HISIM: Analytical Performance Modeling and Design Space Exploration of 2.5D/3D Integration for AI Computing

Zhenyu Wang, *Graduate Student Member, IEEE*, Pragnya Sudershan Nalla, *Graduate Student Member, IEEE*, Jingbo Sun, *Graduate Student Member, IEEE*, A. Alper Goksoy, *Graduate Student Member, IEEE*, Sumit K. Mandal, *Member, IEEE*, Jae-sun Seo, *Senior Member, IEEE*, Vidya A. Chhabria, *Member, IEEE*, Jeff Zhang, *Member, IEEE*, Chaitali Chakrabarti, *Fellow, IEEE*, Umit Y. Ogras, *Senior Member, IEEE*, Yu Cao, *Fellow, IEEE*

***Abstract*—Monolithic designs face significant fabrication cost and data movement challenges, especially when executing complex and diverse AI models. Advanced 2.5D/3D packaging promises high bandwidth and connection density to overcome these challenges, yet it also introduces new electro-thermal constraints. This paper develops a suite of analytical performance models to enable efficient benchmarking of a 2.5D/3D heterogeneous system for energy-efficient AI computing. These models encompass various performance metrics related to computing units, network-on-chip, and network-on-package. The results are summarized into a new tool, HISIM, which is 10^4–$10^6\times$ faster than state-of-the-art AI benchmark tools. Furthermore, HISIM integrates rapid thermal simulation for the 2.5D/3D system, helping shed light on both the potential and limitations of 2.5D/3D heterogeneous integration on representative AI algorithms. The code of HISIM is available at https://github.com/mec-UMN/HISIM.**

***Index Terms*—heterogeneous integration, 2.5D/3D, chiplet, in-memory computing, network-on-package, thermal simulation**

I. Introduction

Artificial intelligence (AI) systems have effectively addressed practical problems across multiple domains. The complexity of state-of-the-art AI models, such as deep neural networks (DNNs), transformers, and graph neural networks (GNNs), grows continuously to enable higher accuracy and capabilities [1], [2]. Existing monolithic integrated circuits (ICs) offer many benefits for AI acceleration with large-scale integration and high performance. However, their scaling trend still lags behind contemporary AI models, particularly on the demand of memory access, on-chip cache capacity, and data movement. For instance, vision transformer (ViT) utilizes over 300 million parameters and 24 layers for image classification [3], [4]. If all the weights were mapped on a 28nm monolithic chip, it could require an area of more than $3,000mm^2$ [5]. Designing accelerators for such big AI models poses a tremendous challenge, especially for edge devices.

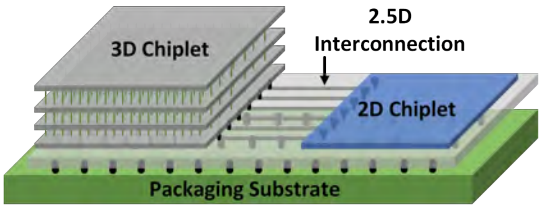

Fig. 1. Architecture overview of a 2.5D/3D heterogeneous system.

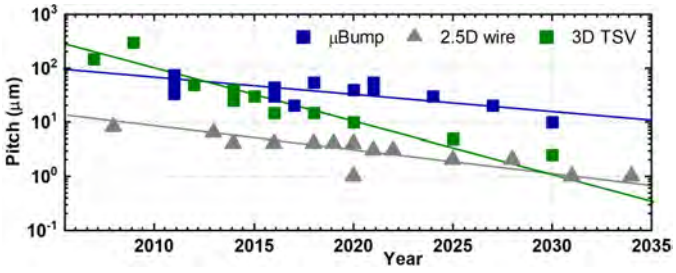

Fig. 2. The scaling trend of various interconnections in 2.5D/3D packaging. Symbols are from references [11]–[26].

Due to the limited monolithic area available for processing big AI models, their training and inference require frequent data movement between external memory and processing units, which becomes the dominant factor of energy consumption and latency [6]–[10]. In this context, recent advances in 2.5D/3D packaging provide a promising solution to address the challenges of high bandwidth, low-power data movement, and large-scale integration that are essential for big AI models.

Figure 1 illustrates a heterogeneous architecture that contains multiple chiplets connected via a 2.5D bridge embedded in silicon interposer, glass, or organic substrate, and/or 3D interconnections (such as through-silicon-vias or TSVs). The network-on-package (NoP) enables inter-chiplet communica-

This work was supported in part by CoCoSys, one of seven centers in JUMP 2.0, a Semiconductor Research Corporation (SRC) program sponsored by DARPA; and in part by the US Department of Energy (DOE) under contract DE-AC05-00OR22725.

Zhenyu Wang, Jingbo Sun, Vidya A. Chhabria, Jeff Zhang, and Chaitali Chakrabarti are with the School of Electrical, Computer, and Energy Engineering, Arizona State University, Tempe, AZ, 85287, USA (e-mail: {zwang586, jsun127, vachhabr, jeffzhang, chaitali}@asu.edu)

Pragnya Sudershan Nalla and Yu Cao are with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN, 55455, USA (e-mail: {nalla052, yucao}@umn.edu)

A. Alper Goksoy, Umit Y. Ogras are with the Department of Electrical and Computer Engineering, University of Wisconsin-Madison, Madison, WI, 53706, USA (e-mail: {agoksoy, uogras}@wisc.edu)

Sumit K. Mandal is with the Department of Computer Science and Automation, Indian Institute of Science, Bengaluru, 560012, India (e-mail: skmandal@iisc.ac.in)

Jae-sun Seo is with the Department of Electrical and Computer Engineering, Cornell Tech, New York, NY, 10044, USA (e-mail: js3528@cornell.edu)

tion with multiple channels, designed to deliver higher bandwidth, lower energy consumption per bit, and lower fabrication costs compared to traditional monolithic designs [27], [28]. Although 2.5D integration is currently available, it requires more physical space compared to 3D integration due to the additional peripheral circuits for inter-chiplet signaling. In contrast, future 3D stacking with TSVs offers more compact integration and denser connections, yet it will face significant thermal and reliability challenges [29]–[31]. Both architectures have a consistent scaling roadmap to reduce the feature size of interconnections, as shown in Figure 2. This trend ensures a continuous improvement in data movement among chiplets.

As the advancement of 2.5D/3D heterogeneous integration (HI) opens enormous opportunities for future big AI computing, the mapping and optimizing target AI algorithms on a HI system present intriguingly complex challenges, attributed to the abundance of design variables involved in this process. These variables include algorithmic structures (such as layers, kernel size, precision, etc.), technology parameters (such as 2.5D, 3D, interconnection, etc.), design choices (such as computing cores, array size, network topology, memory, etc.), and placement and routing schemes. To perform algorithm-hardware co-design in such a vast space, early-stage design space exploration is critical to narrow down the design scope and derive valuable insights at the architecture level. Such exploration requires a 2.5D/3D HI simulator that is *scalable* with essential design variables, *fast* in end-to-end performance prediction, and *accurate* in key performance metrics. This tool can be employed during the initial phase of architectural definition, complementing commercial 2.5D/3D EDA tools that generate physical details in subsequent stages.

This paper presents a benchmark tool, Heterogeneous Integration Simulation with Interconnect Modeling (HISIM). Distinguished from previous simulation tools that contain many technological and circuit details, HISIM focuses on key system-level metrics, such as latency, energy consumption, and data movement. This emphasis achieves a balance between model scalability, accuracy, and efficiency in order to support fast algorithm-hardware co-design. Table I summarizes some recent simulators for chiplet and related network communication. Several simulators target computing units, particularly the latest in-memory computing (IMC) cores, such as NeuroSim [32] and MNSIM [33]. Because these simulators typically use a bottom-up approach, beginning with devices, followed by circuits, and ultimately systems, their simulation speed is usually slow. In addition, there is a need for full consideration of 2.5D/3D network communication among multiple cores. Some other simulators, such as BookSim 2.0 [34] and Ratatoskr [35], provide cycle-accurate simulation and performance evaluation of 2D network-on-chip (NoC) fabrics. With the growing data volume of AI models, it requires a long simulation time to evaluate NoC performance. Our recent tool SIAM [5] integrates 2.5D silicon bridges with IMC units. Yet SIAM is still too slow to explore diverse design configurations. Furthermore, more advanced 3D design issues, such as network-on-package (NoP), thermal analysis, and reliability prediction, must be incorporated for a comprehensive performance evaluation.

To address these challenges of benchmarking 2.5D/3D HI systems, HISIM adopts a systematic approach. It considers the hierarchy in both computing and data movement, abstracts the relationship between key design variables and performance metrics into analytical models, and validates the results with simulation and silicon data. The major innovations include:

TABLE I
COMPARISON OF STATE-OF-THE-ART CHIPLET AND AI SIMULATORS.

Simulator	Computing Unit	Dimension	Thermal Analysis	Simulation Speed
NeuroSim [36]	IMC	2D/3D	Yes	Slow
MNSIM [33]	IMC	2D	No	Slow
BookSim 2.0 [34]	No	2D	No	Slow
Ratatoskr [35]	No	3D	No	Slow
SIAM [5]	IMC	2.5D	No	Slow
HISIM	**IMC and others**	**2.5D/3D**	**Yes**	**Fast**

- *System-level Performance Modeling*: Instead of an extensive stack of models ranging from devices to circuits and to architectures, we aim to construct a set of analytical models that directly connect algorithmic and design decisions with system metrics, such as power, performance, and area (PPA) of various computing chiplets and NoC/NoP. During the model development, each system metric is decomposed into primary design macros, such as a computing array and its peripherals, following the design structure and data flow. Subsequently, the dependence on each macro is calibrated with realistic design databases and process design kits (PDKs).
- *Model Calibration*: To ensure the scalability and accuracy of system-level models, we perform comprehensive validation with various sources, including design synthesis, numerical simulations, published silicon data, and other simulators. By designing and synthesizing diverse chiplet and network configurations, we extract model coefficients and confirm their scalability of design parameters. These results are further verified by published silicon data in the literature, as well as simulations from rudimentary tools. For electrical parasitic modeling of packaging technologies, TCAD simulations are used for model calibration.
- *Thermal Analysis*: Electro-thermal reliability emerges as a critical issue in compact 2.5D/3D integration. The current version of HISIM integrates a coarse-mesh based finite-element method (FEM) to speed up static thermal prediction. Moreover, an even faster method, leveraging graph neural networks, will be adopted by HISIM for further electrical and mechanical reliability analysis.

The outcome of these efforts is HISIM, an analytical model-based tool for fast design exploration of 2.5D/3D HI systems. By inputting the profile of an AI model and its hardware mapping method, HISIM rapidly predicts end-to-end performance, helping shed light on the capabilities and constraints of algorithm-hardware co-design. Initial experiments on representative DNNs, transformers and GNNs demonstrate a speedup of 10^4–$10^6\times$ in PPA evaluation compared to previous benchmark tools, while preserving model scalability and accuracy. The code of HISIM is released at https://github.com/mec-UMN/HISIM, along with a quick tutorial. The following sections present the technical details of model derivation, validation, and experiments.

