Introduction to Molecular Dynamics

Shikha Nangia

Scientific Exploration through Simulations (SETS)
Summer Workshop at Syracuse University
July 9-12, 2012, 201 Link Hall, Syracuse University

What is Molecular Dynamics?

Molecular dynamics (**MD**) is a computer simulation of physical movements of atoms and molecules.

The atoms and molecules are allowed to interact for a period of time, giving a view of the motion of the atoms

Trajectories of molecules and atoms are determined by solving the Newton's equations of motion

Newton’s equation of motion

The molecular dynamics simulation method is based on Newton’s second law or the equation of motion

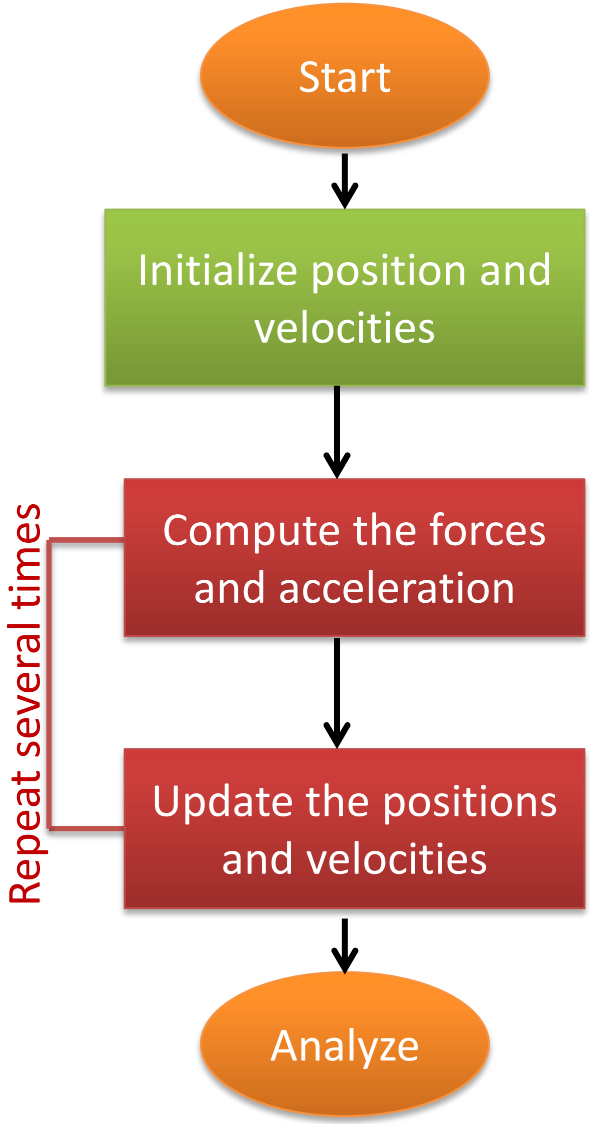
Force = mass x acceleration

From a knowledge of the force on each atom, it is possible to determine the acceleration of each atom in the system.

Integration of the equations of motion then yields a trajectory that describes the positions, velocities and accelerations of the particles as they vary with time.

