Research Article | Vol. 34, No. 19 / 21 Sep 2026 / Optics Express 36260
Optics EXPRESS

Optical processing units vs. GPUs and TPUs: conditions for performance and energy advantage

EGOR O. KIMLICHENKO, DMITRII S. BATANOV, DMITRY A. IVANOV, KONSTANTIN S. DOMARATSKIY, NIKITA S. SHMYRIN, EVGENIA E. SOLOKHINA, LILIANA R. BAYSLANOVA, ARTEM V. CHETVERTUKHIN, ANDREY A. GRUNIN, AND ANDREY A. FEDYANIN*

Lomonosov Moscow State University, Moscow 119991, Russia
fedyanin@nanolab.phys.msu.ru

Abstract: The conditions under which optical processing units (OPUs) can outperform conventional electronic accelerators, such as graphics processing units (GPUs) and tensor processing units (TPUs), are investigated. Because integrated system-level implementations remain unavailable, experimental validation is infeasible; a modeling-driven approach is therefore adopted to identify favorable configurations prior to fabrication. An analytical design space exploration framework is developed that extends roofline-based latency modeling to photonic cores and identifies total computational parallelism—core size squared times core count—as the governing parameter of OPU competitiveness. The latency model, validated against representative GPU and TPU measurements, enables efficient exploration of key variables, including core dimensionality, modulation frequency, and numerical precision. Optical loss, modulation bandwidth, and signal-to-noise limitations of integrated photonic platforms constrain the accessible design space. Physically realizable parameter regimes are identified in which OPUs achieve performance and energy-efficiency parity with, or advantage over, their electronic counterparts. These results establish the conditions under which photonic computing can provide a quantifiable benefit, clarify the corresponding architectural constraints, and yield concrete specifications for subsequent high-fidelity simulation and prototyping.

1. Introduction

One of the main challenges of AI today is limited computational resources, as training state-of-the-art models can take weeks [1] and the emergence of reasoning models further increases overall computational demand, particularly for inference [2]. Furthermore, model performance improves with increasing number of training parameters [3], meaning that further progress requires ever-growing computational power. A fundamental operation underlying these computational demands, particularly in transformer architectures, is matrix-vector multiplication [4].

Typically, AI computations are performed on graphics processing units (GPUs) =, known for their ability to execute parallel computations. However, the energy efficiency of such processors is suboptimal [5]. To address this issue, tensor processing units (TPUs) [6] and ASIC solutions, such as Eyeriss [7,8] have also been introduced. Moreover, neuromorphic photonic solutions have emerged, offering higher performance per watt [9–11].

Due to its high carrier frequency and low absorption in waveguides, photonics offers significant potential for accelerating these computations while maintaining high energy efficiency [12]. A substantial amount of research is directed towards the development of optical accelerators (OPUs) for neural networks, designed to perform matrix-vector multiplication [9,13–15]. Despite these advantages, practical implementations must address constraints imposed by memory access

#604976 https://doi.org/10.1364/OE.604976
Journal © 2026 Received 12 May 2026; revised 5 Aug 2026; accepted 28 Aug 2026; published 16 Sep 2026

Optics EXPRESS

speeds [16] and the analog-digital interface complexity [17]. Therefore, determining the critical parameters of the photonic accelerator and computational tasks required to match or surpass classical hardware is essential [18].

One solution is to simulate optoelectronic components and optical computing processes [19,20]. While this approach can be highly accurate, a full development cycle is resource-intensive. Moreover, using simulators for comprehensive parameter sweeps requires substantial computational time, which complicates broad-scale design space exploration (DSE). Furthermore, such simulators often limit DSE as they are typically tied to a specific task and architecture.

Therefore, a tool for rapid and comprehensive DSE is needed. The present work aims to provide an analytical approach to identify the parameter domains of vector-matrix multiplication (e.g., matrix size, batch size) and the specific parameters of photonic devices through which photonics can outperform electronics in terms of latency. Once the promising parameter domains are identified, more computationally demanding and accurate simulators can be utilized to derive the precise OPU specifications.

2. Basic principles

Performance analysis of optical computing systems and their comparison with electronic counterparts can be systematically approached using two key dimensions: time and energy. Latency serves as the primary temporal metric, while power consumption represents the fundamental energy metric for these computing units. These values serve as the foundation for deriving combined performance metrics such as peak TOPS and TOPS/W. We use a two-stage analytical approach. First, architecture-independent latency analysis establishes parameter domains that are promising in terms of computational time (Section 3). Second, architecture-specific physics-based models constrain these domains to practically feasible regions (Sections 4 and 5).

As the baseline task for determining latency, we consider the time T required by the computational unit to perform matrix-vector multiplication. Our latency model specifically targets the computational pipeline segment from DRAM-resident prepared data through on-chip caching and compute operations to device output. The analysis assumes preprocessed inputs and compiled computational graphs. Such scope enables rapid design space exploration while avoiding the complexity of full system simulation. This model adopts an approach similar to the roofline model [21], where task execution time is determined by the dominant bottleneck: either memory-bound limitations when data transfer rates constrain performance, or compute-bound limitations when processing capacity becomes the restricting factor.

We deliberately simplify memory subsystem modeling — assuming ideal SRAM-to-core bandwidth and omitting detailed cache hierarchy — to enable traversal of thousands of configurations. This trade-off reduces accuracy in memory-bound regimes but preserves validity precisely where photonic advantages emerge: compute-bound operation at large batch sizes ($B_s \geq 1024$).

To enable comprehensive comparative analysis, we develop computational models for GPU, TPU, and OPU architectures. The GPU and TPU models follow well-established analytical approaches and serve primarily as a consistent comparison baseline; experimental validation confirms their accuracy. The principal contribution is extending this modeling framework to OPU architectures, where experimental validation remains infeasible due to the lack of accessible optical processing devices. Nevertheless, the architectural similarities between TPUs and OPUs — both employing large matrix engines with deterministic dataflow — suggest that the analytical principles validated on TPU hardware can be meaningfully extrapolated to OPU predictions, though this does not constitute complete verification.

This framework is fully applicable to the two waveguide-based architectures — the SVD-Clements and the MZI-based crossbar. The latency analysis (Section 3) depends only on the core size N, the core count C, and the assumption of one vector-matrix multiplication per

Optics EXPRESS

cycle, so the N^2C scaling applies to any weight-stationary architecture, independently of the underlying hardware. The physical-feasibility layers, by contrast, are specific to MZI-based waveguide cores: the optical-loss analysis (Section 4) assumes per-node loss accumulation, and the power model (Section 5) is built around phase-shifter and converter energetics. Alternative optical computing approaches relate to this scope unevenly. Thin-film lithium niobate cores [22] retain the same topology and enter mainly as revised parameter values — higher-bandwidth, lower-loss modulation. Microring-resonator and wavelength-multiplexed tensor cores [23] encode parallelism spectrally, adding per-ring and WDM-routing losses on top of the MZI weighting. Free-space routes — reconfigurable metasurfaces [24] and coherent photoelectric architectures [25] — abandon waveguide propagation and the SINAD chain of Section 4; their timing is set by free-space encoding and readout, so they fall outside the present formulation.

3. Latency model overview

3.1. Model parameters

The latency model operates on a set of computational and architectural parameters that quantify execution time T. Specifically, we examine the multiplication of M square matrices of dimension $V_s \times V_s$ by B_s vectors, each of dimension V_s.

A unifying parameter across all accelerators is the number of processor cores, denoted by C, each containing a computational unit of characteristic size N. For GPUs, a core corresponds to a streaming multiprocessor sub partition (SMSP) [30], with N denoting the number of CUDA cores within it; in modern Nvidia architectures such as the H100, each SMSP integrates one tensor core of dimensions $N_1 \times N_2 \times N_3$, so C equals both the number of SMSPs and tensor cores [26]. For TPUs and OPUs, a core is a systolic array or optical core of size $N \times N$, and C is the number of such arrays or cores.

The frequency parameter f represents the operational clock frequency for all devices. For OPUs, we additionally introduce f_{mat} to represent the matrix modulation frequency, as optical architectures require separate modulation of vector inputs (f) and matrix weights (f_{mat}). To characterize memory operations, we introduce W (peak memory bandwidth), D (operand bit-width), D_{acc} (accumulation bit-width), and S (total buffer size).

We use Llama 3.1–70B's [1] internal dimensionality, training sequence length, and the number of SwiGLU feedforward projections as reference values for vector size, batch size, and matrix count, respectively. See Table 1 for all model parameters.

