Exploring Tree-Based Approaches for Collective Communications on Cerebras WSE-3

Saverio Pasqualoni[1,2], Marco Canini[1] (advisor), Daniele De Sensi[2] (advisor)

[1] King Abdullah University of Science and Technology [2] Sapienza University of Rome

1. Introduction

The Cerebras WSE-3 is a single 5nm wafer used as one processor, where 900,000 cores (PEs), organized in a 2D mesh interconnect, can exchange 32-bits messages (wavelets) with its neighbors in a single clock cycle. An MPI-style set of collectives is provided by the Cerebras SDK, but its performance is limited.

We explore tree-based schedules for 1D Reduce because it's the **main building block** for both 2D collectives and AllReduces (broadcasts are near optimal through hardware multicast [1]).

2. Cost model

We use the model established by [1]:

$$T = \max\left(C, \frac{E}{N} + L\right) + (2T_R + 1) \cdot D$$

Link load · SEND cost · Contention · Max wavelet distance · Depth

This however represents a **lower bound** of what an algorithm can achieve, and **implementation matters**. In this work we analyze both algorithmic advantages and implementation decisions.

3. Binomial vs. Bine

Binomial Trees are recursively defined trees of logarithmic depth extensively used in collective communications, due to their $\log_2(P)$ communication rounds. On WSE-3, the cost is [1]:

$$T_{\text{Tree}} = \max\left(B \cdot \log_2(P),\ \frac{BP \cdot \log_2(P)}{2(P-1)} + P - 1\right) + (2T_R + 1)\log_2(P)$$

Bine (Binomial Negabinary) trees [2] are binomial trees built via **negabinary rank encodings**, which keep a $\log_2(P)$ step count (D) and contention (C) while scaling down energy E and maximum distance.

Since $\rho(i) \approx \frac{2}{3}2^{i-1}$ (*Steps magnitude*) Then $E_{\text{Bine}} = \frac{B}{3} P \log_2 P$ And $L = \frac{2(P-1)}{3}$

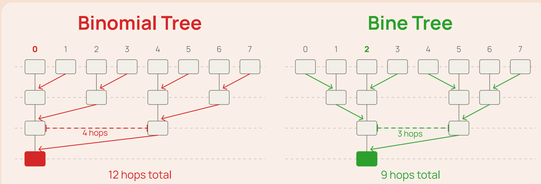


Figure 1: *Binomial Tree and Bine Tree communication schedule on 8 nodes.*

Bine's performance **depends critically on the implementation**. For a 4 byte Reduce on 512 PEs, three transports of the same schedule span 2.2×:

- **CE relay**: each intermediate PE pulls the vector down the ramp and re-sends it — *4071 cycles;*
- **Static routing:** routers forward directly, the wavelets never enter the CE — *2078 cycles;*
- **Synchronous tasks:** one task per rank replaces a continuation per operation — *1841 cycles.*

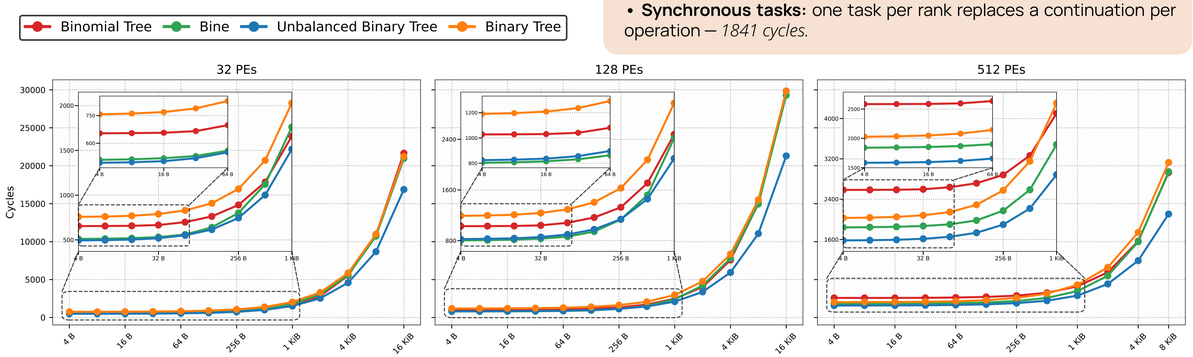


Figure 5: *Cycle counts of the best implementation of each tree schedule at 32, 128, and 512 PEs. Bine outruns the binomial tree at 77–92% of sizes, the margin growing with P; the unbalanced tree beats the balanced one at every size and scale.*



Figure 2: *The best Bine implementation outperforms Tree by 1.28 - 1.40× when B ≤ 64, the gain growing with P toward the $^3/_2$ limit; as contention dominates, both converge within 3%.*

4. Binary Tree

Binary trees [3] are built by **recursively bisecting** the rank range: each subroot sits at the midpoint of its range and the recursion terminates at single PEs. Distances halve with depth, and the tree has $\log_2$P - 1 levels; since each PE can receive only one message per round, the schedule's depth is D=2($\log_2$ P - 1).

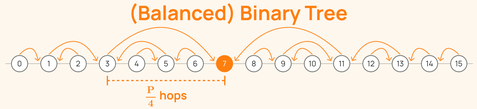


Figure 3: *Binary Tree construction*

The **Unbalanced Binary Tree** cuts each range at **one third** rather than the median, **reducing the total distance** at the cost of a deeper schedule.