Explaining Newton’s equation of motion

Explaining Newton’s equation of motion



Start

Initialize position and velocities

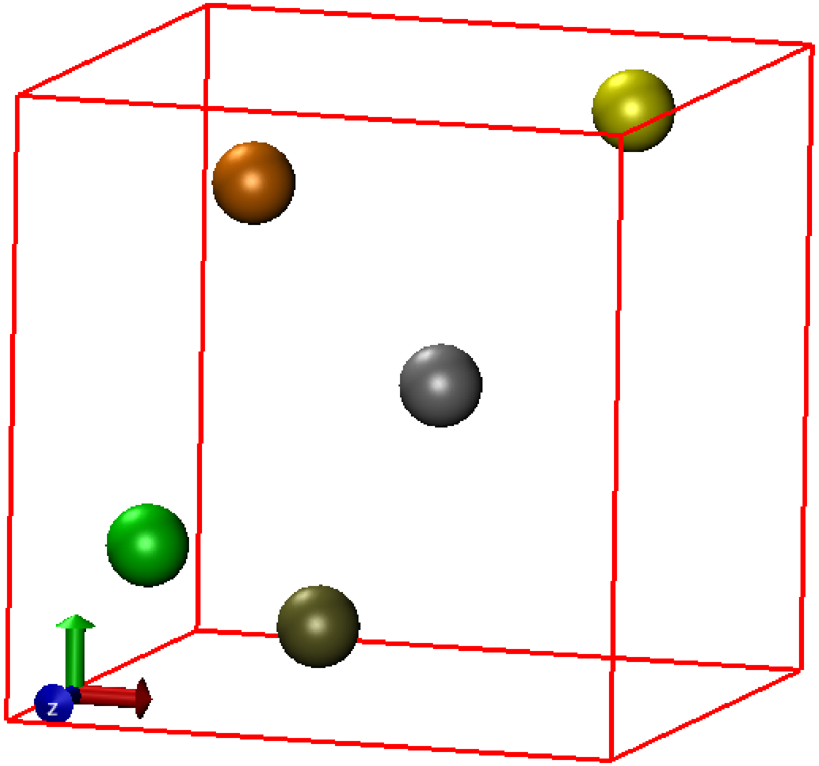
Compute the forces and acceleration

Update the positions and velocities

Repeat several times

Analyze

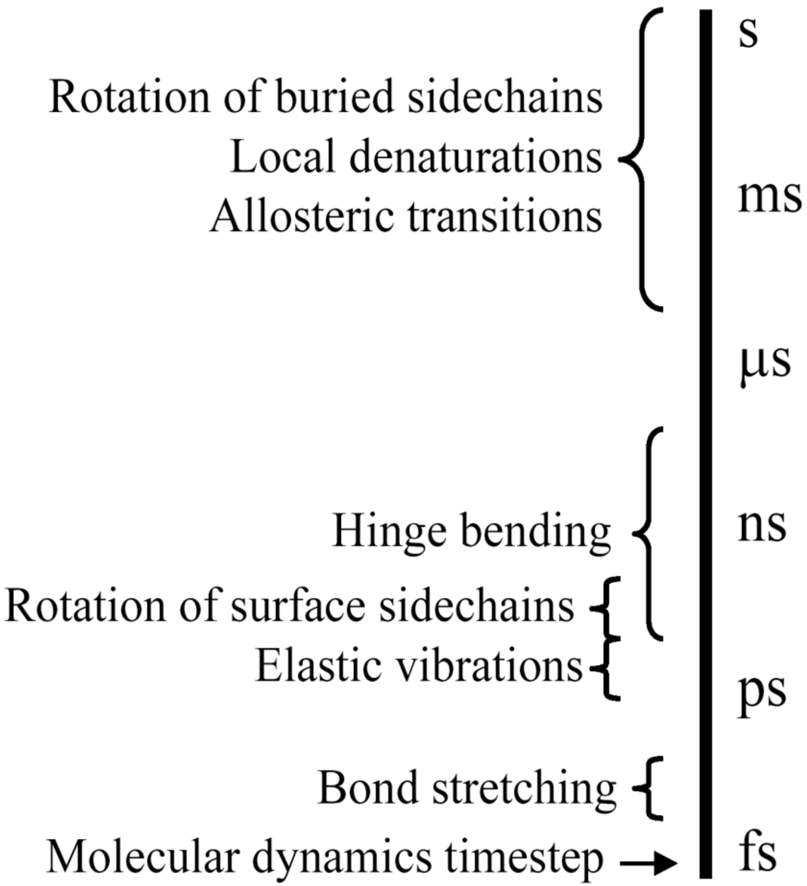
Explaining Newton’s equation of motion



3 dimensional simulation box

Simple example

Timescale Limitations



s

Rotation of buried sidechains

Local denaturations

Allosteric transitions

ms

μs

Hinge bending

ns

Rotation of surface sidechains

Elastic vibrations

ps

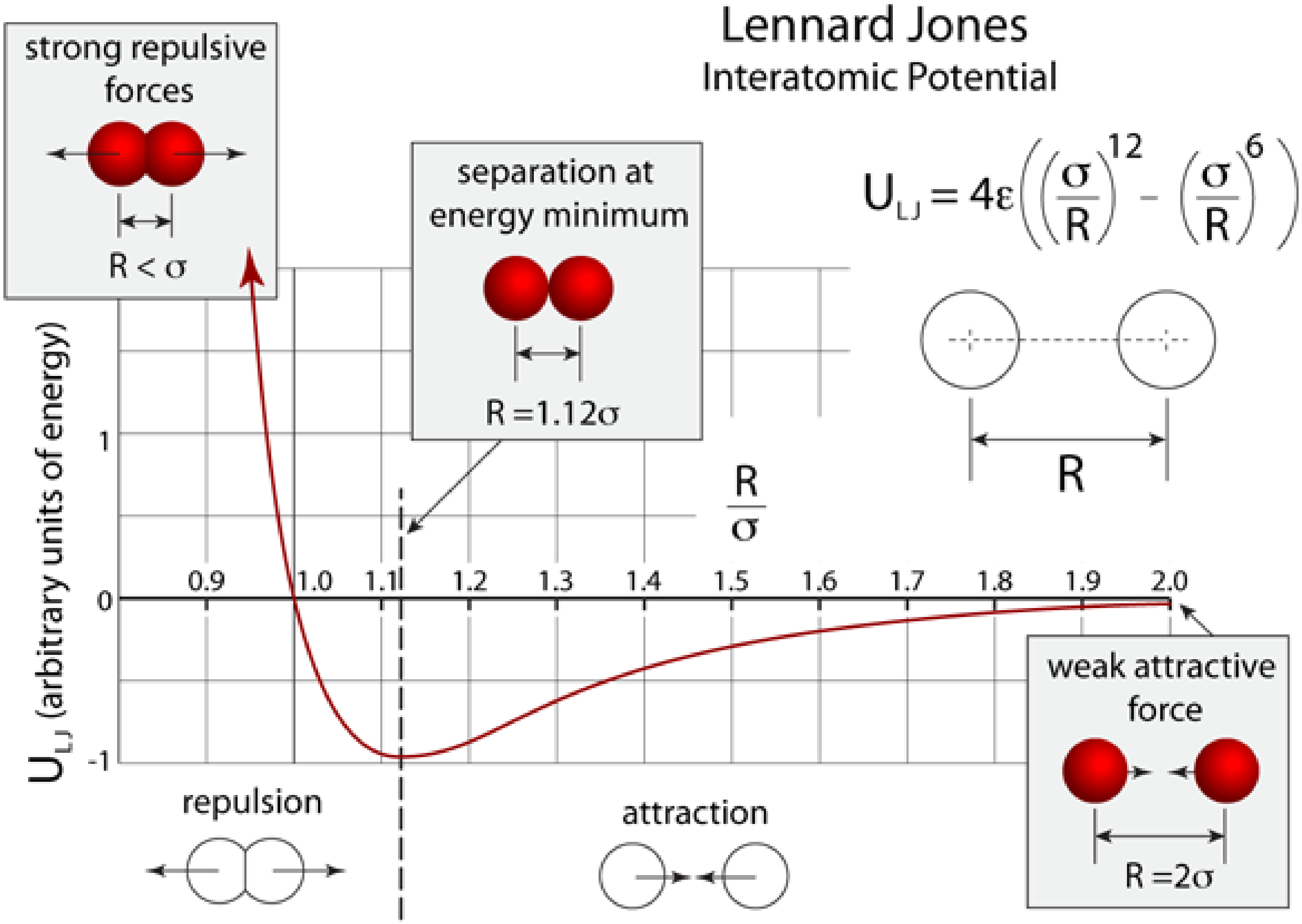
Bond stretching

Molecular dynamics timestep →

fs

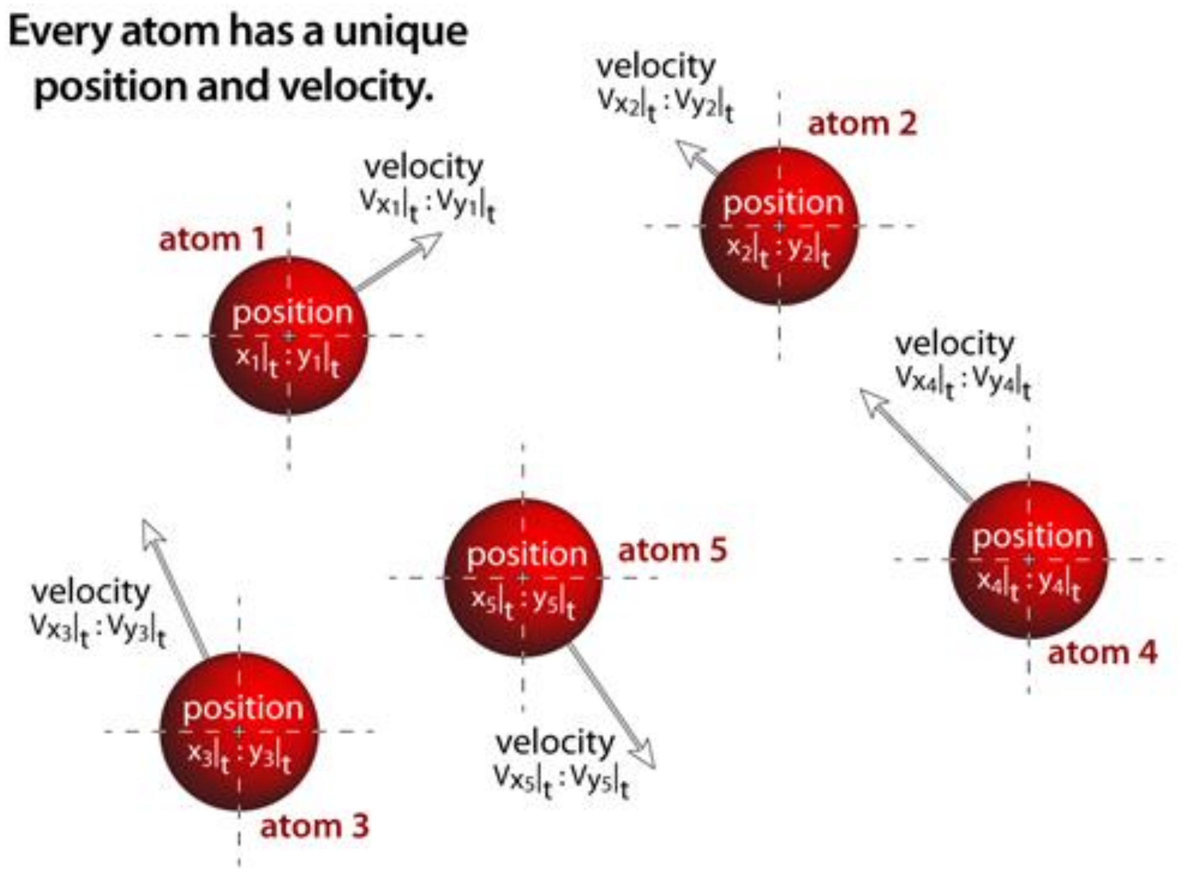
- Protein Folding - milliseconds/seconds (10^{-3}-1s)
- Ligand Binding - micro/milliseconds (10^{-6}-10^{-3} s)
- Enzyme catalysis - micro/milliseconds (10^{-6}-10^{-3} s)
- Conformational transitions - pico/nanoseconds (10^{-12}-10^{-9} s)
- Collective vibrations - 1 picosecond (10^{-12} s)
- Bond vibrations - **1 femtosecond (10^{-15} s)**

How do water atoms interact?



Simulation steps

Step 1.



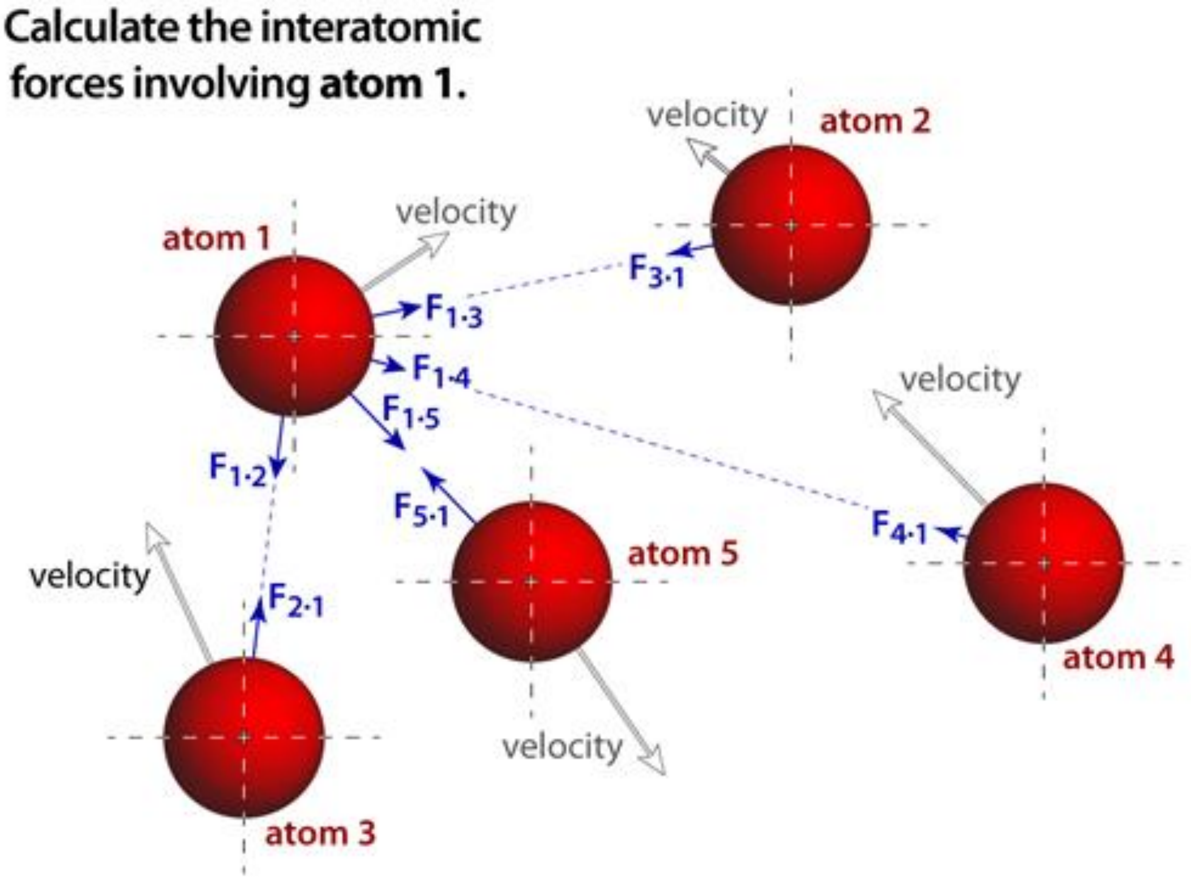
Every atom has a unique position and velocity.

a. Every atom has a unique position and velocity
b. All of the atoms follow Newton's laws of motion

http://www.atomsinmotion.com/book/chapter5/md

Simulation steps

Step 2.



