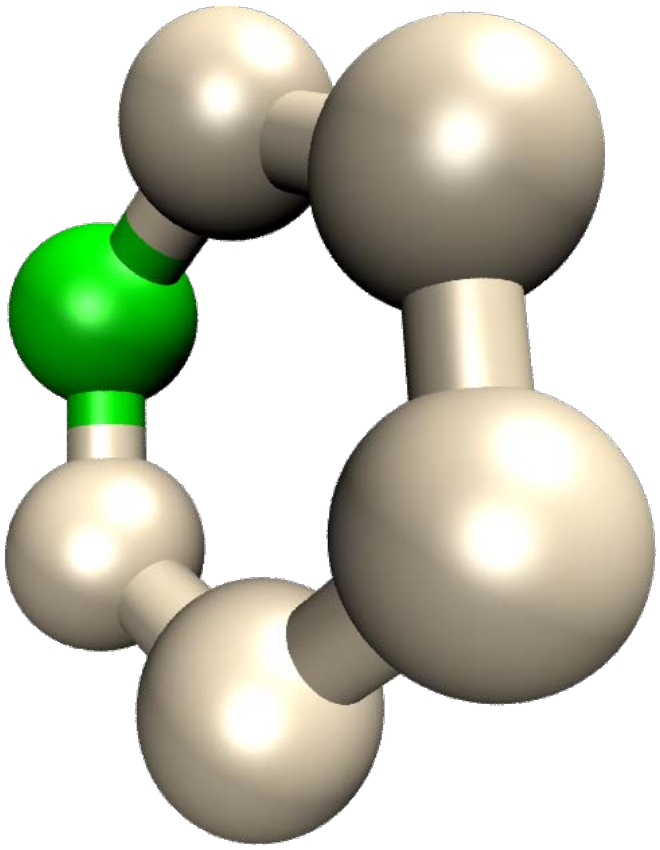
OpenMM Zephyr



User's Manual

Release 0.5

March 29, 2009

Copyright and Permission Notice

Portions copyright (c) 2009 Stanford University and Christopher Bruns.
Contributors: Vijay Pande, Joy Ku, Jeanette Schmidt, Mark Friedrichs, Peter Eastman

Permission is hereby granted, free of charge, to any person obtaining a copy of this document (the "Document"), to deal in the Document without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Document, and to permit persons to whom the Document is furnished to do so, subject to the following conditions:

This copyright and permission notice shall be included in all copies or substantial portions of the Document.

THE DOCUMENT IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS, CONTRIBUTORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE DOCUMENT OR THE USE OR OTHER DEALINGS IN THE DOCUMENT.

Acknowledgments

SimTK software and all related activities are funded by the Simbios National Center for Biomedical Computing through the National Institutes of Health Roadmap for Medical Research, Grant U54 GM072970. Information on the National Centers can be found at http://nihroadmap.nih.gov/bioinformatics.

Table of Contents

List of Figures

1 Overview

1.1 Introduction to Molecular Dynamics

Researchers are increasingly using molecular dynamics (MD) simulations to model molecular motion and help expand our knowledge and understanding of biology. Examples of how MD simulations have been used include:

- Providing hypotheses for biological phenomena that cannot currently be observed experimentally and guiding new experiments [(Marszalek & al., 1999), (Herce & Garcia, 2007), (Kasson & Pande, 2008)]
- Identifying which engineered molecules would be stable or would bind with another molecule [(Kasson & Pande, 2008)]
- Augmenting information provided by static structures, i.e., from x-ray crystallography [(Glazer, RJ, & Altman, 2008)]

The potential uses for MD simulations are exciting, but getting started can be overwhelming and time-consuming. OpenMM Zephyr was designed to get researchers running MD simulations quickly.

1.2 What is OpenMM Zephyr?

OpenMM Zephyr is a freely downloadable application with a graphical user interface that allows users to easily run the OpenMM version of GROMACS, a widely used MD package. The OpenMM version speeds up the GROMACS simulations, so that more complex molecules and/or longer processes can be simulated. Read more about OpenMM and how to take full advantage of it in Section 1.3.

OpenMM Zephyr leads the user through the basic steps required to set up and run a simulation, and also to visualize the simulation results using VMD, a popular molecular visualization program. In addition, OpenMM Zephyr displays the specific GROMACS

commands being used, enabling motivated users to teach themselves how to run GROMACS from the command line so that they can eventually use the other options within GROMACS.

1.2.1 Limitations of OpenMM Zephyr

- Windows XP or Vista only
- 32 bit operating system
- Implicit solvent (no explicit water molecules)
- Amber96 force field only
- Support for only the 20 standard amino acids and 5 canonical nucleotide residues
- For GPU accelerated dynamics:
  - You need to have a supported GPU and drivers (Sections 1.3.1 and 2.2)

1.3 What is OpenMM?

OpenMM is a freely downloadable, high performance, extensible library that allows molecular dynamics simulations to run on high performance computer architectures, such as computer clusters and graphics processing units (GPUs). This makes it more reasonable to simulate complex molecules and/or longer processes. Performance speed ups of over 100X have been achieved in some cases using OpenMM on GPUs. [(Friedrichs, et al., 2009)]

1.3.1 Running OpenMM on GPUs

Preview release 2 of OpenMM enables fast MD simulations on NVIDIA and ATI GPUs. GPUs are computer chips designed for generating computer graphics. Their parallel design makes them also very useful for speeding up a variety of other applications, including MD simulations.

To take advantage of these speed-ups, though, your computer needs to be equipped with one of the supported GPU cards:

Supported NVIDIA GPUs:
http://www.nvidia.com/object/cuda_learn_products.html

Supported ATI GPUs on desktops (Note supported operating systems):
http://ati.amd.com/technology/streamcomputing/requirements.html

Supported ATI Mobility Radeon boards on laptops running Windows XP:
http://game.amd.com/us-en/drivers_catalyst.aspx?p=xp/mobility-xp

Your computer does not need one of these GPU cards to run OpenMM Zephyr, but your simulations will not run as fast without one.

1.3.2 More information about OpenMM

You can learn more about the OpenMM project at http://simtk.org/home/openmm.

1.4 Conventions Used in this Document

This warning icon denotes situations that require special attention or caution.

The clicky pointer icon shows where to click on screen shots

2 Installing OpenMM Zephyr

2.1 Prerequisites

In order to run OpenMM Zephyr, you need a computer that runs Windows XP or Windows Vista. Linux and MacOS will be supported in the future.

You will also need to install Tcl/Tk (see Section 2.1.1). And if you wish to use the visualization capabilities of OpenMM Zephyr, you will need to install VMD (see Section 2.1.2).

To take advantage of GPU accelerated molecular dynamics, you must have a supported GPU (see section 1.3.1). You will also need to have the special programming language(s) used for your particular GPU: CUDA for NVIDIA GPUs or CAL and Brook for ATI GPUs (see Section 2.2).

2.1.1 Installing Tcl/Tk version 8.5

OpenMM Zephyr requires Tcl/Tk, version 8.5. You can download this version for free from http://www.activestate.com/activetcl/ (Figure 2-1).

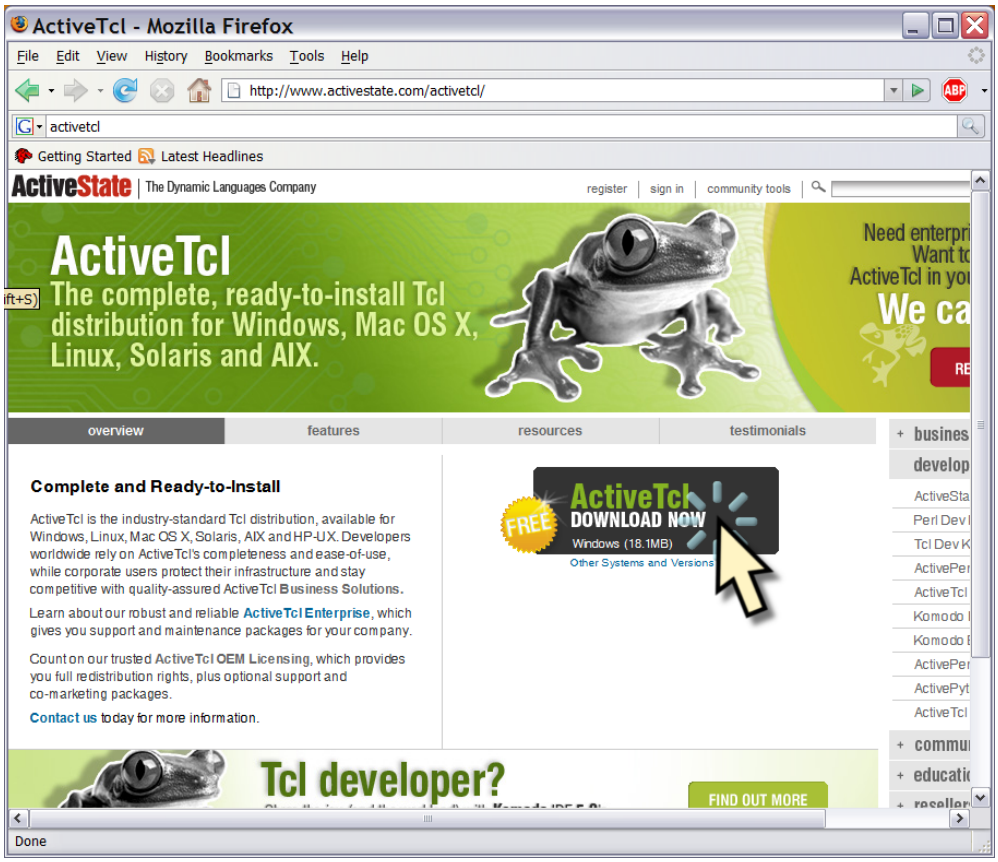


Figure 2-1: Website for downloading Tcl/Tk

From the website, click on the "Download Now" button. A dialog window will appear asking if you want to "Run" or "Save" the program. If you do not see the dialog window, make sure your pop-up blocker has been disabled. Click on "Save" and note where you save the program!

