System Impact of 3D Processor-Memory Interconnect: A Limit Study

Mitchelle Rasquinha, Syed Minhaj Hassan, William Song, Kwanyeob Chae, Minki Cho,
Saibal Mukhopadhyay, Sudhakar Yalamanchili
School of ECE, Georgia Institute of Technology, Atlanta GA-30332
{mitchelle.rasquinha,minhaj,wjhsong,ky.chae,mcho8}@gatech.edu,{saibal,sudha}@ece.gatech.edu

***Abstract*—3D integration with through-silicon-vias (TSVs) can provide enormous bandwidth between processor die and memory die. The central goal of our work is to explore the *limits* of performance improvement that can be achieved with such integration. Towards this end we propose a model of the impact of 3D TSVs on system performance. The model leads to several key observations i) increased miss tolerance (smaller caches) and hence improved core scaling for a fixed die size, ii) higher sustained IPC per core, iii) significantly smaller, energy efficient DRAM banks, iv) redistribution of system power to the cores and on -die interconnect, and v) TSV utilization is a function of the relationship between reference locality and the bandwidth properties of the intradie network. These observations are repeated in cycle level simulations of a 64 tile architecture.**

I. Introduction

The central goal of our work is to explore the *limits* of performance improvement that can be achieved with 3D integration. This is to be distinguished from the large body of work that explores how modern and emerging many core architectures can best exploit the physical properties of 3D packaging, for example partitioning of a design to exploit the physical 3D geometry to realize shorter critical paths, or partitioning of cache architectures across multiple layers [1]. The question we are interested in is whether the physical properties of 3D integration can be amplified into 2X, 5X, or 10X improvements in system performance through system and micro-architecture innovations. The answer to this question has important implications for whether effort should be devoted to achieving these 10X gains or whether 3D integration is primarily a manufacturing technology (with its own attendant challenges) with a direct and predictable impact on performance and without major multipliers of performance. The importance or lack thereof of system and micro-architecture research for 3D systems depends on whether the limit analysis identifies headroom for large performance multipliers. We are not considering embedded systems which clearly have different set of constraints and opportunities to exploit 3D technology. We are more interested in the impact on systems that may be used in data centers and high performance computing.

In this paper we study one aspect of 3D integration - interdie bandwidth that is applied to deliver processor memory bandwidth. We chose this aspect due to the importance of processor-memory bandwidth in commodity systems (both high-end and low-end) and since this organization retains the technology customization of processor vs memory dies as well as recognizes the distinct market segments within which processor and memory vendors operate. The Exa-scale report [2] points out how the imbalance between compute and memory resources increases when clock speeds and feature sizes scale at a rate faster than off-chip memory bandwidth. 3D system integration can be seen as one way to reduce this imbalance as memory bandwidth can be scaled as a function of TSV technology. The question is what is the limit of system performance that can be achieved?

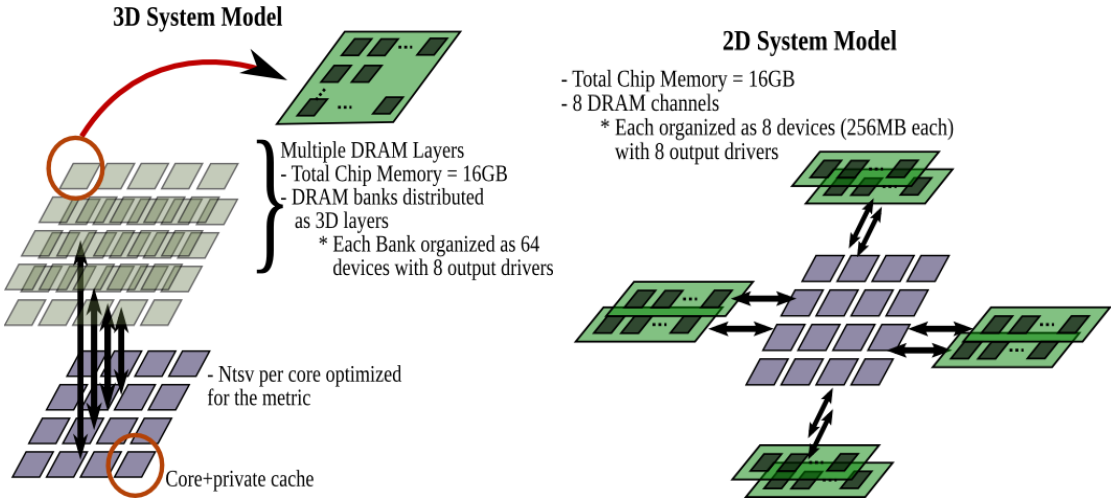
Using a model developed here we have analyzed a design space of 2D/3D systems over a wide range of cores (0-150) and memory bandwidths. We observe performance gains for fixed core complexity, power and area budgets to be in the range of 4X-6X. Furthermore, we have observed that the increased interdie bandwidth can be translated into i) higher cache miss tolerance leading to improved core scaling (via smaller caches) for a fixed die size, ii) increased sustainable IPC, iii) improved performance/watt/mm^2 and iv) redistribution of a greater percentage of system power to the cores and on-die interconnect. However, the increased TSV bandwidth also encourages smaller DRAM bank sizes (for energy efficient design). The consequence of this is that the ability to make effective use of this bandwidth is limited by the relationship between memory reference locality and the bandwidth properties of the on-die network. The predicted trends are reproduced by experiments using cycle level by micro-architectural simulation of a 64-tile system.