II. Background and Related Work

2.5D/3D heterogeneous integration leverages cutting-edge packaging technologies to integrate various chiplet types into a large-scale system. This section reviews relevant topics that lead to the development of HISIM.

A. Chiplet-based Heterogeneous Integration

Chiplet-based heterogeneous integration has emerged as a promising solution to tackle the manufacturing and cost challenges of large-scale monolithic chips, where fabrication costs rise exponentially with increased chip area [5], [37]. Moreover, chiplet architectures offer bandwidth and density advantages over clusters of accelerators [2], improving the performance of various applications [38], [39].

Within a chiplet-based system, multiple chiplets are interconnected via special IO protocols or Network-on-Package (NoP), either within a 2.5D substrate or through 3D connections. For example, Advanced Interface Bus (AIB) [40] is a parallel IO interface boasting a bandwidth of 4Gbps/pin, providing high interconnect density in a 2.5D system. A more recent standard for die-to-die serial buses is Universal chiplet interconnect express (UCIe) [41]. Despite their high IO bandwidth, the number of interconnect links is constrained by the length of the shoreline or die edge.

In contrast, 3D chiplet-based architecture provides significantly higher bandwidth and shorter signal distances between chiplets through various methods, such as face-to-back topology with C4 bumps [42], face-to-face topology with microbumps [26], hybrid bonding [43], or 3D stacking with TSVs. Various 3D implementations have been studied, ranging from logic-on-logic to logic-on-memory [44]. In general, both 2.5D and 3D technologies enable the integration of chiplets manufactured at different silicon technology nodes or even different types of materials [36].

As 2.5D/3D HI significantly enhances the scale, density, and potential of system integration, there is an increasing demand for a comprehensive tool for algorithm-hardware co-design. Emerging topics of interest include the management of diverse computing cores, hierarchical data movement from on-chip to chiplet-to-chiplet and 3D connections, thermal implications [45], power delivery, and workload mapping. This tool should seamlessly incorporate the latest packaging technologies for architecture exploration.

B. In-Memory Computing

To tackle the challenge of extensive memory access during AI computing, in-memory computing has emerged as an advanced accelerator beyond CPUs and GPUs. These accelerators feature a processing element that can be based on either SRAM [46] or non-volatile memory, such as Resistive RRAM (RRAM) [47]. IMC allows for parallel computing of matrix-vector multiplication in analog and digital domains. This operation includes bit-wise multiplication within the memory cell and accumulation along the bit line, achieved through analog resistive/capacitive integration or digital adder trees. For RRAM-based analog IMC, additional peripherals are required, such as ADC and DAC units, bit shifters for weight bit slicing and sequential inputs [47], and MUXs for sharing ADCs across columns. Previous studies, including our research on SIAM [5], have demonstrated simulations and silicon prototypes of IMCs. They are ready for integration into a HI system alongside other types of AI accelerators and more conventional processing cores.

C. Performance Benchmark Tools

There are several benchmark tools for monolithic IMC-based DNN inference and network design, as shown in Table I. These tools usually include IMC crossbars connected by point-to-point interconnection for the on-chip network communication [32], [48]. Neurosim 3D [36] implements monolithic 3D integration (M3D), which partitions memory and logic for RRAM-based IMC, and Neurosim 3D+ [48] extends to heterogeneous 3D integration (H3D) of logic-to-memory with TSVs. However, these works lack the incorporation of logic-on-logic stacking and do not include NoC. Booksim [34] is an NoC simulator, offering the flexibility of network topology, routing, and router architectures. [35] conducts PPA analysis for 3D networks at the RTL level. [5] proposes SIAM that combines IMC circuits, NoC, and NoP for a 2.5D chiplet-based system.

These previous works provide important knowledge of devices, circuits and architecture, especially on IMC design. However, they face long simulation times, which are insufficient to manage big AI models in the vast design landscape of 2.5D/3D integration.

III. HISIM for 2.5D/3D System Analysis

In contrast to the aforementioned benchmark tools, HISIM introduces a suite of analytical models at the system level to speed up performance prediction, covering logic-on-logic architectures across 2D, 2.5D, and 3D integration. Although HISIM currently emphasizes IMC as the main AI accelerator, we plan to evolve HISIM into an open platform capable of simulating other computing, memory, and communication technologies. The potential users of HISIM include, but are not limited to, architecture research on design space exploration before detailed architecture definition, compiler developers to optimize the mapping strategy, and technology developers to benchmark the impact of 2.5D/3D packaging technology on system design. This section elaborates on the benchmark engines in HISIM. Figure 3 illustrates the structure, input, and output of HISIM. The design synthesis and model calibration in this study are conducted at the 32nm/28nm node. Meanwhile, the models and methods in HISIM are scalable to other process technology nodes and assembly design kits (ADKs).

A. Overview of HISIM

Figure 3 overviews the proposed HISIM benchmark tool for evaluating the performance of chiplet-based monolithic (2D), 2.5D, and 3D architectures. Users can define the technology and AI algorithms as inputs to HISIM. On the technology aspect, users have the flexibility to select the type of computing

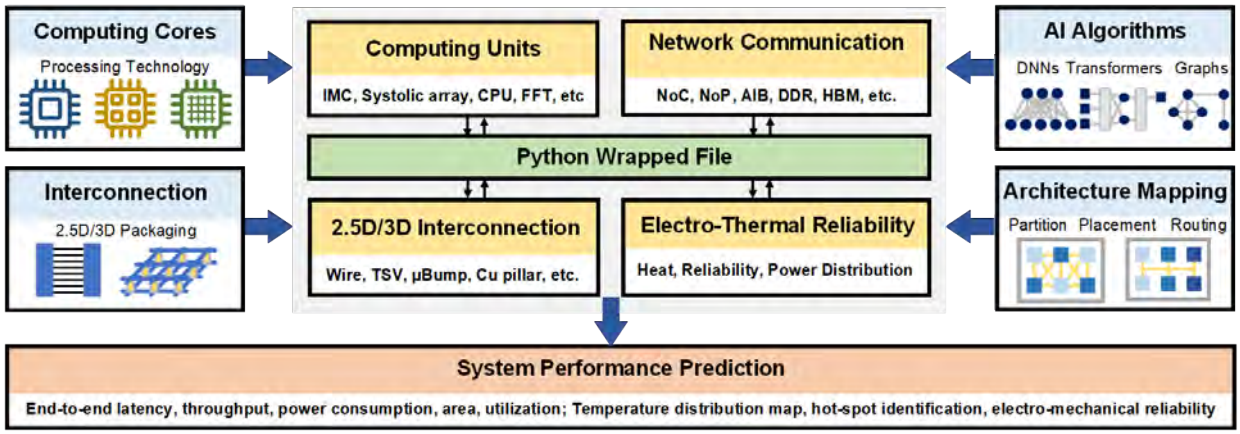

Fig. 3. Overview of HISIM for 2.5D/3D systems. HISIM comprises four engines: computing, interconnection, network and reliability. It integrates inputs from technology, design configuration, AI algorithms and mapping methods to predict system performance metrics, such as PPA and the thermal map.

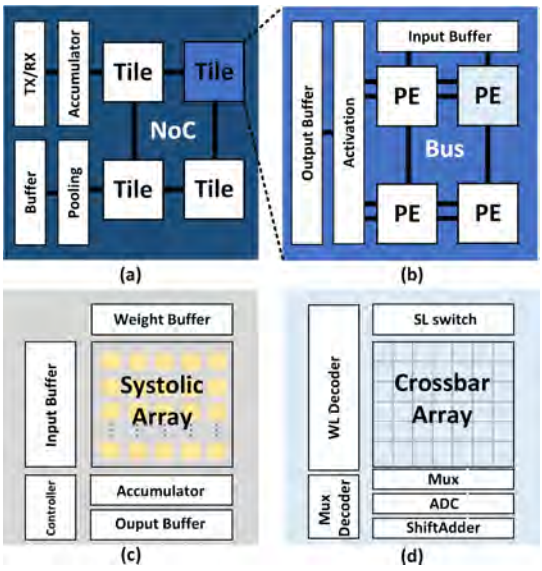

Fig. 4. The hierarchy of the heterogeneous chiplet (a), tile (b), systolic array PE (c), and IMC PE (d); Users can choose the type of chiplet, tile, and PE in HISIM.

cores, interconnect options (such as 2.5D or 3D), and hardware configurations (such as array size and the number of tiers in 3D stacking). On the algorithm aspect, HISIM supports various DNNs, GNNs, and transformers.

In addition, users need to customize the strategy for workload mapping and data flow, such as algorithm partition, task scheduling, data reuse, routing, placement, etc. Based on user inputs, HISIM generates the specific heterogeneous chiplet-based architecture and analyzes area, power estimation, latency, utilization and energy efficiency. The combination of the HI structure and power density is further used to produce the thermal map.

B. Computing Cores

Computing cores are vitally important to processing AI tasks. HISIM currently features analytical models for IMC-based computing chiplets using RRAM-based crossbars and systolic arrays which is shown in Figure 4 (c). The methodology will be generalized to digital IMC and other types of processing elements (GPU, CPU, etc.) in the future.

1) IMC Tiles and Mapping: Figure 4 (a)(b)(d) presents the design levels within an IMC chiplet, with related input and output parameters defined in Table II. Each chiplet contains multiple tiles, N_{tile}, and peripherals connected by NoC. An IMC tile comprises several processing elements (PEs) denoted as N_{pe}, along with the accumulation module responsible for adding partial sums from PEs, input/output buffers for storing activations, and a data bus facilitating data communication within the tile. Additionally, the tile incorporates nonlinear activation modules, such as ReLU, for the activation function of each layer within the neural network. These PEs and macros are interconnected via the databus. This architectural configuration is consistent with other simulators discussed in previous studies [32], [33] and is applicable to other array-based computing cores (such as the systolic array).

The PE is the core computing unit in a chiplet. The PE design depends on the specific processing technology to model. As an example, an analog IMC-based PE has one IMC crossbar along with peripheral circuitry, including drivers, multiplexers, and ADCs. The IMC crossbar performs computations by executing the multiply-and-accumulate (MAC) operation.

In HISIM, rather than limiting users to a fixed compilation method, HISIM aims to support multiple types of compilation approaches, allowing users to optimize their designs as needed. Users can define the partition and mapping strategy for the weights of an AI model onto the IMC crossbar, PE, and tile. In this study, we employ the conventional DNN weight partition

TABLE II
INPUT PARAMETERS OF IMC CHIPLET MODELS.

