Toward Human-Scale Brain Computing Using 3D Wafer Scale Integration

ARVIND KUMAR, IBM Thomas J. Watson Research Center
ZHE WAN, University of California at Los Angeles and IBM Albany Nanotech Center
WINFRIED W. WILCKE, IBM Almaden Research Center
SUBRAMANIAN S. IYER, University of California at Los Angeles

The Von Neumann architecture, defined by strict and hierarchical separation of memory and processor, has been a hallmark of conventional computer design since the 1940s. It is becoming increasingly unsuitable for cognitive applications, which require massive parallel processing of highly interdependent data. Inspired by the brain, we propose a significantly different architecture characterized by a large number of highly interconnected simple processors intertwined with very large amounts of low-latency memory. We contend that this memory-centric architecture can be realized using 3D wafer scale integration for which the technology is nearing readiness, combined with current CMOS device technologies. The natural fault tolerance and lower power requirements of neuromorphic processing make 3D wafer stacking particularly attractive. In order to assess the performance of this architecture, we propose a specific embodiment of a neuronal system using 3D wafer scale integration; formulate a simple model of brain connectivity including short- and long-range connections; and estimate the memory, bandwidth, latency, and power requirements of the system using the connectivity model. We find that 3D wafer scale integration, combined with technologies nearing readiness, offers the potential for scaleup to a primate-scale brain, while further scaleup to a human-scale brain would require significant additional innovations.

CCS Concepts: • **Hardware** → **Die and wafer stacking**; **Neural systems**; *System-level fault tolerance*; • **Computer systems organization** → Neural networks

Additional Key Words and Phrases: Neuromorphic computing

ACM Reference Format:
Arvind Kumar, Zhe Wan, Winfried W. Wilcke, and Subramanian S. Iyer. 2016. Toward human-scale brain computing using 3D wafer scale integration. J. Emerg. Technol. Comput. Syst. 13, 3, Article 45 (April 2017), 21 pages.
DOI: http://dx.doi.org/10.1145/2976742

1. INTRODUCTION

Fueled by the explosion in the Internet of Things and Social Media, the sheer amount of data in the world today is growing at a tremendous pace [Kelly and Hamm 2013]. The bulk of new information being created takes the form of unstructured data – e.g., images, videos, text, news feeds, spatio-temporal trends – and computing systems are increasingly being called upon to do insightful and intelligent analysis very different

This work is supported by the Defense Advanced Research Projects Agency, under grant N66001-15-C-4034.
Authors' addresses: A. Kumar, IBM Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, NY 10598, USA; email: arvkumar@us.ibm.com; Z. Wan and S. S. Iyer, Electrical Engineering Department, Henry Samueli School of Engineering and Applied Science, University of California at Los Angeles, 420 Westwood Plaza, Los Angeles, CA 90095, USA; emails: {z.wan, s.s.iyer}@ucla.edu; W. W. Wilcke, IBM Almaden Research Center, 650 Harry Road, San Jose, CA 95120, USA; email: winfriedwilcke@us.ibm.com.

DOI: http://dx.doi.org/10.1145/2976742

from conventional transactional processing on structured data. Examples of these unstructured computational problems include sensing, learning, and inferring; detecting patterns and anomalies; and predicting and discovering. Along with these new analysis requirements is the hope that programming can be kept at a minimum: a new computing system should learn on its own from its surroundings and correspondingly adapt its model of the world, and communicate with us through natural rather than programming language.

These new increased demands on computing systems may seem extremely challenging in light of the expected saturation of Moore's law scaling [Borkar and Chien 2011]. Indeed, the stunning improvement in computing performance over the last half-century has been largely fueled by technology scaling accompanied by relatively little change in the fundamental computational architecture, as described by Von Neumann and Godfrey [1993]: a central processing unit which retrieves and sequentially carries out instructions on data from a hierarchical memory subsystem. This model has evolved into the current hierarchy in computing systems, which is still suffering from a 'memory wall' between processor and data, mitigated by a relatively small amount of low-latency cache kept near the processor along with large amounts of storage requiring very long retrieval times [Freitas and Wilcke 2008]. This paradigm has worked well for traditional enterprise and high-performance computing applications, but is a poor foundation for unstructured computational problems in which spatial and temporal locality of data is weak [Murphy and Kogge 2007]. Hence, the confluence of the end of technology scaling and the new demands on computing systems can rather be viewed as an exciting opportunity to fundamentally redesign the architecture in a way that is more adapted to the new tasks.

Because the human brain is naturally adept at these types of tasks, it serves as an excellent inspiration about the direction to proceed [Indiveri and Horiuchi 2011]. We may classify today's efforts in three categories. The first category consists of Machine Learning software that runs on a conventional Von Neumann platform, and is usually designed to perform a specific task based on extensive supervised learning. Deep learning has made remarkable progress in the last decade [LeCunn et al. 2015], but is relatively specialized in application and eventually becomes bandwidth- and memory-limited if we want to apply it to a very general problem with many disparate input streams. At the other extreme, many laboratories have demonstrated the early stages of neuromorphic analog devices (see, for example, [Kuzum et al. 2013]) capable of simulating some important aspects of brain functionality [Azghadi et al. 2014]. While such progress is very encouraging, it is important to note that such devices suffer from high variability [Burr et al. 2014] and are thus far from manufacturability, and furthermore will require a complete rethinking of the deterministic programming model that has been relied upon for the last half-century.

Between these two cases lies an intermediate category, in which synaptic states and weights are stored conventionally in a programmable random access memory, but the architecture has been optimized for features of brain-inspired computing such as distributed (de-centralized) processing with large, low-latency memory capacity, and very high communications bandwidth between processors. Examples in this category include IBM's TrueNorth [Merolla et al. 2014] and the SpiNNaker project [Furber et al. 2014]. We propose that a compact and power-efficient path to a massive scaling up of this type of system can be achieved through the use of 3D wafer-scale integration (3D-WSI) [Iyer 2015]. Taking a somewhat different approach from the BrainScaleS project [Schemmel et al. 2012], which uses 3D wafer stacking with mixed-signal (analog and digital) neuromorphic processors, we contend that existing, digital CMOS devices can be used, making early realization of a massively scaled up neuronal system feasible

because the underlying device technology is mature (not even dependent on the latest technology node).

Because for this highly scaled up system we are interested in a system-level view of brain functionality, we will illustrate our ideas by using a cortical algorithm such as the Hierarchical Temporal Memory (HTM) model [Hawkins and Blakeslee 2004]. Nonetheless, many features of the architecture could potentially be used to scale up other neuronal systems based on different approaches. For example, the SpiNNakker project [Furber et al. 2014] is based on a massive network of highly-interconnected parallel processors with a communications infrastructure optimized for delivery of small data packets. A second example is the TrueNorth architecture [Merolla et al. 2014] in which each chip contains 4096 interconnected neurosynaptic cores. Note that both of these approaches are based on spiking neural networks (SNNs) [Maass 1997], in contrast to the HTM model. However, all of these architectures have in common features that are well suited for scaling up using 3D-WSI: a large network of processing cores (currently realized using digital CMOS), a large aggregate memory bandwidth, and message passing between cores supported by a strong communications infrastructure.

Finally, we note that each of these methods is based on the idea of event-driven simulation, meaning that information is sent only when an event such as a neuron spiking (in a SNN) or a cell becoming active (in HTM) occurs. The energy consumption of event communication depends fundamentally on the sparsity of such events, not on the underlying model that triggered the event, and the information is transmitted by a message packet, not a biological spike. Note that if the message to be transmitted also contains numerical data, then representing it as a long series of spikes is not very energy efficient as it will require charging and discharging an electrical wire multiple times to represent the numerical data accurately.

The purpose of this article. is to carry out a feasibility study of 3D-WSI for realizing a neuronal system and to outline both the strong advantages and significant challenges. The main contributions of this article are as follows:

—Presentation of a principal embodiment of a neuronal system using 3D-WSI (Section 2)
—Development of a connectivity model of brain-like function to assess the performance of the system (Section 3)
—Study of the scaling behavior as the neuron count is increased to a level comparable to that of the human brain (Section 4)

We then discuss the routing and fault tolerance (Section 5), and conclude with a discussion of our findings. Table I gives a list of symbols used in this work.

2. 3D WAFER-SCALE INTEGRATION

2.1. Suitability of 3D Wafer-Scale Integration for Neuromorphic Computing

Wafer Scale Integration refers to fabrication of an entire system on a wafer which is not diced into individual dies. Individual fields on a wafer are connected together by a metallization level that stitches across the reticle boundaries. If two such wafers are bonded together, the circuits on the top stratum can be connected to the ones below using Through Stratum Vias (TSVs), which can be made very dense by thinning the top stratum [Lin et al. 2014]. Additional wafers can continue to be thinned and bonded in succession, with each stratum connected to the one below by a network of TSVs. Although 3D Wafer Scale Integration (3D-WSI) offers the potential for a massively parallel, highly interconnected system in a compact form factor, past attempts to use

Table I. List of Symbols

SYMBOL	DEFINITION
m	number of neurons in the domain of a single processor node
M	total number of neurons in the system
b_{syn}	number of bits required to store the address of a synaptic connection and its data
s	average number of synapses per neuron
w	number of input/output wires per processor node
f_{bus}	clock frequency of the parallel data bus
N	total number of processor nodes in the system
l_{sep}	spacing between neurons if arranged in simple cubic lattice
l_{loc}	decay length for connection probability between neurons
q	characteristic length ratio associated with a node
R	total number of regions comprising the vertices of a Small World Network
k	average number of edges per vertex
p	probability of rewiring in a ring lattice
$C(p)$	average cluster coefficient
$L(p)$	average path length coefficient
σ	metric of small-worldiness of a system
α	neuron activity factor
γ	fraction of a neuron's connections that are outside its node
b_{msg}	number of bits in a message, including header and payload
C_{w}	wire capacitance
V_{dd}	supply voltage
G	global coordinate of a neuron
$x.y.z$	local coordinates of a neuron

3D-WSI have resulted in failure for several reasons. From a technological point of view, methods to bond strata with BEOL layers [Skordas et al. 2011] and to achieve submicron overlay tolerance [Lin et al. 2014] have been found only recently. It is projected that 1 μm vias, with a pitch of approximately 2 -2.5 μm with better than 10% overlay tolerance will be possible in a high volume manufacturing environment within three years [Iyer et al. 2015], giving a density of about 200000 vias/mm^2. Moreover, any application of 3D-WSI has two key requirements. First, the yield of the individual dies must be very high. Second, 3D-WSI is mainly suited for low-power applications, since the heat must be removed from the bonded wafer package. Both of these requirements have been in sharp contrast to the tendency of increasing speed and chip complexity characteristic of past technology scaling.

