SambaNova SN40L: Scaling the AI Memory Wall with Dataflow and Composition of Experts

Raghu Prabhakar, Ram Sivaramakrishnan, Darshan Gandhi, Yun Du, Mingran Wang, Xiangyu Song, Kejie Zhang,
Tianren Gao, Angela Wang, Karen Li, Yongning Sheng, Joshua Brot, Denis Sokolov, Apurv Vivek, Calvin Leung,
Arjun Sabnis, Jiayu Bai, Tuowen Zhao, Mark Gottscho, David Jackson, Mark Luttrell, Manish K. Shah, Edison Chen,
Kaizhao Liang, Swayambhoo Jain, Urmish Thakker, Dawei Huang, Sumti Jairath, Kevin J. Brown, Kunle Olukotun

SambaNova Systems, Inc.
first.last@sambanova.ai

***Abstract*—Monolithic large language models (LLMs) like GPT-4 have paved the way for modern generative AI applications. Training, serving, and maintaining monolithic LLMs at scale, however, remains prohibitively expensive and challenging. The disproportionate increase in compute-to-memory ratio of modern AI accelerators have created a memory wall, necessitating new methods to deploy AI. Recent research has shown that a composition of many smaller expert models, each with several orders of magnitude fewer parameters, can match or exceed the capabilities of monolithic LLMs. *Composition of Experts (CoE)* is a modular approach that lowers the cost and complexity of training and serving. However, this approach presents two key challenges when using conventional hardware: (1) without fused operations, smaller models have lower operational intensity, which makes high utilization more challenging to achieve; and (2) hosting a large number of models can be either prohibitively expensive or slow when dynamically switching between them.**

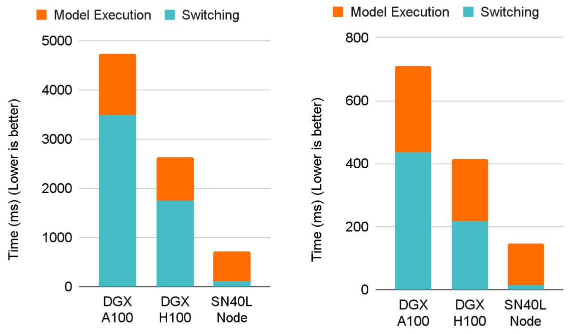
In this paper, we describe how combining CoE, streaming dataflow, and a three-tier memory system scales the AI memory wall. We describe Samba-CoE, a CoE system with 150 experts and a trillion total parameters. We deploy Samba-CoE on the SambaNova SN40L Reconfigurable Dataflow Unit (RDU) – a commercial dataflow accelerator architecture that has been co-designed for enterprise inference and training applications. The chip introduces a new three-tier memory system with on-chip distributed SRAM, on-package HBM, and off-package DDR DRAM. A dedicated inter-RDU network enables scaling up and out over multiple sockets. We demonstrate speedups ranging from 2× to 13× on various benchmarks running on eight RDU sockets compared with an unfused baseline. We show that for CoE inference deployments, the 8-socket RDU Node reduces machine footprint by up to 19×, speeds up model switching time by 15× to 31×, and achieves an overall speedup of 3.7× over a DGX H100 and 6.6× over a DGX A100.

I. Introduction

Recent advances in the training and inference of large language models (LLMs) has taken the world by storm. State-of-the-art generative AI/ML applications like ChatGPT [7] and Gemini [3] are built on top of *monolithic LLMs* that can have billions or trillions of parameters. They are trained with curated datasets that consist of trillions of tokens scraped from the web. However, training and serving a state-of-the-art monolithic LLM is both an extraordinarily expensive affair and a complex systems engineering challenge. Training requires building and operating a supercomputer composed of thousands of hosts, purpose-built networks, power and cooling infrastructure, and thousands of accelerators – typically GPUs [29], [30] or TPUs [46]–[49]. The prohibitive cost and expertise required to train and serve 100s of billions of parameters put state-of-the-art AI capabilities out of reach for many academic researchers and smaller organizations, especially when on-premise deployments are needed. For instance, compute costs to train OpenAI's GPT-4 is estimated to be $78 million USD, and Google's Gemini Ultra to be $191 million USD [57]. Building and deploying large monolithic models may not be sustainable for hyperscalers [14] or any organization needing capable AI models continuously trained and updated on their data [1], [10], [16]. Finally, systems that cater to monolothic models have scaled compute TFLOPs much faster than memory bandwidth and capacity, creating the memory wall [35] where the memory system can no longer feed the compute efficiently.

The ML research community has responded with ecosystems of much smaller, modular models that are just as capable, but are cheaper and easier to train and serve [5], [37], [43], [58]. Smaller models like the 7B-parameter Llama 3 [13], Llama 2 [68], and Mistral 7B [44] are often adequate. They might not match the performance of larger models over a *general* suite of tasks, but smaller models can deliver superior accuracy on a narrower set of *specialized* tasks for which they have been fine-tuned [8], [65]. For example, Flan-T5-XL only has 3B parameters, but it surpasses the 175B-parameter GPT-3's MMLU score by nearly 10% [31]. Proof points like these have bolstered community activity in building and training smaller models by specializing base models to a domain, by fine-tuning base models to a specific task or group of tasks [66], [69], and by distilling or compressing larger models into smaller models. Furthermore, *compositions* of such smaller models have been shown to demonstrate emergent behavior that matches large monolithic models [38], [45], [51], [55], [56]. They bring AI within reach to a broader community.