Inputs	Level	Description
In_x,In_y	AI model	Input size of each layer
K_x,K_y		Kernel size of each layer
F_{in},F_{out}		Input channel, output channel
Q_w,Q_a		Quantization bit of weights, activation
S		Weight sparsity
st		Stride of each layer
Pooling		Followed by pooling or not
N_{tier}	Chip	Number of chiplets or 3D tiers in the system
N_{tile}	Tier	Number of tiles per tier (chiplet)
N_{pe}	Tile	Number of PE in Tile
X_{bar}	PE	Crossbar size
C_i	Mapping	Number of parallel IFM computes
C_w	Mapping	Weight duplication factor within a crossbar

and mapping method to map the weights of AI models, as outlined below:

$$total_{xbar} = \left\lceil \frac{Bit_{col}}{X_{bar}} \right\rceil \times \left\lceil \frac{Bit_{row}}{X_{bar}} \right\rceil \quad (1)$$

$$total_{tile} = \frac{total_{pe}}{N_{pe}}; total_{pe} = \frac{total_{xbar}}{N_{xbar}}; \quad (2)$$

$$Bit_{col} = K_x \times K_y \times F_{in}; Bit_{row} = F_{out} \times Q_w \quad (3)$$

2) Performance Modeling: The performance of the chiplet component is modeled using the design structure and parameters in Table II as follows:

$$L_{chiplet} = L_{pooling} + L_{accum} + L_{noc} + \sum_i L_{layer_i} \quad (4)$$

$$E_{chiplet} = E_{pooling} + E_{accum} + E_{noc} + \sum_i E_{layer_i} \quad (5)$$

Each layer of an AI model is mapped to a set of crossbars using conventional weight partitioning and mapping method described above. The cumulative performance of crossbars mapped for a layer of the AI algorithm is modeled as follows:

$$L_{layer_i} = Lmax_{xbar} \frac{N_{act} Q_{act}}{C_i C_w (st^2)} \quad (6)$$

$$E_{layer_i} = Emax_{xbar} \frac{N_{act} Q_{act}}{st^2} \frac{2 N_W Q_W}{(xbar_x)(xbar_y) st^2} \quad (7)$$

In these equations, N_{act}/st^2 represents the number of convolutional windows, while N_W denotes the total number of weight parameters. The value of N_W varies depending on the layer type, whether it is convolution (CONV), fully connected (FC), or depthwise convolution (DW). Assuming a synchronous design, the maximum latency of a layer depends on the crossbar with the highest latency, multiplied by the number of cycles of crossbar computation. The number of cycles, in turn, is influenced by several factors. These factors include the number of convolution windows, serial computation of input bits, parallel computation of input feature map (IFM), and weight duplication, which is a technique employed for Depthwise convolution [49].

The energy model is derived from its dependency on the sum of the multiplication of factors, such as the maximum energy of the crossbars, the crossbar utilization, and the number of times each crossbar is computed. We then derive the above equation by summing the crossbar utilization, which depends on the cumulative utilization of memory cells. This equation is obtained assuming all the crossbars inside a layer are computed the same number of times, similar to the mapping specified in [49], and the input sparsity is 0%. Additionally, a factor of 2 is considered for signed weights, assuming a double row configuration as detailed in [49].

The maximum latency of a crossbar is influenced by the crossbar mapped with maximum utilization, which further relies on the mapping and the number of output feature maps (OFMs), typically a multiple of the ADC sharing factor for neural networks. Moreover, the maximum energy of a crossbar is influenced by factors such as the current of each cell, read voltage, the total number of cells in the crossbar, and activity factor due to ADC sharing. The area of the tile is estimated based on the crossbar size and the number of processing elements.

TABLE III
COMPARISON WITH REFERENCE RRAM BASED SYSTEM EVALUATED USING DEVICE MEASUREMENT DATA AND SIMULATION [49].

Parameter	HISIM (65nm)	65nm data [49]
VGG16		
Total Crossbar Area	49.9 mm^2	50.7mm^2
Total Energy	6.74 mJ	5.94 mJ
Total ADC Area	261.5 mm^2	261.5 mm^2
Total ADC Power	17.43 W	17.43 W
Mobilenet		
Total Crossbar Area	13.4 mm^2	13.7mm^2
Total Energy	1.97 mJ	1.29 mJ
Total ADC Area	56.544 mm^2	56.54 mm^2
Total ADC Power	3.77 W	3.78 W

3) Model Calibration: The IMC chiplet models are calibrated with realistic designs to reliably predict the physical metrics of individual design components, such as area, latency, and power consumption. These values are then combined into the system performance model. Initially, the features of RRAM devices and ADCs are extracted from measurement data at 65nm [49], and then scaled down to 28nm, following a scaling factor of 4.12 for area and 3.37 for power as obtained from [50]. HISIM offers a discrete number of crossbar sizes for users to select, such as 64×64, 128×128, 256×256, and 512×512. Other peripherals of the crossbar, such as multiplexers and decoders, are synthesized, similar to those used in [51]. At the PE and tile levels, data bus, NoC, buffers, etc., are included, following those in SIAM [5].

Figure 5 assesses the computing latency modeled by HISIM using measurement data from [49] for an RRAM-based IMC design. Two DNN algorithms, VGG16 [52] and MobileNet [53], are used for evaluation. The inputs to HISIM, including mapping dependency, the number of ADCs per layer, and ADC sharing, are configured the same as those in [49]. For example, the filter in a convolution layer is flattened into

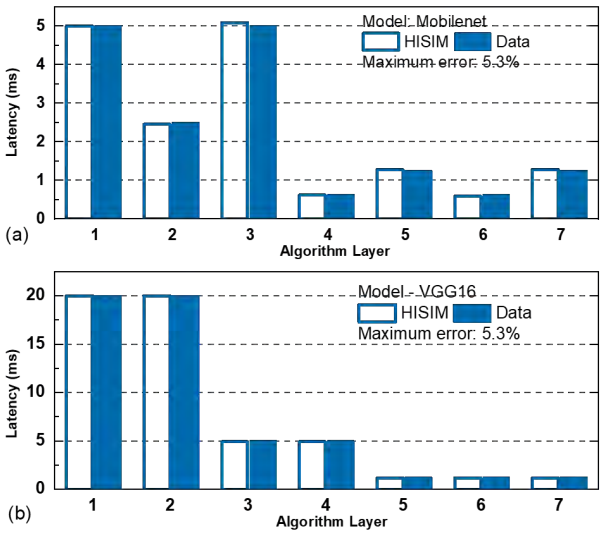

Fig. 5. Computing latency comparison of first seven layers of two DNNs, Mobilenet (a) and VGG16 (b), with 65nm measurement data in [49].

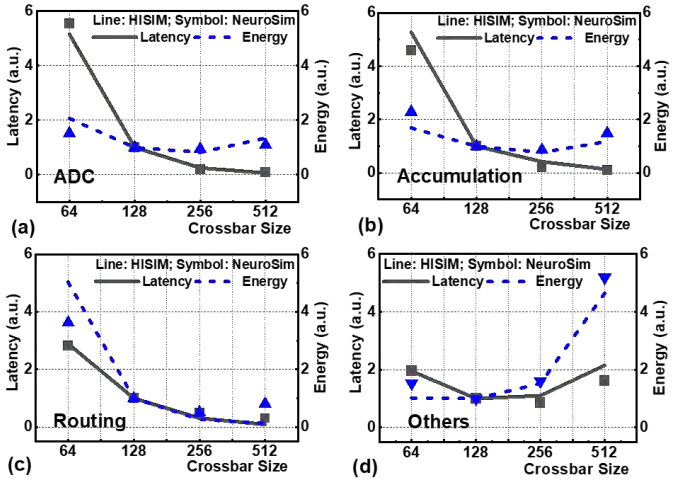

Fig. 6. Comparison of normalized latency and energy with NeuroSim for ResNet110 at 32nm.

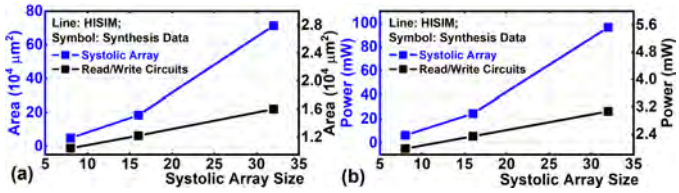

Fig. 7. Comparison of area and power for array and read/write circuits with different systolic array sizes under 28nm.

a 1D vector and mapped onto a column. If the vector does not fit into a single crossbar, it is distributed across multiple crossbars. Additionally, if the number of crossbars required for the layer exceeds the capacity of a single processing element (PE), the layer is mapped onto multiple PEs.

The maximum latency of crossbars mapped to the first layer of VGG16 is first calibrated to that of the reference data. The results are then used to predict the performance of the first seven layers of VGG16 and Mobilenet, as presented in Figure 5, with the maximum error at 5.3%. Table III verifies the total area and energy consumption of both algorithms. The energy values in Table III account for both network energy and computing energy. However, the network configuration in HISIM differs from that in [49], resulting in variations in energy comparison. Furthermore, we evaluate the scalability of HISIM models with design variables, such as crossbar size. In this study, we utilize simulations with NeuroSim to validate the nonlinear dependency of peripheral circuits, such as ADC, accumulation, routing, and other peripherals. Figure 6 summarizes the normalized results. At the crossbar level, our system-level models exhibit comparable scalability with NeuroSim in RRAM-based IMC design.

4) Systolic Array & Other Computing Units: Currently, HISIM supports the systolic array which is calibrated with Scale-Sim [54] and 28nm synthesis which is shown in Figure 7. In addition, we are expanding HISIM towards other types of computing cores, such as digital IMCs and CPUs, to enable more heterogeneous computing. For digital IMCs, we will synthesize the design and extract performance models, which include the array structure and local buffers for storing and loading partial results. We will incorporate the average time consumed in each stage of the pipeline in the CPU model, including fetch, decode, execute, and writeback. We will calibrate the model with silicon data, such as that from RISC-V cores. Furthermore, we plan to open the HISIM platform, allowing other researchers to add their own chiplet models for system exploration.

C. Heterogeneous Interconnection

In AI computing, data movement plays a crucial role in the overall performance of the system. Emerging 2.5D/3D interconnection technologies, with their scaling trend in Figure 2, promise high bandwidth, high connection density, and low energy consumption per bit. The interconnection engine in HISIM integrates compact models that calculate electrical parasitics for various types of interconnections.

1) 2.5D/3D Interconnection: Figure 8 presents our modeling process, from various interconnect technologies to their geometry definition and the extraction of electrical parasitics. The parasitics are transferred to the network engine for latency and power analysis. Table IV lists the key dimensions of the metal wire for monolithic, 2.5D, and 3D interconnections. The parameters l_{wire}, w_{wire}, t_{wire}, and p_{wire} define the geometry of interconnections. They are key to parasitic modeling. For 3D interconnection, many technologies, like μbump, hybrid bonding, and TSVs, enable data communication between tiers vertically. The features of a TSV is defined by d_{TSV}, h_{TSV}, p_{TSV} and t_{ox_TSV}, as shown in Figure 8. As TSV technology continues to scale down, its dimensions become increasingly comparable to those of on-chip wires, implying a similar bandwidth and network capacity through the vertical TSVs as with the on-chip network. Potential users can adjust the dimensions of the interconnects in HISIM according to their specific technology.

