Title
Three-Dimensional Wafer Scale Integration for Ultra-Large-Scale Neuromorphic Systems

Author
Wan, Zhe

Publication Date
2017

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA

Los Angeles

Three-Dimensional Wafer Scale Integration for Ultra-Large-Scale Neuromorphic Systems

A thesis submitted in partial satisfaction

of the requirements of the degree

Master of Science in Electrical Engineering

by

Zhe Wan

2017

ABSTRACT OF THE THESIS

Three-Dimensional Wafer Scale Integration for Ultra-Large-Scale Neuromorphic Systems

by

Zhe Wan

Master of Science in Electrical Engineering

University of California, Los Angeles, 2017

Professor Subramanian Srikantes Iyer, Chair

Recent trends indicate that we will generate exponentially more data from a variety of devices. Neuromorphic computing (a.k.a. brain-inspired computing), is required to cognitively extract useful information in the Internet of Things (IoT). In order to achieve cognitive computing, ultra-large-scale systems that contain billions of neurons and trillions of synapses as the interconnection between those neurons will be needed. In this thesis, we first discuss system integration technologies and the scaling limitation of the von Neumann (vN) machines. Then, based on the system integration technologies, we propose neuromorphic computing systems with anon-von Neumann (NvN) architecture. The proposed systems are modeled, simulated, evaluated and compared with different system integration technologies. We show that the 3-dimensional-wafer-scale-integration (3D-WSI) technology is a potential candidate to enable neuromorphic computing systems at the scale of the human brain, while keeping the system latency and energy consumption at an acceptable level.

The thesis of Zhe Wan is approved.

Jingshen Jason Cong

Kang Lung Wang

Jason C. S. Woo

Subramanian Srikantes Iyer, Committee Chair

University of California, Los Angeles

2017

Table of Contents

List of Figures

List of Tables

List of Abbreviations

2DI: Two-Dimensional circuit board Integration

3D-WSI: Three-Dimensional Wafer Scale Integration

ACM: Association for Computing Machinery

ANN: Artificial Neural Network

b2b: board to board

BEOL: Back End Of Line

BGA: Ball Grid Array

c2c: chip to chip

CMOS: Complementary Metal-Oxide-Semiconductor

CMP: Chemical Mechanical Polishing

CPU: Central Processing Unit

DRAM: Dynamic Random-Access-Memory

EEDN: Energy Efficient Deep convolutional Neural network

FACETS: Fast Analog Computing with Emergent Transient States

FPGA: Field Programmable Gate Array

IoT: Internet of Things

NvN: Non von Neumann

OWI: On-Wafer Integration

PCB: Printed Circuit Board

PCIE: Peripheral Component Interconnect Express

PDF: Probability Density Function

SerDes: Serializer/Deserializer

SOPS: Synaptic Operation Per Second

STDP: Spike-Time Dependent Plasticity

TSV: Through Silicon Via

vN: von Neumann

Acknowledgement

I would like to thank my advisor, Prof. Subramanian Iyer, for the guidance throughout the research on the thesis. In addition, I would like to acknowledge the valuable comments and advice from my committee members, Prof. Jason Cong, Prof. Kang Wang and Prof. Jason Woo. I also appreciate the help from Dr. Arvind Kumar, my industrial mentor, and the cooperation in the research that lays some foundation of this work. Special thanks to Dr. Boris Vaisband and Zhuoyu Jiang for their help during my writing process.

Finally, my deepest gratitude goes to my parents for their constant support.

1. Introduction and Motivation

The concept of neuromorphic computing was proposed by Mead [1] decades ago. In particular, CMOS circuits would be designed to imitate the operation of the major components of the brain, such as neurons and synapses. Thus, a neuromorphic computing system is an electronic system designed to mimic some aspects of the biological brain, including its cognitive capability. Two system level requirements are important for the realization of a neuromorphic system:

(1) Circuit density: the CMOS circuits should be dense enough to contain around 10^{10} neurons and 10^{13}-10^{14} synapses [2,3] (processors and memories) within a reasonable physical space. For example, in the state-of-the-art product, a 28nm CMOS core containing 256 neurons and 65536 synapses is only about $0.09mm^2$ [4] and this area will be further reduced with more advanced technology node. Therefore, up to a point, this requirement is supported by technology scaling.

(2) System architecture: neuromorphic system architecture required for the neuromorphic systems rather than the classical von Neumann (vN) architecture. The classical vN architecture has a centralized processing unit at the top of the hierarchical memory. However, if we consider the neurons and synaptic weights as the "processors" and "memories" of the brain, they are distributed throughout the neural cortex [5]. To highlight the difference, we name such distributed architecture the non-von Neumann architecture (NvN). In addition, the distributed elements in the brain are highly interconnected - each neuron in the brain has on average 10^4 to 10^5 connections (synapses) to other neurons. Consequently, the neuromorphic system demands high density of interconnects throughout the NvN architecture [4]. Unfortunately, the

interconnect scaling has not advanced much from the development of technology: while the dimension of the transistor features has scaled by 1000x in the past 40 years, the scaling of chip-package interconnect has merely scaled by 4x [6].

Table 1 State-of-the-Art Large Scale Neuromorphic Systems.

Project	Total # of neurons	Total # of synapses	Energy consumption	Scale
Neurogrid [7]	1M	8B	3W	16-chip board
IBM TrueNorth [4]	1M	256M	0.07W	1 chip
IBM TrueNorth NS16e [11]	16M	4G	8.88W	16-chip board
SpiNNaker [8]	250K	80M	48W	48-chip board
FACETS [9]	180K	40M	N/A	$5mm^2$-chip wafer (200mm)

State-of-the-art neuromorphic systems are still far from the scale of the human brain (10^{10} neurons with 10^3-10^4 synapses per neuron), at least by a factor of 10^4, as summarized in Table 1. To achieve the scale of a human brain, multiple neuromorphic chips should be integrated on the circuit boards or backplanes to comprise an ultra-large scale computing systems.

However, due to the lack of scaling of package and board level interconnect pitches, the traditional integration schemes cannot support a large scale neuromorphic system that is power efficient. The NvN architecture requires high interconnect density (bandwidth) throughout the system, but the off-chip interconnect pitch expands from several μm (BEOL fat wire pitch) to 400μm (BGA pitch) [6] as the chips are integrated on the board, limiting the number of available I/Os of the chips. Consequently, Serializer/Deserilizers (SerDes) circuits are used to interconnect chips on the board by using high-speed channels to increase the bandwidth of the limited number of available physical channels. However, SerDes come at the price of increased power consumption: a mid-performance system typically spends 30% of the total power on the I/Os [10]. To make things worse, neuromorphic systems which rely heavily on communication between neurons, will spend an even larger proportion of the power budget on the chip I/Os. For

example, the 16-TrueNorth-chip board consumes 8.88W while each single chip consumes only about 0.114W-0.228W [11]. In this case, the overhead that is not from the chips themselves but from the board is about 83% of the total consumed power when the system is active and 76% when the system is idle [11]. As a result, traditional packaging with the use of SerDes for chip I/O will make it very difficult to scale out to billions of neurons and trillions of synapses within affordable power budget.

To reduce the power consumption from due to the I/Os, novel schemes of integration are needed to eliminate the use of SerDes and allow low power interconnect among chips. Such scheme can use fine pitch interconnect from the conventional BEOL process and integrate the chips on the wafer or wafer-like substrate as demonstrated in the FACETS (Fast Analog Computing with Emergent Transient States) project [9]. However, this approach is only valid if the system can be contained within a single wafer, otherwise the interconnect pitch between wafers will still suffer from large pitch of the wafer-to-wafer connectors after single-wafer packaging. For example, 20 of the 200mm wafers in the FACETS project are integrated using five industry-standard 19'' racks in the Human Brain Project (NM-PM) [12], which occupies a large volume (~$3m^3$) and consumes a lot of power (~10kW-100kW).

3D wafer scale integration (3D-WSI) is a promising candidate to scale the neuromorphic systems. 3D-WSI uses wafer bonding technology to stack the wafers and through silicon vias (TSVs) to interconnect between adjacent strata. The state-of-the-art 3D-WSI has been demonstrated for memory on 300mm wafers [13] with $5\mu m$ fine pitch TSVs. With the TSVs, the top and bottom area of the chip are used for interconnects to boost interconnect density. Although this makes the chips susceptible to heating, such tradeoff is appropriate for

neuromorphic systems that are based on memory, and low power due to the low duty cycle of the logic.

In our previous work [22], we have shown that it is possible to use 3D-WSI technology to integrate neuromorphic systems up to the scale a of human brain. Specifically, we presented a principal embodiment of the 3D-WSI neuromorphic system using simple processor cores and DRAM wafers. Biological connectivity data was used to evaluate the performance of the system to study the effect of scaling a neuromorphic system up to the scale of a human brain.

This thesis expands on the aforementioned study. Specifically, the use of 3D-WSI technology to build a neuromorphic system scalable to level of a human brain (10^{10} neurons, 10^{13} synapses). The 3D-WSI neuromorphic system is compared with the traditional circuit-board-integrated (2DI) system in which the neuromorphic chips are integrated on the printed circuit boards (PCBs) and backplanes. We incorporate the connectivity data from biological experiments [14] to the neuromorphic system to evaluate its performance. The evaluation is based on metrics such as latency, bandwidth, power consumption and synaptic operations per second (SOPS), which are also used to study the effect of scaling in both integration technologies.

The thesis is organized as following: in Chapter 2, we briefly introduce neuromorphic computing and explain why integration of the neuromorphic system is crucial for cognitive applications. The integration technologies that can be used for neuromorphic system integration are discussed in Chapter 3. The model of the neuromorphic systems simulation and the results are provided, respectively, in Chapter 4 and Chapter 5. Major findings and an outlook on the future work are presented in Chapter 6.

2. Neuromorphic Computing Overview

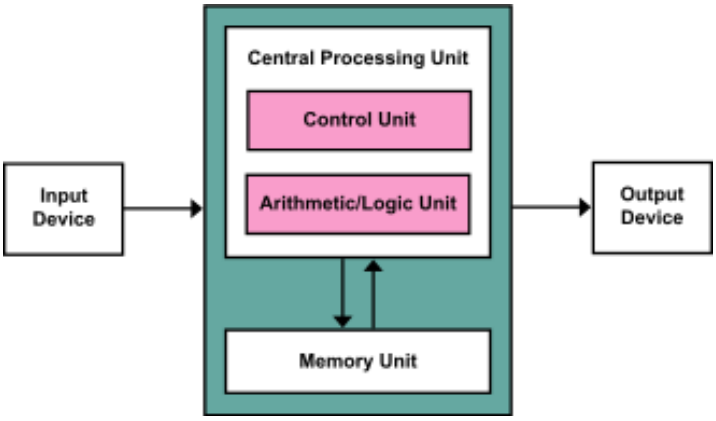


Fig. 1 Schematic of the von Neumann Architecture. [27]

Conventional computing requires structured data and comprehensive programming for good performance. Currently, vN architectures are dominant in the conventional computers. As shown in Fig. 1 [27], the vN architecture is divided into the central processing unit (CPU) and the memory. During operation, the CPU constantly fetches instructions and data from the memory to process the input. Consequently, when the program (solution) is well-defined and the data is well-structured, vN architectures provide efficient computing supported by the high-speed CPU. For example, computers are almost always faster to calculate the product of two ten-digit (decimal) numbers than a human, with perfect accuracy. However, the human brain has much better performance dealing with problems whose solutions are not very well-defined (often because the data is not well-structured) such as general pattern recognition and inference.

As the Internet of Things (IoT) comes of age, exponentially more data are generated from various devices. Such data are not necessarily well-structured and therefore, not useful as direct inputs to vN machines. As a result, vN machines cannot efficiently process the massive IoT data without significant programming innovations and effort. In contrast, neuromorphic computing

emerges to complement conventional computing by interpreting unstructured data to generate useful information, and pipelining this information to the conventional computers for cognitive and high-performance computation [15].

In this chapter, we first introduce the algorithmic approach to neuromorphic computing through an example of an artificial neural network (ANN) that is widely used today (Subsection 2.1). Then we examine the operation of such algorithm in a vN machine – the Bluegene/P supercomputer, and demonstrate that ANN is not scalable in the vN architecture (Subsection 2.2). Finally, we introduce the hardware approach, the NvN architecture which is more compatible with neuromorphic algorithms with respect to resemblance and scalability, and discuss the limitations of the state-of-the-art neuromorphic systems (Subsection 2.3).

2.1 Neuromorphic Algorithm: Artificial Neural Network

The network structure in biological brains is essential to the development of neuromorphic algorithms. Most of the brain is occupied by the cerebrum which performs the major "intelligent/cognitive" operations. The cerebrum is composed of distributed gray matter and white matter. Specifically, the gray matter contains billions of neurons and the white matter contains billions of nerve fibers (axons and dendrites) that interconnects the neurons. Both types of the matters are distributed in the cerebrum, forming an enormous network of neurons without an apparent center. The connection from one neuron cell to another neuron cell is a synapse. An example of a connection between two neurons is shown in Fig. 2 [28]. A neuron can "talk" to another connected neuron by sending a signal (spike, in the form of an electrical potential) from its soma (cell body) to the synapse that is connected to the receiver neuron. The synapse has a membrane potential which is changed continuously by the incoming signals from the sender neuron. When the membrane potential reaches the threshold of the action potential, the synapse

spikes, sends a spike signal to the receiver neuron, and resets its membrane potential to some value as illustrated in the inset of Fig. 2. The dendrite of the receiver neuron is connected to the synapse and takes the spike signal into the soma of the receiver neuron to decide whether the receiver neuron will fire in response to the received spike signal. As a result, the neurons in the network can communicate to each other via the cascaded stream of spikes.

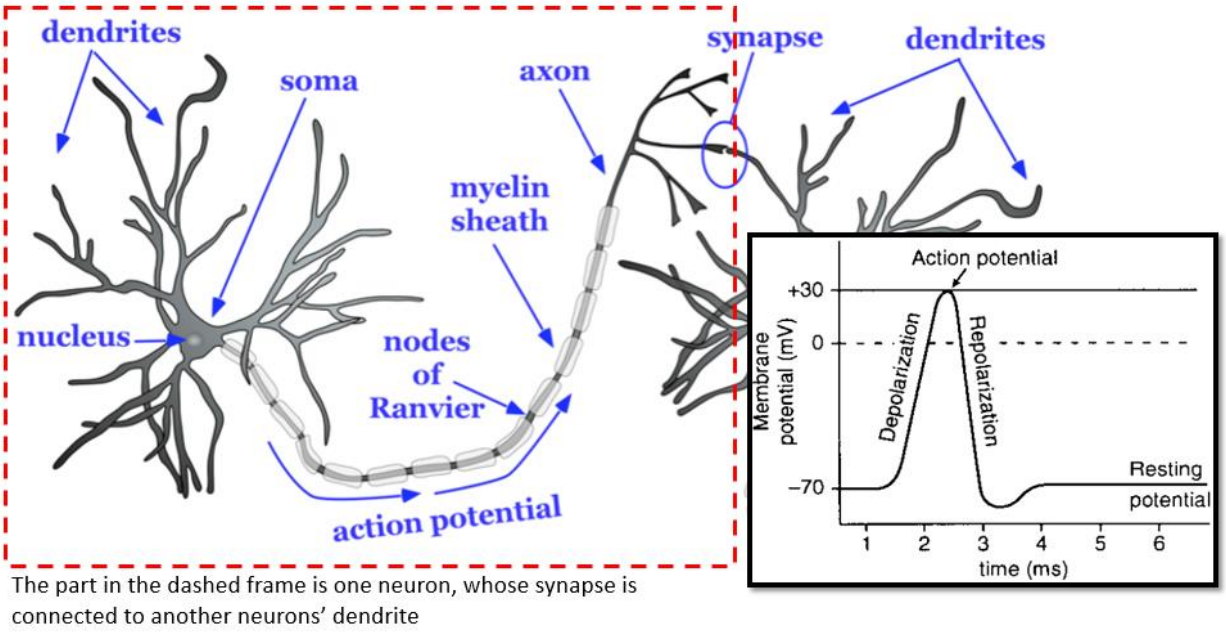
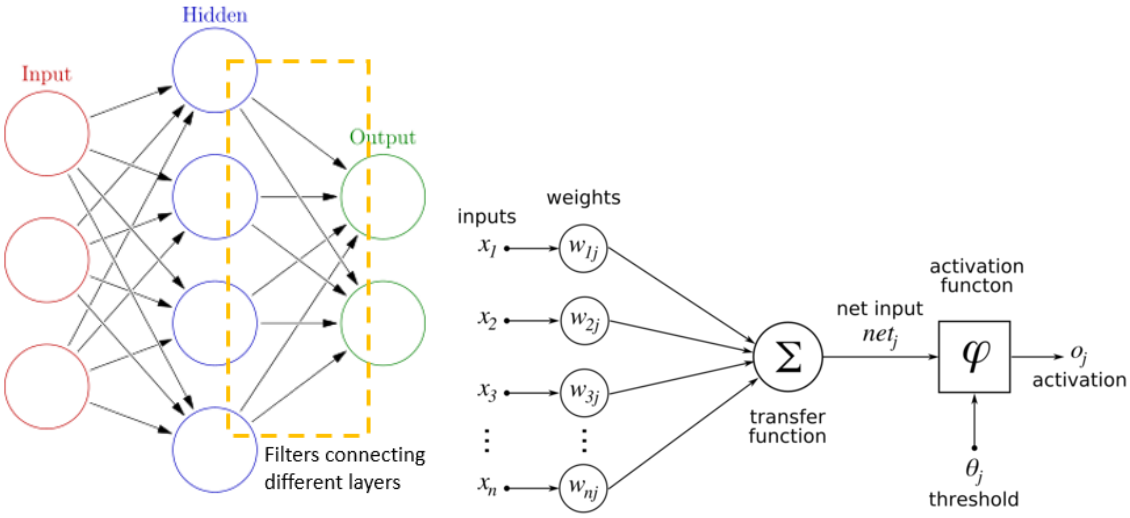


Fig. 2 The structure of a neuron and a neural connection. The membrane potential as a function of time when the neuron is firing at around 2ms, is shown in the inset. [28]

The principle of neuron-to-neuron communication in the biological brain leads to the Artificial Neural Network (ANN) structure that is very popular in current machine learning, widely developed and used for cognitive applications such as image recognition, voice recognition, game AI and language processing. ANNs are composed of an input layer, hidden layer(s), and output layer that are cascaded and connected by non-linear filters as shown in Fig. 3. Each filter represents a neuron. The filters receive several values (spikes) from the previous layer (similar to dendrites connected to the axons of the sender neurons). Then each filter calculates a weighted sum from the transfer function, decides (similar to the operation of a soma) whether to

fire or not according to the non-linear activation function with a certain threshold (action potential), and sends the corresponding activation signal (spike from its axon) to the next layer. For cognitive applications, each input to the network is some value that represents the data (*e.g.* a bit from an image) and each output represents a feature or a decision. The hidden layers help to extract implicit features to improve the accuracy of the output. For a given ANN, the filter size, transfer function, and activation function are usually defined by the application of the network, and the associated parameters are determined during training by numerical methods such as back-propagation [16].

Fig. 3 The structure of artificial neural network (left) and the filter (right). The filters that compute values from a layer and send their output to another layer can each be regarded as a neuron whose I/O connections are the dendrites (inputs) and synapses (outputs).



2.2 Inefficient Neuromorphic Simulation in von Neumann Architecture

There is a significant difference between the architecture of the brain and the architecture of the vN machines; the brain has a distributed network structure without a dedicated CPU or a separated memory as in the vN architecture. Nevertheless, the biological brain can still be simulated in conventional computers, using cortical simulators implemented as an ANN. As previously stated, the numbers of neurons and synapses in the brain of mammals are, respectively, at least 10^{10} and 10^{13}. Consequently, supercomputers are required to simulate the biological brains due to their gigantic scale. The Dawn supercomputer (BlueGene/P, 147k CPUs and 144TB of main memory) has been previously used to simulate an ANN up to the scale of the brain of a cat (1.6B neurons, 8.8T synapses). The simulation shows the oscillation of the electroencephalographic trace of the system, which is in agreement with the observations in animals [17]. However, due to the lack of high-level interpretation of the spiking pattern, the simulation was not able to realize any cognitive function of a real cat, despite the similarity in scale.

The simulation results also revealed an enormous power consumption gap between the supercomputer and the brain of a cat. While Dawn consumes 1MW of power and 643 seconds to simulate one second of cat brain activity, the actual cat brain consumes only around 10W.

From the Dawn supercomputer simulation, the energy consumed to simulate one second of the activity in the brain a cat is denoted by E_{sim} which is the product of the system power (P_{sys}) and the time multiplier (M_{time}). In the Dawn simulation, $E_{sim} = P_{sys} * M_{time} = 1\text{MW} * 643\text{s} = 643\text{MJ}$, yielding a 10^7 gap in the effective power as compared with the 10W cat brain (10J for each second). Let N be the size of the system for a certain number of neurons and

synapses. Also, assume the amount of work done (W), which is essentially the number of spikes processed by the system, to be proportional to the number of synapses such that

$$W \propto N \quad (1)$$

Then P_{sys} is at least proportional to N because the number of running CPUs (assumed to consume constant power) is proportional to N. M_{time} also grows with respect to N since the delay across the system increases with the size of the system. However, the relationship between M_{time} and N varies with hardware implementation, and therefore we assume $M_{time} \propto N^{\alpha}$ in which $\alpha > 0$. As a result, the relationship between the energy consumed by the simulation and the size of the system can be expressed as

$$E_{sim} = P_{sys} * M_{time} \propto N^{1+\alpha} \quad (2)$$

In the Dawn simulation, the communication delay is the major component that contributes to the 643X time multiplier, and it grows linearly with the CPU count, as shown in Fig. 4.

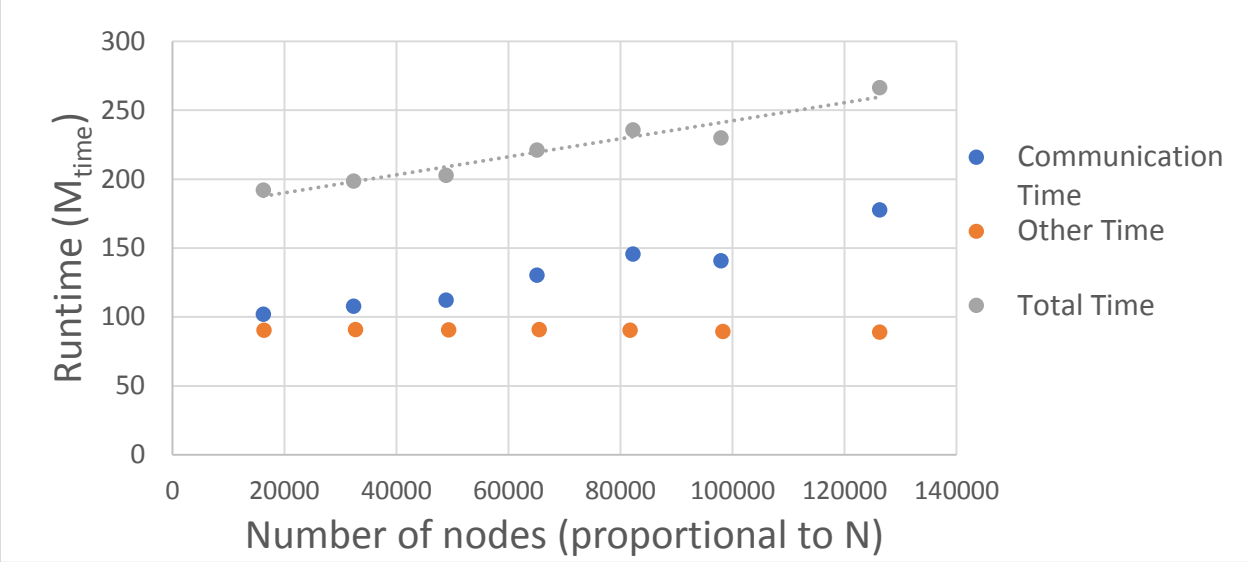


Fig. 4 The runtime breakdown of the BlueGene/P simulation for one second real time with respect to the number of nodes used [17]. The major component is the communication time which grows linearly as the system scales up.

