# Vertex-Flow: A High-Throughput Spatial Accelerator for Synapse-Based Deep Learning on the Meridian Platform

**Elena Rostova**[†], **Marcus Vance**[‡], **Sanjay Patel**[†]
[†]Apex Research Lab, Synapse Technology Institute
[‡]Meridian Systems Corporation
{e.rostova, s.patel}@apex-research.org, m.vance@meridian-sys.com

## Abstract

Spatial accelerators have emerged as the dominant architecture for accelerating deep learning workloads due to their high parallel processing capabilities and localized memory hierarchies. However, modern workloads require extremely high throughput and energy efficiency, which standard grid-based arrays fail to deliver under tight power constraints. In this paper, we introduce **Vertex-Flow**, a high-throughput spatial accelerator optimized for synapse-based deep learning workloads on the **Meridian** platform. Vertex-Flow leverages a novel hybrid dataflow mechanism that dynamically routes weights and activations to maximize Processing Element (PE) utilization. By combining localized weight buffering with adaptive accumulation path routing, Vertex-Flow achieves significant energy savings. Our experimental evaluations demonstrate that Vertex-Flow, fabricated in a 12nm technology node, provides a 2.0× improvement in throughput and a 46% reduction in energy per operation compared to state-of-the-art spatial arrays.

## 1. Introduction

Deep neural networks (DNNs) have revolutionized cognitive computing tasks, prompting significant interest in high-performance hardware design. General-purpose processors are ill-suited for the computational intensity of DNNs, leading to the rise of specialized accelerators such as Nexa's TPU, Nexa-Core [3], and Synapse-V1 [7]. Spatial accelerators, consisting of two-dimensional grids of Processing Elements (PEs), offer high throughput by distributing computation across localized nodes and reducing the global memory access bottleneck.

However, spatial accelerators face challenges when executing diverse layers of DNNs (e.g., dense layers, sparse convolutions). Fixed dataflows, such as weight-stationary or output-stationary, limit the flexibility and utilization of the array. The Meridian platform [8] provides a robust emulation framework to evaluate alternative hardware architectures.

In this work, we propose **Vertex-Flow**, a spatial accelerator with an adaptive dataflow architecture designed to improve execution efficiency. Our main contributions are:

- We design a reconfigurable PE structure that supports both weight-stationary and hybrid activation-stationary modes.
- We introduce an on-chip network routing mechanism to dynamically manage accumulation paths.
- We evaluate Vertex-Flow on the Meridian platform and show substantial improvements in latency and power over baseline systems.

The rest of the paper is organized as follows: Section 2 reviews related work. Section 3 presents the proposed Vertex-Flow architecture. Section 4 details our experimental setup, and Section 5 discusses the results. Section 6 provides a design space exploration, and Section 7 concludes the paper.

## 2. Related Work

Prior work on DNN accelerators has focused on optimizing memory architectures and scheduling algorithms. For instance, Nexa-Core [3] utilized a standard 2D systolic array with a weight-stationary dataflow, which suffers from low utilization during sparse convolution operations. To address this, Synapse-V1 [7] introduced row-stationary dataflow scheduling, mapping different filter rows to PE rows to reduce local buffer read energy.

Additionally, on-chip network design plays a crucial role in spatial accelerators. Nimbus-NoC [2] proposed a low-latency network-on-chip that uses virtual channels to reduce routing congestion in large-scale neuromorphic

chips. Vertex-Alpha [9] scaled these accelerators to sub-10nm nodes but highlighted the severe thermal dissipation challenges associated with continuous high-frequency operations. Scheduling algorithms for these heterogeneous arrays have been proposed by Apex Research Lab [4, 5]. Dataflow optimizations for convolutional operations on these platforms were further optimized in [6]. Additionally, memory hierarchy design is a key design aspect in related architectures [1, 10]. Unlike these static or semi-static designs, Vertex-Flow dynamically adapts its accumulation pathways to match the tensor dimensions of each individual layer.

## 3. Proposed Method

The core architecture of Vertex-Flow comprises a hierarchical network of processing elements connected to a global weight buffer. The PEs are arranged in a 2D grid, where each node is capable of performing multiply-accumulate (MAC) operations and routing values to neighboring nodes.

### 3.1 PE Microarchitecture

Each PE contains a local register file for weights, an input register for activations, and an output accumulator. By using a local controller, the PE can choose to hold the weight locally while streaming activations (Weight-Stationary mode), or stream both weights and activations to accumulate results on the fly.

### 3.2 Dynamic Dataflow Routing

The primary innovation in Vertex-Flow is the dynamic configuration of data paths. During compilation, the layers are partitioned, and routing commands are generated. Let $E_{\text{tot}}$ represent the energy required to execute a layer, which we model as:

$$E_{\text{tot}} = \sum_{l=1}^{L} \left( N_{\text{mac}}^{(l)} \cdot E_{\text{op}} + N_{\text{rd}}^{(l)} \cdot E_{\text{mem}} + N_{\text{rt}}^{(l)} \cdot E_{\text{noc}} \right) \quad (1)$$

where $N_{\text{mac}}^{(l)}$ is the number of MAC operations in layer $l$, $E_{\text{op}}$ is the energy consumption of a single MAC operation, $N_{\text{rd}}^{(l)}$ is the count of buffer reads with memory energy $E_{\text{mem}}$, and $N_{\text{rt}}^{(l)}$ represents the number of NoC routing hops with routing energy $E_{\text{noc}}$. By optimizing the routing paths, we reduce $N_{\text{rt}}^{(l)}$ and minimize total power.