Table 1. Hardware parameters for GPU, TPU, and OPU accelerators[a]

Device	C	N	W (GB/s)	f/f_{mat} (MHz)	$N_1 \times N_2 \times N_3$	S (MB)	D (B)
H100 SXM5 [26]	528	32	3350	1980/–	16×4×8	50	1, 2
TPU v5p [27–29]	8	128	2765	1750/–	–	128	1, 2
OPU (ADEPT) [10]	1	128	~2765	10000/100	–	400	1

[a]Workload parameters from Llama 3.1–70B [1]: M = 240 (SwiGLU projections), B_s = 16384 (training seq. length), V_s = 8192 (hidden dimension). D_{acc} = 4 B for all devices. OPU W assumed equal to TPU W.

3.2. Roofline-based approach

The roofline model assumes that memory transfers and computation overlap via double buffering, so total latency is determined by the dominant bottleneck:

$$T = \max(T_{\text{compute}}, T_{\text{memory}}), \tag{1}$$

where T_{memory} represents HBM-SRAM transfer time, the primary memory subsystem bottleneck in modern AI accelerators [31].

Optics EXPRESS

The model assumes a weight-stationary dataflow [32]. Because weight matrices typically exceed SRAM capacity, they must be loaded in tiles of r complete rows. After each tile is loaded, the entire batch of input vectors is streamed through and multiplied by those r rows. Once the batch is exhausted, the next r rows replace the previous ones in SRAM, and the batch must be re-read from HBM. The total number of such re-reads is $\rho = \lceil V_s/r \rceil$. A larger r reduces HBM traffic, but sufficient SRAM must remain for input vectors and result tiles to keep the compute units continuously fed. To balance these two requirements, we partition the SRAM into four equal quarters: two for double buffering, one for matrix rows, and one for input vectors and result tiles. The number of rows that fit in one quarter is:

$$r = \left\lfloor \frac{S/4}{V_s D} \right\rfloor. \tag{2}$$

Based on these assumptions, the total HBM-SRAM-HBM transfer time for one matrix-batch pair equals:

$$T_{\text{memory}} = \frac{V_s^2 D + \rho B_s V_s D + B_s V_s D_{\text{acc}}}{W}. \tag{3}$$

Here $V_s^2 D$ represents matrix transfer traffic, $\rho B_s V_s D$ corresponds to input batch traffic, and $B_s V_s D_{acc}$ represents output result traffic.

3.3. Compute model

3.3.1. GPU CUDA model

Each CUDA core performs a single fused multiply-accumulate (FMA) operation $a \cdot b + c$ per clock cycle [26]. For a matrix-vector multiplication of size $V_s \times V_s$, computing each resulting element requires V_s multiplications and $V_s - 1$ additions. At scales where V_s is on the order of 10^3, the subtraction of one is negligible. The model assumes that a single SMSP computes one element of the result vector, with operations distributed uniformly across its N CUDA cores — each core performs V_s/N FMA operations, requiring V_s/N cycles. The computation of an entire result vector by one SMSP thus demands $V_s \cdot (V_s/N) = V_s^2/N$ cycles.

The model represents the time to perform B_s matrix-vector multiplications in parallel across all C SMSPs with matrices processed sequentially. The computation time for processing a single matrix is defined as follows:

$$T_{\text{compute}} = \frac{V_s^2 B_s}{NCf}. \tag{4}$$

Equation (4) is derived for FP32 operations. Since each CUDA core processes one FP32 FMA per cycle but can pack two FP16 operations into the same datapath width, FP16 computation effectively doubles the throughput; the compute time is therefore halved.

3.3.2. GPU tensor cores model

A tensor core of dimensions $N_1 \times N_2 \times N_3$ can multiply two matrices of sizes $N_1 \times N_2$ and $N_2 \times N_3$, adding a third matrix of size $N_1 \times N_3$ to the result - all within a single clock cycle [26,33]. Analytically, a tensor core can be viewed as comprising N_3 independent subcores, each capable of multiplying an $N_1 \times N_2$ matrix block by a vector of length N_2 per cycle.

When matrix dimensions exceed the tensor core size, block partitioning is employed. A matrix of size $V_s \times V_s$ is divided into $\lceil V_s/N_1 \rceil \times \lceil V_s/N_2 \rceil$ blocks, requiring that many block-vector multiplications per output vector. With C tensor cores operating in parallel, each providing N_3 subcores, the system processes $N_3 \cdot C$ vector blocks from the batch simultaneously. The total

Optics EXPRESS

execution time for one matrix-batch multiplication is:

$$T_{\text{compute}} = \frac{(\lceil V_s/N_1 \rceil)(\lceil V_s/N_2 \rceil)}{fN_3C} B_s, \tag{5}$$

where $\lceil \cdot \rceil$ represents rounding up. Equation (5) is derived for FP16/BF16 operations. By the same datapath-packing argument, FP8/INT8 halves and INT4 quarters the compute time.

3.3.3. TPU model

The TPU employs a systolic array architecture [34], where processing elements (PEs) are arranged in an $N \times N$ grid. While various dataflow strategies exist (weight-stationary, output-stationary, row-stationary), our latency model abstracts these details: a matrix operation on B_s vectors completes in B_s pipelined cycles [6], yielding a throughput of one vector-matrix multiplication per cycle. When matrix dimensions exceed the array size, block partitioning applies with $(\lceil V_s/N \rceil)^2$ block multiplications required. The model assumes uniform batch distribution across all C cores:

$$T_{\text{compute}} = \frac{B_s}{Cf} (\lceil V_s/N \rceil)^2. \tag{6}$$

Analogous to the tensor cores formulation, when using lower precision computations, the computation time should be divided by 2 or 4 for INT8 and INT4, respectively.

3.3.4. OPU model

The OPU model assumes that a single vector-matrix multiplication takes one cycle, which can be implemented using crossbar and SVD architectures [9,14]. The execution time comprises two components: the time for modulating input vectors at frequency f, and the time for matrix weight updates at frequency f_{mat}. During matrix reconfiguration, computation is paused, which motivates the additive term $1/f_{\text{mat}}$ in the model.

$$T_{\text{compute}} = \left(\frac{B_s}{Cf} + \frac{1}{f_{\text{mat}}} \right) (\lceil V_s/N \rceil)^2. \tag{7}$$

Equation (7) captures parallel processing of a single matrix with batch uniformly distributed across cores, as well as the matrix modulation overhead.

3.4. Latency model validation

We validated the model on Nvidia H100 PCIe and TPU v5e hardware (Table 2), varying batch size (8–32768) for FP32 (CUDA cores), FP16 (tensor cores), and INT8 (TPU) operands at a fixed vector size of 16384. Each configuration was measured 50 times with randomly generated matrices and vectors; results were averaged. Validation extends to our target SXM and v5p variants, which share identical microarchitectures.

Table 2. Hardware parameters for validation

Device	**C**	**N**	**S** (MB)	**W** (GB/s)	**f** (MHz)
Nvidia H100 PCIe [26]	456	32	50	2000	1755
Google TPU v5e [35,36]	4	128	128	819	1500

Figure 1 presents the GPU results. For CUDA cores, prediction errors remain below 5% for batch sizes from 256 onward. The tensor core model exhibits larger discrepancies (~20% for $B_s \geq 1024$), attributable to simplified cache modeling and omitted proprietary NVIDIA optimizations. These error levels are acceptable for rapid design space exploration rather than cycle-accurate prediction.

Optics EXPRESS

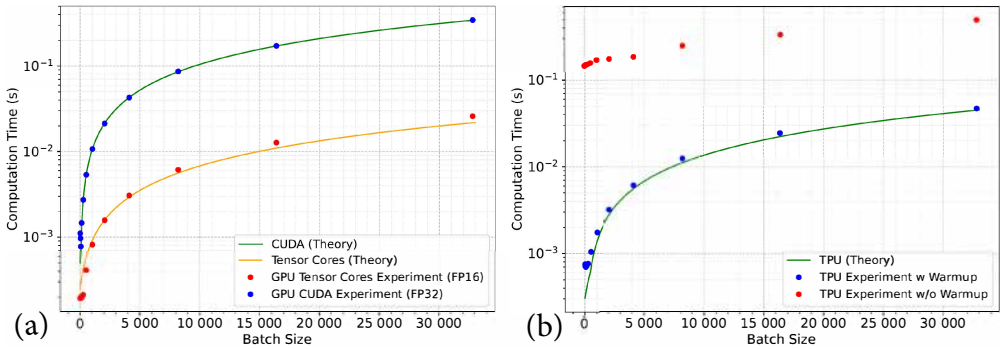

Fig. 1. Theoretical and experimental time of matrix-batch multiplication: (a) Nvidia H100 CUDA cores (FP32) and tensor cores (FP16); (b) Google TPU v5e (int8), warmup improves agreement.

