Research Lab News

Minimizing Data Movement in Multicore Systems

by Daniel Sanchez, Assistant Professor, Computer Science and Artificial Intelligence Lab

Technology trends are drastically changing the way we build computer systems. While Moore's Law still provides an increasing amount of transistors per chip, transistor speed and energy efficiency are barely improving. To improve performance within a limited power budget, systems across all domains, from cellphones to supercomputers, are becoming more parallel, featuring an increasing amount of simpler and more efficient cores. But as computation becomes more efficient, systems face the fundamental costs of data movement. Memory accesses and communication have become orders of magnitude more expensive than basic compute operations. Yet current architectures still use techniques and abstractions designed decades ago, when computation was expensive and data movement was cheap, so they are organized in a way that causes more data movement than needed. To overcome this challenge, my students and I are investigating *data-centric parallel architectures* that seek to minimize data movement as a primary design objective. Excessive data movement often stems from a disconnect between hardware and software, so we are taking a cross-layer approach that combines the strengths of hardware and software techniques to achieve gains that neither hardware-only nor software-only approaches can provide.

Current systems with few cores rely on rigid, hardware-managed memory hierarchies to reduce data movement. For example, Figure 1 shows the four-level memory hierarchy of a recent 8-core processor, including the latency and energy of accesses to each level. Each level provides a larger amount of slower and cheaper storage that is more expensive to access. Moreover, all levels but the last one are hardware *caches*, associative memories that transparently retain recently-accessed data. Memory accesses traverse all the levels until they find the data or reach (non-associative) main memory. Because main memory accesses are so expensive, current multicores devote about half of chip area to caches. Most of this cache space is coalesced in a monolithic last-level cache shared among all cores. While this organization works reasonably well for systems with few cores, it scales poorly.

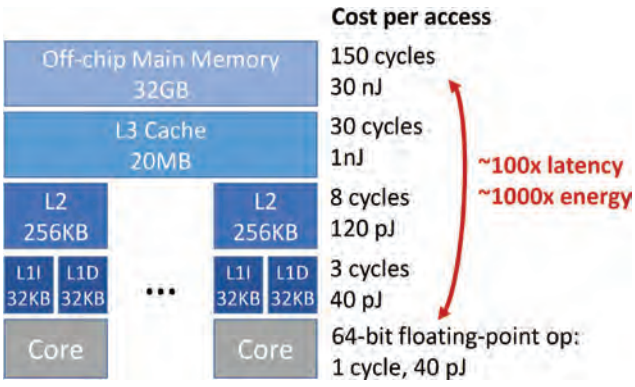


Figure 1: Current chips with few cores use hardware-managed hierarchies to reduce the cost of memory accesses. This figure shows the sizes, latency per access, and energy per access of an 8-core Intel E5-2670 chip running at 2.6 GHz.

As the number of cores grows, it is far more efficient to distribute cache capacity across the chip. Figure 2 (next page) illustrates this organization, showing a 64-tile chip, where each tile has one core and a bank of the last-level cache. Each core has access to a small amount of capacity nearby and a larger amount of capacity further away. To reduce data movement, it is crucial that most accesses are served by nearby banks.

Most prior work has approached this problem by developing hardware techniques that adaptively place data close to the cores that use it. But hardware-only techniques suffer from two key shortcomings. First, software is often in a better position than hardware to place data, as it has better semantic knowledge about data usage. For example, the operating system knows what regions of memory are used by each thread. Second, how the last-level cache is managed has a large impact on the performance of the different threads and processes sharing the chip, and involves making tradeoffs, e.g., speeding up some threads at the expense of making others slower. Software should be able to control these tradeoffs to align them with system-level objectives, such as prioritizing critical applications over background processes.

To tackle the limitations of hardware-only approaches, we have designed Jigsaw [1], a distributed cache organization that gives software full control over the cache efficiently. First, software classifies memory regions into logical partitions, or shares. In our implementation, the operating system performs this classification, so Jigsaw operates transparently to applications. The OS maps thread-private data to per-thread shares, data shared by threads within a process to per-process shares, and data shared by multiple processes (e.g., OS code and data) to a single global share. This coarse-grain classification captures the semantic information that the OS has about memory usage. Shares could also be exposed to individual programs to capture application-level knowledge. Then, Jigsaw lets software divide each cache bank in multiple partitions, and combine multiple bank partitions to form virtual caches, each of which caches data from a specific share. Figure 3 (next page) shows an example division of banks into virtual caches.

Periodically, an OS runtime reconfigures the size and location of each virtual cache to minimize both expensive main memory accesses and on-chip traffic. To guide reconfigurations, cheap hardware monitors sample a small fraction of accesses to estimate the miss curve of each virtual cache. Miss curves capture how many off-chip accesses each virtual cache would

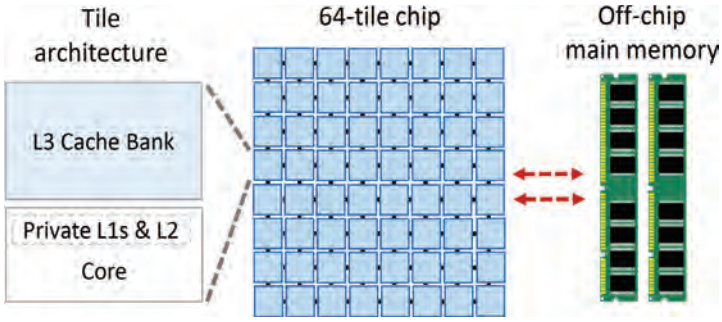


Figure 2: A tiled 64-core chip with a distributed cache hierarchy.

incur at each possible size. Miss curves make it easy for software to perform predictive optimization, finding the right size and placement of each virtual cache without trial and error. Efficient optimization algorithms allow software to reconfigure the cache every few milliseconds, quickly adapting to application changes. At 64 cores, Jigsaw improves performance over a conventional architecture by 38% on average, and reduces energy consumption by 34%. Jigsaw achieves these gains because it reduces both off-chip accesses to main memory (by 23%) and on-chip data movement (by 7x) [3]. Jigsaw yields larger improvements as the number of cores grows.

