

COMMUNICATIONS

CACM.ACM.ORG

OF THE

ACM

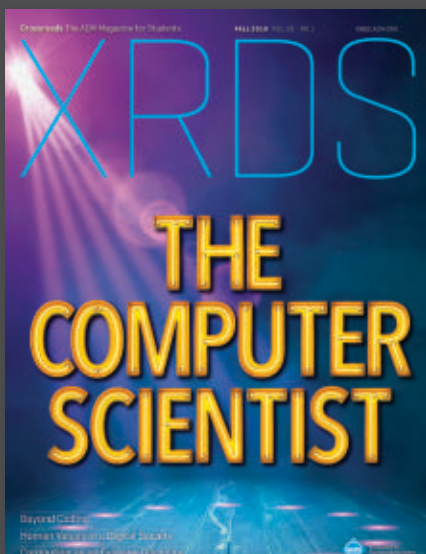
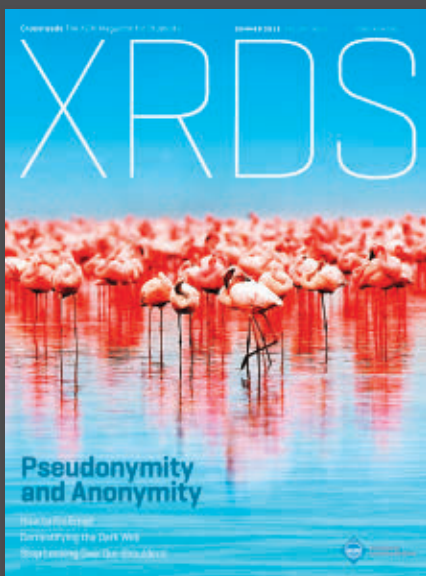
01/2021 VOL.64 NO.01

Does Facebook Use Sensitive Data for Advertising Purposes?

Geometric Deep Learning Advances Data Science
Insights for AI from the Human Mind
Digital Instruments as Invention Machines

Association for
Computing Machinery

acm



XRDS

At XRDS, our mission is to empower computer science students around the world. We deliver high-quality content that makes the complexity and diversity of this ever-evolving field accessible. We are a student magazine run by students, for students, which gives us a unique opportunity to share our voices and shape the future leaders of our field.

Accessible, High-Quality, In-Depth Content We are dedicated to making cutting-edge research within the broader field of computer science accessible to students of all levels. We bring fresh perspectives on core topics, adding socially and culturally relevant dimensions to the lessons learned in the classroom.

Independently Run by Students XRDS is run as a student venture within the ACM by a diverse and inclusive team of engaged student volunteers from all over the world. We have the privilege and the responsibility of representing diverse and critical perspectives on computing technology. Our independence and willingness to take risks make us truly unique as a magazine. This serves as our guide for the topics we pursue and in the editorial positions that we take.

Supporting and Connecting Students At XRDS, our goal is to help students reach their potential by providing access to resources and connecting them to the global computer science community. Through our content, we help students deepen their understanding of the field, advance their education and careers, and become better citizens within their respective communities.

XRDS is the flagship magazine for student members of the Association for Computing Machinery [ACM].

www.xrds.acm.org



Association for
Computing Machinery

Concurrency

The Works of Leslie Lamport

This book is a celebration of Leslie Lamport's work on concurrency, interwoven in four-and-a-half decades of an evolving industry: from the introduction of the first personal computer to an era when parallel and distributed multiprocessors are abundant. His works lay formal foundations for concurrent computations executed by interconnected computers. Some of the algorithms have become standard engineering practice for fault tolerant distributed computing - distributed systems that continue to function correctly despite failures of individual components. He also developed a substantial body of work on the formal specification and verification of concurrent systems, and has contributed to the development of automated tools applying these methods.

Part I consists of technical chapters of the book and a biography. The technical chapters of this book present a retrospective on Lamport's original ideas from experts in the field. Through this lens, it portrays their long-lasting impact. The chapters cover timeless notions Lamport introduced: the Bakery algorithm, atomic shared registers and sequential consistency; causality and logical time; Byzantine Agreement; state machine replication and Paxos; temporal logic of actions (TLA). The professional biography tells of Lamport's career, providing the context in which his work arose and broke new grounds, and discusses LaTeX - perhaps Lamport's most influential contribution outside the field of concurrency. This chapter gives a voice to the people behind the achievements, notably Lamport himself, and additionally the colleagues around him, who inspired, collaborated, and helped him drive worldwide impact. Part II consists of a selection of Leslie Lamport's most influential papers.

This book touches on a lifetime of contributions by Leslie Lamport to the field of concurrency and on the extensive influence he had on people working in the field. It will be of value to historians of science, and to researchers and students who work in the area of concurrency and who are interested to read about the work of one of the most influential researchers in this field.

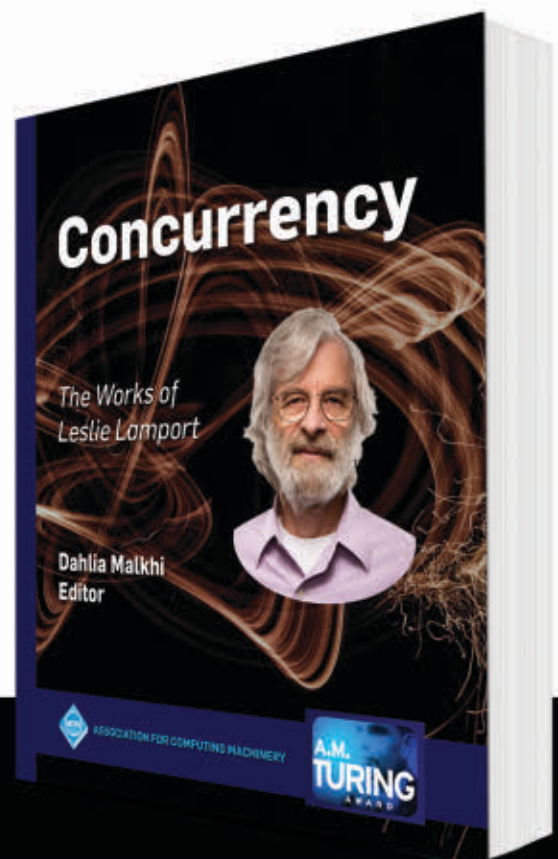
Dahlia Malkhi, Editor

ISBN: 978-1-4503-7271-8

DOI: 10.1145/3335772

<http://books.acm.org>

<http://store.morganclaypool.com/acm>



ACM BOOKS
Collection II

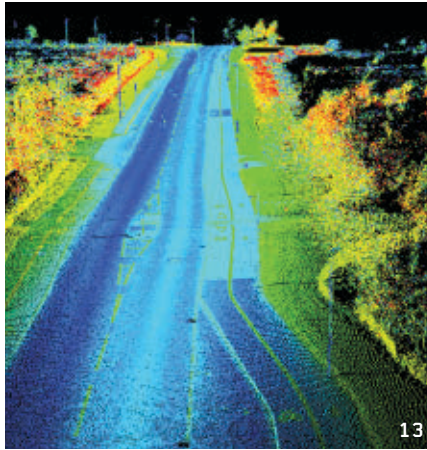
Departments

- 5 **Editor's Letter**
2021: Computing's Divided Future
By Andrew A. Chien
-
- 7 **Vardi's Insights**
Reboot the Computing-Research Publication Systems
By Moshe Y. Vardi
-
- 9 **Career Paths in Computing**
A Career Fueled by HPC
By Dona Crawford
-
- 10 **BLOG@CACM**
Talking about Race in CS Education
Mark Guzdial suggests computer science education needs to change, to better serve the needs of students and society.
-
- 125 **Careers**

Last Byte

- 128 **Upstart Puzzles**
Stay in Balance
No tipping.
By Dennis Shasha

News



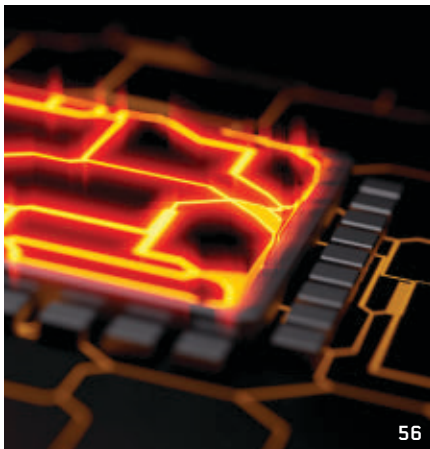
- 13 **Geometric Deep Learning Advances Data Science**
Researchers are pushing beyond the limitations of convolutional neural networks using geometric deep learning techniques.
By Samuel Greengard
-
- 16 **Fugaku Takes the Lead**
Japan tops the Top500 supercomputer rankings, for the moment.
By Don Monroe
-
- 19 **Coalition of the Willing Takes Aim at COVID-19**
Data science can only do so much in the face of a pandemic.
By Chris Edwards

Viewpoints

- 22 **Technology Strategy and Management**
Boeing's 737 MAX: A Failure of Management, Not Just Technology
Tracing the trajectory of management and engineering decisions resulting in systemic catastrophe.
By Michael A. Cusumano
-
- 26 **Security**
Cybersecurity Research for the Future
Considering the wide range of technological and societal trade-offs associated with cybersecurity.
By Terry Benzel
-
- 29 **Law and Technology**
Content Moderation Modulation
Deliberating on how to regulate—or not regulate—online speech in the era of evolving social media.
By Kate Klonick
-
- 32 **Historical Reflections**
The Immortal Soul of an Old Machine
Taking apart a book to figure out how it works.
By Thomas Haigh
-
- 38 **Viewpoint**
Insights for AI from the Human Mind
How the cognitive sciences can inform the quest to build systems with the flexibility of the human mind.
By Gary Marcus and Ernest Davis
-
- 42 **Viewpoint**
Excessive Use of Technology: Can Tech Providers be the Culprits?
Seeking to assess the possible responsibility of tech providers for excessive use patterns.
By Ofir Turel and Christopher Ferguson



Practice



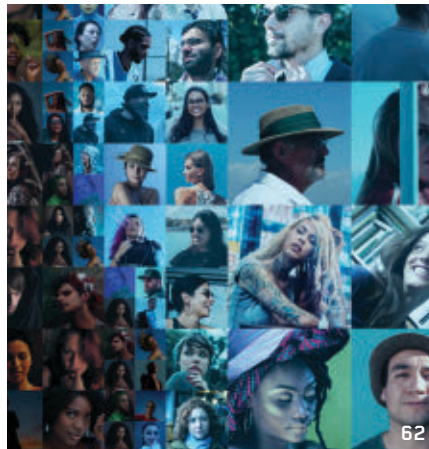
56

- 46 **The Identity in Everyone’s Pocket**
Keeping users secure through their smartphones.
By Phil Vachon

- 56 **The Die Is Cast**
Hardware security is not assured.
By Edlyn V. Levine

Q Articles’ development led by [acmqueue.queue.acm.org](https://queue.acm.org)

Contributed Articles



62

- 62 **Does Facebook Use Sensitive Data for Advertising Purposes?**
Facebook labels 67% of its users with potential sensitive interests, sometimes at great risk to the user.
By José González Cabañas, Ángel Cuevas, Aritz Arrate, and Rubén Cuevas



Watch the authors discuss this work in the exclusive *Communications* video.
<https://cacm.acm.org/videos/does-facebook-use-sensitive-data>

- 70 **Digital Instruments as Invention Machines**
An analysis of patenting history from 1850 to 2010 to detect long-term patterns of knowledge spillovers via prior-art citations of patented inventions.
By Pantelis Koutroumpis, Aija Leiponen, and Llewellyn D.W. Thomas

- 79 **How to Transition Incrementally to Microservice Architecture**
A field study examines technological advances that have created versatile software ecosystems to develop and deploy microservices.
By Karoly Bozan, Kalle Lyytinen, and Gregory M. Rose

Review Articles

- 86 **Secure Multiparty Computation**
MPC has moved from theoretical study to real-world usage. How is it doing?
By Yehuda Lindell



Watch the author discuss this work in the exclusive *Communications* video.
<https://cacm.acm.org/videos/secure-multiparty-computation>

- 97 **The Ethics of Zero-Day Exploits—The NSA Meets the Trolley Car**
Are U.S. government employees behaving ethically when they stockpile software vulnerabilities?
By Stephen B. Wicker

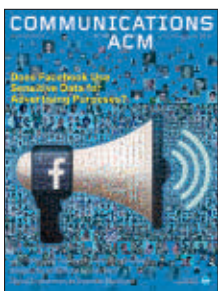
Research Highlights

- 105 **Technical Perspective**
Deciphering Errors to Reduce the Cost of Quantum Computation
By Daniel Gottesman

- 106 **Constant Overhead Quantum Fault Tolerance with Quantum Expander Codes**
By Omar Fawzi, Antoine Grospellier, and Anthony Leverrier

- 115 **Technical Perspective**
SkyCore’s Architecture Takes It to the ‘Edge’
By Richard Han

- 116 **SkyCore: Moving Core to the Edge for Untethered and Reliable UAV-Based LTE Networkst**
By Mehrdad Moradi, Karthikeyan Sundaresan, Eugene Chai, Sampath Rangarajan, and Z. Morley Mao



About the Cover: Facebook labels 67% of its users with sensitive interests, often responding with advertising promoting those interests. Governmental policies to protect privacy have had little impact and in some countries such ads can pose risky consequences. This month’s cover story reviews FB study findings and user recourse. Cover illustration by Charis Tsevis.



Andrew A. Chien

DOI:10.1145/3439708

2021: Computing's Divided Future

THROUGH THE FOG of headline-grabbing tweets and TikTok ban lawsuits, we can see the tectonic plates of China's and the West's computing ecosystems rapid movement apart. The growing importance of computing as a military and intelligence technology makes this inevitable; compounded by its pervasive presence in society, commerce, and government. At this writing, China continues aggressive actions to eliminate democracy in Hong Kong and military demonstrations to intimidate Taiwan and other nations in Southeast Asia.^a China's government has reined in Chinese technology companies such as Ant Financial (Alibaba) and Tencent, overtly signaling its intent for full control.^b These actions also affect foreign multinationals operating in China (for example, Apple and AirBnB^c). The action is two-sided with the U.S. government acting to restrict exports based on critical technologies (semiconductors), disrupting visa programs for students and visitors, and increasing reporting requirements for foreign engagements. U.S. government prosecution of high-profile researchers for undisclosed ties and payments has had a chilling effect on collaboration with researchers

in China.^d And, the global COVID-19 pandemic has only exacerbated misinformation and conflict.

Recent events have framed starkly these concerns in a scope expanding from hardware technology to software (even *algorithms* in the case of TikTok) to user-data collection. As we enter a new year, China and the West's fundamental systemic differences and growing geopolitical competition are increasingly open. The new reality is pulling the computing community apart, and yes, spilling over into the academic and research communities. Some analysts project rapid evolution from one computing community to two Internets and then into two business and technology ecosystems, and ultimately "decoupling" into two largely disjoint technology bases.^e The growing schism is far wider, but computing is unavoidably at ground zero. The fissures have grown over time, but their growth has definitely accelerated over the past two years.^f Computing faces a growing divide, and the computing community confronts shifts from Open Collaboration to Coopetition and perhaps to pure Competition.

What are the implications for multinational organizations such as the ACM? For community? And for individuals who seek to study and collaborate internationally? Perhaps open

collaboration is "paradise lost"; realists clearly see the emergence of a bipolar computing world. What should we do?

► *Be aware.* Just as corporate employees need to adjust for internal and external conversations, researchers working in sensitive areas need to exercise discretion.

► *Differentiate carefully.* Separating basic research from areas that might be sensitive for national security may avoid chilling restrictions.

► *Act responsibly.* Egregious abuse will trigger greater oversight and restrictions, and erosion of open scientific collaboration

None of these issues are unique to computing. Geopolitics' impact on scientific collaboration has strong, parallel precedents in physics (nuclear weapons) and biology^g (biological weapons). Can we do better? Computing's culture of open collaboration is a rich legacy of technology and community. It has a rich web of personal relationships. Can this web help us avoid the harshest outcomes?

Temper distrust, secrecy, and competition by keeping the humanity of your colleagues in mind. As tensions rise, many individuals will face difficult choices torn by conflicting loyalties. As individuals we all want to survive and thrive—personally and professionally. But each of us do so within imperfect systems—perhaps not to our liking and certainly beyond our control.

Let's work to preserve computing's community!

Andrew A. Chien, EDITOR-IN-CHIEF

Andrew A. Chien is the William Eckhardt Distinguished Service Professor in the Department of Computer Science at the University of Chicago, Director of the CERES Center for Unstoppable Computing, and a Senior Scientist at Argonne National Laboratory.

Copyright held by author/owner.

a A. Ramzy, T. May, and E. Yu. China targets Hong Kong's lawmakers as it squelches dissent. *New York Times* (Nov. 11, 2020). China sends warning to Taiwan and U.S. with big show of air power. *New York Times* (Sept. 18, 2020). R. Zhong. In halting ant's IPO, China sends a warning to business. *New York Times* (Nov. 6, 2020).

b R. Liao. China finally grants a game license to Tencent. *TechCrunch* (Jan. 24, 2019).

c S. Liao. Apple officially moves its Chinese iCloud operations and encryption keys to China. *The Verge* (Feb. 29, 2020). I.C. Campbell. Airbnb's Chinese data policies reportedly cost it an executive. *The Verge* (Nov. 20, 2020).

d N. Ord. TSMC reportedly strikes deal with U.S. to supply chips to Huawei but with a caveat. *Hot Hardware* (Oct. 9, 2002). NPR. Acclaimed Harvard scientist is arrested, accused of lying about ties to China (Jan. 28, 2020).

e A. Hoecker, S. Li, and J. Wang. *U.S. and China: The Decoupling Accelerates*. Bain & Co., (Oct. 14, 2020).

f R. Singel and D. Kravets. Only Google could leave China. *WIRED* (Jan. 15, 2010). A. Chien. Sustaining open collaboration in universities. *Commun. ACM* (Sept. 2019). A. Chien. Cracks in open collaboration. *Commun. ACM* (Jan. 2020).

g E. Harris. *Governance of Dual-Use Technologies: Theory and Practice*. AAAS, 2016.

ACM Transactions on Computing for Healthcare (HEALTH)

Open for
Submissions

A multidisciplinary journal for
high-quality original work on how
computing is improving healthcare



Computing for Healthcare has emerged as an important and growing research area. By using smart devices, the Internet of Things for health, mobile computing, machine learning, cloud computing and other computing based technologies, computing for healthcare can improve the effectiveness, efficiency, privacy, safety, and security of healthcare (e.g., personalized healthcare, preventive healthcare, ICU without walls, and home hospitals).

ACM Transactions on Computing for Healthcare (HEALTH) is the premier journal for the publication of high-quality original research papers, survey papers, and challenge papers that have scientific and technological results pertaining to how computing is improving healthcare. This journal is multidisciplinary, intersecting CS, ECE, mechanical engineering, bio-medical engineering, behavioral and social science, psychology, and the health field, in general. All submissions must show evidence of their contributions to the computing field as informed by healthcare. We do not publish papers on large pilot studies, diseases, or other medical assessments/results that do not have novel computing research results. Datasets and other artifacts needed to support reproducibility of results are highly encouraged. Proposals for special issues are encouraged.

For more
information
and to submit
your work,
please visit:

health.acm.org



Association for
Computing Machinery



Moshe Y. Vardi

DOI:10.1145/3437991

Reboot the Computing-Research Publication Systems

A YEAR AGO, I proposed that ACM establish a policy change for its conferences, requiring that authors of accepted papers may opt out from in-person involvement and contribute instead by video.^a “Be careful what you wish for,” says an idiom. By mid-March 2020, conferences were forced to virtualize due to COVID-19. It is now clear that conferences will continue to be virtual at least until the middle, if not the end, of 2021, and perhaps even beyond that. While I am happy this will prevent adding tens of thousands of tons of CO₂ to the atmosphere, the virtualization of conferences sharpened my conviction that the computing-research publication system is badly broken and is in need of a serious reboot.

How did we get here? Back in the 1960s, journals were slow, while the field was new and needed to move fast. Conferences offered a solution: Present a preliminary version in a conference with a six-month submit-decide-present cycle, get feedback and credit, and then publish an archival version in a journal. Program committees then did not review papers; they selected papers for the program.

Over time, the selection process became a review process: decision notices became decision notices with some feedback, feedback evolved into reviews, and reviews eventually led to rebuttals. Furthermore, conferences have evolved from being a venue for *preliminary* publication to, in practice, a venue for *archival* publication. Also, conference-program time pressure, led to selectivity, which led to prestige. As a result, computing-research conferences today are clearly the preferred venue for archival publication.

But conferences were never designed to provide a venue for *high-quality* archival publication, since they lack the appropriate editorial process, whose essential element is *iterative improvement*. It is not uncommon for published journal papers to go through two and even three versions before the reviewers and editors are satisfied. Conferences that run “review experiments,” such as NIPS 2014^b and ESA 2018,^c have concluded there is a huge element of randomness in conference editorial decisions. “The reputation of the peer-review process is tarnished,” concluded Hanna Bast. Practically *everyone* in computing research complains about the “reviewers,” but the reviewers are us! If everyone is unhappy, then the problem must be systemic.

Probably the strongest arguments in favor of conferences are their predictability, in terms of submission and decision timing, and their function as venues for community building. But the promised predictability is an illusion, as papers bounce from conference to conferences until they finally find a home. And when every paper has to be published in a conference, conferences are attended mostly by junior researchers, while 20–30 years ago they offered a truly representative sample of their communities, where junior and senior researchers mingled.

A natural reaction to loss is to recreate the familiar. When a conference was “a journal that meets in a hotel,” space restriction forced a time restriction, so a conference typically lasts 2–4 days. We kept this tradition in virtual conferences, which still meet for 2–4 days, even though the original reason is gone. But if we have learned anything over these past months it is that spending a day in

virtual space is quite difficult and screen fatigue is a real phenomenon. Yet, while we all long for COVID-19 to be over, the climate-change threat is looming larger, and I suspect that virtual conferences are here to stay.

So, we seem to be *stuck* with a dysfunctional, antiquated publication system. It is time to end the debate about journals and conferences. Let us design a new publication system, something we, as computing professionals, should know how to do. We should collect system requirements, design the system, implement prototypes, experiment, and iterate. The publication system is our system. We are in charge! Technology opens new avenues, but we must be imaginative and not be bound by the dogmas of the dysfunctional past.

If we have learned anything from COVID-19 it is that dealing with major societal challenges requires *collective action*. The U.S., with its tradition of “rugged individualism” and under meager federal leadership, is handling the pandemic quite poorly. But enabling collective action is exactly why we have established professional societies. They must lead the way.

It is a cliché that everyone wants change, but no one wants to change. Let us collectively agree to change. We deserve a publication system that meets the needs of science, of scientists, and of society. Let us reboot the publication system for computing research. ACM is launching a Presidential Task force on Future Formats for ACM Conferences. It is a start! ■

Moshe Y. Vardi (vardi@cs.rice.edu) is University Professor and the Karen Ostrum George Distinguished Service Professor in Computational Engineering at Rice University, Houston, TX, USA. He is the former Editor-in-Chief of *Communications*.

Copyright held by author.

a <https://bit.ly/2KgHHR5>

b <https://bit.ly/2UKEhiq>

c <https://bit.ly/3lSIqg0>

SHAPE THE FUTURE OF COMPUTING. JOIN ACM TODAY.

www.acm.org/join/CAPP

SELECT ONE MEMBERSHIP OPTION

ACM PROFESSIONAL MEMBERSHIP:

- Professional Membership: \$99 USD
- Professional Membership plus ACM Digital Library: \$198 USD (\$99 dues + \$99 DL)

ACM STUDENT MEMBERSHIP:

- Student Membership: \$19 USD
- Student Membership plus ACM Digital Library: \$42 USD
- Student Membership plus Print *CACM* Magazine: \$42 USD
- Student Membership with ACM Digital Library plus Print *CACM* Magazine: \$62 USD

- Join ACM-W:** ACM-W supports, celebrates, and advocates internationally for the full engagement of women in computing. Membership in ACM-W is open to all ACM members and is free of charge.

PAYMENT INFORMATION

Name

Mailing Address

City/State/Province

ZIP/Postal Code/Country

- Please do not release my postal address to third parties

Email Address

- Yes, please send me ACM Announcements via email
- No, please do not send me ACM Announcements via email

- AMEX VISA/MasterCard Check/money order

Credit Card #

Exp. Date

Signature

Purposes of ACM

ACM is dedicated to:

- 1) Advancing the art, science, engineering, and application of information technology
- 2) Fostering the open interchange of information to serve both professionals and the public
- 3) Promoting the highest professional and ethics standards

By joining ACM, I agree to abide by ACM's Code of Ethics (www.acm.org/code-of-ethics) and ACM's Policy Against Harassment (www.acm.org/about-acm/policy-against-harassment).

I acknowledge ACM's Policy Against Harassment and agree that behavior such as the following will constitute grounds for actions against me:

- Abusive action directed at an individual, such as threats, intimidation, or bullying
- Racism, homophobia, or other behavior that discriminates against a group or class of people
- Sexual harassment of any kind, such as unwelcome sexual advances or words/actions of a sexual nature

BE CREATIVE. STAY CONNECTED. KEEP INVENTING.



ACM General Post Office
P.O. Box 30777
New York, NY 10087-0777

1-800-342-6626 (US & Canada)
1-212-626-0500 (Global)
Hours: 8:30AM - 4:30PM (US EST)

Fax: 212-944-1318
acmhelp@acm.org
www.acm.org/join/CAPP



CAREER PATHS IN COMPUTING

DOI:10.1145/3434141

Computing enabled me to . . . **A Career Fueled by HPC**



NAME

Dona Crawford

BACKGROUND

Born and raised in Indiana, the youngest of three girls, the first in my family to go to college.

CURRENT JOB TITLE/EMPLOYER

Board Chair, Livermore Lab Foundation

EDUCATION

**MS Operations Research, Stanford University, CA
MA German, Middlebury College, VT
and Johannes Gutenberg University of Mainz, Germany
BS Math, University of Redlands, CA**

I'VE BEEN STUMPING for why supercomputing or high-performance computing (HPC) matters most of my career.

I didn't start out loving HPC. It barely existed when I began as an undergraduate in 1969. But I loved math, the simple elegance of describing the world with equations and having the ability to be right or wrong. In a world of nuance and interpretation, the objectivity of math spoke to me (and still does). As the first in my family to go to college, I didn't know what career to pursue when I graduated. With a degree in German

and an interest in global affairs, I applied to the CIA, the FBI, the NSA, and the Department of Energy (DOE) national security labs. I was fortunate to receive an offer from DOE's Sandia, which saw fit to send me to Stanford for my MS in operations research—applied math, dealing with optimization. Once at work, I started writing code for big computers. My first assignment was to find an algorithm that would precisely evaluate the amount of sunlight hitting a parabolic trough. When it was complete, we ran it over and over again to understand how to set up solar farms for optimal capture of solar energy. This was in the late 1970s when solar farms were a new concept. Pioneering code for large, first-of-a-kind computers was thrilling and knowing that the results made a difference was exhilarating. I was hooked.

HPC has become indispensable to scientific research and discovery. It is different from what's done at Apple or Facebook or Google, among others. Those are cool jobs too, but for me, to be able to use my applied math to do science through the use of very large-scale computers was magic. We could "do hazardous, expensive, centuries-long experiments" via simulation on the computer. As a result, we, and the handful of other places (largely at national laboratories) that pioneered supercomputing fundamentally changed the scientific method from theory and experiment to theory, experiment and simulation.

While I was working at Sandia our models and large-scale computing had a big impact. For example, among other things, our knowledge of how parachutes work contributed to the development of air bags in cars. We also worked closely with Goodyear and helped them recover from financial ruin with a new way to design and produce tires using HPC.

After 25 years at Sandia, it was my privilege to lead one of the premier computing centers in the world. I did this for the last 15 years of my career at Lawrence Livermore National Laboratory (LLNL).

As HPC became more widely accepted, more data was collected and computers grew even larger, it led naturally to utilizing supercomputing for machine learning and what people think of as artificial intelligence (AI). It takes three things to be world-class in AI: the most advanced algorithms, fast computing hardware, and a good supply of data.

Recently, LLNL developed an AI-driven computational platform to simulate the molecular behavior of viruses and antibodies to create drugs for Covid.

What sets the national security labs apart from other excellent institutions is the multidisciplinary approach to solving very large, complex, long-term grand challenges. But the exciting thing about the labs is that their solutions for national security also offer great benefits for society in such domains as energy, environment and medicine.

When I retired in 2016, I had to stay involved in science and computing. That's why I co-founded the Livermore Lab Foundation (LLF). The foundation is leveraging the Lab's computing and data analytics capabilities to crunch through large amounts of data to help identify the cause of amyotrophic lateral sclerosis (ALS), or Lou Gehrig's disease. Our goal is to improve our understanding of ALS and accelerate the development of new therapies and treatments.

Also, in a world examining systemic racism, the LLF encourages black students and other historically underrepresented groups to pursue STEM by providing unique opportunities to engage with Lab facilities and mentors.

I have barely scratched the surface of how computing has changed our world for the better. Our planet faces daunting problems, and HPC is key to addressing many of them. **□**

The *Communications* Web site, <http://cacm.acm.org>, features more than a dozen bloggers in the BLOG@CACM community. In each issue of *Communications*, we'll publish selected posts or excerpts.



Follow us on Twitter at <http://twitter.com/blogCACM>

DOI:10.1145/3433921

<http://cacm.acm.org/blogs/blog-cacm>

Talking about Race in CS Education

Mark Guzdial suggests computer science education needs to change, to better serve the needs of students and society.



Mark Guzdial
CS Teachers,
It's (Past) Time
To Learn
About Race

<https://bit.ly/3ggVGlm>

June 5, 2020

The horrific death of George Floyd and the social unrest in the U.S. have raised awareness of race that has been too long left out of the mainstream conversations. I am explicitly thinking about how to incorporate an understanding of race (and culture and other identity characteristics) when we teach computer science (CS). I am just as guilty of not considering the issues of race in CS education in my everyday practice. I am in a position of privilege. I have not had to face the same life experiences that my students and colleagues have.

Back in March, when the pandemic hit the U.S. and shut down our campuses, I wrote a blog post about how much I was learning about emergency remote teaching from the ACM SIGCSE-Members email list (see <https://bit.ly/3l51pE1>). Now, as I am realizing I have been negligent in incorporating an awareness of race issues in my

CS teaching, I am again finding terrific resources in that email list, started by a note from Monica McGill (one of the leads on the terrific CSEdResearch.org website; <https://csedresearch.org/>). Two crises in just a few months, and SIGCSE is the community offering important resources for both of them.

Broadening Participation in Computing (<https://www.nsf.gov/cise/bpc/>) was the main focus of my research and service agenda for over a dozen years. I thought the goal was to get more women and underrepresented minorities into CS. CS is obviously valuable and important for students. I thought we just had to help students with diverse backgrounds to realize that. Instead, the lack of diversity is the canary in the coal mine.

I am learning our goal should be to change CS Education so that everyone is welcome and supported. CS is not a welcoming place. We CS teachers have structured our systems to keep people out, to limit access to that valuable and important knowledge. We spend so much time and energy on detecting cheating and on finding ways to limit access to our major,

which sends the message that most people don't belong. A better use of that time and energy might be to provide tutoring and change our curriculum so that more diverse students succeed. We need to send the message that we are willing to change in order to address historic and systemic inequities.

We have to change CS so it serves the needs of our students and society. Using methods like Peer Instruction and curricula like Media Computation are steps in the right direction, since they are measurably better for women and underserved populations, but those are results from the few diverse students who even walk in our door—and too few CS teachers are even willing to adopt these small measures.

We do not have a meritocracy. Our CS education systems are structured to disadvantage students who are not like us and the students currently in CS. Frankly, the game is rigged. We used to think that we were about helping students “How to Think Like a Computer Scientist” (<https://bit.ly/3jSEtGk>). But that's just telling all these students that they have to be like us to succeed. Now we have to

change how computer scientists think. We all have to change CS.

Get started educating yourself by reading Nicki Washington's paper in SIGCSE 2020, "When twice as good isn't enough: The case for cultural competence in computing" (<https://bit.ly/3244eO8>). Her paper is a great starting point because she directly addresses issues of undergraduate CS education. It's not just about race, but today, race is the elephant in the room that we (speaking as a white and as a male CS professor, which describes most U.S. CS professors) have ignored for too long. My student Amber Solomon made me aware of intersectionality in her paper "Not just Black and not just a woman: Black women belonging in computing" (<https://bit.ly/34QJriP>). Efforts to attract more Black students to CS often assume Black men. Her experience as a Black woman in computing is different. As you add other identities (like transgender), you realize that when we design our classes for the majority of our students, we are making explicit and implicit choices that make it harder for other groups.

Manuel Perez Quinones gave the most concrete example that made me question how I teach:

I will say that sometimes the problem is not in the lecture, tool, academic intervention, etc. In my experience with underrepresented students the problem is more of a personal nature rather than academic nature. For example, it should not be a surprise to anyone that students from low socioeconomic status tend to be ones that have multiple jobs, sometimes are attending to family members at home, maybe even picking up younger siblings from school, etc. And unfortunately, low socioeconomic status can be a proxy for minorities. In situations like that, having flexible deadlines makes a difference. If students work on weekends, then making a programming assignment on Sunday night (assuming you are giving them more time) is not helping and might actually put those that work at a disadvantage. Similar issues come up with office hours, labs, etc.

Do not assume that if they miss class they are lazy, irresponsible, or don't care. No, they might have other things

"We do not have a meritocracy. Our CS education systems are structured to disadvantage students who are not like us and the students currently in CS. Frankly, the game is rigged."

that are more pressing than 5 points in an assignment.

Liz Johnson shared the book *Grading for Equity* by Joe Feldman (<https://amzn.to/3jMNiS1>) and the (easier to get started) article "How Teachers are Changing Grade Practices with an Eye on Equity" (<https://bit.ly/2JwibOk>). The key idea here is standards-based grading. You set out the standards for what students have to achieve, and you give grades based on that. No pre-allocating or rationing grades. No grades for interacting with you. If your class requires attendance at office hours just to get by, your class is inequitable. You are demanding more from the students than they signed up for when registered.

Leigh Ann DeLyser, executive director of CSforAll (<https://www.csforall.org/>), made the comment that forced me to realize that I have to change for my majority students, too:

Nikki Washington and Owen Astrachan both teach courses where examples are critically examined alongside the examples we so often assume are "colorblind." Look for those examples, be explicit, not so that your black students feel welcome, but so your white students understand the minefield they are walking into.

Some of the other books now on my reading list from these discussions:

► *Race After Technology: Abolitionist Tools for the New Jim Code* by Ruha Benjamin (<https://amzn.to/34R1i97>).

► *Me and White Supremacy: Combat Racism, Change the World, and Become a Good Ancestor* by Layla F. Saad and Robin DiAngelo (<https://amzn.to/3kT2T3M>).

► *So you want to talk about Race* by Ijeoma Oluo (<https://amzn.to/3jSoLey>).

If you read nothing else from this essay, please read these two short posts from my former colleagues at Georgia Tech, Dean Charles Isbell of the College of Computing (<https://b.gatech.edu/360TukH>) and Kamau Bobb, Senior Director of the Center of the Constellations Center for Equity in Computing (<https://bit.ly/2TJTaku>). The life experience of our students and colleagues who are BIPOC (Black, Indigenous, and People of Color) is significantly different than that of the majority of people in CS today. If we ignore that, we do them a disservice.

We have ignored that. We have to correct our mistakes.

My enormous thanks to Melissa Perez, Leigh Ann DeLyser, Betsy DiSalvo, Leo Porter, Chad Jenkins, Wes Weimer, Barbara Ericson, Matthew Guzdial, Katie Guzdial, and Manuel Perez Quinones, who all gave me valuable feedback on this.

Comments

You might be interested in our book Culturally Responsive Strategies for Reforming STEM Higher Education: Turning the TIDES on Inequity.

The TIDES project had teams from quite a diverse population of studies and groups. One of the most useful activities that I participated in during my years as a professor. I learned so much.

The book isn't focused on CS, but rather STEM, but there are lessons for all of us in each chapter. We were led by the amazing Kelly Mack, with assistance from Kate Winter and a team of advisors.

<https://www.acu.org/tides>

—Douglas Blank

Mark Guzdial is professor of electrical engineering and computer science in the College of Engineering, and professor of information in the School of Information, the University of Michigan.

© 2021 ACM 0001-0782/21/1 \$15.00

Publish Your Work Open Access With ACM!

ACM offers a variety of Open Access publishing options to ensure that your work is disseminated to the widest possible readership of computer scientists around the world.



Please visit ACM's website to learn more about ACM's innovative approach to Open Access at:
<https://www.acm.org/openaccess>



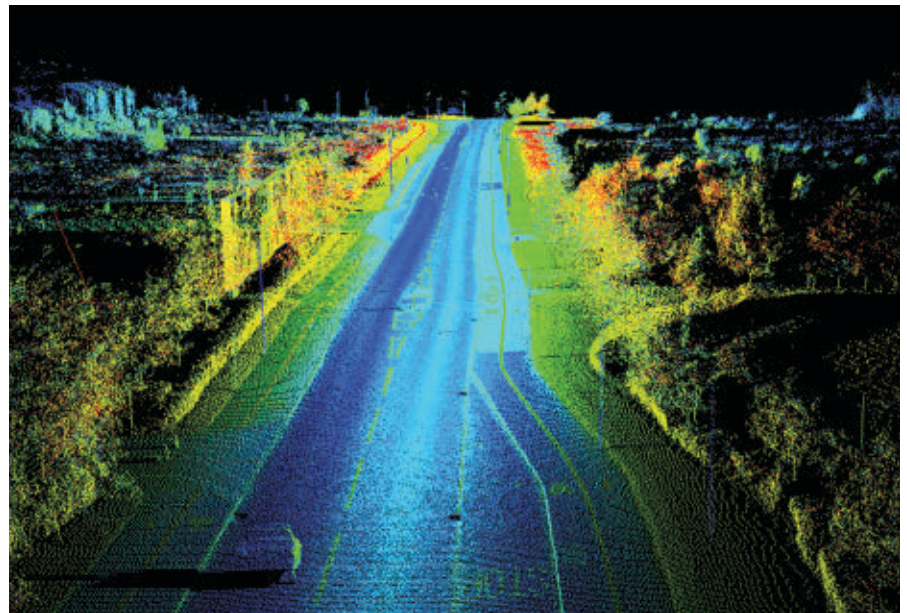
Association for
Computing Machinery

Geometric Deep Learning Advances Data Science

Researchers are pushing beyond the limitations of convolutional neural networks using geometric deep learning techniques.

DEEP LEARNING HAS transformed numerous fields. In tackling complex tasks such as speech recognition, computer vision, predictive analytics, and even medical diagnostics, these systems consistently achieve—and even exceed—human-level performance. Yet deep learning, an umbrella term for machine learning systems based primarily on artificial neural networks, is not without its limitations. As data becomes non-planar and more complex, the ability of the machine to identify patterns declines markedly.

At the heart of the issue are the basic mechanics of deep learning frameworks. “With just two layers, a simple perceptron-type network can approximate any smooth function to any desired accuracy, a property called ‘universal approximation,’” points out Michael Bronstein, a professor in the Department of Computing at Imperial College London in the U.K. “Yet, multilayer perceptrons display very weak inductive bias, in the sense that they assume very little about the structure of the problem at hand and fail miserably if applied to high-dimensional data.”



Convolutional neural networks struggle to tackle the volume and complexity of three-dimensional data, underscoring the need for geometric deep learning.

Simply put, these systems can approximate complex functions, but they do not generalize well with previously unseen data and unfamiliar examples. Thus, when the technology is applied to sophisticated computer vision and image recognition problems, simple neural networks typically require colossal training sets. Although

today’s Convolutional Neural Networks (CNNs) provide a stronger inductive bias by processing images using small local filters, they are designed to operate on 1-dimensional and 2-dimensional (2D) data, such as a photograph or audio file. Designing neural networks that can cope with more complex entities such as mole-

cules, data trees, networks, and manifolds pushes the task into a non-Euclidean world.

That is where a concept called geometric deep learning enters the picture. It relies on a broad class of approaches that use “geometric” inductive biases and concepts to make sense of non-Euclidean structures, such as graphs and manifolds. “When you go to 3D (three-dimensional) deep learning, you greatly increase the possibilities within a convolutional network,” explained Max Welling, professor and Research Chair at the University of Amsterdam, The Netherlands, and vice president of technologies for Qualcomm. “There are many exciting applications for the technology.”

Geometric deep learning aims to expand data science in much the same way that a 3D image offers more insight and perspective than a 2D photo. “There’s a natural connection to physics, in the sense that geometrical properties are typically expressed through symmetries,” said Joan Bruna Estrach, assistant professor of computer science, data science, and mathematics at the Courant Institute and the Center for Data Science at New York University. This includes signals that arise in climate science, molecular biology, and many other areas in the physical sciences.

Deeper Data Explorations

Geometric deep learning builds upon a rich history of machine learning. The first artificial neural network, called “perceptrons,” was invented by Frank Rosenblatt in the 1950s. Early “deep” neural networks were trained by Soviet mathematician Alexey Ivakhnenko in the 1960s. A major advance took place in 1989, when a group of researchers, including New York University professor (and ACM A.M. Turing Award recipient) Yann LeCun, designed the now-classical Convolutional Neural Network (CNN). The group used CNNs to solve computer vision problems that were considered incredibly difficult at that time, including that of handwritten digit recognition.

What imbues a neural network with its expressive power is a “modular design based on connecting neurons into multiple layers that can spot

highly complex problems.” As data passes through the different layers of the CNN, each layer relies on the previous layer to extract more detailed information. For example, in the case of a photo of a butterfly, the initial layer may identify the basic shape from the pixel patterns, a second neural layer may detect features such as antennae and wings, and another layer may detect colors and other features. An algorithm can determine that an object is either a butterfly, or not. The use of convolutional filters endows CNNs with an important property called shift equivariance, which means they can identify objects no matter where they are located within an image.

However, there’s a catch. Many objects and things—from molecules and scans of human organs to the streets on which autonomous vehicles must drive—are 3D and far more complex than a flat photo of a butterfly, zebra, or human face. These 3D objects have many more degrees of freedom and the shortest distance between two points isn’t necessarily how it appears in a 2D image or photo. Thus, the CNN struggles to tackle the volume and complexity of this data. Metaphorically speaking, CNNs lack the capability to see beyond the flat earth of Euclidean geometry. As a result, researchers in fields such as biology, chemistry, physics, network science, computer graphics, and social media have found they are somewhat limited in their ability to explore important data science problems.

In 2015, Bronstein introduced the term “geometric deep learning” to de-

As data passes through the convolutional neural network, each layer relies on the previous layer to extract more detailed information.

scribe neural network architectures with geometric inductive biases that can be applied to data structured as surfaces (or “manifolds” in geometric jargon) and graphs. These graphs, which are mathematical abstractions of networks, are especially useful in a broad range of applications involving systems of relations and interactions. By analyzing an object in a non-Euclidean way, including examining the edge of pixels and changing the way the convolutional neural network filters data, the system learns much more about the relationship between and among pixels.

