Towards Wafer-Scale Acceleration of the Lattice Boltzmann Method on the Cerebras CS-2

Kouki Sakai†, Saho Oriharа†, Takashi Shimokawabe††, and Takaaki Miyajima†
†Meiji University (Japan), ††The University of Tokyo (Japan)
e-mail:takaaki.miyajima@cs.meiji.ac.jp

明治大学 MEIJI UNIVERSITY

Overview

In recent years, high-performance computing systems have adopted architectures integrating hundreds to thousands of compute nodes with GPUs and accelerators. Achieving both scale-up and scale-out performance remains challenging, as the ratio of memory bandwidth to computational performance (Byte/Flops) continues to decline, leading to complex hardware. The Cerebras CS-2 is a new accelerator developed for training large language models, featuring the Wafer-Scale Engine 2 (WSE-2), which uses an entire 300 mm semiconductor wafer as a single processor.

The Lattice Boltzmann Method (LBM) is a computational fluid dynamics algorithm widely utilized in various fields, including engineering and physics, due to its effectiveness in handling complex boundary geometries and multiphase flows. Furthermore, its algorithm exhibits high affinity with parallel processing, making it exceptionally suitable for large-scale simulations.

The ultimate goal of this research is to evaluate the applicability of the CS-2 to large-scale scientific computing. As part of this objective, we have currently implemented a two-dimensional (2D) LBM on the CS-2 and evaluated its computational performance.

Cerebras Wafer Scale Engine 2 (WSE-2)

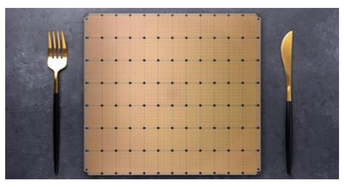
The entire 300 mm wafer is used as one chip

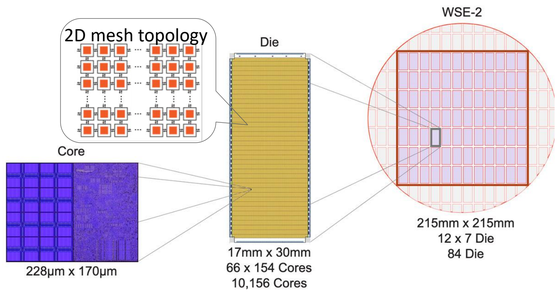
- 84 dies are interconnected with a parallel interface called Cross Scribe Line Connections.
- Process rule: TSMC 7 nm, chip area: 46,225 mm^2.
- Running at 850MHz and 16.5kW for scientific computation.

Cerebras CS-2 System

Specifications

- Total number of PEs: 853,104 (792x1,078), **effective: 745,500 (750x994).**
- Total number amount of on-chip SRAM: 40 GB, **effective: 35.78 GB.**
- Memory bandwidth: 20 PB/s, **effective: up-to 8.80 PB/s.**


