Powering Extreme-Scale HPC with Cerebras Wafer-Scale Accelerators

Adam Lavely, HPC Performance Engineer

Introduction

The Cerebras CS-2 system is the world's largest and most powerful HPC and AI accelerator. This unique system was designed from the ground up to target bottlenecks affecting the time-to-solution for ML/AI and HPC workloads. It has achieved accelerations greater than 100X for some applications. The speed-up comes from the unique architectural design of the wafer-scale processor at the heart of the system. Single-cycle local memory access and communication between the 850,000 cores allow for cluster-level computations to be undertaken within a single system.

In this paper, we will explore the challenges facing HPC developers today and show how the Cerebras architecture can help to accelerate sparse linear algebra and tensor workloads, stencil-based partial differential equation (PDE) solvers, N-body problems, and spectral algorithms such as FFT that are often used for signal processing.

We will also touch on the new Cerebras Software Development Kit (SDK) that allows developers can target the WSE's microarchitecture directly using a familiar C-like interface to create custom kernels for their own unique applications.

Contents

Figure 1. Cerebras CS-2 systems in our colocation facility.

A Distributed Problem

There is a growing realization in the high-performance computing (HPC) field that the traditional scale-out approach – "node-level heterogeneity" – of hundreds or thousands of identical compute nodes loaded up with GPUs or other accelerators has limitations. The efficiency of algorithms tends to decrease as they are split, or "sharded" across many nodes, because moving data between those nodes is such a slow process. Writing code for massively parallel systems is a specialized skill and is very time consuming.

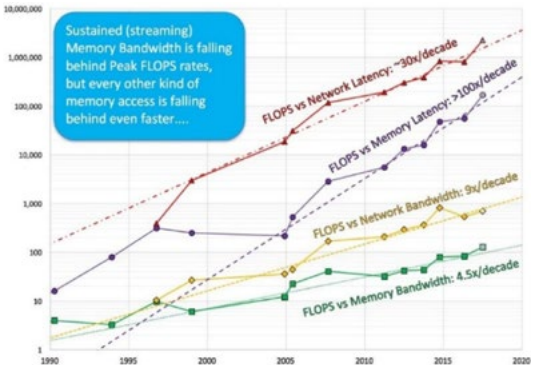


Figure 2. Trends in the relative performance of floating-point arithmetic and several classes of data access for select HPC servers over the past 25 years. Source: John McCalpin.

As John McCalpin from the Texas Advanced Computing Center famously observed, traditional CPUs and GPUs have been improving faster than the networks that connect them for several decades now.[1] Any scaling problem with conventional hardware now will likely just be exacerbated with future commodity hardware. For example, if you are constrained by memory bandwidth now, you are likely to continue to be because the trend is for FLOPs to grow at 4.5X the memory bandwidth.

System Level Heterogeneity

A far better solution is to switch to a completely different architecture and scale up the power of individual compute nodes, that is to add very powerful accelerators to the network as complete, independent compute nodes capable of fitting problems of interest within a single chip. The time saved by keeping communication local to a single node can move many problems from being "communication-bound" – limited by the speed of communication – to being "compute-bound" – limited by the speed of the actual processing elements. This transition allows the computational resources available to be used efficiently, rather than sitting idle waiting for data to arrive. Additionally, the need to implement elaborate schemes to "hide" communication behind available compute evaporates. This simplifies the programming greatly and allows for more time to be spent on compute optimization.

Bronis de Supinski, CTO of Lawrence Livermore National Laboratory's Livermore Computing Center, has described this architecture as "system-level heterogeneity". The Cerebras CS-2 system is the world's most powerful network-attached accelerator.

Overview of the Cerebras System

As mentioned above, the CS-2 system is very different to a convention HPC cluster, with its racks of identical servers wired up with a network fabric such as Ethernet or InfiniBand (Figure 1).

At the heart of the Cerebras CS-2 system is the second-generation Wafer-Scale Engine (WSE-2) The WSE-2 is a massive parallel processor built from a single 300mm wafer. It offers 850,000 cores, optimized for sparse linear algebra with support for FP16, FP32, and INT16 data types. It contains 40GB of onboard SRAM divided between its cores, with 220Pb/s of interconnect bandwidth and 20PB/s of memory bandwidth. The 40GB of high-speed SRAM is evenly distributed across the wafer, ensuring that each core can access its local memory in a single clock cycle. This is orders of magnitude than memory access to off-chip DRAM as found in conventional architectures.

In the following sections, we shall explore how the Cerebras architecture can overcome many of the roadblocks to increasing application performance that challenge traditional HPC clusters. See the appendix for a deeper dive into the ways that the Cerebras architecture can accelerate specific classes of algorithms.