While Jigsaw seeks to reduce data movement by placing data close to the threads that use it, how threads are laid out across the chip greatly affects how well this can be done. For example, if two threads that need a lot of capacity to work well are running in nearby cores, they will contend for capacity at nearby banks and will be forced to place data further away. To solve this problem, we have developed computation and data co-scheduling (CDCS), a technique that jointly places threads and their data to further reduce data movement [3]. CDCS frees programmers and operating systems from managing thread placement, avoids the pathological behavior of fixed policies, and improves performance further, by 46% on average at 64 cores. The key challenge in CDCS is to find high-quality solutions with low overheads. Simultaneously placing computation and data is much more complex than placing data alone. The optimal solution is NP-hard, and the problem

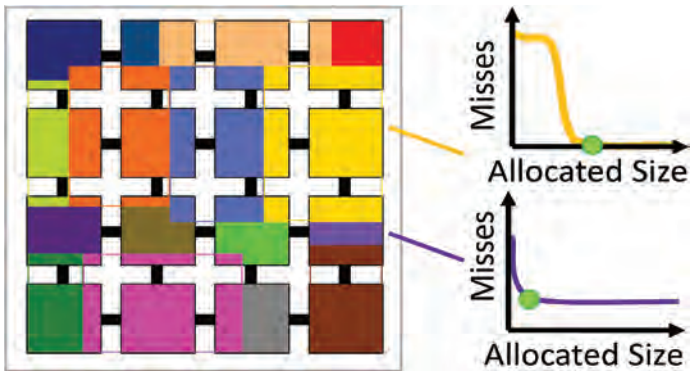


Figure 3: Jigsaw gangs physically distributed cache banks into virtual caches, which are sized to minimize cache misses and placed close to the threads that use them.

is similar in structure to VLSI place-and-route, where solvers use algorithms that are impractically expensive. Instead, we have designed a fast optimization algorithm that achieves within 1% of the performance of these expensive solvers, and runs in about one millisecond. This allows CDCS to continuously monitor and reconfigure the system with low overheads.

A constant challenge in our work is to make hardware predictable and easy to analyze, so that software runtimes can manage it efficiently and programmers can easily understand its performance. Analyzability becomes more crucial as systems become more heterogeneous and complex. Unfortunately, the conventional wisdom is that one needs to sacrifice analyzability for performance and efficiency. For example, in the past, caches implemented the least-recently-used (LRU) policy, which, on a miss, replaces the datum that was used furthest in the past. LRU is simple and predictable, but has common pathologies that cause poor performance. As a result, current chips use adaptive, empirically-designed policies that address LRU's most common pathologies. However, these policies sacrifice LRU's predictability, precluding software management. In Jigsaw and CDCS, we used LRU to allow software management, trading off efficiency for predictability. More recently, we have shown that no such tradeoff is needed. We have designed Talus [2] , a technique that fixes performance pathologies and enables caches to yield smooth and predictable performance gains with additional space. Talus not only bridges the gap between LRU and high-performance policies, but also guarantees convex performance gains with additional space, allowing software to use much simpler and efficient convex optimization methods to manage them.

Though the techniques we have developed so far allow for more scalable and efficient systems, many challenges and opportunities to reduce data movement remain unexplored. While the architecture of chips in the far future is still an open question, achieving substantial efficiency gains will require further innovation across the stack, from reconfigurable memory systems that can accommodate diverse combinations of heterogeneous memory technologies, to new data-centric programming models that allow programmers to easily convey locality and parallelism while shielding them from unnecessary complexity. Addressing these challenges is the focus of our ongoing work.

References

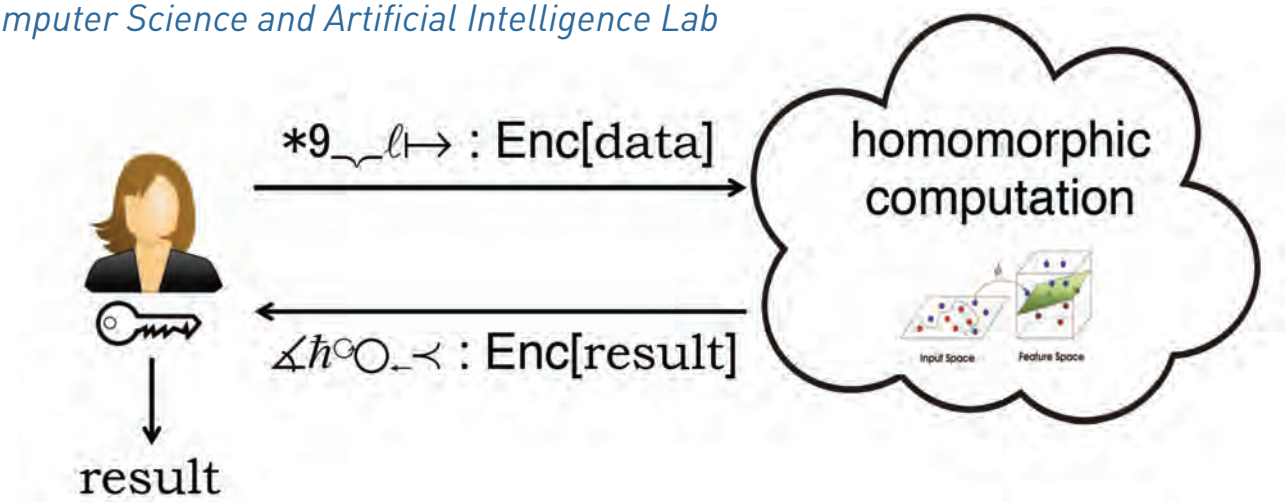
[1] N. Beckmann and D. Sanchez. Jigsaw: Scalable Software-Defined Caches. In Proc. PACT-22, 2013.

[2] N. Beckmann and D. Sanchez. Talus: A Simple Way to Remove Cliffs in Cache Performance. In Proc. HPCA-21, 2015.

[3] N. Beckmann, P.-A. Tsai, and D. Sanchez. Scaling Distributed Cache Hierarchies through Computation and Data Co-Scheduling. In Proc. HPCA-21, 2015.