Double-click on the program you just downloaded to install it. You may get a "Security Warning" that the "publisher could not be verified." Go ahead and click "Run." Follow the directions that appear on the screen to complete the installation.

2.1.2 Installing VMD (optional but recommended)

OpenMM Zephyr also uses the molecular visualization program VMD. If you do not intend on visualizing the MD simulation results within OpenMM Zephyr, you do not need to install VMD. However, to take full advantage of OpenMM Zephyr, it is recommended that you install VMD.

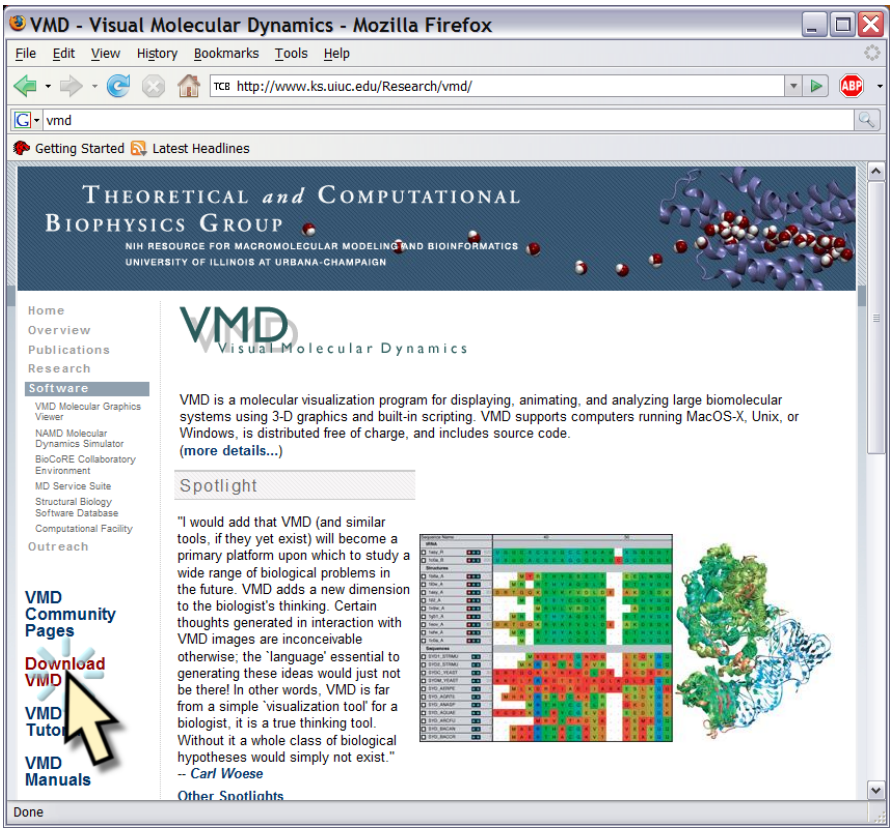


Figure 2-2: Download VMD from the University of Illinois

VMD can be downloaded for free from http://www.ks.uiuc.edu/Research/vmd/. Click on the "Download VMD" link in the left-hand column (Figure 2-2). You will be taken to another webpage, listing all the different versions of VMD. Select the Windows OpenGL distribution of the most recent version.

You will be taken to a registration/login window. Provide a username and password, and accept the license agreement that appears.

A dialog window will then appear asking if you want to "Run" or "Save" the program. If you do not see the dialog window, make sure your pop-up blocker has been disabled. Click on "Save" and note where you save the program!

Double-click on the program you just downloaded to install it. You may get a "Security Warning" that the "publisher could not be verified." Go ahead and click "Run." Follow the directions that appear on the screen to complete the installation.

2.2 Installing GPU software

If you intend to use GPU accelerated molecular dynamics, you need to install CUDA (for NVIDIA GPUs) or CAL and Brook (for ATI GPUs), and test them before running OpenMM Zephyr.

2.2.1 Installing CUDA for NVIDIA GPUs

For NVIDIA GPUs, you need to have CUDA version 2.0 or later installed to get the GPU acceleration. It is recommended that you test your installation before trying to run OpenMM Zephyr.

1. Go to http://www.nvidia.com/object/cuda_get.html

2. Download and install the CUDA Driver, the CUDA Toolkit, and the CUDA SDK code samples. You need version 2.0 or later. The driver and toolkit are needed to get the GPU acceleration. The code samples are required for testing purposes.

3. To verify that you've installed things correctly, run a sample program available with the SDK code samples.

    a. Go to Start -> All Programs -> NVIDIA Corporation -> NVIDIA CUDA SDK -> NVIDIA CUDA SDK Browser

    b. A window appears showing all the different sample programs you can try running (Figure 2-3).

    c. Locate the program "Device Query" and click on the associated "Run" link on the right-hand side. If things are running correctly, a window will appear stating how many devices are running CUDA (there should be at least 1) and that it/they passed the test.

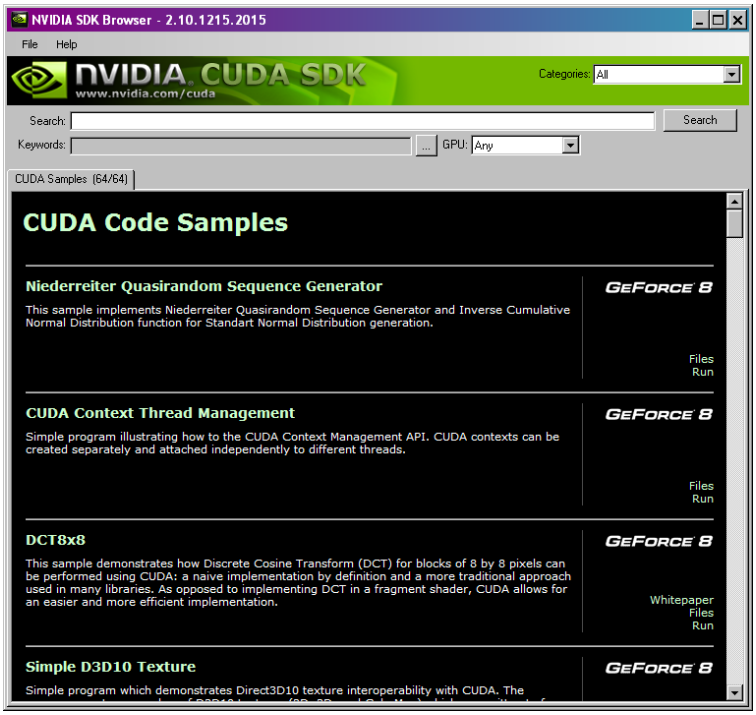


Figure 2-3: Window for browsing the NVIDIA code samples

2.2.2 Installing CAL and Brook for ATI GPUs

For ATI GPUs, you must have CAL and Brook installed to get the GPU acceleration. Catalyst Software Suite 8.12 or later provide these.

2.2.2.1 Checking Catalyst Software Suite version

On Windows, you can check what version of Catalyst Software Suite you have:

1. Go to Start -> All Programs -> Catalyst Control Center -> Catalyst Control Center

2. In the Catalyst Control Center, click Information Center -> Graphics Software -> Catalyst Version. If you don't see a listing for Catalyst Version, you don't have the driver at all and need to download it. Only versions 8.12 and later have the needed software.

2.2.2.2 Other software required

Check http://ati.amd.com/technology/streamcomputing/requirements.html for the latest information about supported operating systems. In particular, you may need to update your Windows operating system to have the latest service pack (SP).

For Windows, you will also need Microsoft .NET Framework 2.0. To check to see if you already have it:

1. Go to Start -> Control Panel -> Add or Remove Programs

2. Look to see if Microsoft .NET Framework 2.0 is listed in the window that appears. If not, you need to download and install it. The AMD/ATI website provides a link for your specific platform via their on-line installation instructions. You can also go directly to Microsoft to download it:

   http://msdn.microsoft.com/en-us/netframework/aa731542.aspx

   Select the redistributable package that corresponds to your computer: x86 version, x64 (64-bit) version or the IA64 (Intel 64-bit) version.

2.2.2.3 Downloading and installing the Catalyst Software Suite

To get the Catalyst Software Suite, which contains the versions of CAL and Brook needed, go to: http://ati.amd.com/support/driver.html. Make sure you get version 8.12 or later.

2.2.2.4 Testing your installation

To verify that you've installed things correctly, download and install SDK from http://ati.amd.com/technology/streamcomputing/sdkdwnld.html.