Therefore, as the system is scaled out, the linear growth of both P_{sys} and M_{time} makes E_{sim} grow as the square of N (*i.e.* $\alpha = 1$).

$$E_{sim} = P_{sys} * M_{time} \propto N^2 \text{ | in Dawn simulation} \quad (3)$$

In the meantime, the energy consumption of a biological brain (E_{bio}) depends only on the amount of spikes (W), which is propotional to the system size

$$E_{bio} \propto N \text{| in biological brain} \quad (4)$$

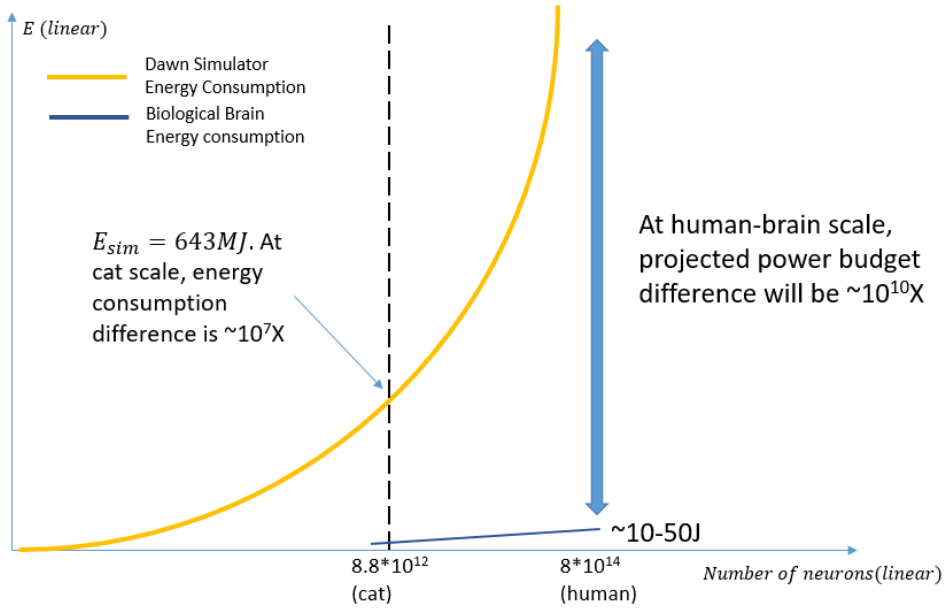


Fig. 5 The energy consumption of the supercomputer simulation with respect to the scale of the system. Due to the linear growth of the simulator power (p_{sys}) and the linear growth of the time multiplier (M_{time}) on the supercomputer, the energy consumption of the simulator (E_{sim}) exhibits a quadratic growth, while the corresponding growth in a biological brain is approximately linear.

As a result, the power gap between the biological brain and the supercomputer simulator will widen as the system grows as illustrated in Fig. 5. Although it is possible to suppress the linear growth of M_{time} by improving the latency and bandwidth of the node interconnect, it usually comes at a cost of increasing P_{sys} and thus it does not necessarily improve E_{sim}. Therefore, such brain-scale simualtion in the supercomputer is not scalable.

2.3 Neuromorphic Hardware: Non-von Neumann Architecture

The power gap between supercomputers and biological brains comes from the difference in the system architecture. Particularly, each computing node within the supercomputer based on vN architecture represents a centralized computing unit (of one or more CPUs) in charge of a large number of neurons and their associated synapses. To reduce this power gap and design scalable neuromorphic systems, a more holistic approach has to include the renovation of the hardware architecture.

2.3.1 Motivation

Enhancement of vN machines is not an effective approach in neuromorphic systems. Suppose the amount of (fast) memory is limited so the nodes have a fixed number of neurons and synapses, and the work required to be done is also fixed. To enhance throughput, more cores can be added to each node. Let *n* be the number of computing cores in a node. The throughput of this node increases when it contains more cores: $M_{time} \propto n^{-\beta}$ in which $\beta > 0$. On the other hand, a linearly larger power per node (p_{node}) is consumed due to more cores. Therefore, the energy consumption per node for one second of simulation (e_{sim}) is

$$e_{sim} = p_{node} * M_{time} \propto n^{1-\beta} \quad (5)$$

From the runtime data of the Dawn simulation in Fig. 6, $0 < \beta \leq 0.58$. Since $1 - \beta > 0$ for all cases, energy consumption per node increases with number of computing cores (n) sub-linearly

$$e_{sim} \propto n^{0.42} \text{| in Dawn simulation} \quad (6)$$

As a result, powerful (and power hungry!) nodes in vN machines are not favored for brain simulation since given the same amount of work, e_{sim} increases with *n*. On the contrary, n should be small to reduce energy consumption. In Dawn simulation, each node is about $5*10^4$ neurons and $2.5*10^8$ synapses, while within the brain each node is a single neuron with 10^3-10^4

synapses. Therefore, the node in the supercomputer is $\frac{n_{Dawn}}{n_{Brain}} = 5 * 10^4$ times more powerful than a single neuron in the brain, making n and e_{sim} extremely large. Not surprisingly, this conclusion agrees with the fact that a biological brain is very power efficient, as compared with the vN machine simulation. Therefore, to capture the nature of the biological brain and approach a similar energy efficiency, a highly distributed NvN architecture is required, in which each processing unit is significantly less powerful (n is small), representing only a few neurons, or ideally one single neuron.

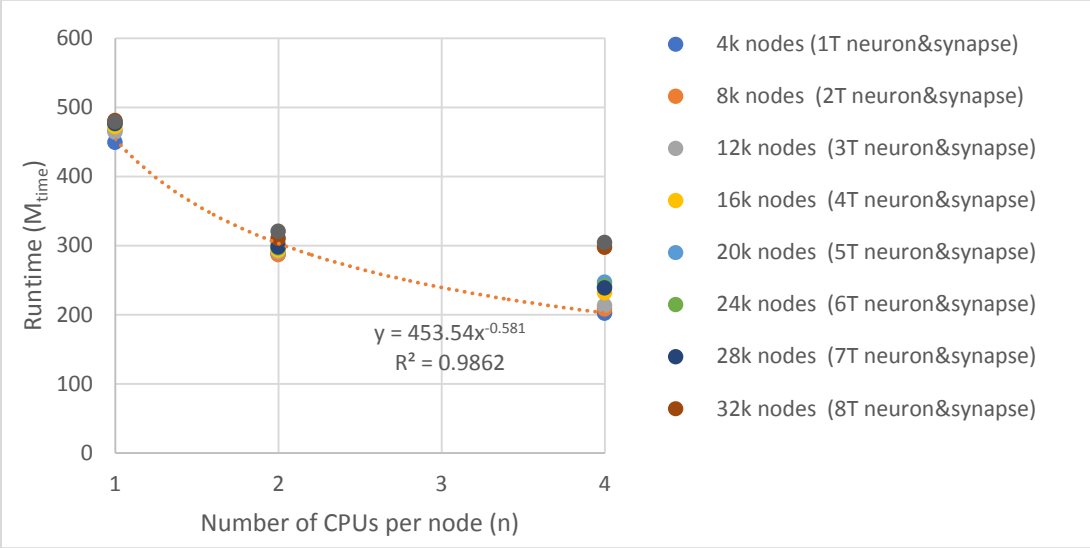


Fig. 6 The runtime (M_{time}) of the BlueGene/P simulation for one second real time with respect to the number of CPUs in each node (n) [17]. The runtime decreases sub-linearly with the increasing number of CPUs in the node.

2.3.2 Non-von Neumann Architecture Hardware Implementation

Most state-of-the-art neuromorphic systems are based on NvN architecture using a larger number of smaller cores on the chip. As a result, these neuromorphic systems exhibit reduced power consumption. One example is the IBM TrueNorth chip which contains 1M neurons and 256M synapses. The TrueNorth chip consumes only 0.2W during image recognition operations using the Energy Efficient Deep Network (EEDN) [16], and reduces the energy per synaptic

event by $1.76*10^5$X from the state-of-the-art supercomputer simulator [4]. In this chip, every 256 neurons are grouped into a core of 256-by-256 virtual crossbar to realize 256 directly connected synapses for each neuron in a block-wise fashion. Each core has an independent route and RAM for synaptic weight storage. 4,096 of these cores are placed on the chip and arranged in a 2D array to extend the on-core 2D mesh topology of the neurons, as shown in Fig. 7.

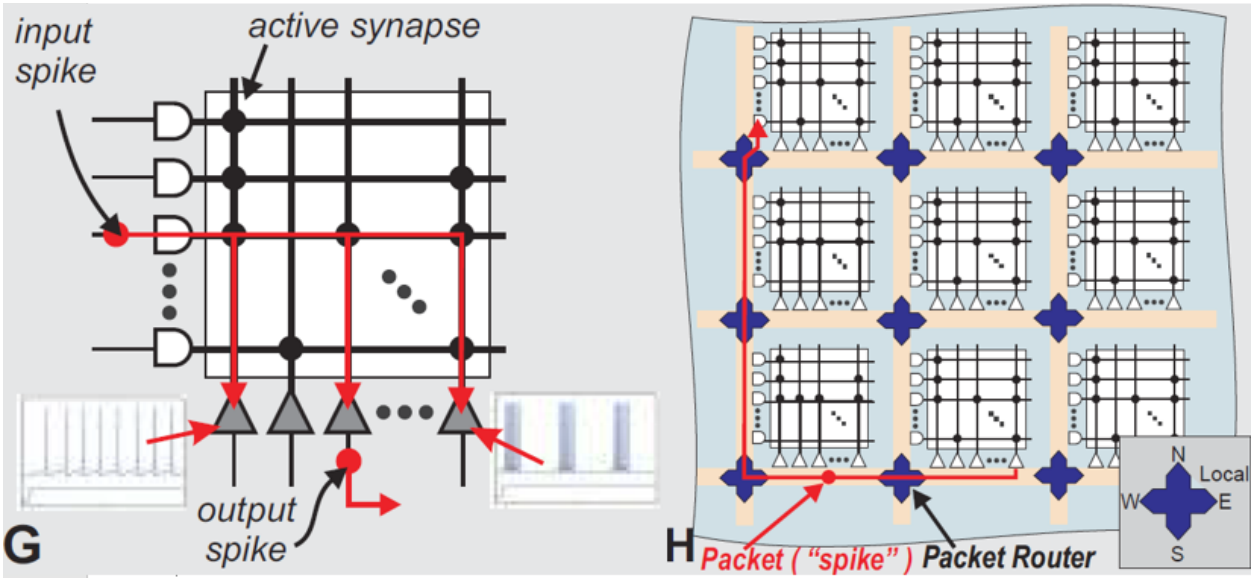


Fig. 7 Left: The crossbar structure in the TrueNorth core. Each core in the TrueNorth chip has 256 neurons and 65,536 synapses (256 synapses per neuron). Right: Each TrueNorth chip has 4,096 of these cores arranged in a 2D-mesh (from [4]).

Due to the small scale of each chip, the EEDN network implemented on the TrueNorth system has a very limited image recognition capability; recent studies show that it is able to classify 32x32 RGB images from 100 classes of single objects, at around 1,500 frames per second [18]. To design a more powerful network, more neurons are required for a larger network and consequently many neuromorphic chips need to be integrated to scale out the overall neuromorphic system for better cognitive capability. In the following chapter, various chip-integration technologies are reviewed and compared.

3. Non-von Neumann System Integration

Since the NvN architecture has distributed neurons and synapses, it can be readily scaled-out using more chips as long as the interconnects among the chips are adequate for efficient chip-to-chip communication. In this chapter, we first give a brief overview of the integration technologies for scaled-out neuromorphic systems, namely the PCB integration 2DI (Subsection 3.1), on-wafer integration (Subsection 3.2) and the 3D wafer scale integration (Subsection 3.3). Then we evaluate the interconnects implemented by these technologies (Subsection 3.4) and discuss the tradeoff between connectivity and energy consumption (Subsection 3.5).

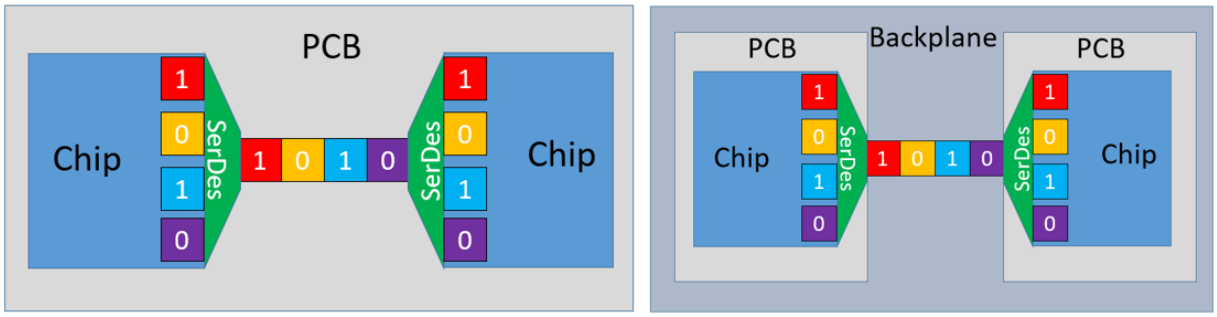


Fig. 8 Serializer/Deserializer (SerDes) used to connect different chips on a circuit board (left) or to connect different circuit boards on a backplane (right).

3.1 Two-Dimensional Circuit Board Integration (2DI)

2DI is the dominant chip integration scheme and it is widely used in system integration nowadays, including neuromorphic system integration, such as in the Neurogrid 16-chip system, SpiNNaker 48-chip system and the TrueNorth 16-chip system [4,7,8]. In 2DI, up to dozens of chips can be integrated on a PCB depending on the size of the chip and the capacity of the board. To go beyond a single PCB, multiple PCBs can be integrated on the backplane(s). The chips on the circuit board(s) are interconnected by SerDes to allow high bandwidth channels from limited number of pins available at the chip-board interface (Fig. 8). As a result, the interconnect density is limited by the number of the pins of the packaged chips and the on-chip area occupied by

SerDes circuits. Typical package-board interconnect pitch (*e.g.* ball grid array) is about 400µm [6]. A high speed SerDes used for chip level high speed communication (>20Gbps) typically occupies around $1mm^2$-$3mm^2$ on-chip area [19]. 2DI is not the optimal solution for neuromorphic systems due to low power efficiency. For example, the TrueNorth 16-chip board (NS16e) dissipates more than 70% of its power due to the overhead on the board [11].

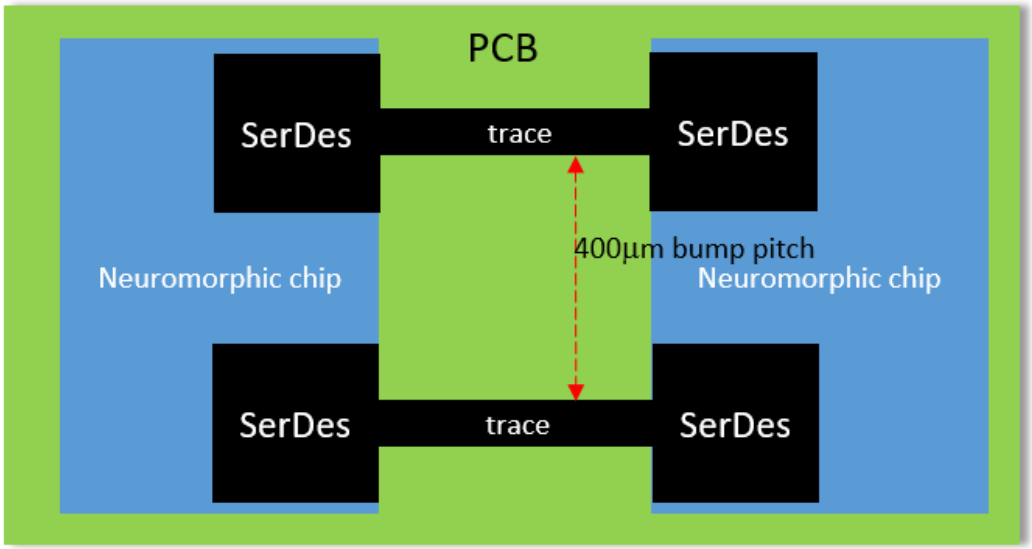


Fig. 9 An example of the schematics showing neuromorphic chips integrated using the SerDes on the PCB.

An example of a 2DI schematic is shown in Fig. 9 . Specifically, the SerDes circuit from [19] takes $3.34mm^2$ of the die area for each channel and provides a maximum of 28Gbps bandwidth while consuming 0.56W of power during operation. The important parameters for the SerDes are summarized in Table 2.

The connectivity, which is the aggregate bandwidth of the die divided by the area, can be used as a figure of merit for the die-to-die interconnect. For SerDes, two cases need to be considered: the area-limited case and the pitch-limited case. Assuming that up to 25% of the die area can be used for SerDes, the connectivity of such die is $25\% * \frac{28Gbps}{3.34mm^2} = 2.1Gbps/mm^2$. However, suppose that further technology scaling can reduce the footprint of the SerDes

significantly, such that the only constraint is the C4 pitch ($\sim 150\mu m$) at the die peripheral, then the maximum connectivity for this die is $4 * \frac{\sqrt{\text{Die Area}}}{150\mu m} * \frac{28\text{Gbps}}{\text{Die Area}} = \frac{112\text{Gbps}}{150\mu m\sqrt{\text{Die Area}}}$. For a die area between 25mm^2 (5mm x 5mm) and 625mm^2 (25mm x 25mm), the connectivity is about 30Gbps/mm^2 ~ 150Gbps/mm^2.

Table 2 Parameters of the SerDes and Fine Pitch Interconnect Used for Analysis

	Wire pitch	Area per Channel	Data Rate	Energy/bit
SerDes [19]	$400\mu m$(chip-board) $150\mu m$ (die-package)	3.34mm^2	28Gbps/channel	20pJ/bit @ 28Gbps 136pJ/bit @ 1.25Gbps
FPI [23]	$2\mu m$	6μm^2/wire (horizontal) 2μm^2/wire (vertical)	1Gbps/wire	0.2pJ/bit @1V, 1Gbps

3.2 On Wafer Integration (OWI)

In OWI technology, the silicon dies can be integrated on the wafer using a BEOL process, without dicing or packaging. In addition, silicon dies can be integrated on Si-based interconnect-fabric. This interconnect technology takes advantage of the BEOL process such that the dies on the same interconnect fabric are integrated as if they are on the same wafer. OWI is high compatible with a NvN architecture because this architecture uses replicates of the neuromorphic circuitry and arranges these replicates in a regular mesh, making it suitable for the current lithography process. OWI can, therefore, be used for neuromorphic systems if the yield of the dies and the interconnect is optimized to an acceptable level. One example of the neuromorphic system using OWI is the FACETS project [9]. In the FACETS project, the neurons are fabricated in the reticles on a 200mm wafer and the reticles on the wafer are interconnected using a wafer level, 5μm pitch, fat wires (Fig. 10).

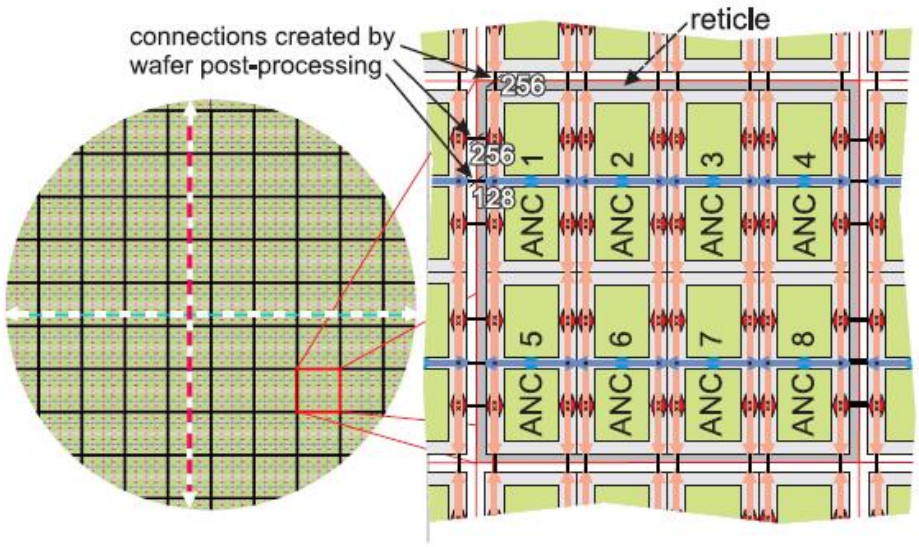


Fig. 10 On-Wafer-Integration in the FACETS project. To fabricate interconnect between reticles, wafer-level BEOL process is used as the post-processing (from [20]).

Another example of OWI technology is the CHIPS Si interconnect fabric using the SuperCHIPS protocol [10, 23]. Bare known-good-dies (KGDs) are integrated directly on a silicon substrate with wiring patterns that are fabricated through BEOL processes. Since KGDs are used, the issue of yield on the wafer is eliminated. In addition, the Si interconnect fabric can be populated by heterogenous KGDs that might originate from different foundries using disparate technologies.

By using OWI instead of 2DI, the pitch of the chip-to-chip interconnect can be greatly reduced, by using ta fine pitch interconnect (FPI), from 400μm (limited by the soldering process) to around 1μm (limited by wafer level overlay). FPI relieves the system from the need of SerDes because a single high speed SerDes link can be replaced with dense parallel horizontal on-wafer Cu wiring. Such change not only increases the aggregate data rate, but also reduces the route detour of a NvN architecture, as shown in Fig. 11.

It is assumed here that the OWI interconnects exhibit a wire width of 1µm and pitch of 2µm, comparable with the fat metal wiring in a BEOL process. The 2µm pitch parallel Cu wires can effectively replace 400µm pitch SerDes links if each Cu wire can sustain more than 1/75 of the data rate supported by the corresponding SerDes link. Given that in OWI technology the connected dies can be placed very close (<100µm) since the package is eliminated, the data rate of each of these short wires can go up to 1Gbps with negligible insertion loss and crosstalk (Fig. 12). Transit delay of such short channels is also expected to be negligible.

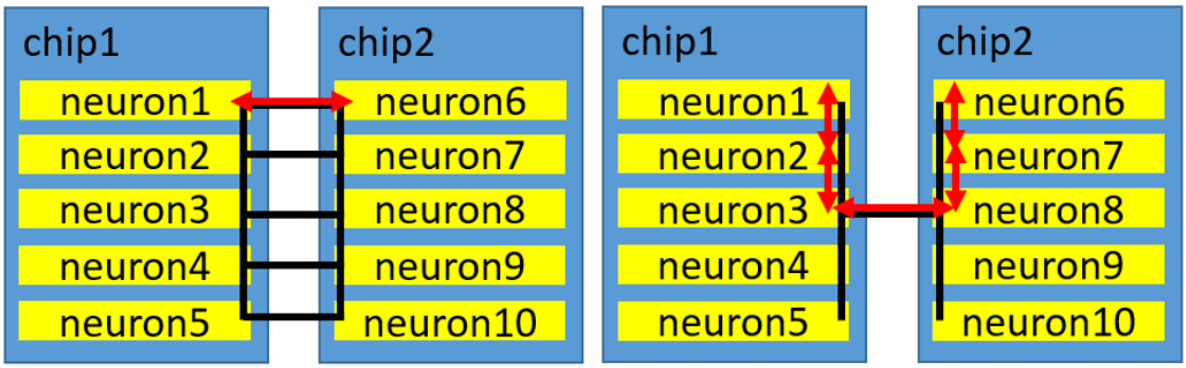


Fig. 11 Path Detour in NvN Architecture: for a spike package traveling from neuron1 to neuron6, no detour is needed for the system with high interconnect density (left) in contrast to the low interconnect density system (right).