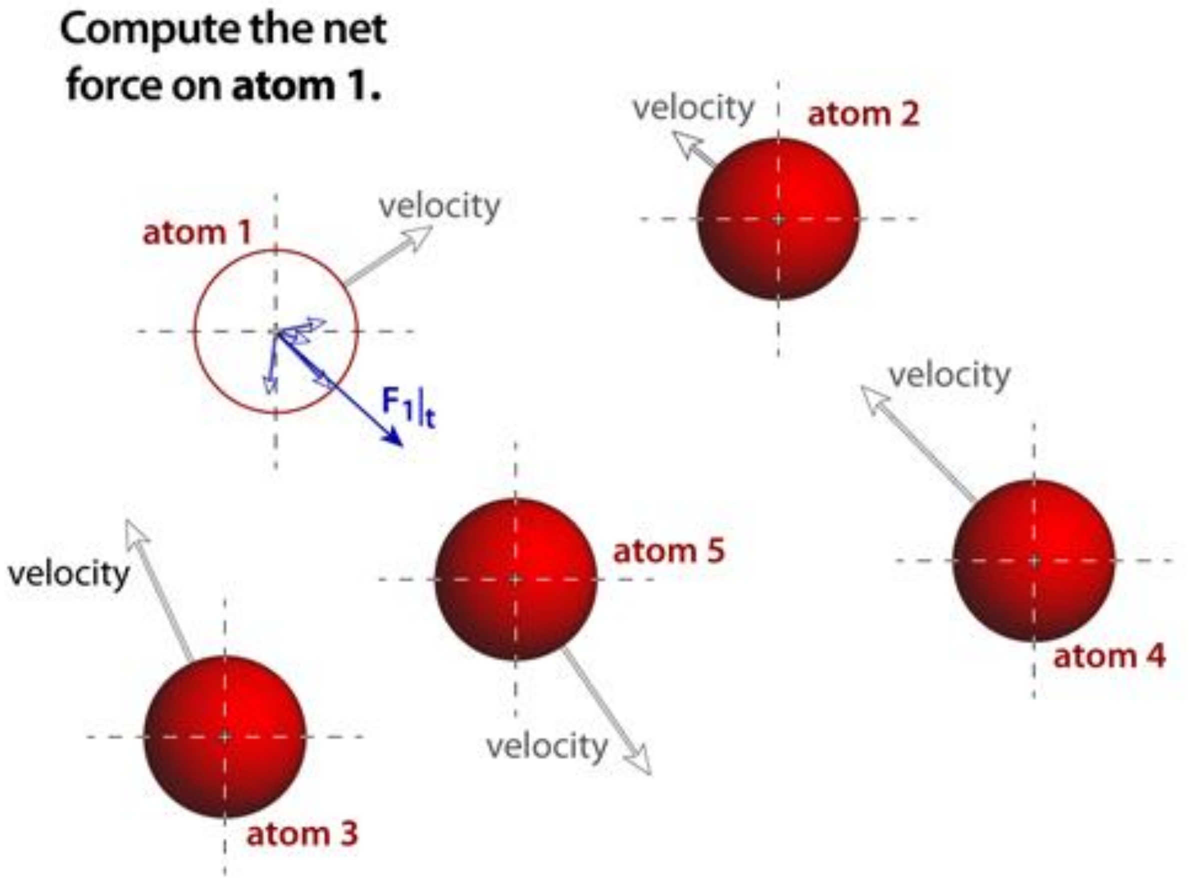
Calculate the interatomic forces involving **atom 1.**

The force that acts on an atom depends on its inetractions with he surrounding atoms

http://www.atomsinmotion.com/book/chapter5/md

Simulation steps

Step 3.

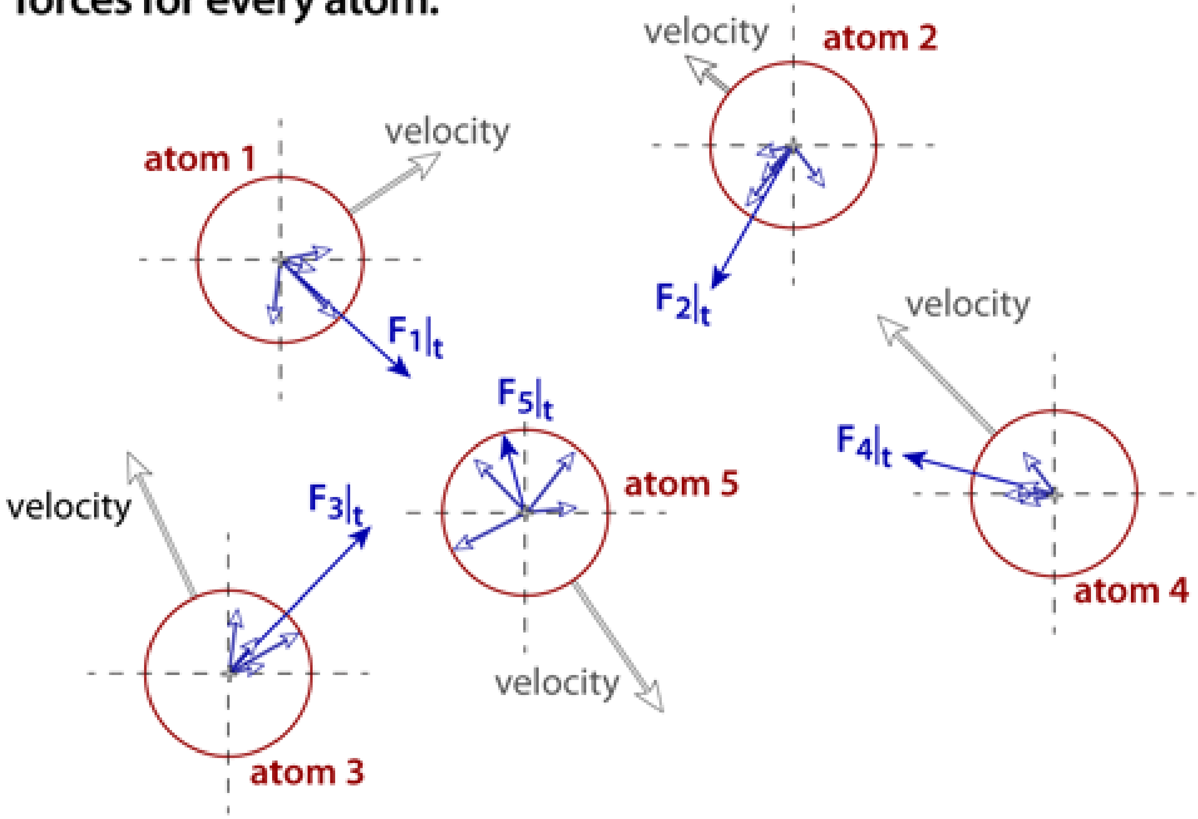


The net force acting on any atom is the vector sum of each of the individual forces that arise from the atomic pairs.

http://www.atomsinmotion.com/book/chapter5/md

Simulation steps

Step 4. Compute the interatomic forces for every atom.

