BlueMatter Problem Setup User Manual

Michael C. Pitman

February 1, 2001

Contents

0.1 Problem Specification

This section details the steps needed to prepare and execute a session with BlueMatter. It is assumed that the proper installation of BlueMatter has been achieved and the relevant environment variables are set (*see installation*).

0.1.1 Introduction

Three files are required to define and execute a BlueMatter session. The **Molecular System Definition** file, or *.msd* file contains the topology of the molecular system, and all parameters required for the evaluation of each force term. The **Dynamic Variable Specification** file, or *.dvs* file specifies the initial coordinates, velocities, and any other dynamical variables that are required at startup. The **Runtime Paramters** file, or *.rtp* file The following sections detail how to prepare each of these files for the particular system of interest.

0.1.2 Preparing the Molecular System Definition

For BlueMatter Version 1.0, the Molecular System Definition file may be prepared from either a CHARMM *.psf* file or a OPLSAA *?* file. In future versions, AMBER files will be supported. For each of these origins for system preparation, the initial target is an internal repesentation that is independant of the force field from which it came. For browsing purposes, an XML version of the representation is created. However, tools and style sheets for viewing will be available in future releases. From the XML representation of the system is processed into an .hpp file, so that the data structures can be compiled into the .msd file. The stages of transformation from a CHARMM or OPLSAA file to a .msd therefore are as follows:

- CHARMM or OPLSAA representation
- XML representation
- .hpp representation
- .msd representation

The following is an example prepares enkp.msd from enkp.psf using the CHARMM22 parameter files.

psf2xml top_all22_prot.inp par_all22_prot.inp enkp.psf XML > enkp.xml

Note that the XML file captured by redirecting stdout to the desired filename. For OPLSAA files, a perl script is used to create the XML file. The script invokes IMPACT from the pdb file and collects all parameters and structural information

to prepare the XML file directly. The following example converts the enkp.pdb file to the XML file.

opls2xml.pl enkp.pdb

This generates enkp.xml.

In preparing the header file, we may choose a fragmentation scheme. The fragmentation scheme may be specified by supplying a file or by selecting the DEFAULT scheme. For peptides, the DEFAULT fragmentation scheme is to cleave at every residue. Water is also placed in its own fragment. The format for fragmentation is given in Section **??**. Next we will prepare the header file with the default fragmentation scheme:

xml2hpp DEFAULT < enkp.xml > enkp.hpp

If we had a specific fragmentation scheme that was specified in file **frag_scheme.frg**, we would prepare the hpp file as follows:

xml2hpp frag_scheme.frg < enkp.xml > enkp.hpp

Up until this point, the definition of the molecular system is machine independent. This next and final stage of .msd prepartion is a machine dependant step. It requires execution on the machine which will run the session. To generate the .msd file from an .hpp file , for example enkp.msd:

hpp2msd enkp

This final step produces the desired .msd file for the particular machine the will run the simulation.

0.1.3 Preparing the initial conditions

The intial conditions will generally be the starting positions and velocities of each site. In fluxating charge models, the initial charges will also be included. In release 1.0, only static charge models are supported. The **.dvs** file is the file containing the positions and velocities. These may be obtained from an external setup or a snapshot file. It is important to note that this file is force field independant. It only contains positions and velocities. However, due to atom reshuffling in CHARMM psf files from their pdb file origin, CHARMM files must be processed separately from OPLSAA files (or any other source that preserves atom ordering). The method of preparation of a **.dvs** file from a **.pdb** file is as follows:

pdb2dvs < *pdbfile* > *dvsfile*

NOTE: CHARMM reshuffles the atom order in translating a pdb to a psf file. The crd file, however, has a consistent order as the psf file. For msd files originating from CHARMM, the dvs file must be derived from the CHARMM crd file with the command below.

crd2dvs < *crdfile* > *dvsfile*

0.1.4 Setting the runtime parameters

The runtime parameters are set in the **.rtp** file. Currently, three such parameters are supported. More will of course be added as new features are implemented.

Parameter Name	Meaning
NumberOfOuterTimeSteps	The number of RESPA outer time steps to execute
OuterTimeStepInPicoSeconds	Lowest frequency time step in picoseconds
InnerTimeStepInPicoSeconds	Highest frequency time step in picoseconds
DefineRespaLevel n m	Defines Respa level n to execute after every m steps of level n € 1
BondRespaLevel	Respa level for bonds
AngleRespaLevel	Respa level for angels
TorsionRespaLevel	Respa level for torsions
InterMolecularRespaLevel	Respa level for nonbonded interactions
BoundingBox	Periodic boundary (lower corner, upper corner)