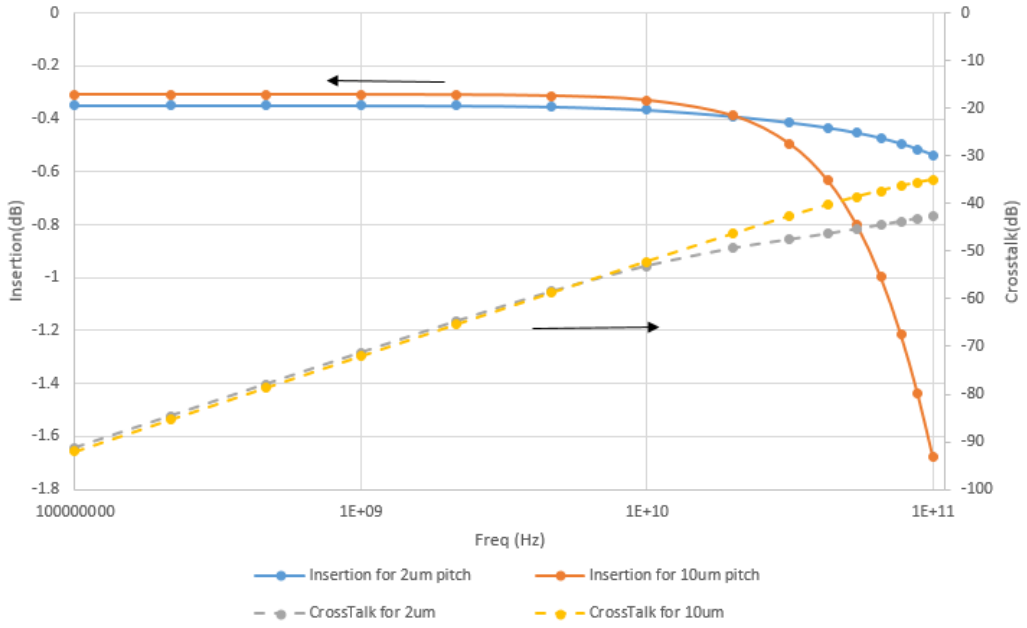


Fig. 12 Insertion loss and crosstalk from FPI HFSS simulation: for 2um Cu wire pitch in SiO2 substrate at 1GHz, insertion loss is -0.35dB, crosstalk with one-layer shielding is -71dB, resistance is 2Ohm and capacitance is 14fF per wire (from [23]).

An example of the OWI schematic is shown in Fig. 13. Specifically, the horizontal FPI, including the necessary driver and pad, occupies 6mm^2 of the die area for each channel and it provides 1Gbps bandwidth while consuming the energy of 0.2pJ/bit [23]. The parameters of the horizontal FPI are listed in Table 2. The connectivity is $4 * \frac{\sqrt{\text{Area}}}{2\mu m} * \frac{1Gbps}{Area} = \frac{4Gbps}{2\mu m\sqrt{Area}}$. For a die area between 1mm^2 (1mm x 1mm) and 625mm^2 (25mm x 25mm), the connectivity is about 40Gbps/mm^2 ~ 2000Gbps/mm^2, about 10X ~ 1000X higher than the connectivity of the 2DI case.

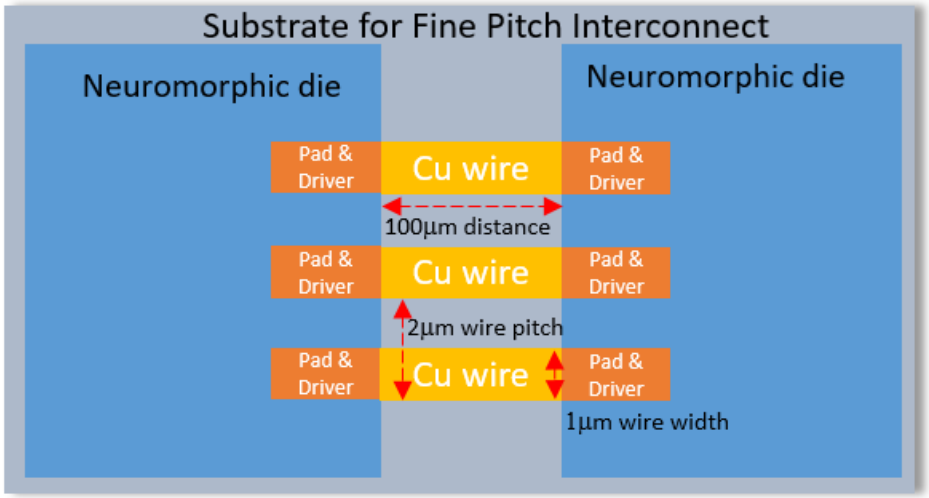


Fig. 13 A schematic of neuromorphic dies integrated on-wafer using fine pitch interconnect.

3.3 Three-Dimensional Wafer Scale Integration (3D-WSI)

One limitation of the OWI technology is that it is not scalable for multi-wafer systems. The substrate is limited to the size of a standard Si wafer due to the requirement of the BEOL process. If a 300mm wafer can be used for integration using a state-of-the-art design (*e.g.* the IBM TrueNorth chip), the 300mm wafer can hold about 100 such dies, which is equivalent to 100M neurons and 25.6B synapses – still more than 100X fewer than the scale of the human

brain. It is possible to use edge connectors (*e.g.* peripheral component interconnect express) to interconnect wafers as shown in Fig. 14, but the pitch of the edge connectors is large (at least in the order of millimeters). As a result, the bandwidth between wafers will be limited by the large edge connector pitch. The pitch limitation can be compensated by using high speed SerDes channels, with the cost of increased power consumption, similar to the chip-to-chip communication on the PCB.

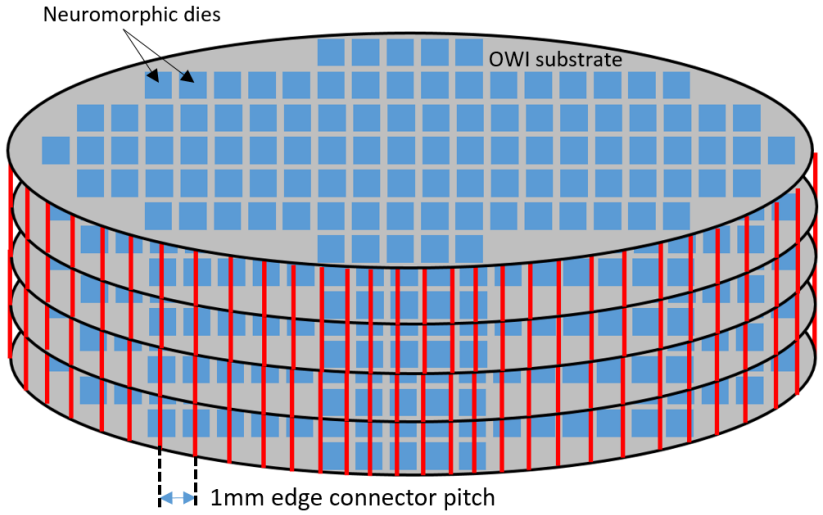


Fig. 14 Use of edge connectors to integrate a multi-wafer OWI system.

To further extend the system with scalable bandwidth, 3D-WSI technology, exhibiting many attractive properties, can be exploited. A simplified process flow of 3D-WSI is shown in Fig. 15. The silicon wafer strata1 is first processed at the front-side. Then, the wafer is bonded to a handle wafer at the same side. Afterwards, the strata1 is thinned from the backside by grinding from around 700µm (300mm wafer thickness) to about 7µm to accommodate 1µm diameter (2µm pitch) through-silicon-vias (TSVs) with proper aspect ratio (7µm: 1µm) for a Cu plating process. The thinning is followed by a backside oxide/dielectric deposition and CMP. Strata1 is then bonded (using the back side) to the front-side of the strata0. Finally, the handle wafer of strata1 is removed prior to further fabrication of the metal layer and TSVs at the front-side. The

top of this stacked wafer pair is similar to the top of strata0, and therefore additional wafers can be bonded to this stack in series.

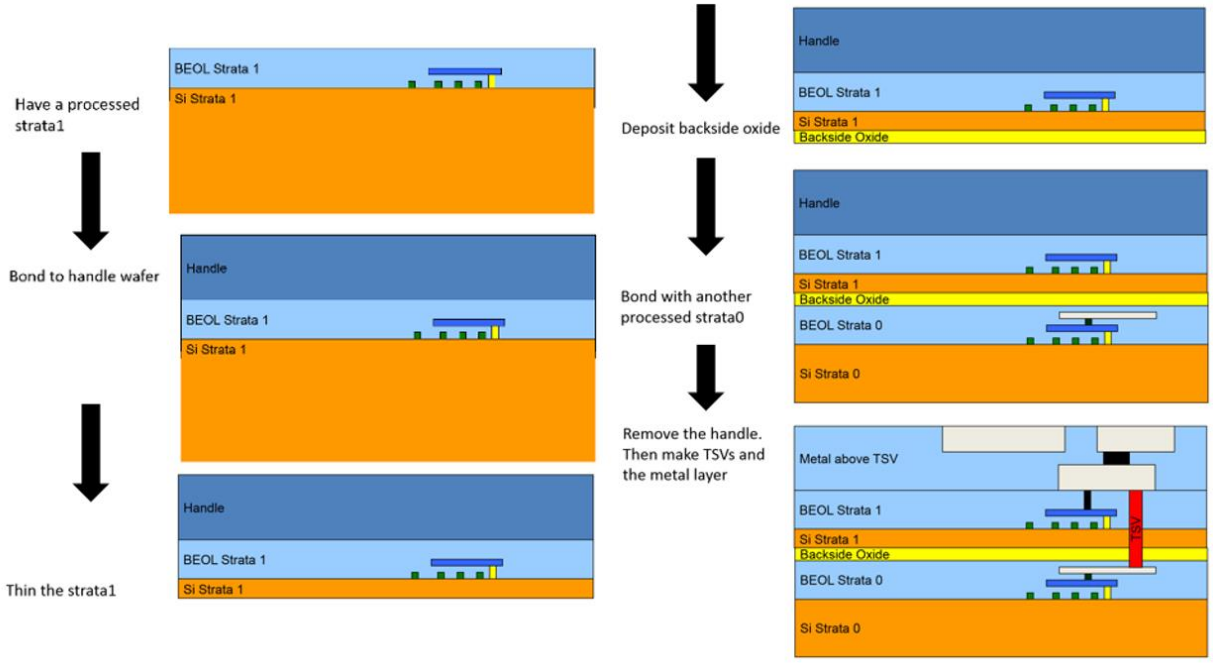


Fig. 15 Sample process flow of the 3Dimensional Wafer Scale Integration [21].

While the TSVs serve as the vertical interconnect channels, for each wafer in the wafer stack, horizontal interconnects are designed similarly to the interconnects in the OWI ,using the BEOL process to achieve the pitch of a fat wire. The vertical interconnect pitch is determined by the TSVs. The state-of-the-art product using 3D-WSI has made 5μm TSVs [13]. TSVs of 1μm diameter with 2μm pitch, comparable to the horizontal interconnect pitch of OWI, are currently in development. As required for the 1μm diameter TSVs, each wafer in the stack is thinned to ~10um. Consequently, the system is very compact in vertical dimension. A 100-layer wafer stack has a thickness of a few millimeters. Although such compactness of the wafer stack will pose a thermal challenge in high-performance systems, in low power neuromorphic systems, the issue is manageable [22].

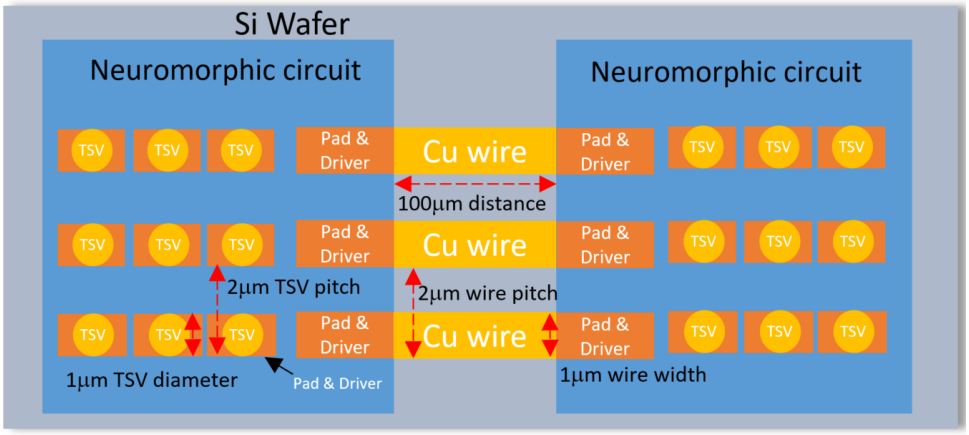


Fig. 16 A schematic of neuromorphic circuits integrated on-wafer using horizontal FPI and vertical TSVs.

An example of the 3D-WSI schematic is shown in Fig. 16, which is in fact a OWI with the addition of TSVs. Due to the short length of TSVs (<10μm) as compared with the horizontal FPI (~100μm), the drivers required for TSVs are therefore smaller. The on-die area occupied by each TSV channel is $2\mu m^2$. The energy consumption of each TSV channel is assumed to be 0.2pJ/bit, similar to the horizontal FPI. Suppose up to 25% of the die area is consumed by the TSVs, the connectivity is now mainly contributed by the TSVs and it is approximately $\frac{1}{4} * \frac{1}{2\mu m^2} * 1Gbps = 1.25 * 10^5 Gbps/mm^2$, more than 1,000X larger than the connectivity of the 2DI system. A detailed plot of the connnectivity and (square) die size for all the mentioned system integration schemes is shown in Fig. 17.

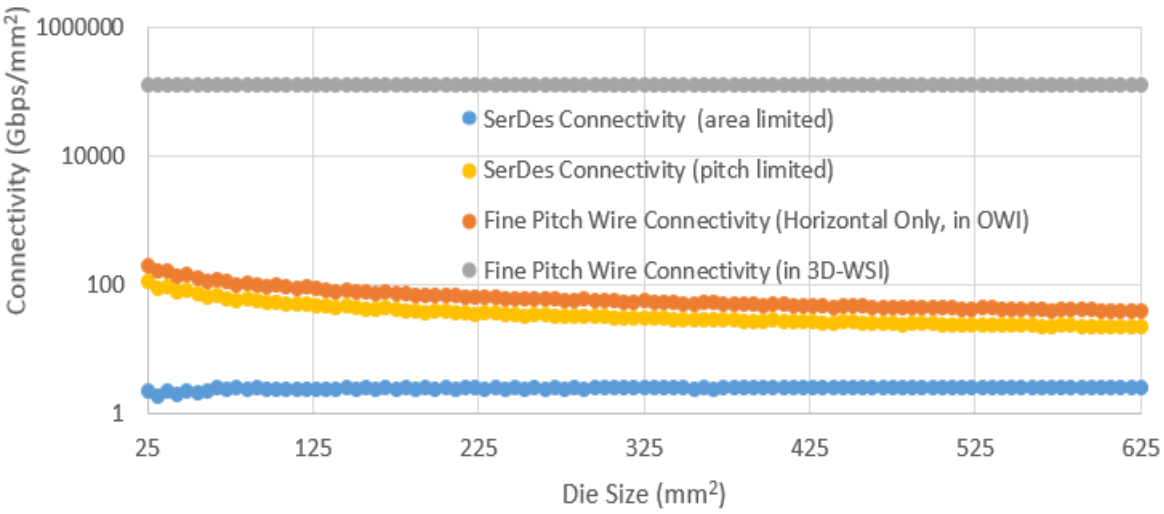


Fig. 17 Connectivity vs. die size (25% die area for SerDes for 2DI or TSVs for 3D-WSI: FPI exhibits improved connectivity by 10X-10,000X, depending on the die size.

3.4 Connectivity and Energy/bit Tradeoff

For a single digital communication channel operating at very high frequency (> 10GHz), the bit rate (f) and the energy/bit (P_{bit}) are related according [10]

$$P_{bit}(f) = \frac{P}{f} \propto CV^2 e^{Kf} \qquad (7)$$

Thus, there is a tradeoff between the single-channel data rate and the energy/bit. Different types of interconnect have different reference points.

Based on the data and calculation presented in previous sections, the reference points of different types interconnect are depicted in Fig. 18. It can be seen from Fig. 18 that as connectivity increases (supported by advanced technologies), the energy/bit drops and the channel is more power-efficient. The technology with low interconnect density requires high speed operation of each channel, and thus suffers from high power consumption associated with the high-frequency overhead.

In conclusion, the technology that provides the highest connectivity, 3D-WSI, is the best candidate to integrate an ultra-large multi-wafer scale neuromorphic system. In the following

chapter, a structure for a biological scale neuromorphic system is proposed. The proposed tructure includes connectivity on the scale of a biological brain and is based on the 3D-WSI technology. This approach is compared with a similar system using a 2DI technology.

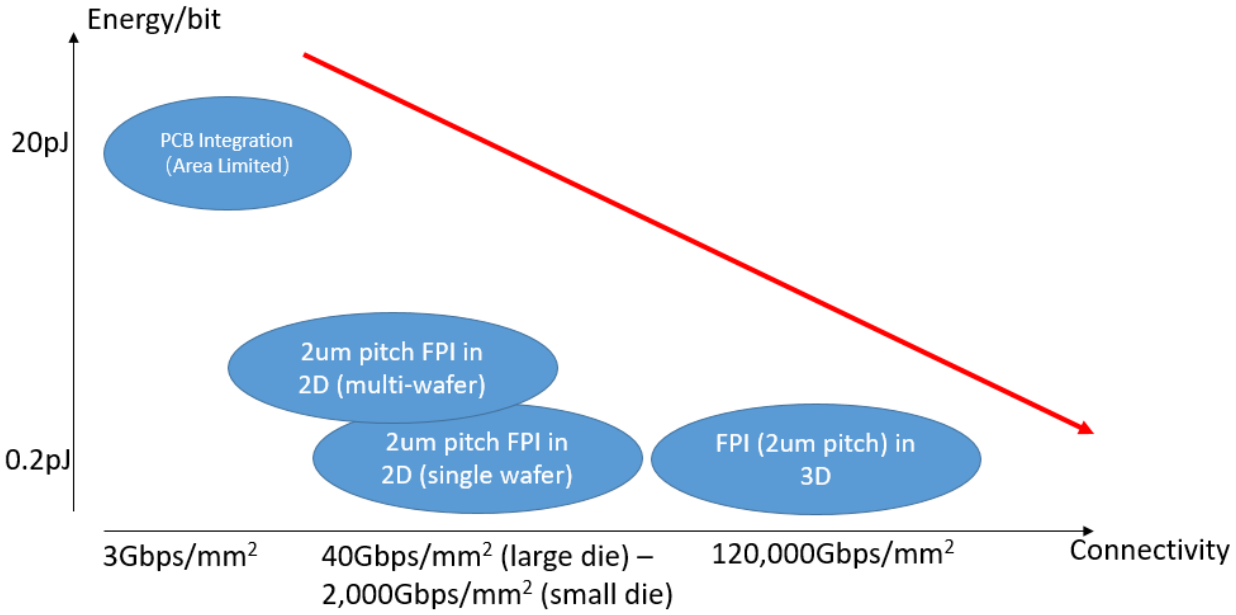


Fig. 18 Energy/bit and connectivity of different technologies with NvN architecture: energy per bit decreases as connectivity increases using FPI.

4. Neuromorphic System Model

This chapter deals with the scaling-out of neuromorphic systems using the 2DI and 3D-WSI technologies described in the previous chapter. Physical systems integrated using 2DI and 3D-WSI technologies and properties of interest related to these systems, such as the system scale, communication latency, bandwidth and communication power consumption, are reviewed in subsection 4.1. Short-range and long-range connectivity that are the backbone of neuromorphic system operation, are described in subsection 4.2. A summary is provided in subsection 4.3. In Chapter 5, the models are simulated and the results are discussed.

4.1 Proposed Neuromorphic Systems

Although three integration methods are discussed in the last chapter. The OWI technology is essentially an intermediate stage before the 3D-WSI technology, which provides integrated single-wafer subsystems to be further integrated by the 3D-WSI technology. Therefore, we propose two types of neuromorphic systems, the 2DI system and the 3D-WSI system.

4.1.1 Physical Layout of the Neuromorphic Systems

Since ultra-large-scale system is the ultimate goal, it is crucial to use simple and straightforward physical layout. In a multi-chip 2DI neuromorphic system based on, for example, the IBM TrueNorth chip, 16 such chips are integrated on a PCB and interconnected as a 4x4 2D array with the help of FPGA [11]. To further scale-out the system, we can use backplane(s) to integrate the PCBs and use the SerDes [19] for interconnection. Consequently, each board has at least 12 SerDes for 12 board-to-board links to implement the 3D mesh as shown in Fig. 19.

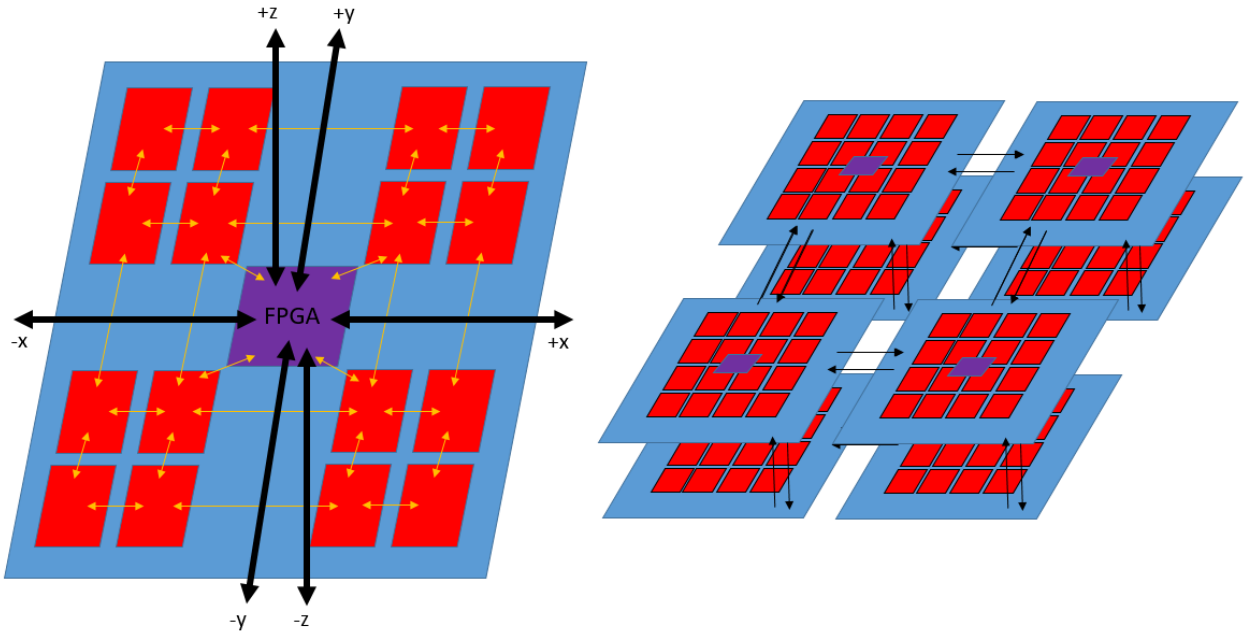


Fig. 19 Schematics of the 2DI system. Multiple 16-chip boards are interconnected in a 3D mesh using SerDes. Red chips are the neuromorphic chips. Purple chips are the FPGA chips that serve as the interface between the boards. Yellow arrows represent the on-board interconnects. Black arrows represent the board-to-board interconnects. The right figure has omitted the on-board interconnects.