However, as initially pointed out by Mead [1990], 3D-WSI is very well-suited for neuromorphic computing. Rather than trying to build a single, centralized processor, 3D-WSI enables realization of a large network of small processing units which can be interconnected with each other and with low-latency memory, at very high bandwidth. Thousands of such small processors can be realized on a single wafer. Because the processor design is vastly simplified (<4 cores), the expected yield is much higher than for enterprise server chips. Nonetheless, as discussed in greater detail later, fault tolerance and repair techniques are a crucial aspect of the design because some defects in the processing units, their memory units, and the interconnects are inevitable in such a large system. Indeed, the biggest reason why 3D-WSI is so well-suited for a neuromorphic computing system is that many neural algorithms, particularly those that allow connections to be flexibly formed and destroyed [Hawkins and Blakeslee 2004], are remarkably robust to defects (we will show a dramatic example of this), in stark contrast to transactional processing applications in which a single point of failure

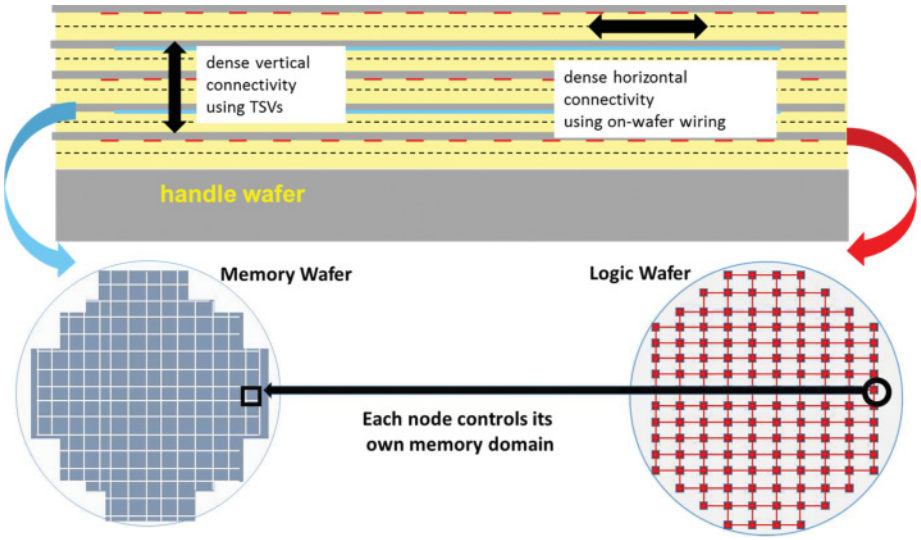

Fig. 1. One possible embodiment of a neuronal system using 3D-WSI, consisting of separate logic and memory wafers bonded together and connected using high density TSVs.

can be crippling. Finally, the overall power density is also greatly reduced, due both to the distribution of processing over the wafer (with simplified processor complexity) and to lower communication power (from much shorter wires).

Before ending this section, we comment on the possibility of using a silicon carrier interposer with packaged chips as an alternate method of scaling up. There are two main issues with this approach: (1) Within the interposer, the chip-to-chip interconnectivity is still limited by the packaging, whose features have scaled very slowly compared to the device features [Iyer 2015], (2) The 3D interconnectivity between interposers is highly limited, as each interposer would need to be mounted on a substrate such as a PCB, and the interposer-to-interposer communication would effectively become a board-to-board connection. While an interposer could be an interesting intermediate step towards full 3D-WSI, it does not allow the ultimate scalability afforded by removing the chip packaging entirely, and hence we choose to focus on 3D-WSI in this work.

2.2. Principal Embodiment

Figure 1 shows conceptually one embodiment of our idea [Kumar et al. 2015]. In this embodiment, a logic wafer is fabricated with a few thousand small processors whose function is to perform the memory-intensive primitive computations required by neuronal simulations. These processors are specialized and designed to accelerate a few highly specific tasks. Some examples of these tasks might include:

—Multiply a vector or a matrix with a constant
—Multiply a matrix with a vector
—Determine whether the overlap of two vectors exceeds some threshold value
—Decode synaptic connection addresses stored in memory from a compressed representation

These examples include operations needed both in traditional machine learning and in machine intelligence algorithms, such as HTM. For example, the first two are

fundamental in back propagation-type machine learning algorithms [LeCunn et al. 2015], while the third is specific to HTM-type algorithms based on sparse distributed representations [Hawkins and Blakeslee 2004]. The last one is useful in a very large system where addresses are stored in a compressed representation, as discussed later.

Each small processor, hereafter referred to as a node, has a sizeable private memory which is directly connected to it, on the memory wafer. Since each node is responsible for storing information about a number of neurons m in its domain, a substantial fraction of its memory is dedicated to storage of all the synaptic connections made by each neuron in its domain. Each connection requires b_{syn} bits for both the connection address and some data about the connection, such as its permanence. Therefore, the minimum memory requirement for the m neurons in the node, if each has an average of s synaptic connections, is mxsxb_{syn} bits. For a neuronal system with M total neurons, the number of address bits required, in an unoptimized representation, is $\log_2 M$. Note that the number of address bits (∼34 for 20 billion neurons estimated in the human cortex [Pakkenberg and Gundersen 1997]) can significantly exceed the number of data bits (∼8), leading to a highly inefficient ratio of address to data bits and a bloated memory requirement. As described in the Appendix, an efficient address scheme [Kumar and Wilcke 2015] can be applied which exploits the observation that the vast majority of synaptic connections in the brain are local to a given neuron. In the example given there for a 64-bit addressing scheme, a highly local address is compressed to 25 bits by using run length encoding for all the leading zeros in the address, which can be optimally arranged due to the locality. That case would correspond to a biological example of 75000 neurons/mm^3, where a neuron would have about 300000 potential connections in a sphere of radius 1 mm, and the worst-case connection at the edge of the sphere, at coordinates (+23, −23, +23), would require 25 (rather than 64) bits. We find on average for this case about 22 address bits are required, which is reasonable considering the 25-bit worst case. Note that this scheme still allows any neuron to connect to any other neuron, which is essential for long-distance, white-matter communication [Wen and Chklovskii 2005], and for a flexible number of synaptic connections per neuron [Hawkins and Ahmad 2015], but keeps the memory requirements tenable. Finally, we assume 8 data bits (a typical number for the permanence in an HTM-type algorithm), so $b_{syn} \sim 30$.

We target an average value for s of 1000, based on Hawkins and Ahmad [2015], which finds that a number in this range would be the minimum necessary for a neuron with distal synapses along a stretch of a dendrite to act effectively as a local coincidence/pattern detector. Eventually, we would like to grow this to an average s of 10000 to be at the higher end of the commonly accepted range of 1000–10000 for synapses per neuron in humans [Worobey et al. 2015], but will use 1000 as a good starting point for the analysis. For $b_{syn} = 30$, we find that 1GB of memory can accommodate about 250,000 neurons.

For the memory wafer, we begin our analysis using DRAM because of its technology maturity. Based on a current estimate of 0.2Gb/mm^2 for DRAM density [TechInsights 2013], a 300mm wafer could contain about 1TB of memory, assuming a utilization factor of 80% for the wafer. Partitioning this into 1GB Sections, we could combine this memory wafer with a logic wafer containing 1000 nodes. Overall this logic-memory wafer pair would have about 250 million neurons, about the number in a small mammal. In order to scale the system up, we could continue to bond multiple logic-memory wafer pairs together. However, since each node occupies only about 1 mm^2, we could reasonably put ∼7000 nodes on a 300 mm wafer (occupying ∼10% of the wafer area), and still leave significant area for the wiring and TSVs. This single logic wafer, with ∼7000 nodes, would then need to be paired with 7 DRAM wafers to have the same 1 GB of memory per processor as before. Via blockage should not be an issue given the high

via density of ∼200000 vias/mm^2. To scale up even more, this stack of 8 wafers could be stacked with perhaps 1–2 other such stacks, but further stacking will likely be very challenging. As discussed later, further scaling will require innovations in the memory density to reduce the number of memory wafers so that the total number of strata is tenable (<30).