Cerebras vs. Poor Scaling

Many applications have algorithmic scaling issues that arise because of communication. A communication bottleneck often plaguing HPC applications is that which is driven by a local communication, such as with all your nearest neighbors. PDE solvers for highly non-linear problems, such as computational fluid dynamics, and stencil-based solvers are often bottlenecked because of the amount of communication that needs to happen between compute units with neighboring mesh elements. Another communication bottleneck is found with algorithms that rely on global communication, such as all-to-alls or reductions. This bottleneck often affects applications that rely on spectral algorithms and particle simulators which require regular communication between all the computing elements.

Cerebras WSE-2

850,000 cores

46,225 mm2 silicon area

2.6 trillion transistors

40 GB on-chip memory

20 PB/s memory bandwidth

220 Pb/s fabric bandwidth

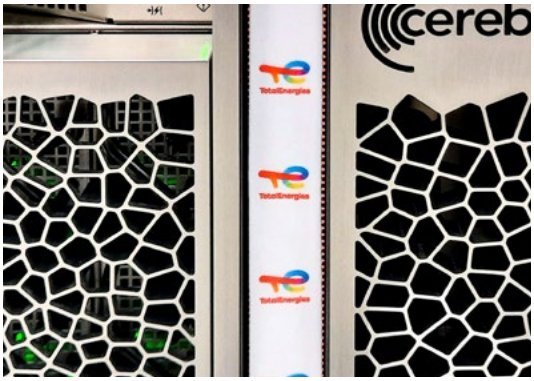
7nm process technology

Figure 3. Characteristics of the Cerebras WSE-2 chip which powers the CS-2 system

The WSE-2 overcomes many communication bottlenecks because of the unique design of the architecture. First, the fabric is built to be high bandwidth and low latency allowing for unparalleled speed for any application. Each core is directly attached to the four neighboring cores, and messages can be sent between them in a single clock cycle. Fine-grained programmability of the routers sending the messages also improves communication efficiency. Rather than consuming processor cycles, each core's integrated router can be pre-configured with various communication commands to pass data as required without intervention. Using the Cerebras Software Language, CSL, data can be sent or received between the router and compute element on each core without entering memory. This can improve computation speed by saving the cycles required to write and then read data back to the processor and can reduce the memory footprint required which allows for even larger simulations to be undertaken.

Applications that can be accelerated due to the high bandwidth and low latency include spectral solvers and particle simulators that rely on regular all-to-all type communications, and partial differential equation (PDE) and iterative solvers for non-linear problems that require regular communication between neighboring compute elements.

An example of the acceleration a CS-2 has to offer due, in part, to the high bandwidth available on the wafer is in the energy industry, where Cerebras has partnered with TotalEnergies to create a seismic solver that is communication-bound on traditional hardware.[2] The Cerebras technology allows for speed-ups of over two orders of magnitude for workflows vital to many energy companies. This speed-up comes from being able to run at a high compute efficiency, where almost no cycles are spent waiting for data to be transferred. The solver becomes "compute-bound" when run on the CS-2, a phrase which is music to the ears of any HPC developer.[3] A detailed tutorial on the optimized 25-point stencil application showcasing adaptive routing and efficient compute methods at the heart of the solver is available in our SDK documentation.



At TotalEnergies, one of the largest energy companies in the world, the CS-2 system outperformed a modern AI GPU by more than 200X on a finite difference seismic modelling benchmark using code written in the Cerebras Software Language (CSL).

Learn more in this blog post[2] and this co-authored paper.[3]

The second way that the Cerebras wafer-scale design allows for codes that were previously communication-bound to accelerate is the sheer size of the WSE. The WSE-2 has 850,000 cores all on the same wafer. While data traversing from one corner diagonally to the other corner does require roughly 2000 cycles to do all the hops from one PE to a neighbor and on, the 2000 cycles is dwarfed by the amount of time conventional hardware would take to pass data across a conventional external network between 850k cores. The communication time incurred during inter-nodal communication on clusters of conventional hardware with NICs and networks is orders of magnitude slower than the self-contained WSE. The locality of the compute allows for the reduction of communication time for almost every type of ML or HPC problem.

Cerebras vs. Slow Data Access

Data access constraints are a bottleneck for many HPC and ML/AI applications. Data access issues are typically caused by two things: 1) not being able to access new data fast enough, and 2) not being able to access the data already loaded in memory quickly. The CS-2 alleviates these problems by design, allowing for far faster computation than with traditional hardware.

The CS-2 system has 1.2 Tb/s of bandwidth onto the system from support nodes using 100 Gbps Ethernet connections. This high bandwidth allows for quick loading and unloading of data for applications that are wafer-resident, where all the data lives on the wafer for the duration of the computation. Additionally, data can be streamed onto and off the wafer quickly as the computation is running. This has the potential to allow for real-time analysis of data on large data sets, such as radar or astronomical data, as the data is being created, or can be used to write out checkpoint or log files without slowing the computation.