The remainder of the paper is organized as follows. Section II describes the analytical model used for predicting the trends in section III and some of the trends validated by simulation is presented in section IV.

II. Parameterized Analytical Model

A. System Model Formulation and Assumptions

In this section we present a parameterized model for evaluating the impact of increased TSV bandwidth on system performance. A 3D integrated system comprises different components belonging to three main categories namely core, network and memory. Analytical models for each of the sub-categories in isolation have been previously introduced [3], [4], [5] and they can be intuitively combined to develop a model for the 3D system. However, owing to the large design space, a direct combination of the individual component models



(a) 3D configuration (b) Baseline 2D configuration

Fig. 1. Modeled System Architecture

can make the problem size intractable. Thus we attempt to capture in our model only those parameters that we think will be dominant as systems scale. For example, memory controller power is not modeled since to adequately utilize the large 3D bandwidth we find that memory controllers are considerably simplified and do not have a major impact on system throughput and power relative to other parameters such as memory latency and intradie network bandwidth which dominate.

The 3D organization (shown in figure 1(a)) consists of multiple layers of DRAM, organized into fine grained sub-banks directly accessible to the cores on the lowest layer. The fine grained sub-banking of DRAM has also been proposed in [6] for optimized energy/bit. The total available DRAM for the system is fixed at 16GB, with individual DRAM banks of 256 MB. The lowest layer comprised of core and cache tiles is interconnected by a 2D mesh network built with 6 port routers— 4 ports to the neighboring tiles, the local core tile and the vertical port to memory. The network is wormhole switched. Multiple DRAM layers are accessible through this memory port. Operationally, memory requests and responses traverse the intradie network in the processor die to the correct tile before traversing the low latency TSVs to the destination DRAM bank.

The 3D system is evaluated in contrast to a (baseline) 2D many-core system 1(b) with the memory subsystem comprising multiple on-chip memory controllers (MCs) at the periphery connected to DDR3 DRAM devices. The total available DRAM for the system is 16GB, with individual DRAM banks of 4GB. Memory requests and responses traverse the on-chip network to/from the MCs. The DRAM access latencies are dependent on the size of the sub-bank and is reported in Table I. Note, the routers for the baseline system are five port routers (excluding the 8 tiles connected to memory controllers).

In both the 2D and 3D organizations each network router pipeline is a generic four stage pipeline with power and area models for buffering, arbitration and the crossbar links.

The base architecture assumptions are the following. Note that assumptions are based in our goal of studying the limits of performance scaling that can be achieved as by product of increased TSV bandwidth.

- The core model is for a SMT core, where the threads are assumed to be able to context switch in a single cycle. This is not unreasonable given the fine grain multi-threading evidenced in modern throughput oriented cores (e.g., NVIDIA's GPGPU processors).
- The instructions per cycle (IPC) achievable by a core is equal to the commit width. This represents an ideal and limiting case.
- For 3D organization, the access latency to any of the DRAM layers is a constant technology dependent value. This is based on the fact that TSV energy-delay is negligible relative to DRAM access energy-delay product.
- The latency through the cache is a fixed average value encompassing the average latency.

With the goal of analyzing the limits of performance improvement and their impact, we adopt the following metric to evaluate for a given workload, under fixed area constraints.

$$metric = \frac{Performance}{Power} \tag{1}$$

The following is a list of the key parameters used in the next sections to describe the analytical model.

- r: Probability of a memory operation.
- N_{cores}: Number of cores.
- $n_{threads}$: Total Number of threads in the system.
- N_{tpc}: Number of threads per core ($n_{threads}/N_{cores}$).
- N_{tsv}: Number of TSVs for 3D and total off-chip DRAM pins for 2D.
- \$C: Cache capacity per core in KB.
- m: Miss rate of the cache for a given \$C.
- T_{memss}:Latency of a memory operation in cycles.
- f_{core}: Frequency of the cores.
- L: Cache line size in bits.
- $t_{cache}, t_{dram}, t_{bus}, t_{net}$: Latency of a cache, DRAM sub-bank, bus/TSV and multi-hop network access respectively in cycles. The bus here refers to the DDR3 bus.

- A_{core},A_{cache},A_{net}: Area for core, cache and router respectively.
- P_{core},P_{cache},P_{net},P_{bus},P_{mem}: Power for core, cache, router, bus and DRAM respectively.
- H: Average round trip hop count in the on-die interconnect for a memory request.

$diearea$	$300mm^2$
IPC	4
r	0.3
L	512
M_0	$10^{0.5}$
f_{core}	2 GHz
v_{dd_bus}	0.68
C_{bus} 2D/3D	$20pf$ /$30fF$
f_{bus} 2D/3D	1.6 GHz/f_{core}
p_{mem_acc} 2D	383mW
Per hop network power 2D/3D	118.5mW/162mW
Average no of hops per memory access 2D/3D	6/2.5
$tmem\_acc$ Access for a sub-bank 2D/3D	3.85ns/ 1ns
prouter acc3d	162mW
N_{tsv} for 3D	512 × L
N_{core}	64
N_{thread}	4*N_{core}
$A_{router} + A_{link}$	$0.22mm^2$
a_{cache}	$C/2400
t_{cache}	4 cycles
Router latency constants (A_0/A_1)	4/4 cycles
Per access core power constants (a/b)	0.0361/0.8
Per access cache power constants (c/d/e)	0.9/1/8
Total available DRAM	16 GB