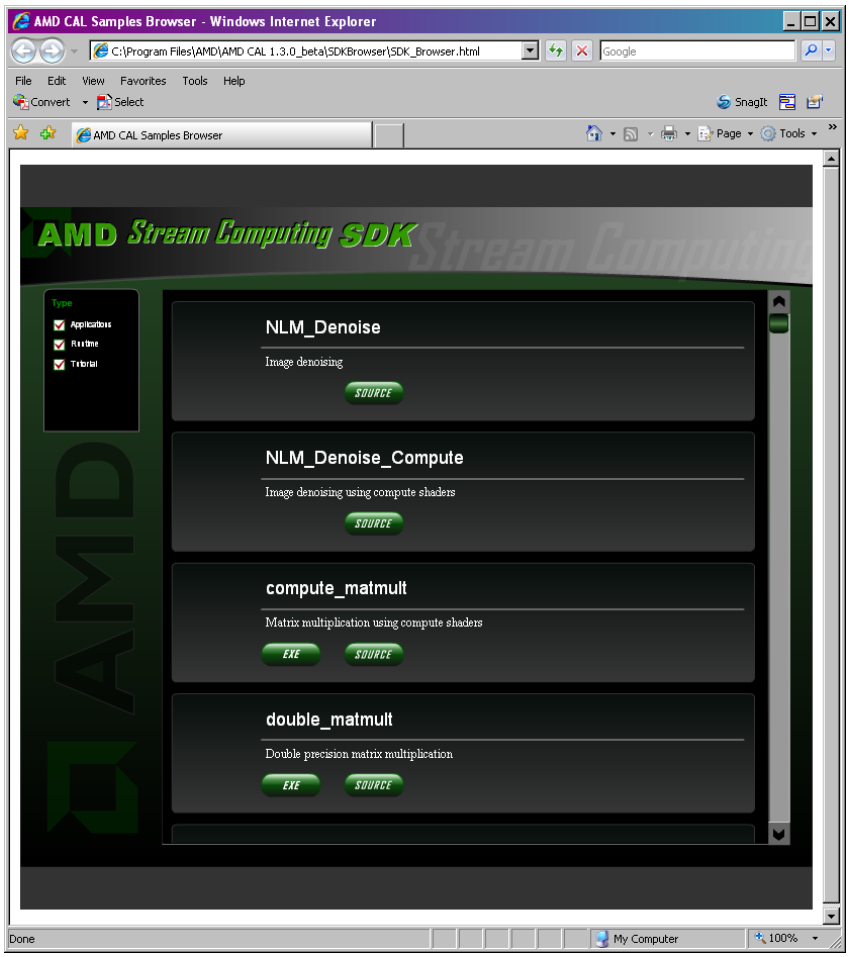


Figure 2-4: Window for browsing the AMD/ATI code samples

1. Go to Start -> All Programs -> AMD -> AMD CAL 1.3.0_beta -> AMD CAL Samples Browser to bring up the window shown in Figure 2-4.

2. Scroll to the bottom of the list and find FindNumDevices. Click on the associated EXE button. When asked if you want to run or save the file, select "Run."

3. If things are running correctly, a window will appear stating how many devices are running CAL (there should be at least 1).

2.3 Getting OpenMM Zephyr

2.3.1 Download OpenMM Zephyr

OpenMM Zephyr can be freely downloaded from https://simtk.org/home/zephyr. From the main project page, click on the "Downloads" link in the left-hand column (Figure 2-5).

You will be taken to the Downloads webpage. Click on the link for InstallZephyr-x.x.x-win32.exe (x.x.x is the version number which will change). If you are not already logged in to Simtk.org, you will be asked to log in (or register). You will then be asked to describe how you plan to use this software. Provide a description and then click "Download Now."

A dialog window will appear asking if you want to open or save this file. Click "Save" and note where you save the file! You will need this information later to run Zephyr.

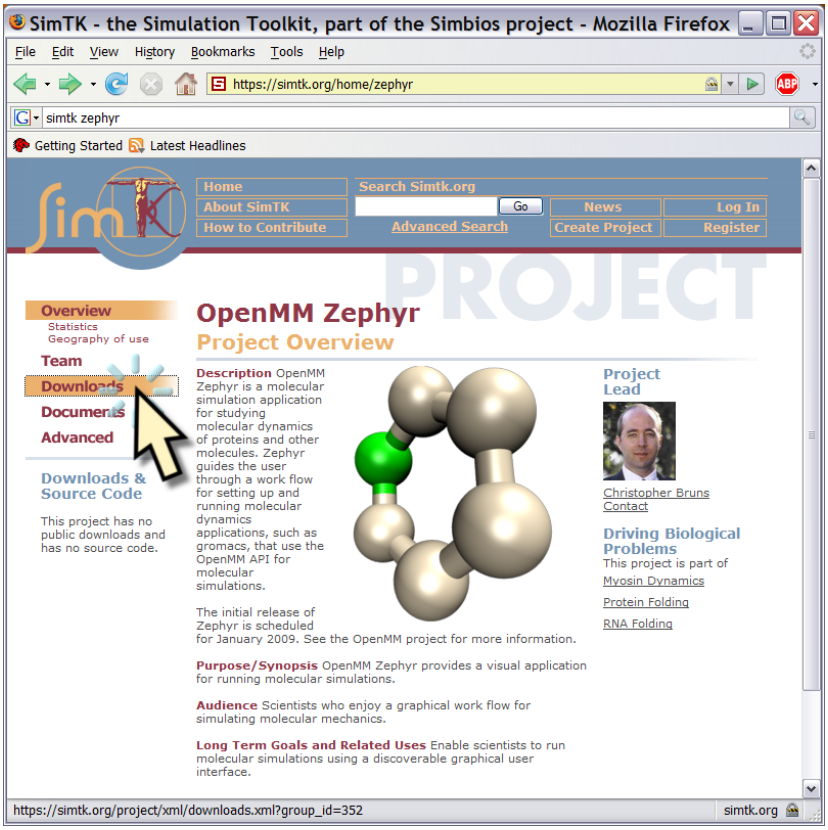


Figure 2-5: OpenMM Zephyr can be downloaded from Simtk.org

2.3.2 Install OpenMM Zephyr application

Save the InstallZephyr-x.x.x-win32.exe program to your computer and then double click to run the installer.

When you install OpenMM Zephyr, you will be asked where to save all the files. You can save the files anywhere you like on your computer. However, **the path name for the installed files must not contain any spaces.** This is a limitation of the version of GROMACS used as the basis for OpenMM Zephyr. For a Windows computer, a good place to save the files is to C:\Zephyr.

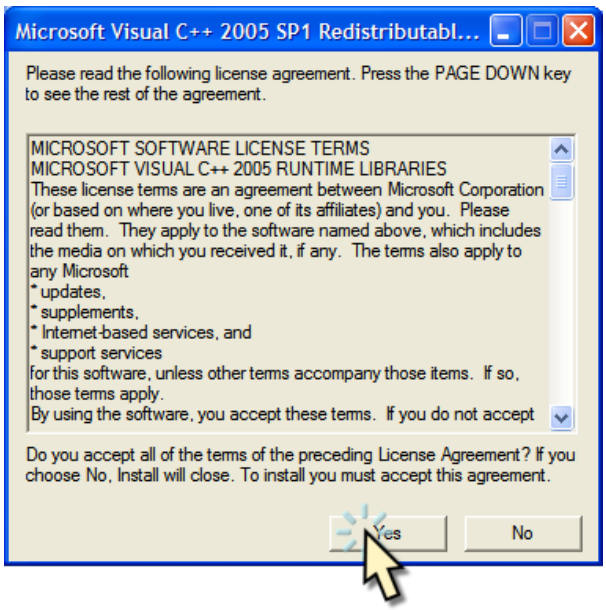


Figure 2-6: The Zephyr installer installs some required Microsoft software libraries. Click "Yes" to install these libraries.

Installing Zephyr to "My Documents" will not work. You will be able to extract the files to that location but will not be able to run OpenMM Zephyr from there.

Remember: the path for the OpenMM Zephyr files must not contain any spaces.

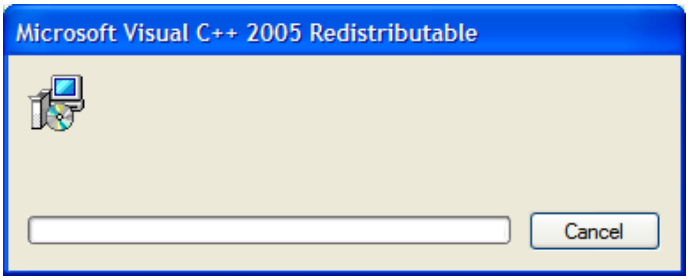


Figure 2-7: Sometimes the installation of the Microsoft libraries takes a while. Please be patient.

If you have successfully installed Zephyr, you should see the following files and directories:

- **OpenMMZephyr or OpenMMZephyr.tcl:** This is the OpenMM Zephyr application. Double-click on this to start the program.
- **doc:** This directory contains the documentation files
- **simulations:** By default, the outputs of your simulations will be saved to this directory
- **testData:** This directory contains example data files to use for starting the example simulations

You will also see three other directories (**bin**, **images**, and **lib**), which provide files used behind the scenes by the OpenMM Zephyr program.

3 Running a Simulation

3.1 Launching OpenMM Zephyr

To start OpenMM Zephyr, double-click the OpenMMZephyr.tcl file. The window shown in Figure 3-2 appears.

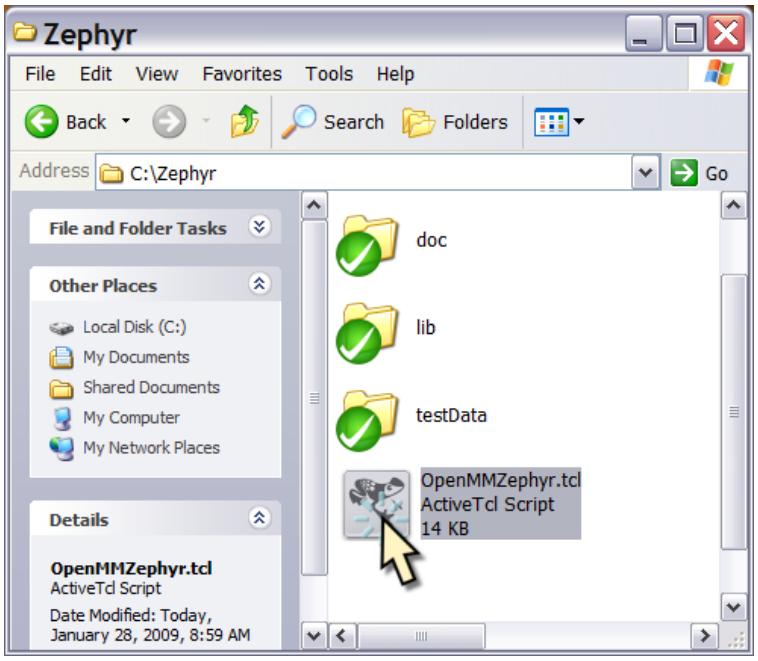


Figure 3-1: Launch Zephyr by double clicking the OpenMMZephyr.tcl icon

Note the series of tabs near the top of the window: Molecules, Parameters, (Idle) (grayed out), VMD console (grayed out), About. The order of the tabs guides you in setting up and running the simulation:

1. Choose a molecule to simulate (the Molecules tab) – See Section 3.2
2. Set the simulation parameters (the Parameters tab) – See Section 3.4
3. Run the simulation (simulation progress shown in the third tab) – See Section 3.5
4. If desired, visualize the simulation results in VMD (the VMD tab) – See Section 3.7.