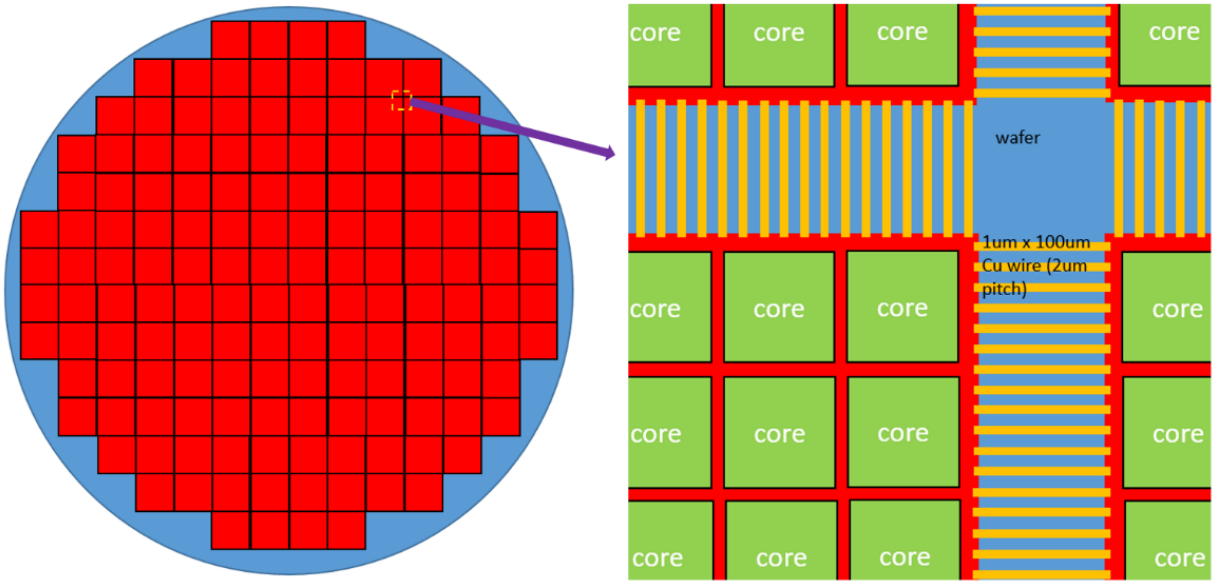


Fig. 20 Schematics of the on-wafer fine pitch interconnect (FPI). The on-wafer FPI (wires in yellow) connects the neuromorphic circuits (cores in green) in horizontal x and y directions.

In contrast, the neuromorphic system scaled-out by 3D-WSI does not require SerDes or FPGA. In 3D-WSI neuromorphic systems, the neuromorphic dies are fabricated directly on the wafer and connected using the 2μm fine pitch interconnect in the horizontal *x* and *y* axes as shown in Fig. 20. The wafers are stacked using the 2μm pitch TSVs, as shown in Fig. 21.

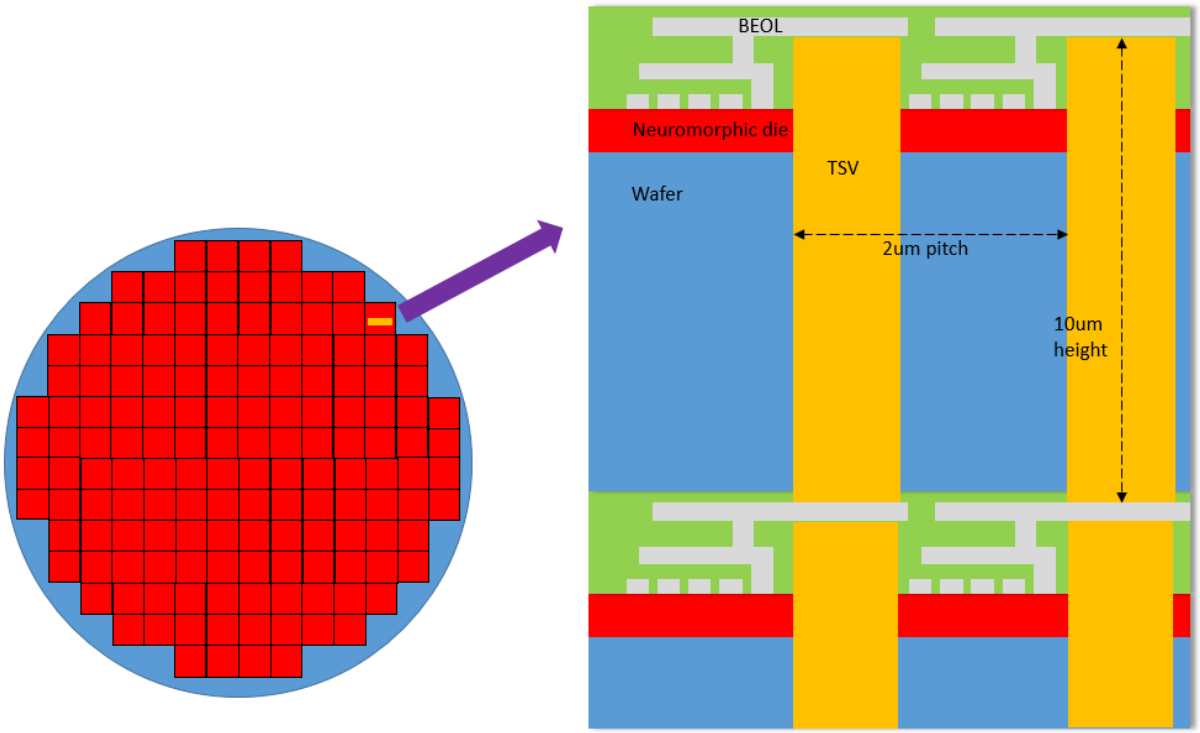


Fig. 21 Cross section of a wafer stack, showing how neuromorphic dies are vertically interconnected using TSVs (in yellow) within 3D-WSI systems.

We introduce the express lanes of the TSVs. An express lane, formed by multiple stacked TSVs, goes across wafers without intermediate stops, to facilitate long-range communication. The cost of express lanes using TSVs is negligible since TSVs are very short ($\sim 10\mu m$) and each TSV occupies a small area ($< 10\mu m^2$).

The structure of an express lane is shown in Fig. 22. In this example, there are three links realized using four TSVs. Die1, Die2, and Die3 are directly stacked in series (they have the same *x* and *y* coordinates on the respective wafers). The blue TSV (in Fig. 22) serves as the link

between Die1 and Die2, and therefore, it connects to the routers belong to those dies. Similarly, the yellow TSV is the link between Die2 and Die3. For Die1 and Die3, the green TSVs form an express lane. This express lane goes through Die2 without connecting to the router on Die2. The signal propagating in the green TSV is enhanced by a repeater on Die2. As a result, the signal traveling through the express lane is accelerated since it is not required to wait for the routing logic on Die2.

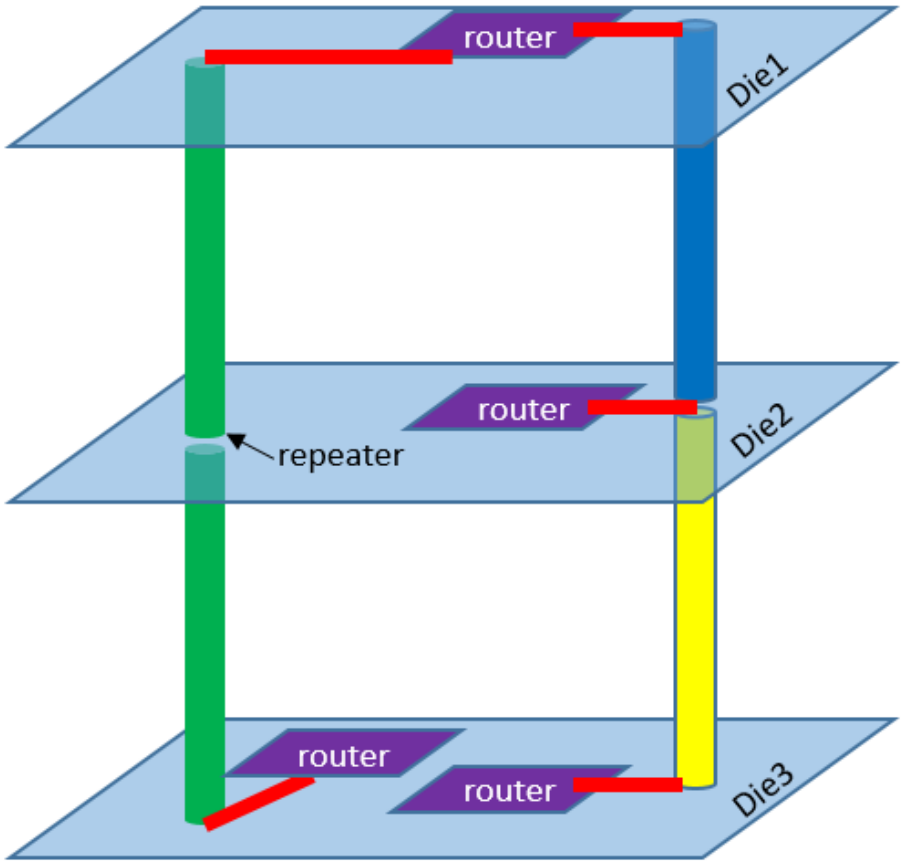


Fig. 22. A schematic of an express lane realized using TSVs in the 3D-WSI system. Three dies, Die1, Die2 and Die3, are spatially in the sample column but on neighboring wafers. The blue and yellow TSVs form the connections between Die1 and Die2, and Die2 and Die3, respectively. The express lane is the green link, which is composed of two physical TSVs. This express lane starts from the router on Die1 and terminates at the router on Die3, without stopping at the router on Die2.

This realization of express lane is hard-wired due to the pre-determined configuration of the on-die repeaters. For all the 3D-WSI systems, a sufficient number of express lanes between every two wafers is assumed. The additional die area occupied by the express lanes is shown, by the later simulation, to be negligible.

Comparing with the express lanes in the *z* direction in the wafer stack, express lanes in *x*, *y* directions can also be built. However, their penalty in area is high as compared with the TSV based express lanes due to large horizontal distance. Specifically, an on-wafer express lane in *x* or *y* direction will require additional take more real estate on the dies that are bypassed by the express lanes. Express lanes in 2DI systems will require high-speed long-range communication links, such as ethernet, which is very power hungry. To reduce the additional design complexities of the systems due to the express lanes, only TSV-based express lanes (described in Fig. 22) will be considered in this work.

4.1.2 Scale of the Systems

It is important to keep a reasonable synapse-to-neuron ratio for neuromorphic system scaling. To approach the level of 10^3-10^4 synapses per neuron similar to a human brain, we treat each TrueNorth-like die (chip) as 256K neurons and 256M (the exact number should be 256K*1,024) synapses in total, instead of the original 1M-neuron, 256M-synapse configuration [4]. Due to the crossbar architecture, the total number of synapses is fixed, and therefore the number of neurons in inversely proportional to the number of synapses per neuron. This property of the system can be used to adjust the number of synapses per neuron. In our model, we assume that each 16 TrueNorth cores can be treated as a single modified core to effectively increase the number of synapses per neuron from 256 to 1024, and by such modification we have reduced the number of neurons in the same circuit by 4x (from 16 * 256 to 4 * 256), as illustrated in Fig. 23.

Consider a brain that has 10 billion neurons, and that each neuron has 1024 synapses. Four cases are listed in Table 3 in which the latter three cases are evaluated. It is assumed that the wafers in the 3D-WSI systems are not fully populated such that the number of neurons hosted in the 3D-WSI system is equivalent to the corresponding fully populated 2DI system. For instance, 1%-brain is four 90% populated wafers (each hosting 133 dies out of the capacity of 148) or 27 100% populated 16-chip boards. The required on-chip memory per chip slightly increases due to the expansion of the address spaces which uses merely about 6% of the total on-

Table 3 Scale of the neuromorphic system integrated using PCBs or wafers.

Case	# of 16-chip boards (2DI)	# of wafers (3D-WSI)	# of neuron	Total on-chip memory (Gb)	on-chip memory per chip (Mb)
1-board	1	0.11	4.1M	6.9	4.42
1% brain	27	4	0.111B	188.8	4.47
10% brain	266	32	1.09B	1874.6	4.51
90% brain	2128	266	8.72B	18894.2	4.55

chip memory (most memory is used to store the status of the spike signal).

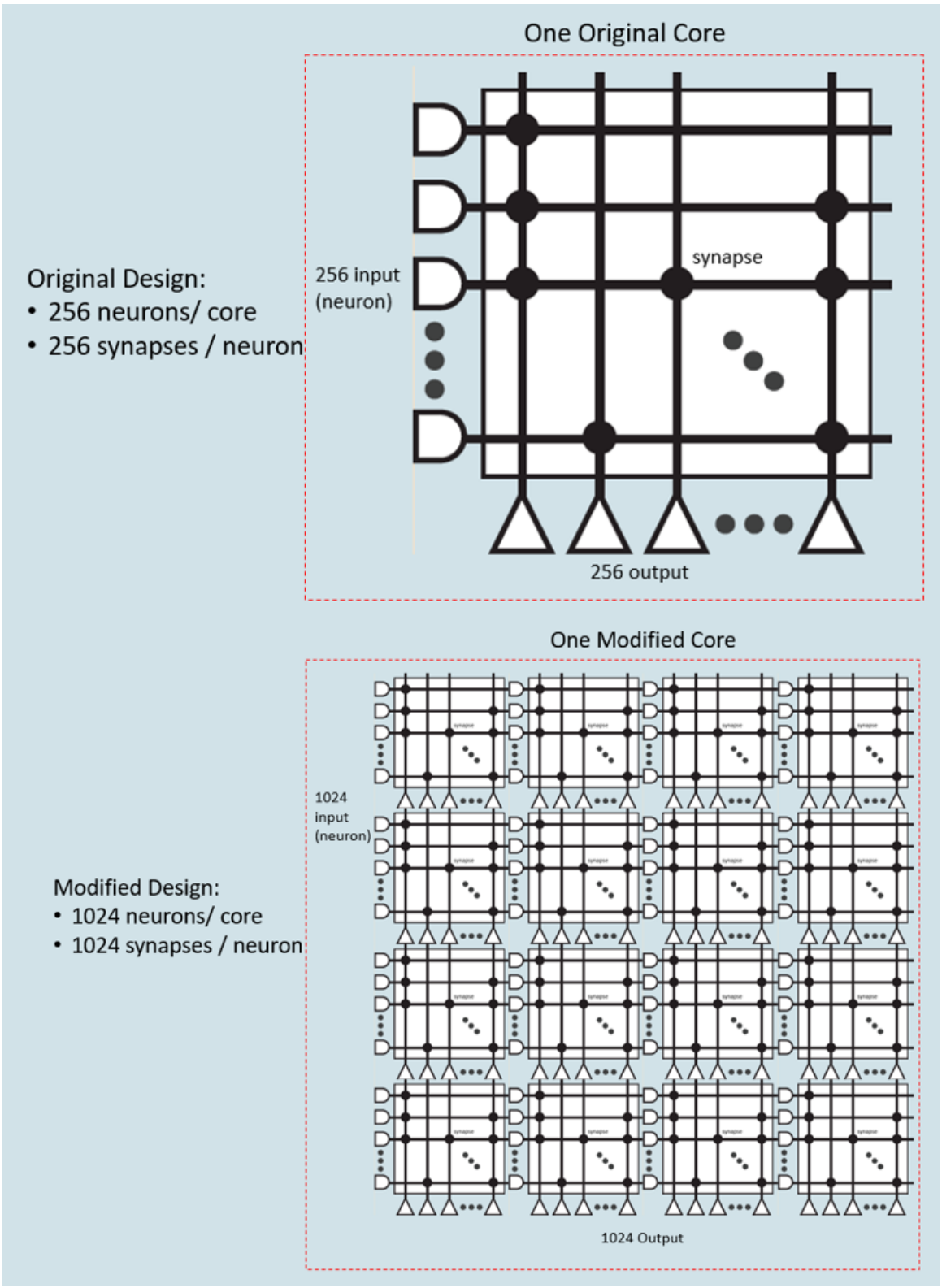


*Fig. 23 Modifying 16 cores in the TrueNorth chip to accommodate 1024 neurons with 1024 * 1024 synapses.*

4.1.3 Communication Latency of 2DI and 3D-WSI Systems

Based on the physical layout of the system, the communication latency of the connection (*a.k.a.* the route of the spike) between any two neurons is modeled. We call each chip (or die) a node and therefore a node is a chip in 2DI system and a node is a die in 3D-WSI systems.

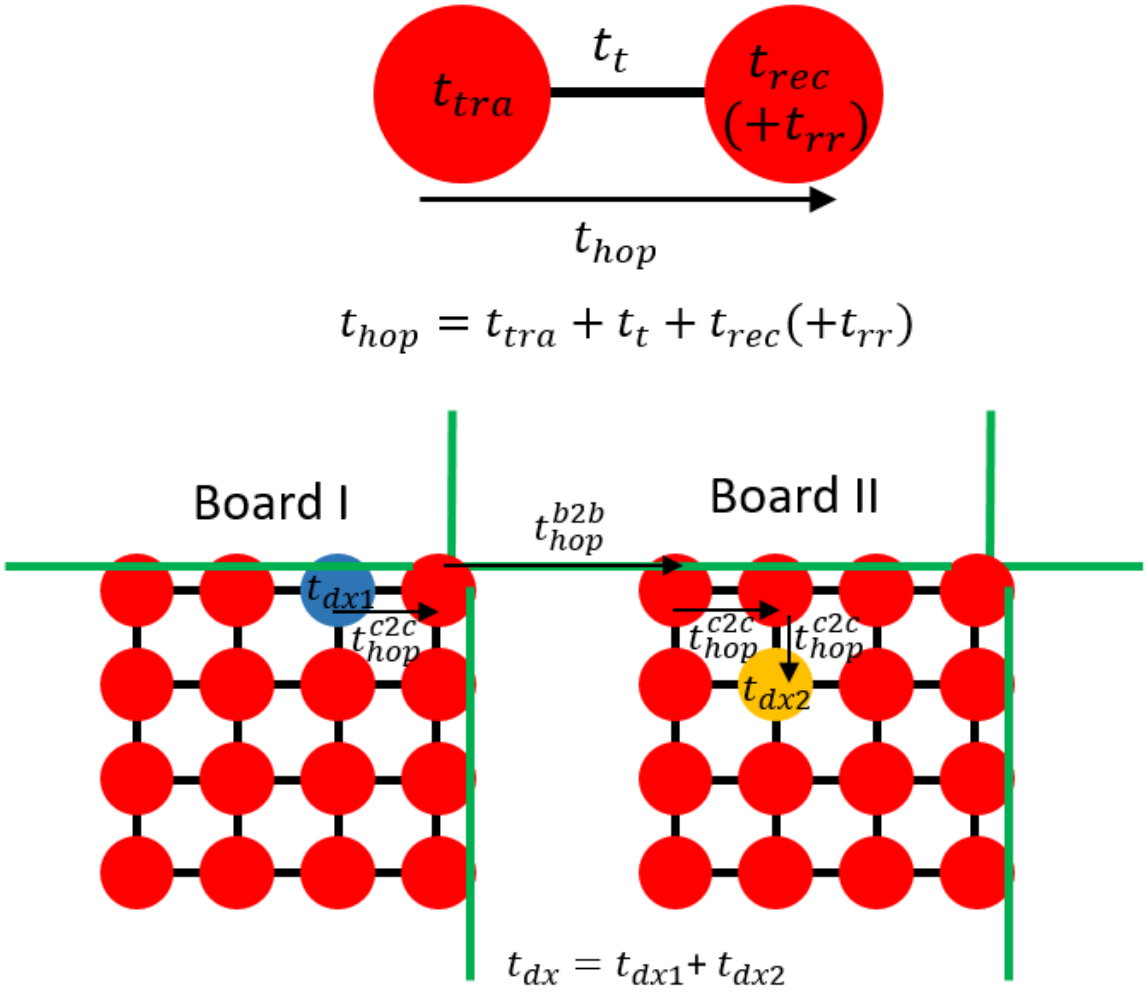


Fig. 24 Model for communication latency. Top: the components of the node-to-node communication latency for one hop. Bottom: the components of the node-to-node communication latency for multiple hops - from the blue node to the yellow node. In 2DI and 3D-WSI system, the components will be the same, but the values are different from each other.

The node-to-node transmission occurs when a neuron in one node needs to send a signal to a neuron in another node. The connection latency is divided into delay components for the 2-DI system, as depicted in Fig. 24. For one-hop transmission, the signal first leaves the starting node in t_{tra} (transmission time), which includes the time to encode and serialize the signal for the transmission. Then the signal propagates through the channels, either a chip-to-chip (c2c) channel or a board-to-board (b2b) channel, with a delay of t_t^{c2c} or t_t^{b2b} representing the chip-to-chip or board-to-board transit time. When the signal reaches nodes on its path, a t_{rec} (receive time due to the de-serialization of the signal) is added to the delay. Additional rerouting time t_{rr} is required if the node is not the correct destination of the signal. In addition, the time for the signal to be converted between the computation and communication domains is t_{dx}, which is the sum of the domain-crossing time at the starting node (t_{dx1}) and the domain-crossing time at the destination node (t_{dx2}). This entire process is identical for c2c and b2b connections. The latency of each hop for a b2b link (t_{hop}^{b2b}) and for a c2c link (t_{hop}^{c2c}) is

$$t_{hop}^{b2b} = t_{tra}^{b2b} + t_t^{b2b} + t_{rec}^{b2b} + t_{rr}^{b2b} \quad (8)$$

$$t_{hop}^{c2c} = t_{tra}^{c2c} + t_t^{c2c} + t_{rec}^{c2c} + t_{rr}^{c2c} \quad (9)$$

We expect $t_{tra}^{b2b} = t_{tra}^{c2c}$, $t_{rec}^{b2b} = t_{rec}^{c2c}$ and $t_{rr}^{b2b} = t_{rr}^{c2c}$ since these parameters only depend on the communication circuits at the nodes which are identical. The total latency of the transmission is the sum of the number of hops transferred by the transmission as N_{c2c} and N_{b2b}. In total:

$$t_{total} = N_{c2c} * t_{hop}^{c2c} + N_{b2b} * t_{hop}^{b2b} + t_{dx} - t_{rr} \quad (10)$$

The last two terms $t_{dx} - t_{rr}$ rises from the fact that at the last hop, the signal needs to cross the domain instead of rerouting.

Equation (8), (9), and (10) hold mostly for the 3D-WSI system with a few slight changes. Specifically, in the 3D-WSI systems, serialization and de-serialization are not required,

and therefore t_{tra} and t_{rec} are eliminated. Lower communication latency makes 3D-WSI systems a more attractive integration technology as compared to a 2DI. In addition, the vertical c2c transmission across wafers in 3D-WSI systems is accomplished using TSVs (~10µm length), which exhibits smaller delay than the horizontal c2c links on the wafer (100µm length). Consequently, the transit time t_t is smaller in z-axis while other parameters are the same. However, since the transit time comprises only a small portion of the total delay, we assume that the c2c connections in 3D-WSI systems exhibit an identical delay of 1ns in all directions (*x*, *y*, *z*). The latency of express lanes is determined from the sum of the delay of all TSVs within the express lane. As described in subsection 4.1.1, express lanes exist between any two wafers. Therefore, each express lane is treated as a single hop. The length of an express lane affects only the transit time of that lane. For an express lane across Δz (*i.e.* the distance in vertical direction in terms of wafer layer) wafers, for example from wafer 1 to wafer $1 + \Delta z$, the latency is

$$t_{hop,express}^{c2c} = t_{tra}^{c2c} + t_{t}^{c2c} * \Delta z + t_{rec}^{c2c} + t_{rr}^{c2c} \quad (11)$$

The total latency for a signal transmission in 3D-WSI system is

$$t_{total} = N_{x,y} t_{hop,short}^{c2c} + N_z t_{hop,express}^{c2c} + t_{dx} - t_{rr} \quad (12)$$

where $N_{x,y}$ is the number of on-wafer hops, and N_z is the number of hops in z direction. The vertical transmission is always executed by a single hop, therefore $N_z = 1$ if $\Delta z > 0$ and $N_z = 0$ if $\Delta z = 0$.