For the TPU v5e (Fig. 1), predictions without warmup diverge by up to 5×. With warmup — a preliminary multiplication that completes preparatory operations in advance — the error reduces to below 25% for $B_s \geq 1024$. A possible reason for the discrepancy is that preprocessing before feeding the input into the systolic array is not accounted for in the model; without warmup, these preparatory operations are included in the measured runtime. Although warmup is an artificial scenario, it represents the TPU's peak achievable performance, establishing a more demanding benchmark for photonic accelerators. An inflection point at $B_s = 545$ marks the transition from memory-bound to compute-bound regime.

Independent vector-size sweeps over $V_s = 16$–32768 at fixed batch size ($B_s = 16384$) confirm model accuracy at the Llama 3.1–70B reference point ($V_s = 8192$): prediction error remains below 15% for the TPU v5e, with agreement improving at larger V_s where the workload better approaches the peak-performance regime captured by the model.

3.5. Design space exploration

We now use the latency model to identify photonic accelerator configurations that can match or outperform electronic counterparts.

Hypothesis 1: *A photonic core configuration can achieve performance parity or superiority compared to NVIDIA H100 GPU and Google TPU v5p operating in both int8 and int4 regimes.* We begin with this hypothesis because it remains independent of specific photonic architectures. This choice of bit depth for comparison is motivated by the typical precision values of photonic devices [10,37]. The computational bit depth of the photonic device will be addressed in Section 4. As a numerical measure of superiority, we understand a 10× improvement in latency. Our methodology prioritizes throughput maximization while imposing constraints on photonic core size, since larger cores exhibit increased optical losses that degrade effective computational bit depth.

We first examine vector modulation frequency f, which directly determines the rate at which input vectors are processed. As demonstrated in Table 3, increasing f from 1 GHz to 10 GHz reduces the required core size by approximately 46% for achieving performance parity with electronic accelerators. Based on this analysis and considering current electro-optical modulator capabilities [38], we select $f = 10$ GHz for subsequent analysis, which has been established as the performance target in contemporary photonic computing research [10,39].

Next, we analyze the impact of matrix modulation frequency f_{mat} on system performance. Matrix weight updates introduce computational stalls, as the processor must pause vector operations during reconfiguration. Figure 2 illustrates this effect, showing the TPU-to-OPU

Optics EXPRESS

Table 3. Core size N for 1 photonic core to achieve parity with the H100 and TPU v5p at different modulation frequencies (f_{mat} = 1 MHz, B_s = 4096, V_s = 8192)

Device	1 GHz	5 GHz	10 GHz
H100 (CUDA, FP16)	274	135	106
H100 (Tensor, FP8/INT8)	1171	713	631
TPU v5p (INT8)	781	456	400

computation time ratio for f_{mat} = 1 MHz — a conservative estimate for MEMS phase shifters [38]. The analysis reveals that achieving performance parity requires either large core sizes or large batch sizes to amortize the reconfiguration overhead.

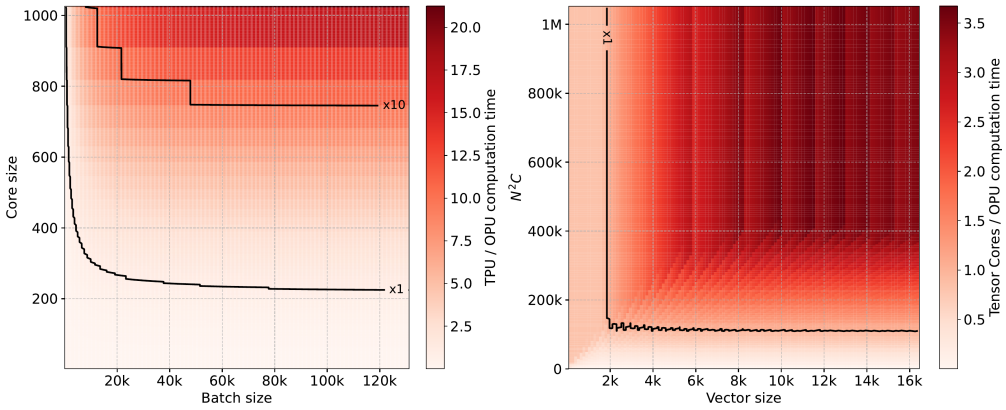

Fig. 2. Electronics-to-OPU computation time ratio. The redder the color, the greater the OPU latency advantage. (a) v5p TPU-to-OPU ratio as a function of B_s and N, with the lines indicating parity and 10× win. 1 photonic core, 10 GHz vector and 1 MHz matrix modulation. Vector size is 8192. TPU works in int8 regime. (b) H100 SXM GPU-to-OPU ratio as a function of V_s and N^2C, with the line indicating parity. 10 GHz vector and 100 MHz matrix modulation. Batch size is 16384. GPU works in int8 regime (tensor cores).

The ratio of the time spent on matrix modulation to the time spent on batch processing can be expressed as $R = \frac{C \cdot f}{B_s \cdot f_{mat}}$. For configurations with $B_s/C = 1024$ vectors per core and $f_{mat} = 100$ MHz, this overhead becomes sufficiently small (R<0.1). Given this small contribution, the matrix modulation term can be omitted from subsequent analysis.

Under these operating conditions, the simplified latency model reveals that photonic performance scales with the product N^2C — combining core size squared and core count into a single design parameter. Physically, this parameter represents the total computational node count, which corresponds to the number of Mach-Zehnder interferometers in photonic architectures or the number of PEs in systolic arrays. This insight enables a systematic exploration strategy: we first determine the required N^2C values for achieving parity with electronic accelerators, then apply physical constraints (Section 4) to decompose this product into feasible configurations.

Having established the simplified scaling relationship, we now examine how vector dimensionality affects the N^2C product required for achieving parity or 10× advantage. As illustrated in Fig. 2, the impact of vector size on the required N^2C becomes negligible once the vector size (V_s) exceeds 2048. Since many contemporary architectures, including transformers (e.g., Llama 3.1), employ vector sizes at or above this threshold, we adopt a vector size of 8192 (corresponding to Llama 3.1–70B) for subsequent analysis.

Optics EXPRESS

Furthermore, for batch-per-core ratios exceeding 1024, the N^2C parameter required for achieving latency parity or advantage shows negligible dependence on batch size, converging to a constant determined by the electronic baseline's compute throughput and the OPU modulation frequency.

Findings 1: *We establish N^2C as the governing parameter for OPU competitiveness, capturing total computational parallelism: N^2 operations per core across C cores. At practical operating points ($f = 10$ GHz, $f_{\text{mat}} = 100$ MHz), achieving parity with H100 SXM requires $N^2C \geq 116\,964$ (INT8) or $N^2C \geq 232\,324$ (INT4); a 10× latency advantage demands $N^2C \geq 1.37M$ and $N^2C \geq 2.69M$, respectively (Table 4). These thresholds become invariant to workload parameters once $V_s \geq 2048$ and $B_s/C \geq 1024$—conditions satisfied by modern transformer architectures and other large-scale MVM workloads.*

Table 4. Required N^2C values for photonic accelerator performance targets. int8 and int4 refer to operating regimes used by both H100 and TPU v5p

	int8		int4	
Target	**Parity**	**10× Win**	**Parity**	**10× Win**
H100 SXM	116 964	1 371 241	232 324	2 686 321
TPU v5p	76 176	466 489	99 856	1 048 576

4. Optical losses and scalability analysis

The earlier analytical model established the parameter N^2C (Section 3.5), which includes core size, that defines the areas where photonics could potentially outperform electronics. This section focuses on evaluating existing optical architectures in terms of their ability to reach these core sizes utilizing a conceivable number of cores. Specifically, the paper examines two widely adopted architectures — SVD-based [9] and crossbar [40] — by analysing their designs, loss mechanisms, and effective number of bits (ENOB).