2) Parasitic Modeling: For wires in 2D and 2.5D technologies (such as the silicon bridge), we follow the models in the Predictive Technology Model [55] and the roadmap in [27] to

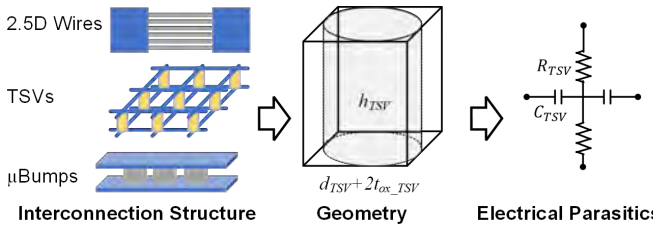

Fig. 8. The workflow of parasitic extraction, including 2D wires, 2.5D wires, 3D TSVs, μBumps, etc.

TABLE IV
MODEL PARAMETERS OF 2D/2.5D/3D INTERCONNECTIONS.

Structure	Interconnect	Parameter	Description
2D/2.5D	Wire	l_{wire}	Wire length
		w_{wire}	Wire width
		t_{wire}	Wire thickness
		p_{wire}	Wire pitch
3D	TSV	d_{TSV}	TSV diameter
		h_{TSV}	TSV height
		p_{TSV}	TSV pitch
		t_{ox_TSV}	Insulation thickness

TABLE V
SCALING OF TSV PARASITICS IN SIX GENERATIONS.

r_{TSV} (μm)	d_{TSV} (μm)	h_{TSV} (μm)	R_{TSV} ($m\Omega$)	C_{TSV} (**fF**)	RC (**fs**)
20	40	400	5.45	888.76	4.843
15	30	300	7.26	502.04	3.645
10	20	200	10.89	225.00	2.450
5	**10**	**100**	**21.78**	**57.64**	**1.255**
2.5	5	50	43.56	15.09	0.657
1.25	2.5	25	87.12	4.10	0.357

calculate their RC parameters. For a TSV link, we assume the TSV is driven by the driver and followed by the receiver. The TSV is then modeled as the resistor (R_{TSV}) and capacitor (C_{TSV}) between the driver and the receiver. We adopt the models in [56] to construct the unit cell, as illustrated in Figure 8. R_{TSV} and C_{TSV} are formulated as:

$$R_{TSV} = \frac{1}{2}\frac{h_{TSV}}{\sigma_{TSV}\pi(d_{TSV}/2)^2} \tag{8}$$

$$C_{TSV} = \frac{1}{2}\frac{\pi\varepsilon_{ox}h_{TSV}}{ln(\frac{d_{TSV}/2+t_{ox\_TSV}}{d_{TSV/2}})} \tag{9}$$

These parasitic models are usually derived from the distribution of the electric field, and validated by finite-element solvers.

3) Interconnect Scaling: HISIM covers multiple generations of interconnect dimensions to facilitate early design exploration. For example, Table V illustrates the TSV roadmap, including the dimensions and RC parasitics [56]. As the size of TSVs continues scaling down, their RC product decreases by more than $10\times$ over these six generations, significantly improving data bandwidth. In addition, the reduction in C_{TSV} contributes to lower energy consumption in data movement. In this paper, we choose the generation with $5\mu m$ TSV radius for simulation analysis.

Currently, HISIM only addresses RC parasitic. Users can also substitute the HISIM RC model with their own parasitic models for simulation. We also plan to extend the modeling effort to inductance and other components to support signal integrity analysis at higher frequencies.

D. Network Communication

For AI acceleration, efficient data communication is critical for the overall system performance. With large volumes of input and output activations being transferred between adjacent algorithm layers, a high-bandwidth, flexible network is required to connect multiple PEs, tiles, and chiplets. This can be a network-on-chip for a 2D monolithic chip or a network-on-package in 2.5D and 3D systems. As the chiplets communicate through the NoP, the network size is directly influenced by the number of chiplets. The size of the AI model's weights and the hardware configuration together determine the NoP size which influences the performance of the network. A 2D NoC or a 3D NoP consists of multiple channels connected by network routers. Figure 9 presents an example of a 3D network. HISIM models 2D/3D network latency and power consumption in the

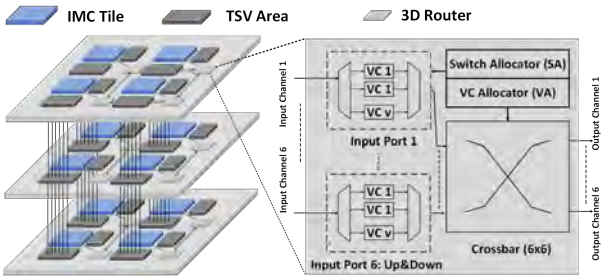

Fig. 9. A 3D network via TSVs and routers.

TABLE VI
INPUT PARAMETERS OF NETWORK MODELS.

Inputs	Description
VC router	
t_{RC}	Delay in routing computation
t_{VA}	Delay in virtual channel allocation delay
t_{SA}	Delay in switch allocation
t_{ST}	Delay in switch traversal
t_L	Delay in channel traversal
t_{enq}	Queuing delay
Channel	
Q_{2d}, Q_{3d}	Data volume for 2D/3D routing
W_{2d}, W_{3d}	2D/3D channel width
Network	
H_{2d}, H_{3d}	2D/3D hop count

network engine for various network topologies. In addition, HISIM covers the IO protocol in 2.5D silicon bridges for chiplet integration.

1) 2D/3D Network Modeling: Cycle-accurate simulators are available for the 2D NoC latency and throughput evaluations, such as Booksim [34]. BookSim traces the data flow through each module. Such a cycle-accurate process requires running simulations with each input data point, leading to a long simulation time. To speed up the simulation while maintaining the accuracy in latency prediction, we generalize BookSim to both 2D and 3D networks and formulate an analytical model for network latency based on equations in [57]:

$$L_{NoC} = (H_{2d} + H_{3d}) \times t_{router} + t_{enq} \times \frac{Q_{2d}}{W_{2d}} + t_{enq} \times \frac{Q_{3d}}{W_{3d}} \tag{10}$$

$$t_{router} = t_{RC} + t_{VA} + t_{SA} + t_{ST} + t_L \tag{11}$$

where H_{2d}, H_{3d} are the hop count from the NoC network for 2D and 3D, respectively. t_{router} is the latency associated with the routing computation, switch allocation, etc., as explained in Table VI. This equation represents the contention-free message latency for the 3D virtual channel router, as shown in Figure 9.

We validate the latency equations with BookSim, as shown in Figure 10. The network trace files are derived from VGG-16 on ImageNet, mapped to 2.5D/3D integration. The prediction from our analytical equation accurately follows the cycle-accurate results from BookSim. As shown in Figure 9, mapping an entire AI algorithm to a 3D stack involves both 2D NoCs (within-tier or on-chip) and 3D NoPs (cross-tier). To calculate the total hops, we sum the H_{2d} and H_{3d} using Algorithm 1. Our analytical model significantly accelerates

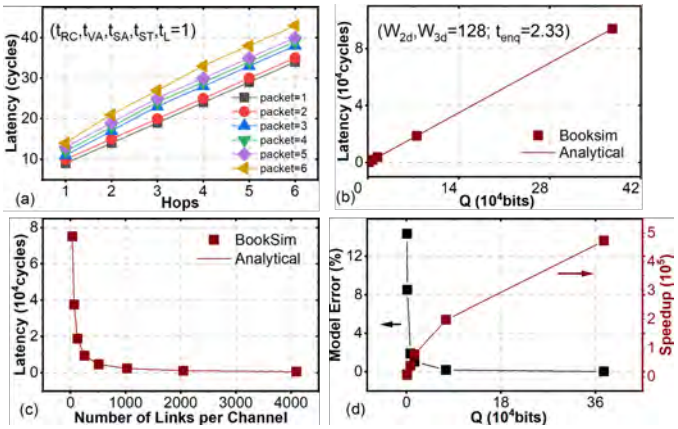

Fig. 10. Calibration of analytical network with cycle-based BookSim simulations: (a) Latency with different sizes of packets; (b) Latency with different data volumes; (c) Latency with various channel bandwidths, represented as the number of links per channel; and (d) Trade-off between model errors and speedup of calculation.

Algorithm 1 Counting 2D/3D hops and data bits for the entire AI model

Input: Number of tiles for each layer (T), Number of input activations for each layer (A), Number of AI model layers (L)
Output: H_{2d}, H_{3d}, Q_{2d}, and Q_{3d};
for *l=1:L* **do**
  Based on the placement method, decide the location of each tile in the 3D system (x_l,y_l,z_l)

  Divide activations A into A_{2d}, A_{3d}
  Q_{2d}+=A_{2d}; Q_{3d}+=A_{3d}
  for *t=1:T* **do**
    Find the position for the source tile in this layer
    Find the position of the destination tile in next layer
    $2d_{hop}$=|x_l-x_{l+1}|+|y_l-y_{l+1}|; $3d_{hop}$=|z_l-z_{l+1}|
  end
  H_{2d}+=$2d_{hop}$; H_{3d}+=$3d_{hop}$
end

latency calculations and achieves high accuracy across typical data volumes.

We adopt the analytical models from Orion 2.0 [58] to assess the power consumption and area of 2D/3D networks:

$$P_{NoC} = P_{2d} + P_{3d} \tag{12}$$

$$P_{2d} = (E_{2d_{cl}} + E_{2d_r})/L_{2d_{NoC}}; P_{3d} = (E_{3d_{cl}} + E_{3d_r})/L_{3d_{NoC}} \tag{13}$$

where P_{NoC} is the total power consumption of the network, including 2D NoCs (within-tier or on-chip) and 3D NoPs (cross-tier). The power consumption in 2D/3D networks relies on energy consumption and associated latency of each component. $E_{2d_{cl}}$, $E_{3d_{cl}}$ represents the energy consumption of 2D/3D channels, and E_{2d_r}, E_{3d_r} represents the energy consumption of 2D/3D routers. The overall power consumption of the 3D network is based on the power consumption of the channels and routers, as well as the ratio of their active time in L_{NoC}. Moreover, the dynamic power of both channels and routers depends on the generation of interconnection technologies, as presented in the previous section.

2) 2.5D AIB Interface: To enable inter-chiplet communication within the 2.5D package, a dedicated signaling interface is necessary to maintain signal integrity. These interfaces, such as Advanced Interface Bus (AIB) [40] or Universal chiplet interconnect express (UCIe) [41], are usually standardized by the design industry. Figure 11 illustrates the AIB 2.0 interface that is modeled in HISIM. Each chiplet can accommodate up to four columns of AIB, each supporting multiples of 4 channels and up to a maximum of 24 channels. Each AIB channel features a balanced transceiver (Tx)-receiver (Rx) configuration and can handle data widths ranging from 20 to 80 for each Tx and Rx channel. Furthermore, each channel is equipped with a Tx and Rx adapter and IO modules. The number of IO ports in a channel is determined by various factors, including the number of AIB Tx/Rx lines and essential IO signals such as clock, control, and sideband signals. Additionally, the adapter incorporates a Tx and Rx FIFO, each of which can be configured to operate at full, half, or quarter rates by adjusting the clock frequencies.