Computation Obscura

by Vinod Vaikuntanathan, Steven and Renée Finn Career Development Assistant Professor, Computer Science and Artificial Intelligence Lab



We live in a world of big data and constant communication. We exchange sensitive personal information through the internet, e-mails and phone calls. Individuals and organizations store and process this information on third party cloud services. All these transactions constitute an incredible treasure-trove of high-value data that malicious hackers, organizations and even powerful state actors could target and profit from. Against this backdrop, the need for encrypting our data seems like a no-brainer. Thanks to modern cryptography, we now have sophisticated algorithms and protocols that enable strong encryption and authentication.

Encrypting data is often compared to locking it inside an opaque box. Anyone with a key can unlock the box and "see" the data inside, but without the key, the box is completely opaque and perfectly immutable. Indeed this is quite an apt analogy for the use of encryption in secure communications and secure storage of data. But just as the analogy suggests, encryption is an all-or-nothing primitive: encrypted data betrays no information and is completely useless until it is decrypted. This is exactly what we want. Or, is it?

The new world of cloud computing requires us to adopt a more nuanced view of encryption, where privacy has to co-exist with usefulness. Not only do we store data on the cloud, we also perform computations on it, without transporting it back to our local machine. We are faced with what seems like an impossible fantasy: how can the cloud compute on encrypted data without decrypting it and without knowledge of the secret key?

The answer to this conundrum lies in a suite of cryptographic techniques collectively referred to as Computation Obscura. This includes fully homomorphic encryption, secure multiparty computation and functional encryption, techniques that help us achieve a fine balance between privacy and usefulness.

Homomorphic encryption is a special type of encryption system that allows us to perform computations on encrypted data without decrypting it. With homomorphic encryption, the cloud can store encrypted data and process it without ever "seeing" the data, the intermediate results of the computation, or even the output. A *fully* homomorphic encryption system is one that supports any computation, however complex, on encrypted data. This notion was first formulated by Rivest, Adleman and Dertouzos [1] in 1978, but a construction eluded cryptographers until Gentry's work [2] in 2009.

The security of encryption systems is always based on unproven mathematical assumptions, such as the hardness of factoring large composite numbers. Initial constructions of fully homomorphic encryption, starting with the work of Gentry, were based on multiple new and untested cryptographic assumptions, which made their ultimate security questionable. Furthermore, the encryption schemes were astronomically inefficient. Private keys and ciphertexts in the encryption system were many gigabytes long, making it difficult even to store them in memory. Computations on encrypted data suffered enormous loss in efficiency, a factor of 10^{14} slow-down compared to computing on plaintext data.

During the last few years, we have invented new mathematical constructions of fully homomorphic encryption that perform several orders of magnitude better, and are based on standard, well-studied cryptographic assumptions. In particular our encryption system, invented together with Brakerski and Gentry [3, 4], is able to perform arbitrary computations on encrypted data with slowdown

factors of 10^5 to 10^6, and special-purpose computations much faster. The system is a cornerstone of a large DARPA project aimed at building practical systems that compute on encrypted data. In the span of five years, homomorphic encryption technology has gone from being a distant dream to the point where it now has the potential to be practical.

To give a sense of how such encryption systems work, let us describe a very rough outline of the ideas underlying the construction. The starting point is to observe that complex computations can be broken down into simple units. Our units of computation will be addition and multiplication of numbers. The next item on the agenda is a way to encrypt numbers. There are many ways to do this, but here is a toy version of the system. Our private key will be a large prime number P (think thousands of digits). The encryption of a number M is simply PQ + M, where Q is also a very large number. Decrypting a ciphertext is easy: simply reduce the ciphertext modulo P. I will leave it to my mathematically enlightened reader to observe that operations on ciphertexts mirror operations on encrypted numbers. Adding two ciphertexts PQ_1+M_1 and PQ_2+M_2 adds the underlying numbers M_1 and M_2, and multiplying the ciphertexts multiplies M_1 and M_2.

Homomorphic encryption is only the beginning of the road. It is but one tool in a growing cryptographic toolkit that allows us to extract utility from data while preserving its privacy. Secure multiparty computation [5, 6], a technique from the 1980s, offers a way for multiple data owners to collaborate and compute a function on the aggregation of their data sets, without revealing their individual data. Although large data owners currently shy away from such collective computation, an efficient secure multiparty computation platform will enable them to collaborate while alleviating their concerns about privacy. A functional encryption system [7, 8] gives us expressive access control of encrypted data, allowing us to encrypt in such a way that we can reveal carefully chosen functions of the data to people with the right credentials.

In order to realize the tremendous potential of these technologies, we are actively investigating ways to make them faster and more efficient. Rather than focus on general purpose computations, our goal is to develop techniques for specific, useful classes of computations in areas such as statistics and machine learning.

Nowadays, one often hears about the tension between privacy and functionality, and between privacy and security. The implicit assumption in such assertions is that privacy of data is antithetical to deriving any usefulness out of it. Modern cryptographic technologies such as homomorphic encryption, secure multiparty computation and functional encryption challenge such notions and demonstrate that, in many scenarios, the dichotomy between privacy and functionality is a false one. Sometimes, it appears, we can eat our cake and have it too.

References

[1] Ronald Rivest, Leonard Adleman, and Michael Dertouzos. On Data Banks and Privacy Homomorphisms, In Foundations of Secure Computation, New York: Academic Press, 1978, pp. 169-180.

[2] Craig Gentry. Fully Homomorphic Encryption using Ideal Lattices, In ACM Symposium on Theory of Computing, 2009, pp. 169-178.

[3] Zvika Brakerski and Vinod Vaikuntanathan. Efficient Fully Homomorphic Encryption from (Standard) LWE, In IEEE Annual Symposium on Foundations of Computer Science, 2011, pp. 97-106.

[4] Zvika Brakerski, Craig Gentry and Vinod Vaikuntanathan. (Leveled) Fully Homomorphic Encryption without Bootstrapping, In Innovations in Theoretical Computer Science, 2012, pp. 209-325.

[5] Oded Goldreich, Silvio Micali and Avi Wigderson. How to Play any Mental Game or A Completeness Theorem for Protocols with Honest Majority, In ACM Symposium on Theory of Computing, 1987, pp. 218-229.