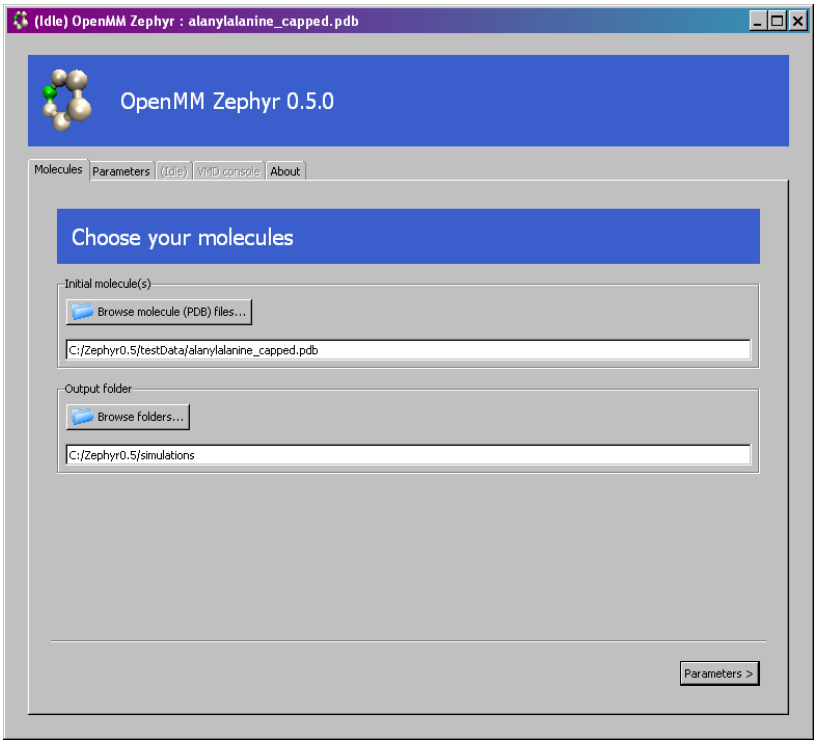


Figure 3-2: OpenMM Zephyr application

3.2 Choose a Molecule

Molecular simulations need an initial molecular structure to start the simulation. OpenMM Zephyr is able to read in structures that are in the Protein Data Bank (PDB) format. See Section 3.2.1 for more information about PDB files.

To choose the molecule you wish to simulate, click on the **Molecules** tab. Click the **Browse molecule (PDB) files...** button in the **Initial molecule(s)** panel to select the file describing the molecular structure. You can also type in the file name.

3.2.1 PDB files

The Protein Data Bank (PDB) format provides a standard representation for macromolecular structure data derived from X-ray diffraction and NMR studies. This format was created in the 1970s and includes information about the molecule, such as its name, the primary and secondary structure information, and details about how the data was collected.

3.2.1.1 Finding PDB molecular structure files

PDB files can be downloaded from http://www.rcsb.org/pdb/home/home.do, but they need to be modified to be used by OpenMM Zephyr (see Section 3.2.1.2).

3.2.1.2 Preparing a PDB file for loading into OpenMM Zephyr

In order to be compatible with the force field used in the OpenMM Zephyr simulations (GROMACS AMBER-96), you will need to edit the PDB file. Note that hydrogen atoms are automatically added to the molecular structure by Zephyr, so hydrogens do not need to be defined in the PDB file. In fact, if any hydrogen atoms are included in the PDB file, they are first removed and then added back in by Zephyr. See http://chemistry.csulb.edu/ffamber/ for complete details.

Open your PDB file using a text editor and make the following changes in residue and atom names:

For nucleic acid structures, residue names "A ", "C ", "G ", and "T " must be changed to "RA ", "RC ", RG ", and "RT " (for RNA) and to "DA ", "DC ", etc. for DNA molecules. Further, the first (most 5') residue must be named with a "5" character at the end. For example, an initial adenylate residue of RNA should have the residue name "RA5". The final (most 3') residue must have a "3" at the end of the residue name, e.g. "RG3" for a final guanylate residue of RNA.

Table 3-1: Selected PDB atom and residue name changes. See ffamber website for more details.

	standard PDB example	modified name
RNA residue	"A "	"RA "
first DNA residue	"G "	"DG5"
first protein residue	"MET "	"NMET"
final protein residue	"PHE "	"CPHE"
lysine residue	"LYS "	"LYP "
final protein residue "O" atom	" O "	" OC1"
final protein residue "OXT" atom	" OXT"	" OC2"
final protein residue	"PHE "	"CPHE"

```
ATOM    576  N   CPHE   76       5.995  -7.109   3.616  1.00  2.54           N
ATOM    577  CA  CPHE   76       6.033  -8.575   3.923  1.00  3.42           C
ATOM    578  C   CPHE   76       4.678  -9.203   3.574  1.00  4.09           C
ATOM    579  OC1 CPHE   76       3.672  -8.540   3.760  1.00  4.51           O
```

Figure 3-3: PDB file excerpt from villin.pdb. Green text shows where atom and residue names have been changed.

3.3 Select Location for Output Files

To select the location for the simulation output files, click on the **Molecules** tab. Click the **Browse folders...** button in the panel for **Output folder** to select the folder for the output files. You can also type in the name of the folder.

3.3.1 What files are saved for a simulation?

Molecular simulations produce what are called “molecular trajectories” or just “trajectories.” By definition, a trajectory is the path of a moving body through space. However, in a molecular simulation, many bodies (atoms) are moving through space, so an intuitive way to capture all the information is to save snapshots of the molecule throughout the simulation. At the end of a simulation, you have information about the molecule (e.g., coordinates of atoms, velocities, etc.) at different time points along the molecular trajectory.

For each simulation, OpenMM Zephyr saves the trajectory information in a .trr file. This is a binary format, so you cannot look at the contents by just opening the file with a text editor. These .trr trajectory files can be viewed in VMD, but only after loading a structure file (.pdb or .gro) containing the molecule.

A number of other files are also saved automatically with each simulation (See Section 3.8.1). These are used by GROMACS to run the simulation and could be used as a starting point to run simulations with GROMACS parameters not provided via the OpenMM Zephyr interface (See Chapter 5).

If you choose to watch your simulation as it runs, the simulation results will be sent to VMD, which can be used to save the trajectory information in a variety of file formats (See Section 3.8.2).

3.4 Parameters

A subset of all the parameters within GROMACS are available through OpenMM Zephyr. These can be set by clicking on the **Parameters** tab. Within the Parameters view (Figure 3-4), you can also choose the hardware on which you run your simulation (CPU or GPU) and the VMD options for visualizing the simulation as it runs.

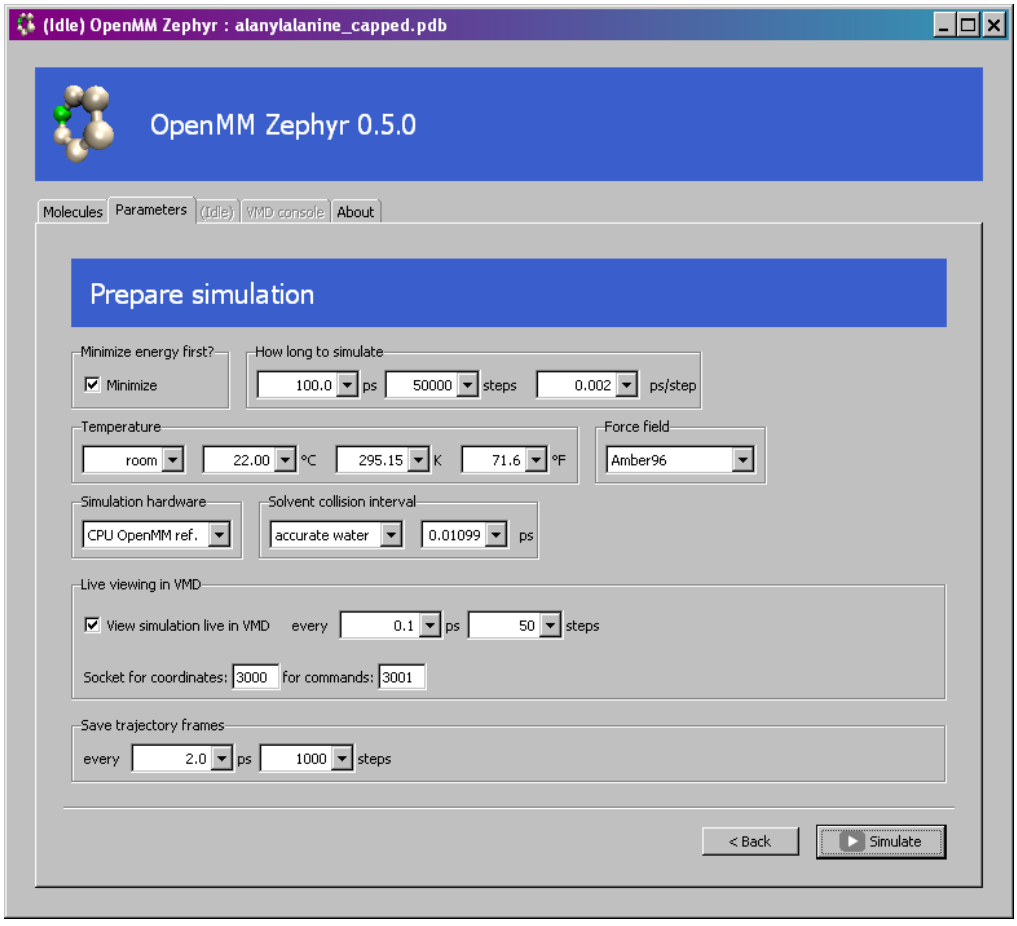


Figure 3-4: Parameters panel

3.4.1 Minimizing energy before dynamic simulation

Sometimes the starting configuration of a molecule in a PDB can have a few close atom contacts or other situation that results in a large calculated potential energy. This situation can cause the simulation to "blow up," in which case the molecular configuration becomes nonsense. It is a good idea to adjust the initial structure by energy minimization to prevent blowing up. This option can be enabled by checking the box for **Minimize** within the **Minimize energy first?** panel within the **Parameters** view.