The delay values used in the evaluation of the 2DI and 3D-WSI systems, based on the RC values of the respective wires and data from technical references [22,25].

Table 4 Parameters used for the simulation of communication latency

System Type	$t_{tra} + t_{rec}$	t_t	t_{rr}	t_{hop}	t_{dx}
2DI	130ns	1ns (c2c), 5ns (b2b)	20ns	151ns (c2c) 155ns (b2b)	60ns
3D-WSI	0	1ns (on-wafer) Δz *1ns (express lane)	20ns	21ns (on-wafer) 20ns+Δz*1ns (express lane)	40ns

4.1.4 Node-Level Bandwidth of 2DI and 3D-WSI Systems

The bandwidth of the system limits the maximum amount of signal traffic in the system. For neuromorphic systems, a design with a higher bandwidth can support more spiking events from each neuron in a unit time (using the same routing strategy). The biological brain does not suffer from signal congestion in the neurons since the firing frequency (~10Hz) is low, wheras the fanout of neurons (10^3-10^4) is high. Neuromorphic systems should, therefore, operate at a clock frequency compatible with the limit of the node-level (chip-to-chip or die-to-die) bandwidth to avoid congestion at all nodes.

To evaluate the bandwidth limitation due to chip-level system integration, we assume that the on-chip bandwidth is always sufficient, at least before congestion occurs within the chip-to-chip interconnect. Similarly, we assume that the limiting bandwidth in the 2DI system is the board-to-board bandwidth, while the chip-to-chip bandwidth (on the same board) is always sufficient, at least before congestion occurs within the board-to-board channels.

As described in subsection 3.5, the connectivity (aggregate data rate per unit area) of the 3D-WSI system is significantly higher than the connectivity of the 2DI system. It is therefore expected that the bandwidth limitation will exhibit increased importance in 2DI systems. Such

limitation can be relieved by using more than 12 SerDes on each board, ideally by a multiple of 12, but with a larger cost in power and area consumption.

4.1.5 Communication Power Consumption of 2DI and 3D-WSI Systems

The communication power in the 2DI system is composed of the on-board communication power (dissipated in the on-board interconnect and FPGA logic), and the board-to-board power (dissipated in the backplane SerDes circuits). For the 16-chip board example, the board will dissipate approximately 4.7W in the on-board communication and FPGA [11]. In addition, if a 12-SerDes configuration (described in subsection 4.1.1) is used, the system will consume additional 0.56W/SerDes from the board-to-board communication. The communication power of the 2DI system can be estimated given the scale of the system as shown in the Table 5.

As discussed in subsection 3.4.3, the energy per bit of the 3D-WSI system is 0.2pJ/bit for die-to-die communication. An estimation of the power consumption in the FPI of the 3D-WSI system is based on a 4x4x1 region of the 3D-WSI system, which has a total of 48 FPI links to other boards: 16 (in +*z* direction) + 16 (in -*z* direction) + 16 (in *x* or *y* direction). The equivalent bandwidth in the 2DI case is 28Gbps/SerDes * 12SerDes/board = 336Gbps/board. As a result, each link within the 3D-WSI system is required to satisfy at least 336Gbps/48links = 7Gbps/link. This rate can be achieved by using seven 1Gbps fine pitch interconnect channels.

Table 5 Power consumption estimation of the 2DI systems (12 SerDes per board, fully used) and the data-rate-equivalent 3D-WSI systems.

System Scale	Number of boards in 2DI system	Number of wafers in 3D-WSI system	Communication power consumption of 2DI system (W)	Communication power consumption of data-rate equivalent 3D-WSI system (W)
1-board	1	0.11	4.7	0.041
1% brain	27	4	126.9	2.12
10% brain	266	27	1250	18.4
90% brain	2128	266	10000	155

Assuming that every link is fully utilized, in both 2DI and 3D-WSI systems, the power consumption due to communication, as summarized in Table 5 and depicted in Fig. 25. In this estimation, the communication power consumption within the 3D-WSI system is about 100x lower than within the 2DI system. However, the assumption that the channels are fully utilized is not accurate for neuromorphic systems, since neuromorphic systems often exhibit a sparsely firing pattern [5]. A more accurate comparison of the communication power consumption, which takes the actual firing patterns into consideration, is obtained from the simulation and presented in the following chapter.

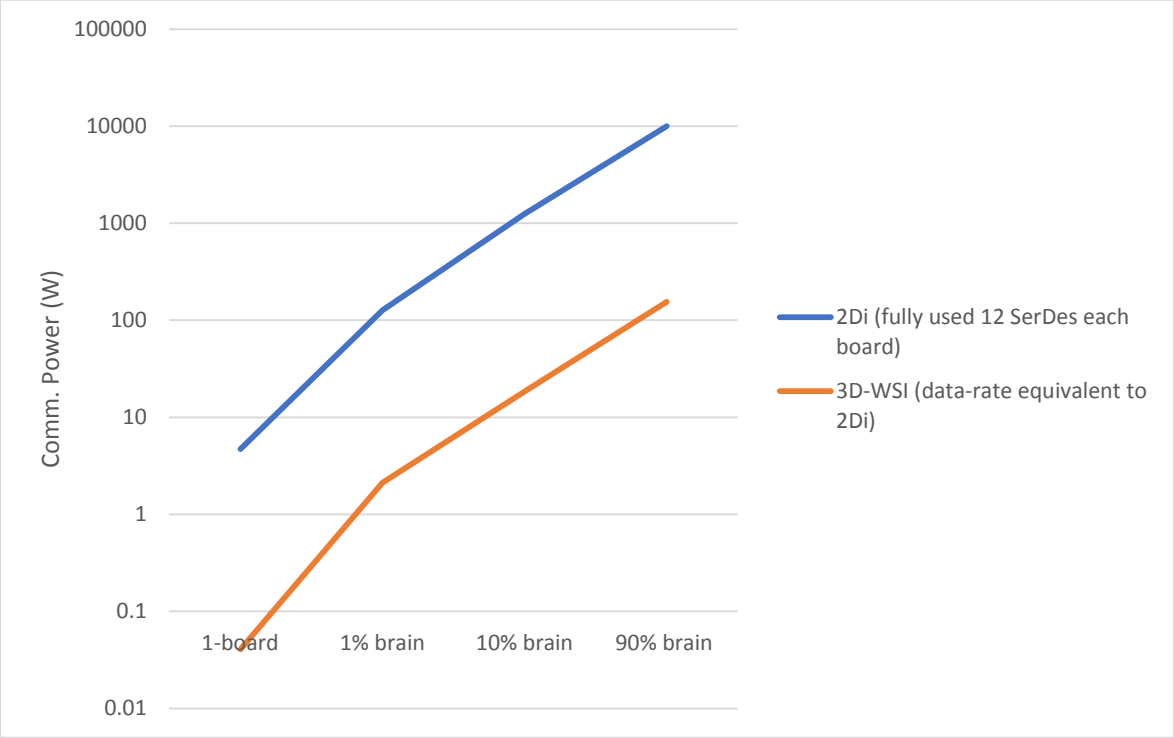


Fig. 25 Estimated communication power consumption

4.2 Neuron-to-Neuron Communication Model

The definition of "connection" between neurons in a neuromorphic system, inspired by the biological brain, and the methodology to integrate these connections within the neuromorphic system, are describe in this section.

4.2.1 Definition of Region and Connection

The neuromorphic system is divided into 266 regions with equal number of neurons and synapses in each region. This configuration draws parallels between the neuromorphic system and the biological brain. The 266 regions in the neuromorphic system correspond to the 266 leaf-regions in the Macaque monkey brain. The probability of a connection between each two regions is different, which is expressed by a proximity connectivity matrix, constructed based on experimental data measured from a Macaque monkey brain [14]. This proximity connectivity is applied to the system evaluation to ensure a statistical resemblance to the connections in a biological brain.

Each neuron in the system has a finite probability to communicate with any other neurons through short-range (local) connections, long-range (global) connection, or a combination of both (local and global) as a mixed type connection. The short-range connections corresponds to the axons in the grey matter of the brain, representing connections within each region. The long-range connections correspond to the axons in the white matter of the brain, representing connections between regions. Fig. 26 is a schematic of such implementation. In our simulations, we adopt the assumption from [22] that 90% of the connections are short-range connections (grey matter) and the other 10% are long-range connections (white matter).

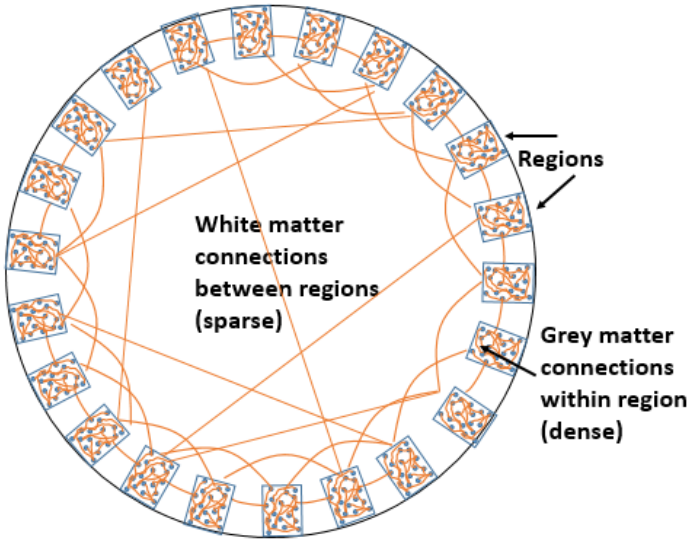


Fig. 26 Neuronal communication model: communication between neurons through the white matter (long-range connections between different regions), and the grey matter (local connections inside the region), from [22].

4.2.2 Gaussian Short-Range Connection

The probability of a short-range connection between two neurons can be modeled by a Gaussian function with respect to spatial distance The Gaussian function peaks at close to zero distance and decays rapidly at greater distance further away as shown in Fig. 27 [26]. The fitted curve (solid line) is used to model the short-range connections.

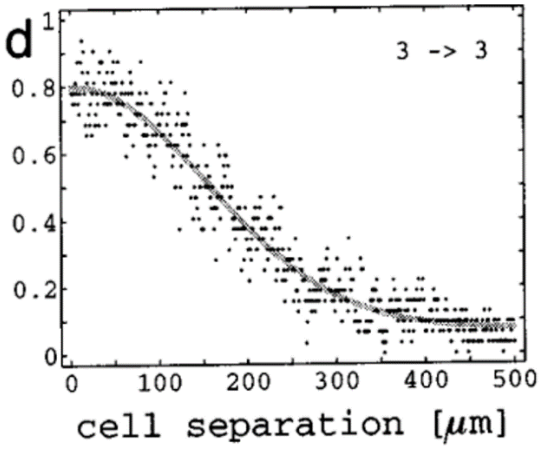
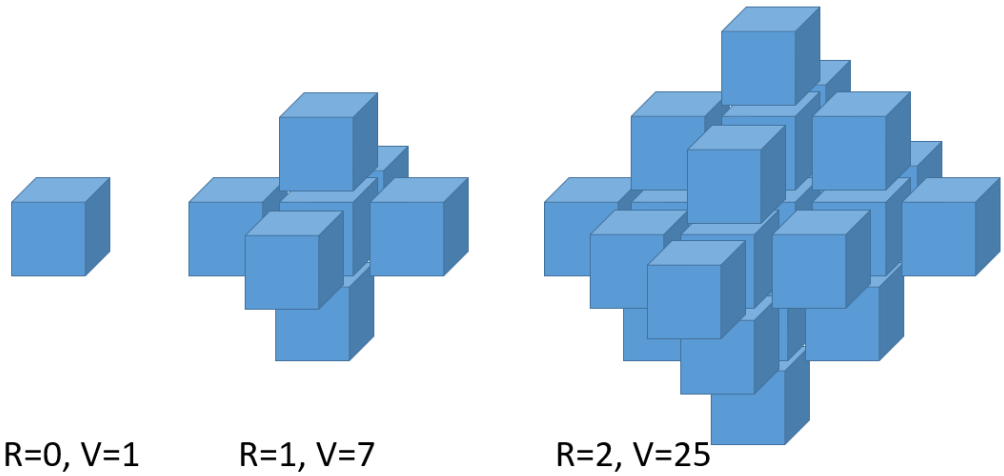


Fig. 27 Gaussian short-range connection: measured connecting probability in the grey matter of a rat cortex. The data dots are fit by a Gaussian function in the solid line (from [26]).

The Gaussian distribution is applied in the L1-space, where the distance between two points (x_1, y_1, z_1) and (x_2, y_2, z_2) is $\Delta = |x_1 - x_2| + |y_1 - y_2| + |z_1 - z_2|$. An L1-space, rather than a Euclidean space, is preferred to accurately represent "axons" that cannot form diagonal connections, using interconnects. The relation between the radius (R) and the volume (V) of an L1-sphere within an L1-space is shown in Fig. 28 is

$$V_{L1}(R) = \frac{4}{3}R^3 + 2R^2 + \frac{8}{3}R + 1 \qquad (13)$$

Fig. 28 The "L1-spheres" defined with radius (R) and volume (V).



Although the probability distribution function (PDF) of a connection between two neurons decreases with respect to the spatial distance according to the Gaussian profile, the

number of candidate neurons to form a connection increases with increased distance. The PDF of a connection with respect to distance does not, therefore, peak at zero, but rather at a finite distance. The PDF of short-range connections with respect to L1-distance is shown in Fig. 29. For each core, block-wise proximity is assumed such that each neuron considers the other 1,023 neurons in the same core as its nearest neighbors. As a result, for each neuron, zero hop is needed to reach any of the other 1,023 neurons within the same core. 1 hop is needed to reach the other 6*1,024 neurons within the neighboring cores in the L1-space. By modeling the short-range connections within a 1,024-neuron core using the Gaussian probability (Fig. 29) applied in an L1-space, the probability distribution of the short-range connection (P_{hop}), in terms of the hop of a neuron core is obtained, as plotted in Fig. 30. From the evaluation based on Fig. 30, 53.1% of the short-range connections are within the core and 43.4% terminate at the nearest neighbor within the L1-space. Less than 0.1% of all the short-range connections connect to cores that are 4 hops or further. Because $V_{L1}(4) = 63$, is much smaller than the total number of cores (1,024) on a single chip, it can be assumed that all of the short-range connections are contained within the chip. These short-range connections do not, therefore, require additional chip-to-chip interconnect resources.

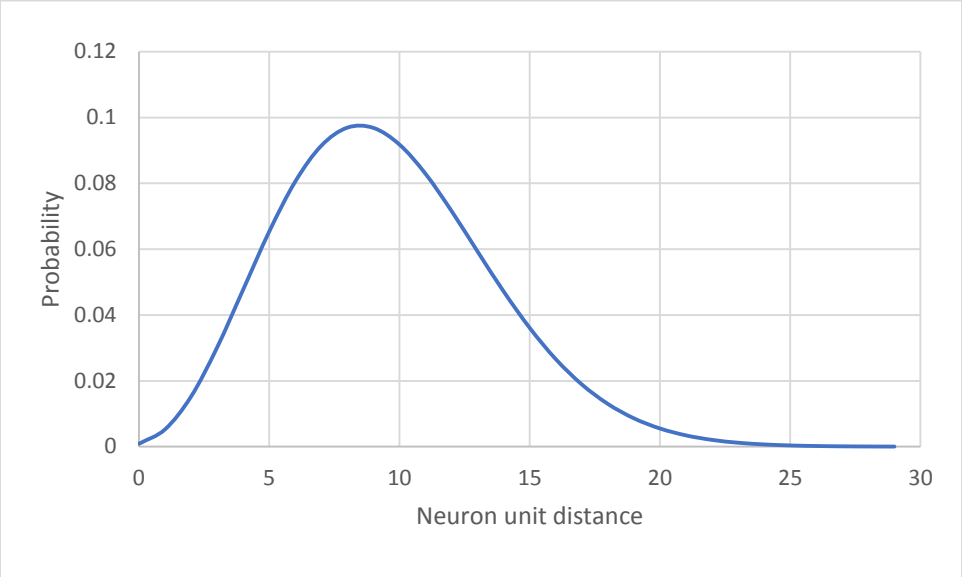


Fig. 29 The probability distribution of the local connection with respect to the neuron unit distance in the "L1-sphere". The probability increases due to the increasing number of possible connections and decreases due to the drop of the Gaussian connection probability shown in Fig. 27.

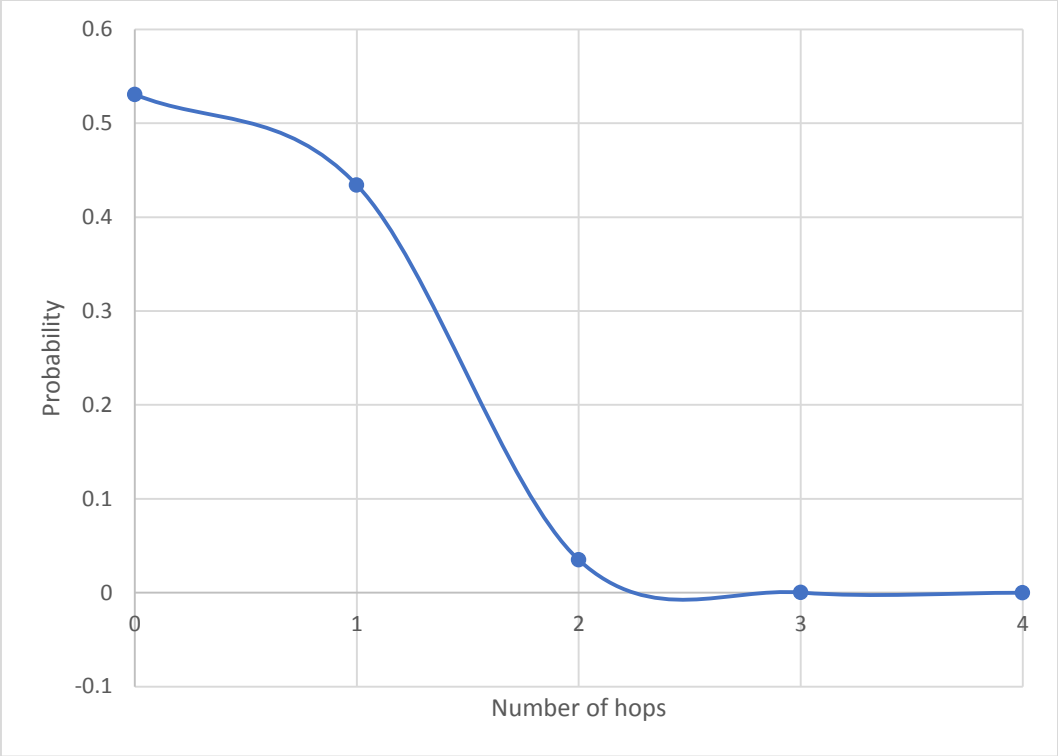


Fig. 30 The probability distribution of the local connections (P_{hop}), in terms of the hop of neuron core (a.k.a. 1,024 neurons per hop).

4.2.3 Long-Range Connection with Biological Connectivity

While the short-range connections are clustered within each region, the long-range connections (white matter) are distributed. The long-range connections are between the regions of the system. The connectivity diagram of the brain of a Macaque monkey [14] is shown in Fig. 31. Quantitatively, the proximity connectivity matrix contains the probability of any unidirectional global connection between two regions. In this simulation, only 266 leaf-regions a total of the 383 regions are used. The colormap of the proximity connectivity matrix is shown in Fig. 32.

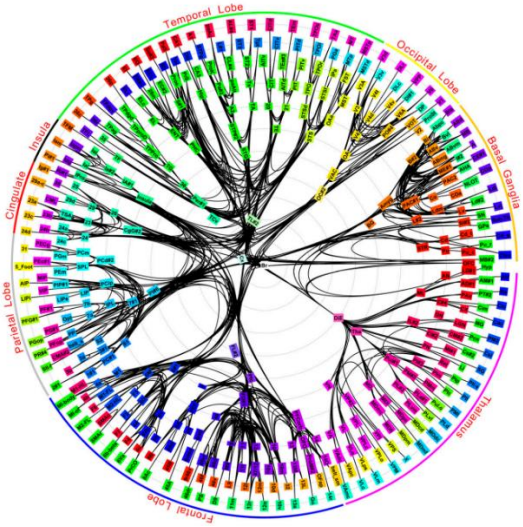


Fig. 31 Connectivity diagram of the Macaque monkey brain (from [14]).

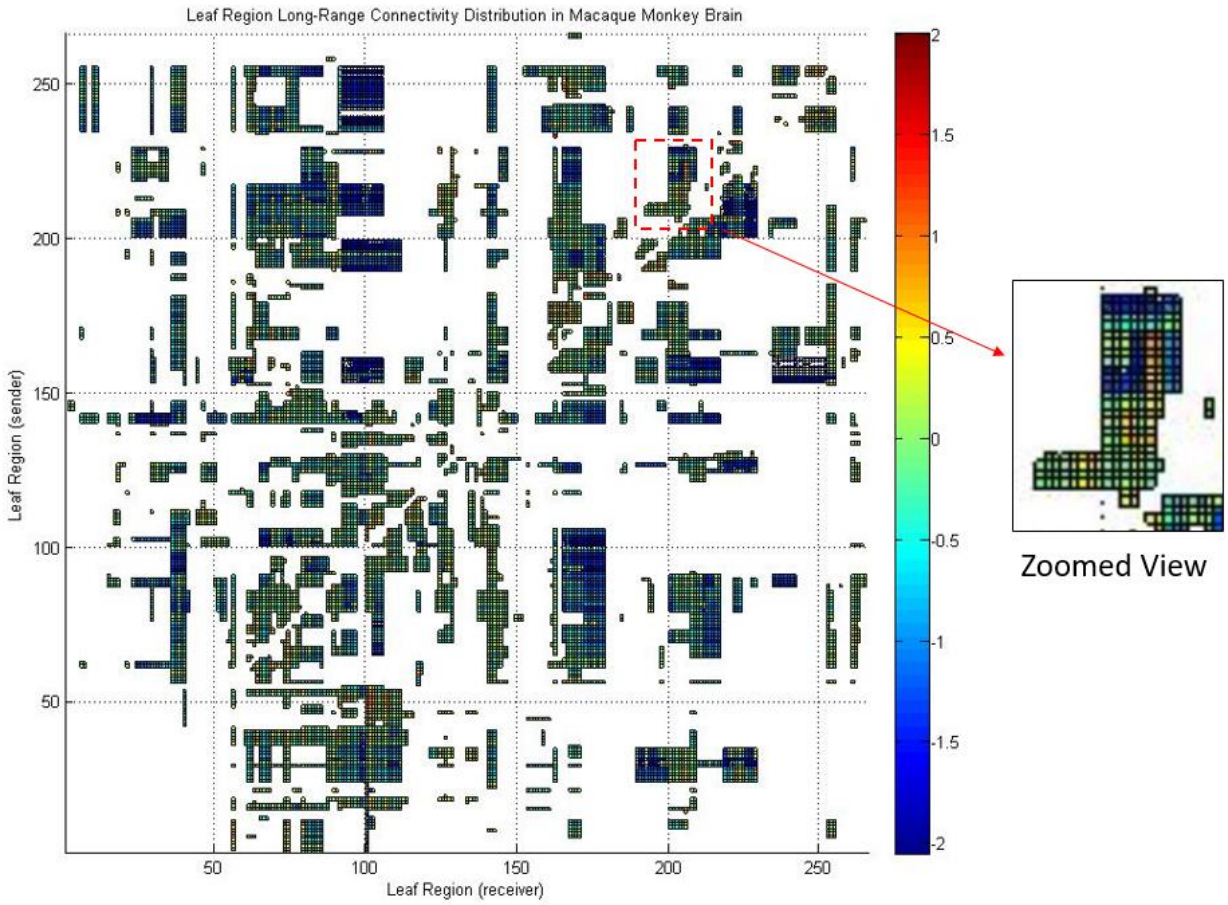


*Fig. 32 The proximity connectivity matrix of a Macaque monkey brain in 1% * log 10 (e.g., "2" in dark red corresponds to 10^2%). Absence of color indicates no direct connection. The inset shows a zoomed view. Generated using data from [14].*

From [14], the proximity connectivity matrix is not symmetrical due to the unidirectional nature of connections. The matrix is normalized such that the probability of sending out spikes by each leaf-region sums up to 100%. The probability of receiving spikes sums up to various numbers exhibiting a maximum of 473.4% (*i.e.* the number of spikes received by this region is ~4.73X of the spikes the region sends out) and a non-zero minimum of 1.1%. This range confirms the asymmetric nature of this proximity connection matrix. The distribution of message sent and received by each region is shown in Fig. 33.