[6] Michael Ben-Or, Shafi Goldwasser and Avi Wigderson. Completeness Theorems for Non-Cryptographic Fault-tolerant Distributed Computation, In ACM Symposium on Theory of Computing, 1988, pp. 1-10.

[7] Dan Boneh, Amit Sahai and Brent Waters. Functional Encryption: A New Vision for Public-key Cryptography, In Communications of the ACM, 2012, pp. 56-64.

[8] Shafi Goldwasser, Yael Tauman Kalai, Raluca Ada Popa, Vinod Vaikuntanathan and Nickolai Zeldovich. Reusable Garbled Circuits and Succinct Functional Encryption, In ACM Symposium on Theory of Computing, 2013, pp. 555-564.

Systemic Risk and Networks

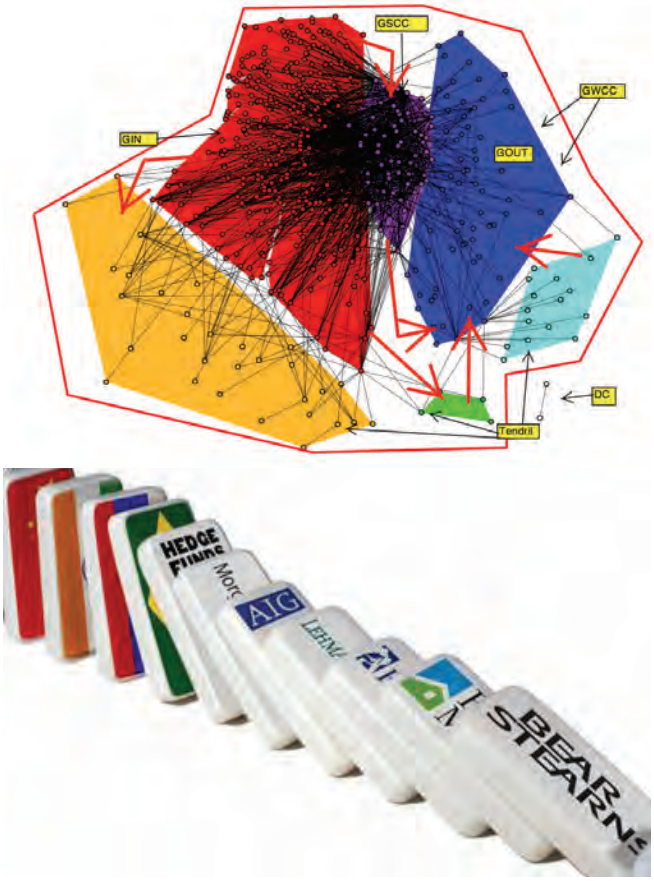
by Asuman Ozdaglar, Professor, Director, Laboratory for Information and Decision Systems

On September 15, 2008, the investment bank Lehman Brothers filed for bankruptcy. What made this event cataclysmic for global financial markets was not that it was the largest bankruptcy filing in the United States, but that Lehman itself was yet one more domino in a financial crisis that had started more than a year earlier, with accumulating losses from subprime loans. Lehman's collapse immediately created financial distress for several large financial institutions that were its counterparties. The next financial institution to come to the brink of bankruptcy was the American Insurance Group (AIG), whose collapse would have meant its inability to pay its counterparties for the credit default swap arrangement it had made. These counterparties included, among others, Goldman Sachs and Deutsche Bank. Federal Reserve and Treasury officials, convinced that AIG's collapse would bring down scores of other financial institutions, intervened and bailed out AIG to stem the rapidly spreading financial contagion.

Of the many lessons learned by policymakers and academics from the turbulent weeks surrounding these events, the most important one is the danger of financial contagion, which can amplify small shocks into systemic risk and even a financial tsunami. But the lesson is only partial. More than six years after these momentous events, there is still a limited understanding of how financial contagion is created and what structures of financial interconnections ("network architecture") paves the way for systemic risk. In fact, many claims in the literature are contradictory. Some maintain, for example, that it is the sparseness of financial connections and the lack of diversified liabilities structures that underpin systemic risk because with more densely-connected financial networks, a negative shock to a bank would be spread across several counterparties, rather than one or two as in a sparse network, making contagion less likely. Yet others take the exact opposite perspective and blame the denseness of financial linkages for the spread of risks and contagion of failure in financial markets based on the argument that a negative shock to a bank can infect many more when this bank has many counterparties.

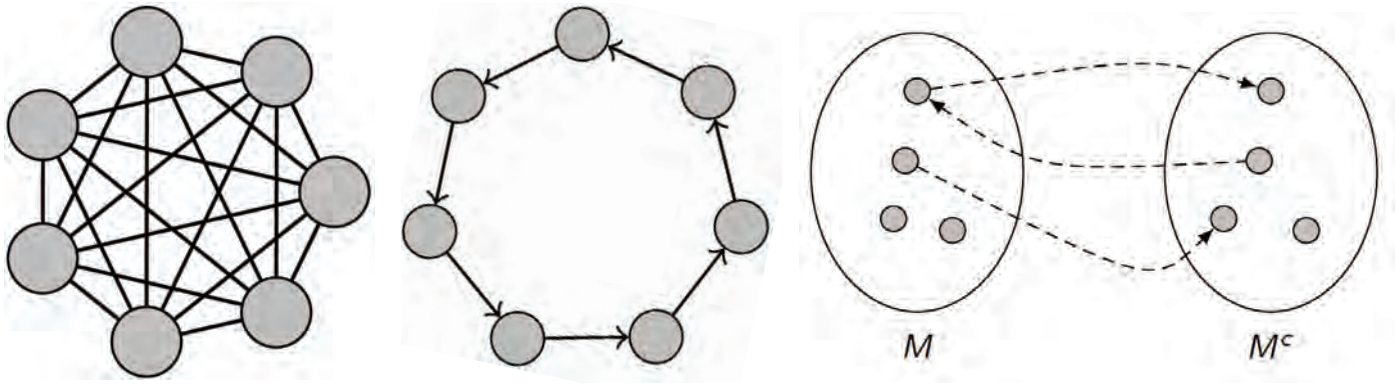
How can we make sense of these conflicting claims? What are the structural properties of financial networks that create systemic risk? Which financial institutions are systemically important and play an oversized role in financial contagion?