TABLE I
TABLE OF BASELINE PARAMETER VALUES

B. Performance Model

The performance model extends the model in [3] wherein

$$Performance = N \times f \times \eta \times IPC \tag{2}$$

where η is the core utilization given by

$$\eta = min\left(1, \frac{N_{tpc}}{1 + T_{memss} \times r \times IPC}\right) \tag{3}$$

Threads on a core are switched on a cache miss. For a single thread, memory accesses are encountered at a rate of $r \times m \times IPC$ per cycle. The number of threads required to keep the core busy is $(1 + T_{memss} * IPC * r)$. We extend the memory access time, T_{memss}, to include latencies due to network congestion and DRAM bus contention as given by:

$$T_{memss} = t_{cache} + m\left(t_{dram} + t_{net} + contention\_lat\right) \tag{4}$$

For an SMT core, with a cache equally divided among threads the miss rate per core, m is given by [7]:

$$m = M_0 \left(\frac{\$C}{N_{tpc}}\right)^{-\alpha} \tag{5}$$

where M_0 is a constant dependent on the unit of cache size considered and α is 0.5.

The DRAM bank access time(t_{dram}) was evaluated using MCPAT [8]. This evaluation advocates small DRAM banks [9] for low access latency, in accordance with [10] which advocates eliminating the L2 cache and using its area for other simple cores. In [11], Dong et.al. also advocate that the L2 and DRAM should be redesigned to make use of the TSV bandwidth.

The network latency t_{net} is given by

$$t_{net} = H \times A_0 + A_1 + k \times m \tag{6}$$

where A_0 is per hop latency dependent on the router micro-architecture and A_1 is the serialization latency due to wormhole switching and is dependent on the link width and packet length (Cache line L+header length in this case). The factor k accounts for network congestion latency in addition to the no load latency. The latency increase due to network congestion is a complex function of several factors (e.g. arbitration cycles, network buffering, packet lengths, traffic distribution and injection rates). However for the limit analysis we simplified the network congestion latency to be a factor of the injection (miss rate), as this is sufficient to highlight the impact of increased TSV bandwidth.

The second component of the network latency, which is the contention for the off-chip DRAM bus in 2D and TSV bandwidth in 3D respectively is given by:

$$contention\_lat = \frac{\beta}{N_{tsv}/\left(L \times N_{core}\right)} \tag{7}$$

where $N_{tsv}/(L * N_{core})$ provides the total number of interconnects (all are used for data) available per core. DRAM bus contention latency is modeled as β, where $\beta \geq 0$ is the average contention factor or number of requests waiting to be serviced per core and can be calculated using queuing theory models [12] based on given program characteristics. We use a basic M/M/1 queue with poison arrival patterns for which β can be given by

$$\beta = \frac{\lambda}{1 - \lambda s} \tag{8}$$

where λ is the arrival rate and s is the service time of the request. The service time is given by $s = t_{dram} + t_{bus}$ and the arrival rate of requests per core is $r * m$. Thus

$$\lambda s = \min\left(1, r \times m \times \left(t_{dram} + t_{bus}\right)\right) \tag{9}$$

The contention factor β is higher in 2D compared to 3D due to (relatively) limited off chip bandwidth (small N_{tsv} - note the notation captures this as the total number of data connections to DRAM in both 2D and 3D) and higher dram access latencies (t_{dram} and t_{bus} are smaller). The main parameters that differ in 2D and 3D organizations in the performance evaluation are t_{dram}, t_{bus} and N_{tsv}. The cache access latency is kept fixed at t_{cache} for both 2D and 3D.

C. Power Model

The system power is given by

$$P_{total} = P_{core} + P_{cache} + P_{network} + P_{bus} + P_{mem} \tag{10}$$

An important alteration to the system power in 3D comes from the reduced power of TSVs and smaller DRAM banks.

Total system core and cache power is given by:

$$P_{core} = p_{core} \times N_{core} \tag{11}$$

$$P_{cache} = p_{cache} \times N_{core} \quad (12)$$

where $p_{core} = \zeta a e^b$ and $p_{cache} = \gamma c m^d * N_{tpc}^e$ are the individual core and cache powers, empirically derived using MCPAT [8]. The constants ζ and γ account for the switching activity. The impact of temperature on leakage is not modeled.

The bus power model derived from [5] is given by the power for p accesses, each q bits wide for a homogeneous random channel

$$P_{bus} = \frac{1}{2} pq \times C \times V_{dd}^2 \times f \quad (13)$$

Given,

- power per access: $C_{bus} * V_{dd}^2 * f_{bus}$, where C_{bus} and f_{bus} is the total switched capacitance and frequency of operation of the bus/TSV for 2D/3D.
- no of accesses $p = r * m * N_{core}$
- bits per accesses $q = L$
- C_{bus} in the case of 2D is large due to the longer DRAM channel wiring and off-chip impedance matching circuitry.
- f_{bus} is assumed to be equal to f_{core} for 3D and fixed at 1 GHz for 2D.
- The values for each of the parameters is reported in Table I.

after substitution the bus power reduces to

$$P_{bus} = \frac{1}{2} rmN_{core} \cdot L \cdot C_{bus} V_{dd}^2 f_{bus} \quad (14)$$

The memory power model is given by

$$P_{mem} = \frac{1}{2} rmN_{core} \times L/W_{DRAM\_ch} \times p_{mem\_acc} \quad (15)$$

where p_{mem_acc} is the power per DRAM bank access obtained from simulation using Cacti [8] at 32nm and scaled to 16nm. The number of accesses is determined by the cache miss rate and percentage of memory operations encountered by the core. It should be noted that on each access the number of bits accessed is equal to the DRAM channel width, hence the factor L/W_{DRAM_ch}.