Indeed, deep learning on graphs, which also goes by the name of “graph representation learning” or “relational inductive biases,” bears many similarities to classical CNNs, but at the same time it is very different. “Similar to convolutional neural networks, graph neural networks perform local operations with shared parameters, implemented in the form of ‘message passing’ between every node and its neighbors,” Bronstein said. However, unlike convolution operations used on grid-structured data, graph operations are permutation-invariant, which means they do not recognize the order of nodes.

A New Dimension of Equivariance

Geometric deep learning is not a complete break from classical deep learning. In fact, “If you look at the algorithms and the architectures that researchers are mostly dealing with, there’s a huge overlap,” Bruna pointed out. “In reality, deep learning represents a continuum of increasingly structured architectures that reflect inductive biases of the physical world.” Bruna said CNNs serve as a “canonical instance” of a more basic translation symmetry. “Geometric deep learning provides a toolkit to express symmetries and [processes] that work best for a specific task or type of computational problem,” he said.

The technique is opening up new vistas for understanding data. A team of researchers at the Netherlands’ University of Amsterdam, including Taco Cohen, a machine learning researcher and Ph.D. candidate, advanced the field in 2018 when they figured out a way to encode basic assumptions

about images and models into geometric deep learning algorithms. By scanning a plane of pixels for an entire volume, creating a 3D map and using the artificial neural net, they were able to leapfrog conventional CNN methods when studying lung cancer computed tomography (CT) scans. The approach produced results on par with conventional CNNs using only about a tenth of the data. “Whereas classical convolutional networks need to learn the appearance of lung nodules in every orientation, our network can automatically recognize nodules no matter their orientation, due to its rotation equivariance property,” Cohen explained.

As the team continued to study various models, they confirmed their approach could address equivariance issues, also known as covariance in physics. In other words, the same data presented in different ways or collected by different systems produced the same results. Then, when they analyzed climate data, they found that conventionally trained CNNs resulted in 74% accuracy in identifying extreme weather patterns, such as cyclones. The same data run through a geometric learning gauge CNN they built detected storms with nearly 98% accuracy.

As researchers attempt to develop models that detect and predict events in biology, chemistry, and physics, the ramifications are clear. “There are a huge number of remarkable insights to be gained by applying ideas used in physics and mathematics to produce new deep learning models,” Welling explained. Although the technology is still in the nascent stages, it already is showing remarkable potential. Bronstein said the approach could revolutionize everything from materials science to medicine, and even social media. It will help scientists discover new combinations of compounds that lead to new types of antibiotics, and more effective cancer drugs.

The advantages don’t stop there, however. Geometric deep learning can disregard nuisance variations that cause conventional CNNs to go completely haywire. “A standard convolutional neural network can recognize visual patterns regardless of how they are shifted in the image plane, but

Scientists are turning to geometric deep learning to explore complex problems that require highly precise results.

can easily get confused by rotated patterns” Cohen said.

A New Model Emerges

Not surprisingly, challenges remain in developing geometric deep learning systems that are fully equipped to solve real-world problems. Bronstein said that for now, scalability is a key factor limiting industrial applications. “Real-life applications often have to deal with very large graphs with hundreds of millions of nodes and billions of edges, such as Twitter and Facebook social graphs. So far, the focus of academic research in geometric deep learning has been primarily on developing new models, and these important aspects have until recently been almost completely ignored. As a result, many graph neural network models are completely inadequate for large-scale settings.”

Another crucial factor limiting geometric deep learning is that real systems are not static; they evolve in time, and hence require methods capable of dealing with dynamic graphs. “This topic has also been only scarcely addressed in the literature,” Bronstein said.

Still another obstacle is developing chips and hardware specifically designed to tackle geometric deep learning. Today’s systems use graphics processing units (GPUs) and central processing units (CPUs)—which are ideal for conventional CNNs operating on a stream of pixels. However, they are not necessarily the best fit for graph-structured data, where data can come in random order. “In the long run, we might need specialized hardware for graphs,” Bronstein says.

Nevertheless, the field continues to gain traction. Scientists are turning to geometric deep learning to explore complex problems that require highly precise results. Among those particularly interested in the field are physicists and chemists who work with large and wildly disparate data sets based on foundational data structures that are known in advance. Geometric deep learning greatly increases their ability to understand molecular structures, cosmological maps and Feynman diagrams with pictorial representations of extraordinarily complex 3D subatomic particles.

Concludes Welling, “Geometric deep learning and gauge-equivariant CNNs are likely to emerge as standard tools in the data science toolkit. They are advancing rapidly because there’s a growing recognition they can tackle new and entirely different sets of problems.”

Further Reading

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-Based Learning Applied to Document Recognition, *Proceedings of the IEEE*, November 1998. Volume: 86, Issue: 11. <https://ieeexplore.ieee.org/document/726791>

Bronstein, M.M., Bruna, J., LeCun, Y., Szlam, A., and Vandergheynst, P. Geometric Deep Learning: Going Beyond Euclidean Data, *IEEE Signal Processing Magazine*. July 2017. Volume: 34, Issue: 4. <https://ieeexplore.ieee.org/abstract/document/7974879>

Masci, J., Rodolà, E., Boscaini, D., Bronstein, M.M., and Li, H. Geometric Deep Learning. SA '16: *SIGGRAPH ASIA 2016 Courses*. November 2016. Article No.: 1, Pages 1–50. <https://doi.org/10.1145/2988458.2988485>

Cohen, T.S., Weiler, M., Kicanaoglu, B., and Welling, M. Gauge Equivariant Convolutional Networks and the Icosahedral CNN, *Proceedings of the International Conference on Machine Learning (ICML)*, 2019. <https://arxiv.org/abs/1902.04615>

Cohen, T.S. and Welling, M. Steerable CNNs, *Proceedings of the International Conference on Learning Representations*, 2017. <https://arxiv.org/abs/1612.08498>

Samuel Greengard is an author and journalist based in West Linn, OR, USA.

© 2021 ACM 0001-0782/21/1 \$15.00

Fugaku Takes the Lead

Japan tops the Top500 supercomputer rankings, for the moment.

JAPAN'S ARM-BASED FUGAKU supercomputing system has been acknowledged as the world's most powerful supercomputer. In June 2020, the system earned the top spot in the Top500 ranking of the 500 most powerful commercially available computer systems on the planet, for its performance on a longstanding metric for massive scientific computation. Although modern supercomputing tasks often emphasize somewhat different capabilities, Fugaku also outperforms by other measures as well.

"It's amazing on all benchmarks. This architecture just wins big time," said Torsten Hoefer of the Swiss Federal Institute of Technology (ETH) Zurich. "It is a super-large step." Hoefer shared the 2019 ACM Gordon Bell Prize with an ETH Zurich team for simulations of heat and quantum electronic flow in nanoscale transistors performed in part on the previous Top500 leader, the Summit System at the U.S. Department of Energy's Oak Ridge National Laboratory (ORNL) in Tennessee.

Fugaku's performance on the Top500's High-Performance Linpack (HPL) benchmark is an impressive 0.4 exaflop/s (10^{18} floating-point operations per second), besting Summit by a factor of 2.8 for double-precision (64-bit) arithmetic. For faster, lower-precision operations, the Fugaku system has already exceeded an exaflop/s.

In his acceptance of the Top500 award, however, Satoshi Matsuoka, director of the Japanese government-funded RIKEN Center for Computational Science (R-CCS) in Kobe, stressed that the design, done in close collaboration with Fujitsu, was motivated by performance on real-world applications. "Our intention was never to build a machine that only beat the benchmarks," said Matsuoka, who shared the ACM Gordon Bell Prize with a team of colleagues in 2011.

Top500 pioneer Jack Dongarra, of ORNL and the University of Tennessee



The Fugaku supercomputer, currently the world's fastest, at the Riken Center for Computational Science in Kobe, Japan.

at Knoxville, said three new systems in the U.S., and possibly others in China, were expected to achieve exaflop/s performance on 64-bit arithmetic within the next year. Even if its supremacy is fleeting, the Fugaku architecture includes innovations, notably vector arithmetic, that could ease programming and exemplify an alternate paradigm for designing high-performance computers.

Race to the Top

The Top500 list includes 500 powerful systems from around the world, but the few near the top get the most attention. These systems tend to be funded as national resources in major facilities like U.S. national laboratories and RIKEN, a research institute supported by the Japanese government. In this, and in their cost, the leading supercomputers are similar to scientific instruments like the Hubble Space Telescope. "The Fugaku machine is reported to be \$1 billion U.S." to develop and build, Dongarra said. "They're pushing the technology and you pay a price for that." Fugaku comprises 158,976 nodes (more

than 7 million CPU cores) distributed among 432 racks. Including the support infrastructure, it draws some 30MW of electricity, enough to power some 20,000 U.S. homes.

Unlike the Hubble, which only does astronomy, these systems run simulations that illuminate a diverse range of scientific challenges. "The top 10 machines are really built to solve problems that no other machine can solve," said Hoefer, including "the big challenge problems in society" such as climate change, brain research, and recently the COVID-19 crisis. Their general-purpose design makes them slightly less efficient than a specialized machine, but ensures broad funding support. Their flagship status also precludes specialized chips, such as those being developed for machine learning. "I think people would think twice before they build a \$200-million machine based on those chips," Hoefer said, especially because the algorithms used for cutting-edge computation continue to evolve rapidly.

Fugaku is built around a Fujitsu processor designated A64FX, developed

for this system in collaboration with ARM. It is expected to find use in other high-powered computers as well, including one system being developed by Cray and others marketed by Fujitsu. “The architecture that is pioneered by systems in the Top500 is going to be used in industry in order to solve real engineering problems,” Hoefler said.

Nonetheless, basing Fugaku on a dedicated chip is a departure from recent top supercomputer architectures, which leverage higher-volume chips designed for less-demanding applications. This approach offloads many costs of design and development needed to keep pace with advancing semiconductor technology. The off-the-shelf approach has its own risks, though. In the summer of 2020, Intel announced manufacturing problems with its latest chips, which may result in delays for the U.S.-based exascale supercomputers that will incorporate them.

Each A64FX chip, manufactured using TSMC’s 7nm FinFET process, contains almost 90 billion transistors and features 48 Arm 8.2A CPUs, whose reduced-instruction-set computing (RISC) design contrasts with most of the processors employed in the Top500. Dongarra says 94% of the Top500 machines use Intel processors, which offer complex-instruction-set computing (CISC) to programmers, while only three currently use ARM. Summit, however, uses the Power9 processor from IBM, which also has a RISC architecture.

TSMC’s Chip-on-Wafer-on-Substrate (CoWoS) process is used to stack high-bandwidth memory (HBM2) on top of the processor chip. “Our studies show that bandwidth is very important to sustain the speedup of the applications,” Matsuoka stressed. The chips also provide interfaces with an updated version of the Tofu interconnect, a system with a six-dimensional torus topology that was previously developed by Fujitsu.

Revenge of Vector Architecture

From an architectural perspective, the most dramatic choice is what Fugaku does not have: graphics processor units, or GPUs. These increasingly powerful computation-intensive chips, often made by Nvidia or AMD, frequently are used as cost-effective accelerators to offload intensive parallel

Dongarra says 94% of the Top500 machines use Intel processors; Fugaku is built around the Fujitsu A64FX processor, developed for it in collaboration with ARM.

computations from CPUs for both high-performance scientific computations and machine learning.

Instead, Fugaku’s CPUs incorporate instructions that ARM calls Scalar Vector Extension (SVE). Compared to GPUs, this vector architecture is “a more elegant and easier-to-compile architecture that’s trying to take advantage of that same level of parallelism,” said David Patterson, professor emeritus at the University of California at Berkeley and co-recipient (with John Hennessy) of the 2017 ACM A.M. Turing Award. “You can explain how it works to scientists, it’s got an elegance that lets it scale to very powerful computers with time, and it’s easy to compile for.”

“It has been a long time since the fastest computer on the Top500 had a vector processor in it,” Patterson noted. “Is that what things are going to look like more in the future? That’s going to be interesting to watch.”

Although fixed-length vector operations have been implemented elsewhere, SVE harkens back to the type of vector operations originally envisioned by Seymour Cray in his early supercomputers. “It’s not a fixed-size vector but a variable-size vector, where you can vectorize whole loops,” Hoefler said.

GPUs traditionally force users to identify throughput-sensitive code and explicitly specify fine-grain parallelism for those operations. “In the Fugaku system, you don’t need to that,” Hoefler said. “Fugaku is kind of the first serious implementation of those [ideas], at least since Cray’s time. Those could

ACM Member News

INNOVATIONS IN ALGORITHMS FOR TENSOR DECOMPOSITIONS



“I have been programming since I was a kid, when I had a Commodore 64 with a tape drive,” says

Tamara Kolda, Distinguished Member of the Technical Staff in the Data Science and Cyber Analytics Department at Sandia National Laboratories in Livermore, CA.

This early interest in computers stayed with her, and Kolda subsequently took computer classes all through college and graduate school.

“Applied Math at my university had a pretty broad definition, and included a lot of computer science,” she says.

Kolda earned her undergraduate degree in mathematics from the University of Maryland, Baltimore County (UMBC), and her master’s and Ph.D. degrees, both in Applied Mathematics, from the University of Maryland, College Park (UMCP).

After obtaining her doctoral degree, Kolda did postdoctoral work at the U.S. Department of Energy’s Oak Ridge National Laboratory in Tennessee. After two years at Oak Ridge, Kolda joined Sandia, where she has remained since.

Kolda’s main area of interest is in tensor decomposition, a tool for unsupervised machine learning. She co-developed the Tensor Toolbox for MATLAB, as well as many other software packages. Other research interests include network/graph algorithms and analysis, data mining, and cybersecurity.

Data science is maturing rapidly and the big strides made initially are becoming harder to find, Kolda says. “Digging into machine learning methods and bringing in more rigor and justification to things that were heuristic will be key,” Kolda says, “particularly when they hit points where there is some blockage to the application or to the success of a particular method.”

—John Delaney

be really easier to program. I'm super-excited about this."

CPUs also typically have needed more power than GPUs, but in the A64FX, "our power efficiency is pretty much in the range of GPUs or the latest breeds of specialized accelerators while being a general-purpose CPU," Matsuoka said. "This was because we really tuned for high-performance computing."

Decades of Progress

The Top500 has been tracking the exponential improvement in supercomputer performance since 1993, based on the Linpack benchmark Dongarra developed in 1979. At the time, he said, floating point operations were expensive, so 64-bit matrix multiplications formed the core of the benchmark. The same metric is still used to judge the Top500 today.

Parallel computing has become particularly important as clock speeds on individual processors hit a ceiling due to chip heating and other issues. However, because any calculation has some parts that must be done serially, adding more processors in parallel gives diminishing returns in speedup.

Nonetheless, more parallel processors do let researchers attack larger problems efficiently. "Not everybody wants to solve the same problem faster," said Patterson. "Linpack really embraced that and allows people to solve any matrix size they want. The bigger the computer, the bigger the matrix. I don't know how many people want to solve a problem that's

10 million by 10 million dense matrix on a side, but that's the problem they're solving." When Linpack was introduced, "these big matrices were the total workload that people were running on those machines," agreed Hoefler, but "following Moore's Law for 40 years, the matrices that people can solve on these machines today are way larger than what anybody would do in practice."

"While it's interesting from a historical perspective, it probably doesn't really reflect the kind of performance we see for what I'll call normal applications run on supercomputers," Dongarra acknowledged. In particular, he said, even in intensive scientific calculations, such as solving the partial differential equations that appear in simulations of complex three-dimensional systems such as climate models, the matrices are sparse, meaning they have only a small number of non-zero entries, arranged in predictable patterns.

To assess such sparse-matrix operations, the Top500 team also tracks the HPCG (high-performance conjugate gradients) benchmark. In addition, machine-learning applications typically don't require full 64-bit accuracy, so Dongarra and his colleagues have introduced a lower-precision version called HPL-AI. Still, on both these benchmarks, Fugaku also ranks highest, achieving 1.4 exaflop/s on HPL-AI.

Nonetheless, Patterson worries "whether the Linpack benchmark is

leading to architecture innovations that allow important algorithms, or ... we're just creating one-trick ponies." He has been supporting an alternative, known as MLPerf, which includes both the training and inference aspects of machine learning. It features a suite of tasks that are frequently updated, including, for example, a large-scale language model within two years of the research paper that introduced it. MLPerf also has an "open" category that leaves the implementation unspecified, to encourage algorithmic innovation. "The benchmark challenge is, how do you have a fair challenge and encourage innovation?" Patterson noted.

Still, Hoefler thinks the continuity of the Top500 provides important context for machines like Fugaku, and notes that machine learning algorithms still rely heavily on the same fused multiply-add operations that power matrix multiplications. "HPL is less relevant than it was, but I believe that it's incredibly important from a historic perspective." **C**

Further Reading

Top500: The List
www.top500.org

Report on the Fujitsu Fugaku System, Jack Dongarra, June 2020, <https://bit.ly/2EQS6Yt>

MLPerf Benchmarks, <https://mlperf.org/>

Don Monroe is a science and technology writer based in Boston, MA, USA.

© 2021 ACM 0001-0782/21/1 \$15.00

Milestones

Sarkar to Receive 2020 ACM-IEEE CS Ken Kennedy Award

ACM and IEEE Computer Society (IEEE-CS) recently named Vivek Sarkar of the Georgia Institute of Technology to receive the 2020 ACM-IEEE CS Ken Kennedy Award for his "foundational technical contributions to the area of programmability and productivity in parallel computing, as well as leadership contributions to professional service, mentoring, and teaching."

An ACM Fellow and an IEEE Fellow, Sarkar is chair of the School of Computer Science and

the Stephen Fleming Chair in the College of Computing at the Georgia Institute of Technology.

The Kennedy Award recognizes Sarkar's leadership in several areas. Sarkar made foundational technical contributions to programmability and productivity in parallel computing, and has developed innovative programming-model, compiler, and runtime technologies for parallel computing. Sarkar has led open source software projects that have had significant impact on the research community, has created new pedagogic materials to make

parallel programming more accessible to undergraduate students using the Coursera learner community, and has mentored junior colleagues and several Ph.D. students.

He also demonstrated leadership in community service by serving as program chair and general chair for major conferences in his research area, serving on U.S. Department of Energy's Advanced Scientific Computing Advisory Committee advisory committee, and on the Computing Research Association (CRA) Board of Directors.

The Kennedy Award carries a \$5,000 honorarium endowed by IEEE-CS and ACM.

ACM and IEEE-CS co-sponsor the Kennedy Award, established in 2009 to recognize substantial contributions to programmability and productivity in computing, and significant community service or mentoring contributions. It was named for the late Ken Kennedy, founder of Rice University's Computer Science program and a global expert on high-performance computing.

Coalition of the Willing Takes Aim at COVID-19

Data science can only do so much in the face of a pandemic.

THE RAPID SPREAD of COVID-19 around the world during the first quarter of 2020 spurred a massive response across the technological base, not least in computer and data science. Scientists and technologists both inside and outside healthcare snapped into action as the scale of the outbreak became clear, some providing techniques they had been working on for years, others proposing new projects all aimed at arresting the virus' progress.

The European Molecular Biology Laboratory's Bioinformatics Institute (EMBL-EBI), for example, already had a multiyear project underway to build a portal for anonymized genetic data from patients. Rolf Apweiler, co-director of EMBL-EBI, says it became clear at an early stage in the pandemic that those who suffered the most serious symptoms were "not only old people with underlying health conditions, but relatively young and healthy people. It is unclear why they are vulnerable and it may be in their genetic makeup. Understanding that is pretty important because if we want to go back to normal life, we want to find people who are vulnerable and need more protection."

According to Apweiler, what would normally take several years was compressed to a matter of months. By mid-April 2020, the group had opened an early implementation of the portal.

Before the pandemic got underway, warnings about a new epidemic came from data mining systems already in place. Social media technology provided the earliest clues to scientists working outside China, when Canadian company BlueDot and two research groups independently registered online chatter about a pneumonia-like disease at the end of December 2019. In internal reports, Chinese authori-



ties had noted the existence of a novel virus-borne disease only a few days beforehand.

As the first wave passed, suppressed in many developed nations by a broad-brush lockdown and social-distancing campaign, the loose coalition of technologists working on data-driven methods to combat the disease turned their attention to ways to build a smarter strategy for controlling the spread of SARS-CoV2, the virus that causes COVID-19.

Statistics from countries in the Far East that made extensive use of testing followed by interviews to trace contacts showed early on how effective those tactics could be without incurring the high economic costs of broadly applied lockdowns. At the end of March, a group of data scientists

working in Europe and the U.S. released a paper in which they proposed using mobile-phone data to help drive a public-health response. They argued failure to do so would be "missing an opportunity."

Whereas social media data of the kind used by BlueDot suffers from biases because it can only reflect the habits of highly engaged users, the prevalence of smartphone use in the population held the promise of delivering much better information at scale. Data from smart devices collected by Apple and Google showed how mobility dropped in the wake of the lockdown in various countries around the world. This data fed into increasingly detailed epidemiological models used to predict the rate at which the virus was expected to spread.

Although like many other governments, prime minister Boris Johnson's administration seized on the idea of using smartphones to make social-distancing and self-isolation more precise, the situation in the U.K. demonstrated a number of the key problems that lie behind any system that relies on consumer devices.

One is a conflict with rights to privacy. Used primarily to support a program of manual contact-tracing, South Korea's Corona 100m app was rolled out in early February and used location data from GPS to warn users if they were in close proximity to people who were infected. Though it has clear ramifications for personal privacy, the government had the power to require the app's usage, thanks to legislation passed in the wake of the 2015 MERS epidemic, together with the promise that such data would be deleted once the emergency passes.

In late April, several hundred scientists and researchers around the world published an open letter warning other governments of their concerns that similar apps would be launched with-

The U.K. government proposed holding contact data for as long as 20 years, which drew immediate criticism from researchers.

out safeguards and lead to the routine tracing of populations.

As the open letter noted, location-based systems have other problems as well. Systems like GPS and Wi-Fi-based triangulation do not have the accuracy required to detect close contacts reliably. The scientists and researchers instead recommended the use of Bluetooth, which came with two stated advantages. That technology's relatively short range makes it a reasonable proxy for proximity detec-

tion, but it does not rely on a reported physical location, so exchanges between nearby devices can be anonymized through frequent key or ID changes. If a user enters a positive test result into their app, one or more weeks' of key are sent to a cloud server. Other users' handsets periodically query that server for recently logged keys and pick up the infection data when they find a match.

In common with France, the British favored the creation of a tracking app based on a centralized architecture, where the keys and matched contacts are stored on a server for long periods, in contrast to an app design proposed by Apple and Google that only stores contact data on the handsets themselves. The U.K. government proposed holding contact data for as long as 20 years, which drew immediate criticism from researchers, not least because of the way it could put off users from downloading and using the app.

Michael Lewis, a professor of life science innovation at the U.K.'s University of Birmingham, says, "People

CACM News Brief

Spooing the Spoofer

Researchers at various universities have come up with cybersecurity software that tricks hackers into revealing the tactics they use to penetrate and control computer systems. Instead of blocking hackers, the software ingeniously invites hackers in, routes them to a decoy Web site or network, and then studies their behavior as they reveal their nefarious methods.

For example, the DEEP-Dig ((DE)CEPTION DIGging) software transforms hackers into "a source of free labor," says Kevin Hamlen, a member of the research team and Eugene McDermott Professor of Computer Science professor at the University of Texas at Dallas.

The ploy of using decoy Web sites and decoy networks to trick hackers has been in use by security administrators since around the turn of the century, according to Richard Forno, senior lecturer in the

department of computer science and electrical engineering of the University of Maryland, Baltimore County (UMBC), and assistant director of the UMBC Center for Cybersecurity.

Approaches to the security deception method vary, but the principle behind them remains the same: enable a hacker to penetrate your network, then trick him or her into thinking they are working with your actual network or data when in fact they are really working with a dummy network or dummy data.

Often, security deception software creates emulations of the inner workings of entire networks or Web sites in an attempt to fool hackers.

The difference with DEEP-Dig's approach to this principle is that it's powered by a deep neural network. Essentially, the software enables a security professional or system administrator to study and react

to, hacker activity with much greater sophistication, according to Reza Curtmola, a professor of computer science in the New Jersey Institute of Technology who specializes in cybersecurity.

Specifically, the DEEP-Dig system is able to do this by recording every point, click, and keystroke a hacker makes while trying to damage a dummy network or steal dummy data from a system. If the hacker is successful, the AI software takes note of the strategy the hacker used to overcome the system, then automatically sets up a defense against that strategy, ensuring it will not succeed in penetrating the system the same way a second time.

Says Shreyas Sen, an associate professor in the school of electrical and computer engineering of Purdue University who specializes in network security and efficiency, "This work adds another tool (in cybersecurity),

utilizing the recent advancement of deep learning."

"It is a good illustration of the 'defense in depth' technique, meaning cybersecurity solutions should have multiple layers of defenses," Curtmola says.

Chen Wang, an assistant professor in the division of computer science and engineering of Louisiana State University who leads its Mobile and Internet Security Lab, agrees. "By moving the Web attackers into decoys to continue studying their malicious activities, this method trains a better intrusion detection model by learning more insights into the attacks and adapts to the variants of the attacks."

—Joe Dysart is an Internet speaker and business consultant based in Manhattan, NY, USA.

Commissioned by CACM Staff

are justifiably concerned about how this data is going to be used.”

Wary of privacy issues themselves, Apple and Google said they would not support projects that had insufficient confidentiality guarantees. This effectively ruled out the use of the code in centralized architectures: the U.K. and France would each have to develop their own protocols to let phones talk to each other. Faced with the prospect of not having an app until late autumn, while Germany and other countries were rolling out software based on the Apple-Google protocol in late May and June, the U.K. government decided to switch to the decentralized architecture so it could make up some lost time by switching to the shared library.

Despite being relatively early to release its own Bluetooth-based TraceTogether app in mid-March, only a quarter of the Singapore population had downloaded and activated the app three months after its rollout. Some researchers pointed to the need for at least 60% of the population to use such an app to make it effective. However, Jason Bay, lead engineer on the project, stressed not long after its launch that an app like TraceTogether could only work as a backup to manual tracing. Weaknesses in the resolution of Bluetooth meant significant potential encounters would simply not register in many cases, and similarly could generate many false positives, such as “contacts” for phones connecting through walls and partitions. In the view of Bay and others, manual tracing based on interviews would yield many of the most important interactions, though they are significantly more expensive to administer.

Modeling showed other issues that would reduce the effectiveness of smartphone-based apps compared with more labor-intensive tracing methods. A report for the U.K. government written by the Royal Society’s Data Evaluation and Learning for Viral Epidemics (DELVE) group in late May amplified the point, arguing that an app’s primary benefit would be one of quickly communicating the possibility of close contact with an infectious person, but would have limited impact. The report emphasized how

Another course being pursued by digital technology researchers is to use the sensors in a smart device to detect symptoms as early as possible.

countries that deployed apps relied on a battery of measures of broader social-distancing measures. Simulations performed by the group indicated that the reduction in virus spread with an app used as the primary mechanism for virus control tops out at 15%. Isolation after users report COVID-19 symptoms, but before taking a confirmatory test, would yield a further 5% reduction, according to the predictions.

Simulations found the key issue lay in testing those most likely to be infected as quickly as possible. At the start of 2020, many countries that had not put into place pandemic-response programs faced testing bottlenecks that delayed the results. In response, governments have worked on improving test capacity, while some research teams try to find more streamlined diagnostics that could provide results in minutes rather than days, and so drive the response time down.

Another course being pursued by digital technology researchers, but one that inevitably takes time to establish as reliable, is to use the sensors in a smart device to detect symptoms as early as possible. Cecilia Mascolo, professor of mobile systems at the University of Cambridge, was leading a team working on one system that detects changes in the voice to detect cardiovascular problems, before switching to Covid-19 detection in the spring. “Being as it is a respiratory disease, we thought many of the symptoms could come out of audio samples,” she explains. “We can use microphones that are already embedded in what we carry:

it should allow scale-up to populations relatively affordably.”

Other teams have decided to focus on consumer wellness devices, such as FitBits and Apple Watches, that record heart rate and other physiological indicators. Duke University started recruiting owners of compatible devices for its CovIdentify project in April, with the aim of obtaining useable results before the end of 2020. They expanded the project in June to try to engage underserved communities by providing low-cost health-tracking devices to volunteers. Though such apps would not replace a test, they could form the basis of future track-and-isolate programs designed to take advantage of the extra reduction in virus spread scientists believe is possible compared to an isolation strategy based on medical-grade testing alone.

Though privacy concerns limit the quantity of information they will be able to use, as the pandemic gradually clears, researchers will have data from the most closely tracked outbreak of its kind to gauge the effectiveness of the many strategies countries have used to tackle Covid-19. But the pandemic has reinforced the notion that there are limits to what data science can achieve, despite the clear cost-savings and efficiency that technology promises. □

Further Reading

Oliver, N. et al
Mobile Phone Data and Covid-19: Missing an Opportunity?
 ArXiv preprint. arXiv:2003.12347

Kaafar, D. et al
Joint Statement on Contact Tracing
 Open letter: <https://cispa.saarland/de/2020/04/20/joint-statement-on-contact-tracing.html>

Sturniolo, S. et al
Testing, tracing and isolation in compartmental models
 MedRxiv preprint.
 DOI:10.1101/2020.05.14.20101808

He B., Zaidi, S., Elesedy, B., Hutchinson, M., Paley, A., Harling, G., Johnson, A., Teh, Y.W.
Effectiveness and Resource Requirements of Test, Trace and Isolate Strategies
 DELVE initiative Technical Document #3.
<https://bit.ly/31dFGT1>

Chris Edwards is a Surrey, U.K.-based writer who reports on electronics, IT, and synthetic biology.

© 2021 ACM 0001-0782/21/1 \$15.00



DOI:10.1145/3436231

Michael A. Cusumano

Technology Strategy and Management

Boeing's 737 MAX: A Failure of Management, Not Just Technology

Tracing the trajectory of management and engineering decisions resulting in systemic catastrophe.

ON NOVEMBER 18, 2020, the U.S. Federal Aviation Administration cleared the Boeing 737 MAX for flight, but the history of how Boeing got to this point remains disturbing.¹ Back in September 2020, the U.S. House of Representatives released a 238-page report on the 737 MAX debacle, concluding an 18-month investigation.⁵ The report blamed the two crashes in October 2018 (Lion Air, in Indonesia) and January 2019 (Ethiopian Airlines, in Ethiopia) on the computerized flight-control system called Maneuvering Characteristics Augmentation System (MCAS). The 737 MAX had been Boeing's fastest-selling plane in history before government authorities worldwide grounded the fleet of nearly 400 aircraft—but only after the second crash. A technical system failure was the proximate cause of the disasters, which cost

billions of dollars in losses to Boeing and the airlines, and, much more tragically, the lives of 346 passengers and crew.

Founded in 1916, Boeing remains one of the world's most renowned engineering companies. Were the 737 MAX crashes truly a failure of technology, an advanced aircraft-control system? Or was it a failure of management? Of course, at many levels, technology and management are inseparable. Nonetheless, executives, managers, and engineers at Boeing were not stumped by the complexity or unpredictability of a new technology. In a series of *decisions*, they put profits before safety, did not think through the consequences of their actions, or did not speak out loudly enough when they knew something was wrong. Let's look at the evidence.

We can start with Boeing's decision to deploy MCAS. The company wanted to put bigger, more fuel-efficient engines

on an older aircraft, the 737NG (Next Generation). Boeing was responding to intense competition from Airbus and demand from airline customers for more fuel-efficient, single-aisle planes. But the new engines significantly changed the pitch angle and stability of the older 737. Rather than redesign the plane, Boeing chose to install MCAS, which it adapted from another aircraft. The idea was that MCAS software would enable the 737 MAX to emulate the handling characteristics of the 737NG model by pushing down the front of the plane when sensor readings indicated the nose was too high. Sounds good.

The original MCAS design had two external "angle of attack" (AOA) air sensors, one on each of the outer sides of the aircraft. However, one sensor was cheaper and simpler, and that became the final design. Boeing engineers also continually increased the power of



A Boeing 737 MAX taking off.

MCAS to push down the nose of the aircraft, without changing assumptions about data and safety. In particular, the final design—with one sensor—assumed pilots could intervene if data was faulty or if anything else went wrong with MCAS. Yet, in 2015, Boeing documented MCAS was vulnerable to sensor failure.¹⁴ The external sensor was prone to damage from birds as well as errors in maintenance and calibration.⁹ A 2018 Boeing memo also revealed pilots had only four seconds to recognize an MCAS misfire and 10 seconds to correct it.¹³ Indeed, the day before the Lion Air crash, a maintenance worker had replaced a malfunctioning sensor. Lion Air did not relay to the pilots that crashed the next day the seriousness of the repair or details of a near-disaster on the prior flight, narrowly avoided with help from a third pilot who knew about MCAS and happened to be in the cockpit.⁵

Boeing decided pilots were the “backup” for MCAS, but the company did not explain in the 737 MAX operations manual how MCAS worked and how little time pilots had to respond. Why? Boeing had another objective: It wanted to treat MCAS and the MAX overall as an *incremental upgrade* in the

737 series. Why was that? The incremental designation allowed airlines to avoid spending millions of dollars on pilot training in new simulators. Meanwhile, Boeing was able to sidestep detailed scrutiny of MCAS and the 737 MAX by the FAA. The FAA also could depend on Boeing engineers to test and certify minor changes to the plane.

The congressional report had extensive access to company email and documents as well as detailed media coverage. These sources all describe the same decisions along with gradual but fundamental changes in Boeing’s strategy and culture.

First, was Boeing’s 1997 merger with McDonnell Douglas, a smaller aircraft maker with perilous finances. Usually, when a bigger company buys a smaller company, the culture of the bigger company dominates. Boeing was known for engineering excellence and safety, but McDonnell Douglas executives persuaded their Boeing owners to focus much more on costs, competition, and shareholder value (stock price). In essence, McDonnell Douglas took over Boeing, prompting one media comment that, “McDonnell Douglas bought Boeing with Boeing’s

money.”⁴ For example, McDonnell Douglas generally tried to upgrade older aircraft incrementally rather than build more costly new models from scratch. Boeing clearly followed this incremental strategy to create the 737 MAX.¹⁴

Second, was Boeing’s decision in 2001 to move its headquarters to Chicago from Seattle, where the company originated and had its primary engineering, manufacturing, and testing facilities for commercial aircraft. This move created physical distance between the leadership of the company and the technical teams focused on the 737 series. According to Boeing executives, the move was a strategic decision to separate management from the commercial aircraft division and to signal investors that Boeing was diversifying. In addition to commercial aircraft, headquartered in Seattle, Boeing now had McDonnell jet fighters, Douglas commercial aircraft, Hughes helicopters, and an aerospace division, all in different locations and easy to reach from Chicago.¹⁰

Third, was intensifying competition from Airbus, the European consortium founded in 1970 with backing from France, Germany, Spain, and the Netherlands. Today, Airbus is the world’s



Advertise with ACM!

Reach the innovators and thought leaders working at the cutting edge of computing and information technology through ACM's magazines, websites and newsletters.



Request a media kit with specifications and pricing:

Ilia Rodriguez
+1 212-626-0686
acmm mediasales@acm.org



largest aircraft manufacturer, ahead of Boeing because of a halt in 737 MAX production. But Airbus had briefly topped Boeing as number one in 2011, and it had a more competitive product in the same segment as the 737 MAX—the A320neo.⁶ Several European governments backing its main competitor probably put Boeing at a constant financial disadvantage. In addition, Airbus had a technical edge: It built the A320 series from scratch, first delivering planes in 1988. By comparison, Boeing retrofitted a much older 737 series, which first went to market in 1968.³

Fourth, was a change in priorities at the CEO and board of director levels. In 2005, James McNerney became the first Boeing chief executive not to be an engineer and he held this position until 2015. McNerney was a Harvard MBA who had worked at McKinsey and Proctor & Gamble before becoming president of GE Aircraft (which made jet engines) and then CEO of 3M. His expertise was in strategy and marketing, and he came in to improve financial performance. The 737 MAX development began in 2011, under McNerney's direction. The plane went into service in 2017 under another CEO, Dennis Muilenburg, who held this job from 2015 to 2019. Muilenburg was an engineer who had spent his entire career at Boeing. However, according to the current Boeing CEO, David Calhoun, Muilenburg carried on with McNerney's strategy and aggressively pushed sales and production of the 737 MAX.⁷ Boeing shareholders would later file lawsuits in June and September 2020 claiming that Muilenburg misled the board of directors about the seriousness of the 737 MAX problems while the board was lax in monitoring the design, development, and safety reports.¹²

In this highly competitive setting, and in a market completely dominated by two firms (their combined share is approximately 99%), Boeing executives, managers, and engineers made several critical decisions. In addition to the MCAS single-sensor design, in July 2014, Boeing decided that pilots experienced on earlier 737 models could fly the 737 MAX without new training on a simulator. Boeing made the same pledge to airline customers.¹¹ Boeing even offered to refund \$1 million per plane if more training proved necessary. Yet it was

We might also worry we have entered an era where software and hardware systems are so complex that government experts cannot independently certify technologies like Boeing put in the 737 MAX.

clear even before the first crash that the plane could be dangerous. Surely, some explanation of potential problems with MCAS called for a clearer warning to pilots about MCAS and the chaos that bad sensor data could create in the cockpit, or even grounding the aircraft after the first crash. Boeing and the FAA did send out notices after the first crash but they did not cite MCAS specifically or provide enough guidance to help the Egyptian crew avoid the second crash.⁸ Nor did Boeing or the FAA ground the aircraft after the second crash, or try to upgrade existing 737 simulators to replicate the MCAS behavior. To the contrary, after the two crashes, Boeing still tried to blame the accidents on “pilot error.”¹⁴

Another critical decision came in 2016, when Boeing decided to allow test pilots to stop flying actual 737 MAX planes and simply use flight simulators to continue testing. Not only did the simulators not properly mimic behavior of MCAS, but there was no simulation of what would happen with faulty data, which Boeing knew was a possibility. As a result, Boeing test pilots never actually tested a flying 737 MAX with a malfunctioning sensor. They never actually experienced what airline pilots in the two fatal crashes experienced.⁹

In an early design, Boeing also included an “AOA Disagree Alert,” telling pilots when the two angle-of-attack sensors disagreed in their readings. The Disagree Alert would have made pilots aware there was a potential data

problem. Boeing also allowed a supplier to tie the alert to an optional “AOA Indicator” display. Airlines were unaware of the importance of the indicator since there was no description of MCAS in the operations manual; most saw no need to pay extra for the alert option. As a result, 80% of the 737 MAX planes shipped without a functioning warning system that would have notified pilots of faulty sensor data.⁵


So what should we take away from this tragic story?

One lesson is that even the best companies can fall prey to competitive pressures as they seek to stay financially viable, grow faster, or profit by shipping products more quickly and cheaply. The venerable Toyota, often heralded as the world’s best manufacturing company, went through a similar period of overly ambitious growth and sloppy testing and quality control, which cost lives and billions of dollars.² One would think that aircraft manufacturers and automobile companies would never compromise safety for profits since they are, essentially, in the business of safe transport. This is not what happens in reality. The Boeing case also resembles the *Challenger* shuttle disaster in 1986. The pressure to launch led NASA managers to overrule engineers who were concerned about the safety of taking off in cold temperatures.¹⁵

Another lesson is we need governments to protect the public as well as to protect companies from themselves—from those competitive pressures that can lead to bad decisions. Lest we assume organizations can police themselves, or that engineers are good and managers bad, note the investigation produced email from Boeing engineers bragging they had “tricked” FAA regulators into believing no new training was necessary for the 737 MAX.¹⁴

We might also worry we have entered an era where software and hardware systems are so complex that government experts cannot independently certify technologies like Boeing put in the 737 MAX. For aircraft as well as automobiles, pharmaceuticals, food, banking, and many other products and services, governments rely mainly on companies to police themselves or to provide critical certification data. We allow “the fox to guard the henhouse,” so to speak. There is no easy solution to this problem, but,

at the least, government regulatory agencies need to be more diligent and hire more or better experts, and rely less heavily on what companies tell them. For their part, executives, managers, and engineers need to find a better balance between safety and cost. Faster and cheaper sounds great in the short term but can lead to disasters if the resulting products are not better or safer.

At least some people at Boeing knew there might not be enough time for pilots to react to an MCAS malfunction, yet the company decided not to inform pilots the system was operating behind the scenes or to provide simulator training. At least some people at Boeing knew MCAS was dangerous because one sensor constituted a single point of a potentially catastrophic failure. In short, the technology did not design itself or fail by itself, and that is why the 737 MAX debacle was primarily a failure of management. 

References

1. Chokshi, N. Boeing 737 Max is cleared by F.A.A. to fly again. *New York Times*, (Nov. 18, 2020); <https://nyti.ms/2UERbys>
2. Cusumano, M.A. Reflections on the Toyota debacle. *Commun. ACM* 54, 1 (Jan. 2011).
3. Duddu, P. Airbus vs. Boeing: A tale of two rivals. *Aerospace Technology*. (Jan. 31, 2020); <https://bit.ly/3pIfv00>
4. Frost, N. The 1997 merger that paved the way for the Boeing 737 Max crisis. *Quartz* (Jan. 3, 2020).
5. House Committee on Transportation and Infrastructure. Final Committee Report: The Design, Development, and Certification of the Boeing 737 MAX. (Sept. 2020); <https://bit.ly/2UDCSKp>
6. Katz, B. Airbus puts squeeze on Boeing’s 737 MAX as crisis drags. *Wall Street Journal*. (Feb. 13, 2020).
7. Kitroeff, N. and Gelles, D. It’s more than I imagined: Boeing’s new CEO confronts its challenges. *New York Times* (Mar. 5, 2020).
8. Levin, A. and Bloomberg News. After the 737 Max Crash, Why Did Boeing’s Pilot Warning Fail to Stop Second Plane from Going Down? *Fortune* (March 9, 2020).
9. Nicas, J. et al. Boeing built deadly assumptions into 737 Max, blind to a late design change. *New York Times* (June 1, 2019).
10. Ovens, A. Inside Boeing’s big move. *Harvard Business Review* (Oct. 2001).
11. Pasztor, A. Congressional report faults Boeing on MAX design, FAA for tax oversight. *Wall Street Journal* (Mar. 6, 2020).
12. Tangel, A. and Pasztor, A. Boeing board accused in lawsuit of lax oversight during 737 MAX crisis. *Wall Street Journal* (Sept. 25, 2020).
13. Tangel, A., Pasztor, A., and Maremont, M. The four second catastrophe: How Boeing doomed the 747 MAX. *Wall Street Journal* (Aug. 16, 2020).
14. The Fifth Estate. How Boeing crashed: The inside story of the 737 Max. (Jan. 19, 2020); <https://bit.ly/3pIfIkq>
15. Vaughn, D. *The Challenger Launch Decision: Risky Technology, Culture, and Deviance at NASA*. University of Chicago Press, Chicago, IL, USA, 1996.

Michael A. Cusumano (cusumano@mit.edu) is a professor and deputy dean at the MIT Sloan School of Management and founding director of the Tokyo Entrepreneurship and Innovation Center at Tokyo University of Science.

The author thanks MIT Sloan Professor Arnold Barnett for his comments.

Copyright held by author.