Recent research by Asu Ozdaglar and Daron Acemoglu (from MIT's Laboratory for Information and Decision Systems (LIDS), and Department of Economics, respectively) and their collaborator Alireza Tahbaz-Salehi (from the Business School at Columbia University) sheds light on these questions. This work considers an interconnected financial network in which each bank simultaneously lends and borrows from other banks and also has real assets (such as loans to firms and consumers) with stochastic returns. A bank makes the specified payments on its liabilities (the debts that it has taken on) as long as it can. When its income from its real assets combined with the payments it receives from loans to other financial institutions fall short of its obligations, however, that bank is forced to default (fully or partially). But once a bank defaults, this creates hardship on other banks expecting payments from it, creating the first step in a chain of dominoes — as was the concern in the fall of 2008 with the failure of AIG to make its payments to financial institutions such as Goldman Sachs and Deutsche Bank.



The network of loans among financial institutions (source: Bech and Atalay, 2008). The connections among financial institutions raise the possibility of cascades, where shocks to some units propagate, creating systemic risk (source: http://intermarket-andmore.finanza.com/banche-usa-rischi-enormi-orizzonte-per-il-2011-21648.html).

The central message that emerges from this analysis is that the nature of financial contagion together with the structural



Which network structures minimize systemic risk?

properties of networks that make them susceptible to systemic risk, depend heavily on the magnitude of shocks hitting individual banks. When these shocks are small (for example, the assets of only one or a few banks are hit by relatively small shocks), diversification concerns are key, and thus more densely-connected financial networks are more stable and less prone to systemic risk. In contrast, financial networks that are sparse, such as rings, are the least stable ones. For example with every network, even a small negative shock to a single bank can start a chain reaction as this bank's default on its liabilities falls on the shoulders of its single creditor, and its single creditor's subsequent default then spreads to another bank and so on.

However, as the size of negative shock to some real assets exceeds a shock threshold, there is a phase transition, and the nature of systemic risk is transformed. In such large shock regimes, it is now densely-connected financial networks that are prone to contagion. The complete network, where each bank's liabilities are equally distributed across all other banks is the least stable one, because this large shock to a single bank now creates hardship to all of its creditors, bringing the entire banking system to its knees. In contrast, financial networks that create islands of weakly connected components are much more resilient against such large shocks. The reason why there is such a complete turnaround in what types of networks underpin systemic risk is related to the fact that densely-connected networks do not have a way of absorbing negative shocks by shifting some of it to senior creditors and thus make the entire shock transmit to other banks.

These insights also help us to understand the claim made by the deputy governor of the Bank of England, Andrew Haldane who suggested that highly interconnected financial networks may be "robust-yet-fragile" and that they "exhibit a knife-edge or tipping point property", in the sense that "within a certain range, connections serve as shock-absorbers [and] connectivity engenders robustness." However, beyond a certain range, inter-connections start to serve as a mechanism for propagation of shocks, "the system [flips] the wrong side of the knife-edge," and fragility prevails. The pattern of financial contagion has this robust-yet-fragile feature emphasized by Haldane. Financial interconnections create stability in response to small shocks but become powerful dominoes when shocks are large.

Another important consequence of this analysis concerns the identification of systemically important banks, which has become a key regulatory concern since the financial crisis. Many practitioners and regulators have relied on applications of standard notions of network centrality, such as degree centrality or various eigenvector centrality measures including Bonacich centrality, for identifying such systemically important financial institutions. But many of these centrality measures are derived from models that have little to do with how financial contagion takes place. The micro-founded model of financial contagion teaches two key lessons about systemic importance. The first is that no unambiguous notion of systemic importance can be derived as witnessed by the fact that which institutions and which financial networks facilitate the chain reaction crucially depends on whether shocks are large or small (whether we are on one or the other side of the phase transition). The second is that even within a regime, different notions of centrality reflecting the exact nature of economic relations arise as the appropriate measures of systemic importance.

This research is obviously not the last word on contagion in financial networks. It nevertheless highlights how careful analytical modeling of economic relations in a networked setting can both pave the way to a comprehensive study of systemic risk and bring new insights on which structural properties of financial networks and which types of financial institutions might create future faultlines.

Towards Terahertz Integrated Systems On Chip

by Ruonan Han, Emanuel E. Landsman (1958) Career Development Assistant Professor, Microsystems Technology Laboratories

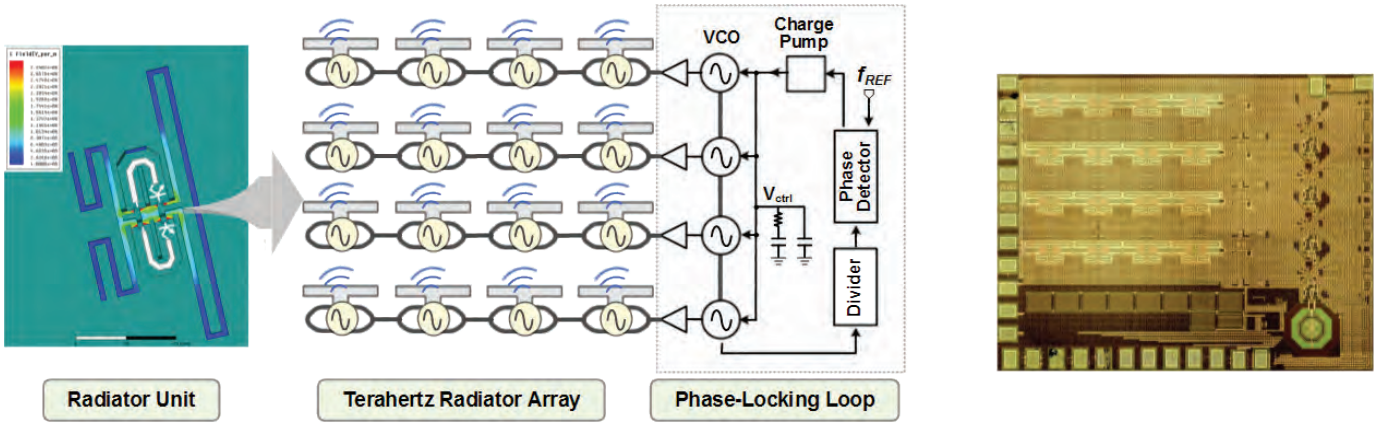
Terahertz frequency, broadly defined from 100 GHz to 10 THz, is an electromagnetic spectrum between microwave and infrared. The radiation in this frequency range has great potential in the applications of biomedical diagnosis, security screening, as well as high-speed communications. For instance, terahertz wave can propagate through non-metallic, non-polar materials with small attenuation. This property, combined with the small wavelength (compared to microwave) and low photon energy (compared to X-ray), makes terahertz wave an ideal option for non-ionizing medical imaging, such as burn injury assessment and skin cancer detection. Utilizing the molecular resonance in this frequency range, terahertz spectroscopy can help us identify hazardous gas (e.g methylchloride) and warfare chemical agents (e.g. sarin) in a remote distance. It is also expected that wireless/wired data link operating at such broad, unallocated band will boost the transmission speed significantly and resolve the spectral congestion issues nowadays. As an example, using a 240-GHz carrier wave, researchers have demonstrated a 30-Gbps link over 40-m distance [J. Antes, et al, IMS 2013].