The network power model is given by

$$P_{net} = H\left(p_{acc\_router} + p_{acc\_link}\right) * L_{pkt}/W_{link} \quad (16)$$

where the per access router and link power was computed from Orion 2.0 [13] at 32nm and scaled to 16nm. We ignore the memory controller power in this evaluation, for 3D we noticed low buffer occupancies for MC queues and for 2D most of the power is consumed in DRAM as opposed to MC (also note MCs are limited compared to 3D), hence its exclusion does not alter the trends.

D. Area Model

A single tile in the case of 2D comprises a core, cache and router (only peripheral tiles have an MC). Each 3D tile is identical, comprising a core, cache, router and MC. The total die area is given by

$$A = [A_{core} + A_{cache} + A_{net}] \times N_{core} mm^2 \quad (17)$$

where the individual core and cache area are functions of IPC and $\$C$ respectively.The core area is given by $a_{core} = (5/9)2^{0.29 \times IPC}$ (router and cache area in table I).

In 3D, the number of MCs increase owing to the fine grained sub-banked design of DRAM, thereby increasing MC area costs. However, we believe that the complexity of each MC can significantly be reduced ie. 3D does not require complex scheduling algorithms for row buffer optimizations as exploiting parallelism via the increased bandwidth has higher performance gains. This observation is motivated by our simulation results on MC buffer occupancies and page hit rates described in section IV. For the area analytical model we ignore the MC area, but the simulation results of section IV is based on equal MC buffering across all MCs in both systems i.e. the total buffering over 8 MCs in 2D is the same as the total buffering over 64 MCs in 3D. This provides an unbiased analysis to the input memory reference stream in both cases.

III. Model Analysis and Observations

In this section we assess the impact of the increased memory bandwidth on system parameters and hence core scaling. Based on our analysis we highlight some key observations on performance gains, potential bottlenecks and consequences of increased memory bandwidth via 3D integration.

A. Performance Scaling

For a fixed problem size with a certain degree of concurrency, the performance for a fixed number of cores becomes dependent on the individual core utilizations. Given a core micro-architecture with fixed complexity in terms of frequency of operation, width of the pipeline, and number of threads available, the core utilization becomes dependent on the memory subsytem time. Once the executing threads become memory bound the core utilization is no longer dependent on the core parameters but the memory access time. The access time of the memory subsystem is a function of the cache miss rate and the available memory bandwidth.

Realizing deeper and larger on-chip caches in order to reduce the miss rate and hence improve core utilization is widely used today due to limited off-chip bandwidth in 2D systems. We emulate this effect by representing the memory access latency from section II-B as a function of the miss-rate:

$$T_{mem\_ss2D}(m) = t_{cache} + m * k1\,(m)$$

. where the function $k1(m)$ captures the latency components of the network and DRAM.

On the other hand for 3D systems, N_{tsv} bandwidth utilization can be improved by increasing the demand to memory or reducing $C.

$$T_{mem\_ss3D}(N_{tsv}) = t_{cache} + \frac{max\left(1, k2\left(contention\right)\right)}{N_{tsv}} + k3$$

Figure 2 shows performance scaling as a consequence of the preceding two models in i) reducing the cache sizes and hence increasing the cache miss rates from 12 to 18% ii) increasing N_{tsv} from 512 to 4M TSVs (miss rate per

core fixed at 36%). In effect the increased TSV bandwidth improves performance scaling as it shifts the point at which the core is memory bound and thus leads to high core utilization. This leads us to the first observation.

Observation 1: *In comparison to 2D, 3D can sustain high core utilization with smaller caches, leading to larger N_{core} for a fixed die size.*

While performance scaling is certainly desirable, it may not be feasible given a fixed power and area budget. Next we will evaluate the impact on system power utilization from the above mentioned approaches.

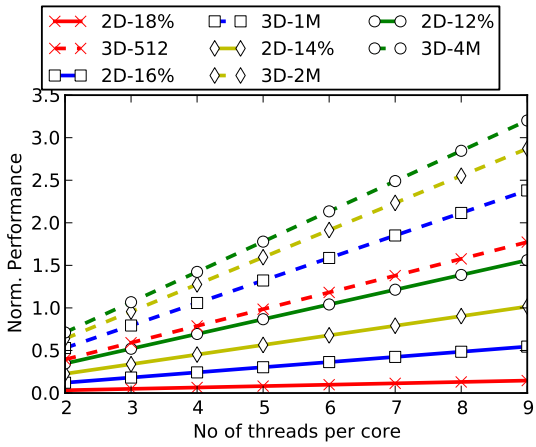


Fig. 2. Normalized performance (eqn. 2) scaling from i) Increasing $C ii) Increasing N_{tsv}

B. System Power Utilization

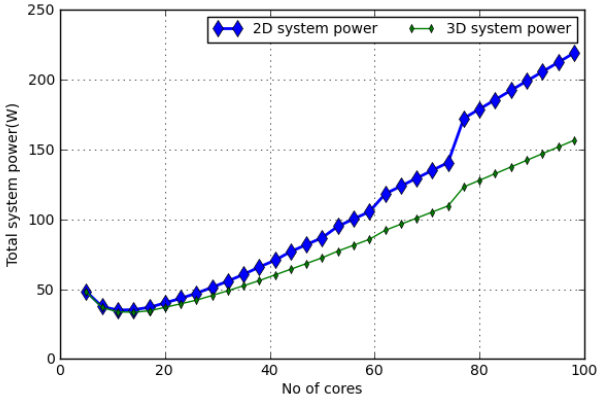


Fig. 4. Total system power in 2D and 3D

3D system integration provides for increased memory bandwidth at low power cost per unit of memory bandwidth. In this section we evaluate the redistribution of system power due to the use of TSVs. Figure 3 shows the system power breakdown. As can be seen, power spent in the off-chip buses is eliminated and the power consumed in DRAM is reduced due to smaller bank sizes. Hence, for a fixed power budget, power utilization is effectively shifted to the cores and the network. Under current design parameters the reduction in power is upto 29% at core counts 80 on a $300mm^2$ die (Figure 4).

