Implementing Workflow for Biophysical Simulations in openDIEL

You Han(City University of Hong Kong) Mentors: Kwai Wong(UTK), John Ossyra(ORNL), Xiaolin Cheng(ORNL)

Background

About BD simulation

Brownian Dynamics simulations (BDS) are a method in computational biophysics used to study protein diffusion processes such as association mechanisms, measuring binding rates.

The biomolecules are regarded as rigid bodies; for each molecule, the motion is computed based on precalculated intermolecular physical potentials

About BD simulation(cont'd)



Fig. 1 shows the organizational structure for preliminary molecular structure computations and their downstream use in Brownian Dynamics Simulation (BDS).

About openDIEL

The **openDIEL**(open Distributive Interoperable Executive Library) is workflow engine that aims to facilitate communication between loosely coupled simulations in large-scale parallel computing. (openDIEL is developed by UT CFD LAB.)

About openDIEL(cont'd)



Fig.2, simple mechanism of openDIEL

Abstract/Objectives

- First stage: Integrate with BDS toolkit from Mr. John Ossyra
- Second stage: build GUI with which users can configure and run BDS in openDIEL
- Third stage: Expand the GUI for parallel computing, insert modules of commonly used biophysical packages such as NAMD, Gromacs into the workflow and build physical coupling between simulations.

GUI V1-configure file & input files

GUI V1 provides a graphical user interface to interact with users, then generate the required input files for BD simulation and a configure file for openDIEL, according to user's definitions.

- Configure file for openDIEL
- Input files for Running BD simulation

GUI V1-configure file & input files(cont'd



Fig.3, the user interface of GUI V1

GUI V1-configure file & input files(cont'd

About configure file:

size of tuple space / executable path / running size for each module, execution order / dependencies of each

```
aroun / etc
 workflow.cfg
                                                        modules=(
  shared_bc_sizes = [];
                                                                 function="ielTupleServer";
    tuple_space_size = 1;
                                                                 args=();
3
                                                                 libtype="static";
                                                                 library="libIELexec.a";
 group8:
                                                                 size=1:
                                                             }
{
     order=("MODULE-11")
     iterations=1
                                                                 function="MODULE-0";
                                                                 args=("../executables/s1a.sh");
 group9:
                                                                 libtype="static";
                                                                 splitdir="pdb2pgr"
     order=("MODULE-12")
                                                                 size=1
     iterations=1
     depends=("group1","group2","group3","group4","group5",
                                                                 points=(
     oup6", "group7", "group8")
                                                                     (())
                                                                 );
```

Fig.4, details of the configure file

GUI V1-configure file & input files(cont'd

About input files for BD simulation:

0.0	0				apbs.2mol.in.tpt	t				
4 ►	apbs.2mol.in.tpt	×								v Qv (a) Sign in
1	read	0.0	0				apl	os.in.tpl		8
2	mol (0)	4 ۲	apbs.in.tpt ×							T Stars
4	mol (0)		read 💿 💿	0					ecm.	in.tpt
5	end		mol {fn_mc	areas in test						
67	elec name prote		elec	ecman.tpt	<u> </u>					
8	dime (0)		{calctype] 1	(0), pdb	000					🗋 ed.in.tpt
9	grid {0}		dime {x_c 3	te	≤ ► ed.in.tpt					
10	ion charge		gcent mol 4	{0}.tcha	1 #		-		b addam addam a	Direction and
12	ion charge		ion charge 6	{0}.bin.grd	2 {vxl_spc}, {		-			ha.in.tpt
13	MOLECULE	10	100 charge 7	pr	4 {ion conc} {	4 Þ	hd.in.tpt			
15	bcfl {0}	12	{equation]	{1}, {2}	5	1		-> -		h, ndimx, ndimy, ndimz
16	pdie (0)	13	bofl {bcf 10	{3}, {4}	6 {fn_mol}.pdt	2	{vx1_sp	c), ()	<_dim}, {y_dim},	{z_d1m} a.b.factor
17	sdie (0) srfm (0)	15	sdie (slv 11	ef	8 {fn_mol}.ed.		{a}, {b), {fi	ictor}	
19	chgm {0}		srfm (sur 13	(0).ecna	9 0		#	1 adb		pfile
20	sdens (0)	17	chgn {cha 14	{5}	10		*			efile, iform
22	swin {0}	19	srad (sur				{fn_mo]	0.0	0	hydropro.dat.tpt
23	temp (0)	20	swin {sur			9 10	•	4 Þ	hydrooro.dat.tot	×
24	calcenergy	22	calcenergy						Ita mall	i Name of molecule
26	end	23	calcforce						HPro	! Name for output file
27	elec name compl	24	write pot						{fn_mol}.pdb	! Strucutural (PBD) file
28	dime (0)	26	print elecEner					4	(mode) (aer).	! Type of calculation ! AFR. radius of primary elements
30	grid (0)	27	quit						6,	! NSIG
31	gcent (0)	28							(sigmin),	! Minimum radius of beads in the shell (SIGMIN)
33	ion charge							8	<pre>(sigmax), {temperature}.</pre>	! Maximum radius of beads in the shell (SIGMAX) ! T (temperature, centigrade)
34	MOLECULE							10	0.0091,	! ETA (Viscosity of the solvent in poises)
35	EQUATION							11	(molar_mass)	! RM (Molecular weight)
37	pdie (0)							13	1.0.	! Solvent density, g/cm3
38	sdie (8)							14	-1	! Number of values of Q
une L	Column 1							15	-1	! Number of intervals
1 m	d Information		6.1					17	1	! IDIF=1 (yes) for full diffusion tensors
+	gradery	time 11	Coumin 10					18	•	! End of file
•	Global Awareness	5	(B) 88 (B)					19		

Fig.5, the input files to print for running BD simulation

GUI V2-Parallel Version(in process)

GUI V1 focuses mainly on running BDS and configuring the precalculation modules to optimize models in the simulation, requiring iteration and other small changes within the workflowso GUI V2 for running biophysical simulations in parallel is on the way.



Acknowledgements

This project is supported by University of Tennessee, NSF, and Oak Ridge National Laboratory.