Coming Next Month in COMMUNICATIONS

**AZERTY Amélioré:
Computational Design
on a National Scale**

**GDPR Anti-patterns
and Cloud-Scale Design**

**Keeping Science on Keel
When Software Moves**

**Semantic Web:
A Review of the Field**

**Differential Privacy
Cryptography**

**Polanyi’s Revenge
and AI’s New Romance
with Tacit Knowledge**

**Cybersecurity: Is It
Worse than We Think?**

**Don’t Dumb Down
Computer Science
History**

**A Q&A with
Andrea Goldsmith**

**3D Localization of
Sub Centimeter-Sized
Devices**

**Efficient Signal
Reconstruction**

**Plus the latest news
about virtual reality
hardware, technology
responses to COVID-19,
and computer performance
after Moore’s Law.**

Security

Cybersecurity Research for the Future

Considering the wide range of technological and societal trade-offs associated with cybersecurity.

THE GROWTH OF myriad cyber-threats continues to accelerate, yet the stream of new and effective cyber-defense technologies has grown much more slowly. The gap between threat and defense has widened, as our adversaries deploy increasingly sophisticated attack technology and engage in cyber-crime with unprecedented power, resources, and global reach. We are in an escalating asymmetric cyber environment that calls for immediate action. The extension of cyber-attacks into the socio-techno realm and the use of cyber as an information influence and disinformation vector will continue to undermine our confidence in systems. The unknown is a growing threat in our cyber information systems.

Nonetheless, while the dark side is daunting, emerging research, development, and education across interdisciplinary topics addressing cybersecurity and privacy are yielding promising results. The shift from R&D on siloed add-on security, to new fundamental research that is interdisciplinary, and positions privacy, security, and trustworthiness as principal defining objectives, offer opportunities to achieve a shift in the asymmetric playing field.

Here, I will discuss three key considerations for cybersecurity research and development: interdisciplinary research themes, the role of experimentation in R&D, and education. Each of these will be the subject of future



columns as we focus on opportunities for dramatically different security and privacy in our daily lives.

Research Themes

The past 10 years have seen a move from R&D in purely defensive enterprise protection concepts to increasingly smart, autonomous, and reactive cybersecurity research. This movement away from boundary protection and after-attack analysis, to proactive autonomic systems has opened the door to new investigations and opportunities that are vital to future R&D. The shift in understanding attacks and vulnerabilities through research based

on increased understanding of threat techniques and increasingly sophisticated attack modes such as advanced persistent threats ransomware, and embedded system attacks provide the basis for next-generation research using AI and machine learning techniques.

The application of AI simultaneously creates new vectors for attacks and malfeasance while giving researchers new tools. New understanding and research into detecting, blocking, and managing misinformation/disinformation, detection of deepfakes and the associated automatically generated images/video/content/dialogue will have far-reaching impacts. A better marriage

between natural language understanding, human behavior, and network signals backed by AI will enhance information systems. At the same time, systems must be designed with an understanding of the threat space of adversarial attacks on machine learning models that will underlie so many mission-critical systems. Some of these research advances and techniques to manage trade-offs can be seen in a number of DARPA-funded research programs such as the Active Social Engineering Defense (ASED) program that is developing approaches to automatically identify, disrupt, and investigate spear-phishing and social engineering attacks. While the Cyber Hunting at Scale (CHASE) program is developing data-driven cyber-hunting tools for real-time cyber threat detection, characterization, and protection within DoD networks.

Studying broadly within our own disciplines is not enough. Cybersecurity is no longer solely an engineering discipline. It requires deep involvement from economists, sociologists, anthropologists, and other scientists to create the holistic research agendas that can anticipate and guide effective cyber-defense strategies.

Finally, we need innovations in data and information sharing between and across academia, government, and industry. One of the key impediments to research is the lack of real, validated data. There is an imbalance between the massive data collected and used by the Big Four (Apple, Facebook, Amazon, Google), industrial contractors and operational components, and that available to academic researchers. This is an issue that touches deeply on issues of privacy, security, and ethics, yet most of the needed advances in research increasingly rely on access to data of this type and scale to train and validate emerging AI and reasoning research.

A Science of Experimentation

Historically, cybersecurity R&D has struggled to prove its value in the commercial marketplace. The scientific basis for assessing the relative strength of theoretical and technological cybersecurity solutions often has been uncertain. This uncertainty has hampered technology transition and widespread cybersecurity adoption.

Cybersecurity is no longer solely an engineering discipline.

My research interests over the past two decades, are in the science of cybersecurity experimentation and next generation distributed experimentation methodologies. In my position as Director of the Networking and Cybersecurity Research Division at the Information Sciences Institute of the University of Southern California, I lead teams developing leading-edge cybersecurity research infrastructure for creating, testing, and evaluating the next generation of R&D. Our testbed technology, provides infrastructure, and methodologies and tools for cybersecurity experimentation. Our cybersecurity experimentation strategy is driven by the following key principles:

- ▶ Support experimentation and testing of hypotheses;
- ▶ Enable creation of repeatable, science-based experiments that can be validated by others;
- ▶ Generate research results that can be leveraged into broad, multi-component solutions in which components demonstrably support one another, making the whole greater than the sum of its parts;
- ▶ Foster methodologies and tools to help guide experimenters toward this new, scientific cybersecurity experimentation discipline; and
- ▶ Provide an open environment for researchers in industry, government, and academia to build on one another's achievements.

A central tenant of our research is enabling researchers to live in the future—allowing researchers to experiment with techniques and tools that do not yet exist and operate in environments only beginning to emerge. This allows highly capable, fluid new approaches to take shape. Living in the future also means enabling continuous R&D infrastructure gains. Our highly connected world is growing

exponentially in scale and complexity. Critical national assets and the threats to them evolve in tandem as well. While there are now various cybersecurity testbed experimentation facilities around the world, only a few are applicable to a wide range of experimentation, and almost none are openly available. Still, their existence is a valuable step toward research into a cross-disciplinary range of cybersecurity experimentation and testing methods and tools. In the future, we need an expansive ecosystem of experimentation laboratories along with clearinghouses and coordination centers to ensure widespread availability and use.

Looking forward, it is clear cybersecurity R&D must be grounded in the same systematic approach to discovery and validation that is routine in other scientific and technological disciplines. To approach these challenging research problems, we must create a paradigm shift in experimental cybersecurity. Only by enabling demonstrable, repeatable experimental results can we provide a sound basis for researchers to leverage prior work, and to create new capabilities not yet imaginable.

Education for the Future

Changing the asymmetric dynamics of cyberspace requires astute, knowledgeable researchers, educators, operators, users, and citizens. However, we are far from this goal. Rapid growth and spread of information technology, dramatically increased system complexity, and the multi-dimensional interdependence of these systems have left us woefully unprepared on many fronts.

The current dearth of cyber-professionals has sparked significant new federal training and education programs aimed at addressing this need. Among these initiatives are: the National Initiative for Cyber Security Education (NICE), the Scholarship for Service program, the National Centers of Academic Excellence in Information Assurance Education, and the Centers of Academic Excellence in Research. While these initiatives are beginning to increase the pipeline of cyber-professionals, their scale, pace, and depth so far are nowhere near sufficient to address the critical needs in the public and private sectors. The challenge now

INTERACTIONS



ACM's *Interactions* magazine explores critical relationships between people and technology, showcasing emerging innovations and industry leaders from around the world across important applications of design thinking and the broadening field of interaction design.

Our readers represent a growing community of practice that is of increasing and vital global importance.



To learn more about us, visit our award-winning website <http://interactions.acm.org>

Follow us on Facebook and Twitter  

To subscribe: <http://www.acm.org/subscribe>

Association for
Computing Machinery



It is clear cybersecurity R&D must be grounded in the same systematic approach to discovery and validation that is routine in other scientific and technological disciplines.

is to help organizations, locate and access programs suited for their needs.


While classroom study and early exposure to research provide foundational cybersecurity education, effective training also demands direct, hands-on involvement. Teaching cybersecurity is challenging. How do you demonstrate system weaknesses, inspire students to create constructive new solutions to vulnerabilities, and provide an environment in which they realistically can explore threat scenarios? We believe that undergraduates with direct cybersecurity experience are most likely to be eager to—and capable of—earning master's degrees. Similarly, graduate students who engage in science-based experimental research are most likely to develop the passion to pursue demanding doctoral and post-doctoral studies, and to obtain the academic positions that will enable them to continue developing the next generation of cyber-warriors.

To fundamentally change the cyber-threat dynamic, however, we need deep intellectual resources as well. These are represented by the brightest, best trained, and most curious and ambitious researchers and educators. Accordingly, we must be prepared to make significant investments in higher education. We must focus on educating the next generation of researchers and educators today so that we can we build the intellectual resources vital to solving tomorrow's problems. We are at serious risk of diminishing our academic programs

due to the number of graduate students and faculty who are lured to industry by astronomical salaries and promises of opportunity. While this trend is advancing commercial offerings, it will have a serious impact on our ability perform leading edge research and to educate and mentor the next generation.

However, the future challenges in emerging topics of AI, quantum, and IoT require that cyber education be much wider spread, more sophisticated, and accessible. Furthermore, the events of 2020 make it clear that we must address issues of diversity, equity, and inclusion in all levels of education. Only 20% of awarded U.S. computer science Ph.D.'s are women and only 3% of the awarded Ph.D.'s are people of color (Black, Hispanic, Native American). Computer science is lacking the involvement of 70% of the population, and thus we cannot hope to address the myriad of challenges in cybersecurity with such a lack of diversity

Summary

This is an exciting time to be a researcher in cybersecurity. The challenges facing the community are more complex than ever and changing at a rapid pace. In the face of these conditions, we are perhaps for the first time, in a position to draw on a wide range of interdisciplinary research themes to tackle these challenges. Artificial intelligence research has advanced in scale and complexity, and can take advantage of new computational support, and is now making regular contributions to the field of cybersecurity research. These advances along with important contributions from economists, sociologists, anthropologists, and other scientists are creating the holistic research agendas that will result in technology that can anticipate and guide effective cyber-defense strategies. I look forward to creating a forum for the community to explore these exciting developments and to debate the technological and societal trade-offs that will inevitably arise. 

Terry Benzel (tbenzel@isi.edu) is Director of Networking and Cybersecurity Research at the Information Sciences Institute of the University of Southern California, Marina del Rey, CA, USA.

Copyright held by author.

► James Grimmelman, Column Editor

Law and Technology

Content Moderation Modulation

Deliberating on how to regulate—or not regulate—online speech in the era of evolving social media.

DEBATES ABOUT SPEECH ON social networks may be heated, but the governance of these platforms is more like an iceberg. We see, and often argue over, decisions to take down problematic speech or to leave it up. But these final decisions are only the visible tip of vast and mostly submerged systems of technological governance.

The urge to do something is an understandable human reaction, and so is reaching for familiar mechanisms to solve new problems. But current regulatory proposals to change how social network platforms moderate content are not a solution for today’s problems of online speech any more than deck chairs were a solution for the *Titanic*. To do better, the conversation around online speech must do the careful, thoughtful work of exploring below the surface.

In September 2016, Norwegian author Thomas Egeland posted Nick Ut’s famous and award winning photograph *The Terror of War* on Facebook. The image depicts a nine-year-old girl running naked and screaming down the street following a napalm attack on her village during the Vietnam War. But shortly after it went up, Facebook removed Egeland’s post for violating its Community Standards on sexually exploitative pictures of minors.

Citing the photograph’s historical and political significance, Egeland decried Facebook for censorship. Because of his



moderate celebrity status, the photo’s removal quickly became global news. Facebook was rebuked by the Norwegian prime minister and in a front-page letter titled “Dear Mark Zuckerberg” *Aftenposten*, one of Norway’s main newspapers, chastised the site for running roughshod over history and free speech. In the end, Facebook apologized and restored Egeland’s post.

The incident served as a turning point, both for the platforms and the public. Though sites like YouTube,

Reddit, and Facebook had long had policies limiting the content users could post on their platforms, the enforcement of those rules was largely out of the public eye. For many users worldwide, *The Terror of War*’s high-profile removal was the first time they confronted the potential deleterious effects of the site’s censorial power. The incident was a foundational lesson not just in how difficult such decisions are but how high the stakes are if platforms get them wrong.



Digital Government: Research and Practice

Digital Government: Research and Practice (DGOV) is an Open Access journal on the potential and impact of technology on governance innovations and its transformation of public institutions. It promotes applied and empirical research from academics, practitioners, designers, and technologists, using political, policy, social, computer, and data sciences methodologies.



For further information
and to submit your
manuscript,
visit dgov.acm.org

In turn, the public backlash was a turning point in how Facebook operationalized its policies and their enforcement. When a post is flagged for removal by another user, the post is put in a queue and reviewed by a human content moderator to determine whether it does or does not violate the site's Community Standards. Those content moderators are typically off-site workers in the Philippines, India, or Ireland, reviewing incidents of flagged content in call centers 24 hours a day, 7 days a week.

The Terror of War photo violated Facebook's rule on nudity of minors, and thus removable, but it was also a picture of historical and newsworthy significance and thus an exception to removal. But historical value or newsworthiness are highly contextual and culturally defined—a difficult thing for someone from another culture, like a human content moderator might be, to recognize. It also introduced many to the opaque and unaccountable world of how private social media companies governed the public right of freedom of expression.

Since the Terror of War incident, we have had no shortage of reminders of the power of Big Tech and its lack of accountability to the users who rely on its services to speak and interact. Near-constant controversies about social media's impact on everything from political ads to violent extremism and from data protection to hate speech have led to various attempts at government regulation—some more successful than others.

In the U.S., the First Amendment prevents most legislative reform around privacy and hate speech. This is because privacy in America is typically understood as “protecting individuals from the dissemination of a particular piece of harmful information, or against particularly intrusive information collection,” which places potential “privacy laws in tension with the First Amendment's protection of free speech ...”¹ In the realm of hate speech, while the First Amendment does not protect violence or incitement to imminent lawless action, it does protect speech that might be offensive and reprehensible to some.

As a result, much of the impetus for reform has come from Europe.

Unburdened by the First Amendment, European jurisdictions have been able to enact broad privacy laws like the General Data Protection Regulation (GDPR), and state-specific anti-hate speech laws like the German NETZDG law.

That the First Amendment precludes sweeping bans on hate speech or dissemination of data, has not diminished the outcry in the U.S. Politicians and activists have largely focused their efforts on two goals. One is reforming Section 230 of the Communications Decency Act, a foundational law that prevents social media platforms from civil liability from suit from communications torts like defamation. The other is using antitrust law to break up, or at least rein in, technology companies. But the fervor for reform has not been matched with enthusiasm for the specifics. So far none of these proposals adequately address the technical realities of platforms' policies or their enforcement—an essential first step to take before tinkering with such powerful tools for democracy and speech.

Underlying these efforts is a claim that has gained significant traction over the last five years: that social media companies regulate speech in a politically biased way. Such charges come from both sides of the aisle, but frequently are missing key facts about the rules and processes behind keeping up or taking down content and accounts and grossly misunderstand the technical workings at play in large-scale commercial content moderation.

In the fall of 2017, for example, activists on the left raged when actress Rose McGowan's Twitter account was suspended during the start of the #MeToo movement. McGowan had posted a screenshot of an email from Bob Weinstein meant to demonstrate his awareness of the sexual abuse perpetrated by his brother, Harvey Weinstein. When McGowan posted her suspension notification from Twitter on Instagram with the caption “Twitter has suspended me. There are powerful forces at work. Be my voice,” the narrative of her suspension immediately turned into a story about the hammer of social media being unfairly wielded against women who speak truth to power and privileging the voices of those on the alt-right. “The game is rigged,” wrote journalist Chuck Wendig, “and seeing

@rosemcgowan getting suspended from Twitter, you don't have to ask for whom the game is rigged."

It is a common fallacy for humans to see a series of events occur and presume causality or even nefarious intent. But a closer examination of the events around such incidents adds nuance to these narratives. In the case of McGowan's suspension, her original screencap of Bob Weinstein's email had also included his personal phone number—which violated Twitter's rules prohibiting sharing other people's personal identifying information (also known as doxing). Ironically, this policy was the result of years of protest by the feminist community—many of whom had been victims of online abusers and trolls who had posted their home address or telephone number to encourage stalking or harassment.

McGowan's tweet was obviously not a call for harassment or abuse, but it also was obviously a violation of the letter of the policy—and her harsh punishment, suspension, was unfortunately what feminists and domestic violence advocates had long called for as remedy for violations of the policy. But few in the general public knew the intricacies of that rule—and even fewer knew the backstory that created it. Instead, McGowan's suspension immediately turned into a cause célèbre about the hegemonic silencing of women. Even after Twitter explained its policy and its mistake in suspending McGowan's account, outcry continued over Twitter's privileging of conservative or alt-right voices.

A similar but different story is true of those claiming social media is biased against conservatives, which reverses the causality of why posts are removed and mistakenly attributes removal to political animus. For the last decade or more tech platforms have met with complaints for not taking down enough harmful speech. Many pieces of content—like adulation for Hitler or white supremacy or calls for violence—were early, relatively easy things for sites to ban.

But since 2016, conservative politicians and media figures in the U.S. have made claims of anti-conservative bias in social media. They assert that sites unfairly remove or reduce distribution of their speech (although

Near-constant controversies about social media's impact on everything from political ads to violent extremism and from data protection to hate speech have led to various attempts at government regulation—some more successful than others.

multiple studies^a have shown no such discrimination²). In a Senate hearing on the issue in April 2019,^b platform representatives described the policies and enforcement mechanisms that had resulted in the appearance of conservative bias.³ And conservative political commentators like Alex Jones and Diamond & Silk, and Republican politicians Sen. Josh Hawley and Sen. Ted Cruz, and Rep. Martha Blackburn have all made such claims either after they or their constituents have content removed from social media. "[T]ech companies ... intentionally censor political viewpoints they find objectionable," claimed Cruz in a statement in 2019 after Twitter accidentally froze U.S. Senate majority leader Mitch McConnell's campaign account.

What is missing from the story, again, are the details. Hate speech comes in many different flavors, and rather than have long lists of specific ideas or language that should come down, platforms developed specific rules with elements—essential requirements that content must have to violate the policy. "We define hate speech as a

direct attack on people based on what we call protected characteristics—race, ethnicity, national origin, religious affiliation, sexual orientation, caste, sex, gender, gender identity, and serious disease or disability," states Facebook's policy rationale on Hate Speech in its Community Standards. This policy or a variation of it, has been in place since at least 2008, so it is not that Facebook is creating policies to ban Alex Jones because he is conservative, it is that when Alex Jones addressed Russia investigation special counsel Robert Mueller on his show and imitated firing a gun while saying, "You're going to get it, or I'm going to die trying" he ran afoul of Facebook's long-established standards on hate speech and incitement to violence. According to many people who work in Trust and Safety at these platforms, the reason the public is hearing more about more conservatives being removed from social media is not because of bias, but because a huge increase in the volume and extremism of "conservative" content.

In September 2019, cybersecurity expert Bruce Schneier gave a talk at the Royal Society in London. It was titled "Why technologists need to get involved in public policy?" but it could just as easily have been called "why public policy needs to get involved in technology." At the crescendo of his 15-minute speech, Schneier argued that "Technologists need to realize that when they're building a platform they're building a world ... and policymakers need to realize that technologists are capable of building a world." Schneier was ostensibly talking about cybersecurity, but his point speaks to the chasm in the middle of almost every technology debate raging today—including one of the most visible: the debate over how to regulate (or not regulate) online speech in the age of social media. □

References

1. Leta Jones, M. and Kaminski, M.E. *An American's Guide to the GDPR*, 98 Den. L. Rev. __ (2020).
2. Media Matters. Study: Facebook is still not censoring conservatives (Apr. 9, 2019); <https://bit.ly/3kDUuQR>
3. Stifling Free Speech: Technological Censorship and the Public Discourse. Senate Judiciary Subcommittee on the Constitution (Apr. 10, 2019); <https://bit.ly/35FXYTq>

Kate Klonick (klonick@gmail.com) is an Assistant Professor at St. John's University School of Law, New York NY, USA.

Copyright held by author.

a See <https://bit.ly/2UH3U3w>

b See <https://bit.ly/2UImXKI>

Historical Reflections

The Immortal Soul of an Old Machine

Taking apart a book to figure out how it works.

THE BEST BOOK ever written about IT work or the computer industry will be 40 years old in August. Tracy Kidder's *The Soul of a New Machine* describes the work of Data General engineers to prototype a minicomputer, codenamed "Eagle," intended to halt the advance of the Digital Equipment Corporation's hugely successful VAX range. It won both the Pulitzer Prize and National Book Award for non-fiction, perhaps the two highest honors available for book-length journalism. Year after year, the book continues to sell and win new fans. Developers born since it was published often credit it with shaping their career choices or helping them appreciate the universal aspects of their own experiences.

Soul's appeal has endured, even though what started out as a dispatch from a fast-growing firm building a piece of the future now reads as a time capsule from a lost world. Back in 1991 I read the book for an undergraduate class, typing my paper on a PC that was already more capable than Eagle yet cost 100 times less. So why are so many people still excited to relive the creation of a pitifully obsolete computer, designed by a team of obscure engineers for a long-forgotten company that never mattered very much anyway? Having spent almost 30 years now trying to take the book apart and figure out how it works, I think I have some answers. Ten of them, in fact.



1: It Does Not Assume You Know Anything

Paradoxically, the obscurity of Data General helps to explain the book's enduring power. My shelves are full of books about Microsoft, Apple, Netscape, and Oracle written while the companies were famous. Their authors assumed anyone who picked the book up was already fascinated with the company, cared deeply about its products, and would enjoy endless pages of gossip, corporate strategy, legal maneuverings, and trivia. They have not aged well.

In contrast, *Soul* delivers a self-contained package, containing everything you need to enjoy the story. Back in 1981 most potential readers had never used a computer of any kind, still less a "super minicomputer." Kidder would have had to self-publish any book written for Data General fans, and anyway he knew nothing about computers when he arrived in Westborough, MA, to follow up a suggestion from his editor. Kidder's only previous book was about a murder and his main life experience, other than a Harvard degree in English and

an MFA from Iowa, was having spent two unhappy but uneventful years as First Lieutenant, Military Intelligence in Vietnam.

2: It Reads Like a Classic American Novel

The book was a milestone in the development of what is now called “literary nonfiction.” To keep us turning pages, Kidder draws deeply on the mythic archetypes of American literature, as in his introduction of Tom West, his protagonist and the project’s leader, during a prologue that recalls Ernest Hemingway: West awes the other crew members of a small sailing boat with his stamina and ruggedly taciturn optimism when hit by a storm. “Whatever he did for a living,” they conclude, “it was probably interesting and obviously important.”

Having hooked us on the enigmatic Tom West, Kidder is cocky enough to spend an entire chapter without mentioning him, instead introducing Data General as “the Darth Vader of the computer industry” (a reference that undoubtedly aged better than he expected). Data General’s corporate culture was defined in equal parts by thrift and aggression. Kidder confides that Data General’s spartan corporate offices were engineered for rapid conversion to factory space even before he gets around to mentioning the lawsuit a rival firm filed to accuse Data General of burning down its factory.

When West reappears, Kidder stimulates our curiosity by presenting him as a figure of mystery to his own team members: a CIA agent, a folksinger, a speed freak, even “a prince of darkness.” I was struck by the similarities with the technique F. Scott Fitzgerald used in *The Great Gatsby* to introduce his title character. Jay Gatsby, urbane host of fabulously swanky parties, turned out (spoiler alert) to be plain old James Gatz of North Dakota, a lovelorn bootlegger desperate to recapture the attention of his lost sweetheart. Less dramatically, corporate computer engineer Tom West, turns out to be Joseph Thomas West III, an engineer from a privileged background who came late to corporate life after taking a year off during college to play folk music, followed by seven slightly offbeat years building and delivering digital clocks for the

Many readers were tantalized by the idea of computer architecture as a creative medium in which experts could read traces of individual flare or, as here, a conservative organizational culture.

Smithsonian Observatory. That doesn’t quite justify the build-up, but whatever: we are already hooked. Confessing to his editor that he was having difficulty capturing West, “whose special vanity had been to make himself mysterious to me as well as to his team of computer engineers,” Kidder had been advised to “do a Gatsby on him.”⁴

3: It Roots for the Underdogs

Gripping stories often ask us to root for underdogs to triumph against the odds. Data General’s brutal corporate culture gave West space to launch his project but deprived his team of resources, leaving them in cramped and uncomfortable conditions. Even pencils were in short supply. If you are thinking that a major minicomputer firm ought to have been able to provide pencils to the elite team building its next-generation system, you’d be right. However, that team and its pencils were down in North Carolina, designing an ambitious all-new 32-bit architecture on a clean sheet of paper. Data General had caught a bad case of what Fred Brooks called the “second system syndrome.”⁴ The tactic of throwing out backward compatibility for a new architecture worked for IBM with its legendary System/360 gamble, but has failed far more often, for example with IBM’s own Future Systems project, at least three times for Intel (iAPX 432,

80960, and Itanium) and even DEC’s much-admired Alpha processor.

The clean sheet approach failed for Data General too. Anticipating this, Tom West rounded up the best of the engineers left behind in Massachusetts to launch a semi-clandestine effort to produce a 32-bit extension of the existing 16-bit Eclipse minicomputer, preserving compatibility by interleaving old and new instructions seamlessly rather than using a “mode bit” to enter a separate legacy mode. (More than 20 years later, AMD played a comparable trick on Intel by extending the standard x86 architecture to 64-bits). For the project to be approved as “insurance” against problems with the North Carolina team, West had to promise the impossible task of producing the entire computer in one year. Thirty engineers crowded together, turning the basement of Data General’s headquarters into a site of relentlessly hard work.

The team’s outsider triumph gives way to an unexpectedly mournful conclusion. West himself is banished to a marketing job in Japan. Kidder, winking at his name, compares West to a gunslinger, who dispatches the bad guys only to be run out of town by the very citizens he saved. “It was a summer romance,” realizes West. “None of it came out the way he had imagined it would, but it was over and he was glad.”

4: It Captures the “Crunch” of Startup Development

Although Data General was a mature company, the project was run more like a startup. West and his lieutenants staffed the team with young men fresh from engineering school, lured with the prospect of being able to design a new computer architecture. They boasted of being “a place where people are really doing the next thing” but cautioned that “there’s a lot of fast people in this group ... a real hard job with a lot of long hours.” In short: “tell him that we only let in the best. Then we let him in.” It’s a classic example of what software engineering writer Ed Yourdon called the “marine corps” justification for a “death march project.”⁷ As Kidder put it, “It was kind of like recruiting for a suicide mission. You’re gonna die, but you’re gonna die in glory.”

Eagle is brought to life more slowly than the team had promised but sooner than Data General had any right to expect. Inexperienced recruits were manipulated into “signing up” to aggressive schedules, because an unreasonable commitment given freely motivates more deeply than one imposed by management. “Signing up required, of course, that you fervently desire the right to build your machine and then you do whatever was necessary for success, including putting in lots of overtime, for no extra pay.” The novice engineers are granted large responsibilities and the freedom to follow their instincts. Young men “dribble away” pieces of their lives as they battle to prove themselves. Some wilt under the pressure; those that remain work frantically and effectively. The hardware team (“The Hardy Boys”) and the microcode developers (“The Mikrokids”) battle constantly and informally against each other to add and remove hardware capabilities from the specification. Together, they take the computer from conception to prototype hardware in six months. Then they have to make it work.

What was the “soul” referenced in the book’s attention-grabbing title? Natalie Angier, an early reviewer, claimed that the “soul of a new machine, says Kidder, is nothing more than the collective soul of those who put the machine together.” That’s plausible, though Kidder notably declined to “say” this directly. The closest he comes, which is not very, is describing a home workshop, full of power tools and carpentry equipment, as “a window on West’s soul”? Nevertheless, his title made me think of the old story of the Golem, animated by magic but created by, and enslaved to, human will. As one of the engineers explained, “I don’t have to get official recognition for anything I do. Ninety-eight percent of the thrill comes from knowing that the thing you designed works, and works almost the way you expected it would. If that happens, part of *you* is in that machine.” Writing on the front page of the *New York Times Book Review*, Samuel C. Florman called the engineers “fanatics, but not purists.”³ Perhaps their many individual sacrifices, of marriages, mental stability, youth, and health, amounted to a ritual through which sundered fragments of

their own souls accumulated to bring the new machine to life.

5: It Gets to Technology Through People

How to make a reader commit to almost 300 pages of finely observed business history of a company they hadn’t heard of, focused on the creation of a computer they would never see, interspersed with technical descriptions of microcode, caching, and instruction formats that might have been more at home in a computer architecture textbook than a gripping best seller? Kidder succeeds by telling us first about people, not about machines, investing us enough in them and their work to follow as he moves deep into descriptions of the problems they were grappling with. People have changed much less than computers over the last 40 years so this material remains gripping today.

When reading or watching the stories of the computer industry’s most successful men we know all the endings. Each retelling of the story of Steve Jobs is like riding a rollercoaster: we hurtle along a fixed track past expected triumphs and tragedies. In contrast, Kidder tells an unfamiliar story, tightly focused in time and place yet larger than any of its individual players.

In the book’s most famous passage, West has a friend sneak him into a data room where a VAX is installed to get a feel for the machine he is trying to beat. Lifting the covers of its central processing unit he counts 27 printed circuit boards: “Looking into the VAX, West had imagined he saw a diagram of DEC’s corporate organization. He felt that VAX was too complicated. He did not like, for instance, the system by which various parts of the machine communicated with each other; for his taste, there was too much protocol involved. He decided that VAX embodied flaws in DEC’s corporate organization. The machine expressed that phenomenally successful company’s cautious, bureaucratic style.”

Many readers were tantalized by the idea of computer architecture as a creative medium in which experts could read traces of individual flare or, as here, a conservative organizational culture. Most chapters tell us a piece of the story from the viewpoint of engineers responsible for design-

ing a part of the machine or for running the debugging process, introducing a rich cast of clearly delineated supporting characters with their own quirks and motivations.

One such chapter, “The Case of the Missing NAND Gate,” begins by introducing several engineers. Kidder sketches the lives, habits, and appearances of Ken Holberger (“Chief Sergeant Detective of the Hardy Boys” who “couldn’t look messy if he tried” but “doesn’t waste time listening to people who aren’t making good, relevant sense”), Jim Veres (whose “stern glare ... makes some people nervous. His managers’ confidence in him is tempered only by their feeling that he works too hard. That is how they express it”) and Jim Guyer (an asthmatic mountaineer who “seems in his busyness, among the happiest of the group”). Kidder follows their interactions while troubleshooting a problem, observing the feelings of each toward possible flaws in their own boards. We care about the bug because we see how much it matters to these engaging characters and to the unseen narrator who leads us confidently through passages such as “The diagnostic program originally puts the target instruction at address 21765, and then, sometime later on, it moves the target instruction to 21766. But the IP never gets word of the change, though the System Cache does.”

By the end of the book we know about microcode, bits and bytes, Boolean algebra, what happens when an instruction is executed (in some detail), memory management, debugging, diagnostics, emulation, and the Adventure game. In his *Times* review, Florman noted that while these descriptions “did not significantly increase” his “own very superficial knowledge” the “uninitiated will find these brief passages abstruse but not bewildering, unfathomable but not boring.”

Kidder’s ability to hold our interest is aided by another structural similarity with Scott Fitzgerald’s masterpiece: a narrator defined primarily by his obsession with the man of mystery. Kidder acknowledges his own ongoing presence with phrases such as “I saw them all collected once ... during a fire drill,” or “West said years later,” or “I saw him at one of the team’s parties” but refuses to make himself a character.

That is a contrast with the work of flashier proponents of 1970s “new journalism” such as Tom Wolfe, author of vivid accounts of the American space program and the worlds of car enthusiasts, or Hunter S. Thompson who made his own erratic behavior the center of every story. Kidder has called this the “first-person minor” or “reasonable person” technique of narration, in which “not much about the narrator is revealed, including the narrator’s opinions.”⁴ Stripping his in-book presence of most identifying marks, to leave a person-shaped avatar, helps us to imagine ourselves in Kidder’s empty shoes as they move through the basement or follow West home. We come to identify with the narrator’s only defined characteristics: initial ignorance, growing fascination, and dogged pursuit of understanding.

6: It Exposes the Materiality of Computing

Kidder calls Data General’s products “machines” as often as “computers” and chose the word “machine” for his title. In 1979, computers were built on a scale where engineers could probe and rewire each logical pathway, giving Kidder something material to describe. That is a contrast to our current discourse in which “the digital” is assumed to be invisible and immaterial. Early minicomputers like DEC’s PDP-8 and Data General’s Nova, cheap and small in comparison to mainframes, were made possible by a stream of innovations to package and assemble electronics more efficiently. By 1979, the Eagle team was building computer logic mostly out of chips, rather than discrete transistors and resistors. Yet despite its innovative use of Programmable Array Logic chips for custom logic the Eagle’s central processing unit still filled many circuit boards.

Kidder emphasizes continuities between tinkering with broken machines, a common activity in the 1970s, and the work of the engineers as they closely observe Eagle’s functioning with logic probes, adding wires or tweaking circuits to fix tiny errors in the design. West boasted “I can fix anything,” which Kidder documents for a diesel engine, televisions, clocks, furniture, a record player, and a house. “What that

Further Reading

I will be tackling two other landmark studies of the culture of IT work, Steven Levy’s *Hackers* and Ellen Ullman’s *Close to the Machine* in later columns, along with the recent TV series *Halt and Catch Fire*. In the meantime, *Soul* may leave you hungry for closely observed studies of development projects.

I can recommend Fred Moody’s *I Sing the Body Electronic* (Viking Penguin, 1995), about a year spent with a Microsoft group developing multimedia CD products, and G. Pascal Zachary’s *Show Stopper!* (Free Press, 1994) about the creation of Windows NT. Michael Lewis is an outstanding writer. Sadly *The New Thing* (Norton, 1999), which follows Netscape co-founder Jim Clark as he pitches half-baked healthcare marketplace and builds a giant robot yacht, is not one of his best but it is still readable and insightful.

Writers have seemed less likely to embed themselves in computer projects recently, perhaps because companies have become increasingly secretive. *Masters of Doom* (Random House, 2003) is based on interviews rather than direct observation but still gives a vivid account of the testosterone-soaked development of breakthrough action games *Doom* and *Quake*. Jason Schrier’s *Blood Sweat and Pixels* (Harper Paperbacks, 2017) is a collection of grim short stories versus Kidder’s elegiac novel, with each chapter looking at a different video game project.

Thomas A. Bass’s *The Eudaemonic Pie* (Houghton Mifflin, 1985) tells the inside story of an eccentric Californian team building shoe-mounted computers and radio links to predict the destination of a ball bouncing on a roulette table. For memoirs focused on product-based startups, try Jerry Kaplan’s *Startup* (Houghton Mifflin, 1994) about a pen computing pioneer crushed by Microsoft, Charles Ferguson’s judgmental *High Stake, No Prisoners* (Crown Business, 1999) about a web editing package sold to Microsoft, and Michael Wolff’s *Burn Rate* (Simon & Schuster, 1998) in which a gossipy journalist describes his largely unsuccessful efforts to obtain millions of dollars from venture capitalists. Reflecting the mood of their era, all three focus more on finance and management than on engineering.

thing was,” Kidder continues, “whether a car’s engine of a computer, did not matter; but since computers were among the most complex of all man-made things, they had seemed to him, he said, to pose interesting challenges.” The main story ends when the prototype is “wheeled down the hall to Software.” Kidder barely mentions this less-tangible side of the project, which accounted for more than half of the total development work.

A single Eagle would sell for a quarter of a million dollars and could support dozens of simultaneous users, each on a separate video terminal. Mass-produced computers, including the Apple II and TRS-80, had been sold to consumers and hobbyists since 1977. But *Soul* ignores them, as did DEC and Data General during that period. Personal computers still seemed like toys, and the chip technology of the era was several years away from being able to create a high-performance 32-bit microprocessor. More fundamentally, the shift to standard processors stripped the heart out of computer engineering. A few years later, West spoke dismissively of “all these people who are putting 68000s on a board and calling it a computer.”⁵

7: It’s Unashamedly Masculine

This is, as you have surely already realized, a remarkably and unself-consciously masculine book—which helps it appeal to men and to those comfortable with the dominant traditions of American literature. While computers were new and unfamiliar, American literature had a long tradition of celebrating the rugged masculinity of civil engineers taming the Western landscape and the resourcefulness of pioneers able to fix or adapt machinery to their needs. Such men were also expected to be taciturn, emotionally restrained, and hard to know. Perhaps overcompensating for the growing association of computing with nerds, Kidder celebrated men doing guy things to an extent that must have seemed old-fashioned even in 1981.

West draws an organization chart on a whiteboard, then puts an X over a rival manager, saying “This guy disappears in time.” The Eagle group themselves come to feel like throwbacks to an earlier, less bureaucratic kind of engineering, joking about ordering dinosaur T-shirts for the team and complaining that “beating people up didn’t seem to get results anymore.” After all the mythologizing, it was a

Distinguished Speakers Program

A great speaker can make the difference between a good event and a WOW event!

Students and faculty can take advantage of ACM's Distinguished Speakers Program to invite renowned thought leaders in academia, industry and government to deliver compelling and insightful talks on the most important topics in computing and IT today. ACM covers the cost of transportation for the speaker to travel to your event.

speakers.acm.org



Association for
Computing Machinery

Computing was a sideshow 40 years ago. Today Apple, Amazon, Microsoft, and Alphabet are the first trillion-dollar companies. Technology and money are inseparable.

shock when I found a YouTube video of West helping to introduce a successor to the Eagle in 1990: an apparently charisma-free middle aged engineer droned through technical specifications and corporate jargon, sweating in his short-sleeve shirt and tie.

Their jobs required them to sacrifice or downplay any commitment to family or human relationships. Kidder mentions in passing that one female engineer was hired, Betty Shanahan. We learn only that her husband was unhappy to be left doing the laundry and that she was given a joke award for “putting up with a bunch of creepy guys.” Eventually Shanahan got tired of putting up, becoming an advocate for diversity as executive director of the Society of Women Engineers. Today there might still be only one woman on the team, but a modern author would surely center a chapter on her.

We learn far more about Rosemary Searle, the project's secretary and surrogate mother to its young men. She tells Kidder that West “never put one restriction on me ... he let me go out and see what I could get done.” When I wrote about the creation of ENIAC, Kidder's sensitivity to Searle's contributions reminded me to highlight what little information I could find on Isabelle Jay, its secretary and its longest serving full-time member.

8: It Dramatizes Ordinary Engineering Work

Eagle fared well after it reached market, as the Eclipse MV/8000. It and its successors sustained Data General for years. The company was saved, kind of, though it never returned to its glory days of industry leading growth rates and profit margins. Before long the entire minicomputer industry was crumbling. In 1999, Data General was gobbled up by EMC for its storage technology. By then even DEC, once second only to IBM, had concluded its slow decline with absorption by PC manufacturer Compaq. In time, all empires turn to dust.

The work of these engineers is challenging and difficult to understand, but Kidder treats them as skilled practitioners of a difficult craft rather than world-shaking genius innovators. You might complain that Kidder wasted his talent on the wrong story, that he should have spent the late-1970s lurking in a garage in Silicon Valley rather than a basement in Massachusetts. If you have heard of Steve Wallach, Carl Alsing, Ed Rasala, or even Tom West himself it is almost certainly because of their appearance in the book, despite their many accomplishments at firms like Convex and Alliant. Personally, I am glad Kidder told the story he did, looking at a part of the computer industry that was far larger at the time and remains more representative of engineering practice.

Novelists know that ordinary lives are full of hidden drama, but most technology journalism chases stories of exceptional success. Reading tributes to the book by engineers, I am struck by how often they note triumphs and tragedies in their own careers that parallel those experienced by Kidder's characters. One was hit by a visceral sense of “grim familiarity” when he encountered a passing reference to the killing of a beloved project during a “big shootout at HoJo's.” “If you haven't yet had your own shoot-out at HoJo's,” he warned, “it is regrettably coming; may your career be blessed with few such firefights!”² Perhaps Kidder was really describing himself when he noted that West “was always finding romance and excitement in the seemingly ordinary.” His next book, after all, discovered equal wonder in the building of a single house.

9: It Makes Engineers Seem Pure and Noble

Kidder invokes Victorian critic John Ruskin's romantic idea that in building Europe's great cathedrals, ancient craftsmen experienced "the sort of work that gave meaning to life." According to Kidder, the engineers likewise "did the work, both with uncommon spirit and for reasons that, in a most frankly commercial setting, seemed remarkably pure." None of Kidder's characters become spectacularly rich or expected to, though they had hoped vainly for some financial recognition. When the team's success went unrewarded with stock options or bonuses, Kidder likened the rewards of computer engineering to those of pinball: the only thing you can win is a chance to play the game again. Key members of West's team leave Data General after he is banished to Japan, looking to play their next game of pinball elsewhere.

Late in the book, a unionized technician drops his pay stub into a trash basket. A senior engineer thereby discovers that the technician is making twice his own pay, thanks to overtime. His supervisor burns the evidence, "so that the troops wouldn't see it." The sacrifices of the engineers seemed even purer in contrast with the sales manager whose declaration on the final page that humans are motivated by "ego and the money to buy things that they and their families want" reads like blasphemy. Kidder finishes the book with: "It was a different game now. Clearly the machine no longer belonged to its makers."

Computing was a sideshow 40 years ago. Today Apple, Amazon, Microsoft, and Alphabet are the first trillion-dollar companies. Technology and money are inseparable. Instead of pinball, a technology career is more like a slot machine where the goal is to pull on the handle repeatedly and hope to win a financial jackpot. Intel routinely granted stock options to engineers. Adopted by software firms like Microsoft, this became the standard way of luring engineering talent to successful companies, creating millionaires in unprecedented numbers. Other developers, seeking a longer shot at greater wealth, sought stakes in the startups that dominant firms increasingly treated as a source, via acquisition, of products and staff.

The founders and early investors in Data General got rich, but not the engineers who sustained its growth. Were they noble, or just exploited? Kidder may have romanticized the motivations of his characters. Twenty years ago, *Wired* magazine found most of them working in senior roles at startups. Some had become rich.⁶ Yet I am myself just romantic enough to fear that something important was lost. Tom West's melancholy pride at the end of the book is surely more representative of the experience of most development teams than the world-changing success and unimaginable riches that dominate the more familiar stories of Gates, Jobs, and Zuckerberg. In fact, many teams are disbanded before their work is done. Almost every component part of a software or hardware system is invisible to the world, the quality of its execution and elegance of its design known only to its creators. If a system passes into the world the quality of that work will be one of many factors deciding whether it thrives or is quickly forgotten. Systems are often doomed by bad marketing, undercapitalization, changing customer tastes, or an idea that was ahead of its moment. I hope that today's developers retain enough of the old ethic of pinball to find an intrinsic satisfaction in difficult work well done, so that they don't feel worthless if the industry eats their youth without paying out a financial jackpot.

10: It Is Beautifully Engineered

Above all, *Soul* is an extraordinarily well-crafted book. That means more than just well-turned sentences and snappy observations. Each finely tuned section fits smoothly into the structure of the book. Kidder's pacing is flawless, his character beats impeccably timed, and he manages to make a mass of contradictions seem like a faithful portrait of a complex world rather than a failure of craft. West remains unknowable and paradoxical. The engineers are both exploited and given an enviable opportunity for meaningful work. The project was a rebellion, tacitly orchestrated by senior managers.