While the shift of power distribution to cores enables computing resources to be scaled, it exacerbates 3D thermal challenges (a largely well established trend). This challenge is being addressed by an orthogonal set of technologies-heterogeneous cores, phase change materials, sophisticated DVFS and power gating schemes, non-volatile technologies. This analysis would have to be repeated in the context of those proposals and is considered beyond our current scope. The system power distribution from figure 3 leads to the following observation.

Observation 2: *At high core count, in 3D systems the percentage of the power budget devoted to off-chip buses and larger DRAM banks is shifted to the core and network.*

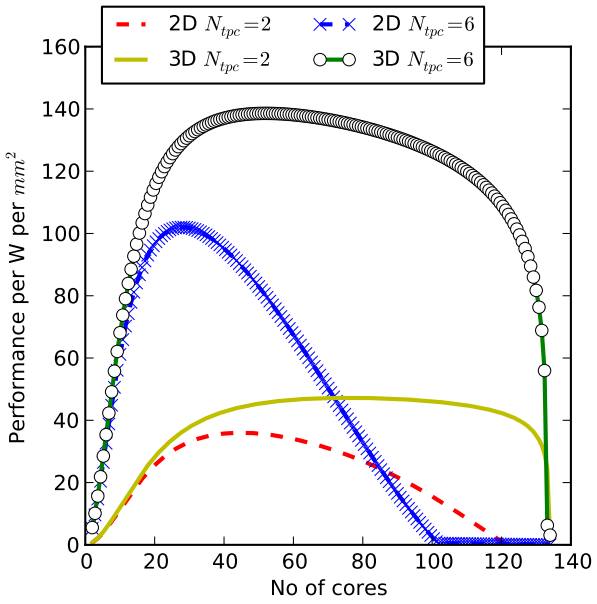


Fig. 5. Performance per unit power per unit area for 2D and 3D systems at varying core complexities.

Figure 5 shows the performance per watt per unit area for both the 2D and 3D cases for two different N_{tpc} leading to the following observation. Performance scaling from increasing on-chip caches is a suboptimal use of system power as caches are both power and area hungry and constrain resources that can be devoted to compute.

Observation 3: *TSV bandwidth enables improved performance and core scaling.*

C. Concurrency and memory bandwidth demand

From the above observations we have established that 3D provides high performance with reduced cache sizes. In this section we evaluate the impact of increased injection rates into the network and memory subsystems. Achieving performance scaling of memory-intensive parallel applications on future micro-architectures requires memory capacity and bandwidth

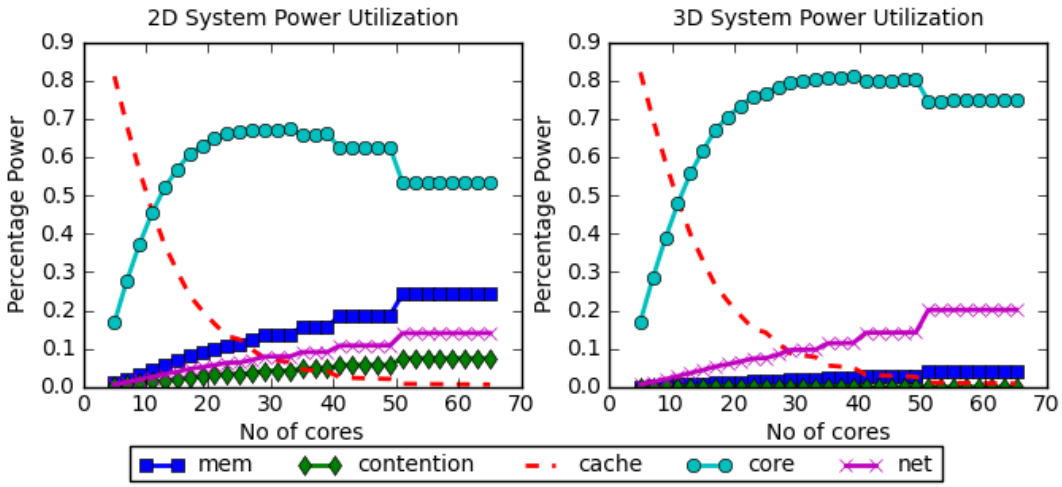


Fig. 3. Distribution of power in 2D and 3D

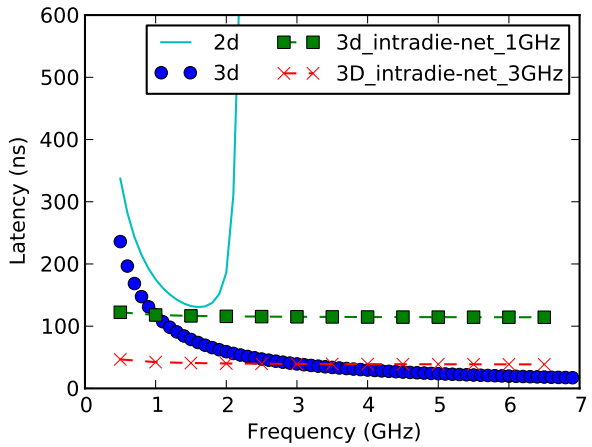


Fig. 6. Average memory latency at varying injection rates.

to scale. While TSVs provide for wide, high speed links in the vertical dimension they also make the network subsytem heterogeneous in nature i.e., intra-link bandwidths are a fraction of TSV links.