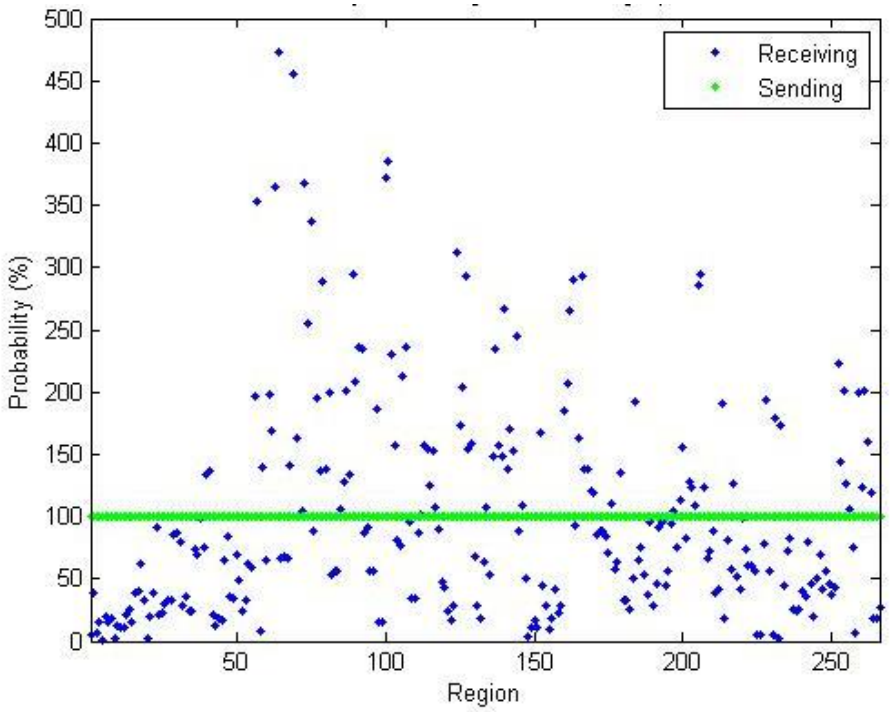
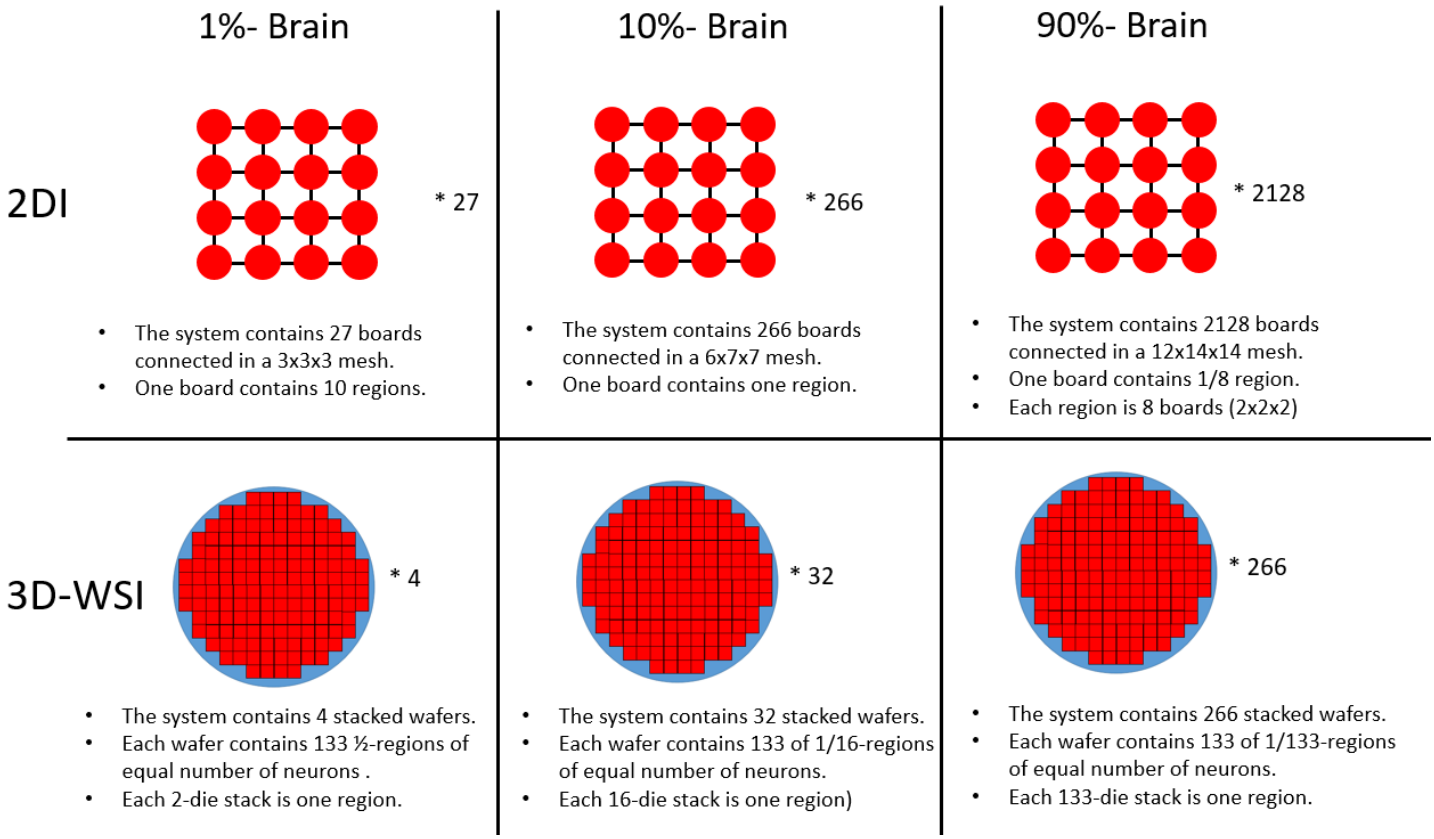


Fig. 33 The sum of the receiving and sending probabilities with respect to the regions. The probabilities of sending is normalized to 100% for all regions.

4.2.4 Node Placement

For the two different integration technologies and the three brain scales listed in Table 3, there is a total of six systems that are evaluated by the simulation. In each system, a "region" is a group of nodes. and therefore, we have designated slots for each region, as described in the Fig.



34. In each case, there are 266 equal-sized slots to put the 266 regions.

Fig. 34 The available slots for the placement of nodes (a chip in 2DI system or a die in 3D-WSI system) in the physical systems

4.3 Overall Model

The overall model, as described in this chapter, is summarized in this subsection. When a neuron receives spikes, it computes the membrane potential (analogous to the soma within the biological neuron) using the embedded spike-timing-dependent-plasticity (STDP) rule, and sends

the output to all of its synapses. Each synapse has a designated destination neuron that it is connected to (address within on-chip memory), and when activated, generates the bit packet including spike, timing, and destination address. The packet is then passed to the router of the core to be delivered to relevant destination as illustrated in the Fig. 35.

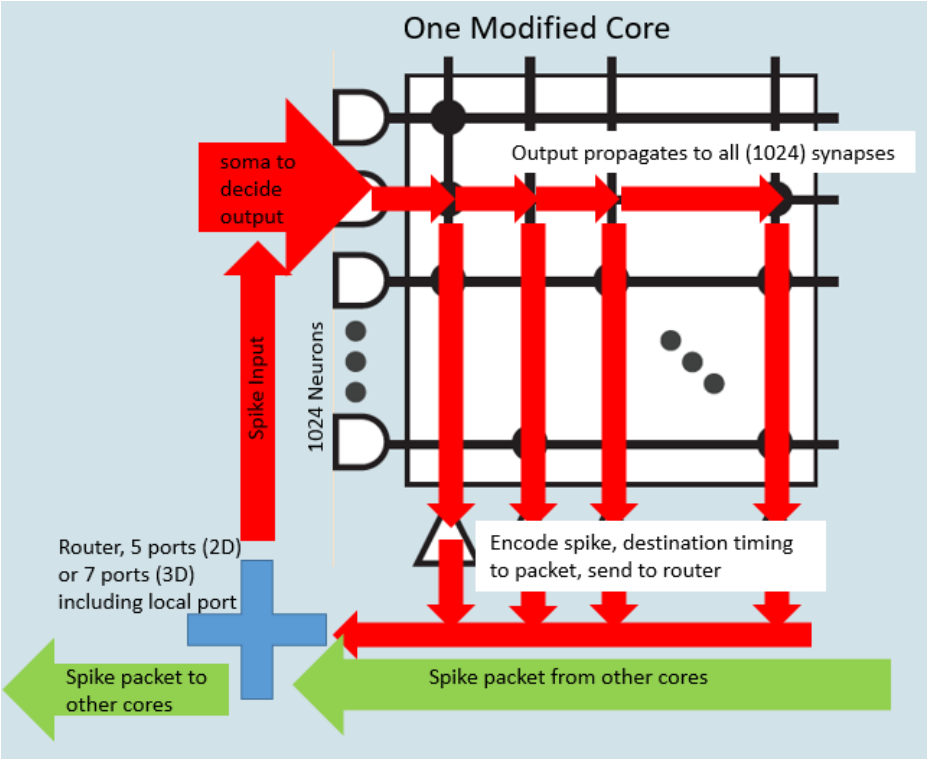


Fig. 35 Schematic for signal transmission in the core of the chips. (adapted from [4])

Since the probability of 4 hops or further connections is below 0.1% (from subsection 4.2.2), the local connections are forced to be within the nodes themselves such that these connections do not occupy additional bandwidth in the chip-to-chip or die-to-die interconnect. The other 10% of the long-range connections does, however, travel across node boundaries depending on the arrangement of the regions within the system, as shown described in Fig. 36.

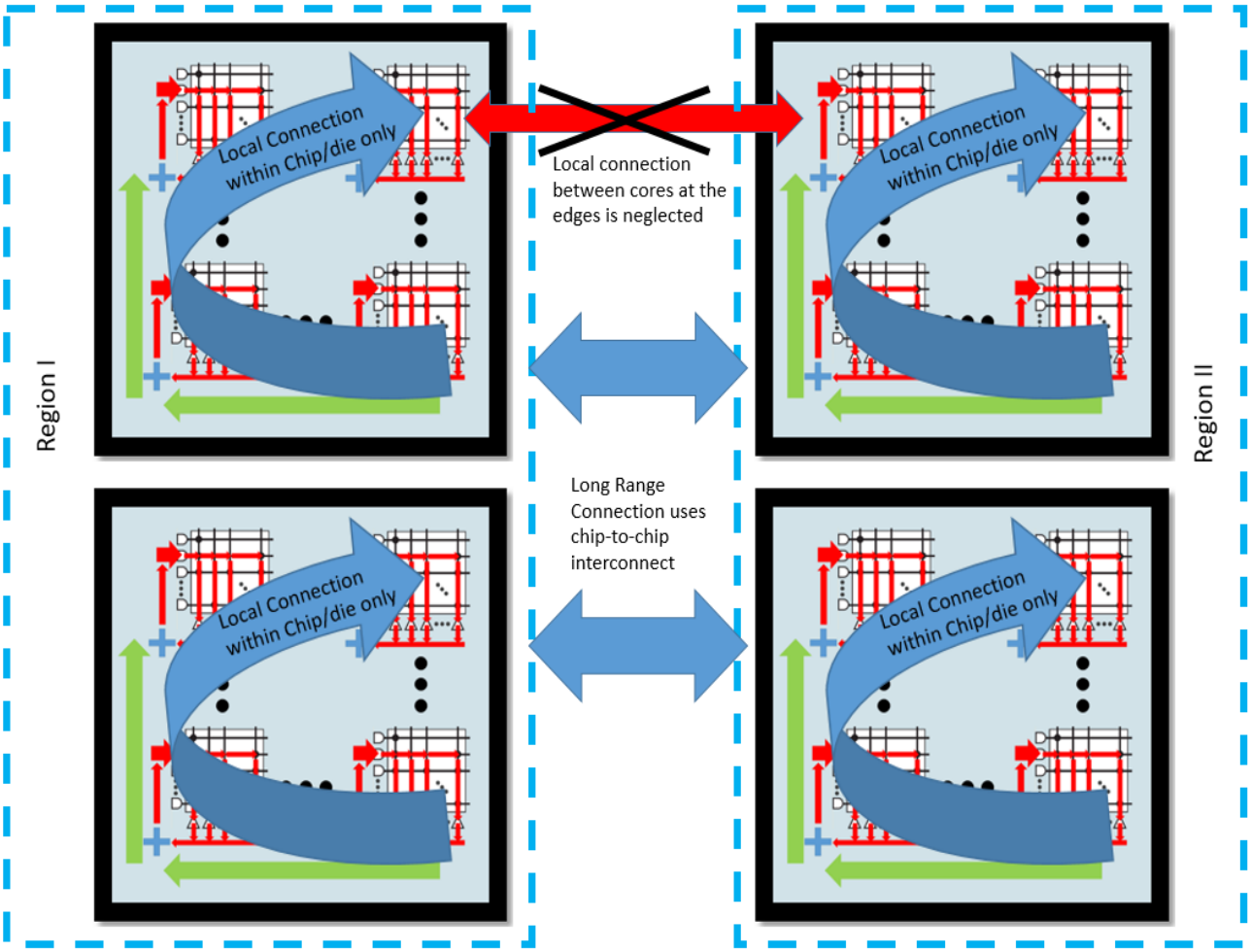


Fig. 36 The short-range and long-range connections of the neuromorphic system. The short-range (local) connections can travel across the cores, but within the same node only. The long-range (global) connections are between two different nodes, which can be physically far apart.

The system is modeled by two types of graphs, one for short-range connections and one for long-range connections. In the short-range connections graph, each vertex is a core and the edges are the interconnections between cores. The local connection graph is a 2D 16-by-16 mesh (corresponding to the die topology), independent of the integration method. In the long-range connections graph, each vertex is a chip (in 2DI), or a die (in 3D-WSI). The vertices are connected by edges that represent the interconnections between chips, dies, PCBs or wafers. Considering the physical layout from different integration schemes that were introduced in

subsection 4.1.1, the long-range connections graph varies with the integration scheme, as shown in Fig. 37. Results of the simulation are presented in the following chapter.

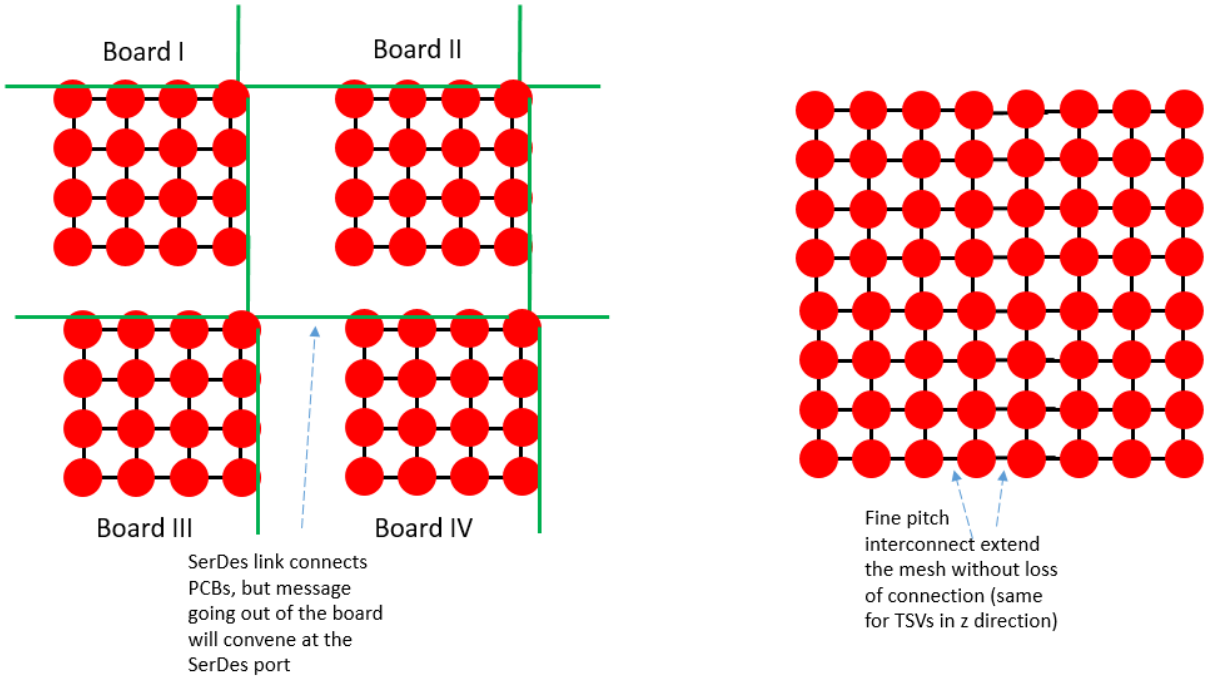


Fig. 37 Graphs for long-range connections: the chip-to-chip communication channels in the 2DI system (left), and the die-to-die communication channels in the 3D-WSI system (right).

5. Results and Discussion

In this chapter, models of the neuromorphic systems, described in the previous chapter, are simulated for six cases. Each case is different in the size and the integration technology, namely, (1) 1%-brain 2DI, (2) 10%-brain 2DI, (3) 90%-brain 2DI, (4) 1%-brain 3D-WSI, (5) 10%-brain 3D-WSI, and (6) 90%-brain 3D-WSI systems. The simulation is performed by self-written code in MATLAB®.

5.1 Communication Latency in 2DI and 3D-WSI Systems

To quantify the communication latency for the simulated systems, the probability density function (PDF) is calculated after simulation. While the simulation calculates the probability of all possible communication events occurred during system operation, the PDF of the latency is obtained by accumulating the amount of connections from each time step (a.k.a. resolution), for example, $2\,\mu s$, occurred during the simulation. The simulated results for the PDFs of the communication latency are shown in Fig. 38 (2DI systems), and in Fig. 39 (3D-WSI systems). The mean and longest-path latency are summarized in Table 6 and plotted in Fig. 40. It is observed that 3D-WSI systems are superior in latency by a factor of 4 to 10, which becomes more significant in larger systems.

The longest-path latency of the systems can be calculated directly from the longest-path of transmission, which can be analytically derived using the given layout. When there is no detour, the longest-path connects the corners of the diagonal of the system. For example, in the 1%-brain case of the 2DI system, where 27 boards each populated with 16 chips, are arranged in a 3x3x3 structure, the longest-path latency is

$$T_{max,2Di} = 2L_{board,max}t_{hop,c2c} + 3\left(L_{system} - 1\right)t_{hop,b2b} + t_{dest} - t_{rr} = 1{,}876\text{ns} \qquad (14)$$

Where $L_{system} = 3$ is the length of the side of the 3x3x3 network. $L_{board,max} = 3$ is largest hop distance between the FPGA chip and the neuromorphic chip on the same board.

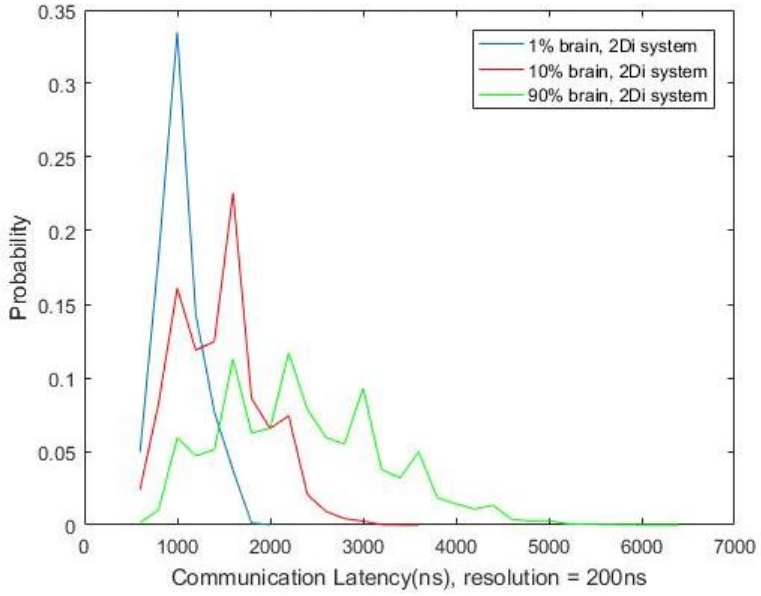


Fig. 38 Simulated probability density function of the communication latency for long-range connections in 2DI systems. The resolution in time is the time step used to accumulate the PDF. For example, the first data point is at x = 200ns, and therefore PDF(x = 200ns) is the probability for the communication event with latency is between 0ns and 200ns. The sawtooth shape is due to the granularity of the time step (resolution). This graph shows that as the size of the system grows, the PDF of the latency spreads out.

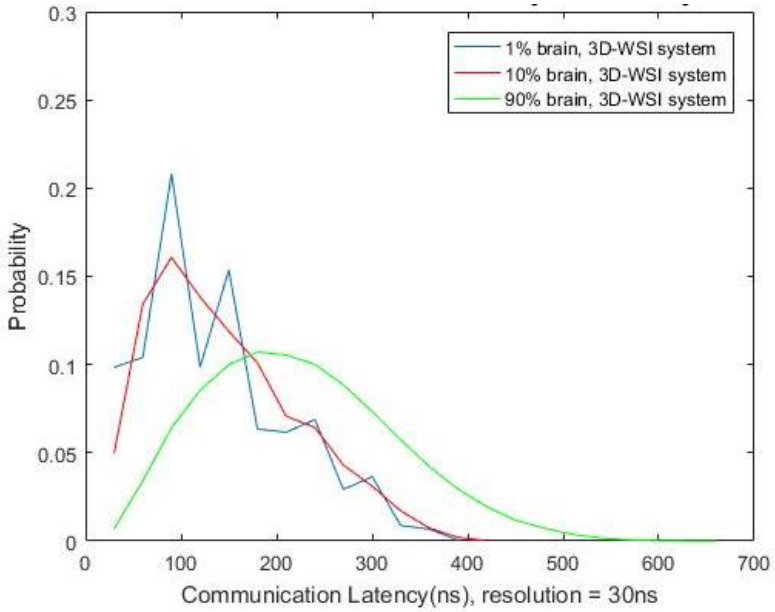


Fig. 39 Simulated PDF of the communication latency for the long-range connections in 3D-WSI systems. This graph shows that as the size of the system grows, the PDF of the latency spreads out.

