

GROMACS - Feature #2934

Feature # 2816 (New): GPU offload / optimization for update&constraints, buffer ops and multi-gpu communication

Feature # 2817 (In Progress): GPU X/F buffer ops

GPU X Buffer ops

04/25/2019 05:02 PM - Alan Gray

Status:	New
Priority:	Normal
Assignee:	
Category:	
Target version:	2020
Difficulty:	uncategorized
Description	
Implement and improve the GPU version of position buffer operations. Gerrit change 9169 implements the functionality, and a follow-up change will improve as below.	
TODOs:	
<ul style="list-style-type: none">• Use pinned host vectors for grid and gridset arrays and remove explicit cudaHostRegister/unregister calls in init fn• Replace allocatedDeviceBuffer with reallocatedDeviceBuffer in init fn• Improve variable naming in init and buffer ops fns• Fix issue with position buffer pinning to allow use of gmx api for memcpy• Implement sync point between PME and NB streams.• Improve mechanism for deciding if position buffer needs to be copied to GPU in advance of buffer op• move GPU conditional out of OpenMP loop in atomdata_copy_x_to_nbat_x. Precompute maxAtomsInColumn and that and store it in the grid object, instead of recomputing it every step.• move the complex coordinate management into a single place (from where it can be refactored easier): code added by early X buf ops changes added technical debt and complexity by conditionally doing data copying and management of xrvcc when this is not done in PME. Consolidation and refactoring needed• refactor grid data members added without any grouping into the main nonbonded data structure.• consider uniform behavior across search and non-search steps (ATM at search the xyzq fmt is generated in search and copied while non-ns steps do the conversion which leads to complexity).	

Associated revisions

Revision f457e9a4 - 05/22/2019 02:29 PM - Alan Gray

Code quality improvements for CUDA position buffer ops

- improved variable naming
- use of reallocateDeviceBuffer

Follow up to Gerrit change 9169
Implements part of #2934

Change-Id: Ie948f624ae6f7e2df7a2b6c6c734d08d862096e5

Revision 6e14d4ac - 05/22/2019 08:53 PM - Alan Gray

Pin X buffer with DD and use proper abstraction in buffer ops

This patch fixes an issue that the coordinate buffer was not being pinned with DD, and replaces a raw cudaMemcpy call in the buffer ops routine with proper abstraction through the copyToDeviceBuffer API.

Follow up to Gerrit change 9169
Implements part of #2934

Change-Id: I64aec66157ceb5eb04bcfd7cfe24b6ea5c18e4ae

Revision 3329a50b - 07/11/2019 05:23 PM - Szilárd Páll

Use HostVector for Grid/GridSet data need on-GPU

Grid.cxy_na_, Grid.cxy_ind_, GridSet.cells and GridSet.atomIndices

have been converted from `std::vector` to `gmx::HostVector`. This allow the code to pin the `HostVector` when X buffer ops is used and to eliminate the hacky pin/unpin in CUDA buffer ops functions.

Part of #2934
Refs #2817

Change-Id: lcca21dd076128ec582f805ed96e253dfab461270

Revision c8951db1 - 07/22/2019 10:10 AM - Szilárd Páll

Conditionally pin GPU-related grid data

Data that is transferred to the GPU when the buffer ops is offloaded is now only pinned when the nonbonded module uses GPU offload avoidign the runtime errors encountered when a GPU-enabled build does not detect a GPU and therefore the CUDA runtime refuses to register the memory.

Refs #2817 #2934

Change-Id: labbc0d9f37fad0e88cd39a078af1346e8f713ec1

History

#1 - 05/28/2019 12:31 PM - Alan Gray

- *Description updated*

#2 - 06/06/2019 03:26 PM - Alan Gray

- *Description updated*

#3 - 06/17/2019 02:20 PM - Szilárd Páll

- *Description updated*

#4 - 08/14/2019 05:58 PM - Alan Gray

- *Description updated*

#5 - 09/24/2019 06:07 PM - Alan Gray

- *Description updated*