3.4.2 Choosing the length of time of your simulation

To set the length of time to run your simulation, click on the **Parameters** tab and use the pull-down menus within the **How long to simulate** panel (located at the top of Figure 3-4). You can choose to set the length of time of your simulation (ps = picoseconds = 10^{-12} seconds) or the number of timesteps (steps). The number of timesteps, combined with the timestep interval (ps/step), determines the total simulation time.

The three pull-down menus (ps, steps, and ps/steps) are linked, so changing one will affect the other values. For example, if you have chosen 0.002 ps/step, then running a simulation for 10.0 ps is equivalent to running it for 5000 steps.

3.4.2.1 Guidelines for choosing the total simulation time

When setting up a simulation, you need to decide how long you are willing to wait for a simulation versus what interesting or useful biology can be observed within a given simulation time.

Proteins fold in microseconds to hours. A folding simulation has to calculate new velocities and positions for each atom every couple of femtoseconds. So to compute 1 microsecond (μs) of actual folding requires 500,000,000 timesteps (assuming a timestep of 2 femtoseconds). The computational time to simulate this many timesteps is staggering. Even with GPUs, simulations are able to compute less than one microsecond per day.

Although the largest values in the simulation time pulldown in Zephyr are 10000 ps (10 ns) and 100000 steps, you can create a longer simulation by typing a larger number into the box. Be aware that such simulations may take a long time, even on a GPU.

Some factors that affect the time needed to simulate one timestep (and thus the total simulation time) include:

- Molecule size: A larger molecule takes longer to simulate than a smaller one
- Hardware: GPU simulations should be faster than CPU simulations
- Frequency of updates to VMD and trajectory file: Sending coordinates less often to VMD and the output trajectory file can result in a faster simulation.

3.4.2.2 Choosing the timestep size

The timestep size needs to be smaller than the fastest phenomenon being simulated in order to accurately simulate it. For molecules, this means the timestep is on the order of a femtosecond (10^{-15} seconds or 10^{-3} picoseconds) to account for the fast bond stretch vibrations in the system. However, the smaller the timestep is, the longer the simulation will take to run. If the time step size is too large, the simulation may become unstable and the molecular structure can diverge to an unreasonable configuration. The technical term for this phenomenon is "blowing up."

3.4.3 Setting the temperature for your simulation

The **Temperature** panel within the **Parameters** view allows you to set the temperature of your simulation. You can set the temperature in either Celsius (^{o}C), Kelvin (K), or Fahrenheit (^{o}F) by typing in the value or using the pull-down menus. The left-most pull-down menu within the **Temperature** panel allows you to select from some commonly used temperatures (e.g., freezing temperature, boiling temperature, human body temperature).

3.4.4 Choosing a force field for your simulation

A force field determines how the particles within your simulation will interact. More specifically, it refers to the equations and parameters that describe the potential energy of a system of particles as a function of the particles' positions. There are many different force fields that have been developed for molecular dynamics simulations. They are empirical

models, attempts to reproduce what has been observed experimentally, and include approximations to simplify the calculations.

The current version of OpenMM Zephyr includes the AMBER-96 force field. This is a classical force field, meaning it treats each atom as a single particle and models their behavior using Newton's equation of motion (F=ma, where F=force, m=mass, a=acceleration). The forces are specified via potential energy functions that define the interactions between atoms (i.e., bonds, Van der Waals, electrostatic, etc.). These potential energy functions are empirically derived and are what distinguish one force field from another. You can read more about the AMBER force fields in (Ponder & Case, 2003) and (Case, et al, 2005).

The force field is set using the pull-down menu in the **Force field** panel within the **Parameters** view.

3.4.5 Choosing the simulation hardware

Use the pull-down menu for **Simulation hardware** within the **Parameters** view to choose what hardware to use to run your simulation:

CPU OpenMM ref.	Simulation runs on your CPU using the OpenMM reference code. Don't expect this mode to be fast. The OpenMM CPU reference code is written for expository pedagogical purposes at the expense of efficiency.
CPU gromacs	This uses standard GROMACS simulation without OpenMM. Beware that this version of GROMACS ignores implicit solvent information, so extreme caution is advised.

GPU AMD/ATI Simulation runs on your AMD/ATI GPU card. If you do not actually have an AMD/ATI GPU card, the simulation defaults to the CPU OpenMM ref. option. If you have an AMD/ATI GPU card but have not correctly installed the GPU software (see Section 2.2), the simulation will start and then immediately end.

GPU nVidia Simulation runs on your NVIDIA GPU card. If you do not actually have an NVIDIA GPU card, the simulation defaults to the CPU OpenMM ref. option. If you have an NVIDIA GPU card but have not correctly installed the GPU software (see Section 2.2), the simulation will start and then immediately end.

3.4.6 Solvent collision interval

The solvent collision interval specifies the average time it takes a solvent molecule to collide with another solvent molecule (in ps). For instance, the experimental value for water is 0.01099 ps. This number provides a measure of the solvent viscosity. A higher value means that there will be longer time gaps between solvent collisions, which means you have a less viscous solvent. A molecule in a less viscous solvent can move more easily and thus will fold faster.

To set this value, use the pull-down menus in the **Solvent collision interval** panel within the **Parameters** view. Currently, you can choose between an "accurate water" collision interval of 0.01099 ps or the "faster folding" time interval of 1.0 ps. This setting corresponds to the tau_t parameter setting in GROMACS' parameters file (see Chapter 5).

3.4.7 Live viewing in VMD

You can watch the simulation as it is running by checking the box for **View simulation live in VMD** within the **Parameters** view. Set the frequency at which VMD

is updated with the simulation output using the pull-down menu for **ps** or **steps** within the **Live viewing in VMD** panel.

If the simulation appears jerky and/or updates very slowly, try increasing the frequency at which information is sent to VMD. However, increasing the frequency (decreasing the number of timesteps or the time period) will also cause your simulation to run slower.

Viewing your simulation in VMD will slow it down. You can minimize the impact of the live viewing by increasing the time period (or the number of steps) for sending information over to VMD.

Within this panel, you can also specify the sockets to use to connect OpenMM Zephyr to VMD. One socket is used for sending coordinates from the simulations to VMD and another is used for sending commands to VMD. The default values are 3000 for the coordinates socket and 3001 for the command socket. You will need to change these values if another application (e.g., another instance of OpenMM Zephyr) is already using these to communicate with VMD.

3.4.8 Saving trajectory frames

Choose the frequency at which the trajectory frames from your simulation are saved by using the pull-down menus for **ps** or **steps** in the **Save trajectory frames** panel within the **Parameters** view. Be aware that saving frames frequently can slow down the simulation, especially with GPU accelerated simulations.

3.5 Starting a Simulation

Once you have specified the molecule you wish to simulate and chosen the simulation parameters, you can start the simulation by clicking the **Simulate** button in the lower-right corner of the **Parameters** view. OpenMM Zephyr will automatically switch to the **Status** view (third tab). See Figure 3-5. This view provides information about how much of the simulation has been completed. It also shows you the output from GROMACS, the underlying molecular dynamics simulation, in black in the **gromacs output** panel. Items in red in this panel are either commands sent from Zephyr to GROMACS or status messages.

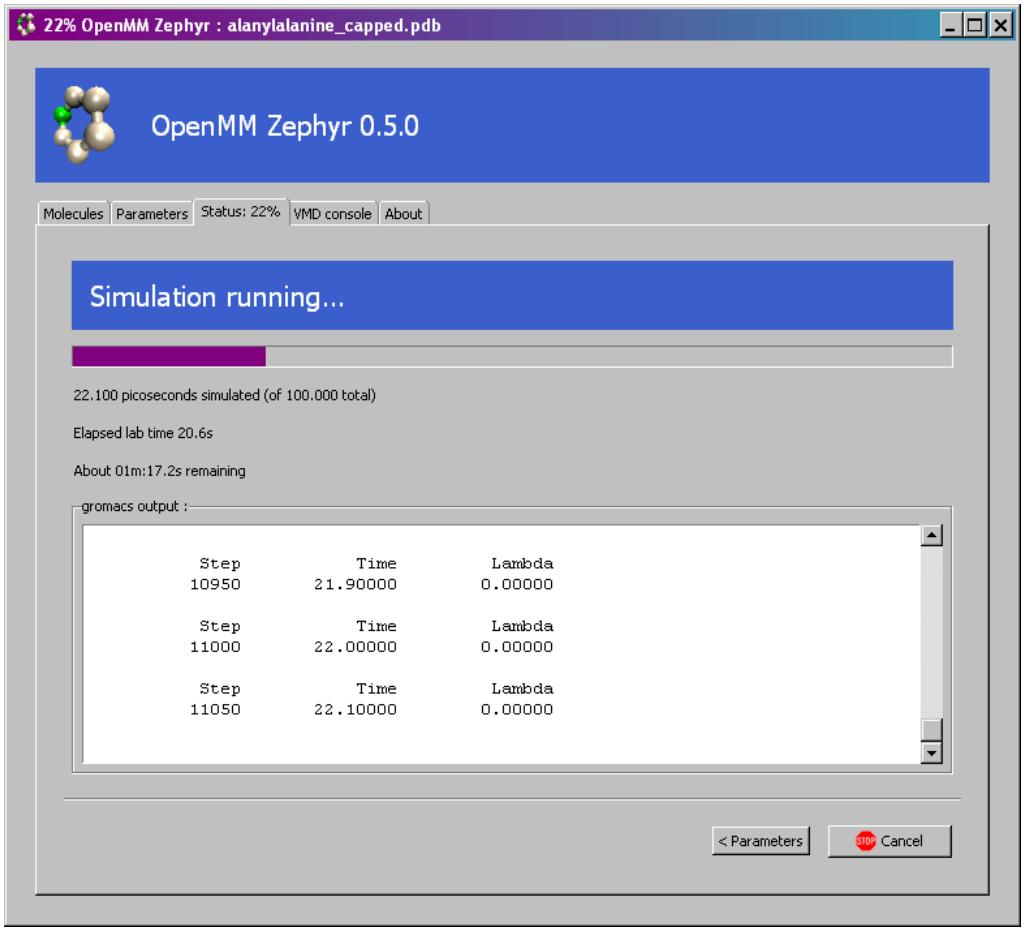


Figure 3-5: View when a simulation is running

If you chose to view the simulation as it was running (See Section 3.4.6), two VMD windows should appear, one of which shows the molecule (Figure 3-6). You may get security alert message(s), as shown in Figure 3-7. If so, just click **Unblock** to allow OpenMM Zephyr to communicate with VMD.