In contrast to a chip-to-chip connection, nodes on a wafer can be interconnected using a wide parallel bus of metal wires on the wafer, avoiding the multiplexing to a small number of input-output pins that becomes necessary on a chip for packaging purposes. The total bandwidth of each node is given by wf_{bus}, assuming 1 bit per wire (unidirectional), where w is the number of input-output wires at the processor and f_{bus} is the clock frequency of the data bus, which must be chosen low enough to avoid skew in the parallel bit stream. For $f_{\text{bus}} = 100$MHz and $w = 2000$ (half incoming and half outgoing), the outgoing bandwidth of each node is about 100 Gbps, and the total system bandwidth would be N times the value per node, a very high value for N in the thousands.

In addition to the high bandwidth connections between processors, 3D stacking of the memory on top of the logic processors enables a wide, high-frequency memory bus interface. High-density TSVs can be used to directly connect each processor to its memory domain, alleviating the memory wall between processor and memory. This redesign of the memory hierarchy using a wide memory bus enabled by 3D stacking can lead to significant speedup and energy savings [Woo et al. 2010].

2.3. Stitching and Local/Express Lanes

The nodes on a wafer must be connected by metal lines that have to cross the field boundaries in order for them to communicate with each other. This can be done by adding a final stitching layer between fields. In addition to local connections between adjacent nodes, wafer scale integration allows additional point-to-point connections to be made between distant nodes. For example, each node can also be connected to another node some fixed distance away both in the x-direction and in the y-direction and even in the z-direction. A network of these express connections greatly reduces the number of hops required to send a message from one node to a distant one. Although they are hard-wired, unlike the brain's flexible connections, they can mimic the brain's ability to reduce conduction delays through direct long-range connections, which is a key feature of its remarkable efficiency [Wen and Chklovskii 2005].

2.4. Current Status of 3D Wafer-Scale Integration

We end this Section with a review of state-of-the-art 3D-WSI [Iyer 2015]. The key enabler of 3D-WSI is aggressive thinning of the wafer which allows, through preservation of the TSV aspect ratio, a reduction in TSV feature size and an increase in TSV density. An innovative feature is implementation of TSVs after thinning and bonding, allowing for tighter overlay tolerance and greater inter-strata connectivity [Iyer 2015]. Today, multi-stacks of 4 silicon wafers, thinned to 5 μm with 0.25-1 μm TSV features, have been demonstrated [Lin et al. 2014], as has functional control of memory on one wafer using logic on another [Batra et al. 2014]. Wafers may be fabricated in parallel and then bonded [Lin et al. 2014] or bonded first and then processed sequentially [Batude et al. 2015], with different tradeoffs. For this application, we favor the parallel approach because each wafer can be processed and tested independently, leading to higher yield and robustness. In particular, logic and memory wafers each have very different and highly complex fabrication requirements which would make processing a number of them in series very challenging. Although the sequential approach offers the potential for extremely high interconnect density and stacking with zero alignment

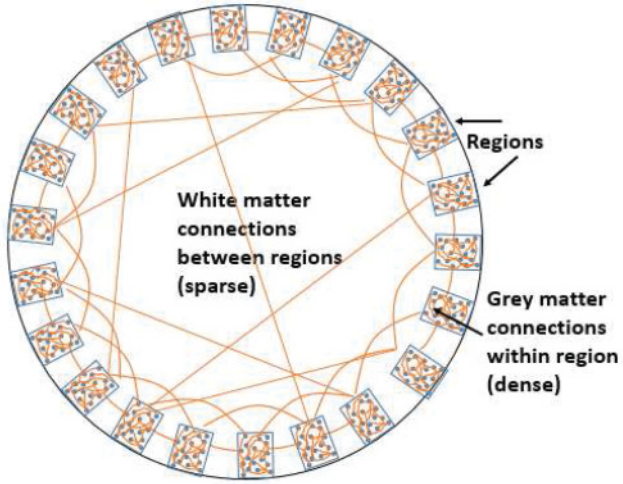

Fig. 2. Conceptual view of connectivity model. Regions are functional units with dense local connectivity ('grey matter'). Regions have sparse global connectivity to each other ('white matter').

overlay, the TSV density offered by the parallel approach ($\sim$200000 vias/mm^2) is more than sufficient for the high-bandwidth requirements discussed later.

3. A SIMPLE CONNECTIVITY MODEL FOR PERFORMANCE EVALUATION

The human brain is characterized by very high neuron count, thousands of synaptic connections per neuron but overall very sparse connectivity, and connectivity at local ('grey matter') and global ('white matter') length scales [Wen et al. 2005]. If our goal is to design a computing system capable of solving unstructured (brain-like) problems, we could reasonably expect that it would need a connectivity comparable to the brain, although of course the actual connectivity will depend on the particular application. In order to assess the performance of a computing system tasked to solve unstructured computational problems, we need a connectivity model which will give us an idea of the networking demands on the system. Figure 2 illustrates a simple model based on physical (experimentally verified) models of brain connectivity at both short and long length scales. It consists of several regions, or functional units, with dense connections between cells within each region, and sparse connections between the regions. We now discuss this model in greater detail.

3.1. Local Connectivity

We first consider the local ('grey matter') connectivity. The connection probability between two neurons generally decreases with the separation distance l_{sep} between their cell bodies. Studies on rat brains [Hellwig 2000; Perin et al. 2011] find a local decay length l_{loc} typically in the 100s of microns, for neuron density $\sim$75000/mm^3, corresponding to $l_{\text{sep}} \sim 25\mu$m if the neurons are evenly spaced in a simple cubic lattice. Using this information, we can calculate how often a neuron connects with other neurons in the same node, nearest neighbor nodes (one hop away), next nearest neighbor nodes (two hops away), and so on.

If the m neurons of a processor are arranged in a simple cubic lattice, the length of one edge of the cube is $m^{1/3}\ l_{\text{sep}}$. Any given neuron will typically connect with other neurons within a distance of a few l_{loc} from the neuron. Since we would like to keep most of the connections within the same node to minimize network traffic, we can define a characteristic "nodal length" ratio $q = m^{1/3}\ l_{\text{sep}}/l_{\text{loc}}$ whose value should be >1.

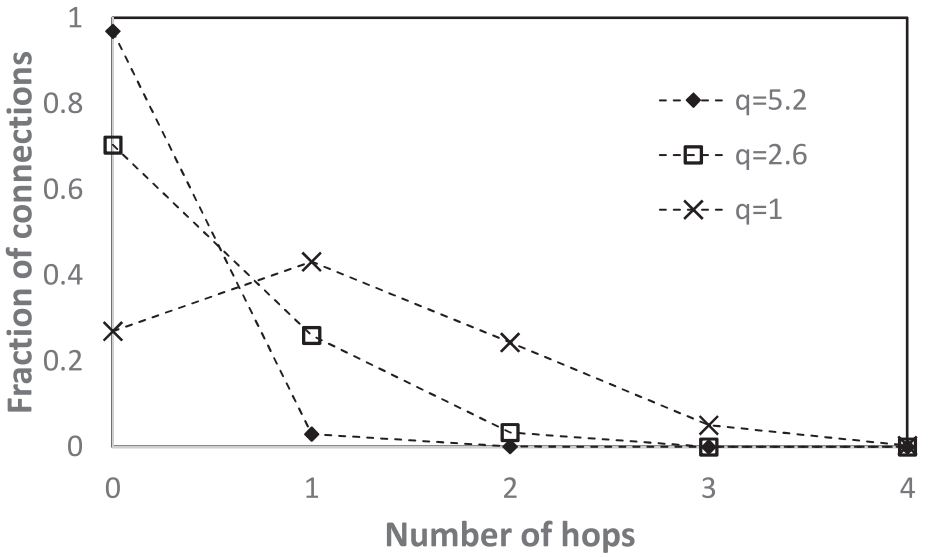

Fig. 3. Fraction of local connections as a function of the number of hops for three values of the parameter q. The case of 0 hops means connections within the same node, 1 means connections to all nearest neighbors, 2 means connections to all next nearest neighbors, and so forth. As q decreases, the fraction of connections in nodes outside the same node increases, requiring more network traffic. The lines are meant only as a guide to the eye, since the number of hops is discrete.

Figure 3 shows the fraction of neuronal connections as a function of the number of hops, using the Gaussian parameterization suggested in Hellwig [2000], for various values of q. The number of hops represents whether the connection is in the same node (i.e., number of hops $= 0$), in a nearest neighbor node (i.e., number of hops $= 1$), and so forth. Note that for $m = 250{,}000$ and $l_{\text{loc}} = 620\mu$m, the longest range found by [Hellwig 2000], $q = 2.6$ and about 73% of the connections are within node and 24% to nearest neighbors, which is a reasonable design point allowing most of the network to be used for the long-range connections to be discussed later.

The curves for different q in Figure 3 can be viewed either as changing l_{loc} for fixed m or as changing m for fixed l_{loc}. However, since q only depends weakly on m as $m^{1/3}$, the design is only very weakly dependent on this initial choice. For example, the three curves represent more than a $100\times$ variation in m for $l_{\text{loc}} = 620\mu$m. This insensitivity is fortunate because, while greatly increasing m may seem beneficial, it also increases the demands on the internal processor, its memory pipe, and on the outgoing bandwidth for synaptic connections that are not local.

Finally, we point out two sources of error in our simplistic model. First, pyramidal neurons are arranged in mini-columns [Buxhoedeven and Cassanova 2002] and preferentially connect to other pyramidal neurons in the same mini-column, so neither neuron arrangement nor connection probability is isotropic. Of course, we could have parameterized each direction by its own l_{loc} and l_{sep}, and our conclusion would be the same provided each l_{loc} is large compared to its l_{sep}. Second, the tails of the connection probability are likely to be underestimated by the Gaussian parameterization. However, in the next section, we will present a global connectivity model with a parameter reflecting the fraction of long distance connections, so any underestimation of the tails can be incorporated by adjusting this parameter upwards. Therefore, neither of these simplifications is expected to have a major effect on our estimates.