So why didn't we exploit this promising spectrum earlier? The main technical constraint is depicted as a "Terahertz Gap": terahertz frequency is too high for electronics mainly due to the limited carrier velocity and breakdown voltage of devices; meanwhile, it is too low for photonics due to the increased loss and the lack of materials with sufficiently small bandgap. As a result, the generated power level and signal detection sensitivity are the poorest among the entire electromagnetic spectrum. To address this issue, significant efforts and progress have been made in both electronics (e.g. devices based on high-mobility III-V semiconductors and vacuum tubes) and photonics (e.g. quantum-cascade lasers and photoconductive switches). However, these solutions are normally too bulky and costly, and lack of decent systematic integration capability. Some also require stringent operational conditions such as cryogenic cooling, which severely limit their applications.

The Terahertz Integrated Electronics research group, led by Prof. Ruonan Han, is focusing on filling such Terahertz Gap using integrated circuit technologies. In the past, we mostly relied on the most accessible platform in the semiconductor industry: silicon CMOS. Although silicon has an inferior speed property compared to many III-V compound semiconductor materials, the performance of silicon transistors has been improved steadily thanks to Moore's Law of technology scaling. Now the cutoff frequency of the main-stream CMOS technologies has reached 300 GHz, which makes the terahertz operation possible. Without a doubt, CMOS will drastically reduce the cost and size of current THz systems. Meanwhile, on the same die, terahertz components can be built with other analog/digital circuitries, enabling unprecedented levels of integration and flexibility. This will trigger tremendous opportunities in the portable equipment market, such as in-vivo tooth cavity detection and handheld breath analyzer for disease diagnostics.

On the other hand, however, we are facing great challenges in the design of terahertz integrated circuits. When a transistor operates near its cutoff frequency, the activity of the device becomes very weak. In order to achieve the maximum gain and to extract the highest power from the device, several optimum conditions have to be met simultaneously. This problem is exacerbated by the passive components and interconnects (such as resonant cavity and transmission lines) that are very lossy in the terahertz range. Lastly, to break the speed limits set by the cutoff frequency, we commonly resort to harmonic-signal generation (namely we distort the waveform as much as possible). The nonlinear analysis and optimization involved in this process further complicate the circuit design.

Figure 1. A terahertz radiator array with integrated phase-locking loop. Each unit inside the array enables maximum oscillation, optimum harmonic generation, as well as efficient on-chip radiation [R. Han, et al, ISSCC 2015].



Due to these challenges, conventional design topologies are severely under-optimized. This is particularly evident in the implementations of terahertz signal source, which is undoubtedly the most critical component inside each terahertz system. When the CMOS harmonic oscillators operating in terahertz range were first reported in 2008 [E. Seok, et al, ISSCC 2008][D. Huang, et al, ISSCC 2008], the achieved power was only a couple of nanowatts.

Over the past few years, significant improvement has been made. For example, we proposed a co-design approach that involves synergistic innovations in device, circuits, electromagnetics, and system architecture. One prototype of such approach is shown in Fig. 1 (page 32). In collaboration with Prof. E. Afshari's group at Cornell and with STMicroelectronics, we invented a very compact 320-GHz harmonic oscillator structure, which utilizes multi-mode propagation inside several specially engineered metal slots. By exciting proper traveling-wave patterns inside the feedback path of the transistors, the oscillation power of the circuit is maximized. Meanwhile, this structure also filters harmonic signals in the way that the oscillation waveform is highly distorted. Another interesting feature of this design is that, without any explicit antenna, the 2nd-harmonic signal at 320 GHz can be directly radiated into free space with a high efficiency. This leads to a very small footprint of the radiator. We are able to integrate sixteen of such radiator units within a 1-mm_2 chip area. The power from each radiator is then coherently added in the far field. Through such "quasi-optical" combining, the output radiation is highly directive and has a total power of 3.3 mW. This is so far the highest output power among all silicon terahertz sources, as indicated in Fig. 2 (upper right) (a). With the future development of the more advanced CMOS devices and processing technologies, we will be able to obtain higher power and higher output frequency.

Another critical merit of the signal sources is the DC-to-THz conversion efficiency. This is particularly important for energy-aware applications. It can be seen from Fig. 2 (b) that, over the past few years, our research community has been able to increase the efficiency by four decades. Currently, our work has demonstrated a record efficiency near 1%, and we aim to keep such momentum for the upcoming years. Lastly, we also demonstrated several signal-processing functionalities required in practical terahertz transmitters such as phase-locking capability (the first time for terahertz radiators in silicon), ultra-narrow-pulse modulation, and broadband frequency doubling.

