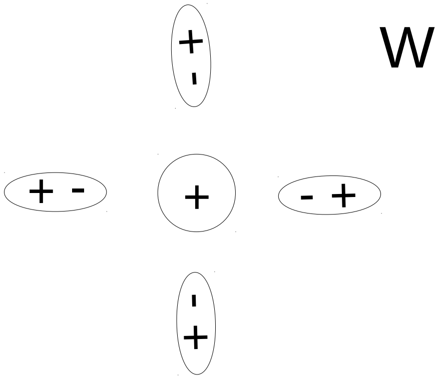
Additional trick: remove interior atoms beforehand



Also used for short-ranged interactions

Desolvation Forces

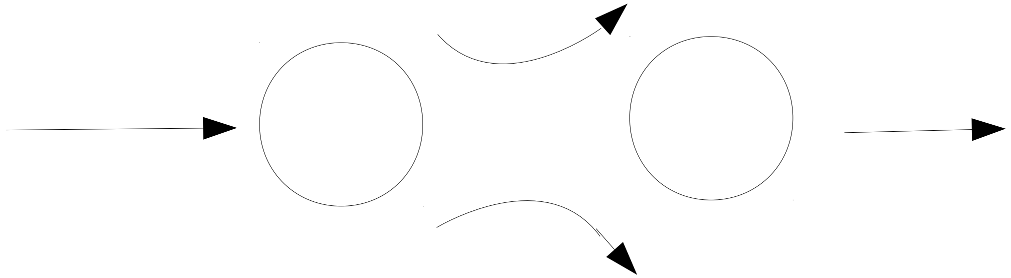
- Charges like to stay in solution
- Repulsive force between charge and surface



Water molecules are dipolar

Hydrodynamic Forces

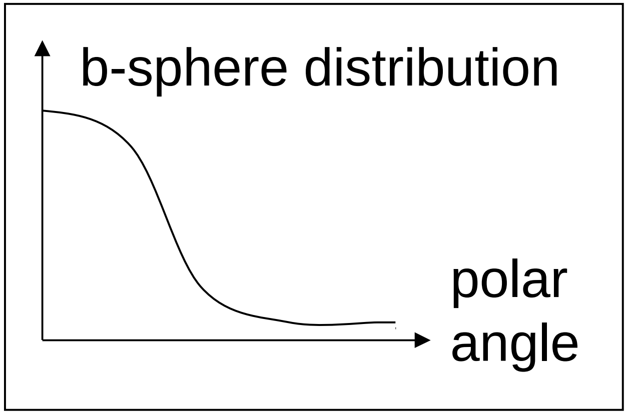
- Movement of molecule causes motion of surrounding fluid
- Fluid pushes on neighboring molecule
- Not in F term in BD equation; rather, in D

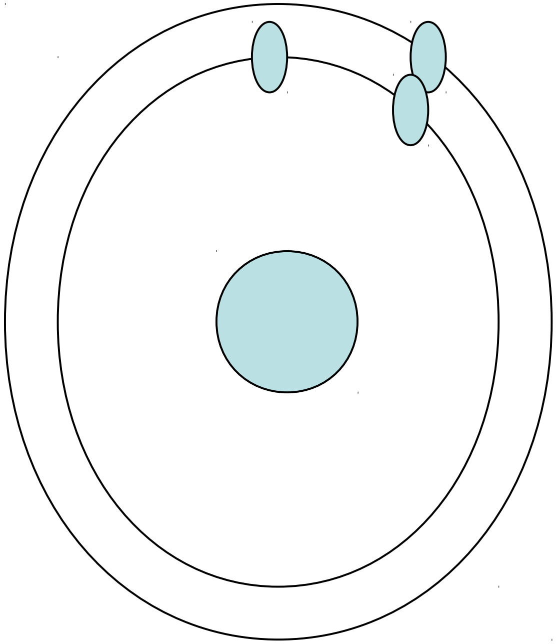


Rate Constant Calculation

- Surround larger molecule with two spheres
- Must be large enough so forces are spherically symmetric
- Builds on work by Luty, McCammon, and Zhou

Rate Constant Calculation (cont.)