To assess the power and performance of the AIB 2.0 interface, we construct an analytical model based on input parameters detailed in Table VII. This model is summarized as follows:

$$\text{Latency: } L_{Tx} = L_{FF_{wr\_clk}} + L_{FIFO} + L_{FF_{fwd\_clk}} + L_{IO} \tag{14}$$

$$\text{Latency: } L_{Rx} = L_{IO} + L_{FIFO} + 2L_{FF_{rd\_clk}} \tag{15}$$

$$\text{Dynamic Power: } P_{AIB} = (P_{clocks} + P_{data})n_{ch} \tag{16}$$

Here, the dynamic power due to clock switching and data switching is further decomposed as:

$$P_{clocks} = P_{fs\_fwd\_clk} + P_{ns\_fwd\_clk} + P_{rd\_clk} + P_{wr\_clk} \tag{17}$$

$$P_{data} = P_{Txdata} + P_{Rxdata} \tag{18}$$

The dynamic power arising from each clock-switching event of a single AIB channel is estimated based on several factors, including the number of Tx/Rx data lines ($n_{tx/rx}$), voltage, specific clock frequency, and the fraction of duration for which the AIB is switched on (α_{aib}), representing the activity factor.

$$P_{clock_i} = (\gamma_{1_i} n_{tx/rx} + \gamma_{2_i}) V^2 \alpha_{aib} f_{clk_i} \tag{19}$$

where γ_{1_i} and γ_{2_i} are calibrated using RTL synthesis at 28nm of open-source AIB codes [59]. Similarly, the dynamic power originating from data switching for a single AIB single channel is assessed, depending on the input switching frequency (f_{in_i}) and an input switching activity factor that is a function of input sparsity (S_{in}).

The area of the AIB interface circuitry is estimated, assuming uniform dimensions for each component, with all IO

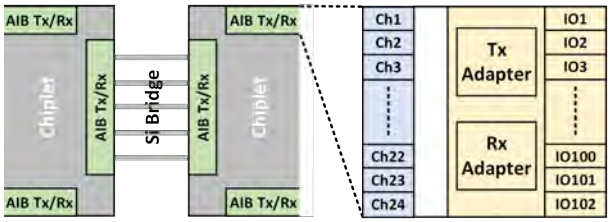

Fig. 11. The architecture of the 2.5D interface, such as AIB, includes Tx/Rx adapters, silicon bridge, etc.

cells aligned along the same edge as the μbump orientation, as specified in [40].

$$\text{Area: } A_{AIB} = (Len_{IOch}(W_{IO} + W_{Adpt}))n_{ch} \tag{20}$$

Here, Len_{IOch} represents the length of the stack of IO cells in a single column, while W_{IO} and W_{Adpt} denote the width of an IO cell and an adapter, respectively.

$$Len_{IOch} = n_{IO}\sqrt{A_{IO}}, W_{IO} = \sqrt{A_{IO}} \tag{21}$$

$$W_{Adpt} = \sqrt{A_{TxAdpt} + A_{RxAdpt} + A_{Other}} \tag{22}$$

where the area of the Tx and Rx adapter relies on the number of Tx/Rx lines:

$$A_{TxAdpt} = \delta_1 n_{Tx} + \delta_2, A_{RxAdpt} = \delta_3 n_{Rx} + \delta_4 \tag{23}$$

In this context, A_{IO} denotes the area of a single IO cell. The area of the μbump array is also estimated, depending on the number of IO signals, channels, number of IOs per column, and row and column pitches:

$$A_{\mu bump} = n_{ch} n_{IOcl} p_{cl} \frac{p_{rw}}{2} ceil(\frac{n_{IO}}{n_{IOcl}}) \tag{24}$$

Different values are provided for low, medium and high-density μbump arrays [40].

All AIB equations within HISIM are calibrated with RTL synthesis at 28nm [59] to extract latency, area, and power parameters. Table VIII evaluates HISIM results with silicon data [60]. The latency and power models within HISIM are developed under a Double Data Rate (DDR) configuration across three distinct modes of operation: full, half, and quarter rate. The selection of these three modes directly dictates the aggregated data width of the Tx/Rx module [40]. Note that the FIFO within the AIB adapter operates asynchronously. HISIM adopts the worst-case latency over the temporal span of the data stream.

Besides the AIB interface, HISIM further incorporates analytical performance models for the silicon bridge between chiplets, as shown in Figure 11. The area of the silicon bridge is a function ofthe parameters in Table VII, and is estimated under the layout configuration in [61]:

$$A_{wire} = (\frac{p_{rw}}{2n_{IOcl}} n_{IO} + W_{wire}) L_{wire} n_{ch} \tag{25}$$

With *RC* parasitics calculated by the interconnect engine, the latency of the bridge is calculated using the Elmore delay model. The dynamic power depends on the frequency of all signals transmitted between AIB interfaces:

$$P_{wire} = \sum_i \alpha_i f_i \sum_j C_j V^2 n_{ch} \quad \forall i \in [1, n_{IO}] \tag{26}$$

where α_i represents the activity factor of signal i, and f_i denotes the frequency of switching for signal i.

3) Other Chiplet Interfaces: While AIB is currently available, we plan to integrate additional types of 2.5D/3D interfaces into HISIM. Examples are UCIe, DDR, and HBM, which will further address data communication between computing chiplets and external memory. Furthermore, we will continually update the HISIM models to accommodate various generations of μbumps, TSVs, and wire lengths and pitches. These improvements will enable a more comprehensive and accurate representation of future packaging systems and their performance characteristics.

TABLE VII
INPUT PARAMETERS OF THE AIB MODEL.

Inputs	Description
Clock	
ns_fwd_clk	Near side clock of Tx IO module
fs_fwd_clk	Far side clock of Rx IO module
rd_clk	Read clock of Tx FIFO
wr_clk	Write clock of Tx FIFO
Interface	
n_{ch}	Number of AIB Channels
n_{IO}	Number of IO ports per channel
n_{Rx}	Number of Rx lines per channel
n_{Tx}	Number of Tx lines per channel
μbump	
n_{IOcl}	Number of AIB IOs per μbump column
p_{rw}	Aligned-row bump-to-bump pitch
p_{col}	Aligned-column bump-to-bump pitch
Silicon bridge	
W_{wire}	Width of 2.5D wires
L_{wire}	Length of 2.5D wires

TABLE VIII
COMPARISON WITH SILICON DATA AT 40 TX AND 40 RX PINS.

Parameter	HISIM	Data [60]
AIB Configuration		
Technology Node	$28nm$	$22nm$
Aligned-row bump-to-bump pitch	$36\mu m$	$36\mu m$
AIB Data rate ($Gbps/pin$)	4	4
Number of AIB Channels (n_{ch})	24	24
Number of AIB IO ports (n_{IO})	102	102
Total AIB Bandwidth ($Tbps$)	7.68	7.68
Area Efficiency		
Total AIB Area (mm^2)	5.7	4.5
Bandwidth Density ($Tbps/mm^2$)	1.35	1.705
IO Buffer Size($\mu m^2/pin$)	100	91.2
Energy Efficiency		
IO Latency (ns)	1.45	1.5
Adapter Energy (pJ/b)	0.29	0.32

E. Thermal Analysis

Thermal analysis is critical to heterogeneous integration, especially in 3D stacking scenarios. The stacking of multiple tiers inevitably reduces the capability of heat dissipation and raises the temperature. As a consequence, degrading the electromechanical reliability of an HI system. Using power consumption data and chip area, we build up the thermal resistance model to generate the temperature map for a 2.5D/3D system. Our thermal models offer a user-friendly interface that allows the integration with other simulators, enabling them to incorporate power and area results for comprehensive thermal analysis.

1) Inputs to Thermal Analysis: Figure 12 presents the process in HISIM. The thermal analysis engine requires the following inputs to accurately predict temperatures.

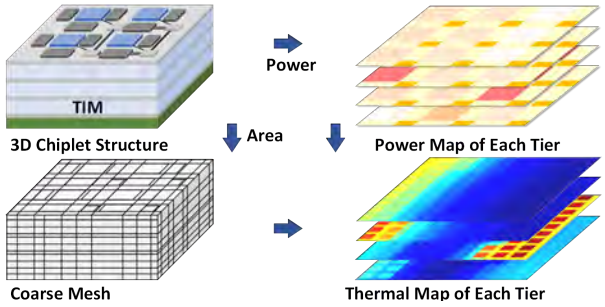

Fig. 12. Thermal prediction for a 2.5D/3D system: HISIM first generates the power and area maps for each component, based on the algorithm workload and the system structure; the thermal engine then computes the temperature map using a coarse mesh.

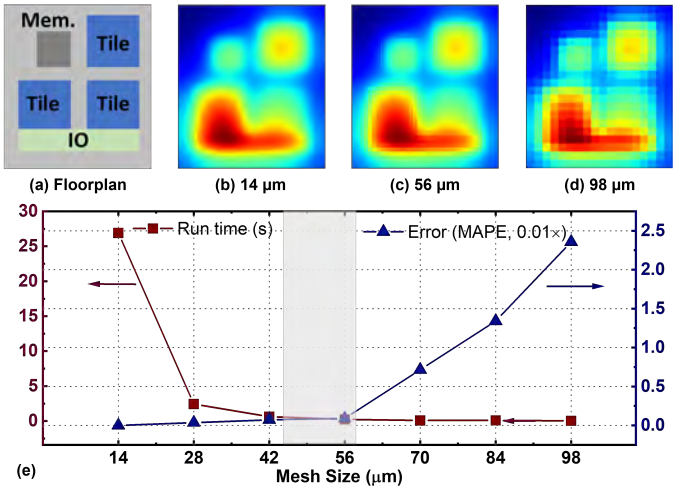

Fig. 13. (a) The structure of a chiplet, at $3 \times 3mm^2$; (b-d) Temperature maps at different mesh sizes; (e) Trade-off between simulation time and MAPEs.

Geometry and material information includes information about the structure (e.g., dimensions and placement) and materials (e.g., the thermal conductivity of the chiplet and thermal interface materials) that constitute the chiplet system.

Power map includes the power profile of computing units and networks that contribute to heat generation.

Granularity defines the resolution of the mesh used for partitioning the system. A smaller mesh size improves accuracy but compromises simulation speed.

Boundary conditions include the ambient temperature, usually set at $298K$, and additional heat dissipation elements like the heat sink, substrate, and the surrounding air layer.