Kidder's only previous book was, in his own estimation, a miserable failure. Kidder has described his 1970s self as "plainly ambitious" yet "young beyond his years." As a journeyman freelance magazine writer, he churned out words with "boundless energy" but

had no idea how to shape them into a publishable story. Richard Todd, his implausibly sympathetic editor at *The Atlantic Monthly* (and West's former college roommate) imposed many rounds of rejection and revision while Kidder fitfully learned his trade.

How, then, could Kidder suddenly produce a masterpiece? Kidder dedicated *Soul* to Todd, who undoubtedly deserved it. Yet Kidder's sudden growth as a writer owed something to his immersion in the culture of engineering. In 1982, newly laden with accolades, he appeared with West at the Computer Museum in Boston. "In some sense," Kidder explained, "writing a book is like building a computer."⁵ He had witnessed the profoundly creative nature of the design process, the aesthetic qualities of good engineering, and the pure joy of finding an elegant solution to a hard problem. As *Soul* took shape, Kidder spent a long confinement in Todd's office, spreading typewritten draft pages in piles on its floor to prune and rearrange them. Todd sometimes lifted prized passages to deliver the crushing news: "you could do without this." Kidder then "began to learn a skill ... how to fall out of love with my own words," so that he could eliminate good material that was "at odds with the whole."

West's team had likewise begun with architectural decisions, ruthlessly subsuming the parts to the whole. Studying the engineer's craft, Kidder had learned something about his own. His unique fusion of engineering and literature has outlasted the MV/8000 and the mini-computer industry. I expect it to outlast the PC and the smartphone too. ■

References

1. Brooks, F.P., Jr. *The Mythical Man Month: Essays on Software Engineering*. Addison-Wesley, Reading, MA, 1975.
2. Cantrill, B. Reflecting on *Soul of a New Machine*. The Observation Deck (blog), (Feb. 10, 2019); <https://bit.ly/35KX3fN>
3. Florman, S.C. *The Hardy Boys and the Microrikids Make a Computer*. *New York Times Book Review* (Aug. 23, 1981), 1.
4. Kidder, T. and Todd, R. *Good Prose: The Art of Nonfiction*. Random House, NY, 2013.
5. Kidder, T. and West, T. Inside 'The Soul of a New Machine'. *The Computer Museum Report* (Spring 1983), 5–8.
6. Ratcliffe, E. O. Engineers! *Wired* (Dec. 2000), 356–367.
7. Yourdon, E. *Death March: The Complete Software Developer's Guide to Surviving "Mission Impossible" Projects*. Prentice Hall, 1997.

Thomas Haigh (thomas.haigh@gmail.com) is a professor of history at the University of Wisconsin—Milwaukee and a Comenius visiting professor at Siegen University. His next book, *A New History of Modern Computing* will appear with MIT Press later this year.

Copyright held by author.

Viewpoint

Insights for AI from the Human Mind

How the cognitive sciences can inform the quest to build systems with the flexibility of the human mind.

WHAT MAGICAL TRICK makes us intelligent? The trick is that there is no trick. The power of intelligence stems from our vast diversity, not from any single, perfect principle.

—Marvin Minsky,
The Society of Mind

Artificial intelligence has recently beaten world champions in Go and poker and made extraordinary progress in domains such as machine translation, object classification, and speech recognition. However, most AI systems are extremely narrowly focused. AlphaGo, the champion Go player, does not know that the game is played by putting stones onto a board; it has no idea what a “stone” or a “board” is, and would need to be retrained from scratch if you presented it with a rectangular board rather than a square grid.

To build AIs able to comprehend open text or power general-purpose domestic robots, we need to go further. A good place to start is by looking at the human mind, which still far outstrips machines in comprehension and flexible thinking.

Here, we offer 11 clues drawn from the cognitive sciences—psychology, linguistics, and philosophy.

No Silver Bullets

All too often, people have propounded simple theories that allegedly explained all of human intelligence, from



behaviorism to Bayesian inference to deep learning. But, quoting Firestone and Scholl,⁴ “there is no one way the mind works, because the mind is not one thing. Instead, the mind has parts, and the different parts of the mind operate in different ways: Seeing a color works differently than planning a vacation, which works differently than understanding a sentence, moving a limb, remembering a fact, or feeling an emotion.”

The human brain is enormously complex and diverse, with more than 150 distinctly identifiable brain areas, approximately 86 billion neurons, hundreds if not thousands of different types; trillions of synapses; and hundreds of distinct proteins within each individual synapse.

Truly intelligent and flexible systems are likely to be full of complexity, much like brains. Any theory that proposes to reduce intelligence down to a single principle—or a single “master algorithm”—is bound to fail.

Rich Internal Representations

Cognitive psychology often focuses on *internal representations*, such as beliefs, desires, and goals. Classical AI did likewise; for instance, to represent President Kennedy’s famous 1963 visit to Berlin, one would add a set of facts such as part-of (Berlin, Germany), and visited (Kennedy, Berlin, June 1963). Knowledge consists in an accumulation of such representations, and inference is built on that bedrock; it is

trivial on that foundation to infer that Kennedy visited Germany.

Currently, deep learning tries to fudge this, with a bunch of vectors that capture a little bit of what's going on, in a rough sort of way, but that never directly represent propositions at all. There is no specific way to represent visited (Kennedy, Berlin, 1963) or part-of (Berlin, Germany); everything is just rough approximation. Deep learning currently struggles with inference and abstract reasoning because it is not geared toward representing precise factual knowledge in the first place. Once facts are fuzzy, it is difficult to get reasoning right. The much-hyped GPT-3 system¹ is a good example of this.¹¹ The related system BERT³ is unable to reliably answer questions like “if you put two trophies on a table and add another, how many do you have?”⁹

Abstraction and Generalization

Much of what we know is fairly abstract. For instance, the relation “X is a sister of Y” holds between many different pairs of people: Malia is a sister of Sasha, Princess Anne is a sister of Prince Charles, and so on. We do not just know that particular pairs of people are sisters, we know what sisters are in general, and can apply that knowledge to individuals. If two people have the same parents, we can infer they are siblings. If we know that Laura was a daughter of Charles and Caroline and discover Mary was also their daughter, then we can infer Mary and Laura are sisters.

The representations that underlie cognitive models and common sense are built out of abstract relations, combined in complex structures. We can abstract just about anything: pieces of time (“10:35 PM”), pieces of space (“The North Pole”), particular events (“the assassination of Abraham Lincoln”), sociopolitical organizations (“the U.S. State Department”), and theoretical constructs (“syntax”), and use them in, an explanation, or a story, stripping complex situations down to their essentials, yielding enormous leverage in reasoning about the world.

Highly Structured Cognitive Systems

Marvin Minsky argued that we should view human cognition as a “society of mind,” with dozens or hundreds of distinct “agents” each specialized for

**Much work
in evolutionary
and developmental
psychology points
in the same direction;
the mind is not one
thing, but many.**

different kinds of tasks. For instance, drinking a cup of tea requires the interaction of a GRASPING agent, a BALANCING agent, a THIRST agent, and some number of MOVING agents. Much work in evolutionary and developmental psychology points in the same direction; the mind is not one thing, but many.

Ironically, that is almost the opposite of the current trend in machine learning, which favors end-to-end models that use a single homogeneous mechanism with little internal structure. An example is Nvidia's 2016 model of driving, which forsook classical modules like perception, prediction, and decision-making. Instead, it used a single, relatively uniform neural network that learned direct correlations between inputs (pixels) and one set of outputs (instructions for steering and acceleration).

Fans of this sort of thing point to the virtues of “jointly” training the entire system, rather than having to train modules separately. Why bother constructing separate modules when it is so much easier just to have one big network?

One issue is that such systems are difficult to debug and rarely have the flexibility that is needed. Nvidia's system typically worked well only for a few hours before intervention from human drivers, not thousands of hours (like Waymo's more modular system). And whereas Waymo's system could navigate from point A to point B and deal with lane changes, all Nvidia's could do was to stick to a lane.

When the best AI researchers want to solve complex problems, they often use hybrid systems. Achieving victory in Go required the combination of deep



Digital Threats: Research and Practice

Digital Threats: Research and Practice (DTRAP) is a peer-reviewed journal that targets the prevention, identification, mitigation, and elimination of digital threats. DTRAP aims to bridge the gap between academic research and industry practice. Accordingly, the journal welcomes manuscripts that address extant digital threats, rather than laboratory models of potential threats, and presents reproducible results pertaining to real-world threats.



For further information
and to submit your
manuscript,
visit dtrap.acm.org

Figure 1. Possible number or letter.



learning, reinforcement learning, game tree search, and Monte Carlo search. Watson's victory in *Jeopardy!*, question-answering bots like Siri and Alexa, and Web search engines use “kitchen sink” approaches, integrating many different kinds of processes. Mao et al.¹² have shown how a system that integrates deep learning and symbolic techniques can yield good results for visual question answering and image-text retrieval. Marcus¹⁰ discusses numerous different hybrid systems of this kind.

Multiple Tools for Simple Tasks

Even at a fine-grained scale, cognitive machinery often consists of many mechanisms. Take verbs and their past tense forms. In English and many other languages, some verbs form their past tense regularly, by means of a simple rule (walk-walked, talk-talked, perambulate-perambulated), while others form their past tense irregularly (*sing-sang, ring-rang, bring-brought, go-went*). Based on data from the errors that children make, one of us (Gary Marcus) and Steven Pinker argued for a hybrid model, a tiny bit of structure even at the micro level, in which regular verbs were generalized by rules, whereas irregular verbs were produced through an associative network.

Compositionality

The essence of language is, in Humboldt's phrase, “infinite use of finite means.” With a finite brain and finite amount of linguistic data, we manage to create a grammar that allows us to say and understand an infinite range of sentences, in many cases by constructing larger sentences (like this one) out of smaller components, such

Figure 2. Context-dependent interpretation.



as individual words and phrases. If we can say, *the sailor loved the girl*, we can use that as a constituent in a larger sentence (*Maria imagined that the sailor loved the girl*), which can serve as a constituent in a still larger sentence (*Chris wrote an essay about how Maria imagined that the sailor loved the girl*), and so on, each of which we can readily interpret.

At the opposite pole is the pioneering neural network researcher Geoff Hinton, who has been arguing that the meaning of sentences should be encoded in what he calls “thought vectors.” However, the ideas expressed in sentences and the nuanced relationships between them are just way too complex to capture by simply grouping together sentences that ostensibly seem similar.^{9,10} Systems built on that foundation can produce text that is grammatical, but show little understanding of what unfolds over time in the text they produce.

Top-Down and Bottom-Up Information, Integrated

Consider the image shown in Figure 1:⁶ Is it a letter or a number? It could be either, depending on the context (see Fig-

ure 2). Cognitive psychologists often distinguish between *bottom-up information*, that comes directly from our senses, and *top-down knowledge*, which is our prior knowledge about the world (letters and numbers form distinct categories, words and numbers are composed from elements drawn from those categories, and so forth). An ambiguous symbol such as shown in the figures here looks one way in one context and different in another, as we integrate the light falling on our retina with a coherent picture of the world.

Whatever we see and read, we integrate into a cognitive model of the situation and with our understanding of the world as a whole.

Concepts Embedded in Theories

In a classic experiment, the developmental psychologist Frank Keil⁵ asked children whether a raccoon that underwent cosmetic surgery to look like a skunk, complete with “super smelly” stuff embedded, could become a skunk. The children were convinced the raccoon would remain a raccoon nonetheless, presumably as a consequence of their theory of biology, and the notion that it's what is inside a

creature that really matters. (The children didn't extend the same theory to human-made artifacts, such as a coffee pot that was modified to become a bird feeder.)

Concepts embedded in theories are vital to effective learning. Suppose that a preschooler sees a photograph of an iguana for the first time. Almost immediately, the child will be able to recognize not only other photographs of iguanas, but also iguanas in videos and iguanas in real life, easily distinguishing them from kangaroos. Likewise, the child will be able to infer from general knowledge about animals that iguanas eat and breathe and that they are born small, grow, breed, and die.

No fact is an island. To succeed, a general intelligence will need to embed the facts that it acquires into richer overarching theories that help organize those facts.¹³

Causal Relations

As Judea Pearl¹⁴ has emphasized, a rich understanding of causality is a ubiquitous and indispensable aspect of human cognition. If the world was simple, and we had full knowledge of everything, perhaps the only causality we would need would be physics. We could determine what affects what by running simulations; if I apply a force of so many micronewtons, what will happen next?

But that sort of detailed simulation is unrealistic; there are too many particles to track, and too little time, and our information is too imprecise.

Instead, we often use approximations; we know things are causally related, even if we don't know exactly why. We take aspirin, because we know it makes us feel better; we don't need to understand the biochemistry. We know that having sex can lead to babies and can act on that knowledge, even if we don't understand the exact mechanics of embryogenesis. Causal knowledge is everywhere, and it underlies much of what we do.

Tracking Individuals

As you go through daily life, you keep track of all kinds of individual objects, their properties and their histories. Your spouse used to work as a journalist. Your car has a dent on the trunk, and you replaced the transmission last year. Our experience is made up of enti-

ties that persist and change over time, and a lot of what we know is organized around those things, and their individual histories and idiosyncrasies.

Strangely, that is not a point of view that comes at all naturally to deep learning systems. For the most part, current deep learning systems focus on learning general, category-level associations, rather than facts about specific individuals. Without a notion something like a database record and an expressive representation of time and change, it is difficult to keep track of individual entities distinct from their categories.

Innate Knowledge

How much of the structure of the mind is built in, and how much of it is learned? The usual "nature versus nurture" contrast is a false dichotomy. The evidence from biology—from developmental psychology and developmental neuroscience—is overwhelming: nature and nurture work together.

Learning from an absolutely blank slate, as most machine-learning researchers aim to do, makes the game much more difficult than it should be. It is nurture without nature, when the most effective solution is obviously to combine the two. Humans are likely born understanding that the world consists of enduring objects that travel on connected paths in space and time, with a sense of geometry and quantity, and the basis of an intuitive psychology.

AI systems similarly should not try to learn everything from correlations between pixels and actions, but rather start with a core understanding of the world as a basis for developing richer models.⁷

Conclusion

The discoveries of the cognitive sciences can tell us a great deal in our quest to build artificial intelligence with the flexibility and generality of the human mind. Machines need not replicate the human mind, but a thorough understanding of the human mind may lead to major advances in AI.

In our view, the path forward should start with focused research on how to implement the core frameworks¹⁵ of human knowledge: time, space, causality, and basic knowledge of physical objects and humans and their interactions. These should be embedded into an ar-

chitecture that can be freely extended to every kind of knowledge, keeping always in mind the central tenets of abstraction, compositionality, and tracking of individuals.¹⁰ We also need to develop powerful reasoning techniques that can deal with knowledge that is complex, uncertain, and incomplete and that can freely work both top-down and bottom-up,¹⁶ and to connect these to perception, manipulation, and language, in order to build rich cognitive models of the world. The keystone will be to construct a kind of human-inspired learning system that leverages all the knowledge and cognitive abilities that the AI has; that incorporates what it learns into its prior knowledge; and that, like a child, voraciously learns from every possible source of information: interacting with the world, interacting with people, reading, watching videos, even being explicitly taught.

It's a tall order, but it's what has to be done. □

References

1. Brown, T.B. et al. Language models are few-shot learners. (2020); arXiv preprint arXiv:2005.14165
2. Darwische, A. Human-level intelligence or animal-like abilities? *Commun. ACM* 61, 10 (Oct. 2018), 56–67.
3. Devlin, J. et al. BERT: Pre-training of deep bidirectional transformers for language understanding. *NAAACL-2019*. (2019), 4171–4186.
4. Firestone, C. and Scholl, B.J. Cognition does not affect perception: Evaluating the evidence for 'top-down' effects. *Behavioral and Brain Sciences* 39, e229. (2016.)
5. Keil, F.C. *Concepts, Kinds, and Cognitive Development*. MIT Press, Cambridge, MA, 1992.
6. Lupyan, G. and Clark, A. Words and the world: Predictive coding and the language=perception-cognition interface. *Current Directions in Psychological Science* 24, 4 (2015), 279–284.
7. Marcus, G. Innateness, alphazero, and artificial intelligence. (2018); arXiv preprint arXiv:1801.05667.
8. Marcus, G. Deep Understanding: The Next Challenge for AI. *NeurIPS-2019* (2019).
9. Marcus, G. GPT-2 and the nature of intelligence. *The Gradient*. (Jan. 25, 2020).
10. Marcus, G. The next decade in AI: four steps towards robust artificial intelligence. (2020); arXiv preprint arXiv:2002.06177
11. Marcus, G. and Davis, E. GPT-3, Bloviator: OpenAI's language generator has no idea what it's talking about. *Technology Review* (Aug. 22, 2020).
12. Mao, J. et al. The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision. arXiv preprint arXiv:1904.12584.
13. Murphy, G. *The Big Book of Concepts*. MIT Press, 2002.
14. Pearl, J. and MacKenzie, D. *The Book of Why: The New Science of Cause and Effect*. Basic Books, New York, 2018.
15. Spelke, E. Initial knowledge: Six suggestions. *Cognition* 50, 1–3 (1994), 431–445.
16. van Harmelen, F., Lifschitz, V., and Porter, B., Eds. *The Handbook of Knowledge Representation*. Elsevier, Amsterdam, 2008.

Gary Marcus (gary.marcus@nyu.edu) is Founder and CEO of Robust.AI, and Professor Emeritus at NYU.

Ernest Davis (davise@cs.nyu.edu) is Professor of Computer Science at NYU. This Viewpoint is adapted from their new book, *Rebooting AI: Building Artificial Intelligence We Can Trust*.

Copyright held by authors.

Viewpoint

Excessive Use of Technology: Can Tech Providers be the Culprits?

Seeking to assess the possible responsibility of tech providers for excessive use patterns.

THE INFLUX OF hedonic online services (including video streaming, social media, video games) has created rather fierce competition for people’s attention, in what is termed the “attention economy—in which every minute of attention and engagement tech companies can “squeeze” out of users counts. To compete in this environment, tech companies, intentionally or unintentionally, have adapted practices that have capitalized on varying features of human decision making and brain physiology to cultivate automatic, and uninterrupted use.⁴

There is a body of evidence—growing yet debated—suggesting that when some technologies are used excessively, the use can interfere with normal functioning, such as with sleep, physical activity, and school performance.¹² What’s more, populations such as children and adolescents may be susceptible to excessive use,² although age related prevalence issues have not always been made clear. We say the evidence is debated because some studies suggest that excessive use may be related to prior mental illness rather than to the technology itself.⁶ Consequently, some scholarly groups have criticized the concept of “technology addiction.”¹ Therefore, we use here the term “excessive use,” which reflects use patterns that are excessive in



that they infringe on normal functioning of users.⁵

The role of tech companies (mostly hedonic online service providers and app developers) in excessive use is an issue that merits further discussion and research. This issue is very timely, given the tendency to blame tech providers for many ills in our society (for example, violence and radicalization on social media and/or the role of arti-

ficial intelligence (AI) in job displacement and reduced human agency). Focusing on excessive use, as is, it is often assumed that it is the sole responsibility of users; they should have controlled their use. This is akin to a speeding driver, in which case if caught, most people will agree that it is purely his or her fault, and not the car manufacturer’s fault for affording speeding. This simplistic one-sided

view, however, has been losing ground in recent years. For example, the use of loot boxes in video games has been equated with gambling, which prompted debate about the need to regulate such tools.³ Similarly, a recent U.S. senate bill proposes social media providers should also take some responsibility for excessive use, and remove psychological mechanisms that reduce people's self-control over their use.¹⁰

In this Viewpoint, we seek to make first strides toward discussing the responsibility of tech providers for excessive use. Initiating this discussion is important, because it can serve as a basis for more informed use practices and interventions.

What Makes Technology Use Excessive?

Excessive use of technologies is not measured by use frequency, or time, because what is excessive for one person or in one situation may be normal, unharmed, and even beneficial for another person or in another situation. For example, spending five hours/day on social media may benefit an unemployed job seeker, but may become excessive when this person starts working. As such, the excessiveness of technology use is typically captured by a range of persistent negative symptoms involving interference with other life responsibilities. Given there are no agreed upon criteria, prevalence rates of excessive use range from 1% to over 17%.¹¹ The high numbers may result from false positives (that is, identifying individuals as excessive users when they are experiencing mundane symptoms).

Motivation for Excess

If excessive use of technology is characterized by persistently hurting other life domains, why would rational people engage in such excessive behavior? In part this may relate to humans' limited ability to control very tempting behaviors,⁸ particularly under times of strain. This explanation is based on dual-system theory, according to which some people have a hyperactive reward processing system that creates strong motivations to engage in tempting behaviors, and in some cases also have hypo-active self-

Hedonic technologies are unique in that they can be consumed nearly anytime and anywhere with relative assumed privacy.

control faculties that prevent them from engaging the "brakes." The use of many personal-hedonic technologies routinely activates the reward faculties in the brain, which makes these technologies susceptible to excess consumption.

While this is also true for many other routine fun activities such as eating and shopping, hedonic technologies are unique in that they can be consumed nearly anytime and anywhere with relative assumed privacy. This has been afforded by the advent of smartphones and ubiquitous high-speed data access (at least in the U.S.). That is, while rewarding behaviors such as eating may be equally or more rewarding than technology use, they typically cannot be performed as routinely. In addition, many hedonic technologies afford socialization with large groups, beyond the physical reach of users. This can be a highly rewarding facet, and it typically cannot be afforded to the same extent by other rewarding activities. Whether these are meaningful or trivial differences remains to be seen from future research.

Both nature and nurture affect difficulties in moderation of fun activities. Regarding arguments for the nurture component, many scholars argue it is driven by the way modern technologies are designed. Tech companies fight for their survival by trying to accumulate use time and engagement, which often translate into increased in-app purchases or advertising reve-

nues.⁴ Some worry they specifically use mechanisms that promote repeated, automatic, tempting behavior through a variable reward schedule⁷ and making behaviors easy and automatic.⁹ Rewarding behaviors produce behavior-reward associations in people's brains, which leads to behavior seeking and reenactment, especially when rewards are obtained on a variable schedule.⁴ Tech companies have mastered the delivery of variable rewards. For example the schedule of "likes" on social media posts is variable; and the wins or content of loot boxes on video games is also variable.³

That said, much of this narrative is speculative. Almost certainly, tech companies attempt to develop ways in which participants remain engaged, although the degree to which such mechanisms are harmful remain hotly contested. The proliferation of modern technology has not been linked to a visible epidemic or upswing of "addicted" individuals in the same manner that irresponsible prescribing of opioids led to an opioid epidemic in the U.S. This need not absolve technology companies from a role in protecting their struggling customers or preventing vulnerable customers from becoming excessive users. However, we argue that narratives that are overly hostile to tech companies, imply they are a primary source of overuse problems, or have sinister intentions, are likely less than helpful. In part this may be because technology overuse may sometimes be symptomatic of other issues.⁶

Are Tech Companies Practices Ethical?

It is not uncommon to hear activists claim that scientists are hired by technology companies to make technology purposefully addictive. Engaging AI to choose and present content (for example, on the social media feed) that will overly engage the users can also be blamed for causing excessive use. However, evidence for such claims is still lacking. Such concerns also appear to confuse addiction (a pathological state) with engagement (a state of continued, enjoyed use, with no significant impairment). However, this need not mean that some mechanisms might not over-

shoot engagement into excessive overuse. One useful test for ethics in this context is whether tech companies act like drug dealers, in that they manipulate people to use their products, their products are harmful, and they themselves do not use their products.⁵ While there is a trend in Silicon Valley for some tech executives to send their kids to tech-free schools,¹² it does not seem that tech executives avoid using their own products. The evidence regarding the harmfulness of technology is also not conclusive, and does not apply to all users. Hence, on its face, it seems that tech companies pass at least some aspects of this ethicality test; yet their personnel present some worries about the potentially harmful nature of technology, at least for young children.

There also appears to be little consensus regarding the ethical ramifications of scientists' involvement with technology companies and/or the use of AI for increasing engagement. Certainly, were scientists to *knowingly* engage in actions they believed might be harmful to consumers; common ethics principles are violated. However, there does not appear to be current evidence to support such claims. On its face, it does not seem to differ much from engaging food scientists for developing tastier foods. One can ask in this case, if the scientists adding sugar to food while ignoring the implications (such as obesity, tooth decay) were ethical. This is of course not an easily resolved issue, but it should be discussed for ensuring we avoid moral panic, while ensuring users who need our help and protection receive it.

Recommendations

One thing that is clear is there is a need for further research to clarify concepts related to excessive use of technology. First, distinguishing whether excessive use behaviors constitute a unique diagnosis or are better conceptualized as risk markers, symptoms or red flags of established mental health disorders would be welcome. Second, current conceptualizations of excessive use tend to rely on symptom profiles adapted from substance abuse. However, critiques of this method suggest it may be too

Almost certainly, tech companies attempt to develop ways in which participants remain engaged, although the degree to which such mechanisms are harmful remain hotly contested.

easy to meet “addiction” criteria as applied to technology use (for example, most people will feel some discomfort/withdrawal when prevented from using their smartphones, but this “withdrawal” in non-comparable with the physical withdrawal people who quit substances feel). Research on symptom sensitivity and specificity is therefore needed. Third, it would be important to consider whether excessive use is distinct from overuse of non-tech behaviors such as shopping. If not, it may be of greater utility to consider an overarching behavioral overuse disorder category that could be applied to any behavior, rather than many microdiagnoses focused on specific behaviors.

Without this greater research clarity, it is unclear what ethical advice to give to scientists working with technology companies. We note that *knowingly* developing technology (for example, algorithms, AI) that would reasonably be expected to lead to excessive use among vulnerable individuals would certainly be unethical. However, we feel that blanket prohibitions against scientists working with technology companies, including related to non-pathological engagement, are not yet warranted. What is needed, as a first step, is much greater transparency and scrutiny of funding

arrangements and potential conflicts of interest by computer and social scientists working with tech providers. Take for example the Cambridge Analytica scandal, which was a non-scrutinized collaboration between academics and industry. Hopefully with further research, we will have greater clarity on these ethical issues, and better insights on best academia-industry collaboration practices. In the meantime, technology companies can help with this by making their considerable anonymized user data available openly to scholars without restrictions regarding the favorability of scholarly findings for those technology companies. They should also meet our concerns with open ears and minds. Academics, for now, can simply employ an ethical mind-set when getting involved in projects that may support excessive use. 

References

1. American Psychological Association Society for Media Psychology and Technology and Psychological Society of Ireland Special Interest Group in Media, t.A.a.C. An Official Division 46 Statement on the WHO Proposal to Include Gaming Related Disorders in ICD-11, The Society for Media Psychology and Technology, Division 46 of the American Psychological Association, 2018.
2. Cerniglia, L. et al. Internet addiction in adolescence: Neurobiological, psychosocial and clinical issues. *Neuroscience & Biobehavioral Reviews* 76 (2017), 174–184.
3. Drummond, A. and Sauer, J.D. Video game loot boxes are psychologically akin to gambling. *Nature Human Behaviour* 2, 8 (2018), 530.
4. Eyal, N. and Hoover, R. *Hooked: How to Build Habit Forming Products*. Portfolio Hardcover, New York, NY, 2014.
5. He, Q., Turel, O. and Bechara, A. Association of excessive social media use with abnormal white matter integrity of the corpus callosum. *Psychiatry Research: Neuroimaging* 278 (2018), 42–47.
6. Jeong, E.J., Ferguson, C.J., and Lee, S.J. Pathological gaming in young adolescents: A longitudinal study focused on academic stress and self-control in South Korea. *Journal of Youth and Adolescence* (2019).
7. Karlsen, F. Entrapment and near miss: A comparative analysis of psycho-structural elements in gambling games and massively multiplayer online role-playing games. *International Journal of Mental Health and Addiction* 9, 2 (2011), 193–207.
8. Osatuyi, B. and Turel, O. Tug of war between social self-regulation and habit: Explaining the experience of momentary social media addiction symptoms. *Computers in Human Behavior* 85 (2018), 95–105.
9. Social Media Addiction Reduction Technology Act *LYN19429*, 2019, 1–14.
10. Tarafdar, M. et al. The dark side of information technology. *MIT Sloan Management Review* 56, 2 (2015), 600–623.
11. Turel, O. Potential 'dark sides' of leisure technology use in youth. *Commun. ACM* 62, 3 (Mar. 2019), 24–27.
12. Weller, C. Silicon Valley parents are raising their kids tech-free—and it should be a red flag. *Business Insider*, 2018.

Ofir Turel (oturel@fullerton.edu) is a Professor of Information Systems at California State University, Fullerton, CA, USA.

Christopher Ferguson (cjfergus@stetson.edu) is a Professor of Psychology at Stetson University in Deland, FL.

Copyright held by authors.

volume
01

number
01

FIRST
ISSUE
PUBLISHED

ACM/IMS Transactions
on Data Science
is now available in
the ACM Digital Library



ACM/IMS Transactions on Data Science (TDS) publishes cross-disciplinary innovative research ideas, algorithms, systems, theory and applications for data science. Papers that address challenges at every stage, from acquisition on, through data cleaning, transformation, representation, integration, indexing, modeling, analysis, visualization, and interpretation while retaining privacy, fairness, provenance, transparency, and provision of social benefit, within the context of big data, fall within the scope of the journal.

Q Article development led by [acmqueue](https://queue.acm.org)
queue.acm.org

Keeping users secure through their smartphones.

BY PHIL VACHON

The Identity in Everyone's Pocket

MOST EVERY TECHNOLOGY practitioner has a smartphone of some sort. Around the world cellular connectivity is more ubiquitous than clean, running water. With their smartphones, owners can do their banking, interact with their local government, shop for day-to-day essentials, or simply keep in touch with their loved ones around the globe.

It's this ubiquity that introduces interesting security challenges and opportunities. Not even 10 years ago, a concept like biometric authentication was a novelty, reserved only for specialized applications in government and the financial services industry. Today you would be hard-pressed to find users who have not had the experience of unlocking their phones with a fingerprint, or more recently by simply looking at the display. But there is more to the picture than meets the (camera's) eye: Deep beneath layers of glitzy user interfaces, there is a world of secure processors, hardware-backed key

storage, and user-identity management that drives this deceptively simple capability.

Newer phones use these security features in many different ways and combinations. As with any security technology, however, using a feature incorrectly can create a false sense of security. As such, many app developers and service providers today do not use any of the secure identity-management facilities that modern phones offer. For those of you who fall into this camp, this article is meant to leave you with ideas about how to bring a hardware-backed and biometrics-based concept of user identity into your ecosystem.

The goal is simple: Make it as hard as possible for attackers to steal credentials and use them at their leisure. Let's even make it difficult for users to clone their own credentials to share with other users. In addition to this protection, let's ensure that adding extra factors such as biometric authentication provides a stronger assurance of who the user is. Bringing keys and other secrets closer and closer to something that is physically attached to the user provides a stronger assurance of the identity of the user who just authenticated to the device.

What Is a Digital Identity?

In the physical world, proving your identity might involve checking an identity document such as a passport, visa, or driver's license, and matching a photo or other *biometric* printed on that document. The value of forging these documents is quite high, so nation-states and other identity issuers go to great lengths to make this difficult. At the same time they must make it easy for a verifier to catch even the most sophisticated forgeries. There is a whole industry behind designing secure documents (for example, <https://www.jura.hu>), developing anti-forgery technologies (<https://www.muehlbauer.de>), and producing these documents at scale. Of course, these efforts are not foolproof, and sometimes the most sensitive use cases warrant a



IMAGE BY ANDREJ BORNS ASSOCIATES, USING SHUTTERSTOCK

closer inspection of the identity document, using “secret” security features embedded in the document itself.

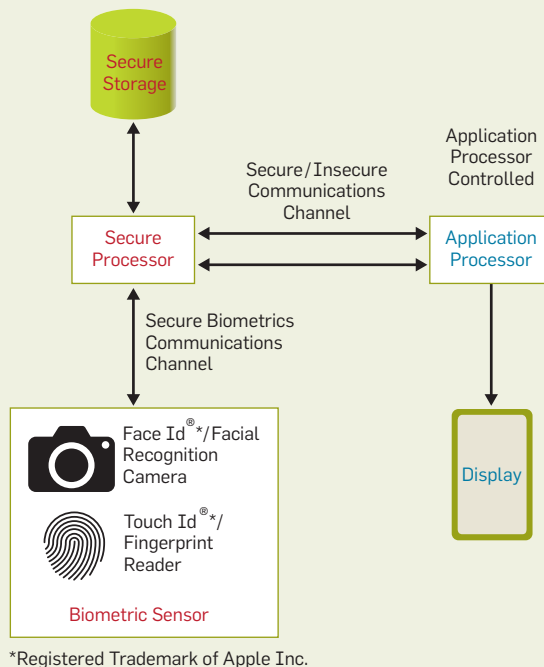
In the realm of technology, an identity is proven through some sort of cryptographic scheme, the identity itself being embodied in a secret key held by the user. Simply possessing this secret, however, is often not enough: Like a physical identity document that doesn’t have a photo to identify the possessor,

some cryptographic secrets can be stolen and used by anybody. For most use cases, the *policy* around how a secret is stored becomes critical. A private key stored on a laptop’s hard disk might not be as trustworthy as a private key stored in a smart card.

Consider, for example, the classic evil maid attack in which the attacker uses privileged access to a physical space (such as a private residence) to alter,

steal, or simply use a device or credentials in a way that the owner would be unable to detect. Where you store a private key can make a difference. While an evil maid might be able simply to copy a private key stored on a laptop’s disk, he or she could not easily do so on a smart card, which is hard to clone and extract material from. The evil maid would not be able to walk away with a smart card without you noticing, and cloning the

Figure 1. The idealized smartphone.



card in short order is a difficult task with most modern implementations. It's also easier to keep a smart card on your person at all times, thus keeping it out of the clutches of the evil maid.

Mobile Phones as Secure Keystores

Many years of development, hard-learned security engineering lessons, and practical experiences have led to extensive security capabilities in most modern smartphones. Before discussing the use of a mobile phone as an identity management device, let's define what this device looks like at a high level.

Figure 1 shows the idealized smartphone. Note the division between the AP (application processor) and SP (secure processor), and how they control different aspects of the phone.

The parts of a smartphone are fairly simple:

- ▶ A display.
- ▶ A biometric sensor (facial recognition, fingerprint recognition).
- ▶ A “secure” processing environment, or SP. (GlobalPlatform prefers the terminology *trusted execution environment*, or *TEE*. Architecturally, these are similar in concept, but the use of *SP* avoids confusion with terminology that is often used to refer to the Android-specific implementation). The SP is where specialized

security software runs, such as Apple's SEPOS (Secure Enclave Processor operating system) or Qualcomm's QTEE; all memory-containing program code and data associated with the SP environment is protected such that not even other CPUs on the same chip can access the SP data (more on this later).

- ▶ A secure storage environment, where secrets and other sensitive information for the SP are stored.

- ▶ The “application” processing environment, or AP, where apps and the phone's operating system (such as iOS or Android) run. The AP can communicate with the SP only through a limited channel.

One assumption made here is that the device is normally in the user's possession. Also, additional protections such as a PIN are assumed to be strong enough to protect the device from an adversary. While it's interesting to think about the attacks a nation-state with unlimited resources could pull off, designing to such a high standard is not always practical.

With this smartphone model you can start reasoning about how to construct a security system. Think of the SP and AP as two separate worlds on one phone. For the iPhone, Apple introduced the SEP. Most Android

phones either have a completely separate chip (such as, Google's Titan M chip in the Pixel 3 and later) or implement the SP as a TEE using TrustZone,⁹ an ARM-proprietary secure virtualized state of the application processor CPU.

The SP has a dedicated secure region of memory that is encrypted and usually authenticated.^{3,6} This encryption also protects the secure memory from attackers in physical possession of the phone, as well as preventing the AP from altering or recovering the SP's state. Without access to the keys used to encrypt memory, anyone would have a difficult time recovering the raw memory contents. This is a hardware-enforced control on the modern SoC (system on a chip) such as those from Apple or Qualcomm. Any breach of this control would be catastrophic, allowing the AP free access to any of the SP's sensitive data in memory. (A vulnerability in the SP's software would allow an attacker to gain access to the SP's memory in a way similar to what the secure hardware is trying to prevent. If this poses too large a risk for your application, you might want to think about other hardware secure tokens, such as YubiKeys, or even building your own.)

The SP also has access to its own set of peripherals (such as the fingerprint sensor or secure external devices for payment processing or secure data storage) that are inaccessible to the AP. Certain features of the SoC, such as cryptographic keys that give a smartphone its unique identity, are also accessible only to hardware available exclusively to the SP. The keys to encrypt all the long-term storage for the secure processor are usually stored using this type of mechanism.

The persistent storage for the SP includes a number of important pieces of data, including secret keys generated by applications, biometric templates representing authorized users, and keys that uniquely identify the phone. In most implementations, these bits of data are cryptographically *wrapped* using the long-term storage keys, making them accessible only to the software running in the SP. The persistent data is then handed back to the AP for long-term storage in flash. This wrapping process keeps this information safe and ensures that no applications running

on the AP would be able to pretend they are the SP. More importantly, wrapping prevents malicious parties from extracting secrets from the phone and cloning them (or worse, subtly corrupting these secrets).

The SP, compared with the AP, runs an extremely simple, minimal operating system. Typically third-party apps can't be installed in this environment, and the code that does run is purpose-built—just for the security applications required for the device. This is designed to minimize the exposed attack surface and reduce the probability of software vulnerabilities compromising the integrity of the SP. As you know, software, even in the SP, is never actually perfect.¹⁰

All communication between the SP and AP is highly regimented. By design, the AP cannot access the memory of the SP, but the SP might be able to see some of the AP's memory. All communication between the two worlds is through RPCs (remote procedure calls), serializing all arguments and data to be passed from the insecure world to the secure world (or vice versa).

Operations defined with this RPC mechanism are usually quite high level. For example: "Generate key pair" generates a new key pair with parameters specified in the command; it then returns the public key and the key ID as a response, "Sign blob with key pair," which takes the key ID and a pointer to a blob of data, then returns the signature as a response. These operations are inflexible, but that is by design: Flexibility introduces more ways things can go wrong.

Figure 2 shows the logical division between the SP and AP. Note how the SP has its own private encryption hardware and how that hardware is the only way to access key material generated during manufacturing. This protects the key even from software compromise. Most SPs do not have enough flash memory for storing all the keys needed but instead pass their data, encrypted using a key only accessible to the SP, to the AP for long-term storage. Secure memory protection prevents the AP from being able to "see" what is going on in the SP's memory space, but lets the SP read and write the AP's memory space.

Figure 3 shows the usage sequence for any key generated and held by the

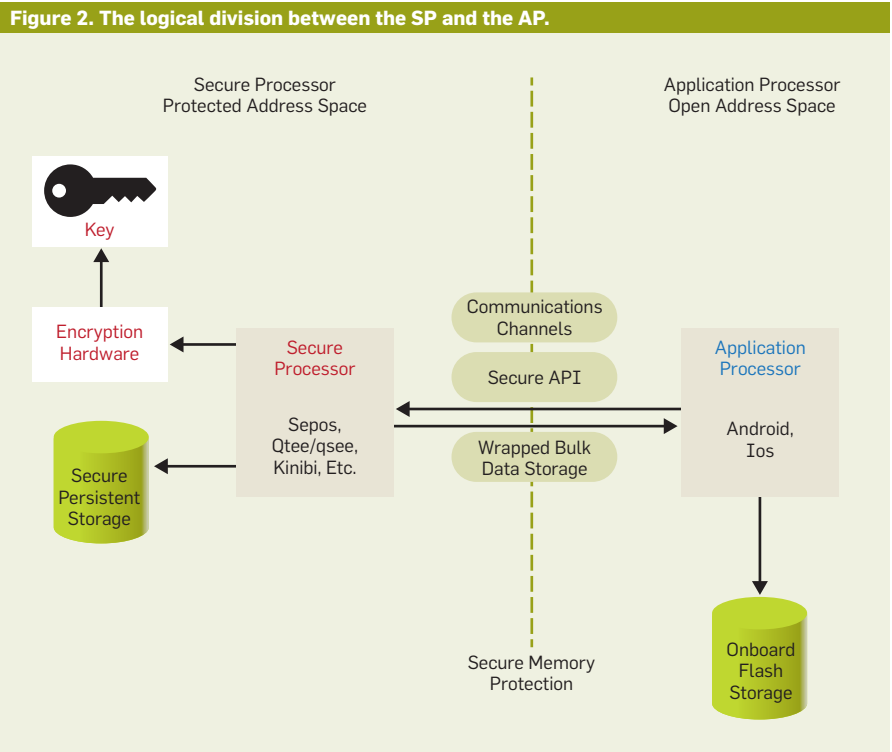
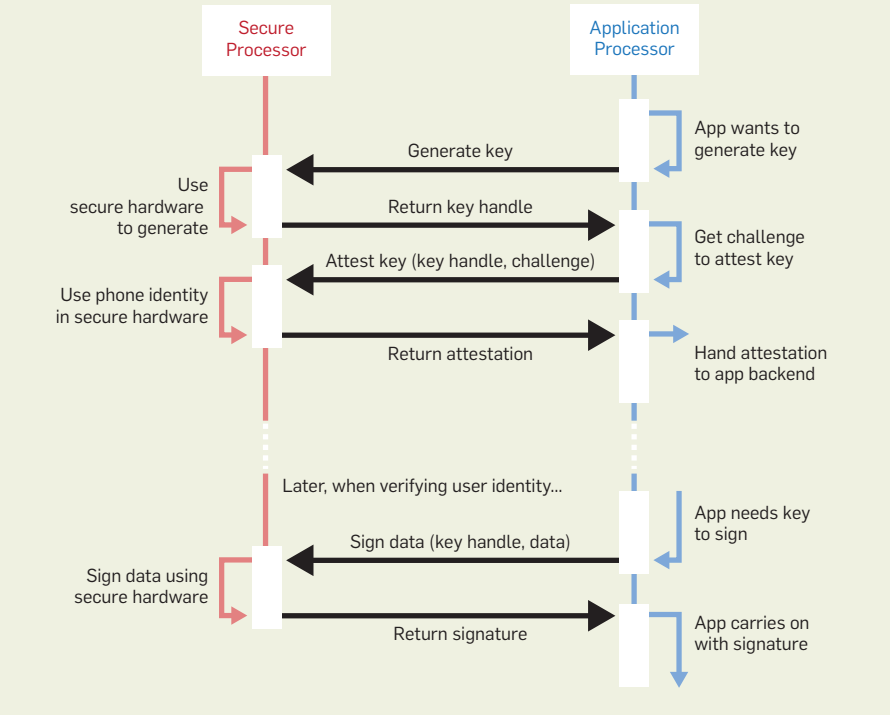


Figure 2. The logical division between the SP and the AP.



SP. Note how the AP can request use of the key only through specific RPCs and never accesses the private key itself.

One important lesson to take away is that there is a carefully choreographed dance occurring between hardware and software running on both the AP and SP to implement the

security features of a modern phone. One mistake, and the entire security model could be compromised.