We believe that successful AI systems of the future will host and execute many small models efficiently. This is reflected both in directions pursued successfully in academia [5], [38], [45], [51], [55], and new products that are being adopted in



(a) BS=8, TP=8

(b) BS=1, TP=8

Fig. 1: CoE latency breakdown between model switching and model execution to generate 20 output tokens from a Llama2-7B expert The SN40L RDU executes CoEs efficiently by combining streaming dataflow and a novel three-tier memory hierarchy of SRAM, HBM, and DDR.

industry [4], [6], [24]. CoE-like compound AI systems play a pivotal role in advancing the AI frontier [60], [72]. In this paper, we refer to such modular systems with compositions of specialized smaller models as ***Composition of Experts (CoE)***.

A CoE consists of several small expert models working in tandem on a task. Outputs from one expert determine which expert(s) to execute next. Running an expert involves loading model parameter weights to the accelerator's main memory, and then executing the model. Consequently, executing a CoE involves a sequence of model switching and model execution. Current state-of-the-art AI accelerators do not handle this sequence of operations efficiently, as shown in Figure 1.

Efficiently accelerating a *CoE* boils down to executing expert models efficiently while minimizing model switching costs. We break this down into three key requirements:

1) ***Aggressive*** **Operator Fusion and Pipeline Parallelism** to execute expert models efficiently. Smaller models have lower operational intensity [67], [70], [74] and complex access patterns between operators [34]. Conventional operator fusion techniques [22], [26], [40] achieve limited success across arbitrary access patterns.
2) **High-Bandwidth Memory** to exploit temporal and spatial locality in weights and intermediate results during generative inference, and
3) **High-Capacity Memory** to minimize switching costs and store the parameters of many expert models

In this paper, we describe a hardware/software solution that overcomes the memory wall by addressing the challenges above.

We first describe the **Samba-CoE**, a trillion parameter CoE system with 150 7B expert models, and how running it efficiently requires hardware support for aggressive operator fusion and a novel memory system. We present the **SambaNova SN40L Reconfigurable Dataflow Unit (RDU)**, a commercial dataflow accelerator that combines **streaming dataflow parallelism** with a novel **three-tier memory system** containing large on-chip SRAM, HBM, and DDR DRAM that is directly attached to the accelerator.

The RDU's streaming dataflow architecture allows us to fuse *hundreds* of complex operations with arbitrary access patterns into a single kernel call – without requiring the programmer to write that kernel by hand. This delivers large speedups by exploiting on-chip hardware support for mixtures of pipeline, data, and tensor parallelism. Our aggressive fusion techniques are well beyond the capabilities of state-of-the-art techniques used with conventional architectures [22], [26], [40], [74].

Fabricated using TSMC 5nm technology, the SN40L RDU is a 2.5D Chip-on-Wafer-on-Substrate (CoWoS) chiplet-based design containing two SN40L Reconfigurable Dataflow Dies (RDDs) and HBM. Each SN40L RDU socket has 638 BF16 TFLOPS of peak compute performance using 1040 distributed Pattern Compute Units (PCUs). These are complemented by 1040 distributed Pattern Memory Units (PMUs) that in aggregate provide hundreds of TBps of on-chip memory bandwidth along with high bank-level parallelism within and across PMUs. Flexible on-chip address generation logic provides high bandwidth for arbitrary tensor memory access patterns. The three memory tiers in SN40L are: 520 MiB of on-chip PMU SRAM, 64 GiB of co-packaged HBM, and up to 1.5 TiB of DDR DRAM (using pluggable DIMMs). Models are loaded from DDR to HBM at over 1 TB/s in a single SN40L Node.

We quantify and discuss the impact of streaming dataflow parallelism on several real world benchmarks, showing speedups ranging from **2**× to **13**× over an optimized baseline. We deploy Samba-CoE on a single *SN40L Node* that contains eight SN40L RDU sockets and a host. We discuss the performance of Samba-CoE on the SN40L Node compared to DGX A100 and DGX H100. We show that for CoE inference deployments, the SN40L reduces machine footprint by up to **19**×, speeds up model switching time by **15**× to **31**×, and achieves an overall speedup of **3.7**× to **6.6**× over DGX H100 and DGX A100, respectively.

This paper is organized as follows: Section II describes Samba-CoE, our trillion parameter CoE. Section III describes streaming dataflow and its challenges that translate to key hardware requirements. Section IV describes the SN40L hardware architecture in detail and lists key changes from prior RDUs [63], [64]. Section V describes the software support for managing DDR and HBM. Section VI quantifies the benefits of streaming dataflow as well as the performance of Samba-CoE. Section VII discusses key learnings from the hardware/software codesign process. Section VIII covers related work. We conclude in Section IX.

II. Background: Composition of Experts

In this section, we describe one instance of a CoE built and deployed on the SN40L, called ***Samba-CoE***. Figure 2 shows the Samba-CoE pipeline from prompt to response.

Samba-CoE consists of several expert models and a router model. Each expert is fine-tuned in a specific domain. We leveraged several excellent expert models fine-tuned on