Hypothesis 2: *The established N^2C targets in Section 3.5 for parity and a tenfold advantage over GPU H100 SXM and TPU v5p are attainable with a feasible number of optical cores, despite the physical limitations of the OPU's core size.*

Understanding loss mechanisms is necessary for evaluating the scalability and functionality of OPUs. Losses directly impact the signal-to-noise and distortion ratio (SINAD) and computational accuracy, imposing constraints on the matrix size that can be processed in a single operation. Additionally, high optical losses reduce energy efficiency and increase the need for amplification, further complicating system design. Moreover, excessive losses limit both the maximum achievable bit depth and core size, restricting the overall computational capacity of the OPU.

This work focuses on photonic integrated circuits (PICs) as an example of an OPU, featuring the Mach-Zehnder interferometer (MZI) as a key component that enables the modulation of light amplitude [41], allowing optical encoding of values between 0 and 1 and thus implementing a multiplication operation. Furthermore, the subsequent combination of multiple signals can lead to an increase in amplitude through constructive interference, which is equivalent to an addition operation.

4.1. SVD architecture

The primary concept behind singular value decomposition (SVD) based architectures is the decomposition of a target real valued matrix using the SVD algorithm, which represents the matrix $W = U\Sigma V^{\dagger}$, where U, V - unitary matrices, Σ - diagonal matrix and $\dagger$ represents conjugate transpose operation. This allows for the utilization of the node, which consists of MZI followed by directional coupler and phase shifter, to perform unitary 2×2 transformation. Using nodes

Optics EXPRESS

and various OPU designs [41–43], it is possible to implement a unitary-diagonal-unitary matrices structure, ultimately achieving any given matrix W. This study specifically investigates the worst-case path, characterized by the maximum number of phase shifters encountered (Fig. 3(b)).

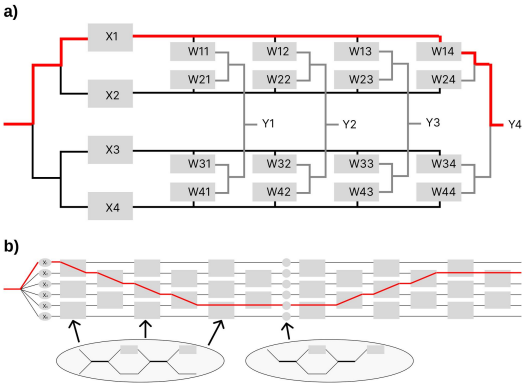

Fig. 3. OPU architectures. (a) Crossbar architecture with light propagation pass. (b) SVD-Clements architecture worst case loss pass and node composition.

4.2. SVD-Clements loss breakdown

To quantify optical losses, it is essential to analyse their accumulation along the propagation path from the laser source to the detector [37]:

$$\text{Loss}_{\text{Cl}} = \log_2(N)L_{\text{split}} + 10\log_{10}(N) + L_{\text{i/o}} + L_{\text{vec}} + W_{\text{l}} + (2N+1)L_{\text{Node}}, \tag{8}$$

where $N \in \{2, 4, 8, \dots\}$ is the core size, and $L_{\text{i/o}}$ denotes the fiber-to-chip coupling losses. The term $\log_2(N) \cdot L_{\text{split}} + 10\log_{10}(N)$ accounts for the physical and 3 dB splitting losses within the Y-splitter array. The modulation losses are defined as $L_{\text{vec}} = 2L_{\text{vecps}} + 2L_{\text{Coup}}$ for the vector encoding modulator and $L_{\text{Node}} = 2L_{\text{matps}} + 2L_{\text{Coup}}$ for the matrix encoding modulator, each comprising two phase shifters and two couplers. In the worst-case optical path, the light propagates through $2N + 1$ nodes, contributing $(2N + 1) \cdot L_{\text{Node}}$ to the total loss. Finally, $W_{\text{l}} = \eta(2N + 1)d$ represents the waveguide propagation loss, calculated using the attenuation per unit length (η) and the node length (d).

4.3. Crossbar architecture

A crossbar architecture consists of a waveguide array where tunable elements are located at the intersections [40,43]. Matrix-vector multiplication is performed as input vector X components are directly multiplied by matrix W elements at the MZI nodes, with the subsequent summation carried out through optical interference to produce an output vector Y (Fig. 3(a)), corresponding to the basic matrix-vector multiplication $Y = WX$.

4.4. Crossbar loss breakdown

Loss assessment in the crossbar architecture follows a similar approach to the SVD-Clements scheme [37]:

$$\text{Loss}_{\text{Xb}} = L_{\text{i/o}} + L_{\text{vec}} + \log_2(N)L_{\text{split}} + 10\log_{10}(N) + L_{\text{Coup}}(N-1) + \log_2(N)L_{\text{Coup}} + L_{\text{node}} + W_{\text{l}}, \tag{9}$$

where $N \in \{2, 4, 8, \dots\}$ represents the core size, and $L_{\text{i/o}}$ accounts for the fiber-to-chip coupling losses. The distribution and routing of light introduce several loss components: $\log_2(N) \cdot L_{\text{split}} +$

Optics EXPRESS

$10\log_{10}(N)$ from splitting light equally to each vector node, $L_{\text{Coup}}(N-1)$ from coupling light through the rows to the last column, and $\log_2(N)\cdot L_{\text{Coup}}$ from summing the products via a coupler array. The local losses at the active components are given by $L_{\text{vec}} = 2L_{\text{Coup}} + 2L_{\text{vecps}}$ for the vector modulator and $L_{\text{node}} = 2L_{\text{matps}} + 2L_{\text{Coup}}$ for the matrix modulator. Finally, $W_{\text{l}} = \eta dN$ denotes the total waveguide propagation loss, defined by the attenuation per unit length (η) multiplied by the physical length of a single node (d) and the core size.

4.5. SINAD calculation

SINAD formula (10) accounts for various factors that influence the noise characteristics of laser, detector and amplifier.

$$\text{SINAD} = \frac{(RGP_{o/p})^2}{\left[\sqrt{2q(RGP_{o/p}+I_d)+\frac{4kT}{R_L}+2\rho_{ASE}R^2GP_{o/p}+\rho_{ASE}^2R^2(2B_o-B_e)+R^2P_{o/p}^2\text{RIN}}+\sqrt{2qI_d+\frac{4kT}{R_L}+\rho_{ASE}^2R^2(2B_o-B_e)}\right]^2 B_e}, \tag{10}$$

where: $\rho_{ASE} = 2N_{SOA}n_{sp}\frac{hc}{\lambda}(G-1)$ represents the amplified spontaneous emission. Here N_{SOA} - number of semiconductor optical amplifiers (SOAs), n_{sp} - spontaneous emission factor, G - SOA gain value. The SINAD and ENOB calculation parameters are listed in Table 5.

Table 5. Parameters used for SINAD and ENOB calculations

Photonic components		Laser		Detector	
$L_{i/o}$	2 dB	P_{laser}	10 mW	R	1 A/W
η	3.6 dB/cm	RIN	−140 dB/Hz	I_d	30 nA
d	500 μm	λ	1550 nm	T	300 K
Y-splitter	0.01 dB	**Amplifier**		DR	10 GS/s
VecPS	0.6 dB	G	17 dB	R_L	50 Ω
MatPS	0.4 dB	N_{SOA}	1	B_0	25 GHz
Coup	0.1 dB	n_{sp}	1.5	B_e	$DR/\sqrt{2}$ GHz

In Eq. (10) q is elementary charge, k - Boltzmann constant, c - speed of light, h - Planck constant, λ - source wavelength, R - detector responsivity, P_{laser} - laser intensity, R_L - load resistance, I_d - dark current, B_0 - optical bandwidth, B_e - electrical bandwidth, DR - data rate, T - absolute temperature, RIN - relative intensity noise and $P_{o/p} = P_{laser}\cdot 10^{-\frac{\text{Loss(dB)}-\text{G}}{10}}$.

4.6. ENOB calculation

Knowing the circuit losses and the noise characteristics of individual components (such as the laser, amplifier and detector), SINAD can be determined. This, in turn, allows the ENOB to be calculated [37], providing an estimate of the bit depth of the system:

$$ENOB = \frac{SINAD - 1.76}{6.02}. \tag{11}$$

The maximum achievable core size is contingent upon the specific hardware components utilized. In particular, different phase shifters may be chosen for vector and matrix encoding as parts of MZI. Optoelectronic shifters trade high losses (1–5 dB) for speed (1–50 GHz), while thermo-optic shifters trade speed (about 100 kHz) for low losses (less than 0.01 dB) [38].