WSE-2 Chip

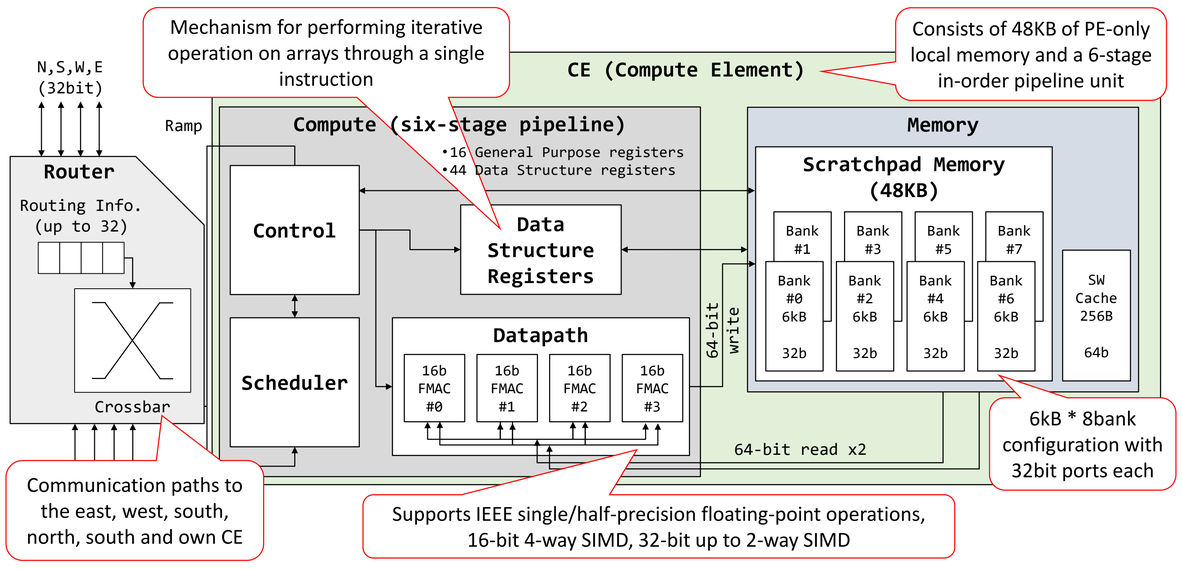


Physical structure of WSE-2

Processing Element (PE): Memory, CE, and Router

PE comes with scratchpad memory, compute element (CE) and router

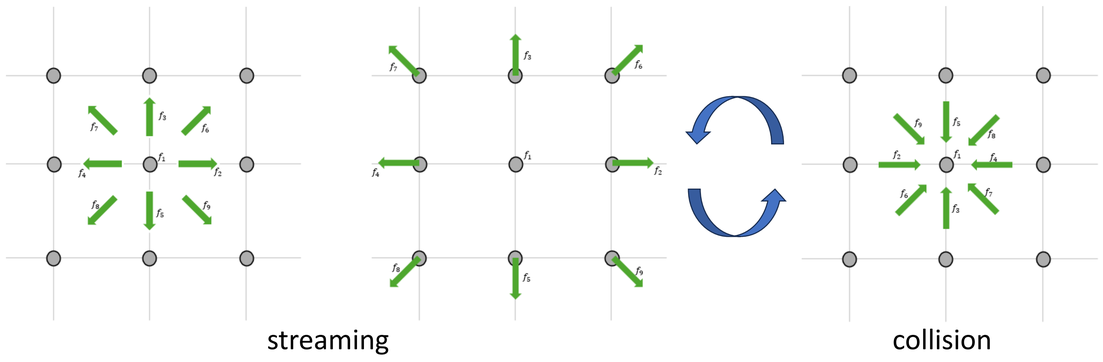
- CE and memory are connected to two 64-bit read ports and one write port.
- 853,104 Processing elements (PEs) with 2-D mesh topology.



Lattice Boltzmann Method (LBM)

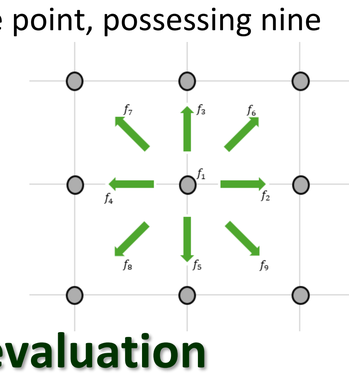
Algorithm overview

- Simulate the streaming and collision of virtual particles at fixed lattice points.
- Its inherent parallelism, where operations at each lattice point are independent, enables high computational efficiency on parallel architectures such as supercomputers and GPUs.



- D2Q9 model (2-Dimension 9-Quadrature) is used in this implementation.
- A distribution function f_i is defined at each lattice point, possessing nine velocity vectors C_i.

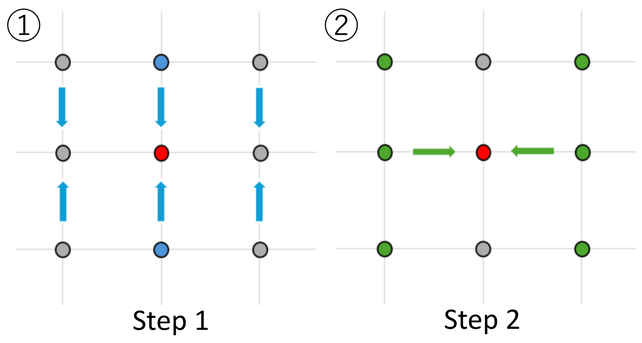
$$\boldsymbol{c}_i = \begin{cases} (0,0), & i = 1, \\ (\pm 1, 0), (0, \pm 1), & i = 2,3,4,5, \\ (\pm 1, \pm 1), & i = 6,7,8,9, \end{cases}$$



Implementation and evaluation

Two-step communication

- The WSE-2 can only communicate in the north, south, east and west direction.
- LBM communicates diagonally, so it is divided into tw-steps.



