GROMACS - Feature #742

Enhancing the performance of the free energy code

04/26/2011 09:40 PM - Luca Bellucci

Status:	New	
Priority:	Normal	
Assignee:		
Category:		
Target version:	future	
Difficulty:	uncategorized	

Description

Dear all,
free energy perturbation (FEP) module of GROMACS 4.5 can be used to perform
Hamiltonian replica exchange molecular dynamics (H-REMD). The main modules that "drive"
H-REMD simulation are src/kernel/repl_ex.c and /src/gmxlib/nonbonded/ nb_free_energy.c
I tried to perform H-REMD decoupling a small peptide in
water. In this case I observed about a 40-50% reduction in speed.
When I tried to decouple water and peptide the performance loss is much more dramatic.
I posted my test files in:
https://www.dropbox.com/link/17.-sUcJyMeEL?k=0f3b6fa098389405e7e15c886dcc83c1
This is a run for a dialanine peptide in a water box.
The cell side cubic box was 40 A.
The directory is organized as :
TEST\
topol.top
Run-00/confout.gro ; Equilibrated structure
Run-00/state.cp
MD-std/Commands ; commands to run the simulation , grompp and mdrun
MD-std/md.mdp
MD-FEP/Commands
MD-FEP/md.mdp
For more details in this issue see also gmx-users mailing list:
http://lists.gromacs.org/pipermail/gmx-users/2011-April/060169.html

FEP code is bug free, but a this stage it is unusable to perform H-REMD
due to the low performance.

Free Energy Perturbation with Replica Exchange Molecular Dynamics offers
a powerful strategy to enhance the sampling of biological system, therefore
I would suggest to improve its efficiency.
Best regards
Luca

Subtasks:

Feature # 1665: improve free energy non-bonded kernel performance	**New**
Task # 2875: SIMD version of the free-energy kernel	**Accepted**

Related issues:

Related to GROMACS - Bug #2014: GROMACS free energy memory footprint	**Closed**
Related to GROMACS - Feature #2601: Free energy calculations, soft-core poten...	**New**

History

#1 - 01/07/2014 02:17 PM - Rossen Apostolov

- Project changed from 9 to GROMACS

#2 - 03/29/2016 05:52 PM - Chris Neale

I would like to add some more information to show how slow this can be. It's basically not an option and yet one wonders why hamiltonian replica exchange should be so slow. My guess is that the code is now computing a whole bunch of things that might be useful for mbar If this is the case, it would be great if there were an option to turn all of that part off since the extra efficiency of mbar vs. wham is unlikely to overcome a 10x performance loss during sampling. I provide information below, in which I am using H-REMD to partially decouple lipids in a bilayer, which enhances relaxation times (http://pubs.acs.org/doi/pdf/10.1021/ct500305u -- though these authors rewrote the HREX code to avoid the really slow speed they also got

when decoupling lots of atoms).

Below is information from either (a) temperature REMD (T-REMD) in gromacs 5.1.2 or (b) Hamiltonian REMD (H-REMD), also in gromacs 5.1.2.

Here is the .mdp for my lowest temperature in T-REMD:

```
$ cat REMD/MD_0.mdp

nsteps = 500000000

integrator = sd
dt = 0.002
tinit = 0

lincs-iter =  1
lincs-order =  6
constraint_algorithm =  lincs

;;; charmm section from http://www.gromacs.org/Documentation/Terminology/Force_Fields/CHARMM
constraints = h-bonds
cutoff-scheme = Verlet
vdwtype = cutoff
vdw-modifier = force-switch
rlist = 1.2
rvdw = 1.2
rvdw-switch = 1.0
coulombtype = PME
rcoulomb = 1.2
DispCorr = no

ns_type = grid
nstcomm = 100

nstxout = 0
nstvout = 0
nstfout = 0
nstxtcout = 50000
nstenergy = 50000
nstlist = 10
nstlog=0 ; reduce log file size

ewald-rtol = 1e-5
fourierspacing = 0.12
fourier_nx = 0
fourier_ny = 0
fourier_nz = 0
pme_order = 4

tc_grps             =  System
tau_t               =  1.0
ld_seed             =  -1
ref_t = 310
gen_temp = 310
gen_vel = yes
unconstrained_start = no
gen_seed = -1
Pcoupl = berendsen
pcoupltype = isotropic
tau_p = 4
compressibility = 4.5e-5
ref_p = 1.0
```