Figure 6 shows the average memory access latency as the system frequency (core+network+MC) is increased. Increasing the core and cache frequency modulates the injection rate into the network and memory- the same effect can be achieved by increasing the number of cores or increasing the core pipeline widths. As can be seen with current 2D systems, the latency drops rapidly but cannot be sustained at high injection rates due to limited off-chip resources. In the case of 3D with the network scaling at the same rate as the injection rate, the latency drop can be sustained even at high injection rates. Memory accesses can be highly concurrent in nature as the number of DRAM channels can be scaled as a function of the number of TSVs . However, locality of reference has a large impact: the intradie network becomes limiting as locality of reference decreases and memory access latency increasingly relies on the intradie network. This increases the importance of current signaling trends in the intradie network- the ability to scale at a rate fast enough to avoid starving the TSV's. For example, if we scale the core frequencies and limit the network frequency to 1 GHz there is no improvement in memory access latency. As expected, running the network at 3 GHz will lead to lower latencies, but no improvements can be seen as the injection rate is increased, thereby indicating that the third dimension bandwidth via TSVs is underutilized. This leads us to our following observation

Observation 4: *Intradie communication properties such as link and bisection bandwidth coupled with locality properties limit the ability to take advantages of intradie bandwidth improvements.*

As the number of cores scale, the percentage of data sharing among the individual tiles is bound to increase and so is the communication costs. The on-chip network becomes a crucial component of such designs, with 3D introducing a new dimension of heterogeneity to it.

D. Area Utilization

In this section we evaluate the area utilization in terms of maximizing the metric(Performance/($W * mm^2$) as a function of the following three parameters i) the IPC of each core ii) number of cores and iii) the total cache size. Given this dependence we swept the design space for all possible values of N_{core}, IPC and \$C, and found the point at which maximum performance can be obtained.

1) Metric vs IPC: As the IPC of a core is increased there is linear performance growth, but exponential power and area increase. Figure 7 shows the metric as a function of IPC. For each of the data points the design space over a range of N_{core} was explored and maximum metric value selected. As expected, in both cases the metric degrades for higher IPC, owing to increased power and area consumption per core. However, the high miss rate tolerance in 3D shifts the metric gain interval, thereby suggesting that relative to 2D, higher IPC cores can be sustained in 3D. This may be of importance in heterogeneous environments (multiple core types), system-on-package type environments and is contrary to the prevailing trend of lower IPC cores. In all cases the corresponding area

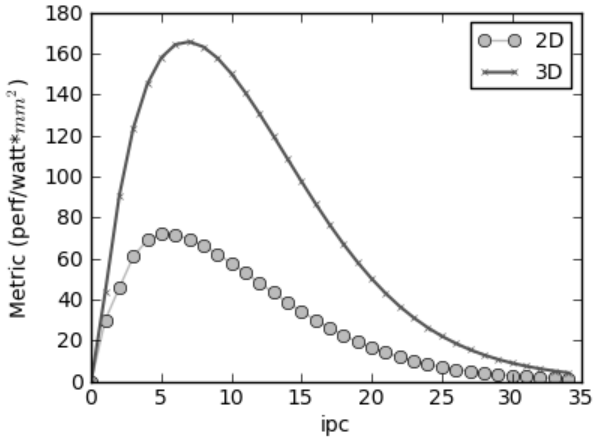


Fig. 7. Metric for 2D and 3D with optimal no of cores of varying IPC

of the die for the selected core counts was $300mm^2$(16nm Technology node).

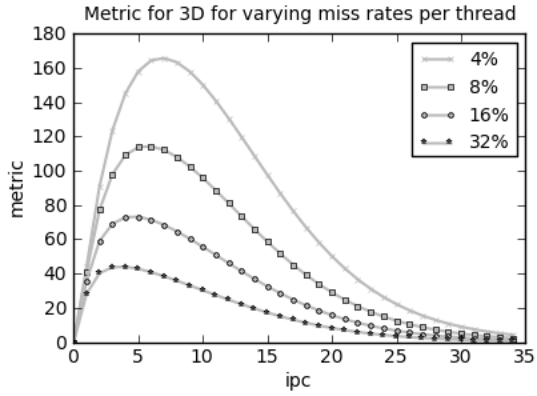


Fig. 8. Metric variation for 3D at different cache sizes.

2) Metric vs IPC for varying cache sizes: Figure 8 shows how the IPC gain region decreases as miss rate per thread increases for the 3D case. If each thread per core has sufficiently large caches to maintain the miss rate per thread at 4%, the metric scales till ≈8 for the individual core IPC, beyond this the power and area constraints dominate.