2) Finite Element Method: Mathematically, the first two inputs described above are represented by two 3D arrays: $\mathbf{K}$ and $\mathbf{P}$, corresponding to the conductivity map and power map, respectively. Each entry within the array corresponds to a voxel, which serves as the smallest unit for thermal analysis. The physical size represented by the voxel is determined by the granularity c_l, c_w, c_h which are for length, width, and height, respectively. To conduct the thermal analysis, we use the nodal analysis $\mathbf{GT} = \mathbf{P}$ to predict the static thermal map $\mathbf{T}$, given the flattened power map $\mathbf{P}$ and the conductance matrix $\mathbf{G}$ [67]. The conductance g_{ij} between two physically connected voxels v_i, v_j is evaluated by

$$g_{ij} = -A_{ij}/(\frac{l_i}{k_i} + \frac{l_j}{k_j})) \tag{27}$$

where A_{ij} is the contact area, l_i, l_j are the distances from the center of each voxel to the contacted region, and k_i, k_j are the corresponding conductivities. The diagonal of $\mathbf{G}$ is evaluated by $g_{ii} = -\sum_j g_{ij}$. After building up the model, we collect data points from publications and calibrate the thermal properties of different materials in the model. Table IX summarizes the calibration.

TABLE IX
CALIBRATION OF HISIM WITH PUBLISHED 3D SYSTEMS.

Die Size (mm^2)	Tier Count	Power Density (mW/mm^2)	Peak Temp. (°C)	HISIM (°C)
100 [62]	2	2.85	27	26.78
36 [63]	3	27.78	68.6	69.25
9 [64]	3	38.99	47	41.09
49 [65]	4	40.81	53.85	56.18
22 [66]	1	140	53	43.38

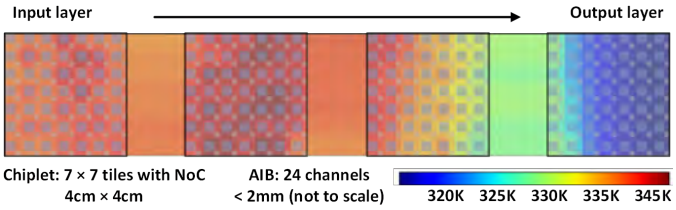

Fig. 14. A 2.5D system for DenseNet-121. Each chiplet consists of 7×7 computing tiles and NoC routers, at an area of $4 \times 4cm^2$. They are connected by AIB. The thermal simulation uses a mesh size of $60\mu m$.

In simulations with the finite-element method (FEM), the selection of granularity (i.e., mesh size) is essential to both accuracy and simulation efficiency. To illustrate that, we simulate a chiplet consisting of three computing tiles, one SRAM, and one AIB. The total chiplet area is $3mm \times 3mm$. The system also has a heat sink on the top, a packaging substrate, and an air boundary layer. Figure 13 presents the trade-off between simulation time and accuracy, with results from a fine mesh size, such as $14\mu m$, as the baseline. As shown in the figure, a mesh size of $50 - 60\mu m$ is found to be optimal for this chiplet configuration before the mean absolute percentage error (MAPE) starts increasing. This value aligns with the results in [68]. A second example is presented in Figure 14. It demonstrates a 2.5D system with four IMC chiplets for end-to-end computation of DenseNet-121. Although both computation and data volume are higher in the middle layers, the workload decreases towards the final output, leading to a cooler temperature.

3) Machine Learning Method: Despite using a coarse mesh, numerical simulations with FEM still consumes considerable time. To speed up thermal analysis for design space exploration, our ongoing research focuses on machine learning techniques. Based on the same mesh size that is consistent with FEM, we propose to develop a graph neural network (GNN) to simulate the heat exchange within a chiplet system. The training of GNN is assisted by FEM results. The inference of GNN involves the operations of aggregation and transformation, which are confined to each node. Therefore, the computation scale of GNN is much smaller compared to FEM, which operates on a full 3D matrix. This advantage promises a significant acceleration in temperature prediction. Such a

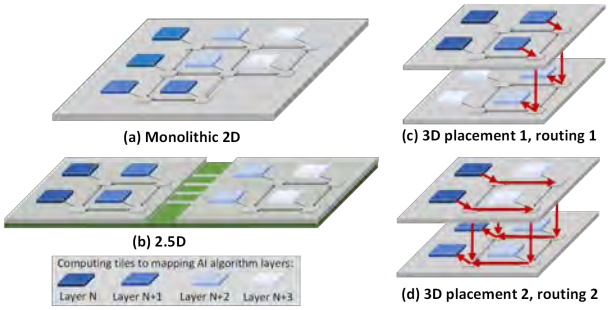

Fig. 15. Placement and routing for 2D/2.5D/3D systems. The 3D system offers more options due to the vertical link.

thermal model has been developed, submitted for publication, and will be integrated into HISIM in the future.

IV. Design Evaluation with HISIM

The primary objective of HISIM is to provide an end-to-end simulator for chiplet-based heterogeneous integration (HI). This simulator needs to be comprehensive to cover a wide range of chiplet architectures and interconnection technologies, ranging from monolithic to 2.5D and 3D structures. By varying the type of computing units and network topologies, users are able to conveniently simulate and explore the design space for further optimization.

A. Experimental Setup

In our experiments, we simulate a set of representative AI models with HISIM, including ResNet-110 (1.7M parameters) on CIFAR-100, DenseNet-121 (7.05M parameters) on ImageNet, a 2-layer graph convolutional network (GCN) on Cora, and a vision transformer (ViT, 86M parameters) on ImageNet. We quantize the weights and activations of AI models to 8-bit. All experiments were performed on the Intel Xeon CPU platform. We use the RRAM-based in-memory computing chiplet for the computing core, as specified in Section III-B1. The default technology node is 32nm/28nm CMOS. We follow the method in Section III-B1 for the partition of algorithms on the IMC crossbar array, assuming stationary weight mapping to the crossbars. Both computing units and networks operate at a default frequency of 1GHz. For the 3D chiplet-based architecture, we follow that in [21] to construct the stack. We adopt the Face-to-Back configuration to build up the 3D multi-chiplet systems with TSVs. The maximum number of tiers is four in this study. The 3D network routers in each tier support data communication both within the chiplet as well as vertically to other chiplets via TSVs. The default configuration includes three virtual channels with a buffer size of 10 per virtual channel. The packet size is 1 and the flit size depends on the channel width. The TSV array consists of 70% signal TSVs and 30% power/ground TSVs [21]. The default diameter for each TSV is $10\mu m$, as highlighted in Table V.

B. Placement and Routing

After partitioning the weights of an AI model into the IMC tiles, the placement and routing of these tiles into a 2.5D/3D architecture further determine the overall system performance. Figure 15 illustrates the scenarios. For a 2D chip or 2.5D chiplet system (Figure 15(a) and Figure 15(b)), HISIM follows the sequence of AI algorithm layers to place and route the tiles, in order to minimize the hops of data movement. Figure 14 uses this method to demonstrate DenseNet-121.

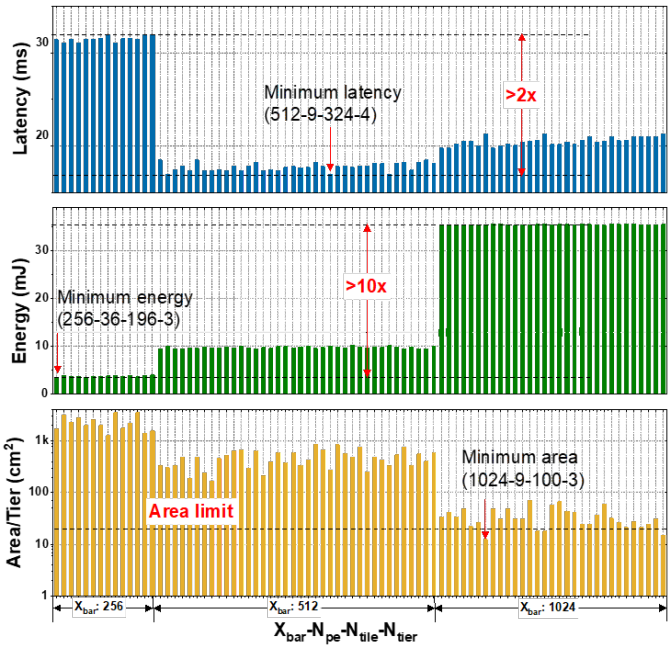

Fig. 16. Design space exploration of ViT on a 3D system. Design configurations include X_{bar}(256,512,1024), N_{pe}(9,16,25,36), N_{tile}(49,64,...,361,400), and N_{tier}(1,2,3,4).

In a 3D system, there are more options because the bandwidth of the vertical link is comparable to that of the links within the tier. As an example, Figure 15(c) and Figure 15(d) present two different placement methods, assuming there are four algorithm layers, with each layer taking two tiles. In the first method (Figure 15(c)), tiles from adjacent layers are placed on the same tier until the tier is full. Then, the tiles from the next layer are placed on another tier. In the second method (Figure 15(d)), tiles from adjacent layers are placed vertically across different tiers and then expand to other areas within the tier. This results in more data communication occurring in the vertical direction compared to the first method. Furthermore, the number of hops within the tier can be adjusted, influencing the number of vertical TSVs that are accessible. Depending on the data volume and sequence of a specific AI algorithm, the preferred placement and routing methods vary to minimize overall latency [69]. In HISIM, users can customize the placement and routing methods to explore the 3D network.

C. Design Exploration for AI Computing

Using the full set of HISIM, we demonstrate end-to-end design exploration of multiple AI algorithms in this section.

1) Speedup in Simulation: We demonstrate HISIM with ResNet-110 on CIFAR-100, DenseNet-121 on ImageNet, GCN on Cora, and ViT on ImageNet. These algorithms are mapped onto a 3D system with various number of tiers. The simulation speed is compared to the combined results of the state-of-the-art IMC simulator [32] and the network simulator [34]. Table X summarizes the end-to-end latency and energy con-

TABLE X
END-TO-END SIMULATION OF REPRESENTATIVE AI ALGORITHMS ON 3D HISIM.

AI Model	X_{bar}	N_{tier}	Latency (ms)		Energy (mJ)		Area/Tier (cm^2)	Simulation Time (ms)		Speedup
			Tile	Network	Tile	Network		HISIM	[32] + [34]	
ResNet-110	256	4	4.80	0.52	0.074	0.013	2.10	18.50	4.25×10^5	2.3×10^4
DenseNet-121	1024	4	20.66	7.19	39.15	0.59	15.50	81.00	8.10×10^6	1.0×10^5
GCN	1024	2	164.40	425.55	834.60	6.48	117.84	29.70	5.30×10^7	1.78×10^6
ViT	1024	3	12.15	0.55	35.25	0.27	12.81	73.56	3.16×10^6	4.33×10^4

sumption, as well as the breakdown to computing tiles and networks. Depending on the algorithm structures, data movement can dominate the overall latency, as observed in the case of GCN. Computing tiles are usually the primary contributors to energy consumption. Compared to conventional simulators, HISIM is $10^4 - 10^6\times$ faster in performance prediction, as shown in Table X. Such acceleration confirms the advantage of analytical performance models.