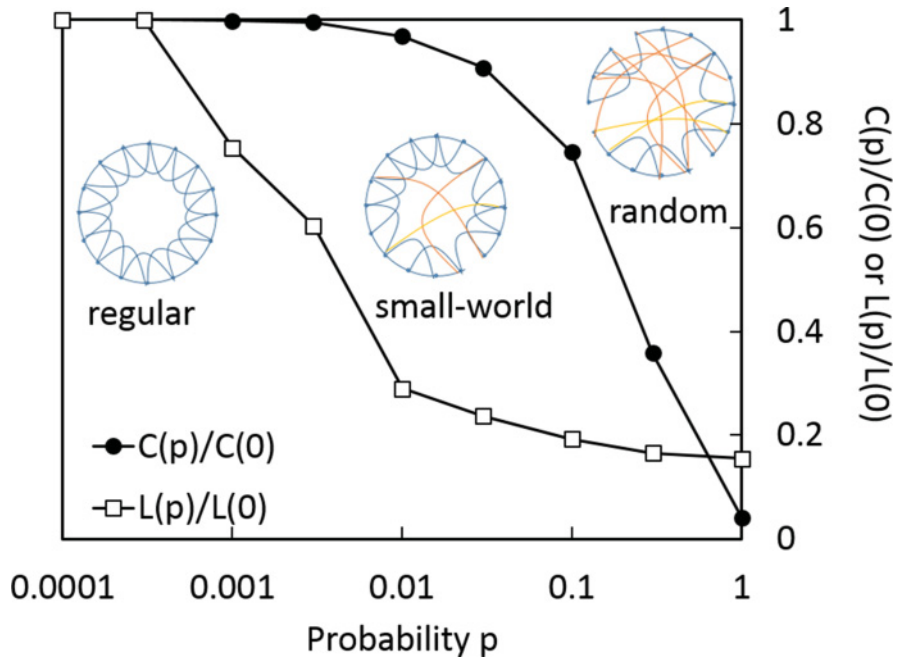

Fig. 4. Cluster *C(p)* and path length *L(p)* coefficients as a function of rewiring probability p, normalized to the regular network case of $p = 0$, using definitions for *C(p)* and *L(p)* from Watts and Strogatz 1998]. Pictures indicate rewiring procedure on a 1D ring lattice to obtain a SWN and random network from a regular one.

3.2. Global Connectivity

The brain may be structurally and functionally divided into many regions which communicate with each other through a network of global ('white matter') connectivity. While mapping the neural pathways that underlie human brain function is a hugely complex undertaking (see, for example, [NIH Brain Initiative website]), there exists a simple yet widely accepted model that can be used to simulate the inter-region connectivity properties of the brain. As summarized in Bassett and Bullmore [2006], numerous studies in cat and macaque brains have found that the functional connectivity exhibits attributes of a Small World Network (SWN), as first proposed in a classic paper [Watts and Strogatz 1998]. Subsequent work using diffusion MRI in humans [Hagmann et al. 2007] has also found global organization in the form of a SWN.

The essential idea of a SWN is illustrated using the concrete example shown in Figure 4. Each vertex of a regular lattice of $R = 512$ vertices, arranged in a one-dimensional ring lattice, is connected to its nearest neighbors by $k = 16$ edges per vertex. (The pictures in the figure are meant only to illustrate the various regimes and have many fewer vertices and edges.) Traveling clockwise, each edge of each vertex is allowed to be rewired to another randomly chosen destination vertex with probability p. For small p, the existence of just a few of these rewired connections greatly diminishes the average path length to traverse the network, but the network retains the high degree of local connectivity ('cliquishness') in the original regular lattice ($p = 0$). This regime of high clustering and short path length, which occurs for a surprisingly wide range of p, is called a SWN, and has been found to describe the behavior of many highly disparate systems [Bassett and Bullmore 2006]. For large p, the network becomes randomly connected, marked by short path length and poor clustering. Figure 4 shows these regimes using the average clustering coefficient *C(p)* and path length *L(p)* (as defined in Watts and Strogatz [1998]) as a function of p.

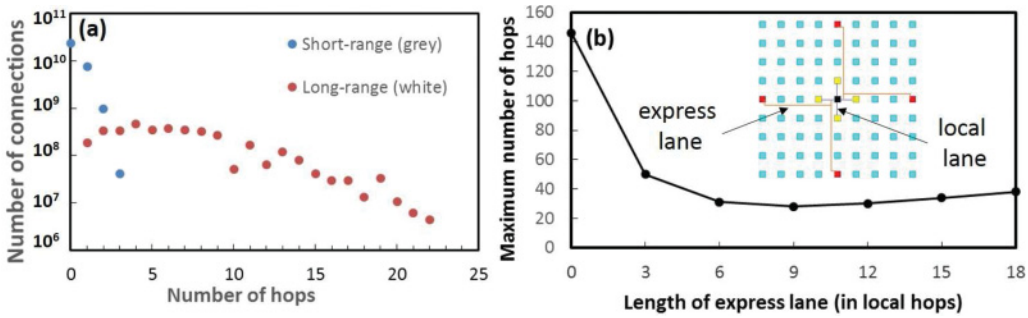

Fig. 5. (a) Distribution of the number of hops, divided into grey- and white-matter connections. For the example shown here, 90% of the connections are chosen to be grey-matter, so the area under the grey-matter curve is 9 times that under the white-matter curve. (b) Maximum number of hops as a function of the length of the express lane (in local hops). The inset shows an illustration of express lanes for one node (similar connections for other nodes are not shown for clarity).

3.3. Application of the Connectivity Model

In order to formulate a reasonable model of the networking demands in our computing system, we must consider both long-range and short-range connectivity. This is biologically plausible since a given neuron might be expected to have a mixture of short-range and long-range connections though proximal and distal arborizations [Ruppin et al. 1993]. However, since the brain seeks to minimize the energy expended on communication, the connectivity is dominated by short-range connections [Hasler and Marr 2013]. We choose to study the behavior of our system treating the proportion of short-range vs. long-range connections as a parameter. While an assumption of $<$10% long-range connections is likely to be typical, we extend our study to 50% long-range connections to study the bandwidth, latency, and power implications in an extreme worst-case scenario.

To apply the global connectivity model, we first group some number of nodes into a region, or functional unit, such that there are a total of R regions in the network. For every edge in a simulated network with R vertices, with k and p chosen to be in the SWN regime as illustrated above, we calculate the total number of hops (local and express) needed to make the connection. By combining this with the local connectivity model in some proportion (here chosen to be 90% short-range (grey matter) and 10% long-range (white matter)), we obtain a distribution such as the one shown in Figure 5(a) for the case of 512 regions, 13824 nodes. While the vast majority of connections belong to grey matter and hence require at most 2 hops, the worst-case network latency is set by the relatively small number of connections requiring $>$20 hops.

The number of required hops in Figure 5(a) benefits greatly from the presence of express lanes, as discussed earlier and illustrated in the inset of Figure 5(b). Figure 5(b) shows that a dramatic reduction in the maximum number of hops, compared to no express lanes (0 on the x-axis) is achievable. This benefit is comparable to that seen in the SWN through the rewiring of a few random links, but is of course more costly since it involves hard-wiring an express channel for every node. While an optimal value for express lane length of $\sim$9 is apparent, it is noteworthy that the large reduction in maximum number of hops has only a weak dependence on this choice.

3.4. Cortical Algorithm

The unified model above tells us of the required connectivity, but understanding the network demands requires an algorithm that will tell us how much and how often

Table II. Scaling Study Cases

Case	Cells	Synapses	Nodes	Nodes per wafer	Logic wafers	DRAM wafers	Regions
1-base	4.5×10^8	4.5×10^{11}	1728	1728	1	2	64
2-primate	3.6×10^9	3.6×10^{12}	13824	6912	2	16	512
3-human	2.9×10^{10}	2.9×10^{13}	110592	6912	16	128	4096

Case	Vertices	Edges per vertex	Probability p	$C(p)$	$L(p)$	σ
1-base	64	4	0.12	0.35	2.99	9.69
2-primate	512	16	0.03	0.64	3.88	14.19
3-human	4096	128	0.00375	0.74	3.00	15.74

data will flow on the network. As an example, we use the Hierarchical Temporal Memory (HTM) algorithm of cortical processing [Hawkins and Blakeslee 2004]. While a full explanation of HTM is beyond the scope of this work, we provide a very brief summary that should help to understand the network demands. During the course of a single HTM iteration, each functional region activates a number of cells based on input from the outside world or other regions and sends a message to each activated cell's connections. While the messages are traveling throughout the network, each node's processor evaluates and updates the state of the synapses of all its affected neurons based on its input. It may create new synapses, or destroy weak ones, which is how it "learns". By the time the next iteration begins, all of the messages must have reached their destinations in order for the network latency to remain effectively "hidden" behind the computation time, and therefore the time of one iteration sets the time scale for data flow on the network. To estimate the time of a single iteration, we have carried out numerous HTM simulations on ARM A9 processors, and found a typical range of 50–200ms per iteration, depending on the number of synapses, which changes as the simulation proceeds. This estimation is also dependent on processor performance (here we have assumed a frequency of 1GHz) on specified tasks. We will later show that the condition for keeping the latency hidden behind the compute cycle is well satisfied for a timestep in the ms range, but note that this condition could be violated if the processors can be greatly accelerated. In that limit, the system will become dominated by the network latency rather than the compute cycle.