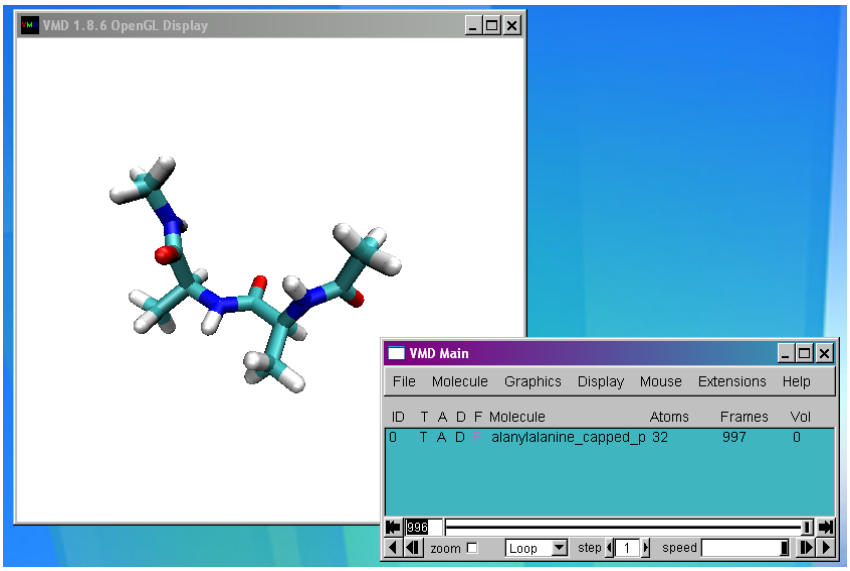


Figure 3-6: VMD can be used to watch the simulation as it runs

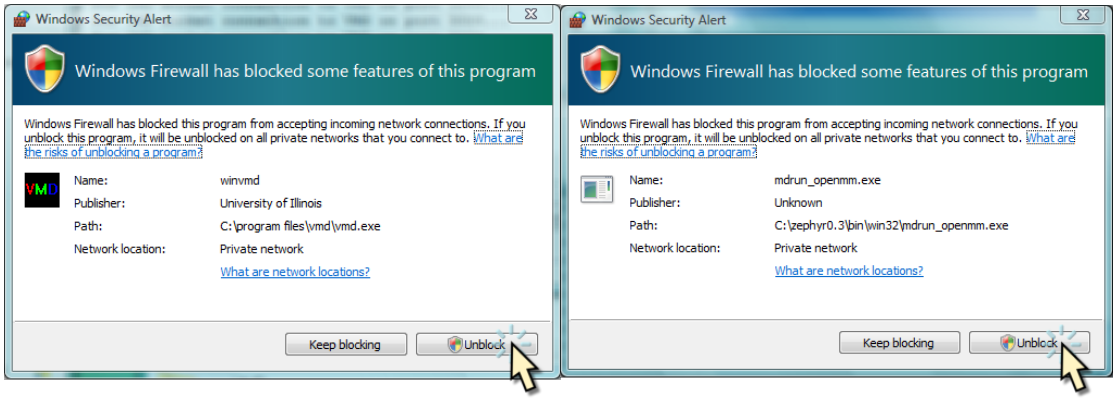


Figure 3-7: Click "Unblock" to grant permission for Zephyr and VMD to communicate through TCP sockets

3.6 Restrictions on Simulations in Zephyr

3.6.1 Amber96 Force Field

Only one force field type is available from the Zephyr interface, Amber96. Use of other force fields may be possible from the command line, and may require different conventions for the naming of atom and residue types.

3.6.2 Simulations are in implicit solvent

OpenMM Zephyr runs simulations using an implicit solvent model. This means that water molecules are not explicitly modeled (that would be the explicit solvent model). Instead, they are included in the simulation as a continuous medium.

Explicit solvent simulations are possible using the command line tools available with the original version of GROMACS; in fact, the currently released version of GROMACS (4.0) *cannot* use implicit solvent. On the other hand, the OpenMM version of GROMACS released with OpenMM Preview release 2 cannot run explicit solvent simulations.

3.7 VMD Console

The VMD console tab provides an interface between OpenMM Zephyr and VMD. VMD commands sent by OpenMM Zephyr (shown in red), as well as VMD output (shown in black) are displayed. There is also a field at the bottom (**Type VMD commands here**) to type in commands and control VMD via the command line. See the VMD User's Guide for more information: http://www.ks.uiuc.edu/Research/vmd/current/ug/.

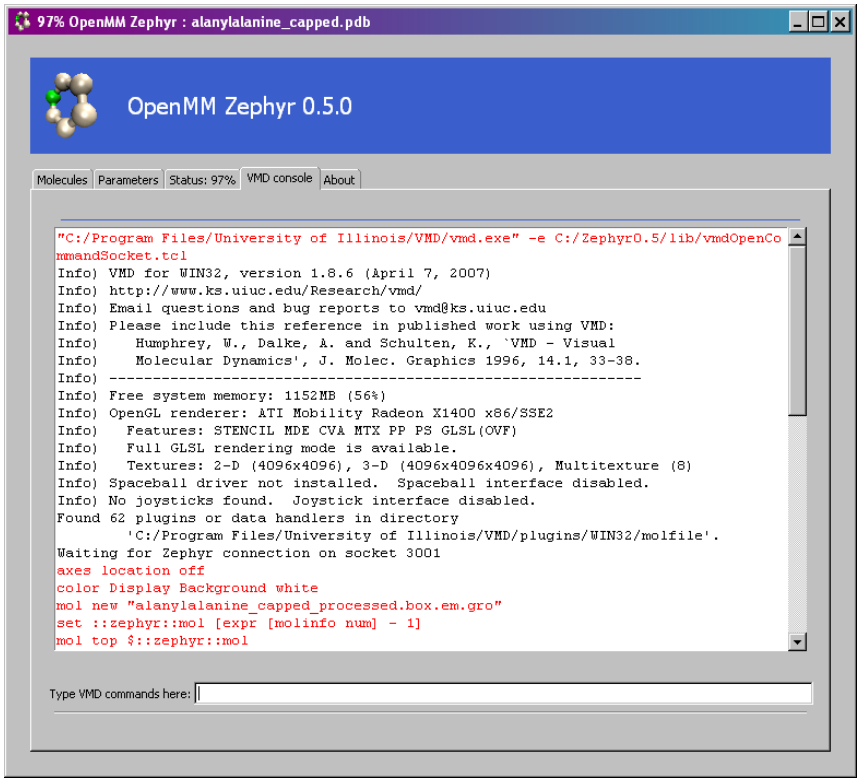


Figure 3-8: VMD console

3.8 Final Simulation Results

The trajectory produced by the simulation is saved by OpenMM Zephyr, along with other files intermediate files used by the simulation (See Section 3.8.1). The trajectory can also be saved from within VMD, if the live viewing option has been selected. (See Section 3.8.2).

3.8.1 Contents of simulation output directory

GROMACS OpenMM, upon which OpenMM Zephyr is built, produces a number of files during the simulation. These are like the standard files produced by GROMACS but they cannot be used with GROMACS, only with the OpenMM version of GROMACS. A partial list of these files is given below.

For a more complete description of the files, refer to the documentation for GROMACS 4.0 (http://www.gromacs.org/component/option,com_wrapper/Itemid,192/). The links to the Flow Chart and to Getting Started -> GROMACS files may be particularly useful.

md.out.(mdp) This is an MDP file (the mdp extension may or may not appear, depending on your browser settings). It is the input parameters file for the simulation, containing information such as how long to run the simulation, the timestep, and the temperature for the simulation.

traj.trr This file contains the trajectory of the simulation. It will include the coordinates, and velocities. This is in a binary format, so it cannot be read using a regular text editor but it can be viewed using VMD.

md(.log) This text file is the GROMACS OpenMM log file for the simulation. The information is what appears in the **gromacs output** panel of the **Status** view.

.tpr file This file contains the starting structure of your simulation, the molecular topology, and all the simulation parameters. It is generated by several pre-processing steps and is in a binary format, so it cannot be read using a regular text editor.

3.8.1.1 Cleaning up when the output directory gets too full

After you have run a very large number of simulations in the same directory, your simulations may stop working. Try removing backup files in the directory, i.e., those with names beginning with a '#' character. This problem is NOT the same as the benign "unable to rename checkpoint file" message that you might see.

3.8.2 Saving a trajectory from VMD

Once your trajectory is loaded into VMD, you can save trajectory and structure files in many file formats from within VMD. These and many other rich features of the VMD program are documented in the current version of the VMD User's Guide, available at http://www.ks.uiuc.edu/Research/vmd/current/ug/.

3.9 Stopping a Simulation

Press the **Cancel** button in the **Status** panel to halt a running simulation. (This works correctly in Zephyr 0.3 and later. For earlier versions, you must also kill the mdrun processes with the task manager).