The final piece of the puzzle to consider is where a phone's identity comes from. Establishing trust requires a bit of proof that the phone was manufactured to the expected security standard.

Traceability back to a secure manufacturing process requires that the device have a cryptographic secret programmed into it during manufacturing, which can be tracked to the manufacturer. A proof of identity signed with this manufacturing secret, combined with knowledge of the physical security of a device holding the key and the software policies around how and where a key you generated might be used, allows you to decide how trustworthy an identity you generated on a device really is.

Basic Identity Model

By now it should be clear that most recent phones have all the pieces required to create a digital identity. How can developers use those pieces to build up an identity that allows them to authenticate a user running an app on the phone to access a service that runs in the cloud or some on-premises infrastructure?

There is a common lifecycle for any identity you generate to support authenticating a user to your application. The basic steps, whether for a smartphone, stand-alone biometric token, or otherwise, are:

1. *Enrollment.* This kicks off the process to generate required keys.
2. *Attestation and delivery.* This verifies that keys are secure, safely stored, and difficult to extract and clone. If this

succeeds, you can deliver some form of identity to the device for future use.

3. *Usage.* The keys are used, perhaps for mutual TLS (Transport Layer Security) authentication or some other out-of-band authentication protocol.

4. *Invalidation.* When something about the user or the phone changes in some way, the user identity keys should be erased, forcing the user to re-enroll.

As we look at the various techniques involved in making such an identity model work, we will expand on what each of these steps means for your application.

Key Pairs

In practical applications, cryptographic identities are represented using asymmetric key pairs. While the details of asymmetric cryptography are well beyond the scope of this article, the author recommends *An Introduction to Mathematical Cryptography* by Jeffrey Hoffstein, Jill Pipher, and J.H. Silverman (Springer 2008) for a deep dive into how cryptographic schemes work; *Practical Cryptography* by Niels Ferguson and Bruce Schneier (Wiley, 2003) is also a great, albeit slightly dated, reference.

Without going into great detail, asymmetric keys can be said to be composed of two parts: a private key

that must be kept secret and can be used to generate cryptographic proofs (in the form of digital signatures), and a public key, which can be used by another party to verify these signatures. The private key must be used under the most controlled circumstances—using the most protected of hardware, ensuring nobody could capture the private key. In the smartphone model of Figure 1, this means all operations performed with the private key are done in the SP's environment.

Keys And Attestation

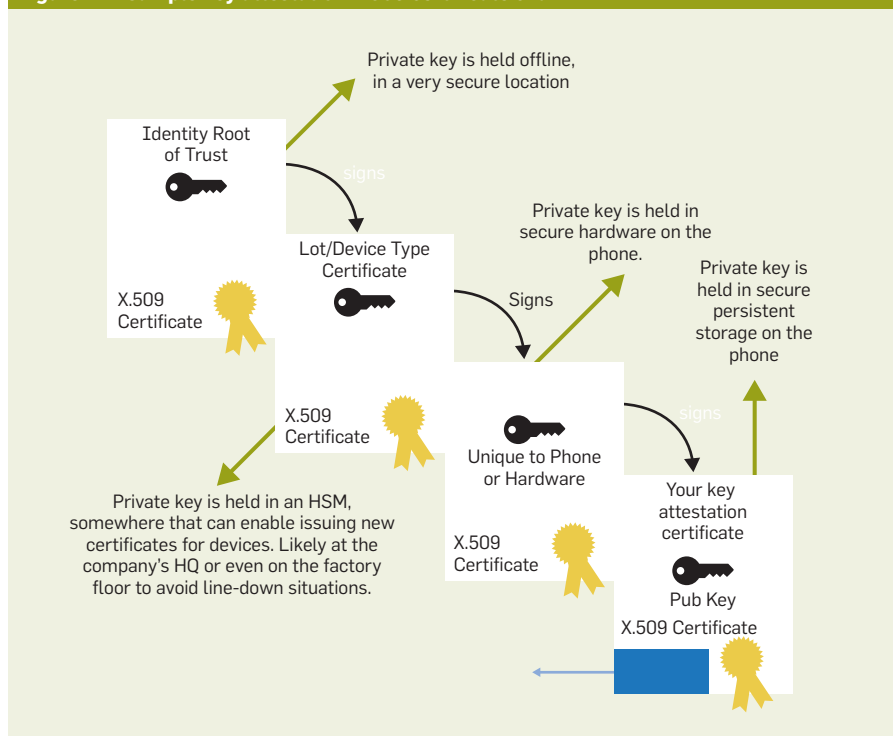
It is very difficult to consider the trustworthiness of a private key and the policy with which it is stored without having a way to verify these attributes. How do you gain confidence that the keys you are generating have been stored in an actual SP?

The process of key attestation uses another private key, usually the one installed during manufacturing, to form a proof. This proof is in the form of structured data that contains the public key you generated and the attributes of the key, including whether or not a biometric factor is required to access this key and other policies around its usage. When this data along with a proof of validity for the key unique to the hardware, plus the detailed security policy of the device, are considered together, you have what is needed to decide whether or not to trust that a key is stored in secure hardware with the specified policy. Let's review the mechanisms for expressing this attestation.

Identity secrets and certificates. Any identity verifier will need a public key to perform some sort of verification. That key can be shared freely, unlike the private key that is held as a secret in the device's SP. Some platforms, such as Apple's SEP, allow the user to generate key pairs on the NIST P-256 elliptic curve. Conversely, Android's hardware-backed keystore implements the RSA (Rivest-Shamir-Adleman) scheme.²

A public key on its own is ambiguous and hard to know anything about; all a public key consists of is a large integer (or pair of integers, in the general case of elliptic curve cryptography) and no other unique metadata. To show the purpose of the key and to

Figure 4. A sample key attestation X.509 certificate chain.



show any policy checks that were done as part of creating the key, there must be a container to carry metadata about the key, as well as a signature wrapping this container to allow a recipient to verify the authenticity of this information.

The most common form of this verification is an X.509 certificate, which is an ASN.1 (Abstract Syntax Notation One) DER (Distinguished Encoding Rules)-encoded object that contains many fields, including:

- ▶ When a certificate becomes valid to use.
- ▶ When the certificate is no longer valid.
- ▶ Who verified the authenticity of the party holding the private key of the certificate.
- ▶ The public key itself.
- ▶ A signature from the trusted authority who validated the attributes of the key and created the certificate, showing the authenticity of the certificate's contents.

Usually an X.509 certificate is grouped with a set of authority certificates that represent who authenticated the contents of the certificate. This complete set of authority certificates, along with the end entity being authenticated, forms a *certificate chain*.

The key attestation process relies on the SP to perform a series of steps that result in an X.509 certificate chain that shows the provenance of the device back to some authority. An iPhone is tied to an authority run by Apple, while for Android this authority is Google. Figure 4 shows a sample key attestation X.509 certificate chain.

Of course, X.509 certificates are ubiquitous. Key attestation is only one limited use. An X.509 certificate chain can be used to identify a service or a user uniquely, as is frequently done for web applications. An X.509 certificate issued for a user would be joined with a private key stored in hardware. After verifying that a key is held in secure hardware and that the policy for that key matches expectations, you would then issue your own X.509 certificate to the user for the key you just attested. This means that you do not need to verify a key attestation every time you want to authenticate the user; it also lets other parties that trust you as the identity provider verify your user's identity.

Biometric factors. Most smartphones include a biometric factor: the bare minimum these days is a fingerprint sensor. Facial recognition is increasingly present in high-end devices, but in the era of COVID-19, where most people are wearing masks, the convenience of facial recognition has been reduced. In fact, to maintain security guarantees Apple made passcode unlocks easier to perform.⁴

A phone must be personalized for a biometric factor to be usable by an application. This means a user must enroll at least one biometric factor through the system's mechanism. An app developer has to make the critical assumption that the user who personalizes the phone is the owner of the phone or an authorized user.

For many use cases, biometric factors exist as a convenience. Rather than typing a long passcode every time a user wants to unlock a device, the user can simply present a biometric factor to prove who he or she is. Most devices require that the user provide the passcode at least once every seven days, as well as after a reboot or other system events. Reducing the frequency of entering a long passcode means users are more likely to choose long, complex passcodes, improving overall device security.

Both iOS and Android make it possible to set a flag such that a key can be used only if the user has successfully performed a biometric authentication—providing that extra level of confidence and forcing the proof of possession of the biometric factor. (An incident in Malaysia shows the extremes to which car thieves were willing to go to steal a biometric factor to unlock a Mercedes S-class.⁸ Most fingerprint sensors today have some form of liveness detection to thwart this kind of grisly attack.)

Biometric authentication can ensure someone is not sharing a passcode among multiple users or has not had a passcode shouldered while unlocking the phone. Requiring users to prove who they are before performing a cryptographic operation provides some assurance that they are at least physically present and authorized to use the device. The SP performs the entire biometric authentication process, ensuring that any operations involving

biometric templates occur in a secure environment and that tampering with the templates isn't practical.

Most implementations have a secure channel between the SP and the biometric sensor. This makes stealing a biometric factor and replaying it later difficult to accomplish. Wrapping the measured biometric values in a secure, authenticated channel makes replay attacks, where the communication between the sensor and the SP is captured, impractical. This provides a stronger assurance that the sensor is physically present.

For evidence of why this secure channel is important, you do not have to look far. In 2018, researchers at Technische Universität Berlin demonstrated an attack where they recovered a latent fingerprint image from a physical card, then removed the fingerprint sensor and built a device to simulate the fingerprint sensor transferring that image to the host CPU.⁷ Since there were no security features to authenticate the communication between the fingerprint sensor and the CPU itself, the attackers were able to unlock the card without the original finger being present. This failure shows why it's important for a secure channel to exist between the sensor and the SP, including the ability to authenticate all communications between the two.

Finally, a policy decision is needed: Do you continue to trust an identity you generated if a user has added new biometric enrollments on his or her phone? This could indicate some sort of compromise of the device. It could also indicate the user was having trouble with the biometric factor and tried to re-enroll, or perhaps the user's children added their own enrollment so they could more easily buy in-game currency. This is a security and usability trade-off to consider. Both Android and iOS enable a policy that will delete a key if any biometric factors have been added.

Trust is a business decision. When a manufacturer imbues a smartphone with a cryptographic identity on the manufacturing line, it has made an assertion: This phone was manufactured in a trusted environment, to specified standards. At this point, businesses need to make a decision: Do they believe

this standard is sufficient for their threat model?

They are deciding whether to trust the assessments that Google and Apple have made of their manufacturing processes. Proving the authenticity of a device is one of the major challenges facing developers today, but it's critical for them to complete the enrollment process and decide if they trust the device to hold on to the secret for normal use.

For modern Android phones, Google provides this assertion. Each Android phone that is built to the requirements Google has set out will receive an X.509 certificate chain, generated for a private key that is held by the secure hardware on the device. The end entity of this chain is a certificate specifically for this device. Thus, this certificate chain can be provided for outside parties so they can verify the authenticity and trustworthiness of a particular smartphone.

This process is also an attestation—attesting the authenticity and uniqueness of a particular phone. It is worth noting that Google generates an attestation key that is shared among up to 10,000 devices, making it difficult to track users directly (more detail about this later).

By extension, generating another key and associated certificate, subordinate to the device identity certificate, will produce a key attestation certificate. Signing this key attestation certificate with the secret key held in secure hardware could support the claim that all the data held in the certificate is true and valid, assuming the integrity of the software and hardware in the SP has not been compromised. This becomes an authenticated, tamperproof way of transporting data about the state of the world as the SP sees it to the outside world. Short of stealing the private key for device identity, an attacker would have a hard time forging one of these key attestation certificates.

This is not a panacea. As discussed, software bugs might allow attackers to extract secret keys and use them to create seemingly valid, but forged, certificate chains. Hardware bugs could expose sensitive data from the SP to the AP or an outside attacker. Again, you must make a business decision: Is this software and hardware sufficiently well

designed to trust with sensitive access credentials for your service? Do you trust Google as an authority to assess whether or not a phone is secure enough to store sensitive secrets used to identify your users? Is Google storing the attestation keys securely enough to ensure they cannot be stolen? One other risk to consider: Could the phone manufacturer have tampered with the software that runs in the SP? That would completely undermine the trust model.

Key attestation has another benefit: By proving a key is stored in secure hardware, you can also have some assurance that an attacker isn't simply emulating trusted hardware. As long as no party is able to extract the device attestation key from the phone, this assertion will hold true. Malware with enough sophistication to emulate cryptographic APIs is not unheard of, and an attacker who can steal all keys generated by an app would allow that attacker to subvert any sort of trust model. As the secrets are held in secure hardware, even an altered version of your app wouldn't be able to steal these secrets, further protecting your users.

Unfortunately, while Apple implements these capabilities in its Secure Enclave, the ability to leverage this attestation is not yet broadly exposed to third-party app developers. This means that you need to take a leap of faith in the enrollment process for any sort of identity on an iPhone, since verifying that your keys are actually stored in secure hardware is impossible. The iOS 14 developer release introduces app attestation, but this functionality had not been enabled for developer experimentation as of this writing. The new API also does not expose the means to control whether or not a biometric factor is required to unlock the key, limiting its usefulness for many identity management applications.

Google has exposed these features to apps since Android 7, though generally devices that shipped with Android 8 or later implement all the required capabilities. Since Apple has not revealed how it will ultimately expose key attestation and biometric identity functionality to third-party apps, let's explore how to use Android's features for key attestation and establishing an identity.

Establishing an Identity: Android Style

On Google Android devices, a hardware-managed identity is created using the hardware-backed keystore, often run in the TEE, the Android implementation of the SP. An application can specify a number of characteristics for the key pair to be used:

- ▶ The asymmetric encryption scheme to be used (RSA, EC, among others) and the size of the key.

- ▶ Whether or not the user needs to have a PIN code, biometric factor, or other requirements to be able to generate the key at all.

- ▶ Whether or not the user needs to present a biometric factor or simply have recently unlocked the phone (or have not done anything at all) to use the key.

- ▶ The purpose or usage of the key (whether it's supposed to be used for signing, encryption, and so on).

- ▶ An attestation challenge, a number-used-once (*nonce*) that was generated specifically for this enrollment attempt by your back-end application, to avoid replay attacks. This must *not* be shared between enrollment attempts and must be a cryptographically random string.

After the key is generated, the app can request an attestation for the key. This returns an X.509 certificate chain with the following members:

- ▶ The key attestation certificate, the end-entity certificate of this chain, containing the key just generated, the attestation challenge nonce, and the policy associated with the key. This is signed with the device attestation key. This type of certificate is issued to the device only if it has a hardware key store.

- ▶ An intermediate certificate, representing and attesting the device attestation key. This is shared, along with the private key, with 9,999 other devices.

- ▶ Another intermediate, which is associated with a batch, used to issue the device attestation key certificates.

- ▶ Google Hardware Attestation Root certification, representing the root of Android device identities. This is held by Google, hopefully in a very secure location.

These certificates are then passed to your back-end service for verification and to validate that the policies and metadata match expectations. Secure


hardware is the only place that the device attestation key resides; this is the only way the end-entity certificate, the key attestation certificate, can be generated. If the certificate chain is rooted in the Google Hardware Attestation Root, then the key is stored in hardware that Google believes to be secure.

Recommendations and sample code on how to perform key attestation for Android are available on the Android developers website.¹


What about the other 9,999 phones with the same certificate? Sharing the same key attestation certificate among 10,000 devices certainly seems counterproductive from a security perspective. This means that attestations from any phones in that group are indistinguishable. How can you be assured that an identity is unique? Several mitigating factors lower the risk.

First, the keys used by the SP to wrap secure material are unique to each phone. Therefore, even if you found two phones with the same key attestation certificate, you would not be able to swap keys generated by one device with the other. This meets the requirement that identity keys must be difficult to extract or clone. The key attestation certificate gives you assurance only that the hardware-backed keystore is holding on to your keys—it is not meant to be used as an identity on its own. As discussed earlier, you would want to generate your own X.509 certificate for the key held in secure hardware after verifying the attestation is accurate.

Second, each attestation should be tied to some other identity verification operation during the enrollment process. For example, when a user logs in to your app for the first time, your application would generate a unique challenge. There is a limited horizon for how long this challenge should be valid—likely on the order of tens of seconds. This means that replaying a key attestation from another device with the same key attestation certificate would be difficult; the challenge would limit how long such an attestation could be valid. Of course, once a user has successfully authenticated to your service, that challenge immediately becomes invalid, so even knowing this challenge would make it difficult for an attacker to exploit this property without a user knowing something is up.



Proving the authenticity of a device is one of the major challenges facing developers today, but it's critical for them to complete the enrollment process and decide if they trust the device to hold on to a secret for normal use.



With this level of care during the initial enrollment stage, sharing key attestation certificates among devices should not pose a major threat to your service.

Revisiting the Identity Model

Let's revisit the identity model and fill in some details about how an implementation might work.

First, there is a core assumption that the user has personalized his or her phone by registering a fingerprint, facial recognition, or other biometric factor. Personalization is a prerequisite for any biometric factor to be usable.

Once the user has completed the personalization process, you need to generate an identity unique to your application in the user's secure hardware. Typically this is done the first time a user authenticates to your service, perhaps using an alternative second factor. Let's review this in the context of the lifecycle described previously:

1. *Enrollment.* After authenticating the user some other way (for example, username, password, one-time password, challenge on screen as a QR code), you can generate a unique asymmetric key pair that will be used to authenticate the user in the future. The private key is stored by the SP, and the app developer needs to specify the parameters of the key, what is required for the user to unlock this key, and so forth.

2. *Attestation and delivery.* Verify that the parameters (usage policies, key lengths, etc.) around the secret meet your requirements, performing the final checks on your service back end. This is a comprehensive set of checks that gives your back-end application some assurances of: where the key is held; what the user must do before the key can be used by a program (for example, provide biometric proof); what the SP should do when the phone's parameters, such as fingerprint or facial recognition templates, change (for example, invalidate key); and what type of biometric parameter can be used to unlock the key for use. If the attestation checks match expectations, you can issue an identity certificate chain to the user and store that on the device for future use.

3. *Usage.* An identity can then be used, based on the policy defined

during enrollment and verified during attestation. This could be used along with a client certificate to verify identity when connecting to a back-end service—for mutual authentication of TLS sessions—or used to sign a cryptographic challenge that is provided out of band.

4. *Invalidation.* Some events can invalidate a user's identity—for example, changing the user's PIN, adding biometric templates, or other changes to policy that affect the security of the phone. These changes would mean there is no way to guarantee that whoever originally generated the identity is still in possession of the phone. The user must re-enroll in order to get out of this state. Return to step 1.

How to Use a Digital Identity

Once the hard work of establishing an identity has been taken care of, the next step is to use that identity. Many use cases might simply be able to directly use the secret held in the trusted hardware as a part of authenticating to a service through mTLS (mutual Transport Layer Security) authentication. The benefits of mTLS are significant: Requiring any communication with your back-end services to be authenticated using this secret means that no device without a valid, attested identity key pair will even be able to connect. Of course, this comes with a host of other challenges around certificate issuance and management that are outside of the scope here.

This certificate is an attestation of the validity of the generated identity, supplied by you as the service provider. This is done, of course, after the attestation of the validity of the key for this purpose from the phone manufacturer. Lots of trusted authorities are involved in this process.

In this case, you issue an X.509 certificate to authenticate a user through a PKI (public key infrastructure) you run yourself. The private key for the certificate is held exclusively in the phone's secure hardware and can only be acted on in the secure hardware. This means that you have a measure of assurance that users connecting to your service are who they say they are (so long as the integrity of the SP has not been compromised).

Alternatively, a bespoke protocol is an option. A simple challenge-response

protocol—where the SP is tasked with signing a nonce—works, especially for legacy environments where OTPs (one-time passwords) have been implemented. Of course, the usual caveat around implementing any cryptographic protocols applies: Here be dragons. The challenges and risks inherent in building such protocols are too numerous to cover in this article, but suffice to say that if you are not aware of how wrong it can go, you should not be considering this approach.

Privacy Challenges

With any sort of unique identifier that is tied to hardware, the question of user privacy is inevitable. Some vendors do not want to build devices that make it easier for advertisers, hackers, or nation-state adversaries to track users. Cryptographic identities have the advantage that they are very hard to forge—but this cuts both ways, and privacy advocates are rightfully concerned these identities could be used abusively to identify user behavior patterns.

The vendor is a threat. Remember that this whole security system is predicated on trusting the phone vendor. You are assuming that the vendor, in good faith, is not going to compromise the key storage and usage model such that malicious users can violate your security assumptions. Apple and Google both made different decisions on how to approach this trust model, and both models have trade-offs. The major differences crop up during the key attestation process, which is critical for the enrollment phase.

Google chose to attest on the device. This means Google has no visibility into what you are attesting, or that you are performing a key attestation at all—all the secrets and attestation certificate generation is performed by the TEE on the phone. What it also means, however, is that the attestation key could be abused by malicious apps to track users, and thus this approach has privacy implications. Reusing this key across 10,000 other devices does make it harder to track a single user based on the attestation key alone. Of course, the value of this is limited in that other factors about the device

could be used to further disambiguate who you are.

Conversely, Apple has a centralized attestation authority. The most naive approach would involve Apple attesting each key in the SEP by asking its centralized authority to generate a certificate. The Secure Enclave encrypts a blob of data containing information about the attributes of the key, the public key itself, the app that requested the attestation, and unique identifiers for the phone. This encrypted blob is then handed to Apple's attestation authority service, which looks up the device, manufacturing details, whether or not it's been marked as stolen (through Find My iPhone), and similar device posture checks. If everything lines up, the service will return an X.509 certificate chain for the key in question.

The upside of this is that the attestation intermediate authority is tied to Apple, not to a secret stored in the phone, a huge benefit for user privacy. This means you're rooting your trust in Apple authenticating the phone, and you know the phone is real—but you don't know exactly which phone this is. You do not know if any attestations for different apps are from the same device, a huge privacy benefit. This naive approach could give Apple a lot of fine-grained information (beyond the scope of App Store telemetry) about how you're using your iPhone and what apps you're using. That's a huge responsibility to hold onto that information.

The details of key attestation for iOS remain to be explored, as Apple has just announced a subset of the functionality to be released in iOS 14. This is not helpful for many user identity use cases, since there is no ability to require biometric factor verification before using the app attestation key.⁵

Think of the User!

Any identity management or security feature you add to your app must consider user experience. While factors such as biometric authentication make life easier for users, a poorly planned policy for your app can result in a disappointing user experience. The important part of all this is to make several judgment calls—including whether you need a biometric factor at all for your app to achieve its desired level of proof of user identity.

One common mistake is thinking that proof of identity is required every time a user performs an operation. Requiring biometric verification every time a key is used will have side effects for TLS connections, where users will constantly have to prove identity, multiple times over. This could get awkward as an app is backgrounded and its network activity times out. In these cases, it is better to unlock a key for a longer period of time, such as the duration of the session for which the user is likely going to use the app.

Enrollment is tricky—well-defined user flows are needed, especially once biometric authentication is integrated into the user's login process for your app. Users have to know exactly what they're doing at each stage, especially if there's an out-of-band challenge/response protocol involved in enrolling a user, such as a QR code on a webpage that contains a challenge for the app to prove the user logged in elsewhere. Also, make sure that feature and device state detection is well implemented and fails rapidly, so users don't get into a process and have confusing failures. Some common states you will need to check include:

- ▶ If you are using these features, is biometric authentication enabled, and are there enrollments that would allow the user to authenticate? Is there even a biometric sensor present at all?

- ▶ If you are relying on secure hardware in the phone, have you checked that the hardware is present, enabled, and capable of providing the features you require?

- ▶ Can you connect to the services that will perform the enrollment process?

An additional consideration is checking whether or not the enrollment was successful. A dry run at enrollment to ensure that the user knows how to use the biometric capability is always helpful.

Never scare your users. Failures and error messages should be honest and succinct but friendly. A message along the lines of, "Your device is not secure enough," is neither accurate nor appropriate. A vague message could scare and mislead a user. A message that is too technical will train users to ignore error messages, which could be even worse down the road. Messages that explain specific failures

in a user-focused fashion are critical, so users can either self-help or know who to contact for support.

Remember that requiring a biometric factor for a key for which you're generating a short-lived certificate will require the user to present the biometric factor just to sign the certificate signing request. This can have an impact on the user experience during certificate renewal, since every time users renew a certificate, they will have to present that factor. It might be tempting just to issue a long-lived certificate and wash your hands of the matter—this isn't necessarily the wrong thing to do, but it might not fit with your security model.

Make sure that such trade-offs are considered carefully when integrating identity into your system. A longer-lived identity certificate might make sense for an app that users are expected to interact with daily. A short-lived authorization might be preferred if the app is used infrequently, and an attestation only has to happen during these rare interactions.

Where Does This Leave Us?

There's no easy answer when it comes to creating a usable, durable, and secure user identity. Mobile phones offer a compelling option, especially where the right features and capabilities are available, but these are not consistent across the major smartphone platforms today. When building these systems you will always have to make trade-offs, ranging from user experience challenges to limitations of the platforms themselves.

As you build such a system, you will find that the devil truly is in the details: How correct is your service in validating the attestation certificate chain? How do you adjust your policy as mobile-phone technology changes (think Apple's migration to Face ID from Touch ID)? How do you handle various types of partially malformed attestation certificates? Do you want to trust Apple and Google with the crown jewels—how users access and authenticate to your service? Do you even have a choice if you want to leverage smartphones as identity devices?

Stealing a user's credentials with such a scheme is now difficult. Maliciously using the credentials is even

more difficult if you require biometric authentication. To use the keys representing your user, the user would have to be prompted—how convenient is that?

Unfortunately, until Apple provides such capabilities as a part of iOS and makes these features available to apps in its App Store, we are going to be a long way off from making strong, hardware-backed identity ubiquitous. Google, on the other hand, has provided this capability for several years, allowing apps to take advantage of the attestation capabilities. ■

Related articles on queue.acm.org

Hack for Hire

Ariana Mirian

<https://queue.acm.org/detail.cfm?id=3365458>

A Threat Analysis of RFID Passports

Alan Ramos, et al.

<https://queue.acm.org/detail.cfm?id=1626175>

Rethinking Passwords

William Cheswick

<https://queue.acm.org/detail.cfm?id=2422416>

References

1. Android Developers. Verifying hardware-backed key pairs with key attestation, 2020; <https://developer.android.com/training/articles/security-key-attestation>.
2. Android Open Source Project. Hardware-backed keystore, 2020; <https://source.android.com/security/keystore>.
3. Apple Inc. Apple platform security, 2020; https://manuals.info.apple.com/MANUALS/1000/MA1902/en_US/apple-platform-security-guide.pdf.
4. Apple Inc. About iOS 13 updates: iOS 13.5, 2020; <https://support.apple.com/en-us/HT210393#135>.
5. Apple Inc. Establishing your app's integrity; https://developer.apple.com/documentation/devicecheck/establishing_your_app_s_integrity.
6. Cai, L. Guard your data with the Qualcomm Snapdragon mobile platform. Qualcomm, 2019; <https://www.qualcomm.com/media/documents/files/guard-your-data-with-the-qualcomm-snapdragon-mobile-platform.pdf>.
7. Fietkau, J., Starbug, Seifert, J.-P. Swipe your fingerprints! How biometric authentication simplifies payment, access and identity fraud. In *Proceedings of the 12th Usenix Workshop on Offensive Technologies*, 2018; <https://www.usenix.org/conference/woot18/presentation/fietkau>.
8. Kent, J. Malaysia car thieves steal finger. BBC News, 2005; <http://news.bbc.co.uk/2/hi/asia-pacific/4396831.stm>.
9. Ngabonziza, B., Martin, D., Bailey, A., Cho, H., Martin, S. TrustZone explained: Architectural features and use cases. In *Proceedings of the IEEE 2nd Intern. Conf. on Collaboration and Internet Computing*, 2016, 445-451; <https://ieeexplore.ieee.org/document/7809736/definitions>.
10. Ryan, K. Hardware-backed heist: extracting ECDSA keys from Qualcomm's TrustZone. NCC Group, 2019; <https://www.nccgroup.com/globalassets/our-research/us/whitepapers/2019/hardwarebackedhesit.pdf>.

Phil Vachon is the manager of the Security Analytics and Identity Architecture team in the CTO office at Bloomberg, leading a team of engineers working on problems related to network and infrastructure security, human and machine identity management, and data science.

Copyright held by author/owner.
Publication rights licensed to ACM.

Article development led by [acmqueue](https://queue.acm.org)
queue.acm.org

Hardware security is not assured.

BY EDLYN V. LEVINE

The Die Is Cast

IN 2011, A fictitious company was created by the U.S. Government Accountability Office (GAO) to gain access to vendors of military-grade integrated circuits (ICs) used in weapons systems. Upon successfully joining online vendor platforms, the GAO requested quotes for bogus part numbers not associated with any authentic electronics components. No fewer than 40 offers returned from vendors in China to supply the bogus chips, and the GAO successfully obtained bogus parts from a handful of these vendors.³ The ramifications of the GAO findings are stark: The assumption of trusted hardware is inappropriate to invoke for cybersecure systems.

Injection of counterfeit electronics into the market is only a subset of vulnerabilities that exist in the global IC supply chain. Other types of attacks include trojans built into the circuitry, piracy of intellectual property, and reverse engineering. Modern ICs are exceptionally complex devices, consisting of upward of billions of transistors, miles of micron-scale interconnecting wires, advanced packaging configurations, and multisystem integration into chips sized on the order of a U.S. quarter. These ICs are designed, manufactured, and assembled by an equivalently complicated, globally distributed supply chain. A semiconductor company can have more than 16,000 suppliers spread around the world.¹⁰ While global-



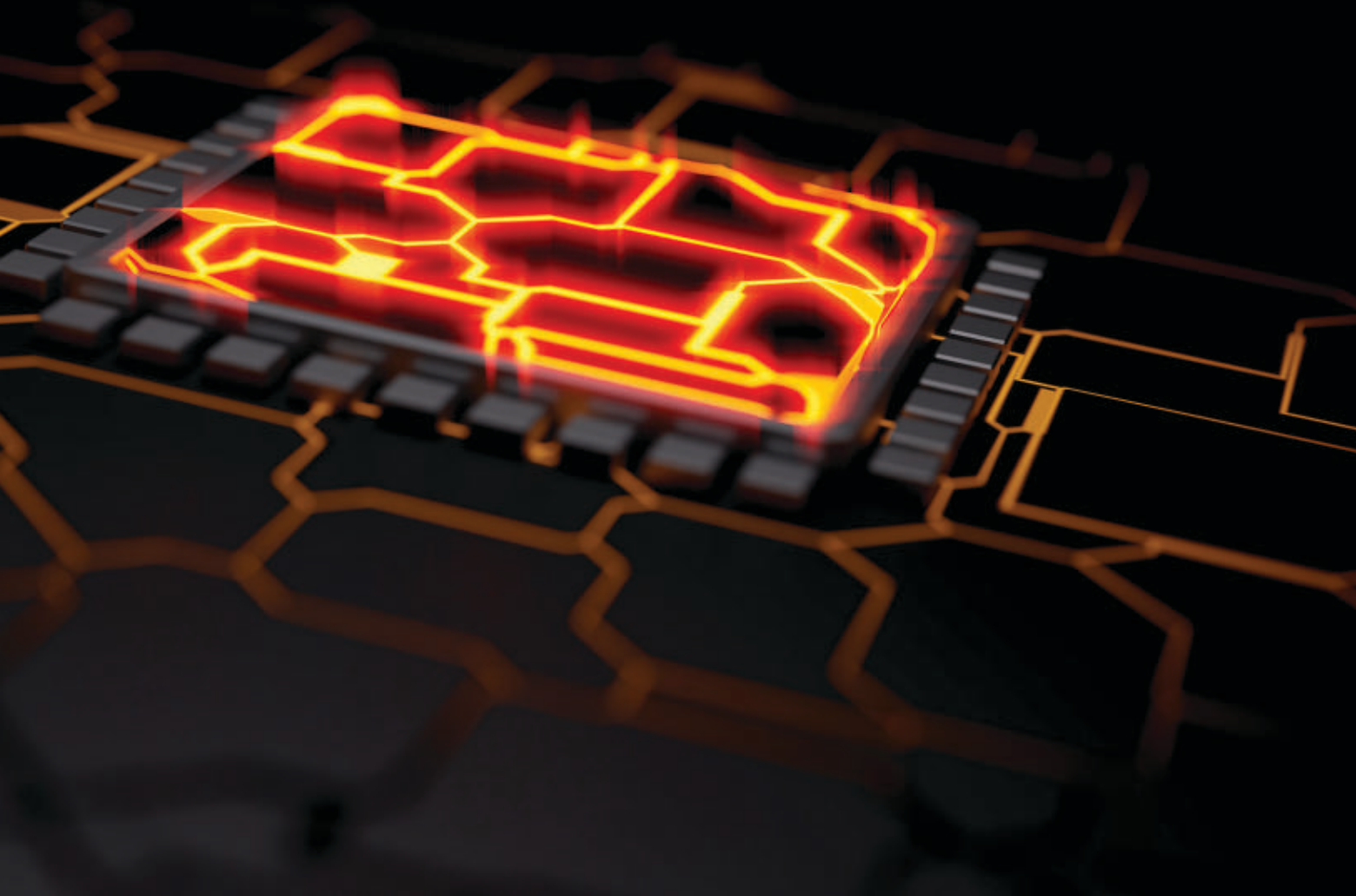


IMAGE BY DBTL/ESCAPISM

ization has drastically reduced industry costs by tapping inexpensive labor markets and economies of scale, it has simultaneously opened many windows of opportunity for attackers to maliciously modify hardware without the knowledge of original device manufacturers (ODMs) or their customers.

The tenet that “trust starts in silicon” underscores hardware as the root of security upon which software protections are implemented. Secure systems cannot be architected on a foundation of compromised hardware. Unlike software, there is no patch update that can fix a malicious hardware insertion short of replacing the device. Securing hardware is a multifaceted problem consisting of shoring

up the manufacturing chain, developing robust means to detect malicious insertions, and designing systems to be secure against the inevitability of hardware compromise.

Innovative research efforts spanning DARPA’s TRUST (Trusted Integrated Circuits) program to its LADS (Leveraging the Analog Domain for Security) program emphasize the increasing spotlight on hardware security as do high-profile reports ranging from the Defense Science Board to the President’s Council of Advisors on Science and Technology. Modern economies and critical systems depend on IC technologies, making the ramifications of hardware attacks increasingly dire.

The Spectrum of Invasive Hardware Attacks

An invasive hardware attack consists of changing the physical layout of a single IC or assembly of ICs. Specific classes of attacks include hardware trojans that modify the layout of a legitimate IC during design and fabrication, counterfeit attacks that substitute an illegitimate chip for a legitimate one, and assembly attacks that include incorporating additional ICs in the end-user device. (This last type of attack was the subject of a now-famous 2018 *Bloomberg Businessweek* article concerning datacenter motherboards.⁸ Even if the events of that article are

From Specs and Sand to Semiconductors: How ICs Are Made

Break open your laptop and you will find on the order of 100 to 1,000 ICs. These range from the CPU to microprocessors to memory. Each of these circuits has crossed the globe multiple times, moving among geographically distributed supply-chain vendors during their evolution from an initial specification to final assembly as a component in the machine sitting in your home or office. IC manufacturing can be broken into three primary stages—design, fabrication, and assembly and testing—each of which presents opportunities for hardware to be altered or assembled systems to be compromised.

Specifications and Design

Designing a new IC begins once the desired specifications for the chip are established. The specs determine the required performance of a chip for a targeted environment, including function, power, size, and timing. Semiconductor design is typically undertaken by teams of engineers who translate the IC specification into a register transfer level (RTL) description of the circuit in an HDL (hardware description language) such as VHDL (Very High-speed Integrated Circuit HDL) or Verilog. The RTL description is synthesized into a gate-level netlist using the logic gates and components from the desired technology library. The netlist is then converted to the transistor level with a fully placed and routed physical layout (shown in a GDSII file, the standard format used to represent the layout) using electronic design automation (EDA) software, thereby completing the circuit description.

Design is undertaken by both IDMs (integrated device manufacturers) that own fabrication facilities and fabless semiconductor companies that outsource semiconductor manufacturing. Throughout the design process, engineers incorporate IP from external vendors. The third-party IP companies develop and license circuit blocks, called IP cores, that are integrated into the overall design of a new chip. IP cores can take the form of synthesizable RTL or of a GDSII representation of the fully placed and routed core design. Leading IP vendors can have their IP cores included in tens of billions of chips manufactured each year.

Fabrication

Completed GDSII files are sent to a semiconductor fabrication facility, called a foundry, for manufacturing. Foundries are either owned and operated by IDMs or exist as stand-alone fabrication companies contracted by fabless semiconductor companies. GDSII files are converted by the foundry or a third party into mask sets that are used for patterning the physical circuit layout into layers in a silicon wafer during photolithography.

The full fabrication process includes multiple steps of material deposition, etching, and patterning, along with the processes of ion implantation and annealing that fine-tune electrical properties of the integrated elements. Once the transistor level has been fabricated, patterned metal wires are deposited to link transistor elements. The geometrical configuration of these interconnections is optimized for the functional specification of the chip, with complex ICs having upward of 20 metal layers. A completed fabricated wafer is tested and cut into individual silicon chips (dies) that are shipped for assembly and further testing.

Assembly, Testing, and Distribution

The packaging of individual silicon dies creates a protective interface between the die and the external environment. Package integration incorporates the silicon die with package wiring, substrates, heat spreaders, and ground planes, thereby creating the required electrical, mechanical, and thermal environment for the chip to interface properly with an external system. The packaged ICs are tested, binned according to performance, and distributed to electronics assembly plants that incorporate the ICs into end-user products.

not verifiable, the attack described represents a realistic threat vector.)

An invasive attack seeks to incorporate a malicious capability in an end-user device. An overt attack has signatures that are potentially detectable by the targeted system once implemented. Examples include kill switches that destroy a system's function, backdoors that enable illegitimate access, and control circuitry that changes a system's behavior. A covert attack seeks to operate undetected for long periods of

time, often with the objective of collecting information to route to the attacker, and may never be detected. The execution of a hardware attack requires knowledge of how ICs are fabricated and how they can be compromised.

Semiconductor manufacturing includes hundreds of steps from specification to distribution, providing many opportunities for invasive attacks (see the accompanying sidebar). Counterfeit attacks and assembly attacks are conducted during the assembly,

distribution, and second-hand supply-chain stages. Insertion of malicious hardware trojans can occur at any stage during IC manufacturing.

Trojans can be categorized according to the fabrication step at which they are inserted, to yield insight into supply-chain risk mitigation. The three classes of trojan insertion are pre-silicon, in-silicon, and post-silicon. Trojans range in their impact on IC performance (function change, backdoors, kill switches, decrease in service lifetime, information leakage), their activation mechanism (always on, internally triggered, externally triggered), their physical location on the chip (I/O, logic, memory, power distribution, clock), and the hardware abstraction level at which they occur.⁴

A pre-silicon attack occurs during the specification and design stages. A trojan can be inserted by changing functional characteristics during specification, such as timing or power consumption, or by modifying features at different hardware-abstraction layers during design, such as register transfer level (RTL), gate level, transistor level, and place-and-route. Every stage of design and every software tool used during design is a potential security vulnerability. The pervasive use of third-party IP cores and standard cell libraries in circuit design affords increased opportunity for external parties to insert malicious functionality. Computer-aided design tools can be tampered with to create compromised IC designs.⁹ Malicious modifications can even be made during the inclusion of design for test functionality before a design is sent to fabrication.

An in-silicon attack occurs during fabrication. An attack of this type requires both detailed knowledge of and access to manufacturing stages for the targeted device. These attacks can range from editing or exchanging the masks to altering the types or concentrations of chemicals used during fabrication. Changing the fine-tuned electrical properties of IC materials can have serious impacts on the function and lifetime of the device. Altering transistor dopant concentration can impact circuit function,¹ and altered composition or dimension of interconnects can lead to increased electromigration of metal atoms and early circuit failure.

A post-silicon attack is conducted after fabrication is completed. Attacks that can occur at this stage include circuit editing, modified package-level circuitry, untrusted testing that fails to reveal trojans, package counterfeiting, and malicious assembly of trusted ICs on a printed circuit board. Assembly attacks can manifest as the inclusion of unwanted ICs or the use of unshielded connections between trusted ICs and their environment, giving rise to electromagnetic-coupling-mediated information leakage.

Detecting Invasive Attacks

Many variants of hardware trojans can be implemented to achieve a range of attacks: from the addition of extra transistors creating new logic to the modification of the wire width of the clock distribution network introducing clock skew. Overt kill switches and shortening of service lifetimes to covert backdoors and information leakage also have different activation mechanisms. Some trojans are always on, whereas others require either internal or external triggers for attack payload activation. A universal objective for all trojans, however, is to escape detection throughout manufacturing and deployment until the trojan's attack is executed.

A trojan is designed to be of minimal size and consume minimal resources on a chip, posing a serious challenge to any effort to detect it. Because of the potential impact of hardware attacks, extensive research efforts have led to the development of sophisticated means of detecting trojans, but there is no smoking gun that ensures the trust of an IC. In principle, detection can be accomplished either by activating the trojan and observing its impact on chip performance compared with known performance specifications, or by comparing the questionable design or fabricated chip with the physicality and functionality of a trusted (golden) copy. Methods for detecting pre-silicon attacks differ from those for in- and post-silicon attacks, the latter ranging from nondestructive to destructive.

Detecting trojans in IC designs requires evaluating and ensuring the trust of third-party IP cores, libraries, and electronic design-automation



Trojans can be categorized according to the fabrication step at which they are inserted, to yield insight into supply-chain risk mitigation.



tools. This is not easy. IP cores are challenging to verify for trust since there is no golden version with which to compare. As such, establishing trust in IP cores typically takes the form of searching for unexpected components or signal output during design performance testing. Internal verification of IP functionality and code coverage analysis is used to identify suspect components and signals.

Automatic test pattern generation (ATPG) uses digital signal inputs to sequentially generate output patterns from a simulation of the designed chip. ATPG can detect trojans consisting of modifications to the known functionality of the chip, but it will not be successful finding trojans that have *added* functionality, such as additional logic, to the design. Having no information about the additional logic makes it impossible for ATPG to conduct a directed search of all possible digital signal inputs that could cause trojan activation. Furthermore, a trojan that activates physical side-channel leakage will go undetected with ATPG alone.

Once the chip has been fabricated, a new suite of trojan-detection methods is brought to bear. Sophisticated tools such as scanning electron microscopy and picosecond imaging circuit analysis can be used to do a full teardown of an IC to extract its physical layout for comparison with a trusted design. This is expensive and time consuming, resulting in partial to full destruction of the device under test, and thus is infeasible for widescale testing of chips set to enter the consumer market.

More tractable, less thorough non-destructive physical inspection and electrical testing leverage everything from x-ray imaging to parametric testing of chip behavior. Other testing methods include trojan activation via ATPG on the physical device, as well as side-channel analysis. The latter method investigates the physical characteristics of the device under test, such as timing and power consumption, to compare with known or golden side-channel behavior. Process variations that naturally occur during the course of fabrication, however, decrease the efficacy of side-channel analysis for trojan detection.