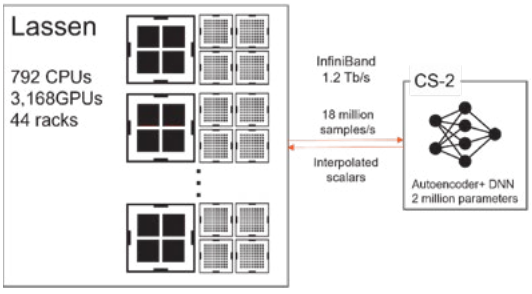


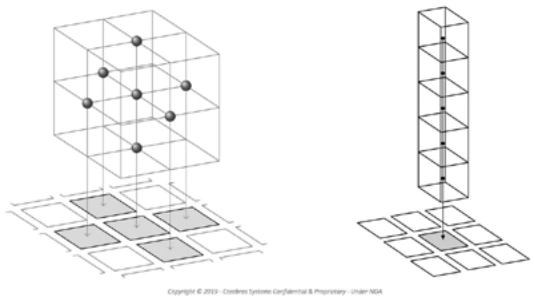
Figure 4. Example of system-level heterogeneity using a Cerebras system in conjunction with the Lassen supercomputer at LLNL.

The high data transfer rates allow for the wafer to be used as a part of complicated workflows. For example, a Cerebras system has been incorporated in the Lassen supercomputer at the National Ignition Facility at Lawrence Livermore National Laboratory (LLNL).[4] Lassen runs a physics package called HYDRA which models a nuclear fusion reaction, and the Cerebras system runs a deep neural network designed to replace a part of the computation that models atomic kinetics and radiation (Figure 4). The ML workload can be called at every time-step of the simulation being run on a large supercomputer because of the high bandwidth between the wafer and the outside compute elements.

Learn more about Cerebras and LLNL's work to blend high performance computing with artificial intelligence here.[4]

The local memory access speed is one of the most impressive aspects of the design of the WSE. Each core can move data from the private SRAM memory pool to the core's processor registers in one cycle. This access speed rivals L1 cache speeds on typical multi-core processors, and is the same for all of the memory on the wafer. Striding or other tricks used to transport data from L2 cache, or slower off-chip memory, into L1 on conventional hardware are thus not required. Computations that are bound by cache size or cache speed are often sped up by orders of magnitude because of the fast access to local memory. Applications that utilize GEMM, GEMV, or sparse matrix operations such as linear algebra solvers, and stencil-based PDE solvers see speed-ups of several orders of magnitude.

An example of how the high Cerebras local memory access speed can accelerate an application is with a computation fluid dynamics (CFD) application developed by the National Energy Technology Laboratory (NETL).[5] The sparse linear equation solver developed by NETL has been shown to accelerate their CFD application by 200 X compared with their in-house Joule 2.0 supercomputer. Additionally, a GEMV implementation can be found within the SDK benchmark suite allowing users to build their own GEMV-based using an optimized implementation as a building block towards their unique applications.



Learn more about Cerebras and NETL's record-setting work in computational fluid dynamics here.[5]

Programming Model

While the Cerebras Software Platform already supports the PyTorch and TensorFlow frameworks, the Cerebras SDK allows programmers to write lower-level code that targets the WSE's microarchitecture directly.[6]

The programmer can write code that targets every core of the wafer such that compute and memory are optimally utilized. Applications can be programmed with a domain-specific programming language called the Cerebras Software Language, or CSL. CSL is similar to C and will thus familiar to most HPC programmers. CSL was built with ease-of-use in mind. We have introduced new, special functionality to easily create code optimized for the wafer while retaining standard syntax and functionality. A programmer can assign many cores to do the exact same thing or give them unique tasks. This flexibility allows for efficient computations by eliminating

idle cycles. The programmability of individual cores allows for SIMD, SISD, MISD or MIMD programming across any portion of the chip that is required.

The CS-2 system is programmed similarly to most accelerators today. A host system is used to load programs on to the CS-2 system for execution. Additionally, Cerebras has created a simulator which can be targeted rather an actual CS-2 system for porting, testing and debugging. Programs intended for the CS-2 system are written in the CSL programming language, which is optimized for dataflow applications. CSL also allows the developer to specify which groups of cores will run which programs. This additional flexibility helps CS-2 system programmers take full advantage of what the platform has to offer.

Learn more about the Cerebras Software Development Kit in this white paper: Cerebras SDK Technical Overview.[6]

Next steps to start your collaboration with Cerebras

If you are curious about programming for wafer-scale or want to evaluate whether the CS-2 system would be a good fit for your application, we encourage you to get in touch at http://www.cerebras.net/hpc. If you do, there are a few data points that will help us answer your questions most effectively: Let us know what languages and development environments you are familiar with, which libraries you rely on, and the known performance bottlenecks that are holding you back in your current compute environment.