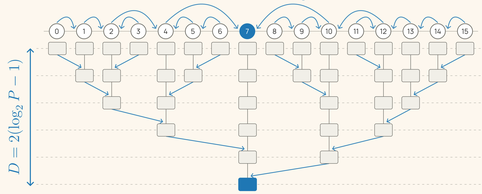


Figure 4: *Unbalanced Binary Tree construction*

$$T_{\text{BT}} \approx \max\left(2B(\log_2 P - 1),\ \frac{E}{P-1} + \frac{P}{2}\right) + (2T_R + 1)\,2(\log_2 P - 1) \qquad E \approx \tfrac{B}{2}P(\log_2 P - 1)$$

Unfused, a node sends only once its vector is complete; fusing each receive with the onward send, children ordered far-side last, is worth 2.1× (balanced) and 3.1× (unbalanced) over the baseline.

5. Two-Phase Tree

Two-Phase [1] combines Tree's low depth with Chain's low contention: a row is partitioned into groups of S = $\sqrt{P}$ consecutive PEs, and each group chain-reduces to its leftmost PE. The $^P/_S$ representatives then chain-reduce among themselves:

$$T_{\text{2P}} \le \max\left(2B,\ 2B - \frac{2B}{\sqrt{P}} + P\right) + \left(2\sqrt{P} - 2\right)(2T_R + 1)$$

We replace the outer chain with **Binomial and Bine trees**, trading the $^P/_S$ - 1 rounds for $\log_2(^P/_S)$. However, the way S is chosen must be tuned in order to reach optimality again:

$$T(S) \approx B\left(1 - \frac{1}{S}\right) + cB\log_2\frac{P}{S} + L + (2T_R + 1)\left((S-1) + \log_2\frac{P}{S}\right)$$

Differentiate over S:

$$\frac{\mathrm{d}T}{\mathrm{d}S} = \frac{B}{S^2} - \frac{cB}{S\ln 2} + (2T_R + 1)\left(1 - \frac{1}{S\ln 2}\right)$$

Thus:

$$S^\star = \frac{1 + cB/(2T_R + 1)}{\ln 2}$$

6. Evaluation of SotA

Methodology:
- times are transmission only, from the cycle-accurate simulator;
- time is measured at the root, which is first to enter the schedule and last to exit;
- Auto-Gen [1] is excluded by kind, since it generates collective code off-chip.

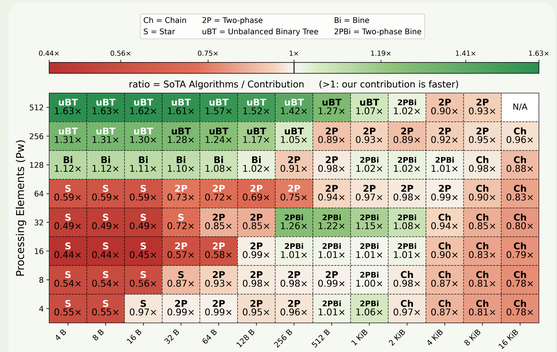


Figure 6: *Best fixed pattern of [1] per configuration against the best of our contributions (ratio > 1: ours faster).*

In **latency-bound** regions with PEs ≥ 128, the tree-based approaches lead SotA: the unbalanced binary tree by up to 1.63× at 512 PEs, Bine by 1.12× at 128. Both owe it to lower communication distances (Fig. 5).

As PEs decrease, distances shrink until fixed per-round overheads dominate and Star [1], where every PE sending directly to the root, leads instead. These small configurations carry little practical weight, however: applications engage most of a row to exploit the wafer [1] [3], placing production reductions precisely where our trees win.

Head to head, the **Two-Phase Bine** does improve standard Two-Phase on long vectors, by up to 1.12×. The improvement is somewhat hidden against SotA, as the Chain algorithm (SDK's implementation) is optimal in that range, sitting on the port bound C=B.

7. Conclusions & Future work

On the WSE a round costs five cycles and a hop costs one: trading rounds for distance wins the latency-bound region, up to 1.63× over the fixed-pattern SotA at 512 PEs.

The other half of the result is the implementation: the same schedule can span more than 2× in runtime depending on how data moves through the fabric, and fusing each receive with the onward send is worth 2–3× on the trees that rely on it most.

As for future work directions, we plan to:

- **Physical WSE-3**: expand kernel validation beyond the cycle-accurate simulator to real hardware;
- **2D collectives & AllReduce**: expanding research evaluation and algorithmic considerations to 2D domains and to complete AllReduce kernels;
- **Applications**: measuring end-to-end gains in workloads built on row reductions.

References

[1] Luczynski, Piotr, et al. '*Near-Optimal Wafer-Scale Reduce*'. Proceedings of the 33rd International Symposium on High-Performance Parallel and Distributed Computing (HPDC'24). https://doi.org/10.1145/3625549.3658693

[2] De Sensi, Daniele, et al. '*Bine Trees: Enhancing Collective Operations by Optimizing Communication Locality*'. In Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC'25). https://doi.org/10.1145/3712285.3759835

[3] He, Congjie, et al. '*WaferLLM: Large Language Model Inference at Wafer Scale*'. 19th USENIX Symposium on Operating Systems Design and Implementation (OSDI 25) https://doi.org/10.48550/arXiv.2502.04563.