It is worth noting that optical amplification can be employed in the OPU to achieve a higher core size. However, this approach necessitates the additional consideration of the amplifier's introduced gain and noise. For the purpose of this analysis, only the addition of a single amplifier is considered. A comprehensive investigation into the delicate balance required when using multiple amplifiers — optimizing their quantity against their noise contributions — is beyond the scope of this work. Research on this topic is conducted in [37].

Optics EXPRESS

Consistent with previous studies [14,37], our model confirms substantial loss accumulation in optical architectures, which is especially pronounced in the SVD-Clements design (Fig. 4). Consequently, the Crossbar architecture is the preferable choice in terms of the ENOB metric, reaching an effective bit depth of approximately 2 with a core size of 256.

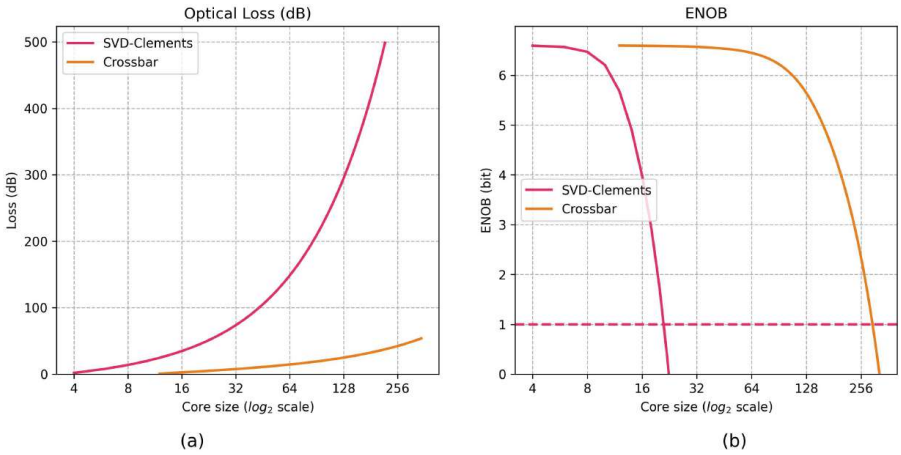

Fig. 4. SVD-Clements (thermo-optical) vs. crossbar (electro-optical) architectures. (a) Optical loss and (b) effective number of bits (ENOB) as a function of core size. Crossbar shows better scalability even with high-loss electro-optic phase shifters.

The ENOB obtained from Eq. (11) captures the link-level signal-to-noise-and-distortion budget — optical loss together with the noise of the laser, amplifier, and photodetector — and is intended as a first-order screening bound, not the end-to-end computational precision of the OPU. Several non-idealities that further limit the realized precision are not included in the present model, notably modulator nonlinearity, finite extinction ratio, crosstalk, thermal drift, fabrication variations, AD/DA quantization, and phase-error accumulation. More complete device-to-system models that treat individual effects have been reported [14,44,45]. Several of these effects can also be partly compensated during operation through calibration and self-calibration schemes that tune the device against fabrication imperfections and environmental drift [46,47]. A full-scale, simulation-based comparison at the matrix–vector and task level is left for future work.

Findings 2: *Optical loss analysis reveals a fundamental core-size/precision trade-off: crossbar architectures support* $N = 298$ *at 1-bit,* $N = 200$ *at 4-bit, and* $N = 108$ *at 6-bit ENOB. We select* $N = 128$ *as the design baseline—the largest power-of-two size achieving INT4-equivalent precision (ENOB* $= 5.6$*), directly compatible with GPU/TPU quantization modes. With* $N^2 = 128^2$ *fixed, the INT4 thresholds from Findings 1 translate directly into core counts under the same operating configuration (*$f = 10$ *GHz,* $f_{\mathrm{mat}} = 100$ *MHz): parity demands 15 cores (vs. H100) or 7 cores (vs. TPU v5p);* $10\times$ *advantage requires 164 and 64 cores, respectively. These targets establish the design space for power-constrained analysis in Hypothesis 3.*

5. Power consumption constraints

The next stage of analysis is to account for energy constraints, which are necessary to determine the feasible number of optical cores C within the latency model. The energy efficiency model demonstrated in this chapter uses an approach similar to that of [37], which offered assessment of energy efficiency for the SVD architecture based on thermo-optical MZIs. This model is extended here by including crossbar architecture and incorporating electro-optical MZIs and matrix frequency as a parameter in Eq. (12); the corresponding model parameters are listed in

Optics EXPRESS

Table 6. This extension is essential for evaluating the impact of the high modulation frequencies that our latency model identifies as one of the key parameters

$$\begin{aligned} P_{\text{tot}}(N) = &\frac{P_{\text{laser}}}{\eta_{\text{WPE}}\gamma} + N\left(E_{\text{E/O}} + E_{\text{drv,vec}}\right) n_{\text{bit}} f + E_{\text{mat}}(n_{\text{ps}}) \\ &+ \frac{n_{\text{ps}}}{2} E_{\text{drv,mat}} \cdot n_{\text{bit}} f_{\text{mat}} \cdot \mathbf{1}_{\{\text{EO}\}} + E_{\text{mem}}(N, n_{\text{bit}}, f_{\text{mat}}, f) \\ &+ N \cdot n_{\text{SOA}} P_{\text{SOA}} + N \cdot E_{\text{AFE}} n_{\text{bit}} f + N \cdot E_{\text{ADC,FOM}} \cdot 2^{n_{\text{bit}}} f \end{aligned} \quad , \tag{12}$$

where:

$$\gamma = 10^{\frac{n_{\text{SOA}} g_{\text{SOA,dB}}}{10}}, \tag{13}$$

$$E_{\text{mat}}(n_{\text{ps}}) = \begin{cases} \frac{n_{\text{ps}}}{2} P_{\text{TO}}, & \text{TO mesh} \\ n_{\text{ps}} E_{\text{E/O}} f_{\text{mat}}, & \text{EO mesh} \end{cases}, \tag{14}$$

$$E_{\text{mem}}(N, n_{\text{bit}}, f_{\text{mat}}, f) = \left\lceil \frac{N}{16} \right\rceil P_{\text{mem,int}} \left[\alpha + \alpha' \frac{f_{\text{mat}}}{f} \mathbf{1}_{\{\text{EO}\}} \right] \quad , \tag{15}$$

where $\alpha(n_{\text{bit}}) = \max(1, \lceil n_{\text{bit}}/4 \rceil)$ is the bit-width multiplier accounting for multi-cycle memory transfers at precisions exceeding 4 bits.

Table 6. Power consumption model parameters and their values used for calculations shown in Fig. 5 Parameter values are adopted from the reference framework of [37]

f	f_{mat}	$E_{\{\text{EO}\}}$	$E_{\text{drv,vec/mat}}$	$P_{\text{mem,int}}$	η_{wpe}	P_{SOA}	E_{FOM}	E_{AFE}	P_{TO}
Vector freq.	Matrix freq.	EO MZI energy/bit	EO driver energy/bit	I/O circuit power	Wall-plug efficiency	SOA power	ADC FoM	AFE power	TO MZI power
10 GHz	100 MHz	200 fJ	2 pJ	5.77 mW	0.1	42 mW	0.335 pJ/step	0.4 pJ	5 mW

In (12), the total power consumption is decomposed into several functional components. The equation first accounts for the laser's electrical power supply $\frac{P_{\text{laser}}}{\eta_{\text{WPE}}\gamma}$, which depends on its wall plug and SOA-related efficiencies. Power required for data encoding is divided into two main parts: writing the input vector $N(E_{\{\text{EO}\}}+E_{\text{drv,vec}})n_{\text{bit}}f$ and modulating the matrix weights $E_{\text{mat}}(n_{\text{ps}})$, with an additional term $\frac{n_{\text{ps}}}{2}E_{\text{drv,mat}} \cdot n_{\text{bit}}f_{\text{mat}} \cdot \mathbf{1}_{\text{EO}}$ for EO matrix drivers. It should be noted that the thermo-optic (TO) variant in Eq. (14) assumes uniformly distributed matrix weights, an approach consistent with the methodology in [37]. The model also includes power consumption for optical amplification $N \cdot n_{\text{SOA}}P_{\text{SOA}}$ and the analog front end $N \cdot E_{\text{AFE}}n_{\text{bit}}f$.

Finally, it incorporates the power for data fetch interfacing circuits $E_{\text{mem}}(\cdot)$ and for the analog-to-digital conversion $N \cdot E_{\text{ADC,FOM}} \cdot 2^{n_{\text{bit}}}f$, derived from the ADC's figure of merit. Equation (12) allows us to estimate the total energy consumption of a photonic core (see Fig. 5).