① Step 1
Communicate in the north-south direction, including adjacent columns.

② Step 2
Communicate in the East-west direction, including diagonal data.

Result

- Compare the computational performance of the single-PE implementation with the multi-PE implementation (17×17 lattice points).
- The evaluation metrics are cycle count, FLOPS/s, and LUPS.
- Lattice Updates Per Second (LUPS) is a standard performance metric widely used in the field of LBM.

$$LUPS = \frac{Total\ Number\ of\ Lattice\ Points \times Total\ Time\ Steps}{Execution\ Time\ [seconds]}$$

Evaluation: processing cycles, performance, and parallel efficiency

Implementation	Proc. cycles	GFlops/s	MLUPS	Speed Up
Single-PE	688,865	0.075	0.275	1.0
289-PE	6,464	8.9	29.5	106.6

Comparison of computation time and performance

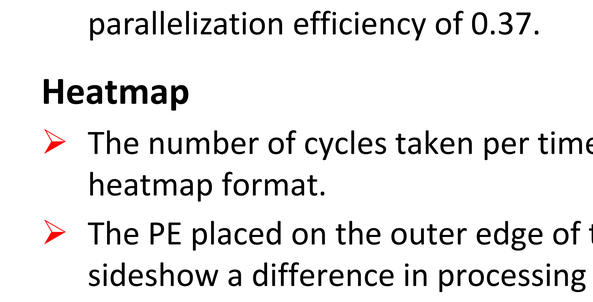
- The multi-PE version achieved about x106 speedup over the single-PE case.
- The ideal speedup is x289, and the achieved speedup corresponds to a parallelization efficiency of 0.37.

Heatmap

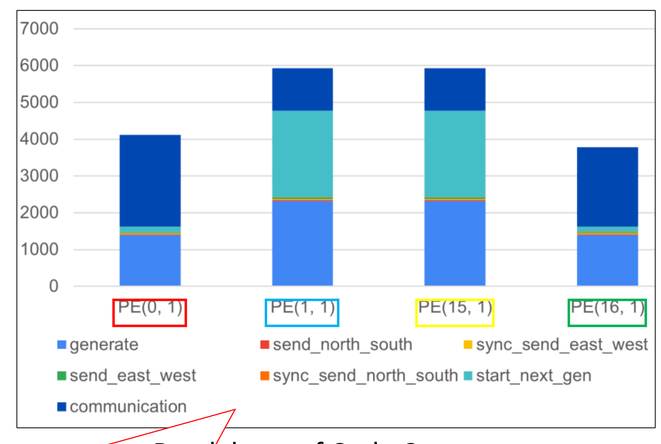
- The number of cycles taken per time step loop for each PE is shown in a heatmap format.
- The PE placed on the outer edge of the grid and the PE placed on the inner sideshow a difference in processing time of about 2000 cycles.

Breakdown of Cycle Counts

- The leftmost and rightmost PEs are responsible for boundary conditions, but their task number of cycles were almost identical.
- Interior PEs, including those not shown in the figure, exhibited the same number of cycles.



Heatmap



Breakdown of Cycle Counts

Computational tasks are mapped according to their physical locations.

The “generate” function runs only at the very beginning of the simulation. The “start_next_gen” function performs the collision processing, and the "communication" function performs the time spent waiting for data arrival during inter-PE communication.`

Future work

- Optimization of PE Inter-Communication.
- Adjusting the Number of Grids Assigned to 1PE.
- Domain Partitioning Strategy Considering Load Balancing.

Acknowledgement and references

[1] DavidH. Bailey, XiaoyeS. Li, and Yozo Hida. 2003. QD: A Double-Double/Quad- Double Package. https://doi.org/10.11578/dc.20210416.14
[2] Y. Hida, X. S. Li and D. H. Bailey, "Algorithms for quad-double precision floating point arithmetic," Proceedings 15th IEEE Symposium on Computer Arithmetic. ARITH-15 2001, Vail, CO, USA, 2001, pp. 155-162, doi: 10.1109/ARITH.2001.930115.
This work was supported by Japan Science and Technology Agency (JST) as part of Adopting Sustainable Partnerships for Innovative Research Ecosystem (ASPIRE), Grant Number JPMJAP2341. This work was supported by JSPS KAKENHI Grant Number 24K14972.