2) Design Space Exploration: For a particular AI algorithm, optimizing the hardware configuration is challenging due to the vast space of design parameters and multiple constraints involved. Therefore, the efficiency of the performance simulator is critical for quickly searching the design space and guiding the user toward an optimal solution. Using the ViT algorithm as the benchmark, Figure 16 demonstrates the process and highlights its importance. Design variables include the crossbar size (X_{bar}), the number of PEs per tile (N_{pe}), the number of tiles per tier (N_{tile}), and the number of 3D tiers (N_{tier}). In total, there are 672 design configurations. HISIM predicts the end-to-end latency, energy, and area per tier for all of them within 48.8 seconds. As observed in Figure 16, there are dramatic differences in power, performance, and area (PPA) across various configurations. Throughout the range of 672 design configurations tested in this experiment, we observe that energy consumption differs by more than $10\times$, while latency varies by more than $2\times$. The specific configurations optimized for low power or high speed also differ significantly. Some configurations may not be practical for hosting the ViT model due to the area constraint of a monolithic chiplet. Therefore, it is critical to conduct design space exploration at an early stage for guiding further design optimization within a viable range of configurations.

A second example is presented in Figure 17. In this study, we map the DenseNet-121 model to 2D (one monolithic chip), 2.5D (multiple chiplets), and 3D (multiple tiers). With stationary weights applied to the IMC design, the total number of computing tiles remains constant across all configurations. The crossbar size is 1024. Transitioning from 2D to 2.5D slightly reduces overall latency and energy consumption. However, the peak temperature increases due to the introduction of the AIB interface. When moving to 3D configurations, there is a significant reduction in latency, highlighting the advantages of 3D TSVs in terms of bandwidth. Nevertheless, the peak temperature rises due to increased power density in the 3D stack. For both examples, early-stage design exploration is essential to identifying the most feasible configurations, balancing the trade-offs between PPAs, and improving the efficiency.

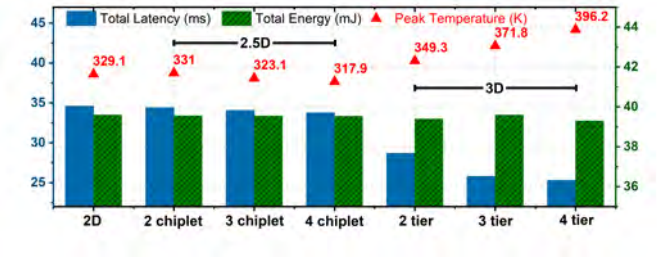

Fig. 17. Comparison of 2D, 2.5D and 3D mapping of DenseNet-121, at a constant number of IMC tiles.

V. CONCLUSION

In this work, we propose a new benchmark tool, Heterogeneous Integration Simulator with Interconnect Modeling (HISIM), for early-stage design exploration of 2.5D/3D systems. HISIM develops a set of analytical performance models for various computing cores, interconnection technologies, and network topologies. It provides fast and accurate electro-thermal analysis, scalable with different design configurations. Compared to conventional benchmark tools, HISIM achieves $10^4 - 10^6\times$ speedup in the evaluation of power, performance, and area (PPA), supporting rapid search in the vast design space. The HISIM code is available at https://github.com/mec-UMN/HISIM. It will be continuously updated with new models and design configurations. This repository provides access to the tool, allowing researchers and designers to perform efficient and accurate design exploration of 2.5D/3D systems.

REFERENCES

[1] A. Mehonic and A. J. Kenyon, "Brain-inspired computing needs a master plan," *Nature*, vol. 604, no. 7905, pp. 255–260, Apr 2022. [Online]. Available: https://doi.org/10.1038/s41586-021-04362-w
[2] Y. Hu *et al.*, "Wafer-scale computing: Advancements, challenges, and future perspectives," *IEEE Circuits and Systems Magazine*, 2024.
[3] A. Dosovitskiy *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.
[4] T. Zhang, K. Kasichainula, Y. Zhuo, B. Li, J.-S. Seo, and Y. Cao, "Transformer-based selective super-resolution for efficient image refinement," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 7, 2024, pp. 7305–7313.
[5] G. Krishnan *et al.*, "Siam: Chiplet-based scalable in-memory acceleration with mesh for deep neural networks," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 20, no. 5s, pp. 1–24, 2021.
[6] G. R. Nair *et al.*, "Fpga acceleration of gcn in light of the symmetry of graph adjacency matrix," in *2023 Design, Automation Test in Europe Conference Exhibition (DATE)*, 2023, pp. 1–6.
[7] S. K. Mandal *et al.*, "Coin: Communication-aware in-memory acceleration for graph convolutional networks," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 2022.
[8] G. Krishnan *et al.*, "3d-isc: A 65nm 3d compatible in-sensor computing accelerator with reconfigurable tile architecture for real-time dvs data compression," in *2023 IEEE Asian Solid-State Circuits Conference (A-SSCC)*, 2023, pp. 1–3.

[9] A. Shafiee *et al.*, "Isaac: A convolutional neural network accelerator with in-situ analog arithmetic in crossbars," *ACM SIGARCH Computer Architecture News*, vol. 44, no. 3, pp. 14–26, 2016.

[10] G. R. Nair *et al.*, "Fpga acceleration of gcn in light of the symmetry of graph adjacency matrix," in *2023 Design, Automation Test in Europe Conference Exhibition (DATE)*, 2023, pp. 1–6.

[11] "https://eps.ieee.org/technology/heterogeneous-integration-roadmap/2020-edition.html." IEEE, 2020.

[12] C. Liu, J. Botimer, and Z. Zhang, "A 256gb/s/mm-shoreline aib-compatible 16nm finfet cmos chiplet for 2.5 d integration with stratix 10 fpga on emib and tiling on silicon interposer," in *2021 IEEE Custom Integrated Circuits Conference (CICC)*. IEEE, 2021, pp. 1–2.

[13] M.-S. Lin *et al.*, "A 7-nm 4-ghz arm^1-core-based cowos^1 chiplet design for high-performance computing," *IEEE Journal of Solid-State Circuits*, vol. 55, no. 4, pp. 956–966, 2020.

[14] B. Zimmer *et al.*, "A 0.32–128 tops, scalable multi-chip-module-based deep neural network inference accelerator with ground-referenced signaling in 16 nm," *IEEE Journal of Solid-State Circuits*, vol. 55, no. 4, pp. 920–932, 2020.

[15] M. Mansuri *et al.*, "A scalable 0.128–1 tb/s, 0.8–2.6 pj/bit, 64-lane parallel i/o in 32-nm cmos," *IEEE Journal of solid-state circuits*, 2013.

[16] M. Gerber, C. Beddingfield, S. O'Connor, M. Yoo, M. Lee, D. Kang, S. Park, C. Zwenger, R. Darveaux, R. Lanzone *et al.*, "Next generation fine pitch cu pillar technology—enabling next generation silicon nodes," in *2011 IEEE 61st electronic components and technology conference (ECTC)*. IEEE, 2011, pp. 612–618.

[17] R. Mahajan *et al.*, "Embedded multi-die interconnect bridge (emib)–a high density, high bandwidth packaging interconnect," in *2016 IEEE 66th Electronic Components and Technology Conference (ECTC)*. IEEE, 2016, pp. 557–565.

[18] B. Banijamali *et al.*, "Advanced reliability study of tsv interposers and interconnects for the 28nm technology fpga," in *2011 IEEE 61st Electronic Components and Technology Conference (ECTC)*, 2011.

[19] M. Su, B. Black, Y.-H. Hsiao, C.-L. Changchien, C.-C. Lee, and H.-J. Chang, "2.5 d ic micro-bump materials characterization and imcs evolution under reliability stress conditions," in *2016 IEEE 66th Electronic Components and Technology Conference (ECTC)*. IEEE, 2016.

[20] J. Liao, A. Liao, S. Peng, G. Lin, T. Lu, and S. Chen, "Die bonding with non-clean flux in fine pitch copper pillar bump study and reliability performance for 2.5 d ic package," in *2016 IEEE 66th Electronic Components and Technology Conference (ECTC)*. IEEE, 2016.

[21] P. Vivet *et al.*, "A $4 \times 4 \times 2$ homogeneous scalable 3d network-on-chip circuit with 326 mflit/s 0.66 pj/b robust and fault tolerant asynchronous 3d links," *IEEE Journal of Solid-State Circuits*, vol. 52, no. 1, 2016.

[22] L. W. Schaper, S. L. Burkett, S. Spiesshoefer, G. V. Vangara, Z. Rahman, and S. Polamreddy, "Architectural implications and process development of 3-d vlsi z-axis interconnects using through silicon vias," *IEEE Transactions on Advanced Packaging*, vol. 28, no. 3, pp. 356–366, 2005.

[23] D. M. Jang *et al.*, "Development and evaluation of 3-d sip with vertically interconnected through silicon vias (tsv)," in *2007 proceedings 57th electronic components and technology conference*. IEEE, 2007.

[24] B. Banijamali, S. Ramalingam, H. Liu, and M. Kim, "Outstanding and innovative reliability study of 3d tsv interposer and fine pitch solder micro-bumps," in *2012 IEEE 62nd Electronic Components and Technology Conference*. IEEE, 2012, pp. 309–314.

[25] Z.-C. Hsiao *et al.*, "Cu/bcb hybrid bonding with tsv for 3d integration by using fly cutting technology," in *2015 International Conference on Electronics Packaging and iMAPS All Asia Conference (ICEP-IAAC)*. IEEE, 2015, pp. 834–837.

[26] D. Ingerly *et al.*, "Foveros: 3d integration and the use of face-to-face chip stacking for logic devices," in *2019 IEEE International Electron Devices Meeting (IEDM)*. IEEE, 2019, pp. 19–6.

[27] Z. Wang *et al.*, "Ai computing in light of 2.5d interconnect roadmap: Big-little chiplets for in-memory acceleration," *IEEE IEDM*, 2022.

[28] R. Radojcic, *More-than-Moore 2.5 D and 3D SiP Integration*. Springer, 2017.

[29] E. Beyne, "Heterogeneous system partitioning and the 3d interconnect technology landscape," in *2020 Symposia on VLSI technology and Circuits*, 2020, pp. SC2–2.

[30] S. Sinha *et al.*, "A high-density logic-on-logic 3dic design using face-to-face hybrid wafer-bonding on 12nm finfet process," in *2020 IEDM*. IEEE, 2020, pp. 15–1.

[31] J. C. Lee *et al.*, "High bandwidth memory (hbm) with tsv technique," in *2016 International SoC Design Conference (ISOCC)*. IEEE, 2016.

[32] P.-Y. Chen *et al.*, "Neurosim: A circuit-level macro model for benchmarking neuro-inspired architectures in online learning," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 37, no. 12, pp. 3067–3080, 2018.

[33] L. Xia *et al.*, "Mnsim: Simulation platform for memristor-based neuromorphic computing system," *IEEE TCAD*, vol. 37, 2017.

[34] N. Jiang *et al.*, "Booksim 2.0 user's guide," *Standford University*, p. q1, 2010.

[35] J. M. Joseph *et al.*, "Ratatoskr: An open-source framework for in-depth power, performance and area analysis in 3d nocs," 2019.