Hypothesis 3: *Specific OPU configurations (N^2C) can achieve parity or superiority in terms of TOPS/W compared to the electronic accelerators considered (A100, H100, TPU v4), given on-par energy consumption.*

As shown in the right panel of Fig. 5, the crossbar architecture with EO modulators achieves the lowest power consumption at 4-bit precision. Furthermore, its shorter optical path length allows larger core sizes, compared to the SVD architecture. Additionally, SVD schemes require data preprocessing. Therefore, we focus exclusively on the EO crossbar design for the remainder of this analysis.

A power breakdown of this configuration identifies two regime-dependent bottlenecks. At $f_{\text{mat}} = 100$ MHz, the ADC power share rises from 21.1% (at 4-bit, core size $N = 64$) to 56.1% (at 8-bit, core size $N = 256$) due to the $2^{n_{\text{bit}}}$ scaling of ADC energy. Conversely, at $f_{\text{mat}} = 1$ GHz, the EO matrix drivers become the primary consumer, accounting for 69.4% to 77.2% of the total

Optics EXPRESS

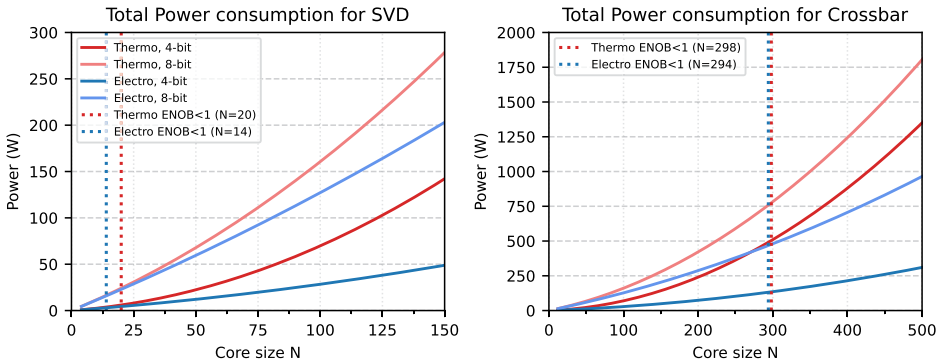

Fig. 5. Power consumption for Crossbar and SVD-Clements architectures using MZIs based on thermo-optic and electro-optic phase shifters.

power across the analyzed configurations. DAC/ADC subsystems and high-frequency matrix modulation drivers thus emerge as the principal targets for power optimization.

The results of the power-performance analysis are summarized in Table 7, which presents the viable design points after applying energy constraints to the physically pruned parameter space (from Section 4). Our methodology proceeds as follows: First, for each core size N and bit depth n_{bit}, we use our energy model and the parameters in Table 6 to calculate the power of the optical core P_{core}, considering physical constraints. Second, we determine the maximum number of cores C by limiting the total OPU power P_{tot} to a given thermal design power (TDP) of ASIC. Finally, we use our latency model to evaluate the performance (in TOPS) of this power-constrained configuration for the same inference task as in the Introduction, using the parameters from Table 1. This results in the final energy efficiency in TOPS/W. By enforcing a fixed power budget across all devices, the energy efficiency metric (TOPS/W) is directly proportional to the performance metric (TOPS). Therefore, achieving performance parity or advantage in our latency model is a necessary and sufficient condition for achieving the same result in energy efficiency. The SRAM power consumption is accounted for as a fixed budget allocation rather than being explicitly modeled (see Findings 3). To ensure a fair comparison, we calculate the TOPS for ASICs using the same latency model applied to the OPU.

Table 7. Equal-power comparison of electronic accelerators and multi-core OPU configurations for electro-optic interface bit-depth n_{bit} = 4/8

Device		**A100 (400W) [33],**			**H100 (700W) [26],**			**TPUv4 (170W) [48],**		
TOPS/W		3.12 / 1.6			6.12 / 3.06			- / 1.6		
N	***P_{core}***	***C***	**TOPS/W**	***P_{tot}***	***C***	**TOPS/W**	***P_{tot}***	***C***	**TOPS/W**	***P_{tot}***
64	16.2/77.7	24/4	4.3/0.8	389/311	42/8	3.9/0.9	680/622	10/2	4.5/0.95	162/155
128	39.4/168.7	8/2	6.2/1.6	315/337	16/4	6.8/1.8	630/675	4/1	7.5/1.9	158/169

For power budget relative to the A100 GPU, our model identifies an OPU configuration (N = 128, C = 2) that reaches TOPS/W parity at n_{bit} = 8. Here n_{bit} denotes the operand width of the electro-optic interface — converters, drivers, and memory traffic — which is set by the numerical format of the workload rather than by the analog fidelity of the optical core. At N = 128 the latter remains bounded by ENOB = 5.6 (Findings 2), so this design point executes an INT8-formatted workload at A100-level energy efficiency while resolving approximately 5.6 effective bits of the result. The configuration consumes power sufficiently below the TDP to leave a 63 W budget explicitly allocated for the SRAM circuitry. Our model further reveals that reducing the target precision to n_{bit} = 4 lowers the per-core power consumption. This enables

Optics EXPRESS

scaling the core count to $C = 8$ while maintaining a core size of $N = 128$. This configuration doubles the TOPS/W efficiency relative to the A100 while maintaining a sufficient power reserve for the associated SRAM circuitry. When compared to an H100 GPU under an identical power budget, a 16-core OPU ($N = 128$, $C = 16$) operating at 4-bit precision delivers a ∼10% superior TOPS/W. This comparison holds under realistic constraints, with the 4-bit operand width of this configuration matched by the effective resolution available at $N = 128$, and with the specified power reserve maintained for its SRAM system.

Findings 3: *Constrained to the $N = 128$ baseline from Findings 2, our energy efficiency analysis demonstrates that photonic accelerators match and exceed GPU efficiency within realistic power budgets. At INT8 operand width, a two-core design ($N = 128$, $C = 2$) matches A100 efficiency on an INT8-formatted workload; this point is an iso-format comparison, since the analog fidelity of the core remains bounded by ENOB = 5.6. The INT4 regime is more favorable with operand width and effective resolution matched: $C = 8$ doubles A100 efficiency; $C = 16$ surpasses H100 by 10%. All configurations reserve sufficient power margin for SRAM circuitry. These results establish that physically realizable OPU configurations achieving efficiency parity and advantage over the A100 and H100 GPUs exist at INT4 precision within practical power constraints.*

The efficiency comparison places the OPU and the electronic baselines on the same footing: both are evaluated with the same roofline model under a common power budget, so their shared idealizations partially cancel in the ratio, and, where hardware validation is available, the model predicts throughput at or above measured hardware, crediting the electronic baselines with their full modeled performance. On this basis the crossbar OPU reaches roughly twice the A100 efficiency and a ∼ 10% margin over the H100 at INT4. These two results differ in robustness: the ∼ 2x A100 margin lies well outside the latency model's validation error, whereas the ∼ 10% H100 margin is comparable to it, and we accordingly treat the H100 crossing as a point of interest whose precise magnitude is left to cycle-accurate simulation and physical prototyping.

6. Discussion

The analytical model, although simple and coarse, is effective for conducting broad DSE across large parameter spaces for both the computational tasks and the physical hardware. For example, it handles the broad spectrum of OPU modulation frequencies, reflecting the diverse underlying technologies, which vary from kHz (thermo-optic) to GHz (electro-optic).

While introducing limitations in memory-bound characterization, the roofline methodology inherently provides conservative estimates that become increasingly accurate as workloads approach peak computational intensity. This is precisely the operating regime where photonic processors demonstrate competitive potential. While smaller workloads expose limitations due to complex cache behavior, the framework is well suited to identifying promising designs for the large-scale training and inference workloads where photonic advantages are most pronounced.

The batch sizes and vector dimensions evaluated in our model were selected to reflect contemporary AI workloads, using as a benchmark Llama 3.1–70B [1] inference operations that utilize 8192-dimensional embeddings — representative of modern transformer architectures.

The physical constraints analysis highlights the challenges in achieving dimensionalities for OPU beyond $N = 256$, where only 2-bit resolution is feasible due to increasing optical losses. As the core size grows, these losses escalate, leading to a reduction in achievable bit depth and limiting computational accuracy. Overcoming this constraint requires advancements in architectural design and the optimization of key components to enhance scalability and efficiency. Improving both the hardware and optical processing mechanisms is essential for enabling larger dimensional operations while mitigating loss-induced degradation.

