Optimizing Large-Scale Plasma Simulations on Heterogeneous Memory with Effective Data Placement Across Memory Hierarchy

Jie Ren[1], Jiaolin Luo[1], Ivy Peng[2], Kai Wu[1], and Dong Li[1]

[1]University of California Merced, [2]Lawrence Livermore National Laboratory

Motivation

Enable large-scale plasma simulation.

❑ **Plasma Simulation**

- Plasma simulations are critical for understanding plasma dynamics in many fields.
- Particle-In-Cell (PIC) technique is one of the most popular algorithms in plasma simulation.

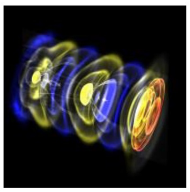


Fig. Plasma simulation

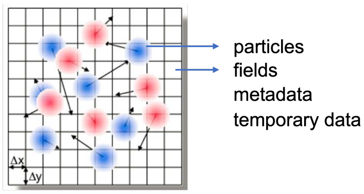


Fig. Particle-in-cell (PIC) Method

- WarpX[1] is a state-of-the-art plasma accelerator simulation code to be deployed in the upcoming exascale HPC system
- WarpX can have large memory consumption
  - ➢ a production run on the Cori supercomputer simulating 62 billions of particles consumes up to 8.9 TB memory

❑ **Persistent memory (PM) enables WarpX simulation in larger scales**

- Compared with Summit and Sierra (the top 2 and top 3 supercomputers), the Optane-based supercomputer increases the simulation scale by 3.1x and 5.5x respectively.

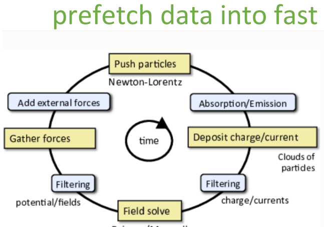
Performance Characterization on WarpX

❑ **Particles & fields dominate memory consumptions of WarpX**

- In WarpX, there are four types of data objects, including particles, fields, metadata, and temporary data
  - ➢ particles and fields should be the major optimization targets

❑ **WarpX has iteratively, streaming like memory access pattern**

- WarpX is an iterative solver; each iteration includes five major computation phases.
  - ➢ The memory access pattern provides opportunity to prefetch data into fast memory/processor cache



Execution phases:
(1) Particle Pusher
(2) Current deposition
(3) Field Solver
(4) Field Gather
(5) Others

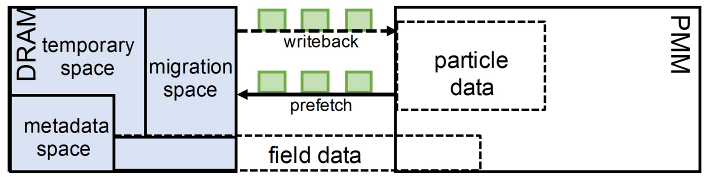
Most time-consuming phases: (1)–(3)

Fig . The computation routine of PIC method

❑ **The execution of WarpX is not bounded by DRAM/persistent memory bandwidth.**

➢ rich bandwidth for data migration

WarpX-PM: An Automatic Data Placement Solution for WarpX



❖ The heterogeneous memory space is partitioned based on the functionality and access patterns of data objects in WarpX.

❑ **Static data placement for metadata space & temporary space**

- Pin performance-critical data into DRAM.

❑ **Processor cache prefetch for field data**

- Prefetch field data to processor-cache directly.
  - fields data that are not stored in a contiguous memory space -> difficult to dynamically migrate between DRAM and PM

❑ **Processor cache prefetch for field data**

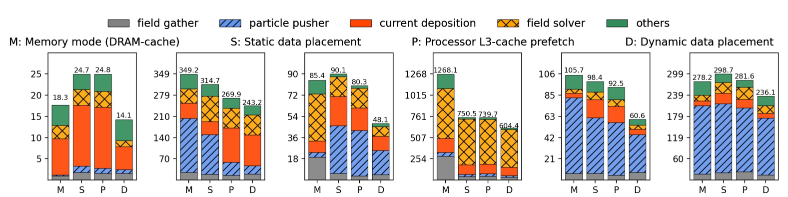
- Dynamic migration (copy) of particles batch by batch using helper threads
  - Particles are periodically migrated into migrations space;
  - Computation always accesses particles in DRAM.

Experiment Results

❖ A combination of **static data placement**, **dynamic migrations** and **cache prefetching** can make the best use of the PM-based memory hierarchy.

➢ Testing bed

- 2 x Intel Xeon Scalable processor
- 35.75 MB last level cache
- 192GB DRAM + 1.5TB Persistent Memory (Intel Optane DC PMM)



✓ WarpX-PM outperforms existing hardware-level memory management (DRAM-cached), OS-level memory management (NUMA first-touch policy)

✓ Different execution phase in different simulation problems exhibit different sensitivity to those techniques.

✓ WarpX-PM with the three proposed techniques achieves the best performance in all problems.

[1] J-L Vay, A Almgren, J Bell, L Ge, DP Grote, M Hogan, O Kononenko, R Lehe, A Myers, C Ng, and others. 2018. Warp-X: A new exascale computing platform for beam–plasma simulations. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 909 (2018), 476–479