[36] X. Peng *et al.*, "Benchmarking monolithic 3d integration for compute-in-memory accelerators: overcoming adc bottlenecks and maintaining scalability to 7nm or beyond," in *2020 IEDM*. IEEE, 2020, pp. 30–4.

[37] G. Krishnan *et al.*, "Big-little chiplets for in-memory acceleration of dnns: A scalable heterogeneous architecture," in *Proceedings of the 41st IEEE/ACM International Conference on Computer-Aided Design*, 2022.

[38] N. Beck *et al.*, "'zeppelin': An soc for multichip architectures," in *2018 IEEE International Solid-State Circuits Conference-(ISSCC)*. IEEE, 2018, pp. 40–42.

[39] Y. S. Shao *et al.*, "Simba: Scaling deep-learning inference with multi-chip-module-based architecture," in *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture*, 2019.

[40] Intel, "AIB 2.0 Specification," 2019, https://github.com/chipsalliance/AIB-specification.

[41] D. D. Sharma, G. Pasdast, Z. Qian, and K. Aygun, "Universal chiplet interconnect express (ucie): An open industry standard for innovations with chiplets at package level," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 2022.

[42] K. DeHaven and J. Dietz, "Controlled collapse chip connection (c4)-an enabling technology," in *1994 Proceedings. 44th Electronic Components and Technology Conference*, 1994, pp. 1–6.

[43] H.-W. Hu and K.-N. Chen, "Development of low temperature cucu bonding and hybrid bonding for three-dimensional integrated circuits (3d ic)," *Microelectronics Reliability*, vol. 127, p. 114412, 2021.

[44] L. Wu *et al.*, "The advent of 3-d package age," in *Twenty Sixth IEEE/CPMT International Electronics Manufacturing Technology Symposium*. IEEE, 2000, pp. 102–107.

[45] G. H. Loh, Y. Xie, and B. Black, "Processor design in 3d die-stacking technologies," *IEEE Micro*, vol. 27, no. 3, pp. 31–48, 2007.

[46] G. R. Nair, P. S. Nalla, G. Krishnan, Anupreetham, J. Oh, A. Hassan, I. Yeo, K. Kasichainula, M. Seok, J.-s. Seo, and Y. Cao, "3d in-sensor computing for real-time dvs data compression: 65nm hardware-algorithm co-design," *IEEE Solid-State Circuits Letters*, pp. 1–1, 2024.

[47] A. Shafiee *et al.*, "Isaac: A convolutional neural network accelerator with in-situ analog arithmetic in crossbars," in *2016 ACM/IEEE 43rd Annual International Symposium on Computer Architecture (ISCA)*, 2016, pp. 14–26.

[48] X. Peng *et al.*, "Heterogeneous 3-d integration of multitier compute-in-memory accelerators: An electrical-thermal co-design," *IEEE Transactions on Electron Devices*, vol. 68, no. 11, pp. 5598–5605, 2021.

[49] Q. Wang, X. Wang, S. H. Lee, F.-H. Meng, and W. D. Lu, "A deep neural network accelerator based on tiled rram architecture," in *2019 IEEE International Electron Devices Meeting (IEDM)*, 2019.

[50] X. Wang *et al.*, "Taichi: A tiled architecture for in-memory computing and heterogeneous integration," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 69, no. 2, pp. 559–563, 2022.

[51] A. Lu, X. Peng, W. Li, H. Jiang, and S. Yu, "Neurosim validation with 40nm rram compute-in-memory macro," in *2021 IEEE 3rd International Conference on Artificial Intelligence Circuits and Systems*, 2021.

[52] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015.

[53] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," 2017.

[54] A. Samajdar, Y. Zhu, P. Whatmough, M. Mattina, and T. Krishna, "Scale-sim: Systolic cnn accelerator simulator," *arXiv preprint arXiv:1811.02883*, 2018.

[55] W. Zhao and Y. Cao, "New generation of predictive technology model for sub-45nm design exploration," in *7th International Symposium on Quality Electronic Design (ISQED'06)*, 2006, pp. 6 pp.–590.

[56] J. Cho *et al.*, "Modeling and analysis of through-silicon via (tsv) noise coupling and suppression using a guard ring," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 1, no. 2, pp. 220–233, 2011.

[57] R. Marculescu *et al.*, "Outstanding research problems in noc design: system, microarchitecture, and circuit perspectives," *IEEE TCAD*, vol. 28, no. 1, pp. 3–21, 2008.

[58] A. B. Kahng *et al.*, "Orion3. 0: A comprehensive noc router estimation tool," *IEEE Embedded Systems Letters*, vol. 7, no. 2, pp. 41–45, 2015.

[59] C. Alliance, "Aib-phy-hardware rtl source codes," 2019, https://github.com/chipsalliance/aib-phy-hardware.
[60] W. Tang *et al.*, "Arvon: A heterogeneous system-in-package integrating fpga and dsp chiplets for versatile workload acceleration," *IEEE Journal of Solid-State Circuits*, pp. 1–11, 2023.
[61] R. Mahajan *et al.*, "Embedded multi-die interconnect bridge (emib) – a high density, high bandwidth packaging interconnect," in *2016 IEEE 66th Electronic Components and Technology Conference (ECTC)*, 2016.
[62] K. Puttaswamy and G. H. Loh, "Thermal analysis of a 3d die-stacked high-performance microprocessor," in *Proceedings of the 16th ACM Great Lakes symposium on VLSI*, 2006, pp. 19–24.
[63] A. R. Menon, S. Karajgikar, and D. Agonafer, "Thermal design optimization of a package on package," in *2009 25th Annual IEEE Semiconductor Thermal Measurement and Management Symposium*. IEEE, 2009, pp. 329–335.
[64] C. Torregiani, H. Oprins, B. Vandevelde, E. Beyne, and I. De Wolf, "Compact thermal modeling of hot spots in advanced 3d-stacked ics," in *2009 11th Electronics Packaging Technology Conference*. IEEE, 2009, pp. 131–136.
[65] Z. Chen, X. Luo, and S. Liu, "Thermal analysis of 3d packaging with a simplified thermal resistance network model and finite element simulation," in *2010 11th International Conference on Electronic Packaging Technology & High Density Packaging*. IEEE, 2010, pp. 737–741.
[66] P. Vivet *et al.*, "Intact: A 96-core processor with six chiplets 3d-stacked on an active interposer with distributed interconnects and integrated power management," *IEEE Journal of Solid-State Circuits*, vol. 56, no. 1, pp. 79–97, 2020.
[67] C.-W. Ho, A. Ruehli, and P. Brennan, "The modified nodal approach to network analysis," *IEEE Transactions on Circuits and Systems*, vol. 22, no. 6, pp. 504–509, 1975.
[68] Y. Wei, J. Hu, F. Liu, and S. S. Sapatnekar, "Physical design techniques for optimizing rta-induced variations," in *2010 15th Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2010, pp. 745–750.
[69] Z. Wang *et al.*, "Exploiting 2.5 d/3d heterogeneous integration for ai computing," in *2024 29th Asia and South Pacific Design Automation Conference (ASP-DAC)*. IEEE, 2024, pp. 758–764.

Zhenyu Wang received a B.E. degree from Huazhong University of Science and Technology in 2018. He is currently pursuing a Ph.D. degree with the School of ECEE at Arizona State University. His current research focuses on the chiplet-based hardware architecture for deep learning acceleration and hardware-software co-design AI acceleration. He worked with Maxlinear, Intel, and TSMC as the intern, in 2022, 2024.

Pragnya Nalla is a PhD student at the University of Minnesota, Twin Cities. Her research domains include power and performance estimations, identifying optimal mapping and chiplet library configurations for 2.5D/3D architectures. She has previously received her master's and bachelor's degrees from the Indian Institute of Technology Madras.

Jingbo Sun received his B.E. degree from the Tianjin University in 2008. He accomplished his M.S. degree from the University of Southern California in 2020 and received the Ph.D. degree from Arizona State University in 2024. His research interests include machine learning algorithms for dynamic systems, such as novelty detection, continual learning, and graph-based perception.

A. Alper Goksoy received his B.S. degree in Electrical and Electronics Engineering from Bogazici University, Istanbul, Turkey and his Ph.D. in Electrical and Computer Engineering at University of Wisconsin-Madison, USA in 2024. His research interests include task scheduling for domain-specific SoCs, in-memory computing, and the design of AI hardware accelerators.

Sumit K. Mandal is an assistant professor in the department of computer science and automation (CSA) at the Indian Institute of Science (IISc), Bangalore. He received his PhD from University of Wisconsin, Madison. His research interest lies in in-memory computing based accelerator for machine learning algorithms, design and analysis of on-chip and on-package interconnect.

Jae-sun Seo is an Associate Professor at the School of ECE at Cornell Tech. His research interests include energy-efficient ASIC and FPGA hardware design of AI algorithms and neuromorphic computing. Dr. Seo was a recipient of the 2012 IBM Outstanding Technical Achievement Award, 2017 NSF CAREER Award, 2020 Intel Outstanding Researcher Award, and 2022 IEEE TVLSI Best Paper Award. He is a Senior Member of IEEE.

Vidya A. Chhabria is an assistant professor in the School of ECEE at Arizona State University. She received her Ph.D. and M.S. in Electrical Engineering from the University of Minnesota in 2022 and 2018. Her research interests lie in computer-aided design (CAD) for VLSI systems, which includes physical design, optimization, and analysis algorithms. She received the ICCAD Best Paper Award in 2021.

Jeff (Jun) Zhang is an assistant professor in the School of Electrical, Computer and Energy Engineering at ASU. He obtained his Ph.D. from New York University. From 2020-2022, Zhang was a postdoctoral fellow at Harvard University. His research interests are in deep learning, computer architecture, embedded systems, and EDA, with particular emphasis on energy-efficient and fault-tolerant design for AI/ML systems.

Chaitali Chakrabarti is a Professor with the School of Electrical Computer and Energy Engineering, Arizona State University (ASU), Tempe, and a Fellow of the IEEE. Her research interests are in the areas of low power embedded systems design,distributed machine learning and VLSI architectures and algorithms for signal processing and communications.

Umit Ogras is the Gene Amdahl Professor in the Dept. of Electrical and Computer Engineering at the University of Wisconsin-Madison. He worked at the Arizona State University as a faculty member between 2013-2020 and at Intel as a research scientist between 2008-2013 before receiving his Ph.D. degree in Computer Engineering from Carnegie Mellon University in 2007. His research interests include chiplet-based platforms, edge AI, domain-specific systems, and low-power VLSI.

Yu Cao (S'99-M'02-SM'09-F'17) received the Ph.D. degree in electrical engineering from University of California, Berkeley, in 2002. He is now the Louis John Schnell Professor of Electrical and Computer Engineering at the University of Minnesota (UMN), Minneapolis, Minnesota. His research interests include neural-inspired computing, hardware design for on-chip learning, and reliable integration of nanoelectronics.