The preceding analysis fixes the core size and count required for competitiveness; these translate directly into a chip-area estimate. For the electro-optic crossbar baseline the footprint

Optics EXPRESS

is dominated by the electro-optic phase shifters, so the active area is approximated by the MZI count. A single $N = 128$ core comprises 128 MZIs for vector encoding and $N^2 = 16,384$ MZIs for the matrix, i.e., 16,512 MZIs in total. We take the phase-shifter arm length equal to the node length d = 500 μm used in the loss model (Table 5), consistent with the length range of integrated electro-optic phase shifters [38], and a transverse MZI pitch of 60 μm reported for interferometer meshes [49]. This yields a per-MZI footprint of 0.03 mm^2 and a single-core active area of approximately 495 mm^2. Parity with the H100 (15 cores) and a tenfold advantage (164 cores) then correspond to roughly 7.4×10^3 mm^2 and 8.1×10^4 mm^2 of active area, respectively; these values are lower bounds, as they omit waveguide routing, couplers, photodetectors, and data-conversion circuitry. Even a single core is already of the order of the single-reticle area limit of current lithography (a few cm per side), and the multi-core configurations required for parity and advantage exceed it by roughly one and two orders of magnitude. These footprints place such systems beyond monolithic integration and necessitate reticle-stitching or wafer-scale approaches [50], which further motivates the wavelength-multiplexing strategy discussed below.

Based on the conducted DSE, several research directions can be identified as essential for achieving a sustainable advantage of photonic accelerators over their electronic counterparts. First, the development of DAC/ADC subsystems and their drivers, specifically optimized for photonic accelerators, has the potential to reduce overall power consumption, thereby improving the energy efficiency of the system. Second, the use of optical spectral multiplexing can increase the number of effective cores on a single OPU chip, enabling near-linear performance scaling with only a marginal rise in power consumption, primarily due to the generation of optical frequency combs or the use of additional light sources. Third, advances in optical phase shifters and more sophisticated optical modulation schemes are expected to reduce cumulative optical losses, thus enabling computation with higher-precision data representations. Furthermore, exploration of output-stationary photonic architectures presents an opportunity to eliminate idle cycles associated with weight matrix reconfiguration, thereby lowering the energy cost of high-frequency weight modulation. Finally, the future design of optical cache memory could reduce both latency and energy overheads associated with memory access.

A natural extension of this work would involve enhancing the memory subsystem model through refined DRAM-SRAM interaction characterization and comprehensive cache modeling, incorporating data tiling and mapping optimizations to improve accuracy in low-throughput operating regimes. The more accurate simulation of SRAM behavior, should incorporate dynamic power consumption, memory capacity, and architectural organization, all of which are sensitive to the type of workload, the underlying compute architecture, operating frequencies, and other system parameters. In the present study, the power consumption of SRAM was conservatively upper-bounded at approximately 50 W. However, SRAM power consumption can increase significantly with higher operating frequencies [10], which must be captured more faithfully in the next phase of modeling.

7. Conclusions

The conditions under which optical processing units (OPUs) can match or surpass GPUs and TPUs are established within a unified analytical framework, validated against H100 and TPU v5e measurements. The product N^2C, representing total computational parallelism, is identified as the governing parameter of OPU competitiveness. The corresponding performance thresholds are shown to become workload-agnostic for vector sizes $V_s \geq 2048$ and batch-per-core ratios $B_s/C \geq 1024$, conditions that are satisfied by contemporary transformer architectures. This invariance allows accelerator specifications to be decoupled from model-specific tuning.

The accessible region of the design space is further constrained by physical limitations of integrated photonic platforms. Optical loss accumulation favors the crossbar architecture and restricts the practical core size to $N = 128$ at INT4-equivalent precision. Within this constraint,

Optics EXPRESS

the energy analysis indicates that OPU configurations can attain energy efficiency comparable to, and in specific cases exceeding, that of the A100 and H100 GPUs: a 16-core configuration is shown to surpass the H100 by approximately 10%, while an 8-core configuration provides roughly twice the efficiency of the A100. These INT4 advantage configurations satisfy the ENOB requirement — operand width matched by effective resolution at $N = 128$ — whereas the INT8 A100-parity point holds as an iso-format comparison bounded by ENOB = 5.6; all remain within realistic power budgets.

The energy breakdown identifies DAC/ADC subsystems as the dominant contributors to power consumption and, accordingly, as the primary targets for subsequent optimization. The low-batch regime ($B_s \approx 1{,}000$) is noted as a direction warranting further investigation in the context of latency-critical inference. A complete summary of the explored parameter space and the corresponding findings is provided in Table 8.

Table 8. Summary of hypotheses' objectives and findings for photonic accelerator design space exploration. All quantitative thresholds correspond to the operating point f = 10 GHz, f_{mat} = 100 MHz

Objective	Key findings
Identify photonic core configurations achieving latency parity or superiority vs. GPUs/TPUs in INT8/INT4 regimes	1) Parameter N^2C governs latency at large B_s, V_s (N—linear core size, C—cores) 2) H100 INT4: parity $N^2C \geq 232$k; 10× win at $N^2C \geq 2.7$M (full set: Table 4) 3) Requirements independent of $V_s \geq 2048$, $B_s/C \geq 1024$
Evaluate physical feasibility of N^2C targets considering optical loss constraints	1) Crossbar selected over SVD: lower loss accumulation, larger feasible N 2) Max core sizes: 298 (1-bit), 200 (4-bit), 108 (6-bit) 3) Baseline: $N = 128$, INT4 (ENOB=5.6) 4) Single core is insufficient for parity ($N^2C = 16{,}384$) 5) Parity: 15 cores (H100), 7 cores (TPUv5p) 6) 10× advantage: 164 cores (H100), 64 cores (TPUv5p)
Determine on-par energy OPU configurations achieving TOPS/W parity or superiority vs. GPU and TPU	1) Calculated task-specific TOPS/W $\propto$ TOPS (fixed power, latency model) 2) $N = 128$, $C = 2 \rightarrow$ parity with A100 (INT8 format, ENOB-limited) 3) $N = 128$, $C = 8 \rightarrow$ 2× higher vs. A100 (INT4) 4) $N = 128$, $C = 16 \rightarrow$ 10% higher vs. H100 (INT4)

The proposed framework is intended as a rapid screening tool rather than a replacement for detailed simulation. Memory subsystem complexities and fine-grained error sensitivity are deliberately abstracted in order to preserve a closed-form analytical formulation. This simplification enables efficient traversal of large parameter landscapes that remain computationally prohibitive for cycle-accurate simulators in broad sweeps. The configurations identified as promising therefore require subsequent verification through cycle-accurate simulation and physical prototyping.

The presented analysis yields concrete specifications, including core sizes, core counts, and operating frequencies, for photonic accelerators capable of outperforming the A100 and H100 GPUs and TPU v5p under INT4 operation. These results delineate the regime in which photonic

Optics EXPRESS

computing offers a quantifiable advantage and provide a basis for the design of subsequent system-level implementations.

Funding. Ministry of Economic Development of the Russian Federation (139-15-2025-012).

Acknowledgment. This work was supported by the the Ministry of Economic Development of the Russian Federation in accordance with the subsidy agreement (Identifier no. 000000C313925P4H0002).

Disclosures. The authors declare no conflicts of interest.

Data availability. Data underlying the results presented in this paper are not publicly available at this time but may be obtained from the authors upon reasonable request.