4. A SCALING STUDY

This section explores the bandwidth, latency, and power implications of scaling up to human brain levels of neuron count using the 3 cases shown in Table II. Case 2 is an 8x scaleup in total neuron count over Case 1, and Case 3 is an 8x scaleup in total neuron count over Case 2. The average number of synapses per neuron is kept fixed at 1000. Hence, the number of DRAM wafers in Case 2 is 8x that of Case 1, and the number in Case 3 is 8x that of Case 2. In each case, a region is assumed to consist of 27 nodes, or about 7 million neurons, which is in the right range for primate brains [Collins et al. 2010].

The bottom part of Table II shows the SWN parameters used for each case. The values of $C(p)$ and $L(p)$ for the chosen values of p are in the range of biological examples [Bassett and Bullmore 2006]. The last column shows the value of $\sigma = (C(p)/C(1))/(L(p)/L(1))$, which is a metric of the "small-worldiness" of the system [Humphries et al. 2006]. The ratio σ should be well above 1 if the system is in the small-world regime since clustering should be high compared to the random case of $p = 1$ ($C(p) >> C(1)$) while average path length should be comparable ($L(p) \sim L(1)$). For the larger cases (Case 2 and Case 3), we have maintained a high value of σ by scaling up the number of edges k and scaling down the probability p by the same factor as the number of regions R is scaled up.

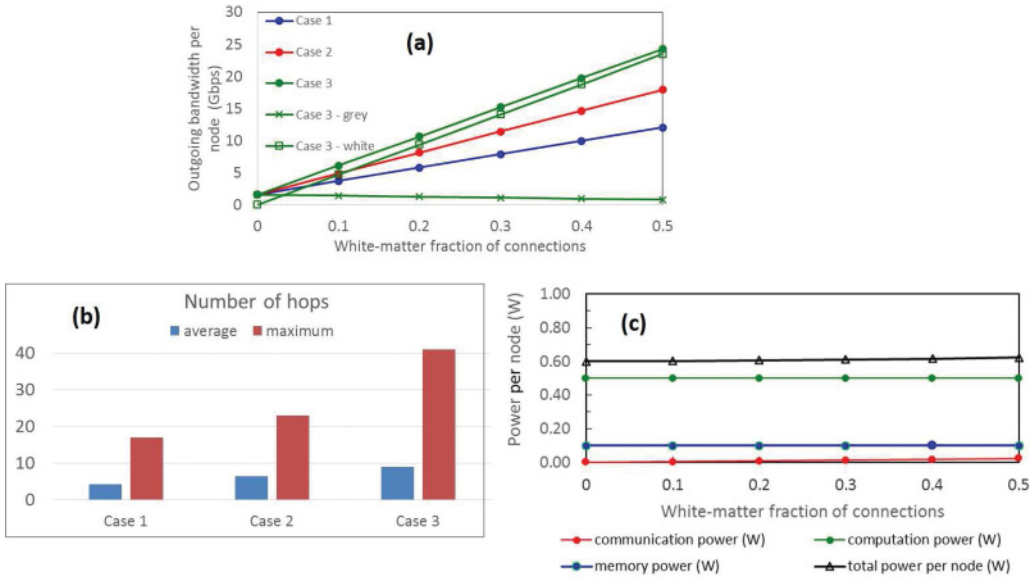

Fig. 6. (a) Outgoing bandwidth per node as a function of the fraction of connections that are long-range (white-matter). (b) Connection-weighted average and maximum number of hops for the 3 cases. (c) Power consumption per node for Case 3 as a function of white-matter fraction and breakdown into logic, memory, and communication power.

4.1. Bandwidth and Latency

The network traffic can be estimated as follows: In a given iteration of the underlying cortical algorithm, a fraction α of all neurons, or αxm neurons per node, become active. We assume that only those neurons whose state has changed since the previous iteration need to send a message to each of their s synaptic connections, leading to (αxmxs) messages to be sent per node in each iteration. If a fraction γ of these connections are outside the node (mainly long-range, but also a few short-range connections), then each node sends γx(αxmxs)xb_{msg} bits onto the network in each iteration. Here b_{msg} is the number of bits contained in a message packet, including the header and payload, and is significantly larger than the b_{syn} bits that was used to store just the address in compressed format and the data bits. Some of those messages (by far the white-matter ones) contain multiple hops and thus need to be rerouted several times during the iteration. Figure 6(a) shows the outgoing bandwidth per node as a function of the fraction of connections that are white-matter (long-range) for the three cases, assuming a iteration of 100ms and an activity factor $\alpha = 0.01$. For Case 3, even for an extreme example of 50% white matter, the 25Gbps is comfortably within our 100 Gbps estimate earlier. Since the curves in Figure 6(a) scale with activity factor α, the outgoing bandwidth could approach the bandwidth capability for high activity factors, as might occur at a few very active nodes, but a very high activity factor is not consistent with biological energy constraints [Lennie 2003]. Also shown in Figure 6(a) is a breakup of the outgoing bandwidth for Case 3 into grey- and white-matter components. Except when the white-matter fraction is very small (below 4%), the network traffic is dominated by the white matter, as it should be for a well-designed system.

As long as the system is not bandwidth-constrained, the latency will be determined by the maximum number of hops, which is set by the long-range connections. Figure 6(b) shows the maximum number of hops as well as the connection-weighted average number of hops for the three cases. We can estimate a time per hop as the

sum of the transit time across the hop and the processing time at each intermediate node where the message has to be rerouted. For the short distances between nodes, the transit time (due to RC) is small compared to the rerouting time. Conservatively estimating the rerouting time as 100ns per hop, the worst case of 40 hops would require on the order of a few μs, so the assumption of latency hiding during the 100 ms iteration is valid, provided the system is not bandwidth-constrained.

4.2. Power Consumption

The total system power can be approximated as the sum of the processor power, memory power, and communication power. Since typical low-end processors consume in the 10's of μW per MHz (see, for example, [ARM website]), a reasonable estimate of the processor power might be 0.5W per node at 1GHz, as we need to add some power for the drivers of the communication lines (explained below). The power consumption of a 1GB DRAM can be estimated from standard datasheets (see, for example, [Micron website]) and from Vogelsang [2010] as approximately 100mW (including refresh), assuming a 1% neuron activity factor which means about 1% of the data will be pulled (random access) per iteration. The communication power can be estimated from $^1/_2 C_w V^2$ based on the capacitance C_w per wire, supply voltage V_{dd}, and the number of connections. For $V_{dd} = 1$ V and typical BEOL wiring capacitance $\sim$2 pF/cm, a local hop of $\sim$0.3 cm would require $\sim$0.3 pJ = 0.3 mW/Gbps, and an express hop of $\sim$3 cm would require $\sim$3 pJ = 3 mW/Gbps. Figure 6(c) shows the total power per node and its breakup into these 3 components as a function of the white-matter fraction for Case 3 (Cases 1 and 2 are similar). Adding these three components together yields a power estimate of about 600 mW per node, or a total of 1 kW for Case 1, 8 kW for Case 2, and 66 kW for Case 3. We have carried out thermal simulations which find, for a stack of 10 wafers, that 1 kW/wafer is sustainable using air cooling and 10 kW/wafer is sustainable using water cooling [Sikka et al. 2015]. Thus, the thermal challenges even for a "human-like" system are manageable.

5. ROUTING AND FAULT TOLERANCE

During each iteration, messages must be passed from one node to another, according to a message passing protocol. This protocol may be deterministic for simplicity, or adaptive to mitigate the congestion of the system. One example of adaptive routing for this system is the algorithm of May et al. [1997], which alleviates traffic congestion by routing a message from one node to another through a randomly selected intermediate node. The routing algorithm must be able to deal with defects and failures, which we now discuss in detail.

Resilience to faults is essential for a wafer-scale system as pre-existing defects during fabrication and real-time failures during operation are inevitable. Fault tolerance can be realized through a combination of redundancy, repair techniques, and algorithms that route around defects, borrowing from a long history of known techniques for mesh networks. However, probably the single most important factor in alleviating vulnerability to failures is the remarkable ability of many neural algorithms to be naturally resilient to faults. We examine each of these in turn.

Redundancy and repair techniques [Arzubi 1973; Robson et al. 2007] play a crucial role in dealing with fabrication defects. For example, because TSVs are freely available due to the high density [Lin et al. 2014], sparse TSV faults can be tested and repaired using redundancy after integration [Chi et al. 2013]. While the processor yield is expected to be very high due to its relative simplicity, extra processors can be added without substantial area penalty. If necessary, power domains can be used to shut off a block of processors containing a fault that could otherwise be fatal to the entire system.