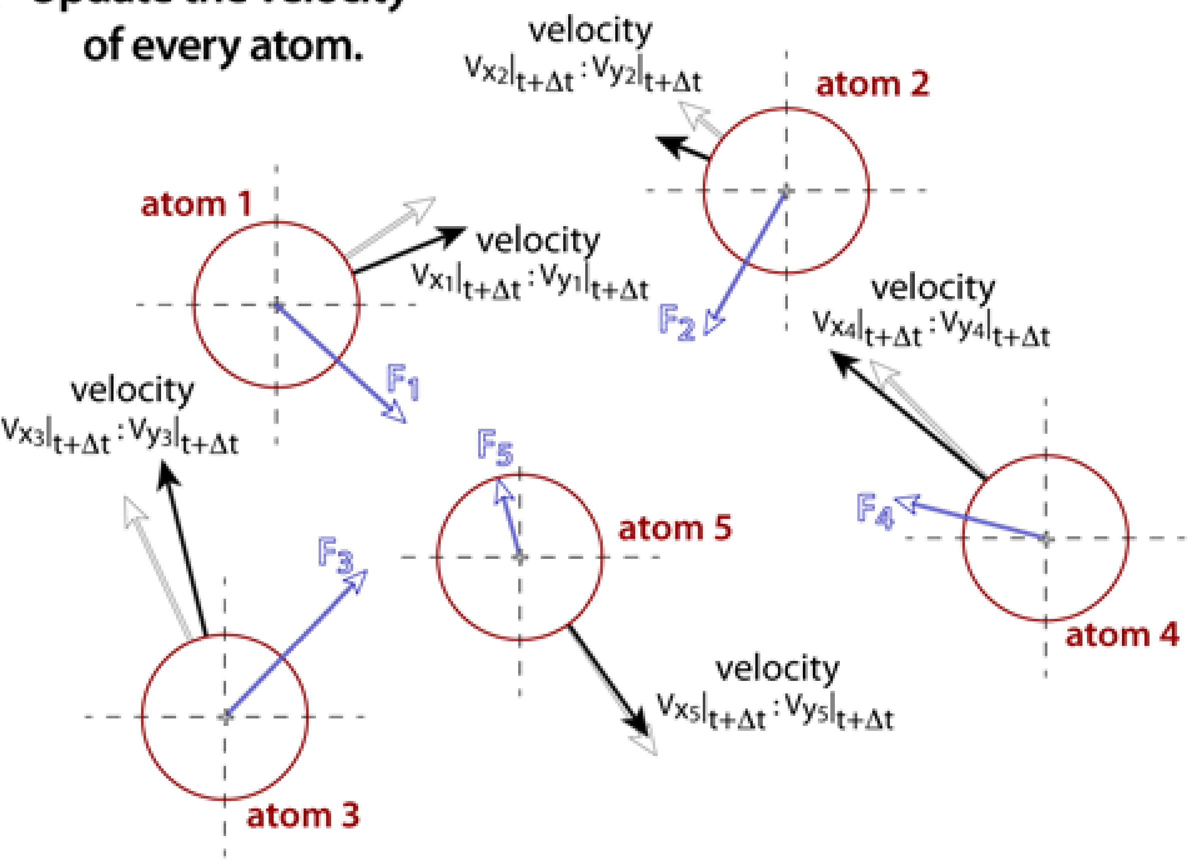


The net force must be computed for every atom in the simulation

http://www.atomsinmotion.com/book/chapter5/md

Simulation steps

Step 5. Update the velocity of every atom.


velocity
$V_{x2}|_{t+\Delta t} : V_{y2}|_{t+\Delta t}$
atom 2
atom 1
velocity
$V_{x1}|_{t+\Delta t} : V_{y1}|_{t+\Delta t}$
F_2
velocity
$V_{x4}|_{t+\Delta t} : V_{y4}|_{t+\Delta t}$
F_1
velocity
$V_{x3}|_{t+\Delta t} : V_{y3}|_{t+\Delta t}$
F_5
F_4
F_3
atom 5
atom 4
velocity
$V_{x5}|_{t+\Delta t} : V_{y5}|_{t+\Delta t}$
atom 3

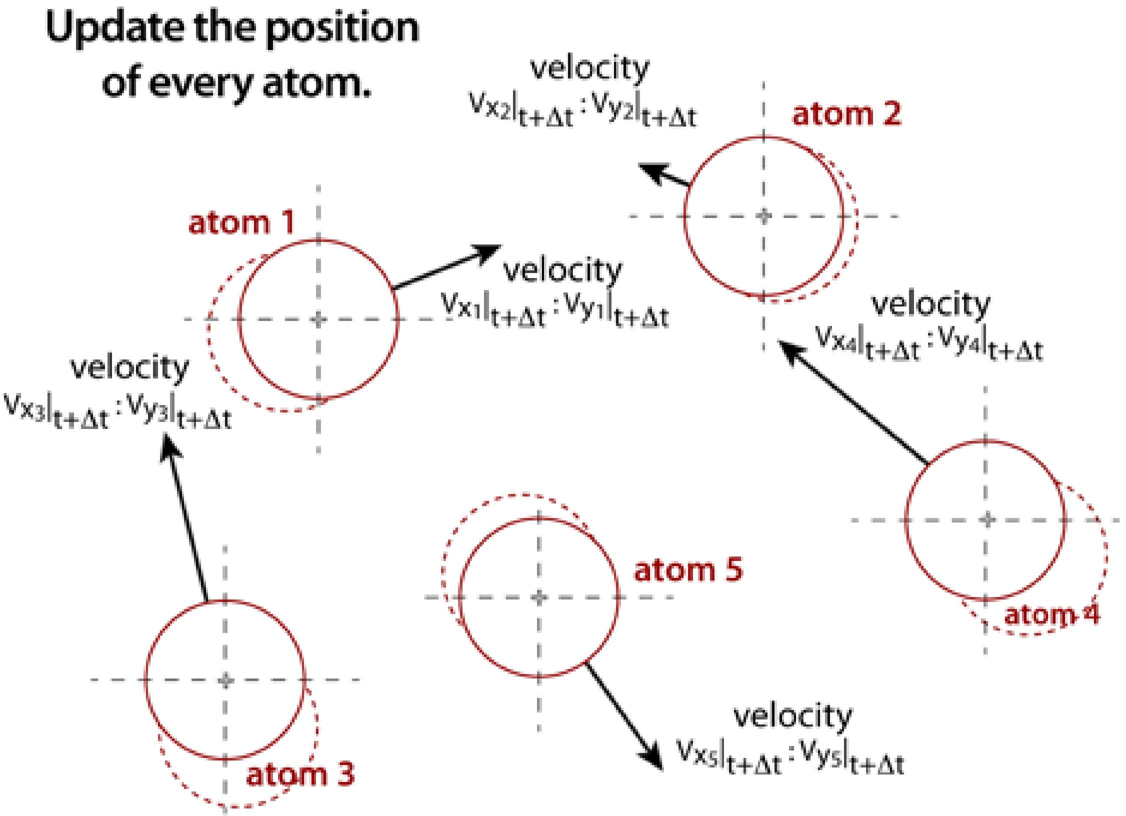
Using the force, velocity for each atom is computed

The time steps are very small, typically femtoseconds (1fs = 1×10^{-15} s)

http://www.atomsinmotion.com/book/chapter5/md

Simulation steps

Step 6.

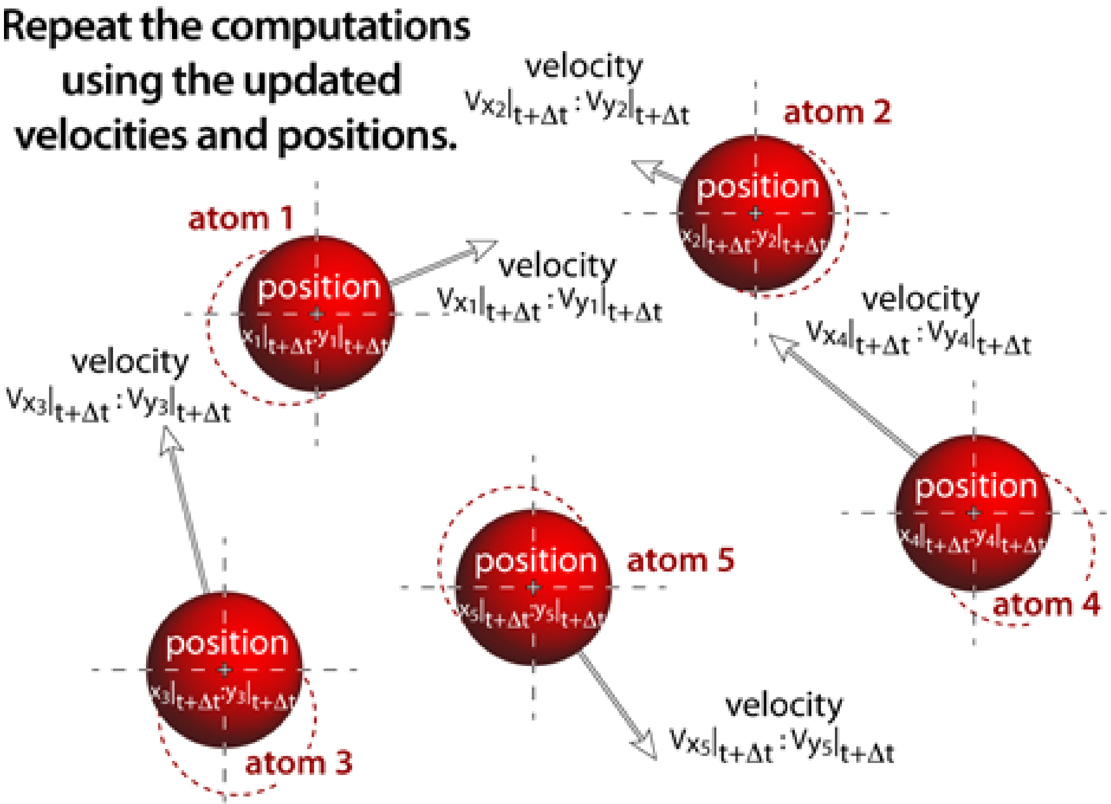

Update the position of every atom.
velocity
$Vx_2|_{t+\Delta t} : Vy_2|_{t+\Delta t}$
atom 2
atom 1
velocity
$Vx_1|_{t+\Delta t} : Vy_1|_{t+\Delta t}$
velocity
$Vx_4|_{t+\Delta t} : Vy_4|_{t+\Delta t}$
velocity
$Vx_3|_{t+\Delta t} : Vy_3|_{t+\Delta t}$
atom 5
atom 4
atom 3
velocity
$Vx_5|_{t+\Delta t} : Vy_5|_{t+\Delta t}$

The positions of each atom must also be updated

http://www.atomsinmotion.com/book/chapter5/md

Simulation steps

Step 7.