And here is a diff of the the above .mdp file with the first replica's mdp file in H-REMD:

```
$ diff REMD/MD_0.mdp REST/MD_0.mdp

54a55,67
> free_energy = yes
> couple-lambda0 = vdw-q
> couple-lambda1 = none
> init_lambda_state = 0
> couple-moltype = POPC
> couple-intramol = yes
> nstdhdl = 200
```

```
> vdw_lambdas = 0 0.01 0.02 0.03
> coul_lambdas = 0 0.01 0.02 0.03
> bonded_lambdas = 0 0.01 0.02 0.03
> calc-lambda-neighbors = 1
> dhdl-derivatives = no
```

Here is the time accounting in T-REMD:

```
    M E G A - F L O P S   A C C O U N T I N G

 NB=Group-cutoff nonbonded kernels    NxN=N-by-N cluster Verlet kernels
 RF=Reaction-Field  VdW=Van der Waals  QSTab=quadratic-spline table
 W3=SPC/TIP3p  W4=TIP4p (single or pairs)
 V&F=Potential and force  V=Potential only  F=Force only

 Computing:                                M-Number         M-Flops  % Flops
-----------------------------------------------------------------------------
 NB VdW [V&F]                           659.164520         659.165     0.0
 Pair Search distance check           14222.443774      128001.994     0.3
 NxN Ewald Elec. + LJ [F]            487803.315024    38048658.572    93.0
 NxN Ewald Elec. + LJ [V&F]            4939.855840      637241.403     1.6
 NxN Ewald Elec. [F]                   1724.128368      105171.830     0.3
 NxN Ewald Elec. [V&F]                   17.509568        1470.804     0.0
 1,4 nonbonded interactions             959.220040       86329.804     0.2
 Calc Weights                          1857.790560       66880.460     0.2
 Spread Q Bspline                     39632.865280       79265.731     0.2
 Gather F Bspline                     39632.865280      237797.192     0.6
 3D-FFT                              136008.942660     1088071.541     2.7
 Solve PME                               63.841600        4085.862     0.0
 Reset In Box                            25.794240          77.383     0.0
 CG-CoM                                  26.818560          80.456     0.0
 Bonds                                  130.875280        7721.642     0.0
 Propers                               1120.420080      256576.198     0.6
 Impropers                                3.192080         663.953     0.0
 Virial                                  63.017890        1134.322     0.0
 Update                                 619.263520       19197.169     0.0
 Stop-CM                                  6.223520          62.235     0.0
 P-Coupling                              61.924800         371.549     0.0
 Calc-Ekin                              248.739040        6715.954     0.0
 Lincs                                  470.676246       28240.575     0.1
 Lincs-Mat                             4341.725248       17366.901     0.0
 Constraint-V                          1582.080664       12656.645     0.0
 Constraint-Vir                          55.582336        1333.976     0.0
 Settle                                 213.582628       68987.189     0.2
-----------------------------------------------------------------------------
 Total                                                40904820.504   100.0
-----------------------------------------------------------------------------

    D O M A I N   D E C O M P O S I T I O N   S T A T I S T I C S

 av. #atoms communicated per step for force:  2 x 25405.3
 av. #atoms communicated per step for LINCS:  2 x 1980.1

 Average load imbalance: 0.8 %
 Part of the total run time spent waiting due to load imbalance: 0.6 %
 Steps where the load balancing was limited by -rdd, -rcon and/or -dds: X 0 %

     R E A L   C Y C L E   A N D   T I M E   A C C O U N T I N G

On 6 MPI ranks

 Computing:          Num   Num      Call    Wall time         Giga-Cycles
                     Ranks Threads  Count      (s)         total sum    %
-----------------------------------------------------------------------------
 Domain decomp.          6    1       1727       3.731         55.836   1.0
 DD comm. load           6    1       1661       0.005          0.068   0.0
 DD comm. bounds         6    1       1727       0.029          0.434   0.0
 Neighbor search         6    1       1662       6.904        103.323   1.9
 Comm. coord.            6    1      38239       1.930         28.878   0.5
 Force                   6    1      39901     277.306       4149.961  76.4
 Wait + Comm. F          6    1      39901       1.729         25.870   0.5
 PME mesh                6    1      39901      40.715        609.316  11.2
 NB X/F buffer ops.      6    1     116379       1.624         24.304   0.4
 Write traj.             6    1          2       1.016         15.209   0.3
```

```
 Update                 6    1      79802       5.591         83.675   1.5
 Constraints            6    1      79802      17.926        268.266   4.9
 Comm. energies         6    1       8046       1.191         17.831   0.3
 Rest                                           3.058         45.769   0.8
-----------------------------------------------------------------------------
 Total                                        362.756       5428.739 100.0
-----------------------------------------------------------------------------
 Breakdown of PME mesh computation
-----------------------------------------------------------------------------
 PME redist. X/F        6    1      79802       4.783         71.572   1.3
 PME spread/gather      6    1      79802      19.101        285.848   5.3
 PME 3D-FFT             6    1      79802       9.627        144.068   2.7
 PME 3D-FFT Comm.       6    1      79802       5.029         75.263   1.4
 PME solve Elec         6    1      39901       2.096         31.374   0.6
-----------------------------------------------------------------------------

               Core t (s)   Wall t (s)        (%)
       Time:     2165.027      362.756      596.8
                 (ns/day)    (hour/ns)
Performance:       19.007        1.263
Finished mdrun on rank 0 Mon Mar 28 21:02:03 2016
```