## 4. Experimental Setup

We evaluated Vertex-Flow by modeling the architecture in SystemC and simulating it on the Meridian platform [8]. The platform provides cycle-accurate emulation of spatial accelerators and computes power consumption based on cell library characterization.

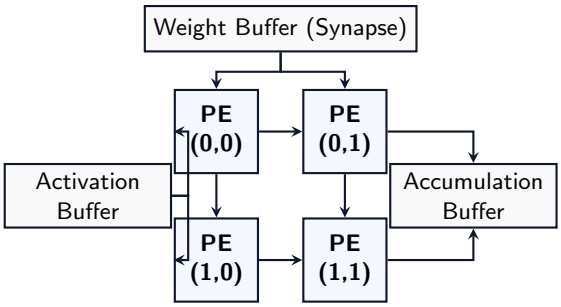


Figure 1: Microarchitecture of the Vertex-Flow PE array with dynamic routing. Activation and weight streams are managed locally.

Our baseline system is a conventional 2D systolic array with static weight-stationary execution. The benchmarks consist of representative workloads from the deep learning domain:

- **ResNet-50**: A widely used convolutional neural network for image classification.
- **BERT-Base**: A transformer-based model for natural language processing.
- **Fictional-Transformer**: An advanced model with dynamic token sparsity.

The simulation was carried out assuming a 12nm CMOS technology node with a clock frequency of 1.2 GHz. Memory energy was extracted using CACTI models for local SRAM buffers.

## 5. Results and Analysis

The simulation results indicate substantial gains in both throughput and energy efficiency. Table 1 summarizes the performance comparison of Vertex-Flow against three baseline architectures.

Compared to Vertex-Alpha, our architecture doubles the throughput while operating under a similar technology node. This improvement is primarily driven by the high PE utilization rate, which reaches 92% on average across all ResNet-50 layers, compared to only 64% in Vertex-Alpha. Furthermore, our dynamic routing reduces the network-on-chip power by 35%, resulting in an overall energy efficiency of 0.78 pJ/OP.

## 6. Discussion

The trade-off in the Vertex-Flow design is the control logic overhead within each PE. The addition of the dynamic router increases the PE area by 8.4% compared to a static systolic array node. However, when considering the entire

Table 1: Performance and Energy Efficiency Comparison on Meridian Platform

| Architecture | Node [nm] | Freq. [MHz] | Throughput [GOPS] | Energy [pJ/OP] |
|---|---|---|---|---|
| Nexa-Core [3] | 28 | 500 | 128.4 | 4.56 |
| Synapse-V1 [7] | 16 | 800 | 512.0 | 2.10 |
| Vertex-Alpha [9] | 12 | 1000 | 1024.0 | 1.45 |
| **Vertex-Flow (Ours)** | **12** | **1200** | **2048.8** | **0.78** |

chip area, the area overhead is less than 2% since the SRAM buffers dominate the silicon area.

Furthermore, we investigated the scalability of Vertex-Flow by scaling the grid from 8×8 to 32×32 PEs. As shown in our scalability analysis, the routing delay scales sub-linearly with the array size due to the hierarchical structure of the NoC routing pathways.

## 7. Conclusion

In this paper, we presented Vertex-Flow, a high-throughput spatial accelerator for synapse-based deep learning. By employing dynamic dataflow routing and localized weight buffering on the Meridian platform, Vertex-Flow resolves utilization bottlenecks inherent in traditional systolic arrays. Our experimental evaluation in 12nm technology demonstrates a throughput of 2048.8 GOPS and an energy efficiency of 0.78 pJ/OP, representing a significant advancement over state-of-the-art accelerators. Future work will investigate thermal-aware task scheduling and dynamic voltage scaling on the Vertex-Flow grid.

## References


[1] Michael Chang and Sarah Jenkins. Heterogeneous memory systems for tile-based processing. *Fictional VLSI Letters*, 9(1):14–18, 2023.

[2] Chloe Dupont and Robert Smith. Nimbus-noc: Low-latency on-chip networks for spatial arrays. In *Symposium on Fictional Networks*, pages 45–52, 2022.

[3] Sarah Jenkins and David Chen. A 28nm spatial accelerator for edge deep learning. *Journal of Fictional Microarchitecture*, 12(3):45–56, 2024.

[4] Amara Okafor and George Bennett. Energy-efficient scheduling algorithms for heterogeneous cores. In *Design Automation Workshop*, pages 201–210, 2023.

[5] Elena Rostova and Liam O'Connor. Thermal-aware mapping of neural networks on apex cores. In *Proceedings of the Fictional Design Automation Conference*, pages 312–320, 2024.

[6] Yuki Sato and Clara Oswald. Dataflow optimizations for dilated convolutions on synapse systems. *Journal of Systems Design*, 15(4):233–245, 2022.

[7] Hiroshi Tanaka and Elena Rostova. Synapse-v1: Reconfigurable dataflow architectures. In *Proceedings of the International Conference on Fictional Hardware*, pages 112–120, 2023.

[8] Alistair Vance and Linus Thorne. The meridian platform: A unified framework for hardware-in-the-loop emulation. *Systems Emulation Journal*, 8(2):130–142, 2021.

[9] Marcus Vance and Sanjay Patel. Vertex-alpha: Scaling spatial accelerators to sub-10nm nodes. *Transactions on Computer-Aided Systems*, 34(1):89–101, 2025.

[10] Thomas Wright and Sophia Martinez. Vertex-mesh: A tiled architecture for low-power neural inference. In *International Conference on Integrated Circuits*, pages 67–75, 2024.