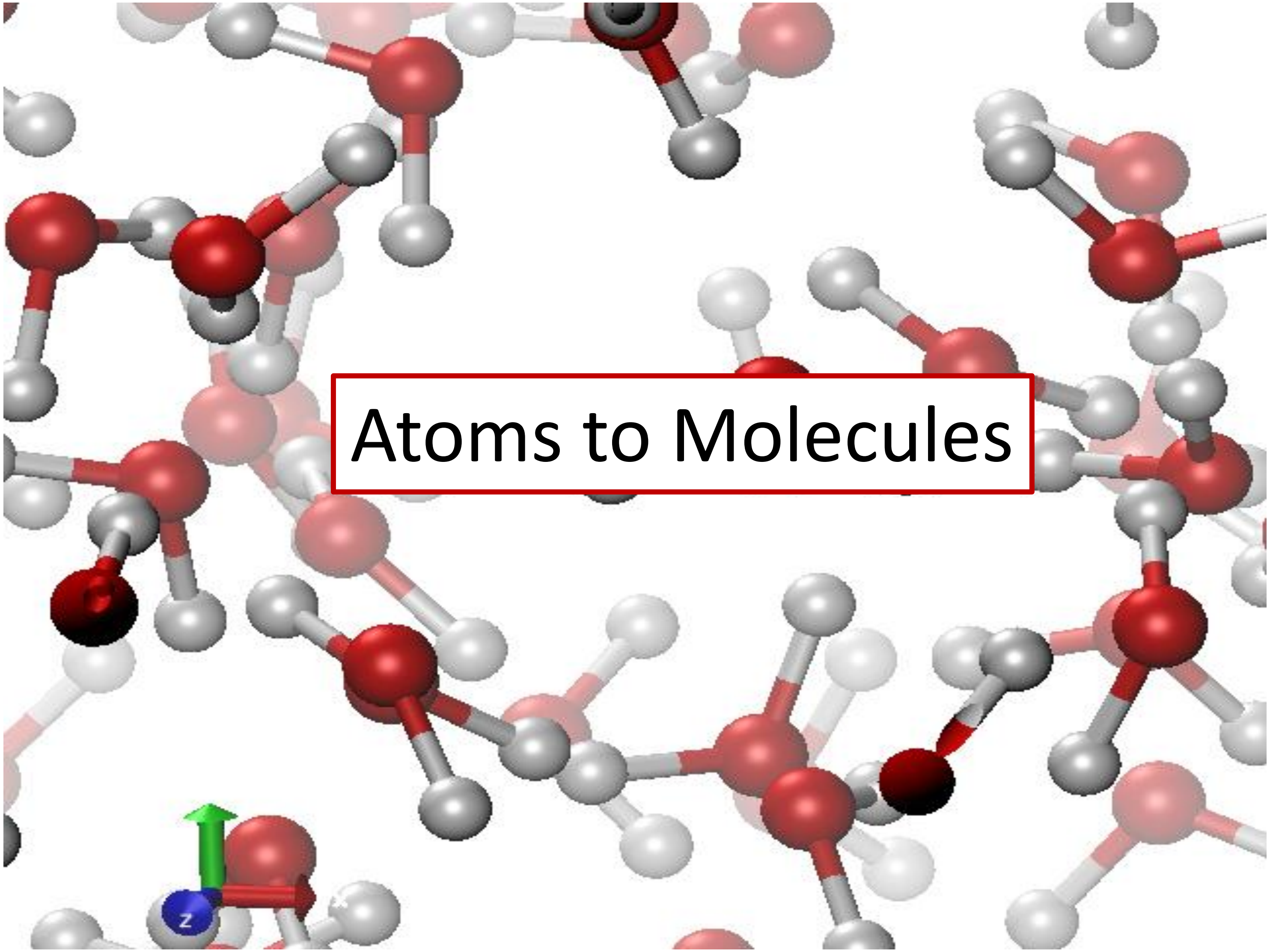


Repeat the computations using the updated velocities and positions.

Molecular dynamics experiments proceed iteratively, where the output from one iteration becomes the input to the next

http://www.atomsinmotion.com/book/chapter5/md

Atoms to Molecules



Example: Water in a beaker

Water

Chemical formula : H_2O

Atomic weights

Hydrogen = 1

Oxygen = 16

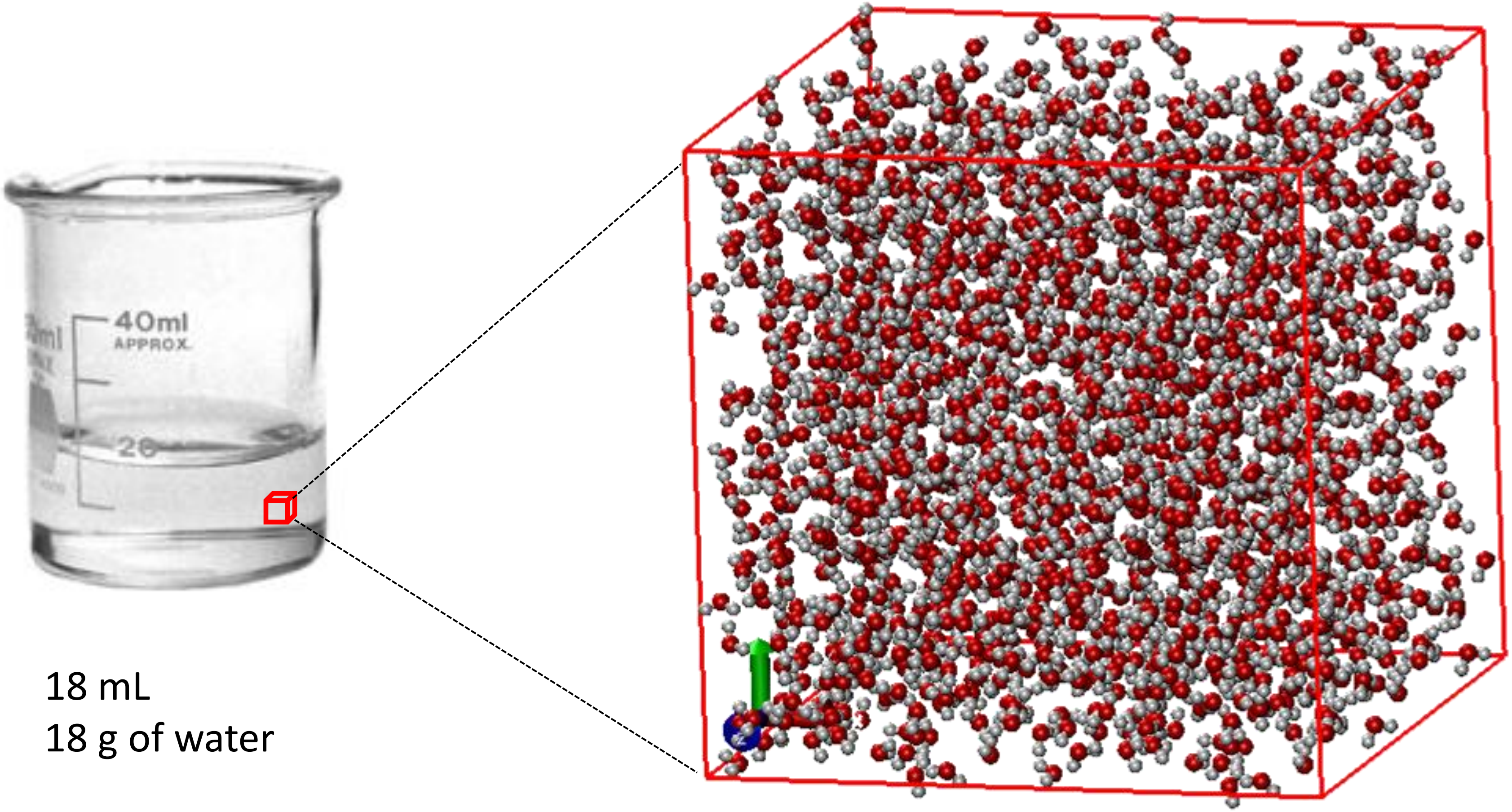
Number of water molecules = 6.023×10^{23}

Volume: 18 mL
Mass : 18 g

Molecule vibrate depending on temperature

http://www.d.umn.edu/~pkiprof/ChemWebV2/Vibrations/vib2.html

Simulating water



18 mL
18 g of water

A simulation box representing the macroscopic chemical environment such as Pressure, Temperature, and density is used

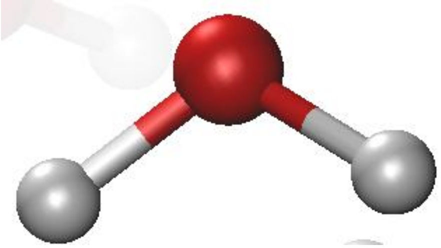
Simulating water

How do molecules interact?

Force field

Bonded interactions

Molecule itself

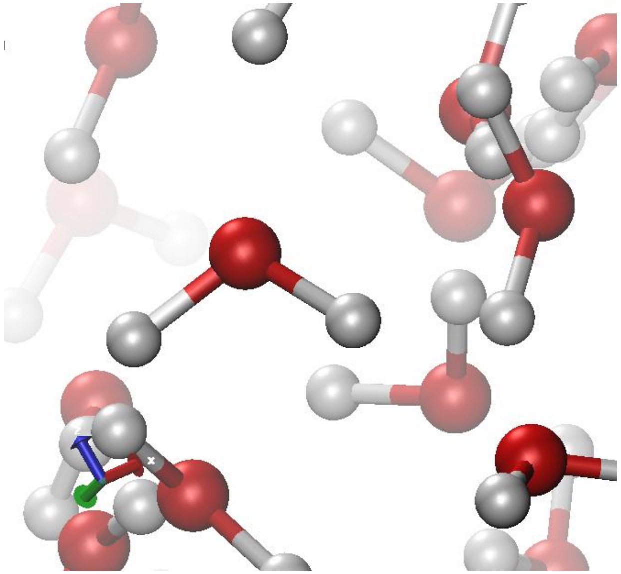


Bond Stretching

Angle bending

Non-bonded interactions

Environment effect



Van der Waal's interactions

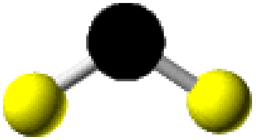
Ionic interactions

How do molecules interact?