There is as yet no assured way of definitively determining whether or not a

chip has been tampered with, despite the large arsenal of testing methods. In many cases the sheer volume of ICs, as well as the lack of access to sophisticated testing equipment, hinders assurance of devices on the market. Testing is typically done by the ODMs or third-party specialists. Testing methods make heavy use of established means used by the microelectronics industry to test for device quality assurance. These techniques, including performance assessment and failure analysis, similarly extend to counterfeit and assembly attacks. Although powerful, these methods are not comprehensive, and increasing emphasis is being placed on adopting either design for security or zero trust in IC manufacturing.

Broadening the Spectrum: Semi-Invasive and Non-Invasive Attacks

The notoriety of recent microarchitectural attacks such as Spectre and Meltdown clearly indicates the book on hardware security does not end with the supply chain. Latent vulnerabilities of trusted ICs can be taken advantage of using semi-invasive attacks such as fault injections and non-invasive attacks leveraging side channels. If you have ever been warned not to yell in a datacenter, you are familiar with the faults that can be introduced in disk-head readers by mechanical vibrations. Analogous fault injection can be introduced by physical coupling or manipulation of ICs. Many examples exist, ranging from corrupted memory isolation induced by disturbance errors injected into DRAM by repeated row hammering,⁶ to violations of trusted execution environments such as Arm TrustZone, to Intel SGX (Software Guard Extensions).⁵

The physical attack plane can also be leveraged for side-channel attacks such as Spectre and Meltdown. Unintended physical or microarchitectural signatures that manifest during the operation of the IC can be leveraged by an attacker to learn information about the circuit that allows the attacker either to compromise secure data or to yield access to secure functions. This was famously first demonstrated with timing attacks.⁷ Increasingly, designing for security seeks to understand and preempt the physical signatures of ICs at the de-

sign stage to anticipate or detect side-channel security vulnerabilities that manifest in the post-fabrication stage.

The Future of Hardware Security

Recognition of the importance of hardware security has shifted focus from traditional software threats to lower levels of the computing hierarchy. Research across hardware security areas from supply chains to side channels has led to a better understanding of hardware threats and increased development of detection and mitigation techniques. Resources such as the TrustHub Trojan database and conferences such as IEEE's HOST (Hardware-oriented Security and Trust) and PAINE (Physical Assurance and Inspection of Electronics) are signs of this shifting focus toward hardware security.

Despite the increased attention and growing corpus of research, no common standards or tools exist and no definitive solutions have been developed. The spectrum of invasive to non-invasive vulnerabilities at the physical attack plane makes hardware assurance a daunting if not insurmountable challenge. As with the rest of the cybersecurity community, hardware security benefits from the recognition that a prevention-only approach to assurance leaves systems vulnerable to successful attacks. This is analogous to a home security system solely dependent on an external fence, with no internal alarms, locks, safe rooms, or police response force should an intruder hop the barrier. As such, focus increasingly leans toward designing hardware capable of identifying, operating through, mitigating, and recovering from an attack.¹¹ However, the economic benefits of security often remain unclear due to the high cost of security and the prevalence of consumers who are willing to risk security for increased compute capability (or who are ignorant of the vulnerabilities).

The future of hardware security will evolve with hardware. As packaging advances and focus moves to beyond Moore's Law technologies, hardware security experts will need to keep ahead of changing security paradigms, including system and process vulnerabilities. Research focused on quantum hacking is emblematic of the translation of principles of security on the

physical attack plane for emerging communications and computing technologies.² Perhaps the commercial market will evolve such that the GAO will run a study on compromised quantum technologies in the not-too-distant future. □

Related articles on queue.acm.org

Why Is It Taking So Long to Secure Internet Routing?

Sharon Goldberg

<https://queue.acm.org/detail.cfm?id=2668966>

What is a CSO Good For?

Kode Vicious

<https://queue.acm.org/detail.cfm?id=3357152>

Building Systems to be Shared Securely

Poul-Henning Kamp and Robert Watson

<https://queue.acm.org/detail.cfm?id=1017001>

References

1. Becker, G. T., Regazzoni, F., Paar, C., Burleson, W.P. Stealthy dopant-level hardware trojans: extended version. *J. Cryptographic Engineering* 4(1) (2014), 19–31; <https://link.springer.com/article/10.1007/s13389-013-0068-0>.
2. Emerging Technology from the arXiv. The next battleground in the war against quantum hacking. *MIT Technology Rev.* (Aug. 20, 2014); <https://bit.ly/3ntYzKj>
3. GAO. DoD supply chain: suspect counterfeit electronic parts can be found on Internet purchasing platforms. GAO-12-375, 2012; <https://www.gao.gov/products/GAO-12-375>.
4. Karri, R., Rajendran, J., Rosenfeld, K., Tehranipoor, M. Trustworthy hardware: Identifying and classifying hardware trojans. *Computer* 43 (10) (2010), 39–46; <https://ieeexplore.ieee.org/document/5604161>.
5. Keegan, R. Hardware-backed heist: extracting ECDSA keys from Qualcomm's TrustZone. *NCC Group Whitepaper* (Apr. 23, 2019); <https://bit.ly/2GyRKPp>
6. Kim, Y., et al. Flipping bits in memory without accessing them: an experimental study of DRAM disturbance errors. *ACM SIGARCH Computer Architecture News* 42, 3 (2014) 361–372; <https://dl.acm.org/doi/10.1145/2678373.2665726>.
7. Kocher, P.C. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In *Proceedings of 1996 Advances in Cryptology*, N. Kobitz, Ed. LNCS 1109. Springer, Berlin, Heidelberg; https://link.springer.com/chapter/10.1007/3-540-68697-5_9.
8. Robertson, J., Riley, M. The big hack: How China used a tiny chip to infiltrate U.S. companies. *Bloomberg Businessweek* (Oct. 4, 2018); <https://bloom.bg/2PY108V>
9. Roy, J. A., Koushanfar, F., Markov, I.L. Extended abstract: Circuit CAD tools as a security threat. In *Proceedings of the 2008 IEEE Intern. Workshop on Hardware-Oriented Security and Trust*, 65–66; <https://ieeexplore.ieee.org/document/4559052>.
10. Semiconductor Industry Association. Nathan Associates. Beyond borders: the global semiconductor value chain, 2016; <https://bit.ly/36Dkd8V>
11. Villasenor, J. The hacker in your hardware. *Scientific American* 303, 2 (2010), 82–87; <https://www.scientificamerican.com/article/the-hacker-in-your-hardware/>.

Edlyn V. Levine is Chief Engineer of MITRE Engenuity and a research associate in the Department of Physics at Harvard University. She is internationally recognized for her contributions in information technology as an AFCEA 40-under-40 award winner.

Copyright held by author/owner.
Publication rights licensed to ACM

Attention: Undergraduate and Graduate Computing Students

There's an **ACM Student Research Competition (SRC)**
at a SIG Conference of interest to you!



Association for Computing Machinery
Advancing Computing as a Science & Profession



It's hard to put the **ACM Student Research Competition** experience into words, but we'll try...



"Attending ACM SRC was a transformative experience for me. It was an opportunity to take my research to a new level, beyond the network of my home university. Most important, it was a chance to make new connections and encounter new ideas that had a lasting impact on my academic life. I can't recommend ACM SRC enough to any student who is looking to expand the horizons of their research endeavors."

David Mueller
North Carolina State University | SIGDOC 2018



"Participating in the ACM SRC was a unique opportunity for practicing my presentation skills, getting feedback on my work, and networking with both leading researchers and fellow SRC participants. Winning the competition was a great honor, a motivation to continue working in research, and a useful boost for my career. I highly recommend any aspiring student researcher to participate in the SRC."

Manuel Rigger
Johannes Kepler University Linz, Austria | Programming 2018



"The SRC was a great chance to present early results of my work to an international audience. Especially the feedback during the poster session helped me to steer my work in the right direction and gave me a huge motivation boost. Together with the connections and friendships I made, I found the SRC to be a positive experience."

Matthias Springer
Tokyo Institute of Technology | SPLASH 2018



"I have been a part of many conferences before both as an author and as a volunteer but I found SRC to be an incredible conference experience. It gave me the opportunity to have the most immersive experience, improving my skills as a presenter, researcher, and scientist. Over the several phases of ACM SRC, I had the opportunity to present my work both formally (as a research talk and research paper) and informally (in poster or demonstration session). Having talked to a diverse range of researchers, I believe my work has much broader visibility now and I was able to get deep insights and feedback on my future projects. ACM SRC played a critical role in facilitating my research, giving me the most productive conference experience."

Muhammad Ali Gulzar
University of California, Los Angeles | ICSE 2018



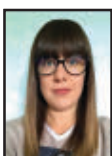
"At the ACM SRC, I got to learn about the work done in a variety of different research areas and experience the energy and enthusiasm of everyone involved. I was extremely inspired by my fellow competitors and was happy to discover better ways of explaining my own work to others. I would like to specifically encourage undergraduate students to not hesitate and apply! Thank you to all those who make this competition possible for students like me."

Elizaveta Tremsina
UC Berkeley | TAPIA 2018



"The ACM SRC was an incredible opportunity for me to present my research to a wide audience of experts. I received invaluable, supportive feedback about my research and presentation style, and I am sure that the lessons I learned from the experience will stay with me for the rest of my career as a researcher. Participating in the SRC has also made me feel much more comfortable speaking to other researchers in my field, both about my work as well as projects I am not involved in. I would strongly recommend students interested in research to apply to an ACM SRC—there's really no reason not to!"

Justin Lubin
University of Chicago | SPLASH 2018



"Joining the Student Research Competition of ACM gave me the opportunity to measure my skills as a researcher and to carry out a preliminary study by myself. Moreover, I believe that "healthy competition" is always challenging in order to improve yourself. I suggest that every Ph.D. student try this experience."

Gemma Catolino
University of Salerno | MobileSoft 2018

Check the SRC Submission Dates: <https://src.acm.org/submissions>

- ◆ Participants receive: \$500 (USD) travel expenses
- ◆ All Winners receive a medal and monetary award. First place winners advance to the SRC Grand Finals
- ◆ Grand Finals Winners receive a handsome certificate and monetary award at the ACM Awards Banquet

Questions? Contact Nanette Hernandez, ACM's SRC Coordinator: hernandez@hq.acm.org



DOI:10.1145/3426361

Facebook labels 67% of its users with potential sensitive interests, sometimes at great risk to the user.

BY JOSÉ GONZÁLEZ CABAÑAS, ÁNGEL CUEVAS, ARITZ ARRATE, AND RUBÉN CUEVAS

Does Facebook Use Sensitive Data for Advertising Purposes?

CITIZENS WORLDWIDE HAVE demonstrated serious concerns regarding the management of personal information by online services. For instance, the 2015 Eurobarometer about data protection¹³ reveals that: 63% of citizens within the European Union (EU) do not trust online businesses, more than half do not like providing personal information in return for free services, and 53% do not like that Internet companies use their personal information in tailored advertising. Similarly, a recent survey carried out among U.S. users⁹

reveals that 53% of respondents were against receiving tailored ads from the information websites and apps learn about them, 42% do not think websites care about using users data securely and responsibly at all, and 73% considers websites know too much about users. A survey conducted by Internet Society (ISOC) in the Asia-Pacific region in 2016⁸ disclosed that 59% of the respondent did not feel their privacy is sufficiently protected when using the Internet, and 45% considered getting the attention of policymakers in their country on data protection a matter or urgency.

Policymakers have reacted to this situation by passing or proposing new regulations in the area of privacy and/or data protection. For instance, in May 2018, the EU enforced the General Data Protection Regulation (GDPR)⁶ across all 28 member states. Similarly, in June 2018, California passed the California Consumer Privacy Act,³ which is claimed to be the nation's toughest data privacy law. In countries like Argentina or Chile, the governments proposed new bills in 2017 updating their existing data protection regulation.¹¹ For this article, we will take as reference the GDPR since it is the one affecting more countries, citizens, and companies.

The GDPR (but also most data protection regulations) define some categories of personal data as sensitive and prohibits processing them with limited exceptions (for example, the user provides explicit consent to process that sensitive data for a specific

» key insights

- **67% of FB users, which corresponds to 22% worldwide citizens, are labeled with some potentially sensitive ad preferences.**
- **The EU's GDPR had a negligible impact on FB regarding the use of sensitive ad preferences for commercial purposes**
- **In October 2018, FB labeled 540k users in Saudi Arabia with the ad preference "Homosexuality." As of Nov. 11, 2020, this number was still 250k users. We observe the same issue in other countries where, like Saudi Arabia, homosexuality is punished with the death penalty.**

ILLUSTRATION BY CHARIS TSEVIS



purpose). In particular, the GDPR defines as sensitive personal data as: “data revealing racial or ethnic origin, political opinions, religious or philosophical beliefs, or trade union membership, and the processing of genetic data, biometric data for the purpose of uniquely identifying a natural person, data concerning health or data concerning a natural person’s sex life or sexual orientation.”

In a recent work,² we demonstrated that Facebook (FB) labels 73% of users within the EU with potentially sensitive interests (referred to as ad preferences as well), which may contravene the GDPR. FB assigns user’s different ad preferences based on their online activity within this social network. Advertisers running ad campaigns can target groups of users that have been assigned a particular ad preference (for example, target FB users interested in Starbucks). Some of these ad preferences may suggest political opinions (for example, Socialist party), sexual orientation (for example, homosexuality), personal health issues (for example, breast cancer awareness), and other potentially sensitive attributes. In the vast majority of the cases, the referred sensitive ad preferences are inferred from the user behavior in FB without obtaining explicit consent from the user. Then advertisers may reach FB users based on ad preferences tightly linked to sensitive information. For instance, one of the authors of this article received the ad shown in

Figure 1 (left side). The text in the ad clearly reflects the ad was targeting homosexual people. The author had not explicitly defined his sexual orientation, but he discovered that FB had assigned him the “Homosexuality” ad preference (see Figure 1 right side).

First, this article extends the scope of our analysis from the EU to 197 countries worldwide in February 2019. We quantify the portion of FB users that have been assigned ad preferences linked to potentially sensitive personal data across the referred 197 countries.

Second, we analyze whether the enactment of the GDPR on May 28, 2018 had some impact on the FB practices regarding the use of sensitive ad preferences. To this end, we compare the number of EU users labeled with potentially sensitive ad preferences in January 2018, October 2018 and February 2019 (five months before, five months after and nine months after the GDPR was enacted, respectively).

Third, we discuss privacy and ethics risks that may be derived from the exploitation of sensitive FB ad preferences. As an illustrative example, we quantify the portion of FB users labeled with the ad preference Homosexuality in countries where homosexuality is punished even with the death penalty.

Finally, we present a technical solution that allows users to remove in a simple way the sensitive interests FB has assigned them.

Background

Advertisers configure their ad campaigns through the FB Ads Manager.^a It allows advertisers to define the audience (that is, user profile) they want to target with their advertising campaigns. It can be accessed through either a dashboard or an API. The FB Ads Manager offers advertisers a wide range of configuration parameters such as (but not limited to): location (country, region, and so on), demographic parameters (gender, age, among others), behaviors (mobile device, OS and/or Web browser used, and so on), and interests (sports, food). The interest parameter is the most relevant for our work. It includes hundreds of thousands of possibilities capturing users’ interest of any type. The FB Ads Manager provides detailed information about the configured audience. The most relevant element for this article is the *Potential Reach* that reports the number of monthly active users in FB matching the defined audience.

In parallel, FB assigns to each user a set of ad preferences, that is, a set of interests, derived from the data and activity of the user on FB. These ad preferences are indeed the interests offered to advertisers in the FB Ads Manager.^b Therefore, if a user is assigned “Watches” within her list of ad preferences, she will be a potential target of any FB advertising campaign configured to reach users interested in watches. It is important to note that ad preferences in the FB ad ecosystem are available worldwide, thus there are not specific ad preferences per country.

The dataset used in this work is obtained from the data collected with our FDVT Web browser extension.¹ The Data Valuation Tool for Facebook Users (FDVT) is a Web browser extension currently available for Google Chrome^c and Mozilla Firefox.^d The FDVT main functionality is to provide users with a real-time estimation of the revenue they generate for FB out of the ads they receive in FB. To compute that estimation we obtain from the FB API the price advertisers are willing to pay to display ads

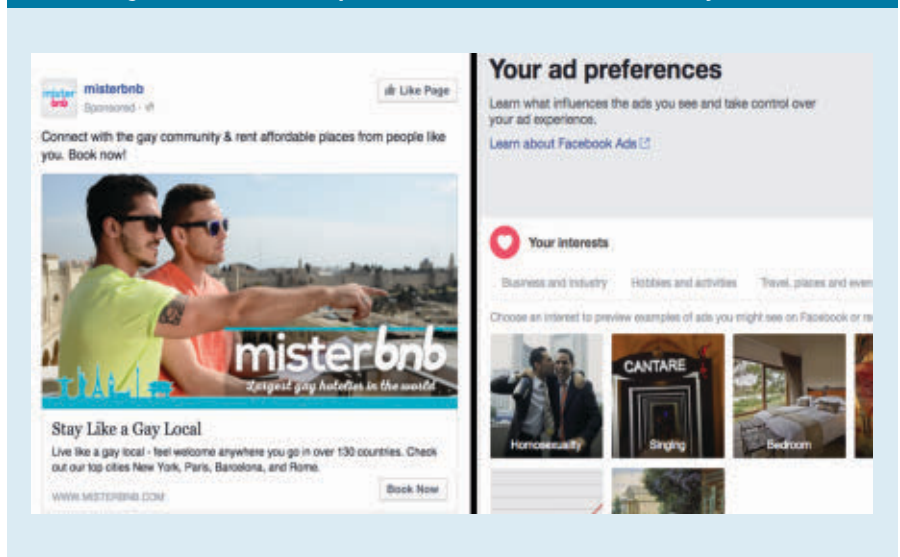
a <https://www.facebook.com/ads/manager>

b Given that interests and ad preferences refer to the same thing, we use these two terms interchangeably in the rest of the article.

c <https://bit.ly/3iMoytw>

d <https://addons.mozilla.org/firefox/addon/fdvt>

Figure 1. Snapshot of an ad received by one of the authors of this article and ad preference list showing that FB inferred this person was interested in homosexuality.



and gather clicks from users with the same profile as the FDVT user and quantify the number of ads the FDVT user receives and clicks during a Facebook session. The FDVT collects (among other data) the ad preferences FB assigns to the user by parsing the user's ad preferences' page^e where any user can find her ad preferences' list. It is important to note that all FDVT users granted us explicit permission to use the collected information (in an anonymous manner) for research purposes. We leverage this information to identify potentially sensitive ad preferences assigned to users that have installed the FDVT.

Finally, for any ad preference, we can query the FB Ads Manager API to retrieve the Potential Reach (that is, FB active users) associated with any FB audience. Hence, we can obtain the number of FB users in any country (or group of countries) that have been assigned a particular interest (or group of interests).

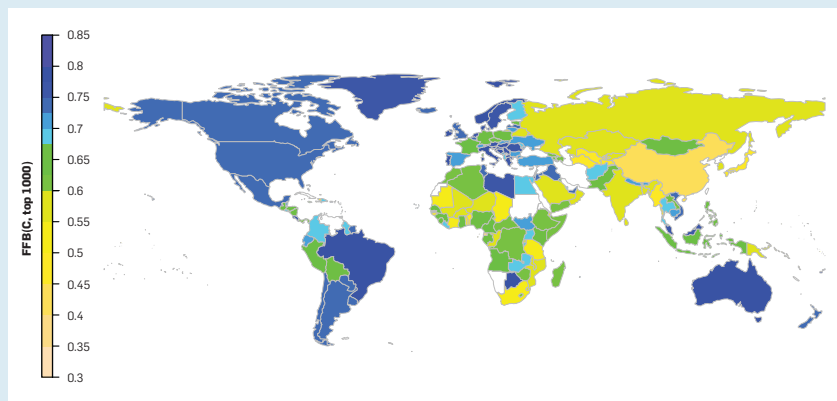
Data and Methodology

We seek to quantify the number of FB users that have been assigned potentially sensitive ad preferences across 197 countries in February 2019. To this end, we follow a two-step process.

First, we identify likely sensitive ad preferences within five of the relevant categories listed as *Sensitive Personal Data* by the GDPR: racial or ethnic origin, political opinions, religious or philosophical beliefs, health, and sexual orientation. This article reuses the list of 2092 potentially sensitive ad preferences we obtained in Cabañas et al.² out of analyzing more than 126k unique ad preferences assigned 5.5M times to more than 4.5k FDVT users.

To extract that list we first implemented an automatic process to reduce the list of 126k ad preferences to 4,452 likely sensitive ad preferences. Next, we recruited a group of 12 panelists who manually classified the 4,452 ad preferences into sensitive, in case they could be assigned to some of the five sensitive categories referred above, or non-sensitive. All of the panelists are researchers (faculty or Ph.D. students) with some knowledge in the

Figure 2. Choropleth map of the number of FB users assigned potentially sensitive ad preferences (FFB(C,1000)) for the 197 countries analyzed in the article.



area of privacy. Each ad preference received five votes, and we used majority voting¹⁰ to classify each ad preference either as sensitive or non-sensitive. Overall, 2092^f out of the 4,452 ad preferences were labeled as sensitive. We referred to this subset of 2,092 ad preferences as the *suspected sensitive subset*. We collected this set in January 2018 and checked that 2,067 out of these 2,092 potentially sensitive ad preferences were still available within the FB Ads Manager in February 2019.

Second, we leveraged the FB Ads Manager API to retrieve the portion of FB users in each country that had been assigned at least one of the Top N (with N ranging between 1 and 2,067) potentially sensitive ad preferences from the suspected sensitive subset. In particular, we retrieve how many users in a given country are interested in ad preference 1 OR ad preference 2 OR ad preference 3... OR ad preference N. An example of this for N = 3 could be “How many people in France are interested in Pregnancy OR Homosexuality OR Veganism.” We have defined the following metric that we use in the rest of the article.

–**FFB(C,N)**. Percentage of FB users in country C that have been assigned at least one of the top N potentially sensitive ad preferences from the suspected sensitive subset. We note C may also refer to all the countries forming a particular region (for example, EU, Asia-Pac-

ic, U.S., among others). $FFB(C,N)$ is computed as the ratio between the number of FB users that have been assigned at least one of the top N potentially sensitive ad preferences and the total number of FB users in country C. Finally, it is important to note that the FB Ads Manager API only allows creating audiences with at most $N = 1,000$ interests. Therefore, in practice, the maximum value of N we can use to compute FFB is 1,000.

Exposure of FB Users to Potentially Sensitive Ad Preferences

We have computed the portion of FB users that have been assigned some of the 2,067 potentially sensitive ad preferences within 197 different countries. Figure 2 shows a choropleth map of $FFB(C,1000)$ for those countries in February 2019.

If we consider the 197 altogether, 67% of FB users are tagged with some potentially sensitive ad preference. This portion of users corresponds to 22% of citizens across the 197 analyzed countries according to the population data reported by the World Bank.^g However, FFB shows an important variation across countries.

We find the most impacted country is Malta where 82% of FB users are assigned some potentially sensitive ad preference. Contrary, the least impacted country is Equatorial Guinea where 37% of FB users are assigned potentially sensitive ad preferences.

More interesting, an overview of the map seems to suggest that western countries have a higher exposure to

e <https://www.facebook.com/ads/preferences/edit>

f <https://fdvt.org/usenix2018/panelists.html>. This resource includes the list of all potentially sensitive ad preferences manually labeled by the panelists along with the 5 votes each of them received from the panelists.

g <https://data.worldbank.org>

Table 1. Pearson correlation and p_value between FFB and six socioeconomic development indicators of the country.

Indicator	correlation FFB	p_value
FB penetration	0.544	2.2e-16
Expected Years of School	0.444	7.249e-09
Access to a mobile phone or Internet at home (% age 15+)	0.395	1.478e-06
GDP per capita (current USD)	0.381	5.733e-08
Voice and Accountability	0.372	1.142e-07
Birth rate, crude (per 1,000 people)	-0.455	4.922e-11

Table 2. Percentage of FB users (FFB) within Africa, America, Asia, Europe, and Oceania assigned some sensitive ad preferences from a list of 15 expert-verified sensitive ad preferences as non-GDPR compliant. Last column “World” shows FFB for the aggregation of all 197 considered countries. Last row shows the result for the 15 ad preferences aggregated.

Ad preference	Africa	America	Asia	Europe	Oceania	World
Alternative medicine	3.40	11.35	3.27	7.17	10.82	6.26
Bible	13.28	14.65	6.31	8.13	14.61	9.68
Buddhism	2.87	5.38	10.36	4.13	7.19	7.23
Feminism	3.22	9.27	2.08	6.52	10.84	5.01
Gender identity	0.08	0.46	0.07	0.20	0.60	0.21
Homosexuality	2.66	7.93	2.27	6.07	8.48	4.57
Illegal immigration	0.26	0.15	0.02	0.03	0.07	0.08
Judaism	11.06	3.72	1.91	.24	2.44	3.33
Lgbt community	3.93	13.89	5.39	11.94	14.82	8.79
Nationalism	1.82	1.11	1.28	1.32	0.95	1.28
Oncology	1.30	1.33	0.38	0.84	0.97	0.81
Pregnancy	11.75	19.17	11.58	17.09	21.41	14.71
Reproductive health	0.36	0.24	0.17	0.07	0.09	0.19
Suicide prevention	0.05	0.30	0.03	0.08	1.02	0.13
Veganism	5.97	14.18	6.83	16.98	22.78	10.61
Union	30.43	40.66	27.62	38.25	46.92	33.45

potentially sensitive ad preferences compared to Asian and African countries. To quantify these effects we have computed the Pearson correlation of the FFB metric with the following socio-economic indicators: FB penetration; expected years of school; access to a mobile phone or Internet at home; GDP per capita; voice and accountability; and birth rate. Note that Western developed countries show higher values in all the indicators but birth rate. Hence, we hypothesize that we will find a positive correlation between FFB and all the indicators but birth rate. Table 1 shows the results of the referred correlations.

The results corroborate our hypothesis since all the indicators but birth rate are positively correlated with FFB. In summary, the results validate our initial observation that FB users in western

developed countries are more exposed to be labeled with sensitive ad preferences than users in Africa and Asia. It is interesting to observe that in the case of South-America we observe a similar pattern in which the most powerful economies and developed countries such as Brazil, Chile, and Argentina show higher exposure to sensitive ad preferences than other countries in South-America.

Exposure of FB Users to Very Sensitive Ad Preferences

Although legislation tries to define what sensitive data is, some people might think that not all different sensitive data items are equally sensitive. For instance, data revealing sexual orientation from somebody could be considered more sensitive than, for example, data showing that one user may be

affected by flu. Therefore, the level of sensitivity of our list of interests is very likely subjective and will depend on each person personal perception.

Here, we zoom in our analysis to a narrowed list of interests that match undoubtedly with the definition of the GDPR for the case of sensitive personal data. We examined a subset of 15 ad preferences not compliant with the GDPR definition of sensitive personal data. We supported our statement asking for validation by an expert from the Spanish Data Protection Authority (DPA). This expert, with both a very deep knowledge of the GDPR and a technical background that allow him perfectly understanding the FB advertisement ecosystem, verified that in his opinion these 15 ad preferences do not comply with the GDPR.

We retrieve the portion of FB users assigned in each of the 197 countries analyzed that have been assigned each of the 15 expert verified ad preferences and the aggregation of them. Since it is unfeasible to show the results for each of the countries within the paper, we have grouped them into five continents: Africa, America, Asia, Europe, and Oceania. To obtain the desegregated results for each country we refer the reader to the following external link.^h

Table 2 shows FFB for each of the expert-verified sensitive ad preferences within the five continents. Besides, the last row referred to as *Union* shows the aggregated results considering all the 15 interests within a group, while the last column *World* depicts the overall results considering all 197 countries. The results show that when considering all the 197 countries 33% of FB users, which corresponds to almost 11% of citizens within those countries, have been labeled with some of the 15 sensitive interests in the table. As it was expected from the correlation results depicted in the previous section, Asia and Africa are showing the lowest values of FFB (27.62% and 30.43%, respectively). The exposition of FB users grows up to 38.25.

If we look in detail some of the ad preferences in the table, we observe

^h https://fdvt.org/world_sensitivities_2019/display_sensitivities.html. This resource is a website in which the reader can select any country in the world and obtain the percentage of users in that country that have been assigned each of the 15 very sensitive ad preferences listed in Table 2.

that the portion of users worldwide labeled with the ad preference homosexuality is almost 5%. This number doubles for the ad preference bible (intimate related to one particular religious belief), and grows up to almost 15% for pregnancy.

Comparison of EU FB Users Exposure to Potentially Sensitive Ad Preferences Before and After GDPR Enforcement

This section aims to analyze whether the GDPR enforcement had some effect on minimizing the use of potentially sensitive ad preferences in the EU. To that end we compare the exposure of EU users to potentially sensitive ad preferences in January 2018² (five months before the GDPR was enforced) to the exposure measured in October 2018 and February 2019 (five and nine months after the GDPR was enforced, respectively).

The first relevant change is that FB had removed 19 ad preferences in October 2018 and 25 in February 2019 from the set of 2,092 potentially sensitive ad preferences we retrieved on January 2018. Although this is a negligible amount, it is worth noting that five of the removed ad preferences are: Communism, Islam, Quran, Socialism, and Christianity. These five ad preferences were included in an initial set of 20 ad preferences verified by the DPA expert as very sensitive. Although we observe the removal of these five elements happened around the GDPR enforcement (between January 2018 and October 2018) we do not know whether the actual reason why FB deleted those ad preferences was a reaction to the GDPR or there was a different motivation.

Figure 3 shows the FFB difference in percentage points between the results obtained in January 2018 and October 2018 (grey bar); and between January 2018 and February 2019 (black bar) across the 28 EU countries, and the EU aggregated labeled as EU28.

If we consider the results of October 2018, we observe that the portion of users labeled with potentially sensitive ad preferences was lower in all EU countries but Spain after the GDPR enforcement (that is, compared to the data obtained in January 2018). However, the aggregated EU reduction is rather small, only three percentage points.

The largest reduction is 7.33 percentage points in the case of Finland.

The slight reduction observed in the results obtained in October 2018 seems to disappear when we observe the results from February 2019. There are 13 countries where the portion of users labeled with potentially sensitive data is higher in February 2019 as compared to January 2018. Overall, the aggregated results show that the portion of users labeled with potentially sensitive ad preferences in February 2019 is only 1% less than in January 2018.

In summary, the overall impact of the GDPR to prevent FB of using potentially sensitive ad preferences for advertising purposes is negligible.

Ethics and Privacy Risks Associated with Sensitive Personal Data Exploitation

The possibility of reaching users labeled with potentially sensitive personal data enables the use of FB ad campaigns to attack (for example, hate speech) specific groups of people based on sensitive personal data (ethnicity, sexual orientation, religious beliefs, and so on). Even worse, in Cabañas,² we performed a ball-park estimation showing that in average an attacker could retrieve personal identifiable information (PII) of users

tagged with some sensitive ad preference through a phishing-like attack⁷ at a cheap cost ranging between €0.015 and €1.5 per user, depending on the success ratio of the attack. Following, we describe other potential risks associated with sensitive ad preferences.

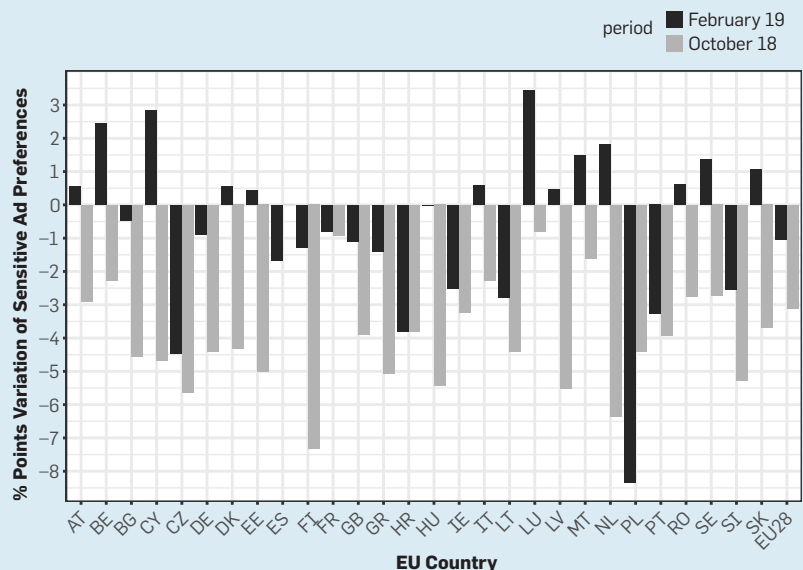
Recently, a journalist of the *Washington Post* wrote an article to denounce her own experience after she became pregnant.ⁱ It seems FB algorithms inferred that situation out of some actions she performed while browsing in FB. Probably FB labeled her with the ad preference “pregnancy” or some other similar and she started to receive pregnancy-related ads. Unfortunately, the journalist had a stillbirth but she kept receiving ads related to pregnancy, which exposed here to a very uncomfortable experience.

Another serious risk, which in our opinion is extremely worrying, is linked to the fact that many FB users are tagged with the interest “Homosexuality” in countries where being homosexual is illegal and may even be punished with the death penalty. There are still 78 countries in the world where homosexuality is penalized^j and a few of them

i <https://wapo.st/2FQrZ4d>

j https://ilga.org/downloads/2017/ILGA_World-Map_ENGLISH_Criminalisation_2017.pdf

Figure 3. Variation of FFB in percentage points for each EU country between: the data obtained in January 2018 and October 2018 (five months before and five months after the GDPR was enacted) represented by the grey bar; the data obtained in January 2018 and February 2019 (five months before and nine months after the GDPR was enacted) represented by the black bar. The last label (EU28) represents the results for all EU countries together.



where the maximum punishment is the death penalty. Table 3 shows the FFB metric results only considering the interest “Homosexuality” in countries that penalize homosexuality with the death penalty. For instance, in the case of Saudi Arabia, we found that FB assigns the ad preference “Homosexuality” to 540K users (2.08% of all FB users in that country). In the case of Nigeria 620k (2.35% of all FB users in that country).

We acknowledge the debate regarding what is sensitive and what is not is a complex one. However, we believe FB should

take immediate actions to avoid worrying and painful situations like the ones exposed in this section, in which FB may unintentionally expose users to serious risks. The most efficient and privacy-preserving solution would be implementing an opt-in process in which users have to proactively accept receiving targeted ads. That solution would empower the users to avoid companies like FB to process personal data (including sensitive one) for advertising purposes, and, therefore, would alleviate the potential privacy risks associated to the use of sensitive ad pref-

erences for users that do not opt-in. However, that is unlikely to happen in the short-term. Meanwhile, a straightforward action should be stopping using the ad preference “Homosexuality” (or similar ones) in countries where being homosexual is illegal, and other very sensitive ad preferences like the 15 ones we list in this article.

FDVT Extension to Allow Users Removing Potentially Sensitive Ad Preferences

The results reported previously motivate the development of solutions that make users aware of the use of sensitive personal data for advertising purposes. In addition, it is also important to empower them to remove in a very simple manner those sensitive ad preferences they do not fill comfortable with. Unfortunately, the existing process FB offers is unknown and complex for most users. To this end, we have extended the FDVT browser extension to inform users about the potentially sensitive ad preferences that FB has assigned them, both the active ones but also those assigned in the past that are not currently active; or allow users to remove with a single click either all the active sensitive ad preferences or those individual ones users do not fill comfortable with.

We have introduced a new button in the FDVT extension interface with the label “Sensitive FB Preferences.” When a user clicks on that button, we display a page listing at the top the potentially sensitive ad preferences included in the user’s ad preference set (both the active ones and inactive ones). Figure 4 shows an example of this page. We provide the following information for each ad preference: Ad preference name; Topic; and, Sensitive, whether the ad preference is potentially sensitive (highlighted in yellow) or not. Besides, next to each ad preference there is a button *Delete Ad Preference* to individually remove those ad preferences. Moreover, we provide another button *More Info* to individually display the historical information for the ad preference, which includes the period(s) when the ad preference has been active and the reason why FB has assigned that ad preference to the user. Finally, at the top of the page we include a search bar to look for specific preferences and two buttons: *Delete All Sensitive Ad Preferences* and *Delete All Ad Preferences* to remove all currently active

Table 3. Percentage of FB users (FFB) tagged with the interest “Homosexuality” in countries where being homosexual may lead to death penalty. Note we do not include Iran and Sudan since FB is not providing information for those countries.

Code	Country	Homosexuality
AF	AFGHANISTAN	12.31
MR	MAURITANIA	0.99
QA	QATAR	2.35
SO	SOMALIA	1.44
PK	PAKISTAN	1.54
AE	UNITED ARAB EMIRATES	3.00
BN	BRUNEI	5.24
NG	NIGERIA	2.35
SA	SAUDI ARABIA	2.08
YE	YEMEN	1.08
IQ	IRAQ	3.20

Figure 4. Snapshot of FDVT new feature to allow users deleting sensitive ad preferences.

Checking & Deleting Sensitive Ad Preferences

Look for any ad preference...

Total #Ad Preferences: Active: 4 - Removed: 2 - Inactive: 2

Preference Name	Topic	Sensitive	Remove	More Info	Status
Homosexuality	Lifestyle and culture	Sensitive	Delete Ad Preference	More Info	ACTIVE
Democracy	Lifestyle and culture	Sensitive	Delete Ad Preference	Less Info	ACTIVE

HISTORICAL INFORMATION

This ad preference appeared in your profile in the following periods:

From 2016-09-16 to 2016-09-20. Reason: You have this preference because you clicked on an ad related to Democracy.

From 2019-01-14 to NOWADAYS. Reason: You have this preference because you liked a Page related to Democracy.

Coupons	Shopping and fashion	Non-Sensitive	Delete Ad Preference	More Info	ACTIVE
Shopping	Shopping and fashion	Non-Sensitive	Delete Ad Preference	More Info	ACTIVE
Universidad de Chile	Removed interests	Non-Sensitive		More Info	REMOVED
Televisions	Removed interests	Non-Sensitive		More Info	REMOVED
Real Madrid C.F.	Sports and outdoors	Non-Sensitive		More Info	INACTIVE
TripAdvisor	Business and industry	Non-Sensitive		Less Info	INACTIVE

HISTORICAL INFORMATION

This ad preference appeared in your profile in the following periods:

From 2016-09-16 to 2016-09-20. Reason: You have this preference because we think it may be relevant to you based on what you do on Facebook, such as pages you've liked or ads you've clicked.

potentially sensitive ad preferences and all currently active, respectively.

Related Work

We published a prior article² in which we already analyzed the use of sensitive information on FB. That article focuses on the European Union a few months before the GDPR was enacted. The research community asked us in various forums that it would be interesting to further extend our analysis to cover the use of sensitive information on FB worldwide and not just in the EU, and to understand the potential impact that the GDPR could have on reducing the exposure of users to sensitive ad preferences. This article covers both requests and, in addition, it adds two more contributions: We present two clear scenarios in which the use of sensitive ad preferences could have serious consequences for the users; and we introduce an improvement of the FDVT that allows users to remove in a simple way potentially sensitive ad preferences they do not like.

Few previous works in the literature address issues associated with sensitive personal data in online advertising, as well as some recent works that analyze privacy and discrimination issues related to FB advertising and ad preferences.

Carrascosa et al.⁴ propose a new methodology to quantify the portion of targeted ads received by Internet users while they browse the web. They create bots, referred to as *personas*, with very specific interest profiles (for example, persona interested in cars) and measure how many of the received ads match the specific interest of the analyzed persona. They create personas based on sensitive personal data (health) and demonstrate that they are also targeted with ads related to the sensitive information used to create the persona's profile.


Castellucia et al.⁵ show that an attacker that gets access (for example, through a public WiFi network) to the Google ads received by a user could create an interest profile that could reveal up to 58% of the actual interests of the user. The authors state that if some of the unveiled interests are sensitive, it could imply serious privacy risks for users.

Venkatadri et al.¹⁴ and Speicher et al.¹² exposed privacy and discrimination vulnerabilities related to FB advertising. In Venkatadri,¹⁴ the authors demonstrate how an attacker can use FB third-

party tracking JavaScript to retrieve personal data (for example, mobile phone numbers) associated with users visiting the attacker's website. Moreover, in Speicher,¹³ authors demonstrate that sensitive FB ad preferences can be used to apply negative discrimination in advertising campaigns (for example, excluding people based on their race). This work also shows that some ad preferences that initially may not seem sensitive could be used to discriminate in advertising campaigns (for example, excluding people interested in Blacknews.com that are potentially Black people).

Conclusion

Facebook offers advertisers the option to commercially exploit potentially sensitive information to perform tailored ad campaigns. This practice lays, in the best case, within a gray legal area according to the recently enforced GDPR. Our results reveal that 67% of FB users (22% of citizens) worldwide are labeled with some potentially sensitive ad preference. Interestingly, users in rich developed countries present a significantly higher exposure to be assigned sensitive ad preferences. Our work also reveals that the enforcement of the GDPR had a negligible impact on FB regarding the use of sensitive ad preferences within the EU. We believe it is urgent that stakeholders within the online advertising ecosystem (that is, advertisers, ad networks, publishers, policymakers, and so on) define an unambiguous list of personal data items that should not be used anymore to protect users from potential privacy risks as those described in this article.

Acknowledgments. The research leading to these results has received funding from: the European Union's Horizon 2020 innovation action programme under grant agreement No 786741 (SMOOTH project) and the grant agreement No 871370 (PIMCITY project); the Ministerio de Economía, Industria y Competitividad, Spain, and the European Social Fund (EU), under the Ramón y Cajal programme (Grant RyC-2015-17732), and the Project TEXEO (Grant TEC2016-80339-R); the Ministerio de Educación, Cultura y Deporte, Spain, through the FPU programme (Grant FPU16/05852); the Community of Madrid synergy project EMPATIA-CM (Grant Y2018/TCS-5046); and the Fundación BBVA under the project AERIS. 