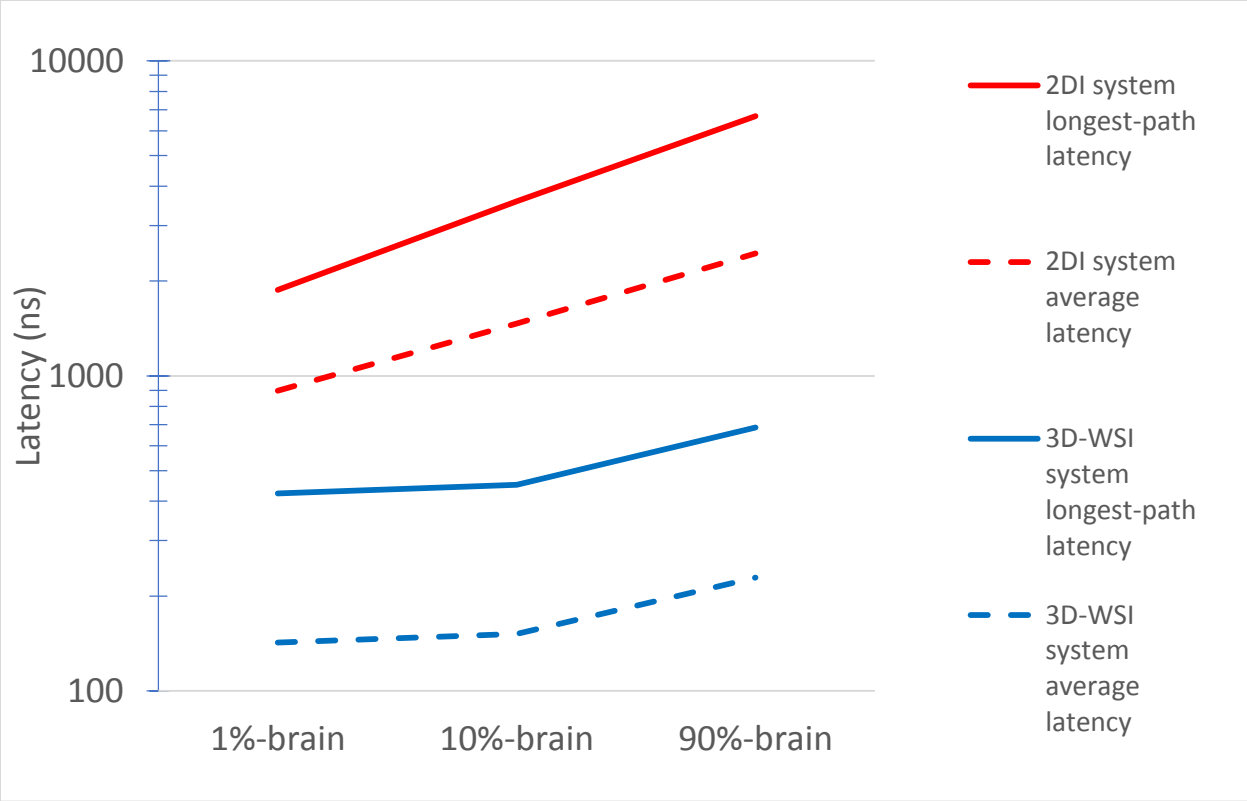


Fig. 40 The simulated average and longest-path latency of all 6 systems, showing that the longest-path and average latency for all system sizes are significantly reduced in 3D-WSI systems.

Table 6 Average (T_{avg}) and longest-path (T_{max}) latency of all simulated systems

Case	$T_{avg,2DI}$ (ns)	$T_{avg,3DWSI}$(ns)	$T_{max,2DI}$(ns)	$T_{max,3DWSI}$(ns)
1%-brain	762.1	114.7	1,876	423
10%-brain	1,360	123.6	3,581	451
90%-brain	2,250	196.1	6,681	685

5.2 Region Allocation Optimization

It can be observed from Fig. 32 that some regions are interconnected more strongly, such characteristics of the connections are exploited to optimize the data traffic within the systems. Intuitively, the strongly connected regions can be placed closer to each other to decrease the latency of the communication events between these regions. For the 266 regions involved, there are roughly 266! unique arrangements of the regions, and optimization by enumeration is therefore computationally unviable. Three methods of region allocation are proposed and evaluated, namely, (1) random allocation, (2) global optimization, and (3) min-cut optimization. For a fair comparison, all optimization methods are evaluated on a 1%-brain 3D-WSI system.

(1) Random allocation simply assigns all 266 regions randomly into the available slots defined in Fig. 34 from subsection 4.2.3. The figure of merit for each allocation is the associated weighted long-range average latency ($\overline{T_{LR}}$):

$$\overline{T_{LR}} = \sum_i P_i T_i \qquad (15)$$

where T_i is the latency of the *i*th long-range connection, and P_i is the probability of this connection. The statistics of $\overline{T_{LR}}$ is collected after 10,000 random allocations. For each iteration, the region allocation changes, and therefore T_i changes while P_i remains the same. The histogram of $\overline{T_{LR}}$ after 10,000 iterations is shown in Fig. 41, with an average of 159.86ns and a standard deviation of 1.63ns.

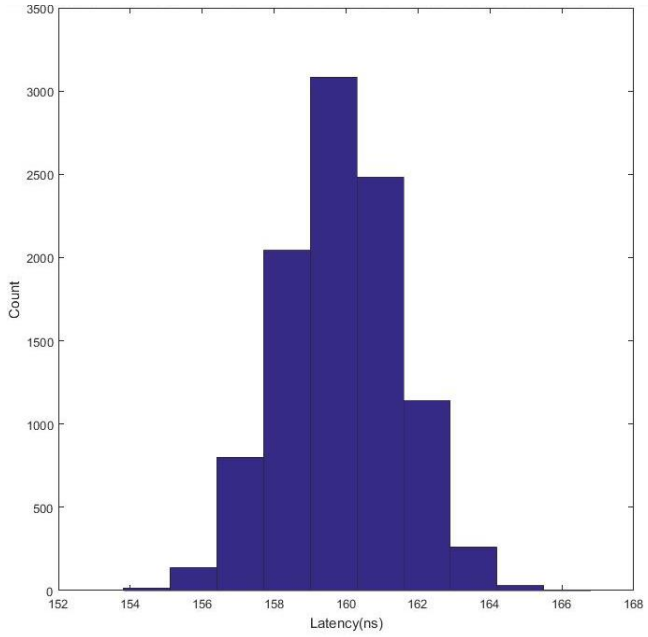


Fig. 41 Statistics of weighted long-range average latency from random allocation: the distribution of the weighted long-range average latency within the 1%-brain, 3D-WSI system with 10,000 trials of random region allocation. $\overline{T_{LR}} = 159.86ns \pm 1.63ns$

(2) Global optimization allocates the regions based on their "popularity", which is the probability of a certain region to send to and receive spikes from all other regions in the system. This probability is summed for each region to determine the popularity if the region. All regions are ranked according to popularity. The most popular region is placed at the geometric center of the system. Consequently, the more popular a region is, the closer it is placed to the center of the system. Simulation result shows that this method is the best among the three presented methods. In this case, $\overline{T_{LR}} = 142.22\text{ns}$ is lower than the $\overline{T_{LR}} = 159.86\text{ns} \pm 1.63\text{ns}$ achieved by random allocation.

(3) Min-cut optimization adopts the min-cut method [29] used in circuit placement. The system is treated as a graph, whose nodes are the regions and the edges are weighted, representing the connectivity between the regions. To initialize the process, all regions are first

randomly assigned to the available slots. The entire system is first cut by a slice into two (almost) equal halves. The system is then optimized to minimize the weighted sum of the edges going across this slice, by exchanging the nodes from one side of the slice to another. The cut-optimization process is iterative, and therefore it is performed until the subsystem contains only two regions. In this case, $\overline{T_{LR}} = 114.7\text{ns}$ is significantly lower than global optimization ($\overline{T_{LR}} = 142.22\text{ns}$), and random allocation ($\overline{T_{LR}} = 159.86\text{ns} \pm 1.63\text{ns}$).

The communication latency PDFs of the min-cut optimization and random allocation are compared in Fig. 42. The PDF curve is shifted significantly to the left when min-cut optimization is applied.

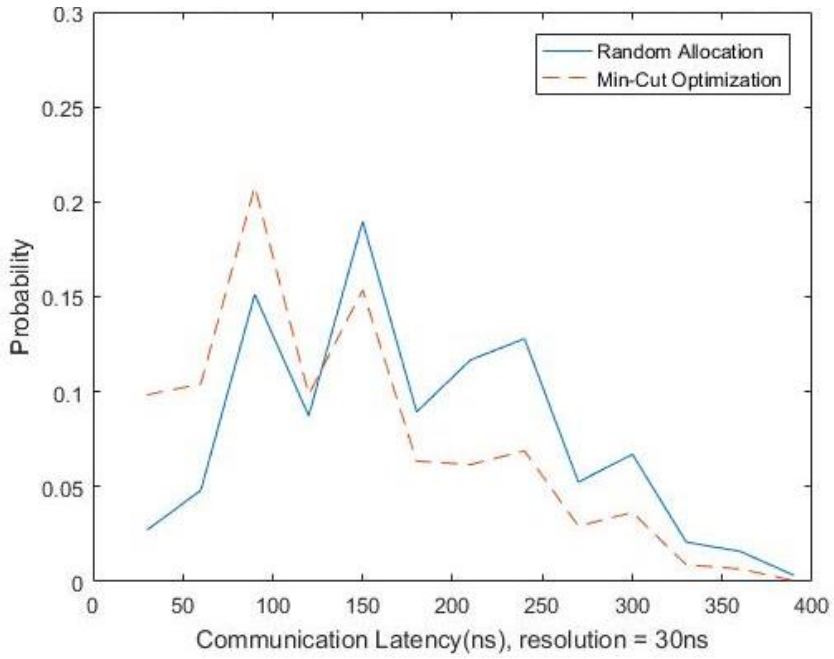


Fig. 42 PDF of the long-range latency from two allocation methods. The min-cut optimization shifts the PDF significantly to the left, and consequently lowers the weighted average of the long-range latency.

In conclusion, the min-cut optimization is the superior method (out of the three evaluated methods) for region allocation. Min-cut optimization is, therefore, used for the simulations, including the communication latency PDF presented in the previous subsections.

5.3 Bandwidth Usage of 2DI and 3D-WSI Systems

The required bandwidth based on the simulation results is derived in this subsection. First, we adapt some important parameters from biological brains:

1. $f_{fire} = 10\text{Hz}$ is the frequency of the clock of the neurons in the human brain.
2. $P_{fire} = 1\%$ is the probability that a neuron would spontaneously send a spike in each time frame ($1/f_{fire}$).
3. $P_{SR} = 90\%$ is the probability that a spike is intended for a short-range connection. Similarly, $P_{LR} = 1 - P_{SR} = 10\%$ is the probability for a long-range connection.

Then, we adapt the following parameters based on the TrueNorth hardware:

1. $n_{synapse} = 1{,}024$ is the number of synapses per neuron. When a neuron sends out a spike, all of its $n_{synapse}$ of synapses receives the spike.
2. $n_{neuron,core} = 1{,}024$ is the number of neurons in each core of the chip.
3. b_{packet} is the size of the spike packet in bits, which increases as the system scales up due to the expansion of the address space. For a single 16-chip board configuration, the packet size is $b_{packet,1board} = 30\text{bits}$, including the chip-level address space which is $\log_2 16 = 4\text{bits}$, and other spike related information (*e.g.* synapse type, axonal delay, and on-chip address) exploiting the remaining 26bits. The size of packet for the simulated cases are discussed in subsection 4.1.2 and summarized in Table 3.

5.3.1 Bandwidth Usage at Biological Frequency

Short-range connections are within the chip, and they exploit only the on-chip interconnect between cores. Thus, the bandwidth used by outgoing spikes for each core within a chip, due to the short-range connections ($BW_{core,SR,out}$) is

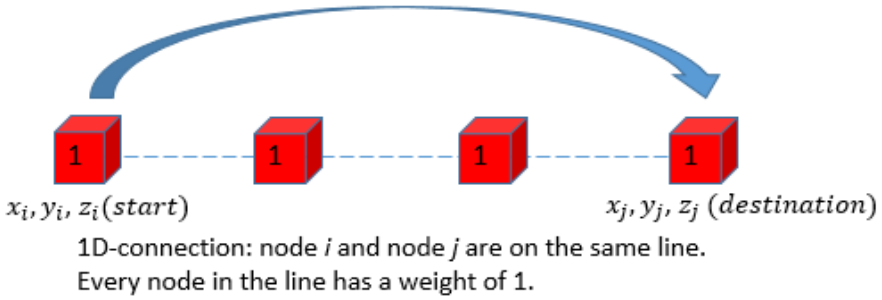
$$BW_{core,SR,out} = \mathrm{f_{fire}} * P_{fire} * n_{synapse} * n_{neuron,core} * P_{SR} * b_{packet} = 94{,}400 * b_{packet}/s \quad (16)$$

The bandwidth used by the incoming spikes is:

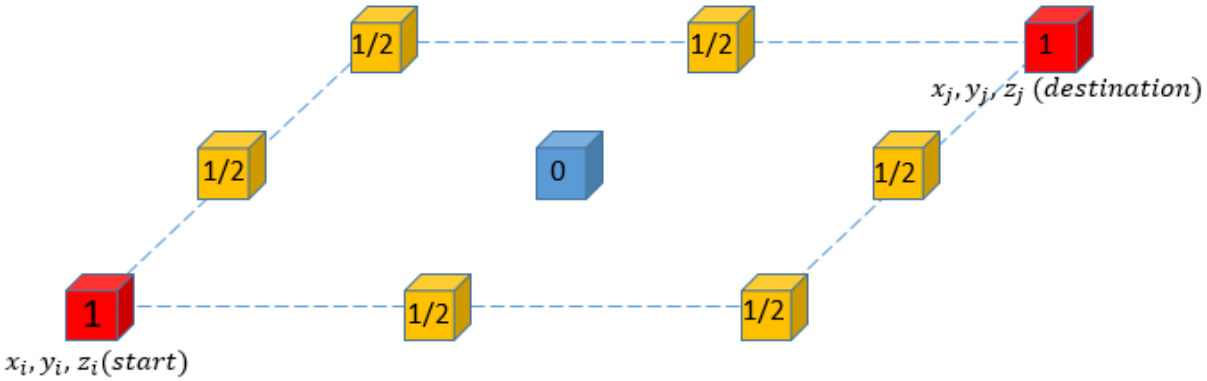
$$BW_{core,SR,in} = BW_{core,SR,out} * \sum_{r=1}^{\infty} [V_{L1}(r) - V_{L1}(r-1)] * P_{hop}(r) \approx 12.3 * BW_{core,SR,out} \quad (17)$$

As previously defined in subsection 4.2.2, $V_{L1}(r)$ is the volume of the L1-sphere with radius *r*, and $P_{hop}(r)$ is the probability of receiving a short-range spike from a core that is *r* hops away. For the 90%-brain case, $BW_{core,SR} = BW_{core,SR,out} + BW_{core,SR,in} \approx 48\mathrm{Mbps}$ at biological frequency.

On the other hand, long-range connections are between the chips/boards/dies, and consequently requires chip-to-chip/board-to-board bandwidth within 2DI systems, or die-to-die bandwidth in 3D-WSI systems. Each chip within 2DI systems is a node with a chip-level coordinate (x_i, y_i, z_i). This node belongs to a certain region denoted by the function $R(x_i, y_i, z_i)$. Similarly, each die in the 3D-WSI system is a node with a die-level coordinate. For every two nodes belonging to two different regions R_1 and R_2, many possible paths exist to connect R_1 and R_2. To optimize the data traffic associate with the communication among all nodes, the path branching function $M(x_i, y_i, z_i, x_j, y_j, z_j, x, y, z)$ is introduced, as shown and explained in Fig. 43. The purpose of such path distribution is to route the data traffic away from the center, such that the nodes at the center of the system, which often suffer from a massive data traffic, are less congested, while the communication still spans the shortest possible path.

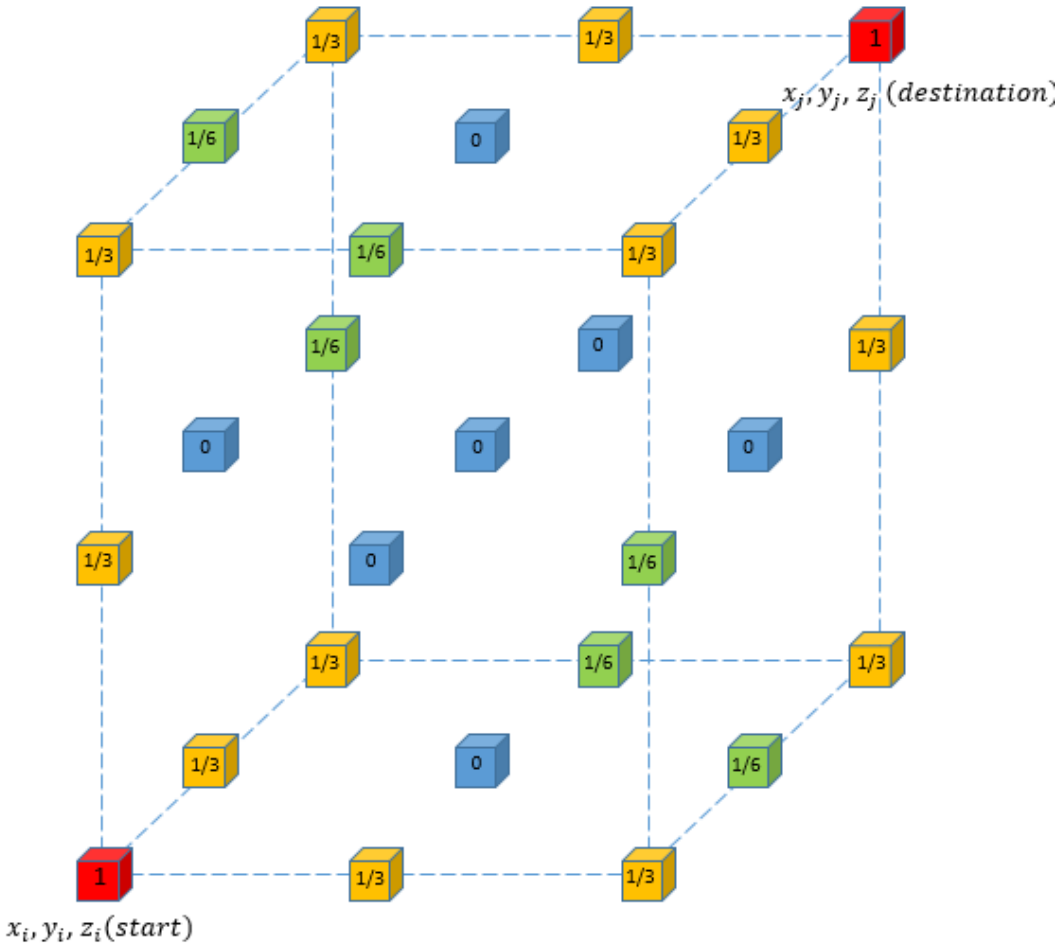


1D-connection: node *i* and node *j* are on the same line.
Every node in the line has a weight of 1.



2D-connection: node *i* and node *j* are on the same plane.
The path is split into 2 branches with equal (1/2) weight.
The center node(s) has zero weight.

Fig. 43 (1) Some examples of the path branching function $M(x_i, y_i, z_i, x_j, y_j, z_j, x, y, z)$. *This figure continues to next page.*



3D-connection: The path is split into 3 branches with equal (1/3) weight.

Fig. 43 (2) Another sample of the path branching function $M(x_i, y_i, z_i, x_j, y_j, z_j, x, y, z)$. For a given spiking event that is from the node i at (x_i, y_i, z_i) to the node j at (x_j, y_j, z_j), this function describes the bandwidth used at any node in the system. In the 1D-connection case in Fig. 42(1), with $x_i < x_j, y_i = y_j, z_i = z_j$, all nodes between and including node i and node j are used by the connection with a probability of 1 because the shortest path must use them. In the 2D-connection case in Fig. 42(1), with $x_i < x_j, y_i < y_j, z_i = z_j$, the edge of the square defined by (x_i, y_i, z_i) and (x_j, y_j, z_j) are used, and each of the two branches has a ½ probability of being used. Similar idea is used for the general 3D-connection case in Fig. 42(2). The values in the cubes indicates the associated probabilities. In all of the presented cases, all the nodes that are not shown has a zero probability since using them requires detour of the signal transmission. Such path branching aims to alleviate the congestion of the nodes in the center, while still using the shortest possible route.

For a node at (x, y, z) within the system, the bandwidth required for all long-range connections during the simulation ($BW_{node,LR}$) is

$$BW_{node,,LR}(x, y, z) = K_{LR} * \sum_{x_1, y_1, z_1} \sum_{x_2, y_2, z_2} P(R_1, R_2) * M(x_1, y_1, z_1, x_2, y_2, z_2, x, y, z) \quad (18)$$

where $P(R_1, R_2)$ is the probability of the spiking event from region $R_1(x_1, y_1, z_1)$ to region $R_2(x_2, y_2, z_2)$, and $M(x_1, y_1, z_1, x_2, y_2, z_2, x, y, z)$ is the path branching function for the node evaluated at (x, y, z) due to the spiking event from (x_1, y_1, z_1) to (x_2, y_2, z_2). Since the probability *P* and the path braching function *M* are normalized, a scaling coefficient (K_{LR}) is used to recover the actual value of the bandwidth. Specifically, for a system with N_{neuron} of total neurons, the scaling coefficient (K_{LR}) is

$$K_{LR} = f_{fire} * P_{fire} * N_{neuron} * n_{synapse} * P_{LR} * b_{packet} \quad (19)$$

Two cross section maps of aggregated bandwidth per board in the 90%-brain 2DI system for *z*=1 and *z*=6 ($z_{max} = 14$) are shown in Fig. 44. The busiest board of the system is at (7,7,6). The center of the system tends to require a higher bandwidth as expected, despite the alleviation of the central nodes by the path branching function (Fig. 43).

Two cross section maps of aggregated bandwidth per die in the 90%-brain 3D-WSI system for *z*=1 and *z*=133 ($z_{max} = 266$) are shown in Fig. 45. The busiest die of the system is at (7,7,133). Similarly, the center of the system tends to require a higher bandwidth as expected. It is also expected that the wafers at the center of the stack exhibit higher data traffic than the wafers at the top and bottom of the stack.