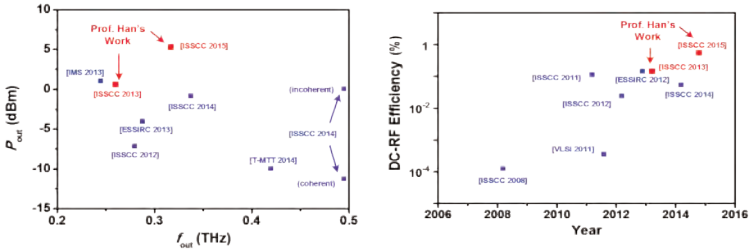


Figure 2. (a) The output power and (b) DC-to-RF efficiencies of the state-of-the-art terahertz radiators based on silicon integrated circuits.

To pair with the signal sources, a sensitive receiver is indispensable. Generally, the photon energy of terahertz wave is too small to directly excite carriers across the bandgap of semiconductor materials. In integrated circuits, our approach of detecting terahertz wave is to couple the radiation into nonlinear devices via an on-chip antenna, and then convert the power into a DC signal using the self-mixing operation inside the device. It is therefore essential to develop devices that have high speed and low noise. On a standard 130-nm digital CMOS process, a Polysilicon-Gate-Separated Schottky barrier diode (PGS SBD) was reported with a measured cutoff frequency of 2 THz [S. Sankaran, et al, ISSCC 2009]. Based on such device, we (in collaboration with Prof. Kenneth O's group at UT-Dallas) have demonstrated highly sensitive detectors and imaging arrays from 280 GHz to 860 GHz. Shown in Fig. 3, we are also able to construct high-resolution terahertz images to see through many objects and to monitor the hydration level of plants. These provide straightforward evidences for prospective applications of the silicon chips we developed. Currently, we are working towards even higher sensitivity and spatial resolution using heterodyne sensing and beam forming technologies.

Our research has demonstrated a feasible path towards future terahertz microsystems, leading to better understanding and fundamental speed limits of integrated electronics. We also endeavor to extend these technologies into other non-silicon devices, such as gallium nitride high electron mobility transistors (GaN HEMTs). Meanwhile, to form high-performance larger scale systems, we are pursuing a holistic solution to integrate the terahertz building blocks that we have developed. In the long run, these on-chip systems are expected to revolutionize the electronic infrastructures for communication, biomedicine, and sensing.

Figure 3. Using a terahertz CMOS image sensor based on Schottky diode, we performed active imaging for various objects [R. Han, et al, ISSCC 2013].

Next Generation Video Coding: more pixels, fewer bits, less watts

by Vivienne Sze, Emanuel E. Landsman (1958) Career Development Assistant Professor, Research Laboratory of Electronics

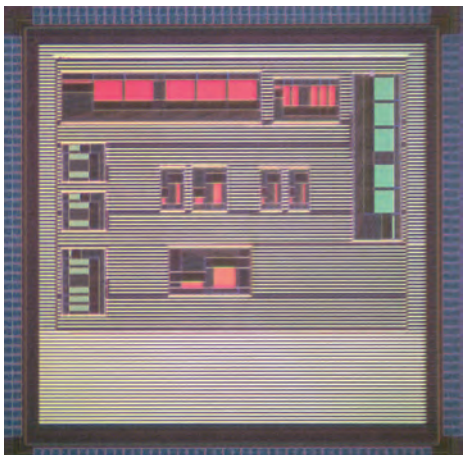
Over 60% of the bits that flow through the Internet today are used to transport video. This can be attributed to the growing popularity of applications such as video streaming, video conferencing and video surveillance. In addition, the amount of video content being generated is staggering: over 100 hours of video are uploaded to YouTube every minute; and over 400 petabytes of data, equivalent to 92 million DVDs, are collected from security cameras every day. The demand to transmit and store video continues to grow exponentially with the increasing number of low cost cameras and the diversity of video-based applications. Thus, advances in video compression, which enable us to represent video with fewer bits, and squeeze more pixels through bandwidth-limited channels, are critical in supporting both today and tomorrow's demand for video.

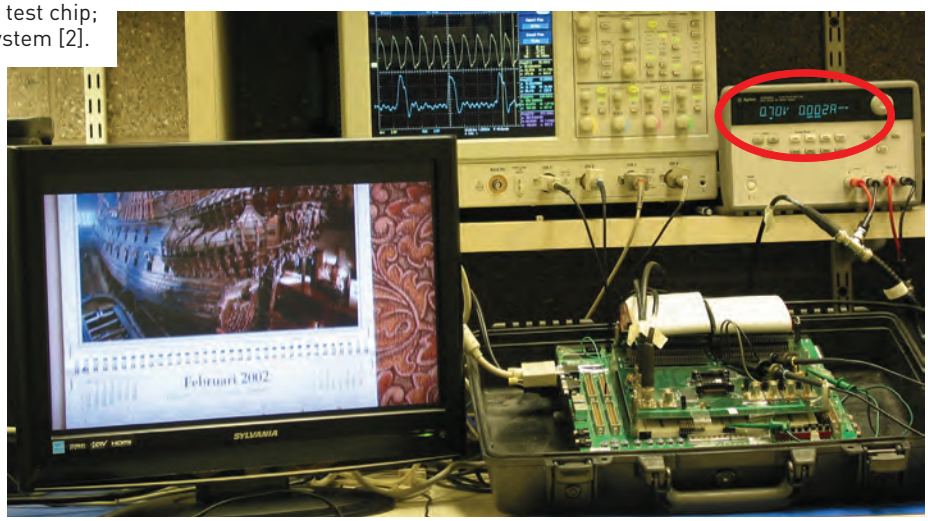
However, as we continue to push for higher coding efficiency, higher resolutions (e.g. Ultra-HD) and more sophisticated multimedia applications, the required number of computations per pixel, and the pixel processing rate, will grow exponentially. This poses significant power and performance challenges for battery-operated devices such as smart phones and tablets, as well as emerging devices such as wearable cameras and Internet of Things with cameras. For instance, the battery life of Google glasses is limited primarily due to video processing and computer vision [1]. Thus, next generation video compression systems not only need to deliver high coding efficiency, but also address implementation challenges such as power and throughput.

An effective approach to address the tight power and throughput requirements of video compression is through the use of parallelism. Parallelism can be used to increase pixels rate and any additional throughput can be traded-off for reduced power consumption with voltage scaling. Our earlier work showed that by using efficient architectures that exploit parallelism, the power consumption for decoding video sequences compressed using H.264/AVC, today's most widely used video compression standard, can be reduced by a factor of 10x (Fig. 1, below) [2].