References

1. Cabañas, J., Cuevas, A., and Cuevas, R. FDVT: Data valuation tool for Facebook users. In *Proceedings of the 2017 CHI Conf. Human Factors in Computing Systems*. ACM, New York, NY, USA, 3799–3809; <https://doi.org/10.1145/3025453.3025903>
2. Cabañas, J., Cuevas, A., and Cuevas, R. Unveiling and quantifying Facebook exploitation of sensitive personal data for advertising purposes. In *Proceedings of the 27th USENIX Security Symp. USENIX Assoc.*, Baltimore, MD, 2018, 479–495; <https://www.usenix.org/conference/usenixsecurity18/presentation/cabanass>
3. California State Legislature. California Consumer Privacy Act, 2018; <https://www.cprprivacy.org/>
4. Carrascosa, J., Mikians, J., Cuevas, R., Erramilli, V., and Laoutaris, N. I always feel like somebody's watching me: Measuring online behavioural advertising. In *Proceedings of the 11th ACM Conf. Emerging Networking Experiments and Technologies*. ACM, New York, NY, Article 13; <https://doi.org/10.1145/2716281.2836098>
5. Castellucia, C., Kaafar, M., and Tran, M. Betrayed by your ads!. In *Intern. Privacy Enhancing Technologies Symp.* Springer Berlin Heidelberg, 2012, 1–17.
6. European Union. Regulation 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC General Data Protection Regulation; <http://eur-lex.europa.eu/eli/reg/2016/679/oj>
7. Hong, J. The state of phishing attacks. *Commun. ACM* 55, 1 (Jan. 2012), 74–81; <https://doi.org/10.1145/2063176.2063197>
8. InternetSociety.org. The Internet Society survey on policy issues in Asia-Pacific 2016; <https://bit.ly/3ckqSpj>
9. Janrain.com. Consumer Attitudes Toward Data Privacy Survey 2018; <https://bit.ly/33JdOw0>
10. Narasimhamurthy, A. Theoretical bounds of majority voting performance for a binary classification problem. *IEEE Trans. Pattern Analysis and Machine Intelligence* 27, 12 (2005), 1988–1995.
11. PWC. Privacy in the Data Economy; <https://www.pwc.in/assets/pdfs/publications/2018/privacy-in-the-data-economy.pdf>
12. Speicher, T. et al. Potential for discrimination in online targeted advertising. In *Proceedings of the 1st Conf. Fairness, Accountability and Transparency*. S. A. Friedler and C. Wilson (Eds.). PMLR 81, 2018, New York, NY, USA, 5–19.
13. TNS Opinion and Social. Special Eurobarometer 431 Data Protection, 2015; http://ec.europa.eu/commfrontoffice/publicopinion/archives/ebs/ebs_431_en.pdf
14. Venkatadri, G. Privacy risks with Facebook's PII-based targeting: Auditing a data broker's advertising interface. In *Proceedings of the IEEE Symp. Security and Privacy*. (San Francisco, CA, USA, 2018) IEEE, 89–107.

José González Cabañas (jgcabana@it.uc3.es) is a Ph.D. candidate and FPU scholarship holder in the Department of Telematic Engineering at the Universidad Carlos III de Madrid, Spain.

Ángel Cuevas (acrumin@it.uc3m.es) is a Ramón y Cajal Fellow in the Department of Telematic Engineering at Universidad Carlos III de Madrid, Spain, and Fellow at UC3M-Santander Big Data Institute, Spain.

Aritz Arrate (aritz.arrate@alumnos.uc3m.es) is a Ph.D. candidate in the Department of Telematic Engineering at the Universidad Carlos III de Madrid, Spain.

Rubén Cuevas (rcuevas@it.uc3m.es) is an associate professor in the Department of Telematic Engineering at the Universidad Carlos III de Madrid, Spain and Deputy Director and Fellow at UC3M-Santander Big Data Institute, Spain.

Copyright held by authors/owners.
Publication rights licensed to ACM.



Watch the authors discuss this work in the exclusive *Communications* video.
<https://caom.acm.org/videos/does-facebook-use-sensitive-data>

DOI:10.1145/3377476

An analysis of patenting history from 1850 to 2010 to detect long-term patterns of knowledge spillovers via prior-art citations of patented inventions.

**BY PANTELIS KOUTROUMPIS, AIJA LEIPONEN,
AND LLEWELLYN D.W. THOMAS**

Digital Instruments as Invention Machines

THE HISTORY OF invention is a history of knowledge spillovers. There is persistent evidence of knowledge flowing from one firm, industry, sector or region to another, either by accident or by design, enabling other inventions to be developed.^{1,6,9,13} For example, Thomas Edison’s invention of the “electronic indicator” (US patent 307,031: 1884) spurred the development by John Fleming and Lee De Forest in early 20th century of early vacuum tubes which eventually enabled not just long-distance telecommunication but also early computers (for example, Guarnier¹⁰). Edison, in turn, learned from his contemporaries including Frederick Guthrie.¹¹ It appears that little of this mutual learning and knowledge exchange was paid for and can thus be

called a “spillover,” that is, an unintended flow of valuable knowledge, an example of a positive externality.

Information technologies have been a major source of knowledge spillovers.^a Information is a basic ingredient of invention, and technologies that facilitate the manipulation and communication of information should also facilitate invention. Indeed, Koutroumpis et al.¹⁷ found that information technology patents receive more citations than patents in other technology sectors. Similarly, Klevorick et al.¹⁶ found that advances in information technologies can generate broader technological development by enhancing technological opportunities in adjacent industries.

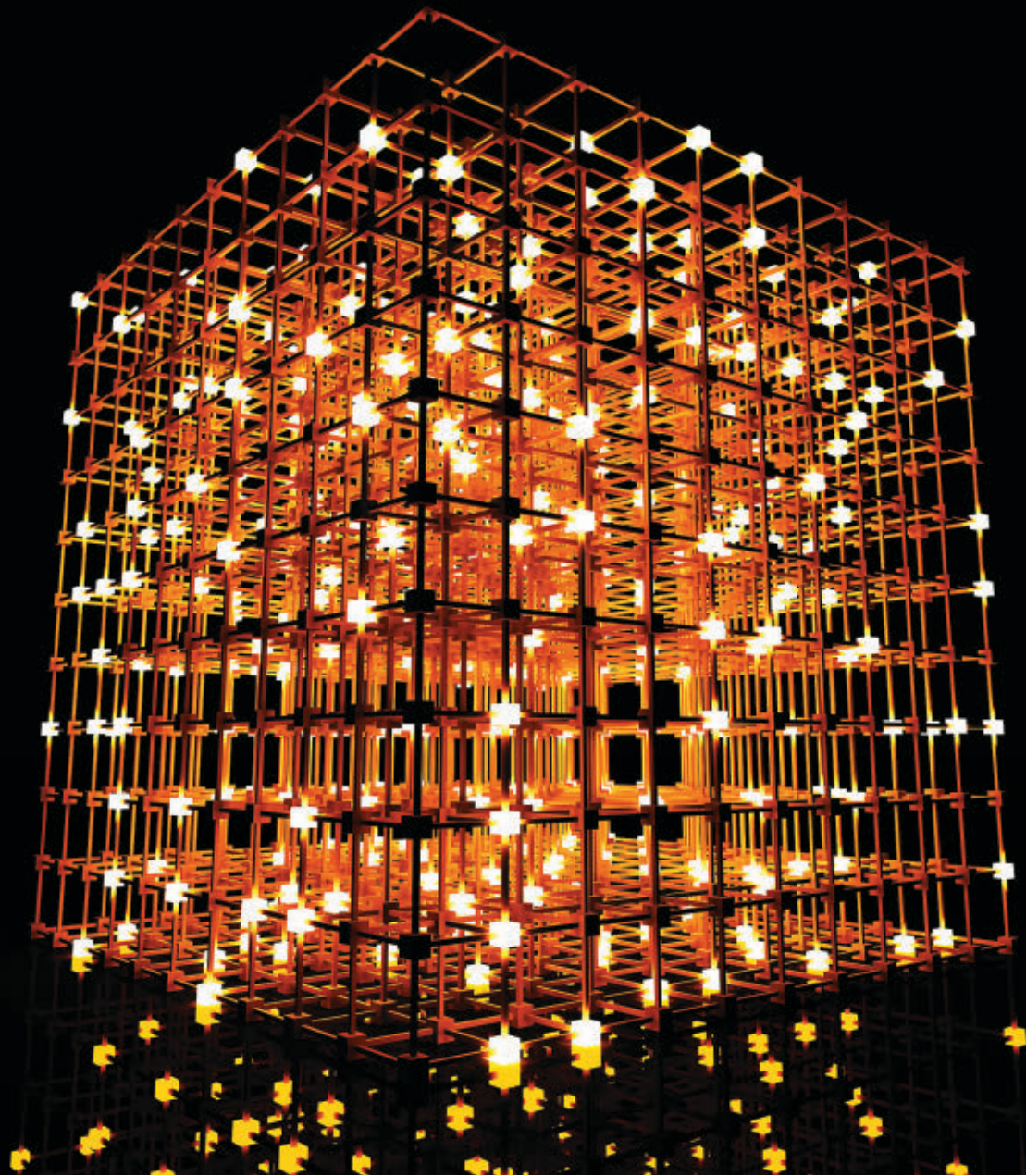
Another technology sector that has been theorized as generating outside spillovers are instrument technologies.^b Instrument technologies include measuring equipment ranging from scales, rulers, chronometers, thermometers, and accelerometers to more complex devices such as interferometers, spectrometers, oscilloscopes, medical transducers, and electron microscopes. Despite arguments by historians of science that instrument technologies enabled industrial revolutions (for instance, see Baird² and Price²¹), they have not been extensively studied. The studies that do exist are primarily qualitative descriptions of patterns of innovation rather than quantitative empirical analyses. For instance,

a Standard Industrial Classifications 357, 367.

b Standard Industrial Classification 38.

» key insights

- Knowledge spillovers from instrument technologies are exceptional and sustained over a long period of time.
- Knowledge spillovers from information technologies accelerate dramatically after 1970 and are closely related to instruments.
- Digital instruments—the intersection of information and instrument technologies—have been particularly generative in spurring invention in other sectors.



Rosenberg²² illustrated how improvements in the instruments used for measuring, testing and manipulating materials also affect the results of industrial R&D. Aware of the scarcity of empirical research on instrument technologies, Rosenberg called for “more research [that] may powerfully illuminate the course of scientific progress in the twentieth century ... [and that] ... the scope of such research must be international.”

In this article, we undertake such an empirical study. While there are many studies on knowledge spillovers from specific technologies (such as broadband, see for instance Becchettiet et al.³ and Bertschek et al.⁵), there are few that have considered broader technological classes and economy-wide spillover

effects. The study closest to ours is Hall and Trajtenberg¹² who conducted analyses of patent citations to identify technologies that can be characterized as general-purpose technologies because of their generality and association with rapidly evolving technology sectors. Although we also conduct patent-level analyses, we are not interested in identifying general-purpose technologies. Instead, we seek to evaluate the impact of instrument and information technologies as sources of invention by investigating *which technology sectors and fields are the most prolific sources of knowledge inputs into the invention activities of other sectors, and how these sources have changed over time*. Thus, we are interested in identifying the evolution of the sources of invention, and in

particular, the roles played by instrument and information technologies.

To do so, we examine how new technologies are disseminated and adopted across industries to characterize long-term shifts in the sources of technological change. We conduct a long-term descriptive analysis of cross-sectoral knowledge flows through analysis of patent citations. We take a big data approach and consider the entire technological progress of the world for more than the past century. We analyze the technology sectors of patented inventions and their prior art citations to describe the direction and volume of knowledge flows among between technology fields. This allows us to describe evolving relationships between technology fields that are difficult to discover

with a short or industry-specific sample. Our methodology allows us to present an accurate image of large-scale technological trends in the economy. While specific inventions are impossible to predict, this approach can be used to pinpoint emerging areas of exceptional R&D productivity and impact. As such, our study provides the first quantitative empirical analysis of instrument and information technologies as a key source of knowledge for other fields of invention.

Our results highlight, first, that spillovers from instrument technologies, as anticipated by historians of science, are exceptional and sustained over a long period of time. We also find the spillovers from information technologies accelerate dramatically after 1970 and that they are closely related to instruments. In fact, the intersection of information and instrument technologies, which we call “digital instruments,” has been particularly generative in spurring invention in other sectors.

Second, we conceptualize these exceptionally generative classes of technologies as “invention machines” due to their critical roles in the processes of invention in many sectors of the economy. We suggest that both information and instrument technologies should be considered as types of “Turing machines of invention.” Per Rosenberg, information technologies enable the manipulation of information, whereas instrument technologies enable the manipulation of physical matter (chemical substances, artifacts, physical processes, biological organisms). Thus, they are not only general-purpose technologies that can be utilized in many different sectors but also general invention technologies that facilitate the discovery of other technologies. Together, instrument and information technologies as digital instruments have been used to automate a wide range of industrial processes since early 1970s. They constitute an essential combination whose fundamental nature in technological change has thus far gone unnoticed.

Finally, we suggest that digital instruments constitute the core of industrial systems and other “smart” systems, often known as the Internet of Things (IoT). The evolution of these technologies began decades ago and reflects a convergence of industrial in-

strumentation with digital communication technologies. While the processing and transmission capacity of such systems has multiplied in recent years, the core ideas and technologies have existed for a long time. We suggest the continuing coevolution of instrument and information technologies will generate very powerful invention machines for the coming decades, spurring a potential technological flourishing in the adopting sectors. We now describe our methodology to analyze the patent database and present our empirical results primarily through visualizations.

Method

Given the difficulty in measuring knowledge flows between firms, patent citations have long been considered proxies for the flow of knowledge from the inventors whose patents are cited to the inventors making the citations. Much economic research has attempted to measure and assess the implications of such spillovers by analyzing citations made in patent documents to predecessor inventions. To verify this measurement strategy, Jaffe et al.¹⁴ surveyed the meaning of patent citations and concluded that a substantial part (but by no means all) of such citations involve actual flows of knowledge. Thus, patent citations are a noisy but meaningful indicator of knowledge spillovers in an economy. However, patent citations must be used carefully, as citations can be added not only by the inventors, but also by the patent attorneys and the patent examiners involved with the patent application, with the final decision ultimately lying with the patent examiner.⁸ That said, patent data remains a valuable, even if imperfect, tool with which to measure how new technological knowledge is disseminated in the economy.

Our data source is PatStat, a comprehensive resource from the European Patent Office covering more than 170 publication authorities (patent offices), 88 million awarded patents, 160 million citations, and more than 200 control variables covering the period from 1850 to 2018. In this article, we consider the four primary PatStat technology sectors (mechanical engineering, chemistry, electrical engineering, and instruments), resulting in 32 fields (details are provided online, see Appendix 1 at <https://dl.acm.org/>

doi/10.1145/3377476).^{cd} Our analysis is based on a simple count-data model of the number of citations received by each patent, controlling for several confounding factors that may influence our estimates. The basic model is the following:

$$C_i = \beta_{kt} F_{kt} + \gamma_i X_i + \varepsilon_i \quad (1)$$

C_i is the sum of all citations received by patent i , F_{kt} is a binary variable equal to 1 for patents that belong to field k and were published in year t , and 0 otherwise. This model reports estimators at the field-year level conditional on a set of controls. These controls are included in X_i , the vector of patent characteristics, and ε_i is the error term. β_{kt} captures the number of citations received by each field and year, all other things being equal. Our analysis is done at the patent level allowing for maximum degree of flexibility in the estimates.

Given our analytic interest in instrument and electrical engineering technologies, we exploit the fact that patents can be classified to multiple patent sectors. Building upon the growing impact of electrical engineering patents after the 1970s (see Figures 1 and 2 and Koutroumpis et al.¹⁷), we use a differences-in-differences model to measure the change in spillovers to other sectors that originate from instruments before and after 1970. We construct S_{ik} as the sum of citations that originate from sectors j other than k ($k \neq j$), with $S_{ik} = \sum_{k \neq j} C_{ik} < C_i$, where C_i is the sum of all citations received by patent i . The simple model from Eq. 1 now becomes:

$$S_{ik} = \alpha_{kt} \text{Post}_{\text{year}=1970} + \beta_{kt} F_{\text{Sector},t} * F_{\text{Instruments},t} + \delta_{kt} \text{Post}_{\text{year}=1970} * F_{\text{Sector},t} * F_{\text{Instruments},t} + \gamma_i X_i + \varepsilon_i \quad (2)$$

c We do not present the results for the “Other Fields” technology sectors as there are only limited data.

d Occasionally patent classification schemes are modified, and patents can change their classification. For our analysis we use the most recent classifications. We do not believe that past reclassifications will influence our analysis, as most reclassifications happen at quite granular (3 or 4 digit) levels, and our analysis is at the rather coarse sectoral and field levels. Put differently, it is unlikely for a patent to be reclassified between technology sectors. We thank Paola Criscuolo for pointing this out to us.

where $F_{Sector,t}$ takes the values Sector = [electrical, mechanical, chemical], $F_{Sector,t} * F_{Instruments,t}$ is a binary variable equal to 1 for patents that list technology sectors in electrical, mechanical or chemical and instruments sectors published in year t , and 0 otherwise. The $Post_{year=1970}$ binary variable is 1 for years after 1970 and 0 otherwise. The interaction of $Post_{year=1970}$ by the selected $F_{Sector,t} * F_{Instruments,t}$ measures the post 1970 effect in the treated groups. We report estimators at the field-year level using the same set of controls included in X_i , the vector of patent characteristics including tech sector fixed effects. ε_i is the error term and β_{kt} reflects the number of citations received by each field and year, all other things being equal.

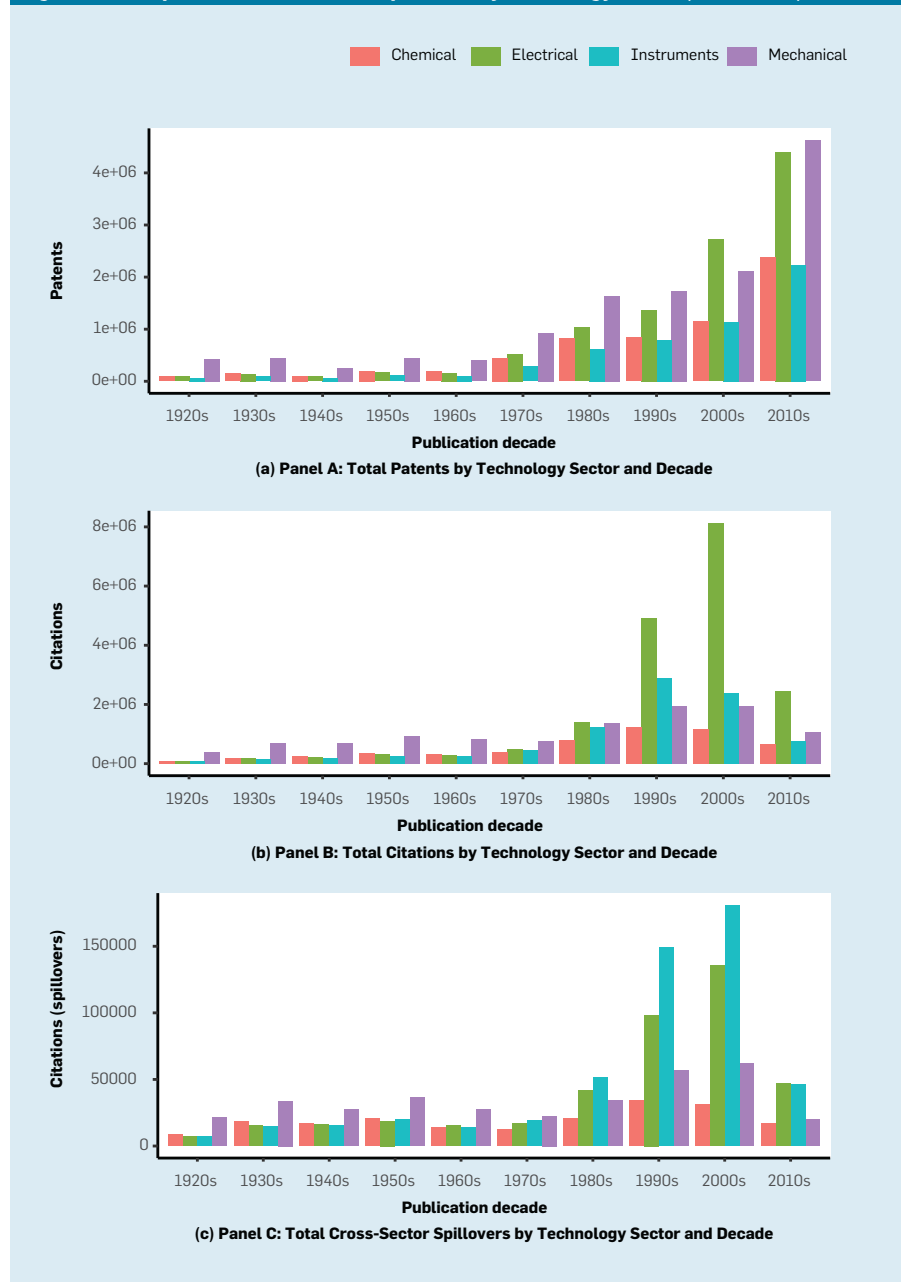
There are several factors that may influence patent citation counts. First, the number of citations is linked to the procedures followed by publication authorities (national or regional patent offices) that oversee the application and grant process. This can change over time as new processes within the patent offices affect the ways they attribute citations. Second, prior art citations have been rising in recent years thus introducing a secular trend. We therefore control for year and patent office effects allowing direct comparisons across jurisdictions and over time. Third, a patent may belong to a family of inventions that are submitted to multiple patent offices. The size of such a patent family can affect the visibility of the invention and hence increase the likelihood of the patent being cited. We compute and control for the numbers of patents that belong to each family and each “extended” family.^e Fourth, technology sectors have varying publication and citation patterns. We control for the total number of inventions granted (annual patent flows) and the total number of citations within each patent class each year. These metrics correct for potentially inflated citation counts in sectors with more inventions (hence with a higher likelihood of being cited) and sectors that cite

patents and non-patent literature more extensively than others. Further, we control for the citations made by patent examiners and the number of claims to capture the extent and scope for protection sought. Lastly, we capture seasonal effects with controls for the month of publication.

All these controls reassure us about the validity of comparisons over time, across patent offices, and across technology fields. Our assumption is that patents submitted in a patent office, at the same time, within the same field, and with the same family size will be treated equally by the authorities. Given

that recently published patents have a shorter window of observations, to avoid a systematic bias, we only consider results with a varying cut-off year ranging from 2005 to 2018 in our analysis. (In the online appendix, Figure 1 presents the average number of citations received by all patents in the years after publication; <https://dl.acm.org/doi/10.1145/3377476>). We also only focus on priority patents (the patent with the first application filing date) and attribute all citations for subsequent applications (across patent offices) to them. This latter choice along with the increasing patenting activity in recent

Figure 1. Total patents, citations, and spillovers by technology sector (1920–2010).



^e This broader definition of a patent family takes domestic application numbers as additional connecting elements and includes patents having the same scope but lacking a common priority (www.epo.org).

years—for which we explicitly control—reduces our sample to approximately 54 million (from 88 million in total).^f

Results

Relative influence. To understand the overall long-term pattern of knowledge spillovers via prior-art citations of patented inventions, we first look at the spillovers from all technology sectors (but not within technology sector) across 10 decades, from the 1920s to the 2010s.^g We present these results in Figure 1.

Panel A in Figure 1 presents the total patents by technology sector and illustrates the growth of patents over the past century. It clearly shows, with the exception of the 2000s, that the most common patents awarded are

mechanical engineering technologies, ranging from 422,504 in the 1920s to 4,623,680 in the 2010s. Panel B in the figure presents the total citations by technology sector and illustrates that mechanical engineering technologies were the most cited patents up until the 1970s, when electrical engineering technologies became dominant. Citations of electrical engineering patents rose from 501,283 the 1970s to peak at 8,138,699 in the 2000s. Panel C in the figure presents the total cross-sector spillovers (that is, excluding the same sector citations) by technology sector and illustrates that mechanical engineering technologies were the main source of sectoral spillovers up until the 1970s, when instrument technologies (rising from 5,365 to 24,674 in the 2010s) and electrical engineering technologies (rising from 10,651 to 28,856 in the 2000s), became the main source of

cross-sector spillovers. Put differently, Figure 1 highlights a shift from mechanical and chemical technologies to electrical and instrument technologies as the main source of cross-sector spillovers since the 1970s.

We now look at the relative influence of the four primary technology sectors, namely instruments, electrical engineering, mechanical engineering, and chemistry (Figure 2). Although the four technology sectors display distinct citation profiles, the pattern changes for all of them in the early 1970s.^h As anticipated, we observe that instruments are the most widely cited sector both within and

f Full tables are available from L. Thomas.

g The figures for the 2010s are up until the end of 2018.

h In 1970, many publication offices were added in the dataset (including the Japanese JPO and the European EPO). The inclusion of these patent offices increased the nominal number of patents but not necessarily their citation counts. We do not believe this change influences our results.

Figure 2. Patent citations by sector, field, and year.

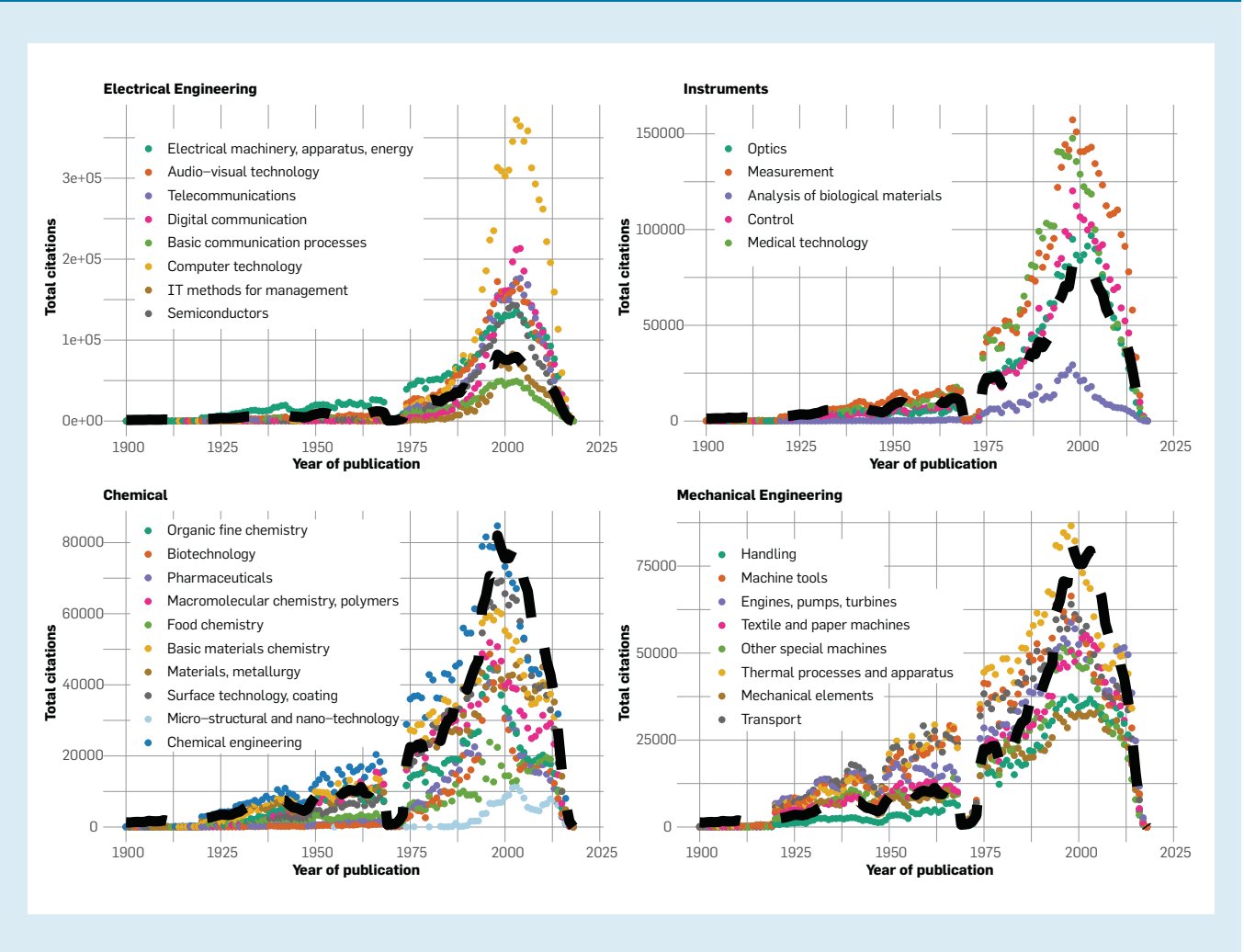
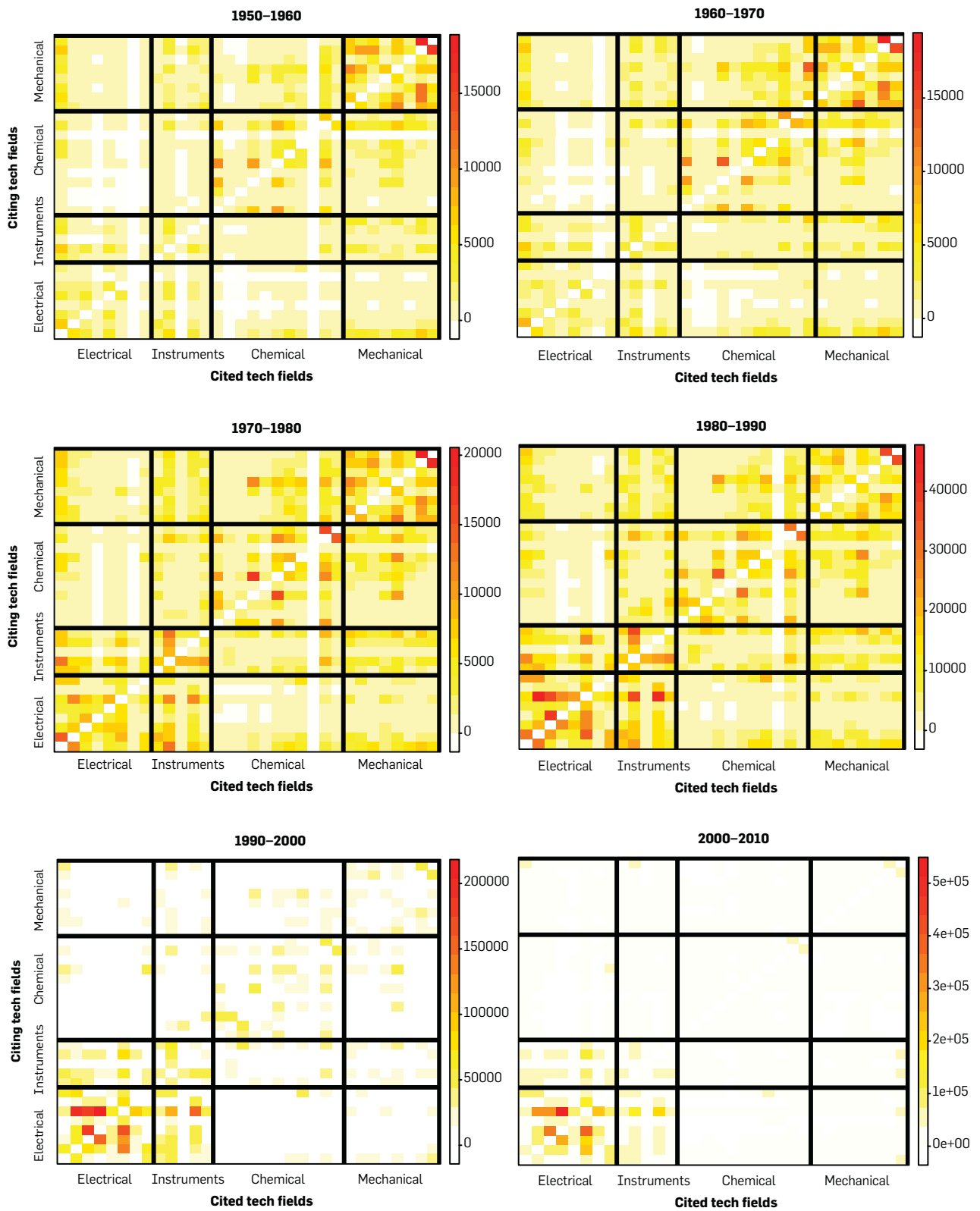


Figure 3. Spillovers for all patent technology fields by decade, 1950–2010.



across other industrial activities, exhibiting the highest mean numbers of citations per patent for the entire period. The most striking changes within instrument technologies appear to both measurement and medical instruments, which consistently receive

more citations than the mean of all sectors for the entire period of study. Also of interest are control instruments that become more influential after the 1970s. Control instruments typically relate to the manipulation and management of machinery.

For electrical engineering technologies, we observe that prior to 1970 this technology sector follows the mean, after which it begins to increase more rapidly, peaking just after the 2000s. Within electrical engineering, most technology fields are above the mean post 1970, except for basic communication processes. Of note are the fields of computer technology, digital communication, audiovisual technology and telecommunications, which exhibit significant spillovers. Considering the other technology sectors, while mechanical engineering was consistently above the trend prior to 1970, and chemistry technology sectors closely followed the general trend, soon after 1970 their influence starts to decline to below trend.

Figure 3 presents the spillovers for all patent technologies from the 1950s to the 2000s.ⁱ This shows the direction and size of the relative cross-sectoral citations. As the number of patents increases significantly with time, so do the level of spillovers across decades. For example, the maximum spillovers from a technology field range from 17,642 in the 1950s to 513,275 in the 2000s. We clearly show over this time period a shift from mechanical and chemistry technologies as the main source of cross-sector spillovers to electrical and instrument technologies, with the effect beginning in the 1970s.

Taken together, the patterns presented in Figures 1–3 correspond to a shift from mechanical and chemical technologies to electrical technolo-

ⁱ Results prior to 1950 are similar to those of the 1950s; for conciseness we do not report them here. They are available from the corresponding author upon request.

Table 1. Spillovers for instruments cross-referenced with other patent sectors, before and after 1970.

	(1)	(2)	(3)
Estimation method	OLS	OLS	OLS
Dependent variable	X-sector Spillovers	X-sector Spillovers	X-sector Spillovers
Post _t (dummy=1 after 1970)	0.096** (143.23)	0.061** (139.25)	0.133** (140.58)
Digital Instruments	1.32** (432.28)		
Digital Instruments X Post	0.133** (123.53)		
Mechanical Instruments		1.394** (846.08)	
Mechanical Instruments X Post		-0.034** (19.51)	
Chemical Instruments			1.632** (505.94)
Chemical Instruments X Post			-0.08** (23.82)
Observations	53,980,888	53,980,888	53,980,888
R ²	0.14	0.12	0.22
Year FE	yes	yes	yes
Publication Authority	yes	yes	yes
Stock of published patents by field and year	yes	yes	yes
Family Size	yes	yes	yes
Family Size Broad	yes	yes	yes
Publication Claims	yes	yes	yes
Citations (#) by examiners	yes	yes	yes
Stock of citations by field and year	yes	yes	yes

Notes: The dependent variable is the number of citations from all other sectors excluding Digital Instruments (column 1; specifically Electrical Engineering and Instruments), Mechanical Instruments (column 2), Chemistry Instruments (column 3) and Other Instruments (column 4). Standard errors clustered at the patent family level are reported in parenthesis below coefficients: *significant at 5%; **significant at 1%.
Source: Authors' calculations based on data from PATSTAT.

Table 2. Spillovers from instrument technology fields (sub-sectors) interacted with other patent sectors after 1970.

Instrument Fields:	(1)	(2)	(3)	(4)	(5)
	Optics	Measurement	Analysis of biological materials	Control	Medical technology
Chemical	0.306** (68.63)	0.066** (25.53)	0.035** (68.24)	-0.045** (104.34)	0.838** (55.95)
Electrical Engineering	0.242** (65.90)	0.212** (98.74)	-0.034** (81.08)	0.040** (99.21)	-0.174** (14.08)
Mechanical Engineering	0.080** (20.17)	0.035** (15.24)	-0.028** (61.59)	-0.021** (56.63)	1.012** (76.42)

Notes: The dependent variable is the number of citations from the Instrument patent field. Standard errors clustered at the patent family level are reported in parenthesis below coefficients: *significant at 5%; **significant at 1%. **Source:** Authors' calculations based on data from PATSTAT.

gies. They are also aligned with findings by Jovanovic and Rousseau¹⁵ and Koutroumpis et al.¹⁷ regarding the ICT revolution commencing about this time.

Cross-sector and cross-field spillover effects. Table 1 presents our regression analyses of the technology sectors that instrument patents are co-listed with. We define as “digital instruments” those patents that list both electrical engineering and instrument sectors, and “mechanical instruments” those patents that list both mechanical engineering and instrument sectors. Similarly, “chemical instruments” list both chemistry and instrument sectors.

We find that instrument technologies appear to have generated substantial and sustained knowledge spillovers over several decades regardless of the underlying technological base. Overall, chemical instruments (coefficient 1.632) are the most influential, followed by mechanical instruments (1.394) and digital instruments (1.320). When we consider citation patterns after 1970 (X Post), more specifically considering the ICT revolution, we find that spillovers from digital instruments increase by 0.388 citations after 1970 (column 1), while those from mechanical instruments (-0.034) and chemical instruments (-0.080) drop after 1970 (columns 2 and 3). This suggests that instrument technolo-

gies have become increasingly digitized post 1970.

Given the important inflection point of 1970 in the transition to digital technology-based instruments, we now focus on post 1970 spillovers and investigate how each instrument field interacts with the main sectors, namely, electrical engineering, chemistry, and mechanical engineering technologies. Table 2 presents these coefficients.

Here we again see the confluence of electrical engineering with instrument technologies, with electrical engineering strongly co-listed with optical (0.242) and measurement (0.212) instruments (columns 1 and 2). These results are suggestive of digital sensor technologies. However, we also find positive effects for optical instruments (column 1) co-listed in the chemistry sector (0.306), suggestive of laser technologies, and for medical technology instruments (column 5) co-listed in the chemical (0.838) and mechanical engineering (1.012) sectors.^j

Given the importance of electrical engineering to instruments post-1970 (Table 2), we carry out a similar analysis of the fields of the electrical engineering sector. Table 3 reports the in-

^j These are normalized citation counts, when the dummy becomes one the coefficient reflects that additional citations a patent with these characteristics would receive (i.e., the implied influence).

teractions of the electrical engineering fields with the instrument fields post 1970. Of note is the strong interaction of optical instruments (column 1) with all electrical engineering fields except for basic and digital communication. Table 3 also shows strong interaction of measurement instruments (column 2) with semiconductors (0.246), electrical machinery (0.245), computer technology (0.129), and basic communication (0.113). Both are highly suggestive of the technologies that comprise the digital sensors that typify industrial connected devices. There is also evidence of the digitization of medical instruments (column 5) with the strong interaction of electrical machinery (1.368) and computer technology (0.512), and the digitization of control systems (column 4), with a strong interaction effects with IT management methods (0.443) and computer technology (0.157).

Conclusion and Limitations

To investigate the hypothesized spillover effects of instrument and information technologies,^{2,16,22} we analyzed the entire global history of patenting from 1850 to 2018 to detect long-term patterns of knowledge spillovers via prior-art citations of patented inventions. We found that information and instrument technologies generate the most substantial and widespread spillovers of knowledge used in other technology fields. For this reason, we

Table 3. Spillovers for instrument technology fields interacted with electrical engineering technology fields after 1970.

	(1)	(2)	(3)	(4)	(5)
Instrument Fields:	Optics	Measurement	Analysis of biological materials	Control	Medical technology
Electrical machinery, apparatus, energy	0.464** (41.54)	0.245** (37.82)	0.011** (8.54)	0.046** (53.05)	1.368** (36.58)
Audiovisual technology	0.597** (53.03)	-0.123** (18.82)	-0.005** (4.11)	0.023** (22.89)	-0.231** (6.14)
Telecommunications	0.130** (11.45)	0.029** (4.46)	-0.007** (5.34)	0.074** (66.61)	-0.189** (4.99)
Digital communication	-0.109** (9.14)	-0.164** (23.79)	-0.007** (4.90)	0.056** (43.37)	-0.381** (9.58)
Basic communication processes	-0.037** (2.66)	0.113** (14.07)	-0.011** (7.21)	0.001** (0.37)	-0.435** (9.38)
Computer technology	0.160** (14.63)	0.129** (20.32)	0.006** (4.75)	0.157** (155.50)	0.512** (13.96)
IT methods for management	0.154** (10.69)	-0.144** (17.34)	-0.005** (3.21)	0.443** (201.90)	-0.411** (8.56)
Semiconductors	0.732** (57.59)	0.246** (33.50)	0.013** (9.02)	-0.030** (25.05)	-0.311** (7.33)

Notes: The dependent variable is the number of citations from Instrument patent field. Standard errors clustered at the patent family level are reported in parenthesis below coefficients: *significant at 5%; **significant at 1%. **Source:** Authors' calculations based on data from PATSTAT.

call them “invention machines.” The greatest spillover impact is generated by digitized optical and measurement instruments.

Digital instruments form the technological base of industrial connected devices, such as smart buildings (including “smart” lighting, heating, ventilation, and air conditioning and physical security systems), process sensors for manufacturing, and real-time location and sensing devices for healthcare. Digital instruments also comprise the sensors a standard smartphone has—accelerometer, gyroscope, magnetometer, GPS, barometer, proximity sensors, and ambient light sensors—without which many of the functions of the phone cannot work.^k There were approximately 8.4 billion of these types of connected devices in 2017,^l and by 2020, it is estimated there will be between 30 to 50 billion connected devices.¹⁹

Thus, we argue the convergence of digital technologies and instrument technologies is likely to bring about the next generation of invention machines. Advanced digital communications make it possible to simultaneously and immediately utilize information in a wide variety of contexts. As such, digital instruments will allow the observation and manipulation physical, chemical, biological, and social processes in connected industrial activities in a vast set of contexts. One can view these technologies as key enablers for the “Second Machine Age” vision of the future of Brynjolfsson and McAfee.⁷ Therefore, we suggest the emergence and adoption of the digital instruments in coming years is likely to generate a flurry of invention in many if not most technology fields.

The increasing reach of digital instruments and knowledge spillovers will potentially speed up the rate of invention both within industries and within firms. As the onslaught of networked automation may continue to create industrial value but also societal upheaval via creative destruction of jobs, occupations, and organizations,

k Gizmodo, 23/07/17: <http://fieldguide.gizmodo.com/all-the-sensors-in-your-smartphone-and-howthey-work-1797121002>; retrieved 26/10/17.
l Gartner, 17/02/17: <https://www.gartner.com/newsroom/id/3598917>; retrieved 26/10/17.

it is interesting to note that the set of technologies that fundamentally enables this, instruments, has gone relatively unnoticed in the economics and management of technology.

Despite the volume of data analyzed, this study is subject to several limitations. First, patents represent only a subset of technological knowledge that may also appear in other forms and channels including non-patent literature, tacit or organizational knowledge, open innovation, and software. Capturing these links would strengthen the interpretation of spillovers and could also help explain or even predict the launch of new inventions and technologies. We consider this along with a parallel research of academic and open source software a promising area of future research.

Second, we can identify influential patents ex-post but have not created a robust method that predicts the new patents that will appear. One approach for this could combine the spectrum of possible applications of a patent using our baseline specification and machine-learning techniques to analyze a very granular dataset.

Third, there is a long literature linking the inventor capacity, networks, legislation and resource allocation to the subsequent success of inventions.^{4,16,20,23} Our work has not utilized this type of information to predict the commercial success of various inventions, and we believe some fruitful research can be undertaken here.

Acknowledgments. This work was supported by EPSRC project EP/K039504/1 and the EIT ICT Labs. We thank Paul Romer, Shane Greenstein, Eric Brynjolfsson, Chris Forman, Kristina McElheran, Frank Nagle, Avi Goldfarb, Achim Luhn, the NBER Digitization workshop participants (2016) and the Academy of Management Big Data Symposium (2016) participants for their comments. ■

References

1. Aharonson, B.S., Baum, J.A.C., and Feldman, M.P. Desperately seeking spillovers? Increasing returns, industrial organization and the location of new entrants in geographic and technological space. *Industrial and Corporate Change* 16, 1 (2007), 89–130.
2. Baird, D. Analytical chemistry and the ‘big’ scientific instrumentation revolution. *Annals of Science* 50, 3 (1993), 267–290.
3. Becchetti, L., Bedoya, D.A.L., and Paganetto, L. ICT investment, productivity and efficiency: Evidence at firm level using a stochastic frontier approach. *J. Productivity Analysis* 20, (2003), 143–167.