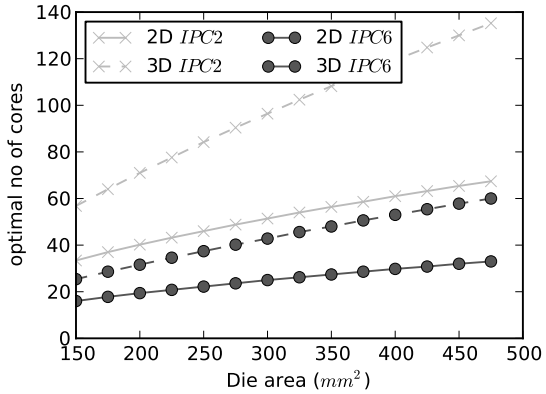


Fig. 9. Area utilization for varying number of cores.

3) Metric vs N_{core}: Figure 9 shows the total number of cores for maximum performance at varying die sizes. For a fixed area the design space was explored across cores. As can be seen at low IPCs the reduced memory service time in 3D allows for ≈1.8X more cores to be realized at die-areas of ≈$300mm^2$. However, for higher IPC cores the gain drops, due to high stress on memory bandwidth (For a $300mm^2$ die area ≈1.7X more cores with an IPC of 6 and ≈1.5X more IPC cores with an IPC of 10 can be realized). It should be noted that figure 9 is based on current technology parameters(16nm nodes) and future feature sizes will shift the point of divergence to smaller die areas.

This leads us to the following observations:

Observation 5: *At modest IPC values (<10) TSV bandwidth enables relatively higher IPC cores.*

Observation 6: *3D improves area utilization in favor of compute as more of the die area can be devoted for a larger number of cores that can sustain much higher miss rates per thread.*

IV. Results

In this section we validate some of the above observations through trace-based simulation of benchmarks from SPEC integer and floating point benchmark suites.

A. Simulation Methodology

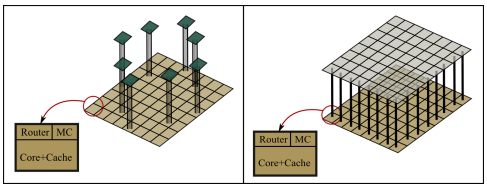


Fig. 10. Simulated Models

N_{core}/Topology	64 tiles mesh network
N_{thread} for each trace	8
Router	5stage on-chip router
Memory scheduling algorithm	FRFCFS [14]
2D DRAM configuration	DDR3 1600MHz
3D DRAM configuration	timing parameters scaled 3X $bus_{speed} = core_{speed}$
Core Speed	2GHz

TABLE II
Simulation parameters

Figure 10 shows the simulated system models for both 2D and 3D with cores on the bottom die. The simulations were carried out on our in house cycle level network and memory simulator with traces extracted from a cycle level micro-architectural simulator [15]. The trace was captured from a single core running 8 different threads, and then replicated on each of the tiles. The traces were captured at the back of last level cache on a micro-architectural simulator with varying cache sizes. For the 2D system there are 8 memory controllers placed as shown in figure 10. Other simulation parameters are given in Table II.

As predicted by the analytical model the 3D system showed high tolerance to smaller caches in comparison to the 2D system. The traces drove injection of memory packets into the network at a controlled injection rate using miss-status holding registers (MSHR). The results closely tracked the trends from the model with 3D delivering high performance even at high injection rates. Figure 11 shows the average memory latency for 2D and 3D in a simulation similar to the analysis in section III-C. From simulation we observed that the

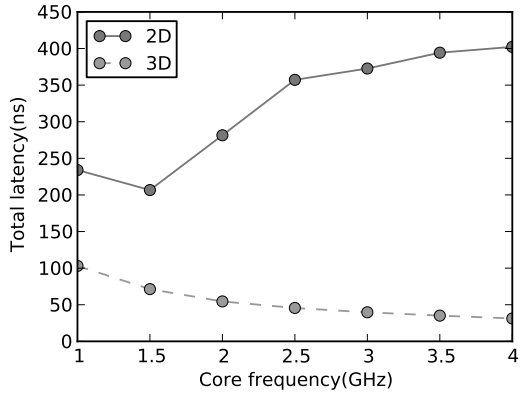


Fig. 11. Average memory latency for applications on a 3D and 2D system when the core, cache and network memory is scaled to evaluate the stress on memory bandwidth.

average round trip latency per memory request dropped from 582 (291ns) cycles for the 2D organization to 109 (54.5ns) cycles for the 3D system. This was mainly due to reduced memory latencies. The average memory latency across all memory requests was 501 cycles and 31 cycles in 2D and 3D respectively. In addition to the reduced memory latency, we observed very low buffer occupancies (on average 16% and 1% for 2D and 3D) in the memory controller queues for 3D. For 3D we noticed low row buffer hit rate and high thread interference at the individual memory controllers (90%). This suggests that the MC for the small banked DRAM can be highly simplified as the scope for the scheduling algorithms is negligible. Further. a closed page DRAM policy may be more suited for energy efficiency.

In order to study the relationship between the interdie and intradie bandwidth we conducted experiments with the individual address space of a core distributed across various DRAM banks in 3D. We refer to memory bank directly above a core as the local memory bank and all other memory banks as remote memory banks. For least stress on the intradie network with very low network latency, the address space of a core should be mapped to its local bank. Similarly the average network latency of a request is the worst when its address space is mapped on remote memory banks. Figure 12 shows the average latency per memory access for different degrees of locality. In the system model a remote memory request will first traverse the intradie network and then the TSV to the memory bank. Even at average locality it can be seen that about 50% of the traffic uses the intra-die network, with latencies 40% higher than the best case latency. This suggests that innovative 3D networks that can stress the TSV bandwidth bypassing the intra-die network may lead to improved performance.