Energy = Bonded interactions + Non-bonded interactions

For water molecules

Bond Stretching



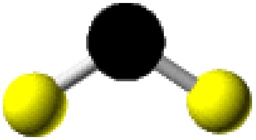
Symmetric stretch

How do molecules interact?

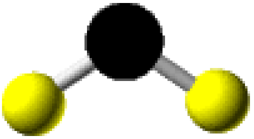
Energy = Bonded interactions + Non-bonded interactions

For water molecules

Bond Stretching



Symmetric stretch



asymmetric stretch

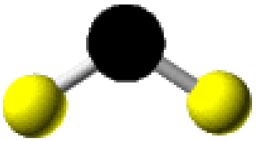
How do molecules interact?

Energy = Bonded interactions + Non-bonded interactions

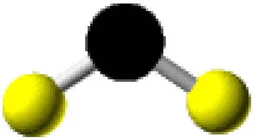
For water molecules

Bond Stretching

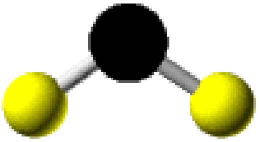
Angle bending



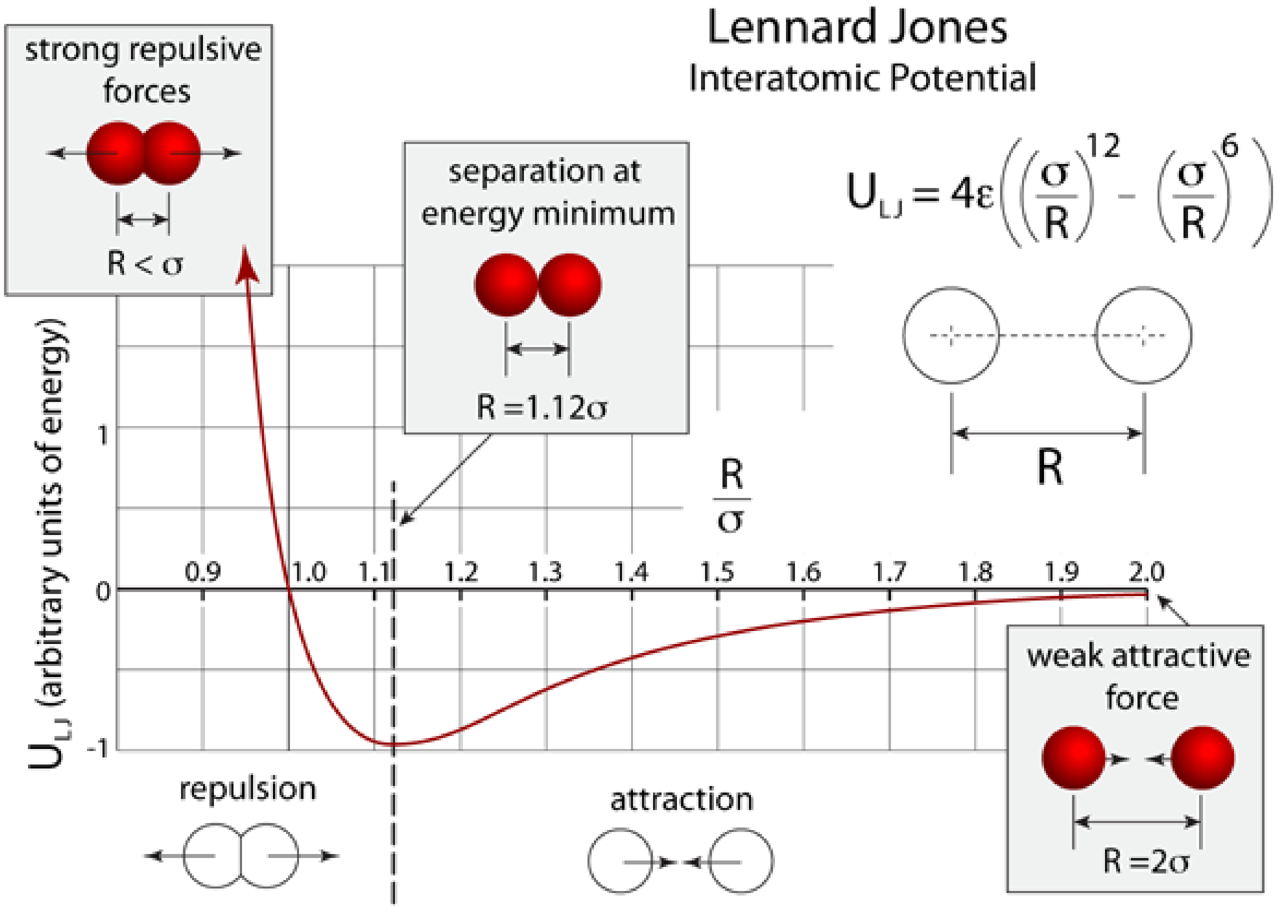
Symmetric stretch



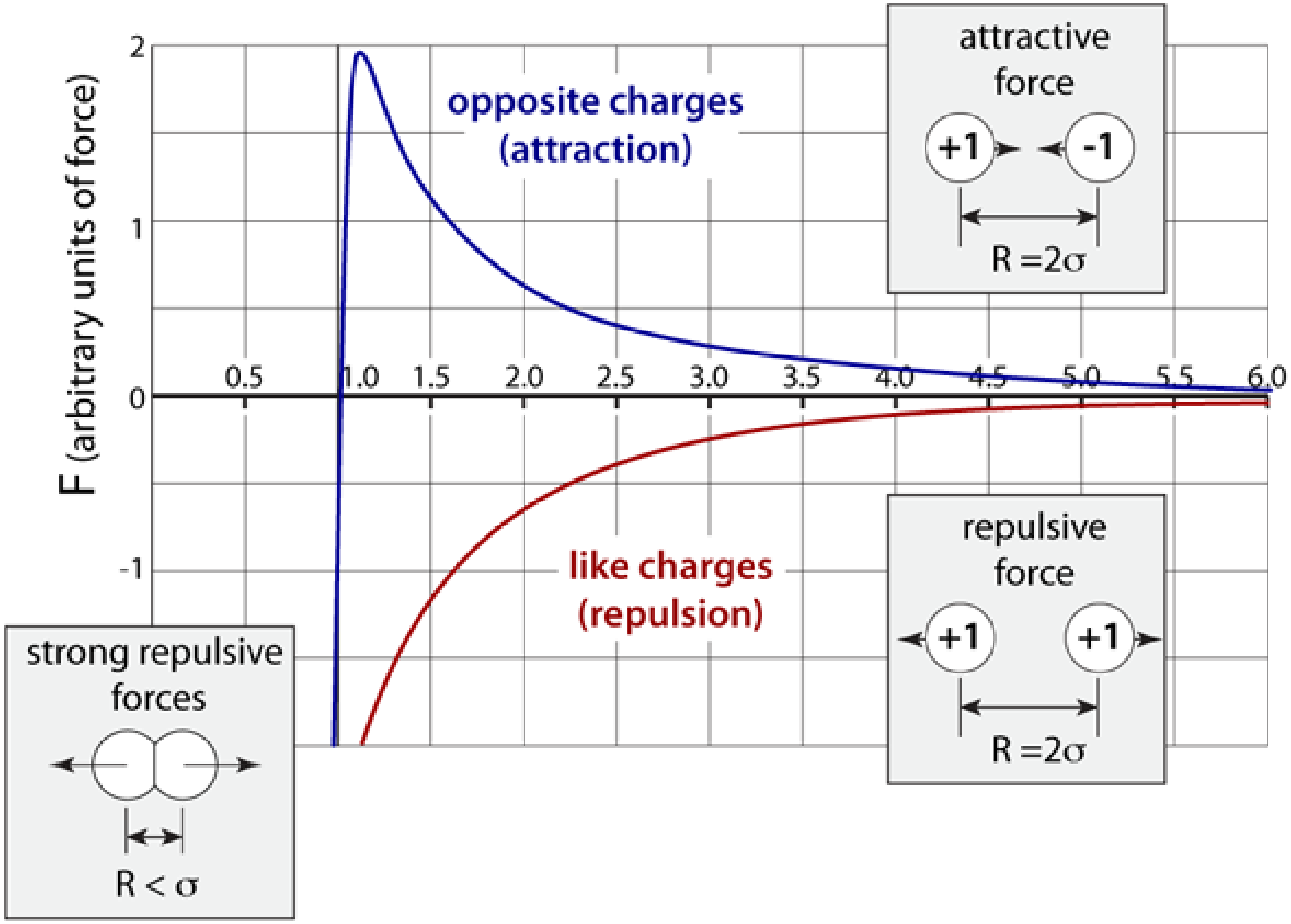
asymmetric stretch



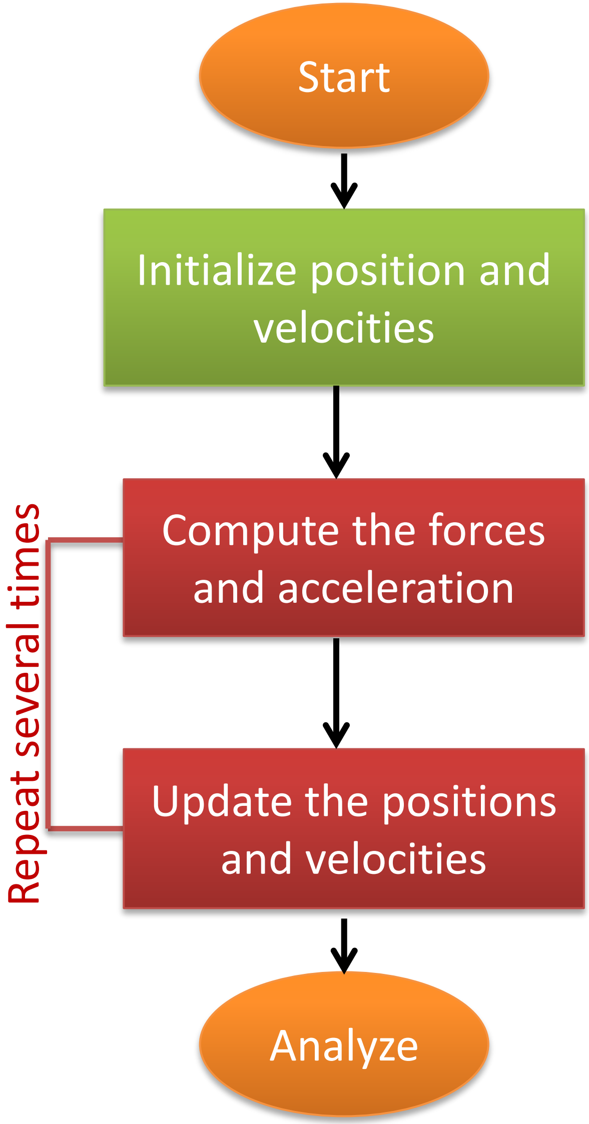
Non-bonded interactions



Ionic interactions

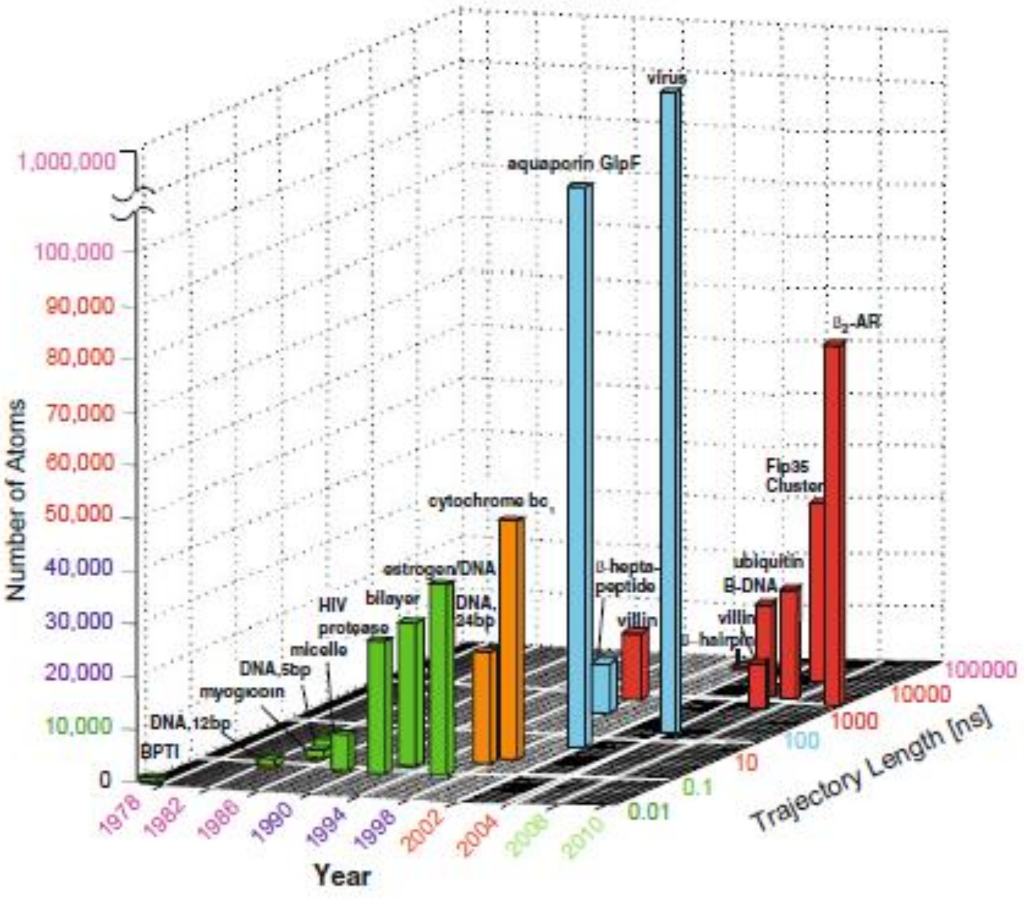


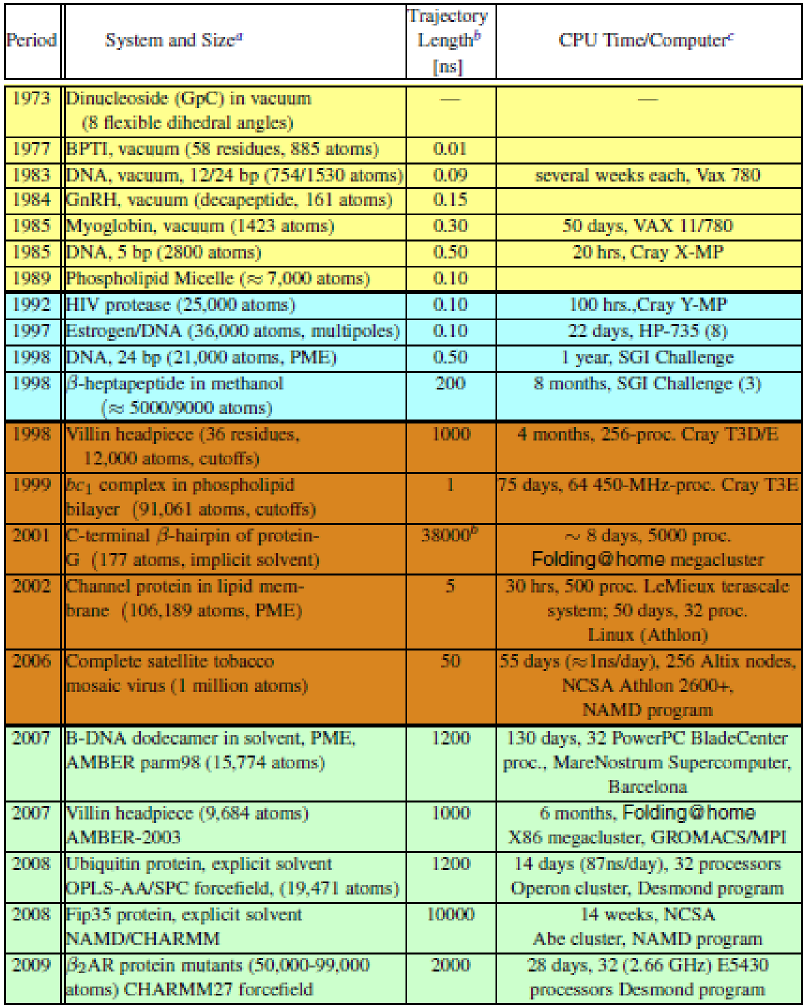
Molecular dynamics flow chart



Start
Initialize position and velocities
Compute the forces and acceleration
Update the positions and velocities
Repeat several times
Analyze