Appendix: Cerebras and the Seven Dwarfs

In 2004, Phillip Colella of Berkeley gave a brief presentation called Defining Software Requirements for Scientific Computing where he outlined seven algorithms that were present in most parallel computing applications. These have come to be affectionately known as the "Seven Dwarfs of Symbolic Computation." This appendix offers a deeper dive into how the Cerebras architecture can accelerate each of these algorithms:

1. Sparse Linear Algebra
The CS-2's memory access speeds and fabric bandwidth allow for excellent acceleration for typical sparse linear algebra applications. Having the entirety of memory one cycle from the processors reduces the time for arbitrary memory access by orders of magnitude, and the communication beyond nearest neighbors is able to utilize the on-chip bandwidth rather than do any inter-nodal communication.

2. Dense Linear Algebra
The CS-2's memory access speeds can allow for accelerations of portions of the workflows. Traditional compute hardware may be able to overlap communication and compute for some types of problems, not allowing the low-latency, high bandwidth fabric on the wafer to be advantageous. Applications in which the CS-2 will provide faster time-to-solution than traditional hardware are those which have greater communication needs.

3. Spectral Methods
The WSE-2's fabric, size and memory access speeds make the CS-2 an excellent choice for many spectral applications. The all-to-alls and reductions typical of most spectral methods can use the high-bandwidth and low-latency fabric between the PEs. Furthermore, having access to

850,000 processing elements without having to incur any sort of off-chip, let alone intra-nodal, communication time hit allows for dramatic acceleration of many applications. Additionally, the memory access speeds allow for high use of the processing elements, with no idle cycles waiting on data that is already in memory to be read into the registers.

4. N-body Methods
The communication pattern required for many N-body methods can utilize the WSE-2 fabric's high-bandwidth and low-latency well. The adaptive routing capability can be utilized to have excellent performance for any hierarchical-type communication patterns. Additionally, the ability to read incoming into the registers directly rather than to memory can allow for the compute the requires communicated data to run well. Furthermore, the data that is in memory can be accessed incredibly quickly. The CS-2 is an excellent choice for many N-Body type applications.

5. Structured Grids
The CS-2's fabric, memory access speeds, size and unique architecture can speed up almost every aspect of typical structured grid problems. The high-bandwidth fabric is useful for all communication, especially as stencil sizes grow larger. The adaptive routing can optimize communication times, and the ability to read directly from incoming communication to the registers gives the CS-2 a dramatic boost in speed. Furthermore, the ability to do problems using 850,000 cores without any off-chip latency hit is incredibly useful.

6. Unstructured Grids
The CS-2 can provide increased performance for some unstructured grid applications. Applications that have complicated meshes typically require multiple levels of memory reference indirection, where data access speeds can dominate time-to solution. These applications can be accelerated due to the entirety of the local memory only being a single cycle from the registers. Additionally, applications with complex communication patterns will see reduced time-to-completion as the fabric's bandwidth and latency advantages will reduce communication times.

7. Monte Carlo
Typical Monte Carlo, or embarrassingly parallel, type simulations may see acceleration if there are high levels of data movement between the wafer and host system due to the high bandwidth on/off the CS-2 system. Additionally, the high-speed local memory access may provide speed-up for certain computational kernels being run, and the wafer size of 850,000 cores may provide the raw compute power necessary for some Monte Carlo applications.

References

1 Tiffany Trader, "STREAM Benchmark Author McCalpin Traces System Balance Trends", HPC Wire, 2016, https://www.hpcwire.com/2016/11/07/mccalpin-traces-hpc-system-balance-trends/

2 Rebecca Lewington, "TotalEnergies and Cerebras: Accelerating into a Multi-Energy Future", Cerebras, https://cerebras.net/blog/accelerating-into-a-multi-energy-future/

3 Mathias Jacquelin, Mauricio Araya-Polo, Jie Meng, "Massively scalable stencil algorithm", Submitted to SuperComputing 2022, https://arxiv.org/abs/2204.03775

4 Lawrence Livermore National Laboratory customer spotlight page https://cerebras.net/spotlight-lawrence-livermore-national-laboratory/

5 Kamil Rocki, Dirk Van Essendelft, Ilya Sharapov, Robert Schreiber, Michael Morrison, Vladimir Kibardin, Andrey Portnoy, Jean Francois Dietiker, Madhava Syamlal, Michael James, "Fast Stencil-Code Computation on a Wafer-Scale Processor", 2020, https://arxiv.org/abs/2010.03660

6 Justin Selig, "The Cerebras Software Development Kit: A Technical Overview" https://f.hubspotusercontent30.net/hubfs/8968533/Cerebras SDK Technical Overview White Paper.pdf