3.10 Modifying Parameters Not Shown in the OpenMM Zephyr GUI (advanced)

OpenMM Zephyr uses a modified version of GROMACS (GROMACS OpenMM) to run molecular simulations. Power users who are familiar with GROMACS may be able to run a wider variety of simulations using the command line. See Chapter 5 for a brief tutorial on how to run GROMACS OpenMM from the command line. The GROMACS user documentation at http://www.gromacs.org/ may also be helpful.

4 Troubleshooting

4.1 Why is my simulation updating in VMD so slowly?

If the VMD view of the simulation is changing very slowly, try decreasing the interval for VMD updates, and restarting the simulation. This will result in a smoother display in the VMD window.

4.2 I am getting the message "Waiting for IMD socket connection to VMD on port 3000"

Click to the VMD console view to see what is happening with VMD. Depending on the message you see, you will need to take different actions:

VMD console message/issue	**Action**
couldn't open socket: address already in use	Cancel the simulation by clicking the **Cancel** button in the **Status** view. Close the VMD windows that appeared. Click the **Parameters** tab. Change the values of **Socket for coordinates** and **for commands** (See Section 3.4.6)
VMD appears to have hung	Scroll up through the list of VMD commands and outputs, and locate the command "imd connect..." Copy and paste the command in quotes in the field for VMD commands.

4.3 The “Simulation complete” message appears but the “Elapsed lab time” does not stop

The simulation is completed when the message appears. However, there are other tasks, such as writing out the files, that are being performed and are included in the calculation of the **Elapsed lab time**. So the **Elapsed lab time** will stop when all tasks are done.

5 Going Further: Running GROMACS OpenMM from the Command Line

5.1 Generating Initial Parameter and Simulation Input Files

You can use OpenMM Zephyr to generate initial parameter files for your simulation. You can modify these files to run simulations using parameters that are not accessible through the OpenMM Zephyr GUI.

1) Locate your Zephyr install, typically C:\ZephyrZ.Z, where Z.Z is the current release number.
2) Create a new folder in C:\ZephyrZ.Z named “testRun”
3) Launch Zephyr
    a) Select the **Initial molecule** of your choice
    b) Set **Output folder** to C:\ZephyrZ.Z\testRun
    c) Click on **Parameters**. You will run Zephyr to get initial parameter and simulation input files and to minimize the energy of your structure before the simulation. Suggested initial parameters are as follows: (see Figure 5.1 below)
        a. Set the length of the simulation time to 0.1 ps.
        b. Make sure **Minimize energy first?** box is checked.
        c. Uncheck **View simulation live in VMD**
        d. Select the **Simulate** button

The run should finish very quickly.

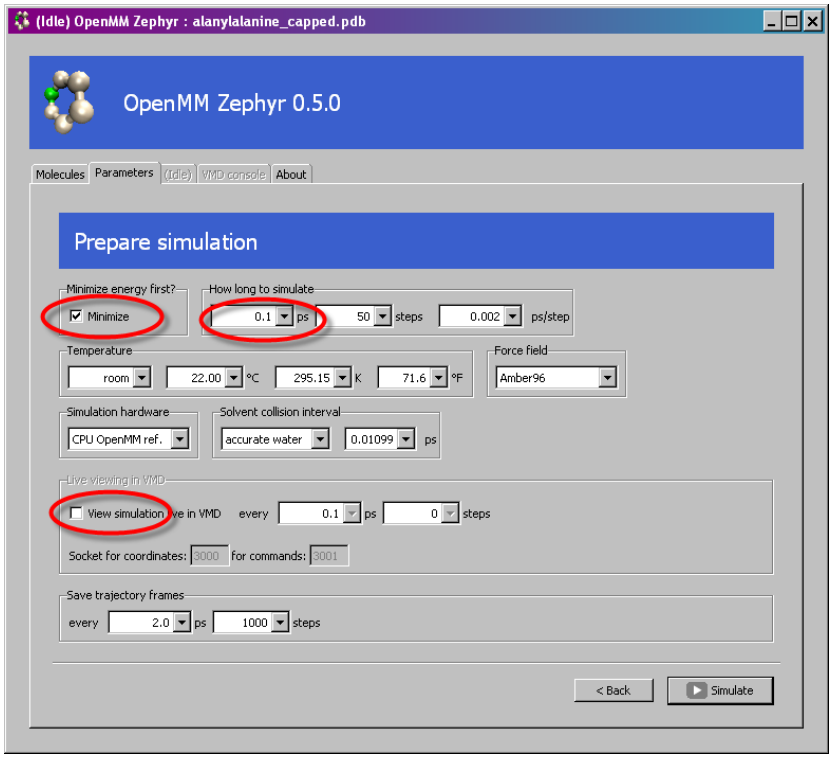
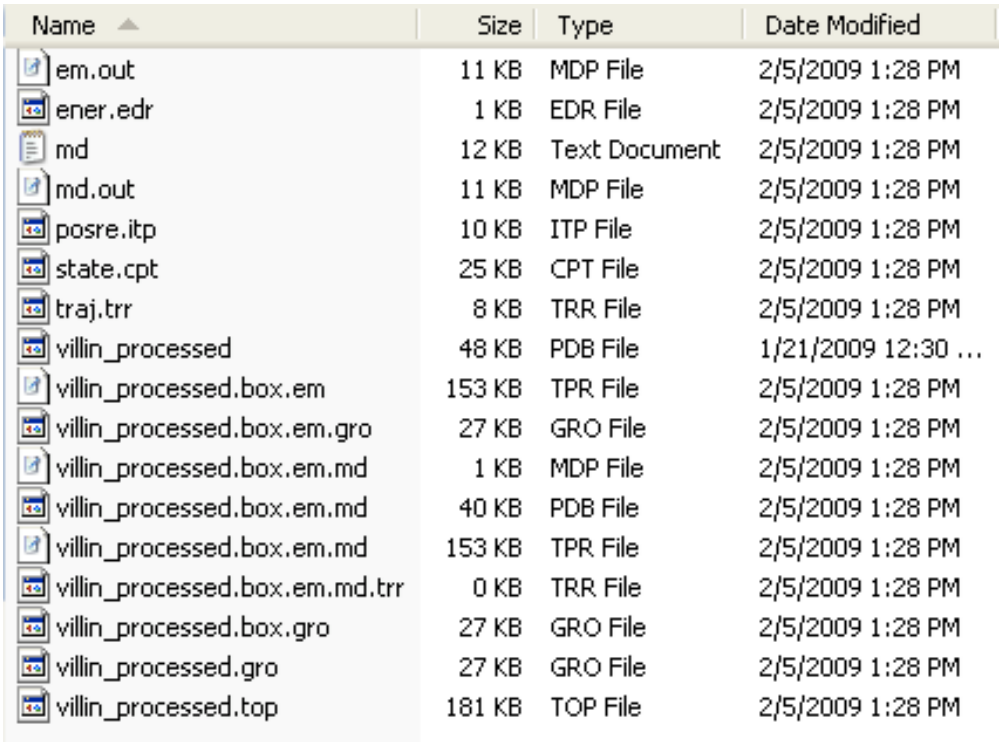


Figure 5-1: Suggested parameter settings to generate initial files to run a GROMACS simulation

5.2 Preparing for a Command Line mdrun_openmm

mdrun_openmm is the command to actually start the simulation. There are a few more steps that you need to take before you are ready to start the simulation.

1) Verify the content of your C:\ZephyrZ.Z\testRun folder. Zephyr should have added the files shown below in Figure 5-2 (files starting with # were deleted). Note the molecule name in the example below was “villin.”



Name	Size	Type	Date Modified
em.out	11 KB	MDP File	2/5/2009 1:28 PM
ener.edr	1 KB	EDR File	2/5/2009 1:28 PM
md	12 KB	Text Document	2/5/2009 1:28 PM
md.out	11 KB	MDP File	2/5/2009 1:28 PM
posre.itp	10 KB	ITP File	2/5/2009 1:28 PM
state.cpt	25 KB	CPT File	2/5/2009 1:28 PM
traj.trr	8 KB	TRR File	2/5/2009 1:28 PM
villin_processed	48 KB	PDB File	1/21/2009 12:30 ...
villin_processed.box.em	153 KB	TPR File	2/5/2009 1:28 PM
villin_processed.box.em.gro	27 KB	GRO File	2/5/2009 1:28 PM
villin_processed.box.em.md	1 KB	MDP File	2/5/2009 1:28 PM
villin_processed.box.em.md	40 KB	PDB File	2/5/2009 1:28 PM
villin_processed.box.em.md	153 KB	TPR File	2/5/2009 1:28 PM
villin_processed.box.em.md.trr	0 KB	TRR File	2/5/2009 1:28 PM
villin_processed.box.gro	27 KB	GRO File	2/5/2009 1:28 PM
villin_processed.gro	27 KB	GRO File	2/5/2009 1:28 PM
villin_processed.top	181 KB	TOP File	2/5/2009 1:28 PM

Figure 5-2: Content of testRun directory after initial Zephyr run (files starting with # were omitted). Note the molecule name in this example was "villin."

2) Locate the following files:
   a. md.out of type MDP → this is the parameter file
   b. villin_processed.box.em.gro [with "villin" replaced by your molecule name]
   c. villin_processed.top" [with "villin" replaced by your molecule name]

3) Open a command prompt by selecting
   Start → All Programs → Accessories → Command Prompt.

4) Type **cd C:\ZephyrZ.Z\testRun** in the command prompt window to change to the directory where your output files are located (change Z.Z to the release number).

5) Open the md.out file using NotePad or another text editor and change the parameters. At a minimum, you want to set **nsteps** to at least 10000 to get a trajectory (.trr file) with a few frames. For a full list of parameters please consult http://www.gromacs.org/ and search for "mdp options". Note that not all of these have been implemented within GROMACS OpenMM (See Appendix A).

6) Set environment and PATH variables so that the program knows where to find the programs files it needs. See[1] for setting these permanently. See Figure 5-3 for copy/paste instructions into the Command Prompt window.