- Start ligand on inner b-sphere

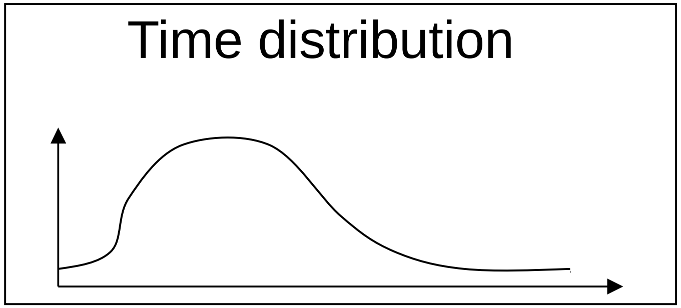
- If it reaches outer sphere before reacting, then:

  - Compute probability of ever returning to b-sphere

  - Roll the dice: if it returns, place back onto b-sphere according to precomputed probability distribution

  - Figure out how much time was spent in transit by drawing a time from a time distribution

  - Update the rotational states using transit time



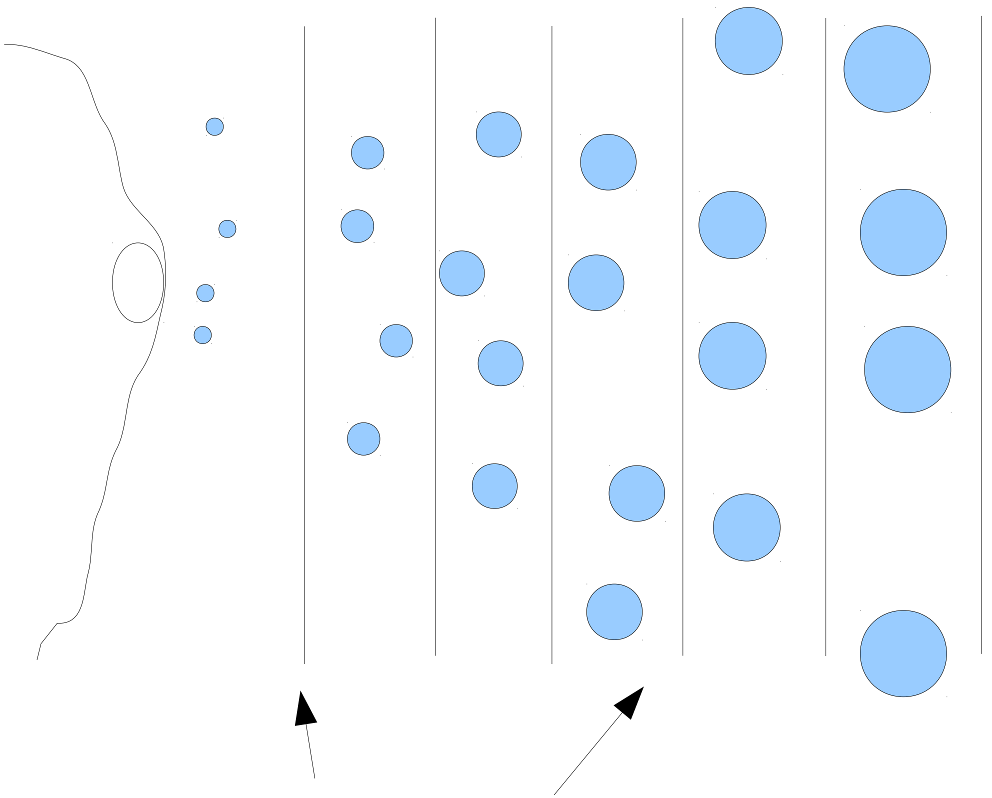
Rate Constant Calculation (cont.)

- One trajectory
  - Start on b-sphere
  - Keep going until reaction or escape
- Run many trajectories
  - Probability of reaction vs. escape is used to compute rate constant
  - Need to run enough trajectories to get good statistics and good estimate on rate constant
- Site-site probability – start on first site, run until escape or reaching of second site

Infrequent Reactions

- Must run very large number of trajectories
- Instead, run many copies at once and use statistical weighting to get better sampling in less time
- Weighted Ensemble Method