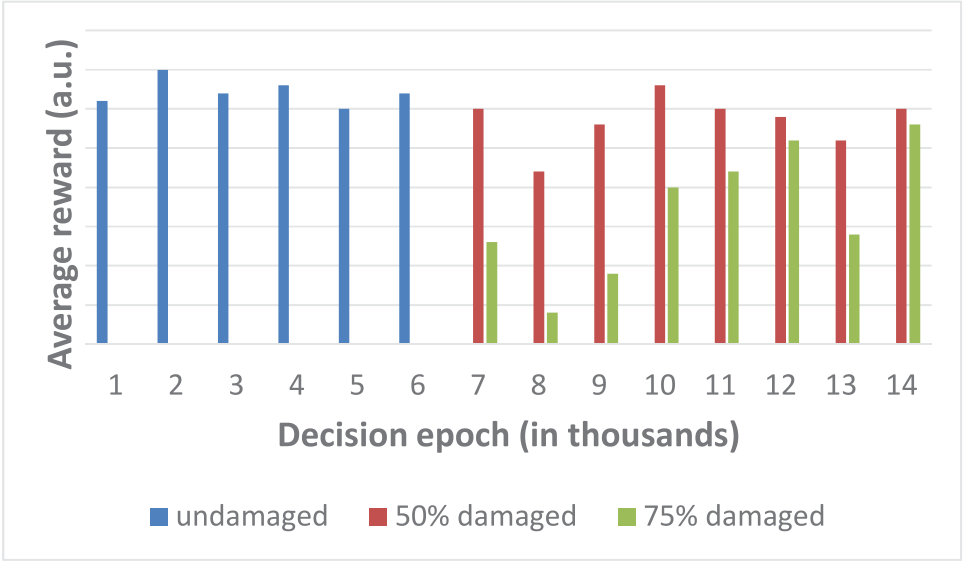

Fig. 7. Effect on performance (average reward) of suddenly disabling 50% or 75% of the columns after 7000 decision epochs in an HTM-type algorithm. Although the performance shows an initial drop, it is able to reconfigure its resources autonomously and recover to nearly its original performance level. Reprinted with permission from J. Marecki.

Unfixable defects on a wafer, such as non-functional nodes, can be addressed using well-known routing algorithms, in some cases extensions of mature 2D routing algorithms such as those used for Network-on-Chip. Because the TSVs of the stacked wafers have RC characteristics comparable to the planar wiring load [Lin et al. 2014], the routing algorithm can treat the TSVs in the vertical dimension the same as the wiring in the x and y directions on the wafer from the graph point of view. Many known techniques for fault avoidance as deadlock-free and livelock-free routing [Boppana and Chalasani 1995] can also be adopted in 3D. Routers for 3D need two more ports due to the addition of +z and –z directions, but the area and power consumption will be in the same scale [Bahmani et al. 2012]. The third dimension is also beneficial in providing additional paths to route around faults.

Due to the robustness of neural algorithms, the presence of a few faults, which may occur during operation, will not substantially affect the performance of the system. Figure 7 is a simulated result showing a dramatic illustration of this robustness. Here an HTM-like cortical algorithm is subjected to a sudden disabling of a large percentage (50% or 75%) of the mini-columns, each of which contains a group of cells. Although the performance shows an initial drop, the system shows a remarkable resilience to reconfigure its resources autonomously and recover to nearly its original performance level within a reasonable time. Behind this ability to recover is the ability of the system simply to form new synapses when existing ones are destroyed.

6. DISCUSSION AND CONCLUSION

The purpose of this work has been to do a feasibility study of using 3D-WSI to realize a neuronal system approaching human brain levels of neuron count and connectivity. We now review our findings to understand where the biggest gaps and challenges are.

Using a simple model to emulate network connectivity, we found that the high bandwidth afforded by metal lines directly on the wafer and by TSVs between wafers is

quite adequate for the expected traffic. When compounded over the entire system with thousands of nodes, the total bandwidth capability compares very favorably to the bandwidth capability of the human brain which has been estimated at 1Tbps [Laughlin and Sejnowski 2003]. Communication latency also appears to be very low in our system. Aided by express channels, the latency can be well hidden during the compute cycle even for the worst case. Interestingly, it appears that the human brain is possibly more limited by the communication time (with axonal propagation times in the 20 ms range [Wen and Chlovskii 2005]) while the 3D-WSI system is more constrained by the compute time because of its inability to parallelize below the level of a single node. Still, the iteration time in the 10's of ms should allow for sensory perception on the time scale of a second. Finally, the estimated power consumption in the 10's of kW, while high, is manageable if advanced cooling techniques are used.

The biggest challenge appears to be in the memory requirements since we desire storage-class density with access time typical of volatile memories. The primate-like case described above, requiring 2 logic wafers and 16 DRAM wafers, is still feasible at today's DRAM densities, using wafer bonding techniques which to date have demonstrated stacks of 4 wafers. However, scaling up to the human-scale case using 128 DRAM wafers at today's density would not be feasible, pointing out the need for further memory innovation which would become even more important due to the slower scaling of DRAM feature size compared to digital CMOS. For the human-scale case, a 10x increase in density would be needed to bring the number of wafers down to a feasible level (<30) for 3D-WSI. These density increases may be possible with advances in emerging memory technologies [Meena et al. 2014]; one promising example is given in Cappelletti [2015].

Before advocating for the use of 3D-WSI, it is worthwhile to ask whether this system could practically be realized using the conventional method of packaged chips mounted on boards. While it is possible to achieve a comparable bandwidth using high-speed SERDES MGTs (multi-gigabit transceivers) [Kimura et al. 2014], the very high bandwidth consumes significant power to drive the high capacitance chip-to-chip and board-to-board connectors [Hasler and Marr 2013] and to maintain synchronization. Using Kimura et al. [2014], in which 28Gbps is achieved with 560mW in a 28nm technology, we estimate an added 4W per node just for the high bandwidth communications, resulting in an added 400kW for a human scale system. Each SERDES operation also adds a delay of $\sim$100ns for serialization and deserialization to the time for each hop. The beauty of not serializing lies in the inherent simplicity and power savings of parallel communication: power is consumed only when used, not all the time as is needed in a SERDES MGT to maintain clock synchronization. Finally, with approximately 25 chips on a board, the human scale system would require a roomful of racks to implement. The 3D-WSI system would be much more compact, even when control and input/output chips, power supply, and cooling system are included.

Finally, while we believe that 3D-WSI represents a strong next step in advancing brain computing, we comment on what could be next steps in a neuromorphic roadmap for further scaleup. First, as discussed, we have chosen as a starting point an average of 1000 synaptic connections per neuron, while a 10x increase to an average of 10000 per neuron would be desirable to improve the contextual capabilities of the system [Hawkins and Ahmad 2015]. In addition to much higher memory requirements, the network traffic would also increase markedly. As a possible futuristic enhancement, optical interconnects (see, for example, Schow et al. [2010]) offer one possible solution to greatly increasing the bandwidth-distance product. In addition, they offer the possibility of a freespace interconnect that can be flexibly reconfigured to meet changing network demands [Katayama et al. 2013].

Original absolute address of connection			
G	x	y	z
1011011001001011	0101001010101010	1011011101010110	1000101011010111
1011011001001011|0101001010101010|1011011101010110|1000101011010111 64 total address bits to store			

Relative address of connection, with run length encoding				
sign (dG,dx,dy,dz)	dG	dx	dy	dz
0010 for ++-+	0000000000000000	0000000000011101	0000000000011101	0000000000011101
0010|011001|11101|0000000000011101|0000000000011101 4 sign bits + 6 leading zeros bits (27=011011 in binary) + 37 information bits (5 from x, 16 from y, 16 from z) 47 total address bits to store				

Relative address of connection, with run length encoding and rearrangement		
sign (dG,dx,dy,dz)	dG	dx_{msb}, dy_{msb}, dz_{msb}, dx_{smsb}, dy_{smsb}, dz_{smsb},...
0010 for ++-+	0000000000000000	00000000000000000000000000000000000111111111000111
0010|110001|111111111000111 4 sign bits + 6 leading zeros bits (49=110001 in binary) + 15 information bits 25 total address bits to store		

Fig. 8. Example of address compression for a single synaptic connection in a 64-bit addressing space. The top frame shows the original representation of the connection using its absolute address. The middle frame shows some improvement when the relative address is used along with run length encoding. The bottom frame shows significant additional improvement when relative address, run length encoding, and rearrangement are used. In the bottom frame, msb refers to most significant bit, smsb to the second most significant bit, and so forth.

In conclusion, we have found that 3D-WSI offers a feasible path for realization of a next-generation computing system with significant cognitive capabilities that could complement today's enterprise servers. The ability to tackle unstructured computation problems would have significant impact on a wide variety of fields such as cybersecurity, healthcare, public safety, economics and finance, and robotics.

APPENDIX

This Appendix describes a method to store the addresses of synaptic using a highly compressed representation (run length encoding) that takes advantage of the observation that the overwhelming majority of a neuron's connections are located in the local vicinity of the neuron. Naively, for M total neurons in the system, $\log_2 M$ address bits are required to specify an absolute address for each synaptic connection. If M is very large, the number of address bits can significantly exceed the number of data bits being stored with the address, resulting in a highly inefficient ratio of address bits to data bits. However, we can improve on this situation by storing, for a given neuron, the address of each of its connections relative to the address of the given neuron, rather than as an absolute address. Since the vast majority of connections are local, the relative addresses will contain a large number of leading zeros which can be stored compactly by noting only the number of nonzeros in the relative address.

Figure 8 gives a concrete example for the case of 64-bit addressing. First, the original absolute address is divided into 4 fields consisting of a 16-bit global address field G of 16 bits and three 16-bit local address fields, x,y, and z. Suppose that a given neuron, whose relative address we take to be all 0's, makes a connection with another neuron located in the same region, but +23 units away in x, −23 units away in y, and

+23 units away in z, which we denote as (+10111,−10111,+10111) in binary. As a first compression, we could represent this address, consisting of 27 leading zeros followed by 37 bits of information, using 47 bits (6 bits for the number of leading zeros, 37 bits of information, and 4 bits for the sign of each address field), instead of the original 64 bits. However, if we regroup the x, y, z subfields such that we take the most significant bit of each, then the second most significant bit of each, and so forth, we can increase the number of leading zeros from 27 to 49, requiring only 25 bits to store (6 bits for the number of leading zeros, 15 data bits, and 4 bits for the sign of each address field).

