

GROMACS - Task #2452

Reduce data dependencies in mdrun algorithms

03/14/2018 12:47 PM - Berk Hess

Status:	In Progress
Priority:	Normal
Assignee:	Berk Hess
Category:	core library
Target version:	
Difficulty:	hard
Description	
<p>Currently mdrun can and does assume full dependency of data between algorithms in one domain/rank. This limits options for task parallelization, overlap and leads to a large number of OpenMP barriers.</p> <p>The minimal, practical unit of interdependent work in a simulation that uses constraints only on H-bonds is a constraint group. The goal should be to make data blocks that consist of a limited number of constraint groups.</p> <p>A prerequisite for this is put constraint groups instead of atoms on the nbxn search grid.</p>	

History

#1 - 03/14/2018 12:59 PM - Berk Hess

- Difficulty hard added

- Difficulty deleted (uncategorized)

#2 - 03/14/2018 08:47 PM - Szilárd Páll

I think this actually consists of at least two distinct set of todos:

- encapsulating the current large/coarse tasks with dependencies as well as serial/comm tasks in the current sequentially expressed code (mainly documentation and mapping out all different paths with various dependencies)
Goal: given a flexible task scheduler, be able to write a schedule by simply mapping out tasks and dependencies (without having to track down corner-cases), e.g. to execute in arbitrary order and arbitrary concurrency/overlap of independent coarse tasks
- using an intra-domain decomposition generating finer-grained tasks that allow: i) independent scheduling on accelerators without DD ii) flexible data affinitized concurrent SMP scheduling
_Goal: finer-grained tasking, staggered update _