Here is the time accounting in H-REMD:

```
   M E G A - F L O P S   A C C O U N T I N G

 NB=Group-cutoff nonbonded kernels    NxN=N-by-N cluster Verlet kernels
 RF=Reaction-Field  VdW=Van der Waals  QSTab=quadratic-spline table
 W3=SPC/TIP3p  W4=TIP4p (single or pairs)
 V&F=Potential and force  V=Potential only  F=Force only

 Computing:                               M-Number         M-Flops  % Flops
-----------------------------------------------------------------------------
 NB VdW [V&F]                            57.423520          57.424     0.0
 NB Free energy kernel                 3335062.688538     3335062.689    47.4
 Pair Search distance check               1239.899470       11159.095     0.2
 NxN Ewald Elec. + LJ [F]                42585.090944     3321637.094    47.2
 NxN Ewald Elec. + LJ [V&F]                445.850656       57514.735     0.8
 NxN Ewald Elec. [F]                       147.593376        9003.196     0.1
 NxN Ewald Elec. [V&F]                       1.507392         126.621     0.0
 1,4 nonbonded interactions                 83.563040        7520.674     0.1
 Calc Weights                              161.842560        5826.332     0.1
 Spread Q Bspline                         6905.282560       13810.565     0.2
 Gather F Bspline                         6905.282560       41431.695     0.6
 3D-FFT                                  23697.004320      189576.035     2.7
 Solve PME                                  11.123200         711.885     0.0
 Reset In Box                                2.250400           6.751     0.0
 CG-CoM                                      2.343520           7.031     0.0
 Bonds                                      11.401280         672.676     0.0
 Propers                                    97.606080       22351.792     0.3
 Impropers                                   0.278080          57.841     0.0
 Virial                                      5.510710          99.193     0.0
 Update                                     53.947520        1672.373     0.0
 Stop-CM                                     0.558720           5.587     0.0
 P-Coupling                                  5.400960          32.406     0.0
 Calc-Ekin                                  21.681440         585.399     0.0
 Lincs                                      41.048794        2462.928     0.0
 Lincs-Mat                                 377.249760        1508.999     0.0
 Constraint-V                              137.970858        1103.767     0.0
 Constraint-Vir                              4.865245         116.766     0.0
 Settle                                     18.630994        6017.811     0.1
-----------------------------------------------------------------------------
 Total                                                    7030139.356   100.0
-----------------------------------------------------------------------------

   D O M A I N   D E C O M P O S I T I O N   S T A T I S T I C S

 av. #atoms communicated per step for force:  2 x 25389.0
 av. #atoms communicated per step for LINCS:  2 x 2001.2

 Average load imbalance: 1.4 %
 Part of the total run time spent waiting due to load imbalance: 1.3 %
 Steps where the load balancing was limited by -rdd, -rcon and/or -dds: X 0 %
```

```
    R E A L   C Y C L E   A N D   T I M E   A C C O U N T I N G

On 6 MPI ranks

 Computing:          Num   Num      Call    Wall time         Giga-Cycles
                     Ranks Threads  Count      (s)         total sum    %
-----------------------------------------------------------------------------
 Domain decomp.         6    1        150       0.328          4.904   0.1
 DD comm. load          6    1        144       0.001          0.008   0.0
 DD comm. bounds        6    1        150       0.003          0.049   0.0
 Neighbor search        6    1        145       1.961         29.340   0.5
 Comm. coord.           6    1       3331       0.185          2.772   0.1
 Force                  6    1       3476     342.581       5126.810  93.3
 Wait + Comm. F         6    1       3476       0.158          2.361   0.0
 PME mesh               6    1       3476      10.697        160.076   2.9
 NB X/F buffer ops.     6    1      10138       0.154          2.302   0.0
 Write traj.            6    1          2       1.142         17.092   0.3
 Update                 6    1       6952       0.497          7.431   0.1
 Constraints            6    1       6952       5.805         86.867   1.6
 Comm. energies         6    1        701       1.161         17.371   0.3
 Rest                                           2.342         35.055   0.6
-----------------------------------------------------------------------------
 Total                                        367.012       5492.439 100.0
-----------------------------------------------------------------------------
 Breakdown of PME mesh computation
-----------------------------------------------------------------------------
 PME redist. X/F        6    1      10428       5.509         82.446   1.5
 PME spread/gather      6    1      13904       2.333         34.909   0.6
 PME 3D-FFT             6    1      13904       1.655         24.762   0.5
 PME 3D-FFT Comm.       6    1      13904       0.835         12.500   0.2
 PME solve Elec         6    1       6952       0.352          5.264   0.1
-----------------------------------------------------------------------------

               Core t (s)   Wall t (s)        (%)
       Time:     2196.030      367.012      598.4
                 (ns/day)    (hour/ns)
Performance:        1.637       14.665
Finished mdrun on rank 0 Mon Mar 28 22:31:05 2016
```