Weighted Ensemble Method



Set up bins

Run many copies

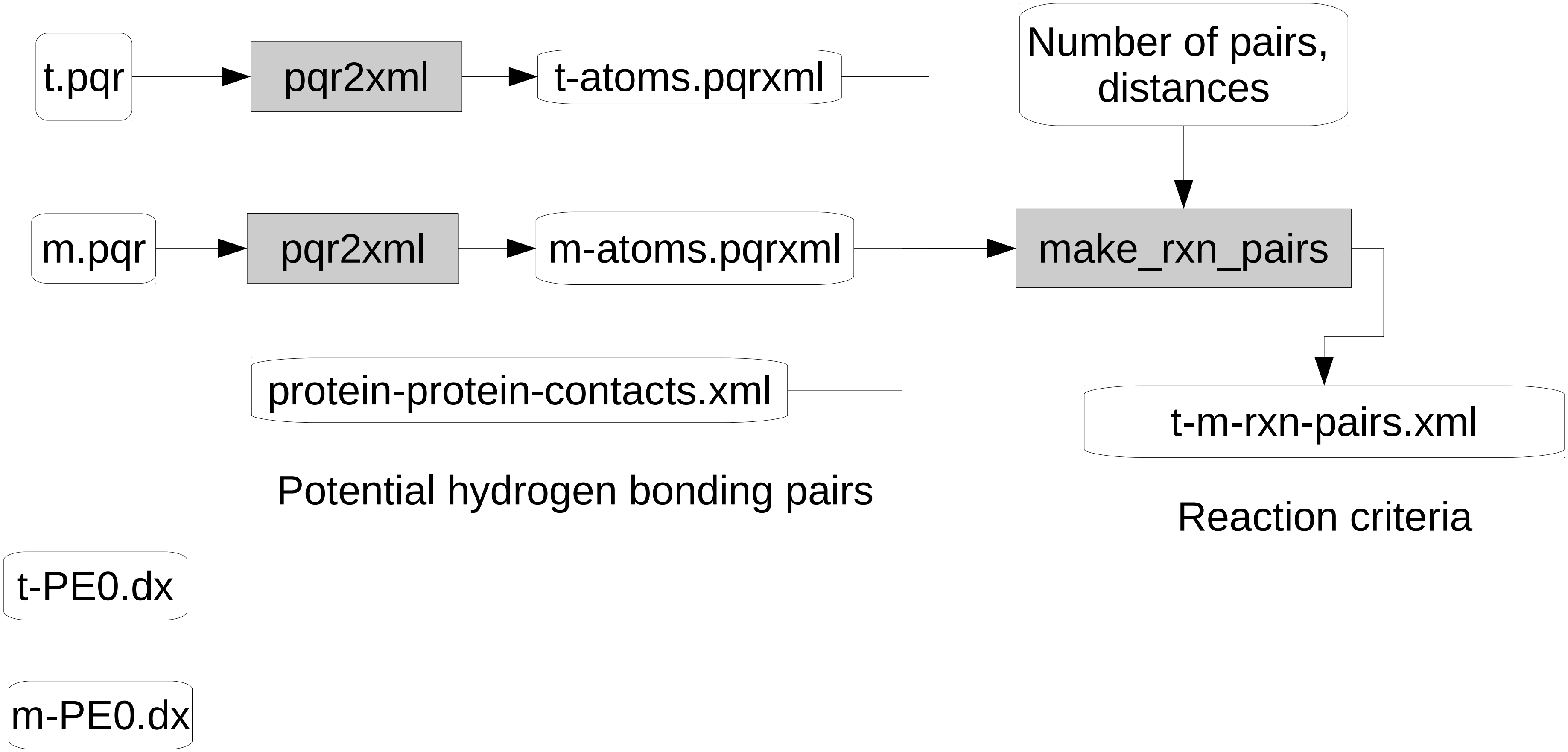
Split and combine copies to keep equal numbers in each bin