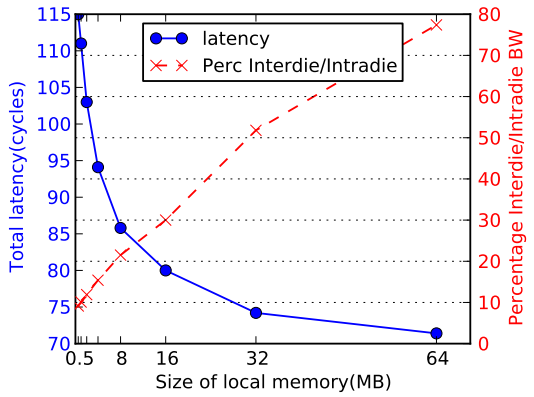


Fig. 12. Average memory latency for varying network traffic (Note: total system DRAM 16 GB).

V. Concluding Remarks

While 3D integration permits for memory to be stacked on logic with increased memory bandwidth via TSV's, the limits of performance gains and effective utilization of TSV bandwidth are still unclear. We develop an analytical model to evaluate these gains. Future work will be to improve on the models, based on the thermal impacts of 3D and current trends in interconnect scaling for the intradie layer.

References

[1] G. H. Loh, "Extending the effectiveness of 3d-stacked dram caches with an adaptive multi-queue policy."

[2] K. Bergman, S. Borkar, D. Campbell, W. Carlson, W. Dally, M. Denneau, P. Franzon, W. Harrod, J. Hiller, S. Karp, S. Keckler, D. Klein, R. Lucas, M. Richards, A. Scarpelli, S. Scott, A. Snavely, T. Sterling, R. S. Williams, K. Yelick, K. Bergman, S. Borkar, D. Campbell, W. Carlson, W. Dally, M. Denneau, P. Franzon, W. Harrod, J. Hiller, S. Keckler, D. Klein, P. Kogge, R. S. Williams, and K. Yelick, "Exascale computing study: Technology challenges in achieving exascale systems peter kogge, editor and study lead," 2008.

[3] Z. Guz, E. Bolotin, I. Keidar, A. Kolodny, A. Mendelson, and U. Weiser, "Many-core vs. many-thread machines: Stay away from the valley," *Computer Architecture Letters*, vol. 8, no. 1, pp. 25 –28, 2009.

[4] J. Duato, S. Yalamanchili, and N. Lionel, *Interconnection Networks: An Engineering Approach*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2002.

[5] T. Givargis and F. Vahid, "Interface exploration for reduced power in core-based systems," in *Proceedings of the 11th international symposium on System synthesis*, ser. ISSS '98. Washington, DC, USA: IEEE Computer Society, 1998, pp. 117–122. [Online]. Available: http://portal.acm.org/citation.cfm?id=293016.293040

[6] A. N. Udipi, N. Muralimanohar, N. Chatterjee, R. Balasubramonian, A. Davis, and N. P. Jouppi, "Rethinking dram design and organization for energy-constrained multi-cores," in *Proceedings of the 37th annual international symposium on Computer architecture*, ser. ISCA '10. New York, NY, USA: ACM, 2010, pp. 175–186.

[7] J. L. Hennessy and D. A. Patterson, *Computer architecture*. Amsterdam [u.a.]: Kaufmann [u.a.], 2007.

[8] S. Li, J. H. Ahn, R. Strong, J. Brockman, D. Tullsen, and N. Jouppi, "Mcpat: An integrated power, area, and timing modeling framework for multicore and manycore architectures," in *Microarchitecture, 2009. MICRO-42. 42nd Annual IEEE/ACM International Symposium on*, 2009, pp. 469 –480.

[9] G. H. Loh, "3d-stacked memory architectures for multi-core processors," in *In International Symposium on Computer Architecture*.

[10] T. Kgil, A. Saidi, N. Binkert, S. Reinhardt, K. Flautner, and T. Mudge, "Picoserver: Using 3d stacking technology to build energy efficient servers," *J. Emerg. Technol. Comput. Syst.*, vol. 4, pp. 16:1–16:34, November 2008.

[11] D. H. Woo, N. H. Seong, D. Lewis, and H.-H. Lee, "An optimized 3d-stacked memory architecture by exploiting excessive, high-density tsv bandwidth," in *High Performance Computer Architecture (HPCA), 2010 IEEE 16th International Symposium on*, 2010, pp. 1 –12.

[12] W. Chou, "Queueing systems, volume ii: Computer applications–leonard kleinrock," *Communications, IEEE Transactions on*, vol. 25, no. 1, pp. 180 – 180, Jan. 1977.

[13] A. Kahng, B. Li, L.-S. Peh, and K. Samadi, "Orion 2.0: A fast and accurate noc power and area model for early-stage design space exploration," in *Design, Automation Test in Europe Conference Exhibition, 2009. DATE '09.*, 2009, pp. 423 –428.

[14] S. Rixner, W. J. Dally, U. J. Kapasi, P. Mattson, and J. D. Owens, "Memory access scheduling," in *Proceedings of the 27th annual international symposium on Computer architecture*, ser. ISCA '00. New York, NY, USA: ACM, 2000, pp. 128–138.

[15] G. H. Loh, S. Subramaniam, and Y. Xie, "Zesto: A cycle-level simulator for highly detailed microarchitecture exploration," in *In Proc. of the Int. Symp. on Performance Analysis of Systems and Software*, 2009.