Both were run like this:

```
ibrun -np 24 /oasis/scratch/comet/cneale/temp_project/exec/gromacs-5.1.2/exec_openmpi/bin/gmx_mpi mdrun -notun
epme -deffnm MD_ -dlb yes -npme 0 -cpt 60 -maxh 0.1 -cpi MD_.cpt -multi 4 -replex 200
```

There is the plumed route, which is viable and I have used it. The problem is that the plumed release cycle lags behind the gromacs release cycle and so some of the efficiencies and options in newer gromacs versions are not always available when using plumed to do the coupling.

Thank you,
Chris.

#3 - 03/30/2016 06:42 PM - Mark Abraham

- Target version set to future

As noted in discussion on gmx-users, the performance of the GROMACS free-energy kernels is indeed relatively poor. It seems likely we can do a lot better, once we have some new infrastructure sorted out. But that won't happen before GROMACS 2017, sorry.

#4 - 03/30/2016 06:49 PM - Chris Neale

- File RESTandREMDtprs.tgz added

tarball uploaded with .tpr and short run outputs for a single system with both T-REMD and H-REMD (the later to do REST). README file describing usage is enclosed. Please note the system is quite small because it is designed for testing only. I realize that the problem here is that I'm trying to use a tool outside of its design specifications and that the free energy code has a lot of parts that are really useful for decoupling a small number of atoms. However, one would hope that eventually we can somehow decouple large numbers of atoms without taking more than a 5% efficiency hit if we are willing to give up some of the useful PMF-related tools available in the core free energy code. Of course, there would have to be sufficient interest and that part is not clear.

#5 - 06/23/2016 04:01 PM - Mark Abraham

- Related to Feature #1665: improve free energy non-bonded kernel performance added

#6 - 07/11/2016 10:42 PM - Mark Abraham

- Target version changed from future to 2018

Noted. Chris's reported slowdown (factor of ~13 in number of steps in that -maxh, attributable to the branchy FE code path and lack of SIMD support, in his case) is indeed horrible. There is a rather extreme amount of configurability in the free-energy kernel, but one of the planned advantages of the new kernel generation scheme is that we can probably support a mode where by default one gets a slow kernel like now, but perhaps issues a run-time warning about how one can do a custom CMake call to optimize for the actual target(s) required in a future run. Or maybe (simpler) a "really, I don't care, compile everything" configuration option.

It's possible to make a start on that even before the new scheme materializes - compiling out run-time constants can take place even if the kernels are still using group-scheme neighbourlists and no SIMD. Or maybe it's easier to start from a modified form of the Verlet plain C kernels?

My suspicion is that Chris's "extra stuff for mbar" theory isn't the main problem, but I'd verify that before starting work.