ACKNOWLEDGMENTS

The authors would like to thank Sameh Asaad, Geoff Burr, Charles Cox, Doug Joseph, Yasanao Katayama, Toshi Kirihata, Dean Lewis, Pritish Narayanan, Ahmet Ozcan, Thomas Roewer, and Campbell Scott for stimulating discussions; Spike Narayan for management support; and Janus Marecki for the use of Figure 7.

REFERENCES

ARM website, www.arm.com/products/processors/cortex-m.

Luis M. Arzubi. 1973. Memory system with temporary or permanent substitution of cells for defective cells. (Aug. 1973). Patent No. US 4112512 A, Filed Jun. 1, 1972, Issued Aug. 28, 1973.

Mostafa R. Azghadi, Nicolangelo Iannella, Said F. Al-Sarawi, Giacomo Indiveri, and Derek Abbott. 2014. Spike-based synaptic plasticity in silicon: Design, implementation, application, and challenges. *Proc. IEEE* 102, 5 (May 2014), 717–737. DOI:http://dx.doi.org/10.1109/JPROC.2014.2314454

Maryam Bahmani, Abbas Sheibanyrad, Frédéric Pétrot, Florentine Dubois, and Paolo Durante. 2012. A 3D-NOC Router implementation exploiting vertically-partially-connected topologies. In *Proceedings of the 2012 IEEE Computer Society Annual Symposium on VLSI (ISVLSI'12)*. IEEE, 9–14. DOI:http://dx.doi.org/10.1109/ISVLSI.2012.19

Danielle S. Bassett and Ed Bullmore. 2006. Small-World Brain Networks. *The Neuroscientist* 12, 6 (Dec. 2006), 512–523. DOI:http://dx.doi.org/10.1177/1073858406293182

Pooja Batra, Spyridon Skordas, Douglas LaTulipe, Kevin Winstel, Chandrasekharan Kothandaraman, Ben Himmel, Gary Maier, Bishan He, Deepal Wehella Gamage, John Golz, Wei Lin, Tuan Vo, Deepika Priyadarshini, Alex Hubbard, Kristian Cauffman, Brown Peethala, John Barth, Toshiaki Kirihata, Troy Graves-Abe, Norman Robson, and Subramanian Iyer. 2014. Three-dimensional wafer stacking using Cu TSV integrated with 45 nm high performance SOI-CMOS embedded DRAM technology. *J. Low Power Electron. Appl.* 4 (2014), 77–89. DOI:http://dx.doi.org/10.3390/jlpea4020077

Perrine Batude, C. Fenouillet-Beranger, L. Pasini, V. Lu, F. Deprat, L. Brunet, B. Sklenard, F. Piegas-Luce, M. Cassé, B. Mathieu, O. Billoint, G. Cibrario, O. Turkyilmaz, H. Sarhan, S. Thuries, L. Hutin, S. Sollier, J. Widiez, L. Hortemel, C. Tabone, M.-P Samson, B. Previtali, N. Rambal, F. Ponthenier, J. Mazurier, R. Beneyton, M. Bidaud, E. Josse, E. Petitprez, O. Rozeau, M. Rivoire, C. EuvardColnat, A. Seignard, F. Fournel, L. Benaissa, P. Coudrain, P. Leduc, J.-M. Hartmann, P. Besson, S. Kerdiles, C. Bout, F. Nemouchi, A. Royer, C. Agraffeil, G. Ghibaudo, T. Signamarcheix, M. Haond, F. Clermidy, O. Faynot, and M. Vinet. 2015. 3DVLSI with CoolCube process: An alternative path to scaling. In *Proceedings of IEEE Symposium on VLSI Technology (VLSI Technology'15)*. IEEE, T48–T49. DOI:http://dx.doi.org/10.1109/VLSIT.2015.7223698

Rajendra V. Boppana and Suresh Chalasani. 1995. Fault-tolerant wormhole routing algorithms for mesh networks. *IEEE Trans. Comput.* (Jul. 1995) 44, 7, 848–864. DOI:http://dx.doi.org/10.1109/12.392844

Shekhar Borkar and Andrew A. Chien. 2011. The future of microprocessors. *Commun. ACM* 54, 5 (May 2011), 67–77. DOI:http://dx.doi.org/10.1145/1941487.1941507

Geoffrey W. Burr, Robert M. Shelby, Severin Sidler, Carmelo di Nolfo, Junwoo Jang, Irem Boybat, Rohit S. Shenoy, Pritish Narayanan, Kumar Virwani, Emanuele U. Giacometti, Bülent N. Kurdi, and Hyunsang Hwang. 2014. Experimental demonstration and tolerancing of a large-scale neural network (165,000 synapses), using phase-change memory as the synaptic weight element. In *Proceedings of the IEEE International Electron Devices Meeting (IEDM'14)*. 29.5.1–29.5.4. DOI:http://dx.doi.org/10.1109/IEDM.2014.7047135

Daniel P. Buxhoedeveden and Manuel F. Cassanova. 2002. The minicolumn hypothesis in neuroscience. *Brain: A Journal of Neurology.* 125, 5 (May 2002), 935–951. DOI: http://dx.doi.org/10.1093/brain/awf110

Paolo Cappelletti. 2015. Non volatile memory evolution and revolution. In *Proceedings of the IEEE International Electron Devices Meeting (IEDM'15)*. IEEE, 10.1.1–10.1.4. DOI:http://dx.doi.org/10.1109/IEDM.2015.7409666

Chun-Chuan Chi, Cheng-Wen Wu, Min-Jer Wang, and Hung-Chih Lin. 2013. 3D-IC Interconnect test, diagnosis and repair. In *Proceedings of the 2013 IEEE 31st VLSI Test Symposium (VTS'13)*. 1–6. DOI:http://dx.doi.org/10.1109/VTS.2013.6548905

Christine E. Collins, David C. Airey, Nicole A. Young, Duncan B. Leitch, and Jon H. Kaas. 2010. Neuron densities vary across and within cortical areas in primates. *Proc. of the National Academy of Sciences.* 107, 36 (Jul. 2010), 15927–15932. DOI: http://dx.doi.org/10.1073/pnas.1010356107

Richard F. Freitas and Winfried W. Wilcke. 2008. Storage-class memory: The next storage system technology. *IBM J. Res. Dev.* 52, 4/5 (Jul./Sep. 2008), 439–447. DOI:http://dx.doi.org/10.1147/rd.524.0439

Steve B. Furber, Francesco Galluppi, Steve Temple, and Luis A. Plana. 2014. The SpiNNaker project. *Proc. IEEE* 102, 5 (May 2014), 652–665. DOI:http://dx.doi.org/10.1109/JPROC.2014.2304638

Patric Hagmann, Maciej Kurant, Xavier Gigandet, Patrick Thiran, Van J. Wedeen, Reto Meuli, and Jean-Philippe Thiran. 2007. Mapping Human Whole-Brain Structural Networks with Diffusion MRI. *PLOS One* 2, 7, e597 (Jul. 2007), 9 pages. DOI:http://dx.doi.org/10.1371/journal.pone.0000597

Jennifer Hasler and Bo Marr. 2013. Finding a roadmap to achieve large neuromorphic hardware systems. *Frontiers in Neuroscience* 7, Article 118 (10 Sep. 2013), 29 pages. DOI:http://dx.doi.org/10.3389/fnins.2013.00118

Jeff Hawkins and Subutai Ahmad. 2015. Why neurons have thousands of synapses, a theory of sequence memory in neocortex. arXiv preprint arXiv:1511.00083

Jeff Hawkins and Sandra Blakeslee. 2004. *On Intelligence*. Times Books, New York, NY.

Bernard Hellwig. 2000. 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. DOI:http://dx.doi.org/10.1007/PL00007964

Mark D. Humphries, Kevin Gurney, and Tony J. Prescott. 2006. The brainstem reticular formation is a small-world, not scale-free, network. In *Proceedings of the Royal Society of London B: Biological Sciences* 273, 1585 (Feb. 2006), 503–511. DOI:http://dx.doi.org/10.1098/rspb.2005.3354

Giacomo Indiveri and Timothy K. Horiuchi. 2011. Frontiers in neuromorphic engineering. *Frontiers in Neuroscience* 5, 118 (Oct. 2011). DOI:http://dx.doi.org/10.3389/fnins.2011.00118

Subramanian S. Iyer. 2014. Three-dimensional integration: An industry perspective. *MRS Bulletin*. DOI:http://dx.doi.org/10.1557/mrs.2015.32

Subramanian S. Iyer. 2015. Monolithic three-dimensional integration for memory scaling and neuromorphic computing. *Proceedings of the IEEE SOI-3D-Subthreshold Microelectronics Technology Unified Conference (S3S'15)*. DOI:http://dx.doi.org/10.1109/S3S.2015.7333508

Yasunao Katayama, Atsuya Okazaki, and Nobuyuki Ohba. 2013. Software-defined massive multicore networking via freespace optical interconnect. In *Proceedings of the ACM International Conference on Computing Frontiers*. DOI:http://dx.doi.org/10.1145/2482767.2482802

John E. Kelly III and Steve Hamm. 2013. *Smart Machines*. Columbia University Press, New York, NY.