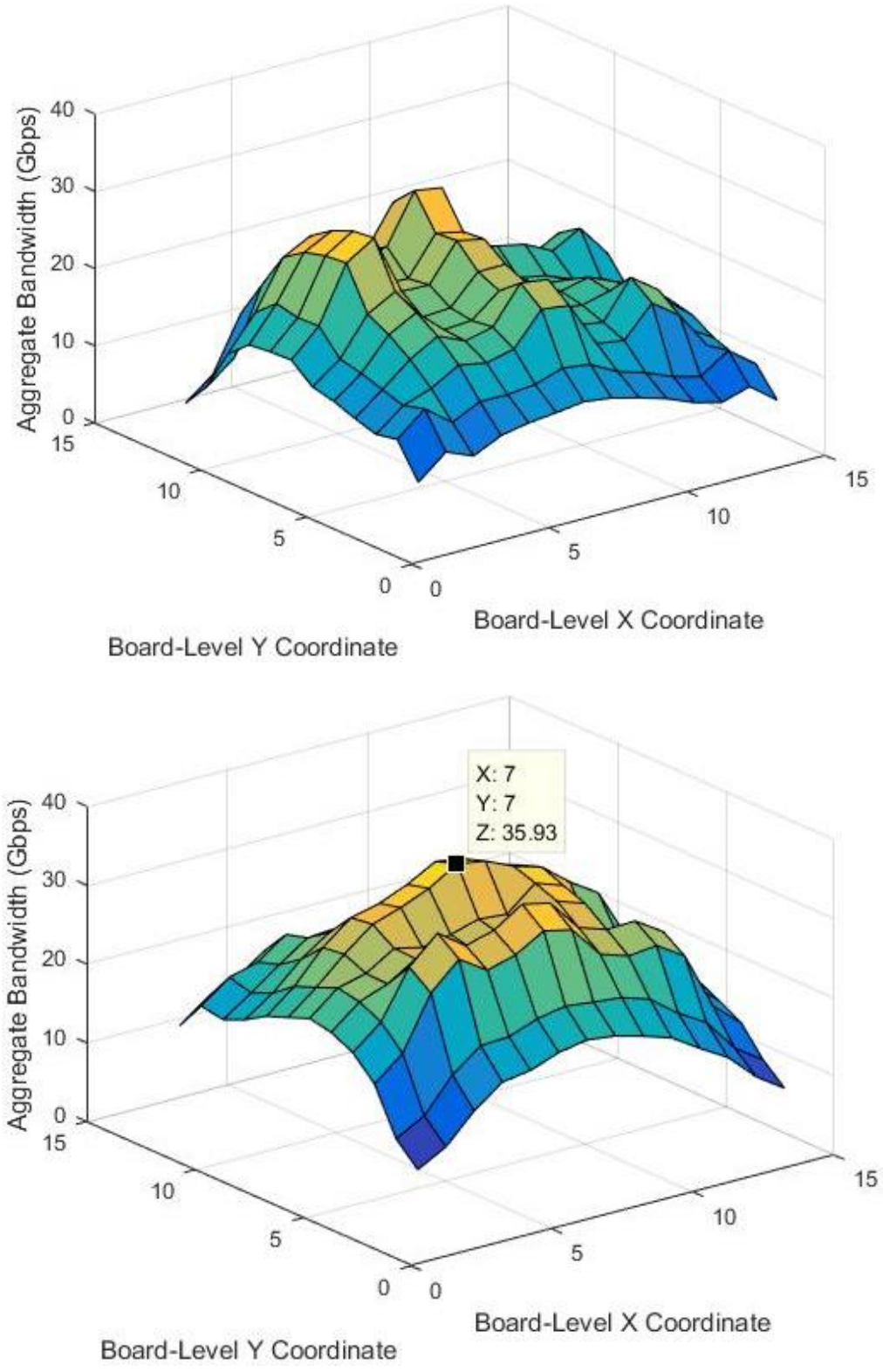


Fig. 44 Cross section maps of aggregated bandwidth required for the boards in the 90%-brain 2DI system. (0<z<15), for z=1 (top plot) and z=8 (bottom plot)

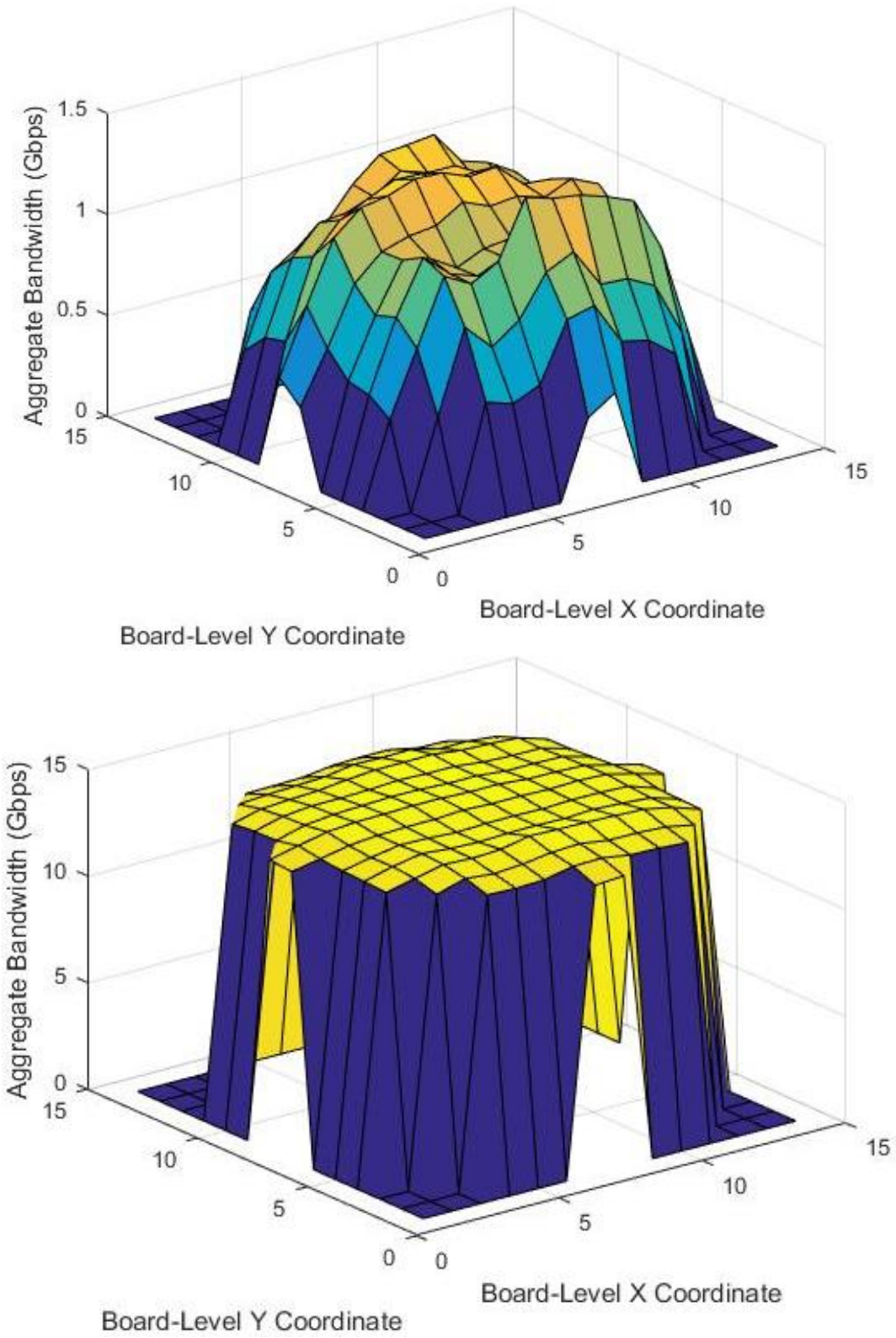


Fig. 45 Cross section maps of aggregated bandwidth required for the dies in the 90%-brain 3D-WSI system (0<z<267), for z=1 (top) and z=133 (bottom)

Since each node within the system has identical configuration for the interconnect (ignoring the effect on the edges), the bottleneck of the bandwidth depends on the busiest node that undertakes the most data traffic, denoted by $BW_{node,LR,max}$. The simulated values for $BW_{node,LR,max}$ are summarized in Table 7. This requirement increases as the system scales out.

The simulation shows that, for the biological model with $f_{fire} = 10\text{Hz}$, the bandwidth can be sufficiently supported by the 12-SerDes-channels-per-board configuration of the 2DI systems (bandwidth $= 12 * 28\text{Gbps} = 336\text{Gbps}$). Similarly, the 3D-WSI systems, with maximum bandwidth per die of $4.8 * 10^7 Gbps$, can also satisfy the bandwidth requirement. Since the required bandwidth only uses a small portion of the largest supported bandwidth, the dedicated one-to-one express lanes occupy only a negligible area of the die.

Table 7 The bandwidth required for the busiest node (a board in 2DI or a die in 3D-WSI).

Case	$BW_{node,LR,max,2Di}$ ($f_{fire} = 10\text{Hz}$)	$BW_{node,LR,max,3DWSI}$ ($f_{fire} = 10\text{Hz}$)	$BW_{node,LR,max,2Di}$ ($f_{fire} = 1{,}000\text{Hz}$)	$BW_{node,LR,max,3DWSI}$ ($f_{fire} = 1{,}000\text{Hz}$)
1% brain	5.93Gbps	1.23Gbps	0.593Tbps	0.123Tbps
10% brain	17.1Gbps	2.70Gbps	1.71Tbps	0.27Tbps
90% brain	35.9Gbps	14.8Gbps	3.59Tbps	1.48Tbps

5.3.2 Accelerated Systems to Enhance Synaptic Operations per Second (SOPS)

Since the neuromorphic system in this study is digital, it is possible to increase the frequency from the biological frequency of 10Hz. However, higher frequency requires more bandwidth and more power from the communication channels. From the literature, the maximum frequency of the TrueNorth chip is 1,000Hz [4], therefor the frequency of the system can be increased from 10Hz to 1,000Hz. By increasing the frequency, the number of spikes sent per second from neurons increases, and the synaptic operations per second also increases. The SOPS

of the system due to long-range connections is the scaling coefficient normalized by the packet bit size:

$$SOPS_{LR} = \frac{K_{LR}}{b_{packet}} \quad (20)$$

Similarly, since $\frac{P_{SR}+P_{LR}}{P_{LR}} = 10$, the SOPS of the system due to all connections is

$$SOPS_{all} = \frac{K_{LR}}{b_{packet}} * \frac{P_{SR}+P_{LR}}{P_{LR}} = 10 * SOPS_{LR} \quad (21)$$

The SOPS for all the systems at $f_{fire} = 10Hz$ are listed in Table 8.

Table 8 SOPS of the simulated systems at f_{fire}=10Hz and f_{fire}=1,000Hz

Case	$SOPS_{all,2Di}$ ($f_{fire} = 10$Hz)	$SOPS_{all,3DWSI}$ ($f_{fire} = 10$Hz)	$SOPS_{all,2Di}$ ($f_{fire} = 1{,}000$Hz)	$SOPS_{all,3DWSI}$ ($f_{fire} = 1{,}000$Hz)
1% brain	$1.12*10^{10}$	$1.39*10^{10}$	$1.12*10^{12}$	$1.39*10^{12}$
10% brain	$1.12*10^{11}$	$1.12*10^{11}$	$1.12*10^{13}$	$1.12*10^{13}$
90% brain	$8.93*10^{11}$	$9.27*10^{11}$	$8.93*10^{13}$	$9.27*10^{13}$

From equation (19) that defines $\frac{K_{LR}}{b_{packet}}$, the SOPS for a given system configuration is proportional to the fire frequency f_{fire}. Therefore, the boosted frequency increases the SOPS of the system, until the maximum $f_{fire} = 1{,}000Hz$ is reached. The corresponding bandwidth requirement and SOPS are summarized, respectively, in Table 7 and Table 8.

In the 90%-brain system, the busiest board within the 2DI system requires 4.54Tbps of aggregated bandwidth, which is about 20 SerDes at each direction ($+x, -x, +y, -y, +z, -z$). On the other hand, the busiest die in the 3D-WSI system requires 33.4Tbps, which is about 100x smaller than its limit.

In conclusion, both 2DI and 3D-WSI systems can be designed to satisfy respective bandwidth requirement from $f_{fire} = 10Hz$ to $f_{fire} = 1{,}000Hz$. The power consumption from these communication events is evaluated in the following subsection.

5.4 Communication Power Consumption 2DI and 3D-WSI Systems

For 2DI systems, the power consumption for communication is divided into two parts: the in-board part and the board-to-board part. The in-board power consumption is approximated by 4.7W per board, including the FPGA logic and intra-board SerDes channels [11]. The board-to-board power consumption depends on the number of board-to-board SerDes channels and their usage. Particularly, the SerDes used for the simulation has two modes: 28Gbps high speed mode (560mW) and 1.25Gbps low speed mode (170mW). The communication power consumption of the 2DI system is

$$P_{2Di} = \sum_{all\ boards} P_{inboard} + P_{SerDes} \quad (22)$$

where $P_{inboard} = 4.7\text{W}$. The board-to-board power consumption P_{SerDes} in each of the six possible directions $(+x, -x, +y, -y, +z, -z)$ is:

$$P_{SerDes} = \begin{cases} 170mW & if\ 0 < BW_{LR} \leq 1.25\text{Gbps} \\ 560mW * ceil\left(\frac{BW_{LR}}{28\text{Gbps}}\right) & if\ mod(BW_{LR}, 28\text{Gbps}) > 1.25\text{Gbps} \\ 560mW * floor\left(\frac{BW_{LR}}{28\text{Gbps}}\right) + 170\text{mW} & otherwise \end{cases} \quad (23)$$

where BW_{LR} is the bandwidth required for the board in one of the six directions. This algorithm minimizes the power consumption of the SerDes by using low speed mode of the SerDes when possible.

For 3D-WSI systems, the communication consumes power per bandwidth used. Particularly, the 2μm pitch interconnect channels introduced in the previous chapter transmit the

spikes at the cost of 0.2pJ/bit. Since no FPGA or any on-wafer logic is used, the communication power consumption of the 3D-WSI system is

$$P_{3D-WSI} = \sum_{all\ dies} P_{FPI} = \sum_{all\ dies} BW_{LR} * 0.2pJ/bit \quad (24)$$

The simulated data for communication power consumption, for all simulated cases, are shown in Fig. 46. The 3D-WSI system exhibits a 100x to 1,000x improvement.

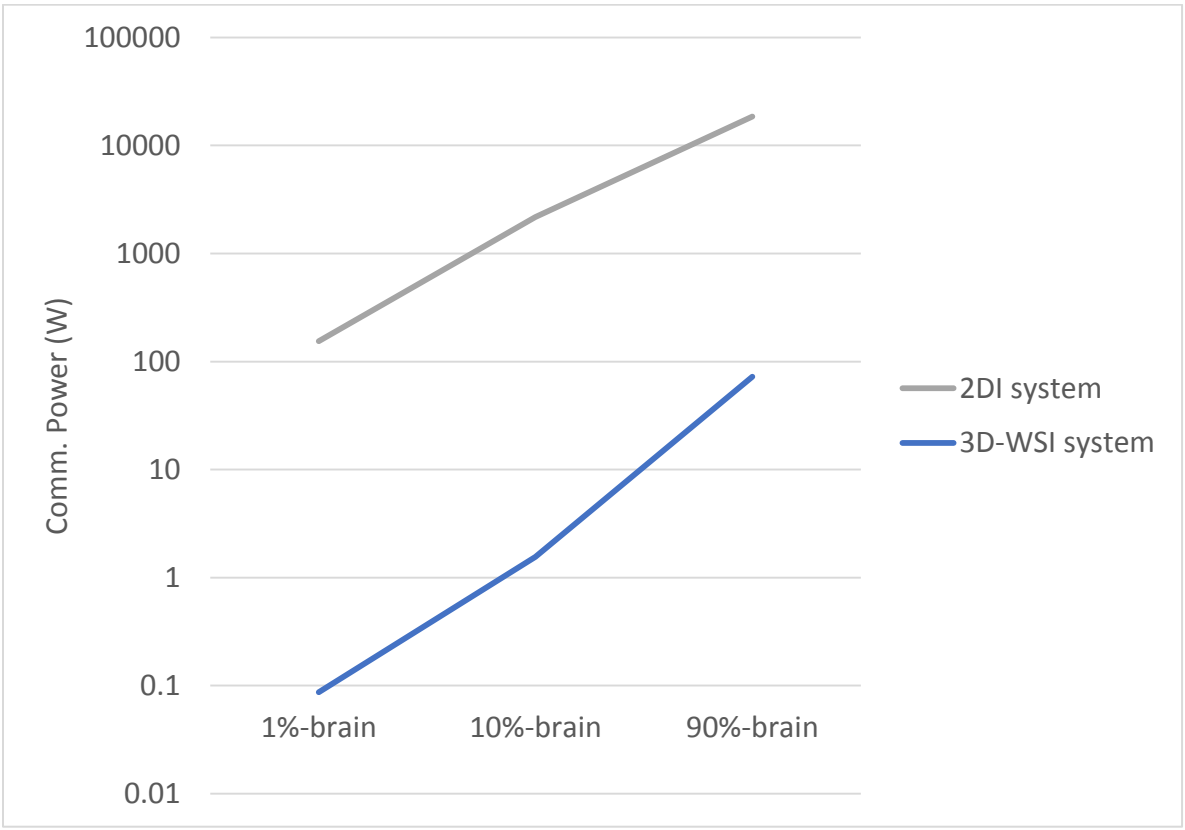


Fig. 46 Communication Power Consumption of all 6 cases from the simulation at biological frequency (f_{fire}=10Hz).

It is noteworthy that, as the system scales out, the power consumption from the routers, which is not included here, may become a significant part of the total power dissipated by the system. All the data traffic passing through a certain node is processed within the routers. Adaptive routers may be applied to alleviate the congested nodes, with the expense of additional power.

6. Summary and Outlook

In this thesis, we developed and simulated models for representative neuromorphic systems scaled out by different integration technologies. Since the realization of human-brain scale neuromorphic systems is still technically challenging, this study provides insight into the possible solutions. The emerging 3D-WSI technology is proposed as a promising approach to rigorous scaling of neuromorphic systems.

In chapter 2, the concept of neuromorphic computing was introduced. We examined the obstacles to realize neuromorphic systems using von Neumann architectures. The performance scaling of the neuromorphic computing system requires a paradigm shift in the hardware architecture, namely, from a centralized von Neumann architecture to a distributed non-von Neumann architecture.

The traditional 2DI integration, and a novel 3D-WSI technologies were introduced in Chapter 3. The chip-level interconnect schemes from both technologies were analyzed. We showed that the fine pitch interconnect, enabled by the 3D-WSI technology, significantly enhances the connectivity (bandwidth per unit area), which is crucial for neuromorphic system integration and performance scaling.

The models to be simulated for both the 2DI and the 3D-WSI systems were discreibed in detail in Chapter 4. The comprehensive models include the physical layout, communication latency, bandwidth, power consumption, etc. The models adapt the experimental neuronal connectivity data from mammals to describe the operation of a neuromorphic system.

Simulation results and discussion were offered in Chapter 5. In many important metrics such as average latency, longest-path latency, and power consumption, the systems based on 3D-WSI are shown to be superior. In addition, methods are developed to optimize the system

performance with respect to the average communication latency. The enhancement in synaptic operations per second is also explored. The comparison between 2DI and 3D-WSI shows that the 3D-WSI system exhibits improved average and longest-path communication latency by 4X-10X, and reduces communication power consumption by 100X-1,000X. Therefore, the overall improvement, defined by the product of these two factors, is 400X-10,000X.

Future work includes designing and fabricating the actual neuromorphic system. A small-scale system, such as a single wafer system or a multi-die stack system using TSVs, is the initial objective. The preliminary systems can further verify the advantages of exploiting novel integration technologies, including 3D-WSI. Afterwards, ultra-large-scale neuromorphic systems can be designed and built.

7. References

1. Mead, Carver. "Neuromorphic electronic systems." *Proceedings of the IEEE* 78.10 (1990): 1629-1636.
2. Azevedo, Frederico AC, et al. "Equal numbers of neuronal and nonneuronal cells make the human brain an isometrically scaled‐up primate brain." *Journal of Comparative Neurology* 513.5 (2009): 532-541.
3. Tang, Yong, et al. "Total regional and global number of synapses in the human brain neocortex." *Synapse* 41.3 (2001): 258-273.
4. Merolla, Paul A., et al. "A million spiking-neuron integrated circuit with a scalable communication network and interface." *Science* 345.6197 (2014): 668-673.
5. Hawkins, Jeff, and Sandra Blakeslee. *On intelligence*. Macmillan, 2007.
6. Iyer, Subramanian S. "Three-dimensional integration: An industry perspective." *MRS Bulletin* 40.03 (2015): 225-232.
7. Benjamin, Ben Varkey, et al. "Neurogrid: A mixed-analog-digital multichip system for large-scale neural simulations." *Proceedings of the IEEE* 102.5 (2014): 699-716.
8. Furber, Steve B., et al. "The spinnaker project." *Proceedings of the IEEE* 102.5 (2014): 652-665.
9. Schemmel, Johannes, et al. "A wafer-scale neuromorphic hardware system for large-scale neural modeling." *Circuits and systems (ISCAS), proceedings of 2010 IEEE international symposium on*. IEEE, 2010.
10. Iyer, Subramanian S. "Heterogeneous integration for performance and scaling." *IEEE Transactions on Components, Packaging and Manufacturing Technology* 6.7 (2016): 973-982.
11. Cassidy, Andrew S., et al. "Real-time scalable cortical computing at 46 giga-synaptic OPS/watt with." *Proceedings of the international conference for high performance computing, networking, storage and analysis*. IEEE Press, 2014.
12. SP9 partners UHEI, UMAN, CNRS-UNIC, TUD and KTH, *Neuromorphic Platform Specification - public version*, report of Human Brain Project (HBP) 2016.
13. Batra, Pooja, et al. "Three-dimensional wafer stacking using Cu TSV integrated with 45 nm high performance SOI-CMOS embedded DRAM technology." *Journal of Low Power Electronics and Applications* 4.2 (2014): 77-89.
14. Modha, Dharmendra S., and Raghavendra Singh. "Network architecture of the long-distance pathways in the macaque brain." *Proceedings of the National Academy of Sciences* 107.30 (2010): 13485-13490.
15. Kelly III, John, *"Computing, cognition and the future of knowing."* IBM white paper, 2015
16. Esser, Steve K., et al. "Backpropagation for energy-efficient neuromorphic computing." *Advances in Neural Information Processing Systems*. 2015.
17. Ananthanarayanan, Rajagopal, et al. "The cat is out of the bag: cortical simulations with 109 neurons, 1013 synapses." High Performance Computing Networking, Storage and Analysis, Proceedings of the Conference on. IEEE, 2009.

18. Esser, Steven K., et al. "Convolutional networks for fast, energy-efficient neuromorphic computing." *Proceedings of the National Academy of Sciences* (2016): 201604850.
19. Kimura, Hiroshi, et al. "A 28 Gb/s 560 mW multi-standard SerDes with single-stage analog front-end and 14-tap decision feedback equalizer in 28 nm CMOS." *IEEE Journal of Solid-State Circuits* 49.12 (2014): 3091-3103.
20. Schemmel, Johannes, Johannes Fieres, and Karlheinz Meier. "Wafer-scale integration of analog neural networks." *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*. IEEE, 2008.
21. Lin, Wei, et al. "Prototype of multi-stacked memory wafers using low-temperature oxide bonding and ultra-fine-dimension copper through-silicon via interconnects." *SOI-3D-Subthreshold Microelectronics Technology Unified Conference (S3S), 2014 IEEE*. IEEE, 2014.
22. Kumar, Arvind, et al. "Toward Human-Scale Brain Computing Using 3D Wafer Scale Integration." *ACM Journal on Emerging Technologies in Computing Systems (JETC)* 13.3 (2017): 45.
23. Jangam, SivaChandra, et al *"Latency, Bandwidth and Power Benefits of the SuperCHIPS Integration Scheme",* Proc. of 67th IEEE Electronic Components and Packaging Technology (ECTC) 2017, Orlando, FL, accepted
24. Hawkins, Jeff. "What the brain tells us about the future of silicon." *Energy Efficient Electronic Systems (E3S), 2015 Fourth Berkeley Symposium on*. IEEE, 2015.
25. Xilinx, *"7 Series FPGAs GTP Transceivers (User Guide)."* December 19, 2016
26. Hellwig, Bernhard. "A quantitative analysis of the local connectivity between pyramidal neurons in layers 2/3 of the rat visual cortex." *Biological cybernetics* 82.2 (2000): 111-121.
27. Kapooht, "Von Neumann architecture scheme", https://en.wikipedia.org/wiki/Von_Neumann_architecture
28. Furtak, Sharon, "Neurons", http://nobaproject.com/modules/neurons
29. Breuer, Melvin A. "A class of min-cut placement algorithms." Proceedings of the 14th Design Automation Conference. IEEE Press, 1977.