   At the command prompt, enter (remember to substitute Z.Z):

   ```
   set GMXLIB=c:\ZephyrZ.Z\bin\gmxlib
   PATH=c:\ZephyrZ.Z\bin\win32;%PATH%
   ```

   If you are executing on an **AMD/ATI GPU**, add:

   ```
   set OPENMM_PLUGIN_DIR=c:\ZephyrZ.Z\bin\win32\lib\openmm\ati
   ```

   If you are executing on an **NVIDIA GPU**, add:

   ```
   set OPENMM_PLUGIN_DIR=c:\ZephyrZ.Z\bin\win32\lib\openmm\nvidia
   ```

   Type `set` (to verify that GMXLIB and `OPENMM_PLUGIN_DIR` are defined correctly)
   Type `path` (to verify that your path contains `c:\ZephyrZ.Z\bin\win32`)

7) Run grompp. This is the GROMACS preprocessor. For more information, consult http://www.gromacs.org/ and search for grompp.

   - Paste the command below (after replacing "villin" with your molecule name) in your command prompt window (see also SUGGESTION below):

     ```
     grompp -f md.out.mdp -c villin_processed.box.em.gro -p
     villin_processed.top -o villin_processed.box.em.md.tpr -po
     md1.out.mdp
     ```

SUGGESTION: copy the commands into a text file. Replace "villin" with your molecule name. Copy and paste into the command prompt window (see Figure 5-3 for copy/paste commands). Note: You can also copy the above command into a ".bat" file (say gromppRun.bat) and simply type gromppRun.bat into the command prompt window.

---

[1] To permanently set your environment variables:
Go to "Start → Control Panel → System → Advanced → Environment variables"
Under "User Variables for [user]" select "New" then
Set Variable Name: `GMXLIB`
Set Variable Value: `C:\ZephyrZ.Z\bin\gmxlib` -- where Z.Z is the Zephyr release.
Repeat for the `OPENMM_PLUGIN_DIR` variable name with the appropriate value and similarly edit PATH.

8) You are now ready to run mdrun_openmm and start your simulation.
   - Paste the command below into the command prompt window (after replacing "villin" with your molecule name)

```
mdrun_openmm -s villin_processed.box.em.md.tpr -o
villin_processed.box.em.md.trr -c villin_processed.box.em.md.pdb
```

Note that "`villin_processed.box.em.md`" is repeated many times in the above command. To avoid re-typing this "prefix" GROMACS allows you to execute the shorter equivalent command:

```
mdrun_openmm -deffnm villin_processed.box.em.md
```

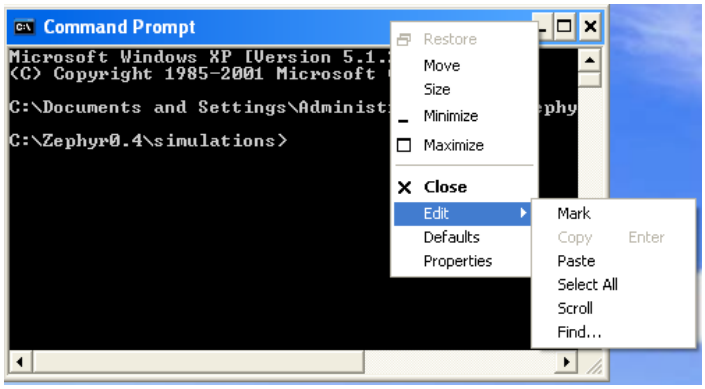


Figure 5-3: To paste commands into your Command Prompt window you need to right-mouse click the top blue bar and select Edit→ Paste.

5.3 Visualizing Your Trajectory in VMD

After the completion of mdrun_openmm, you can visualize the resulting trajectory in VMD using the `".trr" and ".gro" files as follows.`

1) Start VMD. **Start -> All Programs -> University of Illinois -> VMD -> VMD X.X.X** (where X.X.X. refers to the version number)
2) In the VMD Main window, select **File → New Molecule**
    a. Enter **Filename**.
    Browse to C:/ZephyrZ.Z/testRun/villin_processed.box.em.md.trr, where "villin" is replaced by your molecule name and Z.Z. by the Zephyr release number.
    b. Hit **Load; do not close the window**
    c. Enter **Filename**.
    Browse to C:/ZephyrZ.Z/testRun/villin_processed.box.em.gro, where "villin" is replaced by your molecule name and Z.Z by the release name.
    d. Hit **Load**

You can now "play back" your trajectory.

See http://www.ks.uiuc.edu/Research/vmd/current/docs.html#tutorials for a complete set of options in VMD.

6 Giving Back: Submitting Bug Reports and Feature Requests

Feedback about OpenMM Zephyr is very much appreciated. Please submit bug reports and feature requests on-line at http://simtk.org/home/zephyr. Click on Advanced -> Features & Bugs -> Submit New.

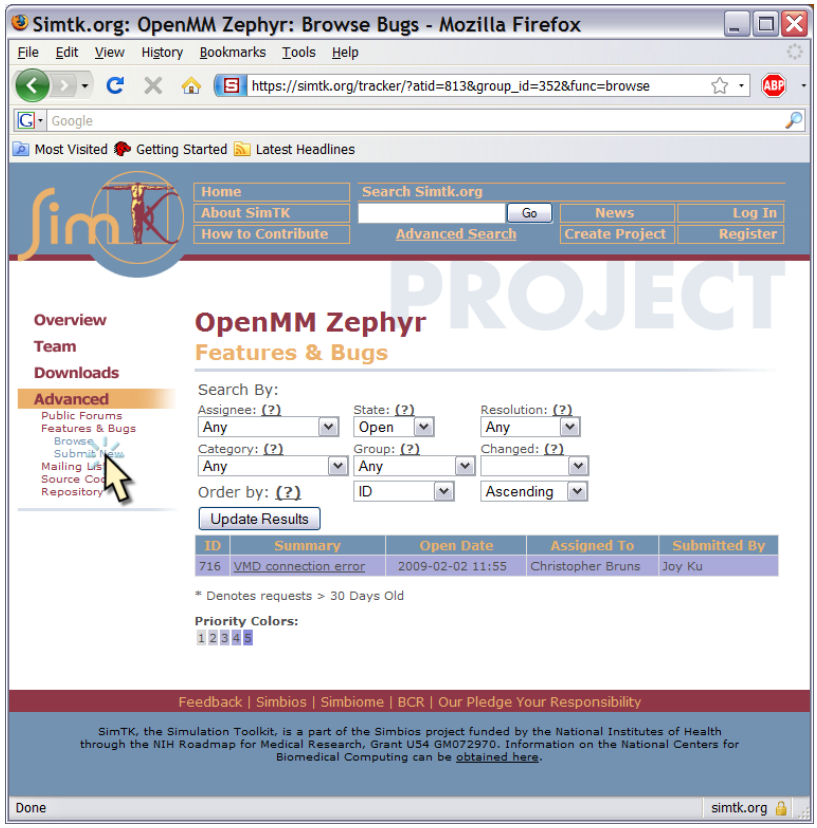


Figure 6-1: Investigate, submit, or track feature requests and bug reports at Simtk.org

7 Appendix A: GROMACS-OpenMM Limitations

Only a fraction of the features of Gromacs 4 are supported in this release of Gromacs-OpenMM. Here are some of the most significant limitations to the allowed options in the MDP file.

The only supported integrators are "md", "sd", and "bd". The "bd" integrator is not supported on ATI GPUs.

The only supported settings for the "constraints" options are "none" or "hbonds".

Neither cutoffs nor periodic boundary conditions are supported. The "coulombtype" and "vdwtype" settings are ignored, as are the various cutoff settings.

The only supported value for the "tcoupl" and "pcoupl" settings is "no". Temperature coupling can be achieved by using the "sd" or "bd" integrator. Pressure coupling is not a meaningful concept without periodic boundary conditions.

Rigid water molecules are not supported.

As a result of these limitations, explicit solvent cannot usefully be simulated with this version of Gromacs-OpenMM. Implicit solvent can be added to a simulation by including the following three lines in the MDP file.

```
implicit_solvent = GBSA
gb_algorithm = OBC
gb_epsilon_solvent = 78.3
```

8 References

D.A. Case, T.E. Cheatham, III, T. Darden, H. Gohlke, R. Luo, K.M. Merz, Jr., A. Onufriev, C. Simmerling, B. Wang and R. Woods. The Amber biomolecular simulation programs. *J. Computat. Chem*. **26,** 1668-1688 (2005).

Friedrichs, M., Eastman, P., Vaidyanathan, V., Houston, M., Legrand, S., Beberg, A., et al. (2009). Accelerating Molecular Dynamic Simulation on Graphics Processing Units. *Journal of Computational Chemistry* , in press.

Glazer, D., Radmer, R.J., & Altman, R. (2008). Combining molecular dynamics and machine learning to improve protein function recognition. *Pac Symp Biocomput* , 332-343.

Herce, H., & Garcia, A. (2007). Molecular dynamics simulations suggest a mechanism for translocation of the HIV-1 TAT peptide across lipid membranes. *PNAS , 104*, 20805-20810.

Hess, B., Kutzner, C., van der Spoel, D., & Lindahl, E. (2008). GROMACS 4: Algorithms for Highly Efficient, Load-Balanced, and Scalable Molecular Simulation. *J. Chem. Theory Comput. , 4*, 435-447.

Humphrey, W., Dalke, A., & Schulten, K. (1996). VMD - Visual Molecular Dynamics. *J. Molec. Graphics* , 33-38.

Kasson, P., & Pande, V. (2008). Structural basis for influence of viral glycans on ligand binding by influenza hemagglutinin. *Biophysical Journal , 95*, L48-L50.

Marszalek, P., & al., e. (1999). Mechanical unfolding intermediates in titin molecules. *Nature , 402* (6757), 100-103.

Ponder, J.W. and Case, D.A. (2003). Force fields for protein simulations. *Adv. Prot. Chem., 66*, 27-85.