References

1. A. Dubey, A. Jauhri, A. Pandey, *et al.*, "The Llama 3 herd of models," arXiv (2024).
2. M. Barnett, "The promise of reasoning models," (2025). Accessed: 2025-08-18.
3. J. Kaplan, S. McCandlish, T. Henighan, *et al.*, "Scaling laws for neural language models," arXiv (2020).
4. A. Ivanov, N. Dryden, T. Ben-Nun, *et al.*, "Data movement is all you need: a case study on optimizing transformers," Proceedings of Machine Learning and Systems **3**, 711–732 (2021).
5. J. You, J.-W. Chung, and M. Chowdhury, "Zeus: understanding and optimizing GPU energy consumption of DNN training," in *20th USENIX Symposium on Networked Systems Design and Implementation (NSDI 23)*, (2023), pp. 119–139.
6. N. P. Jouppi, C. Young, N. Patil, *et al.*, "In-datacenter performance analysis of a tensor processing unit," in *Proceedings of the 44th annual international symposium on computer architecture*, (2017), pp. 1–12.
7. Y.-H. Chen, T. Krishna, J. S. Emer, *et al.*, "Eyeriss: an energy-efficient reconfigurable accelerator for deep convolutional neural networks," IEEE J. Solid-State Circuits **52**(1), 127–138 (2016).
8. Y.-H. Chen, T.-J. Yang, J. Emer, *et al.*, "Eyeriss v2: a flexible accelerator for emerging deep neural networks on mobile devices," IEEE J. Emerg. Sel. Topics Circuits Syst. **9**(2), 292–308 (2019).
9. Y. Shen, N. C. Harris, S. Skirlo, *et al.*, "Deep learning with coherent nanophotonic circuits," Nat. Photonics **11**(7), 441–446 (2017).
10. C. Demirkiran, F. Eris, G. Wang, *et al.*, "An electro-photonic system for accelerating deep neural networks," J. Emerg. Technol. Comput. Syst. **19**(4), 1–31 (2023).
11. F. L. Ivanov, V. V. Krasnikov, A. A. Grunin, *et al.*, "Optoelectronic sensors with photonic memory for trajectory prediction," APL Machine Learning **4**, 036102 (2026).
12. P. L. McMahon, "The physics of optical computing," Nat. Rev. Phys. **5**(12), 717–734 (2023).
13. J. Spall, X. Guo, T. D. Barrett, *et al.*, "Fully reconfigurable coherent optical vector–matrix multiplication," Opt. Lett. **45**(20), 5752–5755 (2020).
14. G. Giamougiannis, A. Tsakyridis, Y. Ma, *et al.*, "A coherent photonic crossbar for scalable universal linear optics," J. Lightwave Technol. **41**(8), 2425–2442 (2023).
15. H. Zhou, J. Dong, J. Cheng, *et al.*, "Photonic matrix multiplication lights up photonic accelerator and beyond," Light: Sci. Appl. **11**(1), 30 (2022).
16. T. Zhou, W. Wu, J. Zhang, *et al.*, "Ultrafast dynamic machine vision with spatiotemporal photonic computing," Sci. Adv. **9**(23), eadg4391 (2023).
17. J. T. Meech, V. Tsoutsouras, and P. Stanley-Marbell, "The data conversion bottleneck in analog computing accelerators," arXiv (2023).
18. G. V. Fomin, D. S. Batanov, E. O. Kimlichenko, *et al.*, "Promise and pitfalls of random projection algorithms for optical processors," Opt. Express **34**(13), 24098–24113 (2026).
19. Z. Yin, M. Zhang, A. Begovic, *et al.*, "SimPhony: a device-circuit-architecture cross-layer modeling and simulation framework for heterogeneous electronic-photonic AI system," arXiv (2024).
20. H. Zhu, J. Gu, H. Wang, *et al.*, "Lightening-transformer: a dynamically-operated optically-interconnected photonic transformer accelerator," in *2024 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, (IEEE, 2024), pp. 686–703.
21. S. Williams, A. Waterman, and D. Patterson, "Roofline: an insightful visual performance model for multicore architectures," Commun. ACM **52**(4), 65–76 (2009).
22. Z. Lin, B. J. Shastri, S. Yu, *et al.*, "120 GOPS photonic tensor core in thin-film lithium niobate for inference and in situ training," Nat. Commun. **15**(1), 9081 (2024).
23. H. Ouyang, Z. Tao, J. You, *et al.*, "Computing dimension for a reconfigurable photonic tensor processing core based on silicon photonics," Opt. Express **32**(18), 31205–31219 (2024).
24. V. V. Yushkov, A. S. Shorokhov, and A. A. Fedyanin, "Ultrafast optically tunable GaAs metasurfaces for analog image processing," ACS Omega **11**(2), 3311–3318 (2026).
25. R. Hamerly, L. Bernstein, A. Sludds, *et al.*, "Large-scale optical neural networks based on photoelectric multiplication," Phys. Rev. X **9**(2), 021032 (2019).
26. NVIDIA Corporation, "NVIDIA H100 Tensor Core GPU Architecture," Tech. rep., NVIDIA Corporation (2022).
27. Google Cloud, "System Architecture: TPU VM," Tech. rep., Google LLC (2023).
28. Google Cloud, "Cloud TPU v5p Documentation," Tech. rep., Google LLC (2023).

Optics EXPRESS

29. Google Cloud, "Cloud TPU v5p Training," Tech. rep., Google LLC (2023).
30. NVIDIA Corporation, "NVIDIA Ampere GA102 GPU Architecture Whitepaper (v2)," Tech. rep., NVIDIA Corporation (2020).
31. Y. Dong, Y. Miao, W. Li, *et al.*, "Accelerating LLM inference throughput via asynchronous KV cache prefetching," arXiv (2025).
32. S.-C. Kao, S. Subramanian, G. Agrawal, *et al.*, "Flat: an optimized dataflow for mitigating attention bottlenecks," in *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2*, (2023), pp. 295–310.
33. NVIDIA Corporation, "NVIDIA A100 Tensor Core GPU Architecture," Tech. rep., NVIDIA Corporation (2020).
34. R. Xu, S. Ma, Y. Wang, *et al.*, "Configurable multi-directional systolic array architecture for convolutional neural networks," ACM Transactions on Architecture and Code Optimization (TACO) **18**, 1–24 (2021).
35. JAX Team, Google, "How to think about TPUs," https://jax-ml.github.io/scaling-book/tpus/ (2025). Accessed: 2025.
36. D. Patel and A. Kostovic, "TPUv5e: the new benchmark in cost-efficient inference and training for <200B parameter models," https://semianalysis.com/2023/09/01/tpuv5e-the-new-benchmark-in-cost/ (2023). SemiAnalysis. Accessed: 2025.
37. M. A. Al-Qadasi, L. Chrostowski, B. J. Shastri, *et al.*, "Scaling up silicon photonic-based accelerators: challenges and opportunities," APL Photonics **7**(2), 020902 (2022).
38. H. Sun, Q. Qiao, Q. Guan, *et al.*, "Silicon photonic phase shifters and their applications: a review," Micromachines **13**(9), 1509 (2022).
39. D. Sturm and S. Moazeni, "Scalable coherent optical crossbar architecture using PCM for AI acceleration," in *2023 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, (IEEE, 2023), pp. 1–6.
40. J. Feldmann, N. Youngblood, M. Karpov, *et al.*, "Parallel convolutional processing using an integrated photonic tensor core," Nature **589**(7840), 52–58 (2021).
41. M. Reck, A. Zeilinger, H. J. Bernstein, *et al.*, "Experimental realization of any discrete unitary operator," Phys. Rev. Lett. **73**(1), 58–61 (1994).
42. W. R. Clements, P. C. Humphreys, B. J. Metcalf, *et al.*, "Optimal design for universal multiport interferometers," Optica **3**(12), 1460–1465 (2016).
43. A. Tsakyridis, G. Giamougiannis, A. Totović, *et al.*, "Fidelity restorable universal linear optics," Adv. Photonics Res. **3**(10), 2200001 (2022).
44. P. S. Kincaid, N. Andriolli, G. Contestabile, *et al.*, "Addressing optical modulator non-linearities for photonic neural networks," Commun. Eng. **4**(1), 58 (2025).
45. A. Marchisio, F. Da Ros, V. Curri, *et al.*, "Comprehensive model of MZI-based circuits for photonic computing applications," Commun. Phys. **8**(1), 277 (2025).
46. Q. Yan, H. Ouyang, Z. Tao, *et al.*, "Multi-wavelength optical information processing with deep reinforcement learning," Light: Sci. Appl. **14**(1), 160 (2025).
47. J. Wang, X. Xu, X. Zhu, *et al.*, "Microcomb-enabled parallel self-calibration optical convolution streaming processor," Light: Sci. Appl. **15**(1), 149 (2026).
48. N. Jouppi, G. Kurian, S. Li, *et al.*, "TPU v4: An optically reconfigurable supercomputer for machine learning with hardware support for embeddings," in *Proceedings of the 50th annual international symposium on computer architecture*, (2023), pp. 1–14.
49. I. A. Williamson, T. W. Hughes, M. Minkov, *et al.*, "Reprogrammable electro-optic nonlinear activation functions for optical neural networks," IEEE J. Sel. Top. Quantum Electron. **26**(1), 1–12 (2019).
50. T. J. Seok, K. Kwon, J. Henriksson, *et al.*, "Wafer-scale silicon photonic switches beyond die size limit," Optica **6**(4), 490–494 (2019).