However, efficient architectures alone are not sufficient as the video compression algorithms limit the amount of parallelism that can be exposed. Video compression works by removing redundancy in the video sequences, which naturally introduces dependencies in the data. Accordingly, advanced compression algorithms that add a lot of dependencies for increased compression are more difficult to parallelize. An example of this is the entropy-coding engine of the video codec called Context-Adaptive Binary Arithmetic Coding (CABAC). Entropy coding is a form of lossless compression used at the last stage of video encoding (and the first stage of video decoding), after the video has been reduced to a series of syntax elements. Syntax elements describe how the video sequence can be reconstructed at the decoder. Entropy coding achieves compression by mapping elements to bits based on the probability of occurrence (e.g. in the English alphabet, you would assign fewer bits to vowels, and more bits to consonants); thus it is important to accurately model the probabilities of the elements in order to achieve high coding efficiency. CABAC uses several hundred probability models to capture the distribution of the various syntax elements and uses a sophisticated finite state machine to select the correct probability model for each element. The models are updated on-line during the compression and decompression process. Although CABAC delivers higher compression than alternative entropy coding approaches, the complex probability model selection and update lead to tight data dependencies in the form of feedback loops (Fig. 2); this limits the overall throughput of the video codec making it difficult to achieve the desired pixel rate or trade-off the throughput for increased battery life. throughput of the video codec making it difficult to achieve the desired pixel rate or trade-off the throughput for increased battery life.

Figure 1: (Left) Die photo of low power decoder test chip; (Right) Setup of real-time HD video decoding system [2].





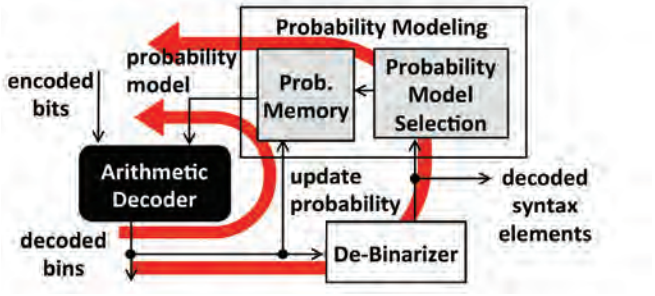


Figure 2:Feedback loops in CABAC entropy decoding engine.

Accordingly, the CABAC was re-designed using joint algorithm and architecture optimization to increase throughput while maintaining high coding efficiency. One key insight was that the encoded data could be reorder such that the dependencies within the feedback loops could be reduced and enable multiple loops to run in parallel. Removing dependencies also reduced memory accesses, which sped up each of the loops. The new CABAC algorithm was able achieve over 10x higher throughput compare to state-of-the-art H.264/AVC CABAC implementations [3], which translates to a 3x power reduction when combined with voltage scaling. Several concepts from this work (e.g. simplified probability model selection, line memory reduction and wavefront parallel processing based on interleaved entropy slices) were adopted into the latest video coding standard High Efficiency Video Coding (HEVC) [4], a.k.a. H.265. Based on these design principles, HEVC contains multiple built-in implementation-friendly features while still delivering 50% higher coding efficiency compared to its predecessor H.264/AVC [5, 6]. HEVC is now being deployed on numerous devices (e.g. televisions, phones, set-top boxes).

In the Energy-Efficient Multimedia Systems Group, we are investigating the use of optimization methods to further improve the coding efficiency and reduce the power consumption of next-generation of video compression systems that could be incorporated into future standards (e.g. 'H.266'). While video compression continues to be a challenge that needs to be addressed, many of the emerging video-driven applications do not require the complete reconstruction of the compressed video. Instead, it may be sufficient to extract and store/transmit only the relevant information from the video rather than the pixels for the video itself, which can result in much more significant compression. For instance, in retail and traffic surveillance applications, it may necessary to only store/transmit the number of customers or vehicles that appeared in the video over a certain time period. The processing required to extract the desired information typically occurs near the camera, where energy is constrained. Thus, we are also investigating low power methods of extracting this information using computer vision algorithms such object detection and recognition. In our recent work, we showed that processing the gradient image rather than the original pixels (Fig. 3) reduces the energy cost of image scale generation, required for detecting objects of different sizes, by 43% with only a 2% reduction in detection accuracy [7]. Enabling real-time energy-efficient video processing can impact a wide range of emerging video-based applications ranging from improved safety through elderly assistance, advanced driver assistance systems and crime prevention to increased efficiency through structural monitoring, smart homes, navigation of unmanned vehicles and traffic control.

[1] E. Ackerman. (2013, January) IEEE Spectrum. [Online]. http://spectrum.ieee.org/consumer-electronics/gadgets/google-gets-in-your-face
[2] V. Sze, D. Finchelstein, M. E. Sinangil, A. P. Chandrakasan, "A 0.7-V 1.8-mW H.264/AVC 720p Video Decoder," IEEE Journal of Solid-State Circuits, November 2009.
[3] V. Sze, A. P. Chandrakasan, "A highly parallel and scalable CABAC decoder for next generation video coding," IEEE Journal of Solid-State Circuits, January 2012.
[4] V. Sze, M. Budagavi, "High-Throughput CABAC in HEVC," IEEE Transactions on Circuits and Systems for Video Technology, Dec. 2012.
[5] V. Sze, M. Budagavi, G. J. Sullivan, "High Efficiency Video Coding (HEVC) – Algorithms and Architectures," Springer, 2014.
[6] ITU-T and ISO/IEC, ITU-T Rec. H.265 and ISO/IEC 23008-2: High Efficiency Video Coding, April 2013.
[7] A. Suleiman, V. Sze, "Energy-Efficient HOG-based Object Detection at 1080HD 60 fps with Multi-Scale Support," IEEE International Workshop on Signal Processing Systems (SiPS), October 2014.
[8] INRIA Persons Dataset http://pascal.inrialpes.fr/data/human/

Figure 3: (Left) Original figure from INRIA Persons Dataset [8]. (Right) Perform gradient pre-processing of image before object detection to reduce energy consumption.