Evolution of Molecular Dynamcis Simulations



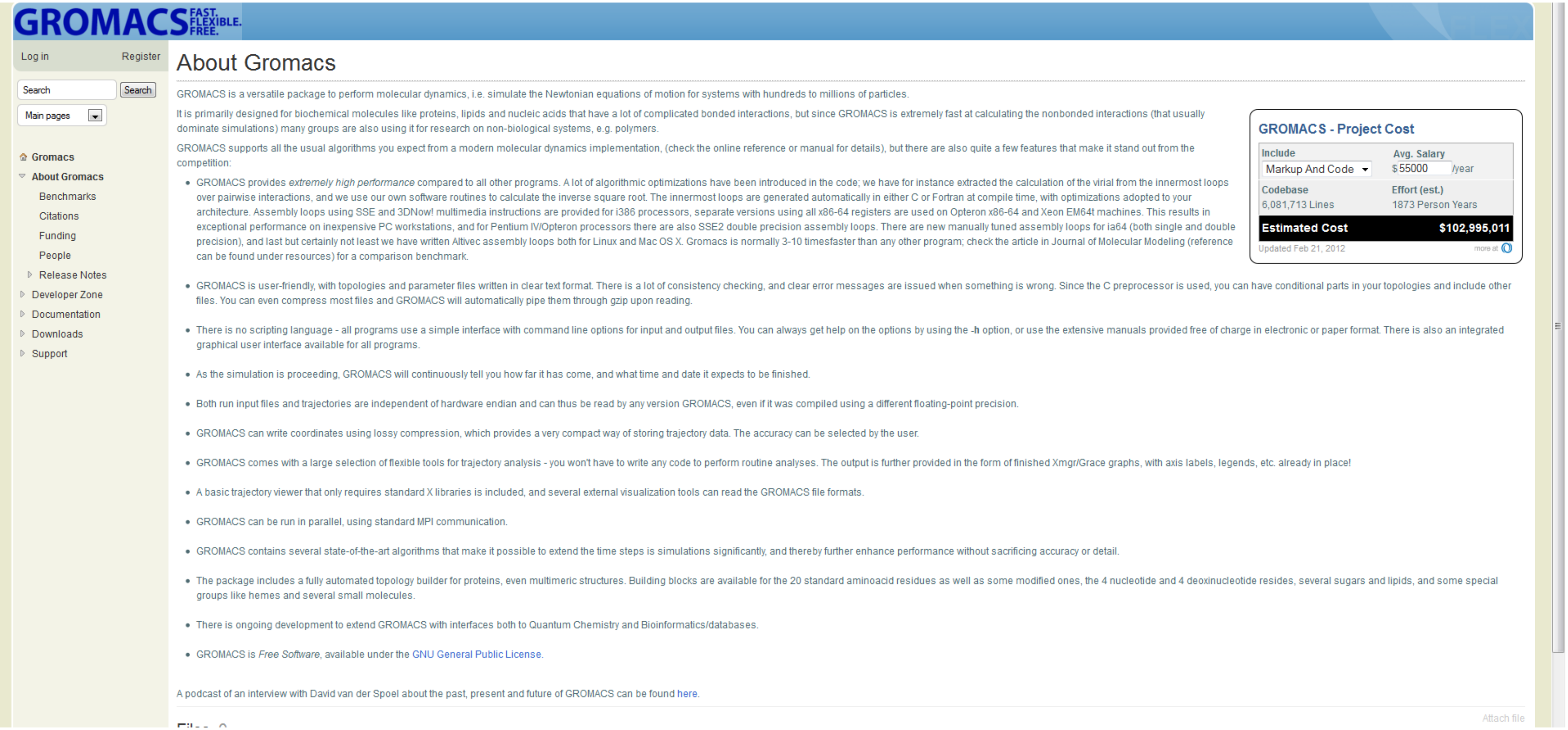


Period	System and Size^a	Trajectory Length^b [ns]	CPU Time/Computer^c
1973	Dinucleoside (GpC) in vacuum (8 flexible dihedral angles)	—	—
1977	BPTI, vacuum (58 residues, 885 atoms)	0.01	
1983	DNA, vacuum, 12/24 bp (754/1530 atoms)	0.09	several weeks each, Vax 780
1984	GnRH, vacuum (decapeptide, 161 atoms)	0.15	
1985	Myoglobin, vacuum (1423 atoms)	0.30	50 days, VAX 11/780
1985	DNA, 5 bp (2800 atoms)	0.50	20 hrs, Cray X-MP
1989	Phospholipid Micelle (≈ 7,000 atoms)	0.10	
1992	HIV protease (25,000 atoms)	0.10	100 hrs.,Cray Y-MP
1997	Estrogen/DNA (36,000 atoms, multipoles)	0.10	22 days, HP-735 (8)
1998	DNA, 24 bp (21,000 atoms, PME)	0.50	1 year, SGI Challenge
1998	β-heptapeptide in methanol (≈ 5000/9000 atoms)	200	8 months, SGI Challenge (3)
1998	Villin headpiece (36 residues, 12,000 atoms, cutoffs)	1000	4 months, 256-proc. Cray T3D/E
1999	bc_1 complex in phospholipid bilayer (91,061 atoms, cutoffs)	1	75 days, 64 450-MHz-proc. Cray T3E
2001	C-terminal β-hairpin of protein-G (177 atoms, implicit solvent)	38000^b	∼ 8 days, 5000 proc. Folding@home megacluster
2002	Channel protein in lipid membrane (106,189 atoms, PME)	5	30 hrs, 500 proc. LeMieux terascale system; 50 days, 32 proc. Linux (Athlon)
2006	Complete satellite tobacco mosaic virus (1 million atoms)	50	55 days (≈1ns/day), 256 Altix nodes, NCSA Athlon 2600+, NAMD program
2007	B-DNA dodecamer in solvent, PME, AMBER parm98 (15,774 atoms)	1200	130 days, 32 PowerPC BladeCenter proc., MareNostrum Supercomputer, Barcelona
2007	Villin headpiece (9,684 atoms) AMBER-2003	1000	6 months, Folding@home X86 megacluster, GROMACS/MPI
2008	Ubiquitin protein, explicit solvent OPLS-AA/SPC forcefield, (19,471 atoms)	1200	14 days (87ns/day), 32 processors Operon cluster, Desmond program
2008	Fip35 protein, explicit solvent NAMD/CHARMM	10000	14 weeks, NCSA Abe cluster, NAMD program
2009	β_2AR protein mutants (50,000-99,000 atoms) CHARMM27 forcefield	2000	28 days, 32 (2.66 GHz) E5430 processors Desmond program

(T. Schlick, Molecular Modelling and Simulation, 2010)

GROMACS

It is a freely available molecular dynamics software designed for simulations of biomolecular systems such as proteins and lipids



GROMACS FAST. FLEXIBLE. FREE.

Log in Register

Search [Search]

Main pages

- Gromacs
- About Gromacs
  - Benchmarks
  - Citations
  - Funding
  - People
  - Release Notes
- Developer Zone
- Documentation
- Downloads
- Support

About Gromacs

GROMACS is a versatile package to perform molecular dynamics, i.e. simulate the Newtonian equations of motion for systems with hundreds to millions of particles.

It is primarily designed for biochemical molecules like proteins, lipids and nucleic acids that have a lot of complicated bonded interactions, but since GROMACS is extremely fast at calculating the nonbonded interactions (that usually dominate simulations) many groups are also using it for research on non-biological systems, e.g. polymers.

GROMACS supports all the usual algorithms you expect from a modern molecular dynamics implementation, (check the online reference or manual for details), but there are also quite a few features that make it stand out from the competition:

- GROMACS provides *extremely high performance* compared to all other programs. A lot of algorithmic optimizations have been introduced in the code; we have for instance extracted the calculation of the virial from the innermost loops over pairwise interactions, and we use our own software routines to calculate the inverse square root. The innermost loops are generated automatically in either C or Fortran at compile time, with optimizations adopted to your architecture. Assembly loops using SSE and 3DNow! multimedia instructions are provided for i386 processors, separate versions using all x86-64 registers are used on Opteron x86-64 and Xeon EM64t machines. This results in exceptional performance on inexpensive PC workstations, and for Pentium IV/Opteron processors there are also SSE2 double precision assembly loops. There are new manually tuned assembly loops for ia64 (both single and double precision), and last but certainly not least we have written Altivec assembly loops both for Linux and Mac OS X. Gromacs is normally 3-10 timesfaster than any other program; check the article in Journal of Molecular Modeling (reference can be found under resources) for a comparison benchmark.
- GROMACS is user-friendly, with topologies and parameter files written in clear text format. There is a lot of consistency checking, and clear error messages are issued when something is wrong. Since the C preprocessor is used, you can have conditional parts in your topologies and include other files. You can even compress most files and GROMACS will automatically pipe them through gzip upon reading.
- There is no scripting language - all programs use a simple interface with command line options for input and output files. You can always get help on the options by using the **-h** option, or use the extensive manuals provided free of charge in electronic or paper format. There is also an integrated graphical user interface available for all programs.
- As the simulation is proceeding, GROMACS will continuously tell you how far it has come, and what time and date it expects to be finished.
- Both run input files and trajectories are independent of hardware endian and can thus be read by any version GROMACS, even if it was compiled using a different floating-point precision.
- GROMACS can write coordinates using lossy compression, which provides a very compact way of storing trajectory data. The accuracy can be selected by the user.
- GROMACS comes with a large selection of flexible tools for trajectory analysis - you won't have to write any code to perform routine analyses. The output is further provided in the form of finished Xmgr/Grace graphs, with axis labels, legends, etc. already in place!
- A basic trajectory viewer that only requires standard X libraries is included, and several external visualization tools can read the GROMACS file formats.
- GROMACS can be run in parallel, using standard MPI communication.
- GROMACS contains several state-of-the-art algorithms that make it possible to extend the time steps is simulations significantly, and thereby further enhance performance without sacrificing accuracy or detail.
- The package includes a fully automated topology builder for proteins, even multimeric structures. Building blocks are available for the 20 standard aminoacid residues as well as some modified ones, the 4 nucleotide and 4 deoxinucleotide resides, several sugars and lipids, and some special groups like hemes and several small molecules.
- There is ongoing development to extend GROMACS with interfaces both to Quantum Chemistry and Bioinformatics/databases.
- GROMACS is *Free Software*, available under the GNU General Public License.

A podcast of an interview with David van der Spoel about the past, present and future of GROMACS can be found here.

GROMACS - Project Cost

Include	Avg. Salary
Markup And Code ▾	$ 55000 /year
Codebase	**Effort (est.)**
6,081,713 Lines	1873 Person Years
Estimated Cost	**$102,995,011**

Updated Feb 21, 2012 more at

Files 0

Attach file

http://www.gromacs.org/