Hiroshi Kimura, Pervez M. Aziz, Tai Jing, Ashutosh Sinha, Shiva Prasad Kotagiri, Ram Narayan, Hairong Gao, Ping Jing, Gary Hom, Anshi Liang, Eric Zhang, Aniket Kadkol, Ruchi Kothari, Gordon Chan, Yehui Sun, Benjamin Ge, Jason Zeng, Kathy Ling, Michael C. Wang, Amaresh Malipatil, Lijun Li, Christopher Abel, and Freeman Zhong. 2014. 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 J. Solid-State Circ.* 49, 12, (2014), 3091–3103. DOI:http://dx.doi.org/10.1109/JSSC.2014.2349974

Arvind Kumar and Winfried Wilcke. 2015. Space-efficient dynamic addressing in very large sparse networks. Patent filed in USPTO (Nov. 2015).

Arvind Kumar, Winfried Wilcke, Subramanian Iyer, Toshiaki Kirihata, Daniel Berger, and Troy Graves-Abe. 2015. Architecture and implementation of a cortical system, and fabricating an architecture using 3D wafer scale integration. Patent filed in USPTO (May 2015).

Duygu Kuzum, Shimeng Yu, and HS Philip Wong. 2013. Synaptic electronics: materials, devices and applications. *Nanotechnology* 24,38, 382001. DOI:http://dx.doi.org/10.1088/0957-4484/24/38/382001

Simon B. Laughlin and Terrence J. Sejnowski. 2003. Communication in neuronal networks. *Science* 301 (26 Sep. 2003), 1870–1874. DOI:http://dx.doi.org/10.1126/science.1089662

Yann LeCunn, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep Learning. *Nature* 521 (28 May 2015), 436–444. DOI:http://dx.doi.org/10.1038/nature.14539

Peter Lennie. 2003. The cost of cortical computation. *Current biology* 13, 6 (2003), 493–497. DOI:http://dx.doi.org/10.1016/S0960-9822(03)00135-0

Wei Lin, Johnathan Faltermeier, Kevin Winstel, Spyridon Skordas, Troy Graves-Abe, Pooja Batra, Kenneth Herman, John Golz, Toshiaki Kirihata, John Garant, Alex Hubbard, Kris Cauffman, Theodore Levine, James Kelly, Deepika Priyadarshini, Brown Peethala, Raghuveer Patlolla, Matthew Shoudy, James

J Demarest, Jean Wynne, Donald Canaperi, Dale McHerron, Dan Berger, and Subramanian Iyer. 2014. Prototype of multi-stacked memory wafers using low-temperature oxide bonding and ultra-fine-dimension copper through-silicon-via interconnects. In *Proceedings of the IEEE SOI-3D-Subthreshold Microelectronics Technology Unified Conference (S3S'14)*. IEEE, 1–3. DOI:http://dx.doi.org/10.1109/S3S.2014.7028246

Wolfgang Maass. 1997. Networks of spiking neurons: The third generation of neural network models. *Neural Netw.* 10, 9 (Dec. 1997), 1659–1671. DOI:http://dx.doi.org/10.1016/S0893-6080(97)00011-7

David May, Peter Thompson, Brian Parsons, and Christopher Walker. 1997. Message Routing. Tech. Rep. University of Bristol, Bristol, UK.

Carver Mead. 1990. Neuromorphic electronic systems. *Proc. IEEE* 78, 10 (Oct. 1990), 1629–1636. DOI:http://dx.doi.org/10.1109/5.58356

Jagan S. Meena, Simon M. Sze, Umesh Chand, and Tseung-Yuen Tseng. 2014. Overview of emerging nonvolatile memory technologies. *Nanoscale research letters* 9, 1 (2014), 1–33. DOI:http://dx.doi.org/10.1186/1556-276X-9-526

Paul A. Merolla, John V. Arthur, Rodrigo Alvarez-Icaza, Andrew S. Cassidy, Jun Sawada, Filipp Akopyan, Bryan L. Jackson, Nabil Imam, Chen Guo, Yutaka Nakamura, Bernard Brezzo, Ivan Vo, Steven K. Esser, Rathinakumar Appuswamy, Brian Taba, Arnon Amir, Myron D. Flickner, William P. Risk, Rajit Manohar, and Dharmendra S. Modha. 2014. A million spiking neuron integrated circuit with a scalable network communication and interface. *Science* 345 (8 Aug. 2014), 668–673. DOI:http://dx.doi.org/10.1126/science.1254642

Micron Website, www.micron.com/products/dram.

Richard C. Murphy and Peter M. Kogge. 2007. On the memory access patterns of supercomputer applications: Benchmark selection and its implications. *IEEE Trans. Computers* 56, 7 (Jul. 2007), 937–945. DOI:http://dx.doi.org/10.1109/TC.2007.1039

NIH Brain Initiative website, www.braininitiative.nih.gov.

Bente Pakkenberg and Hans Jørgen G. Gundersen. 1997. Neocortical neuron number in humans: effect of sex and age. *J Comp. Neurol.* 384, 2 (1997), 312–320. DOI:http://dx.doi.org/10.1002/(SICI)1096–9861(19970728)384:2%3C312::AID-CNE10%3E3.0.CO;2-K

Rodrigo Perin, Thomas K. Berger, and Henry Markram. 2011. A synaptic organizing principle for cortical neuronal groups. *PNAS* 108,113 (29 Mar. 2011), 5419–5424. DOI:http://dx.doi.org/10.1073/pnas.1016051108

Bipin Rajendran, Yong Liu, Jae-sun Seo, Kailash Gopalakrishnan, Leland Chang, Daniel J. Friedman, and Mark B. Ritter. 2013. Specifications of nanoscale devices and circuits for neuromorphic computational systems. *IEEE Trans. Elect. Dev.* 60, 1 (Jan. 2013), 246–253. DOI:http://dx.doi.org/10.1145/1188913.1188915

Norm Robson, John Safran, Chandrasekharan Kothandaraman, Alberto Cestero, Xiang Chen, Raj Rajeevakumar, Alan Leslie, Dan Moy, Toshiaki Kirihata, and Subramanian Iyer. 2007. Electrically programmable fuse (eFUSE): From memory redundancy to autonomic chips. In *Proceedings of the 2007 IEEE Custom Integrated Circuits Conference (CICC'07)*. 799–804. DOI:http://dx.doi.org/10.1109/CICC.2007.4405850

Evtan Ruppin, Eric L. Schwartz, and Yehezkel Yeshurun. 1993. Examining the volume efficiency of the cortical architecture in a multi-processor network model. *Biolog. Cybernet.* 70, 1 (1993), 89–94. DOI:http://dx.doi.org/10.1007/BF00202570

Johannes Schemmel, Andreas Grubl, Stephan Hartmann, Alexander Kononov, Christian Mayr, Karlheinz Meier, Sebastian Millner, Johannes Partzsch, Stefan Schiefer, Stefan Scholze, Rene Schuffny, and Marc-Olivier Schwartz. 2012. Live demonstration: A scaled-down version of the BrainScaleS wafer-scale neuromorphic system. In *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS'12)*. 702–02. DOI:http://dx.doi.org/10.1109/ISCAS.2012.6272131

Clint Schow, Fuad Doany, and Jeffrey Kash. 2010. Get on the optical bus. *IEEE Spectrum* 47, 9 (Sept. 2010), 32–56. DOI:http://dx.doi.org/10.1109/MSPEC.2010.5557513

Kamal Sikka, Arvind Kumar, and Babar Khan. 2015. Methods of Cooling and Power Delivery for a PssWafer Level Compute Board. Patent filed in USPTO (Dec. 2015).

Techinsights. 2013. Technology Roadmap of DRAM for Three Major manufacturers: Samsung, SK Hynix and Micron. (May 2013). Retrieved from http://www.techinsights.com/uploadedFiles/Public_Website/Content_-_Primary/Marketing/2013/DRAM_Roadmap/Report/TechInsights-DRAM-ROADMAP-052013-LONG-version.pdf.

Thomas Vogelsang. 2010. Understanding the energy consumption of dynamic random access memories. In *Proceedings of the 2010 43rd Annual IEEE/ACM International Symposium on Microarchitecture*. 363–374. DOI:http://dx.doi.org/10.1109/MICRO.2010.42

John Von Neumann and Michael D. Godfrey. 1993. First draft of a report on the EDVAC. *IEEE Annals of the History of Computing* 15, 4 (1993), 27–75. DOI: http://dx.doi.org/10.1109/85.238389

Duncan J. Watts and Steven H. Strogatz. 1998. Collective dynamics of 'small-world' networks. *Nature* 393 (4 Jun. 1998), 440–442. DOI: http://dx.doi.org/10.1038/30918

Quan Wen and Dmitry B. Chklovskii. 2005. Segregation of the brain into gray and white matter: A design minimizing conduction delays. *PLOS Computat. Biol.* 1, 7, e78 (Dec. 2005), 14 pages. DOI: http://dx.doi.org/10.1371/journal.pcbi.0010078

Dong H. Woo, Nak H. Seong, Dean L. Lewis, and Hsien-Hsin S. Lee. 2010. An optimized 3D-stacked memory architecture by exploiting excessive, high-density TSV bandwidth. In *Proceedings of the IEEE 16th International Symposium on High Performance Computer Architecture (HPCA'10).* IEEE, 1–12. DOI: http://dx.doi.org/10.1109/HPCA.2010.5416628

John Worobey, Beverly J. Tepper, and Robin B. Kanarek. 2015. *Nutrition and Behavior, A Multidisciplinary Approach.* Cabi.

Received December 2015; revised May 2016; accepted July 2016