Size shown is statistical weight; does not affect physics

Can always measure something even for very rare events

Program Flow

t – thrombin; m - thrombomodulin



t.pqr

pqr2xml

t-atoms.pqrxml

Number of pairs, distances

m.pqr

pqr2xml

m-atoms.pqrxml

make_rxn_pairs

protein-protein-contacts.xml

t-m-rxn-pairs.xml

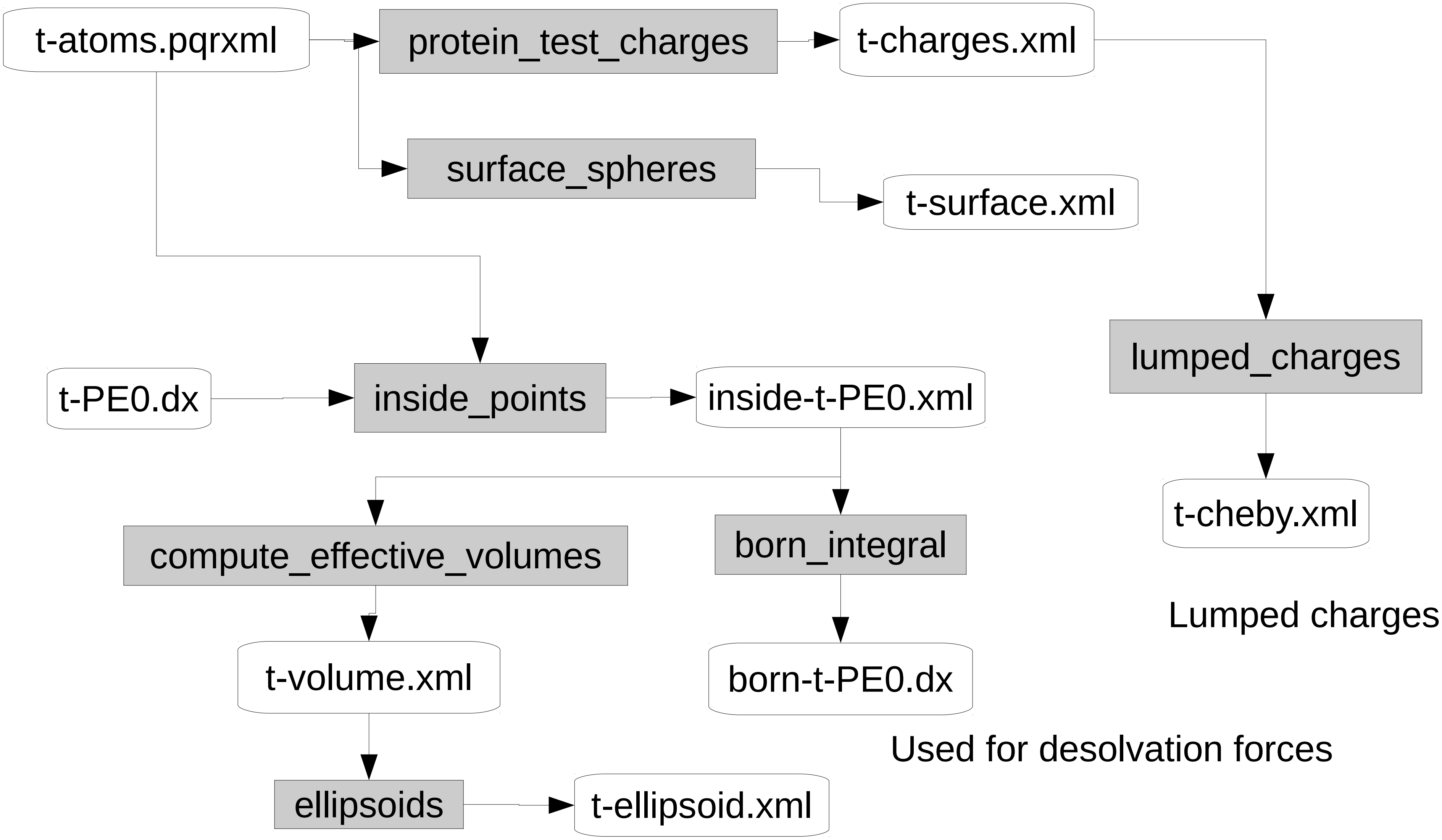
Potential hydrogen bonding pairs

Reaction criteria

t-PE0.dx

m-PE0.dx

Program Flow (cont.)



Lumped charges

Used for desolvation forces

Used for hydrodynamic interactions

Simulation Programs

- nam_simulation (Northrup-Allison-McCammon)
  - Computes one trajectory at a time
- we_simulation
  - Uses weighted ensemble method
- Overall scheme
  - bd_top runs the preprocessing programs
  - In addition to the various files, an overall simulation file is generated (t-m-simulation.xml in example)
  - User then runs simulation programs

Limitations

- Rigid body model
- Need to know structure of complex
- Rates are sensitive to reaction criteria distances
    - Big fat adjustable parameter!
    - Okay to tune distances to match rates at one set of conditions, and use for different set (ionic strength, pH, mutations)

Conclusion

- browndye.ucsd.edu – code and documentation
- Computer Physics Communications 181 (11), 1896-1905 (2010)
- Thanks to Andy McCammon and others
- Howard Hughes Medical Institute, NIH