#7 - 07/28/2016 05:35 PM - Mark Abraham

- Related to Bug #2014: GROMACS free energy memory footprint added

#8 - 03/15/2017 05:40 PM - Mark Abraham

- Target version changed from 2018 to future

I'm not aware of work underway that could be done in time for 2017 release

#9 - 12/07/2017 12:09 PM - Sebastian Wingbermühle

- File standardMD.tpr added

- File FEP.tpr added

- File nb_free_energy.cpp added

- File standardMD.log added

- File FEP.log added

- File tuned-FEP.log added

Hello!

Here are some results concerning the performance of the FEP code compared to standard MD simulations using alanine dipeptide in TIP4P water as test system (see .tpr files attached). The simulations were intended to be a performance test for the solute tempering approach known from replica exchange. Therefore, the potential energy terms of the complete alanine dipeptide are scaled, whereas the potential energy of the water is not modified. In this setup, lambda = 0 corresponds to the state in which the potential of alanine dipeptide is not changed and lambda = 1 means full scaling. For a fair comparison, physically identical simulations were carried out, i.e. standard MD simulations are compared to simulations using the FEP code with lambda = 0. Soft-core potentials and BAR/MBAR calculations were turned off. Moreover, I modified the non-bonded free-energy kernel by commenting out all if-loops and switch-case constructions dealing with different user options for the treatment of Coulomb and van der Waals interactions such that the resulting kernel can only be used for my settings (see attached). The performance obtained with the modified kernel is listed as "tuned FEP" below. Further technical details are:

version:
GROMACS 2016.3

command to launch simulations:
gmx mdrun -pin on -deffnm REST -cpnum -cpt 60 -cpo cpt/state

hardware:
CPU: Xeon E3-1270 V2 CPUs (4 cores at 3.5 GHz)
GPU: 1 Nvidia GeForce GTX 770 graphics card (1536 CUDA cores at 1.0 GHz and 2 GB GDDR5 (non-ECC) memory)

Results (compare .log files attached for further details):

Setup	Performance (ns/day)
standard MD	490.878
FEP	126.319
tuned FEP	132.393

#10 - 03/15/2018 02:51 PM - Mark Abraham

Thanks for the feedback. We know these kernels are a problem that needs quite a bit of attention. One useful tip is that the intel compiler does a much better job than gcc on the FEP short-range interactions that are the bulk of the problem here. Your data is interesting because it seems to suggest that the performance improvement from icc is probably more about better vectorization than better code generation in the face of lots of branches.

#11 - 04/17/2018 06:51 PM - Roland Schulz

Could you update the tpr files or upload the files to generate the tpr files? Those tpr can't be execute with 2018 version.

#12 - 08/08/2018 03:30 PM - Mark Abraham

- Related to Feature #2601: Free energy calculations, soft-core potential added

#13 - 08/26/2019 04:07 PM - Magnus Lundborg

- File testosterone_lj_decoupling.tpr added

- File testosterone_q_decoupling.tpr added

Two more examples of affected systems attached.

#14 - 08/26/2019 04:28 PM - Magnus Lundborg

- File testosterone_unperturbed.tpr added

This is the same system as above, but without FE.

#15 - 08/26/2019 04:55 PM - Berk Hess

I would suggest an approach analogous to do_pairs_simple()
We should vectorize over j and duplicate the last j to get to multiples of the SIMD width. Then nearly all reals in the kernel needs to be templated, so they can be real or a simd real.

For good performance we likely need to template the kernel on interaction types to get rid of all conditionals.

All off this should be rather straightforward, I think.

Files

RESTandREMDtprs.tgz	6.13 MB	03/30/2016	Chris Neale
standardMD.tpr	66.4 KB	12/07/2017	Sebastian Wingbermühle
FEP.tpr	66.7 KB	12/07/2017	Sebastian Wingbermühle
nb_free_energy.cpp	41.2 KB	12/07/2017	Sebastian Wingbermühle
standardMD.log	612 KB	12/07/2017	Sebastian Wingbermühle
FEP.log	643 KB	12/07/2017	Sebastian Wingbermühle
tuned-FEP.log	643 KB	12/07/2017	Sebastian Wingbermühle
testosterone_q_decoupling.tpr	1.1 MB	08/26/2019	Magnus Lundborg
testosterone_lj_decoupling.tpr	1.1 MB	08/26/2019	Magnus Lundborg
testosterone_unperturbed.tpr	1.1 MB	08/26/2019	Magnus Lundborg