4. Becker, B., and Gerhart, B. The impact of human resource management on organizational performance: Progress and prospects. *Academy of Management J.* 39, 4 (1996), 779–801.
5. Bertschek, I., Cerquera, D., and Klein, G. J. 2013. More bits—more bucks? Measuring the impact of broadband internet on firm performance. *Information Econ. and Policy* 25, 3 (2013), 190–203.
6. Breschi, S., and Lissoni, F. Knowledge spillovers and local innovation systems: A critical survey. *Industrial and Corporate Change* 10, 4 (2001), 975–1005.
7. Brynjolfsson, E., and McAfee, A. *The Second Machine Age—Work, Progress, and Prosperity in a Time of Brilliant Technologies*. W W Norton and Co., New York, NY 2014.
8. Criscuolo, P., and Verspagen, B. Does it matter where patent citations come from? Inventor vs examiner citations in European patents. *Research Policy* 37, 10 (2008), 1892–1908.
9. Griliches, Z. Issues in assessing the contribution of research and development to productivity growth. *The Bell J. of Econ.* 10, 1 (1979), 92–116.
10. Guarnieri, M. The age of vacuum tubes: Early devices and the rise of radio communications. *Industrial Electronics* 9, 1 (2012), 41–43.
11. Guthrie, F. *Magnetism and Electricity*. William Collins Sons and Co., London, U.K., 1876.
12. Hall, B.H., and Trajtenberg, M. Uncovering GPTS with patent data. Working Paper, National Bureau of Economic Research, 2004.
13. Jaffe, A.B. Characterizing the ‘technological position’ of firms, with application to quantifying technological opportunity and research spillovers. *Research Policy* 18, 2 (1989), 87–97.
14. Jaffe, A.B., Trajtenberg, M., and Fogarty, M.S. The meaning of patent citations: Report on the NBER/Case-Western Reserve Survey of Patentees. Working Paper, National Bureau of Economic Research, 2000.
15. Jovanovic, B., and Rousseau, P.L. *General Purpose Technologies. Handbook of Economic Growth Vol. 1 Part B*. P. Aghion and S.N. Durlauf, (Eds.). Elsevier BV, Amsterdam, NL, 2005, 1181–1224.
16. Klevorick, A.K., Levin, R.C., Nelson, R.R., and Winter, S.G. On the sources and significance of interindustry differences in technological opportunities. *Research Policy* 24, 2 (1995), 185–205.
17. Koutroumpis, P., Leiponen, A., and Thomas, L.D.W. How important is ‘it’? *Commun. ACM* 60, 7 (July 2017), 62–68.
18. Moser, P. How do patent laws influence innovation? Evidence from nineteenth-century world’s fairs. *The American Econ. Rev.* 95, 4 (2005), 1214–1236.
19. Nordrum, A. Popular internet of things forecast of 50 billion devices by 2020 is outdated. *IEEE Spectrum* (2016), 18.
20. Paruchuri, S. Intraorganizational networks, interorganizational networks, and the impact of central inventors: A longitudinal study of pharmaceutical firms. *Organization Science* 21, 1 (2010), 63–80.
21. Price, D. The science/technology relationship, the craft of experimental science, and policy for the improvement of high technology innovation. *Research Policy* 13, 1 (1984), 3–20.
22. Rosenberg, N. Scientific instrumentation and university research. *Research Policy* 21, 4 (1992), 381–390.
23. Subramaniam, M., and Youndt, M.A. The influence of intellectual capital on the types of innovative capabilities. *Academy of Management J.* 48, 3 (2005), 450–463.

Pantelis Koutroumpis is the Lead Economist of the Programme of Technological and Economic Change in the Oxford Martin School at the University of Oxford, U.K.

Aija Leiponen is a professor and program director of the Master in Professional Studies in the Dyson School of Applied Economics and Management, SC Johnson College of Business at Cornell University, Ithaca, NY, USA.

Llewellyn D.W. Thomas is an associate professor at LaSalle Universitat Ramon Llull in Barcelona, Spain and a visiting professor at Imperial College Business School in London, U.K.

A field study examines technological advances that have created versatile software ecosystems to develop and deploy microservices.

BY KAROLY BOZAN, KALLE LYYTINEN, AND GREGORY M. ROSE

How to Transition Incrementally to Microservice Architecture

REVENUE CYCLE MANAGEMENT (RCM) is a complicated process that involves several steps and considerable data flow. A software development organization (SDO) that was building an RCM application quickly realized the complexity of this task. Specifically, the cyclomatic complexity of the application was in the

thousands. The SDO could not easily scale the application or add features without having an impact on the entire code base.

In the early 2000s, no clear means was available to overcome this challenge until, per the suggestion of a consultant, the SDO began to discuss how to simplify the system by isolating the RCM steps into smaller, independent services. Later, this idea became known as microservice architecture (MSA), which has recently been touted as a promising software architecture alternative. Generally, an architecture style denotes a plausible and reusable pattern of solutions, backed by experience, to a set of known programming problems.¹⁰ The architecture conveys a highly abstracted conceptual model of structure and behavior of the software, given its design goals and constraints. The choice of the architecture has an impact on the ways the software will be implemented and how its development can be organized.

A judiciously chosen architecture style helps reduce technical debt and enhances software efficiency and quality.⁹

Generally, a MSA solution is founded on the idea of “orchestrating” the software, comprising loosely coupled and independent “services.”^{13,18,22} In an MSA solution, each service is responsible for a dedicated, well-defined, and

» key insights

- **MSA offers a flexible, speedy, scalable software development paradigm founded on loosely coupled, cohesive, reusable, and easy to replace software services.**
- **Transitioning to MSA challenges software development organizations as they migrate their legacy code built around monolithic architectures.**
- **A roadmap is provided to adapt legacy code to MSA across four process phases where decision makers, mechanisms, and outcomes for each phase differ along with different benefits, risks, and organizational impacts.**

scalable function, which may and often is expected to serve, at the same time, other applications. Credit card processing, product rating, and checkout business functions are examples of microservices in e-business. Each service is developed, tested, and deployed independently without making strict assumptions as to how the deployment of the service will affect its use as part of other applications. Under MSA, some services are developed internally, while others are developed by third parties and linked to the final application through APIs through service orchestration. From a historical perspective, MSAs are a testament to the software community's aspiration to develop and manage highly modular, well-organized software assets.³

In the past, software has been built mainly around a tightly coupled codebase called a monolith architecture. This style mainly organizes software as a modular single tier or, later, as a horizontally isolated n -tier architecture (Internet stack). Software built using a monolith architecture faces significant challenges to its feature growth, scalability, or performance when the software continues to evolve, often under tight time pressure.² As new requirements emerge or innovative technologies are adopted, the “legacy” code often fails to support a rapid implementation of such changes.

Recent technological advances have created versatile software ecosystems to develop and deploy microservices. For example, Docker, a container platform, provide a means to operate system-level virtualization to package software in lightweight “containers” orchestrated by Kubernetes.⁵ A growing number of software service startups now offer support services for container deployment, management, and security. This also allows small SDOs to increasingly deploy software assets previously available for large SDOs, such as Amazon.

Moving to MSA, however, is neither easy nor risk free. It calls for a strategic, disciplined approach that avoids the disruption of current operations and user experience with software. Many SDOs remain uncertain of the benefits of the transition and remain on the sidelines.¹⁵ The published success stories of transition offer often unfounded and far too positive claims and give the appearance

that a decoupling will automatically lead to scalability, flexibility, and decreased time to market. The most publicly known successful examples come from large and born-digital operations, such as Amazon, whereas most SDOs that consider the transition lack the resources, skills, and project management capabilities of such software giants. Therefore, companies with legacy systems need to carefully consider alternatives to accomplish the transition.

Per our field study, most SDOs with significant software assets select an incremental strategy to decrease transition risks and ensure a smooth transition. Under the incremental strategy, the monolith legacy software and the new, modular MSA-based software will be simultaneously present for a significant period. This calls for a deeper appreciation of their joint impact on software assets, people, the organization, and managerial practices.⁸ SDO management must prepare for a wide array of changes at multiple levels of the organization while the transition unfolds and to assess the impact of the benefits and inevitable risks. We next identify some of those changes and related challenges identified in a field study that focused on leading industry practices during a successful MSA transition.

Study Design

The field study data was collected between February 2018 and June 2019. The study relied on semi-structured interviews that focused on changes, challenges, and opportunities during the architectural transition and the impacts of the transition on SDO's management of software assets and their software process and organization. The data was collected from nine SDOs that had experienced an incremental transition in different forms and stages. The SDOs were of varying sizes and operated in six industries. The study included 23 interviews with 31 software experts, whose titles included CTO, Global Director, Senior/Principal Architects, Application Architects, and Lead Developers. The data collection progressed through three iterative rounds. In the first round, we sought to understand the impact of emerging technologies on software development. During this round, informants unanimously identified MSA as a critical architectural trend. In

the second round, we sought to identify the motivations and decision logics that guided the selection of architectural styles and transitions. During the third round, we sought to identify and understand pivotal challenges in operating co-existing monoliths and MSA architectures and solutions. We also validated our study findings and conclusions among a subset of informants. The transition strategies and challenges discussed here were inductively derived from the transcripts. We coded data simultaneously with the data collection to ensure higher validity and reliability of emerging themes, guide follow-up interviews, and identify saturation in data and code. Further details of data collection and analysis as well as examples of coding trees are reported in Appendix A available online at <https://dl.acm.org/doi/10.1145/3378064>.

Incremental Transition

During incremental architectural transition, microservices are introduced in a piecemeal fashion while software assets are iteratively and successively re-architected. Table 1 provides a summary of the benefits, risks, and organizational impacts of the incremental transition to MSA, as derived from our field study. The benefits and drawbacks of the two most common alternative strategies—big bang and zero alternative, no transition—are reported in Appendix B available online at <https://dl.acm.org/doi/10.1145/3378064>. In the incremental strategy, the choice of applications to be re-architected and the way microservices are spun out from monolith code base are guided primarily by the determination of whether the re-architecting provides recognizable value to the business and immediate benefits to the application users. Business needs drive which services will be isolated and developed independently. Under the incremental strategy, the final goal is not necessarily to decouple the system fully unless it comes with clearly defined and measurable advantages. Overall, the goal is to continually balance transition risks²³ with expected benefits.^{1,7,12}

The four phases of incremental transition. Typically, an incremental transition process unfolds through four phases: identify goals for transition; identify the scope and level of architectural changes; prepare for

resource readiness; and change critical development practices. The phases are distinguished by:

- ▶ Different *decision makers* who initiate and are responsible for the types of changes related to phase;
- ▶ Different *mechanisms* that allow decision makers to carry out the transition in that phase; and
- ▶ Different *outcomes* for each phase.

The mechanisms are defined organizational capabilities that are used to make related decisions or to implement these decisions. SDOs will deploy such mechanisms when making and executing decisions about the transition directions while developing their software assets. The accompanying figure provides a summary of the principles that separate the phases and the logical dependencies between them. The top-down arrows in the figure illustrate the increasingly granular focus of the incremental transition as it moves across the phases. Initially, the leadership must put mechanisms in place that help to identify the gaps between an SDO's strategic goals and the extent to which the current software development practices align with them. These mechanisms help to identify the limitations that must be overcome and the opportunities that need to be capitalized on during the MSA transition. The strategies in an incremental transition that help to achieve goals the most often involve scalable applications, enhanced application flexibility, and velocity-improved time to market. Based on identified gaps, the SDO leadership needs to establish specific strategic goals for incremental transition.

During the next phase, mechanisms need to be established to identify a select set of applications and related services as targets for decoupling. This determines the scope of the architectural choice guidelines and, potentially, the impact of the proposed service splits. Such impacts can be evaluated only by knowledgeable software architects. In the next phase, project managers need to reorganize software teams to execute the splits. This calls for establishing new roles and responsibilities to properly develop, deploy, and maintain microservices. The mechanisms here relate to the knowledge and skills to reorganize teams and execute related organizational change.

Table 1. Incremental transition to MSA.

Benefits	Risks	Organizational Impact
Incremental learning curve, increase confidence with less pressure	Co-existing architecture and team composition	Better alignment of business goals and technology deployment
Immediate impact on services with targeted for active development	Resource constraints	Longer uninterrupted operation
Resilience	Need to rapidly apply new skills	Better performance and reliability for critical services
Internally trained resources, need for fewer experienced external consultants	Additional roles and responsibilities for developer	Eliminate resource bottlenecks, improve customer experience
Less up-front investment, quicker infrastructure set up	Complex vendor management	Gain new customers and increase loyalty
Better work-life balance	Introduce radically new development practices (for example, DevOps)	Higher reliability through automatic monitoring results in fault isolation
Apply technologies that best fit each service, easier to switch	Data sharing (owner vs. user), control transfer between legacy and split service	Better security: Role-based access controls between services, secure network communications
Clearer ownership: "You build it, you own it."	Quality standards for diverse technology	Uninterrupted support for organizational changes
		Experimenting culture, innovative solutions

Incremental transition process phases.

Transition Process Phases	Key Decision Makers	Mechanisms	Outcomes
Strategic Goals	Business and technical executives	Strategic	Pain points and opportunities
Architectural Change	Technical leadership	Technical	Service splits
Resource Readiness	Project managers	Social	Role and skill structure
Development Practices	Software developers, software managers	Software development principles	Software development practices

In the final phase, software developers must learn and make the transition to new development practices, and software managers will acquire new responsibilities to manage service vendor relationships. The mechanism here aims to change local software development principles and provides the means to change project management practices and related development guidelines, which then result in new software development practices, such as DevOps.^{3,9}

The bottom-up arrows depict the expected impact of each phase, that is,

learning-based feedback loops. For example, a change in development practices is likely to have an impact on the teams and their skills and behaviors. A proper team structure will drive a feasible separation of services, while each decoupling step needs to be checked against the established goals to ensure the split will bring business value.

Understanding the role and impact of all four phases in regard to transition outcomes is a vital precursor to a successful MSA transition. Phasing activities and related role changes help SDOs to gradually prepare for and manage

the incremental transition effectively. In particular, if each phase is properly planned and managed, and related tasks are addressed prior to moving to the next phase, the transition is likely to be more successful in balancing the risks and benefits. Each phase involves multiple critical *activities*. Table 2 provides a summary of the key activities and facets as well as their impacts on the organization.

1. **Identify strategic goals.** The SDO must identify salient strategic goals that are currently not being properly met due to faults in software assets and related organizational capabilities. Understanding the readiness for and urgency of the MSA transition is vital for a

successful transition, or “burning platform.” SDO managers should lead the transition by asking for valid business reasons to change the architecture of software assets. Typically, these goals relate to software velocity, scalability or flexibility.¹⁷ Only such well-defined strategic goals warrant the effort to engage in the transition and calls for the SDO to carry out the following activities:

- *Recognize strategic goals for the transition.* Successful SDOs typically set up well-defined and measurable goals for MSA transition. These goals guide decision making and implementation of the organizational and technological changes to achieve them.^{11,16} One SDO, for example, established a

strategic goal of providing innovative solutions to their customers by responding to new demands in a more timely manner. This goal became the key driver for the MSA transition, which involved decoupling highly embedded and dependent services from the code base for better responsiveness and flexibility. To accomplish this, the SDO prioritized the development of modular microservices to support key business functions, enabling experimental development and innovation.

- *Identify pain points in SDO operations and assets.* The legacy system’s features and related practices that obstruct the realization of strategic goals need to be identified. As one executive

Table 2. Incremental transition phases, activities, key facets, and organizational impact.

Impact	Activity	Key Facets	Organizational Impact
Identify Organizational Strategic Goals			
	Recognize strategic goals for the transition	Strategic goals must justify the transition to MSA	Align MSA solution to support SDO's current strategic goals
	Identify pain points in SDO operations and assets	Identify parts of the legacy system that hinder the realization of goals	
	Map microservice(s) to pain points	Analyze how obstructing services can be mapped to a feasible microservice solution	
Identify Architectural Changes			
	Assess the complexity of microservice(s)	Each service split will increase complexity	Establish boundaries between monolith core and services to decouple
	Service split rules/decisions	Realize the benefit from the service split	Code freeze
	Contain the monolith core	Avoid breaking main static business logic	Gradual customer impact
Advance Resource Readiness			
	Re-organize teams	Create service-focused teams to promote the service split	Dedicate champions, change agents Ownership: “You build it, you own it.”
	Identify and train key developers	Motivate and train developers, involve consultants to train	Detach microservice teams from monolith responsibilities
	Shift roles and responsibilities	Understand the criticality of infrastructure support roles	Involve operational resources
	Obtain buy-in	Demonstrate MSA value to build internal capability	Culture shift
Change Software Development Practices			
	Support hybrid architectures	Integrate microservices with monolith core	Dual project management methodologies Controlled communication across teams
	Establish due process to find the right technology	Find fitting technology for each service	Experimenting and failure tolerant culture Failing forward
	Manage vendors	Address complex vendor relationships	Establish compliance and regulatory compliance
	Establish minimal software quality	Ensure consistently quality across technologies	Consistent service quality despite polyglot system
	Monitor user behavior	Learn from data	Real-time feedback on customer usage behavior
	Establish communication structure	Understand communication needs between monolith and MSA	Avoid cross-team dependencies

stated, “I always ask why they want to use microservices. What problem do they plan to fix with moving to microservices architecture?” For example, if the strategic goal is to increase innovativeness of solutions and improve the customer experience, bottlenecks that impede the achievement of this goal must be identified. Tightly coupled services are time consuming to change due to the dependencies and require complex and slow testing arrangements. Decoupling such services allows simpler testing, faster code change, and easier monitoring of performance and user behavior.

► *Map microservice(s) to pain points.*

A set of services must be identified as a cause of bottlenecks. By implementing a microservice in lieu of a monolith augmentation, the new service should help to achieve the identified goals. Consider a ride-hailing service similar to Uber, whereby a service is provided by using GPS coordinates for pick up and drop off, while the prices for the service are determined by dynamic supply and demand information. The data needed to run such service will have high volume, as the application must be able to compare historic data of supply and demand quickly and combine it with applicable customer characteristics to determine the price on the spot. The processing of such high volume data, while using a monolith, could easily put significant constraints on which types of pricing outcomes can be provided in real time while offering such a service. Therefore, using Apache’s Kafka Stream application as a microservice to handle the data would emerge as an appealing alternative microservice solution that would better meet system requirements.

2. Identify architectural change. The second phase helps the technical leadership to engage in critical technical decisions that relate to the scope of the architectural change needed. This change occurs by establishing a clear understanding of the technical scope and risks of the transition. Services to be isolated need to be identified and their dependencies accounted for. This helps an SDO to conduct a proper initial scoping of the transition. In this phase, it is important not only to assess the complexity that new microservices

add to the software management but also to identify those parts of the monolith that should not be decoupled, as the benefits would be marginal or the risks would outweigh them. The following activities are conducted in the architectural change phase:

► *Assess the complexity of microservice(s).*

Technical leadership needs to identify and agree on services to be split and prioritize the decoupling schedule. This requires a detailed assessment of the complexity that each split introduces to the application. Understanding and mitigating possible risks, organizational tolerance for such risks, and expected benefits are important elements to consider. For example, messaging between microservices will quickly grow in complexity and become error prone. If a code freeze is necessary until the services are developed and integrated, customers need to be notified whether this is expected and whether the reported enhancements will be delayed.

► *Service split rules/decisions.* The order of services to be decoupled should be dictated by service characteristics and related business value. In customer-facing services, if the service is deployed independently, rapid development and deployment can soon realize business value. The services to be split should be large enough and further divided as the need arises to avoid unnecessary complexity. Leadership must understand the value of proper scoping and avoid the creation of “mini-monoliths,” which often result from poorly defined cross-boundary services. A common approach is to split two to three services in a single development effort and to leave everything else in the monolith’s core. The splitting rules are not, however, always driven by business logic. They also are affected by inherent constraints of database design and data dependencies. As one interviewee stated, “The code is actually not that difficult to pull apart; it is almost always [that] the data is [sic] so much harder to pull apart because you have to know which service owns which pieces of the data.”

When a microservice has been identified for separation, either building it internally in an isolated fashion or searching for a third-party service provider is a viable option. For example, if a product rating service is added to a commercial

sales site, it can be developed in-house. Alternatively, a third-party may offer a fully tested, feature-enhanced product rating service ready to be deployed. In such a situation, it is important to understand and anticipate the future maintenance, customization options, and consequences for data ownership before committing to any final decision.

► *Contain the monolith core.* The main business logic of the application is not likely to change dramatically over time. Therefore, it is often inefficient and unnecessary to break it fully into services, especially if no scale-related bottlenecks are present. All identified services need to be interfaced with the remaining core and data shared accordingly. One microservice developer described the significance of keeping the core as follows: “You’ll always have to deal with monoliths and part of the legacy system but you are not adding onto monolith.”

3. Advance resource readiness. The identified architectural scope will drive the extent to which this phase will permeate the entire organization. If the implementation team is inexperienced with MSA transition, the order of service separation should start with a less mission-critical one. For example, a product-rating service in an e-commerce application would be a reasonable choice in contrast to the checkout or payment function. The first service isolation needs to be treated as a showcase project to demonstrate the business benefits and to obtain the organization’s buy-in. Therefore, a proper assembly and management of teams will be crucial for the long-term success of the transition. The architectural change phase helps to identify the needs for proper composition of the teams responsible for developing these first services. The phase consists of activities focused on building mechanisms that hone individuals’ skills and enables successful team formation, as described here:

► *Re-organize teams.* A software development team’s organization typically mirrors the modular software organization expressed in Conway’s Law:⁶ An organization that designs software will produce a software organization that structurally represents the organization’s current division of labor. A shift to MSA seeks to break the present software

organization, and, thus, it is necessary to divide the software teams differently to produce a new structure in the organization. Developers need to be reorganized into smaller, independent, service-focused units with less dependencies and narrower communication structures.¹⁴ The new organization needs to isolate critical independent, multi-application services within their autonomous development units. The new team structure puts pressure on maintaining the legacy system and how its current social organization operates when parts of software functionality are split into autonomous services with new responsible DevOps teams. If the organization does not have significant experience with carrying out such a radical re-organization, it is advisable to start with just one service team responsible for a non-critical service to get a sense how manage and organize such teams.

► *Identify and train key developers.* Another challenge is to identify team members who will truly advocate for the idea of MSA and are willing to undergo training and learning. Developers must learn new skills and need to remain dedicated to carrying out the change while untried technologies are integrated into a technology portfolio. Those with good business domain knowledge (especially of the services to be decoupled) and a natural drive to learn technical skills are the best candidates. The selected candidates should be fully removed from their monolith-related responsibilities to create a fully capable team that can effectively decouple, develop, deploy, and maintain the service as an independent unit. Because both architecture styles will co-exist, it is important to understand that all remaining developers will not be interested in learning new skills. These developers should be allocated to maintain the legacy code. In some situations, if all developers are interested in working on microservices, choosing to rotate between the developer roles is advisable to prevent issues between the team members.

Acquiring critical MSA design and implementation skills, such as continuous delivery, represents a radical departure from the past. Training for the DevOps skills and associated code management must be conducted. Outside consultants are commonly

invited initially to guide developers to execute the first split smoothly and to manage the code change. The training should cover the skills to build relationships with external service providers that engage in the transition effort. Developers should attend professional conferences to build up developers' critical knowledge base and create learning networks.

► *Shift roles and responsibilities.* Large SDOs commonly use established consulting firms for internal talent development. Smaller SDOs encourage self-based and paced learning but hire external consultants to lead the initial process of decoupling code and assimilating the chosen technologies. Some infrastructural and testing support roles in the new environment will be less desired, as development and operations are now largely run on and managed in the cloud. The transition team needs to involve members from business operations so that the business side understands why these changes take place. This is also necessary to create successful DevOps practices³ that place autonomous development teams at the forefront of not only developing services but also of testing and deploying them with higher velocity.²³

► *Obtain buy-in.* The success of the initial service split will create momentum for the architectural transition and help to obtain a buy-in from developers and business units. The MSA transition generally induces a change within the SDO to support the creation of modular and independent operational units responsible for specific microservices. Because MSA assumes that there are more software components to develop, test, deploy, and manage, DevOps practices need to be introduced prior to forming teams. This grants time to erect the proper infrastructure and document and develop proper metrics to measure MSA transition outcomes.¹⁰

4. Change software development practices. This phase focuses on changing key development mechanisms and technologies with the goal of finding appropriate technology to implement selected microservices. This phase also establishes standards for quality assurance and prepares the organization for more complex service vendor management. Key activities include:

► *Support hybrid architectures.* Although the teams that support MSA solutions and the monolith core will be separated by relatively limited and well-defined interfaces, they need to understand the impact of their work on one another. Well-defined service responsibilities will guarantee more rapid fault recognition and resolution. With the isolated services, the need for agility will change significantly. Management needs to introduce associated changes in project management methods to properly entrain sets of activities related to the core and microservice development. They must run agile DevOps teams to become more reactive in microservice development. The monolith application benefits from stability and requires more predictive long-term support and related practices.

► *Establish due process to find the right technology.* Developing microservices grants teams higher flexibility to choose “fit” technologies for services for which they are responsible. The chosen technologies, however, need to be formally evaluated at the higher level to ensure that their use aligns with the holistic needs of the organization. In this way, SDOs can better control diffusion and deployment of varied technologies and mitigate the risks of growing heterogeneity, poor choices, and cost escalation. If the selected technology proves to be a poor fit for the service, it can be relatively easily replaced if proper microservice specifications have been used. In this regard, one microservice developer stated, “We build this process in such a way that we make it okay to drop a new technology if we see that it’s not working out.” Most SDOs encourage experimenting, and failing early is accepted as part of the process, which ultimately promotes innovative service development.

► *Manage vendors.* The cloud platforms necessary for the development and deployment of microservices introduce myriad potential new vendors within the MSA ecosystem. Coordinating service deployment, managing contracts and compliance, and understanding and mitigating associated risks become significant new challenges. SDO management needs to prepare for and consider vendor issues prior to starting the transition.

► *Establish minimal software quality.* The proliferation of diverse technologies comes with advantages, as it helps to search for the best solutions, given particular implementation challenges. This diversity, however, needs to be matched with the call for quality and the need to minimize errors. Establishing quality standards and delivering consistently high quality across deployed technologies in this new environment remain significant challenges. One developer observed: “We have all these different microservices written in different languages, giving us different challenges during deployment ... some will be more buggy than others when the teams deploy them.”

► *Monitoring user behavior.* The real-time understanding of user interactions with the orchestrated application calls for constant instrumentation, monitoring, and analysis of large volumes of user stream data. In MSA environments, data scientists often become as important as business analysts for understanding customer needs and identifying service enhancements.

► *Establish communication structure.* Monoliths, due to a large number of unaccounted dependencies, often call for communications among all involved developers. Microservices should cut down communication overhead, as they rely primarily on a narrow band of asynchronous communications established through interface specifications. Yet, SDO management needs to allow “enough” interactions among developers across teams to avoid building additional dependencies manifested in “mini-monoliths.” Co-located teams identified this as a huge challenge, and an analyst stated, “Keeping the walls up around each microservice, I think is certainly more challenging when you have like a co-located focused product team versus a distributed team. I think the distributed team lends itself, kind of forces you to better keep those services decoupled.”


Conclusion

MSA is not a silver bullet. It will not solve all persistent problems of managing software assets. For example, decoupling monoliths and isolating critical microservices will not fix the problems that arise from designing a flawed system or those emerging from writing poor-quality code. An incremental

transition, however, can realize better business value derived from software assets if guided by the realistic and measurable goals and the SDO is aware of the cascading impact of the transitioning to the entire organization.

The SDO must have a solid reason for the MSA transition. If a realistic strategic goal can be established for the MSA transition, then the SDO needs to prepare and build the mechanisms that it can, make competently necessary decisions, and implement them properly. The transition strategy should be incremental unless the organization has significant resources and deep software experience, or the full transition will be necessary due to drastic operational failures or other business reasons.

The SDO needs to state clear, measurable benefits for service isolation that outweigh related risks and higher organizational complexity. The SDO should not decouple service without understanding its benefits for more flexible development, scalable applications, or enhanced time to market. The SDO must also establish stringent technological discipline with well-founded, standardized splitting rules. It should execute the service splitting iteratively, following established priorities, as determined by business needs.

The key to success in an MSA transition is to understand and deal with the softer social underbelly of the organization. MSA is fundamentally about a deep change in the SDO’s structure, minds, and hearts, triggered by a new technological opportunity. It shapes the organization in a holistic and punctuated manner and results, over time, in a deep transformation of the organization’s status quo, whereby its structures, roles, responsibilities, skills, incentives, and routines all are affected. SDO managers should not expect these incumbent and deeply entrenched structures to adapt to MSA automatically or by fiat. The SDO needs to be prepared for continued change and to learn to adjust the organization in a disciplined manner to the new architectural regime. Poor execution will create confused and disheartened employees, unfit applications, and, ultimately, the loss of competitiveness. 

References

- Abbott, M. and Fisher, M. *Scalability Rules: 50 Principles for Scaling Web Sites* (2nd ed.). Addison-Wesley Professional, 2016.
- Andriole, S.J. The death of big software: We are past the tipping point in the transition away from 20th-century big software architectures. *Commun. ACM* 60, 12 (Dec. 2017), 29–32.
- Balalaie, A., Heydarnoori, A., Jamshidi, P. Microservices architecture enables devops: Migration to a cloud-native architecture. *IEEE Softw.* 33, 3 (2016) 42–52.
- Bucchiarone, A., Dragoni, N., Dustdar, S., Larsen, S., and Mazzara, M. From monolithic to microservices: An experience report from the banking domain. *IEEE Softw.* 35, 3 (2018), 50–55.
- Burns, B., Grant, B., Oppenheimer, B., Brewer, E., Wilkes, J., Borg, Omega, and Kubernetes. *Commun. ACM* 59, 5 (May 2016), 50–57.
- Conway, M.E. How do committees invent? *Datamation* 14, 5 (1968), 28–31.
- Dragoni, N., Giallorenzo, S., Lafuente, A.L., Mazzara, M., Montesi, F., Mustafin, R. and Safina, L. Microservices: Yesterday, today, and tomorrow. *Present and Ulterior Software Engineering*. M. Mazzara and B. Meyer (Eds). Springer, 2017, 195–216.
- Floyd, C. *Managing Technology for Corporate Success*. Gower Publishing Ltd., 1997.
- Forsgren, N. DevOps delivers. *Commun. ACM* 61, 4 (Apr. 2018), 32–33.
- Forsgren, N. and Kersten, M. DevOps metrics. *Commun. ACM* 61, 4 (Apr. 2018), 44–48.
- Karimi, J., Bhattacharjee, A., Gupta, Y.P., and Somers, T.M. The effects of MIS steering committees on information technology management sophistication. *J. Management Info. Syst.* 17, 2 (2000) 207–230.
- Killalea, T. The hidden dividends of microservices. *Commun. ACM* 59, 8 (Aug. 2016), 42–45.
- Klock, S., van der Werf, J. M.E.M., Guelen, J.P., and Jansen, S. Workload-based clustering of coherent feature sets in microservice architectures. In *Proceedings of the IEEE Intern. Conf. Software Architecture*, 2017, 11–20.
- Knoche, H. and Hasselbring, W. Using microservices for legacy software modernization. *IEEE Softw.* 35, 3 (2018), 44–49.
- Kromhout, B. Containers will not fix your broken culture (and other hard truths). *Commun. ACM* 61, 4 (Apr. 2018), 40–43.
- Markus, M.L., Robey, D. Information technology and organizational change: Causal structure in theory and research. *Management Science* 34, 5 (1988), 583–598.
- Newman, S. *Building Microservices* (1st ed.). O’Reilly, 2016. Sebastopol, CA.
- Pahl, C. Containerization and the PaaS cloud. *IEEE Cloud Computing* 2, 3 (2015), 24–31.
- Parnas, D. L. On the criteria to be used in decomposing systems into modules. *Commun. ACM* 15, 12 (1972), 1053–1058.
- Peyrott, S. Intro to microservices, Part 4: Dependencies and data sharing. Nov. 9, 2015; <https://auth0.com/blog/introduction-to-microservices-part-4-dependencies/>
- Taibi, D. and Lenarduzzi, V. On the definition of microservice bad smells. *IEEE Softw.* 35, 3 (2018), 56–62.
- Thönes, J. Microservices. *IEEE Software* 32, 1 (2015), 113–116.
- Wiedemann, A., Forsgren, N., Wiesche, M., Gewald, H., and Kromar, H. Research for practice: the DevOps phenomenon. *Commun. ACM*, 62, 8 (Aug. 2019) 44–49.
- Yegger, S. Stevey’s Google Platforms Rant. Oct. 11, 2011; <https://plus.google.com/110981030061712822816/posts>; <https://gist.github.com/chitchcock/1281611>

Karoly Bozan (bozank@duq.edu) is an assistant professor in the Palumbo-Donahue School of Business at Duquesne University, Pittsburgh, PA, USA.

Kalle Lyytinen is a Distinguished University Professor and Iris S. Wolstein Professor of Management Design at Case Western Reserve University, Cleveland, OH, USA.

Gregory M. Rose is an associate professor in the Carson College of Business at Washington State University, Pullman, WA, USA.

MPC has moved from theoretical study to real-world usage. How is it doing?

BY YEHUDA LINDELL

Secure Multiparty Computation

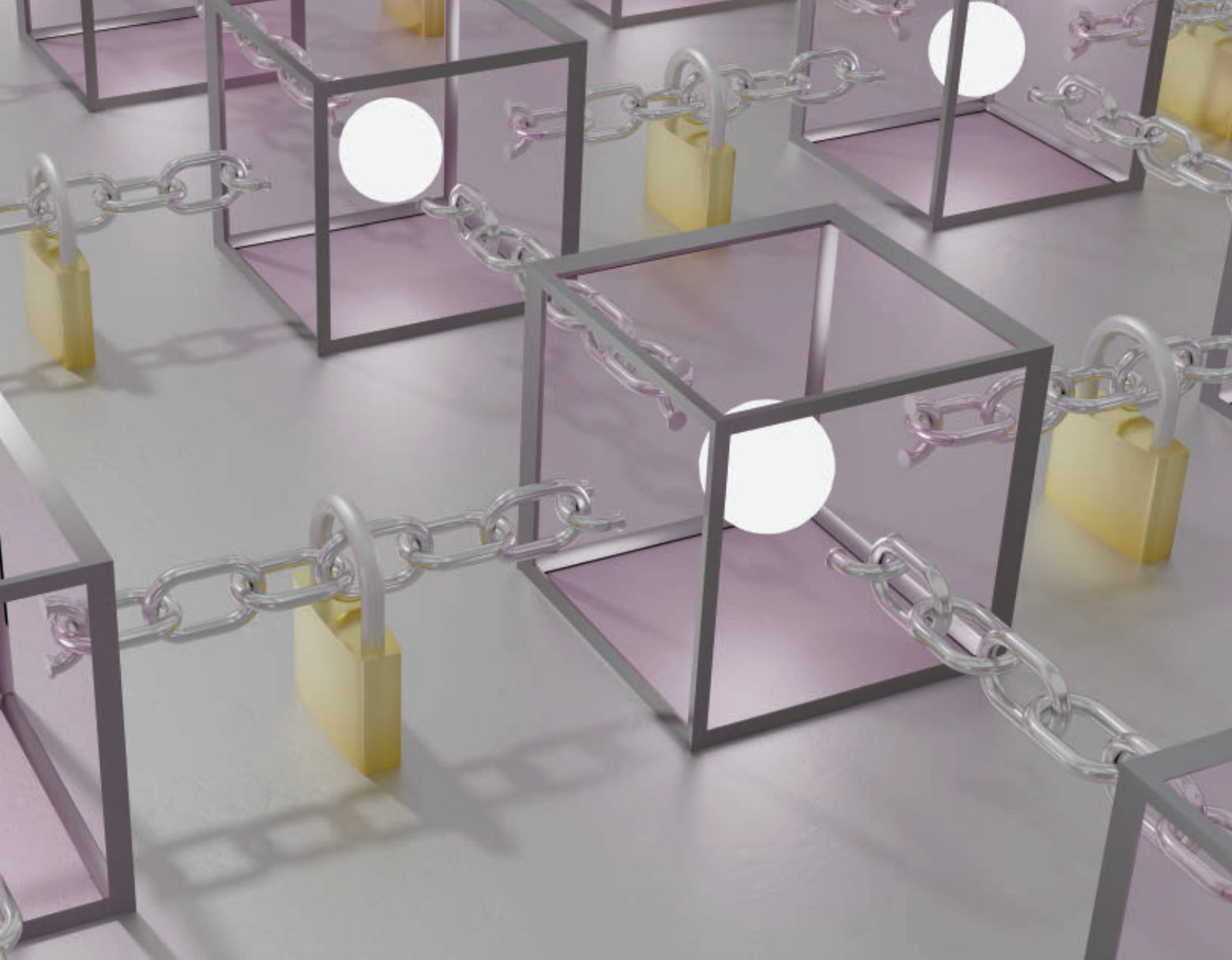
DISTRIBUTED COMPUTING CONSIDERS the scenario where a number of distinct, yet connected, computing devices (or parties) wish to carry out a joint computation of some function. For example, these devices may be servers that hold a distributed database system, and the function to be computed may be a database update of some kind. The aim of *secure multiparty computation* is to enable parties to carry out such distributed computing tasks in a secure manner. Whereas distributed computing often deals with questions of computing under the threat of machine crashes and other inadvertent faults, secure multiparty computation is concerned with the possibility of deliberately malicious behavior by some adversarial entity (these have also been considered in the distributed literature where they are called Byzantine faults). That is, it is assumed that a protocol execution may come under “attack” by an external entity, or even by a subset of the participating parties. The aim of this attack may be to learn private information or cause the result of the computation to be incorrect. Thus, two important requirements on any secure computation



protocols are *privacy* and *correctness*. The privacy requirement states that nothing should be learned beyond what is absolutely necessary; more exactly, parties should learn their output and nothing else. The correctness requirement states that each party should receive its correct output. Therefore, the

» key insights

- **Secure multiparty computation (MPC) is an extremely powerful tool, enabling parties to jointly compute on private inputs without revealing anything but the result.**
- **MPC has been studied for over three decades in academia and has strong theoretical foundations. In the past decade, huge progress has been made toward making MPC efficient enough for use in practice.**
- **In the past few years, MPC has started to be used in commercial products. There are performance costs associated with MPC protocols, but there are many real-life problems that can be solved today using existing techniques.**



adversary must not be able to cause the result of the computation to deviate from the function that the parties had set out to compute.

Secure multiparty computation can be used to solve a wide variety of problems, enabling the utilisation of data without compromising privacy. Consider, for example, the problem of comparing a person's DNA against a database of cancer patients' DNA, with the goal of finding if the person is in a high risk group for a certain type of cancer. Such a task clearly has important health and societal benefits. However, DNA information is highly sensitive, and should not be revealed to private organizations. This dilemma can be solved by running a secure multiparty computation that reveals only the category of cancer that the person's DNA is close to (or none). In this example, the privacy requirement ensures that only the category of cancer is revealed, and nothing

else about anyone's DNA (neither the DNA of the person being compared nor the DNA of the patients in the database). Furthermore, the correctness requirement guarantees that a malicious party cannot change the result (for example, make the person think that they are at risk of a type of cancer, and therefore need screening).

In another example, consider a trading platform where parties provide offers and bids, and are matched whenever an offer is greater than a bid (with, for example, the price of the trade being some function of the offer and bid prices). In such a scenario, it can be beneficial from a game theoretic perspective to not reveal the parties' actual offers and bids (because this information can be used by others in order to artificially raise prices or provide bids that are lower than their utility). Privacy here guarantees that only the match between buyer and seller and the resulting price is revealed,

and correctness would guarantee that the price revealed is the correct one according to the function (and, for example, not some lower value). It is interesting to note that in some cases privacy is more important (such as in the DNA example), whereas in others correctness is more important (such as in the trading example). In any case, MPC guarantees both of these properties, and more.

A note on terminology. In the literature, beyond secure multiparty computation (with acronym MPC, and sometimes SMPC), there are also references to secure function evaluation (SFE). These notions overlap significantly and are often used synonymously. In addition, special cases of MPC often have their own names. Two examples are private set intersection (PSI), which considers the secure computation of the intersection of private sets, and threshold cryptography, which considers the secure

computation of digital signatures and decryption, where no single party holds the private key.

Security of MPC

The definitional paradigm. As we have mentioned, the setting that we consider is one where an adversarial entity controls some subset of the parties and wishes to attack the protocol execution. The parties under the control of the adversary are called corrupted, and follow the adversary's instructions. Secure protocols should withstand any adversarial attack (where the exact power of the adversary will be discussed later). In order to formally claim and prove that a protocol is secure, a precise definition of security for multiparty computation is required. A number of different definitions have been proposed and these definitions aim to ensure a number of important security properties that are general enough to capture most (if not all) multiparty computation tasks. We now describe the most central of these properties:

- (1) *Privacy*: No party should learn anything more than its prescribed output. In particular, the only information that should be learned about other parties' inputs is what can be derived from the output itself. For example, in an auction where the only bid revealed is that of the highest bidder, it is clearly possible to derive that all other bids were lower than the winning bid. However, nothing else should be revealed about the losing bids.
- (2) *Correctness*: Each party is guaranteed that the output that it receives is correct. To continue with the example of an auction, this implies that the party with the highest bid is guaranteed to win, and no party such as the auctioneer can influence this.
- (3) *Independence of Inputs*: Corrupted parties must choose their inputs independently of the honest parties' inputs. This property is crucial in a sealed auction, where bids are kept secret and parties must fix their bids independently of others. We note that independence of inputs is *not* implied by privacy. For example, it may be

possible to generate a higher bid, without knowing the value of the original one. Such an attack can actually be carried out on some encryption schemes (that is, given an encryption of \$100, it is possible to generate a valid encryption of \$101, without knowing the original encrypted value).

- (4) *Guaranteed output delivery*: Corrupted parties should not be able to prevent honest parties from receiving their output. In other words, the adversary should not be able to disrupt the computation by carrying out a "denial of service" attack.
- (5) *Fairness*: Corrupted parties should receive their outputs if and only if the honest parties also receive their outputs. The scenario where a corrupted party obtains output and an honest party does not should not be allowed to occur. This property can be crucial, for example, in the case of contract signing. Specifically, it would be very problematic if the corrupted party received the signed contract and the honest party did not. Note that guaranteed output delivery implies fairness, but the converse is not necessarily true.

We stress that this list does *not* constitute a definition of security, but rather a set of requirements that should hold for any secure protocol. Indeed, one possible approach to defining security is to just generate a list of separate requirements (as mentioned) and then say that a protocol is secure if all of these requirements are fulfilled. However, this approach is not satisfactory for the following reasons. First, it may be possible that an important requirement was missed. This is especially true because different applications have different requirements, and we would like a definition that is general enough to capture all applications. Second, the definition should be simple enough so that it is trivial to see that *all* possible adversarial attacks are prevented by the proposed definition.

The standard definition today⁵ therefore formalizes security in the following general way. As a mental experiment, consider an "ideal world" in which an external trusted (and

incorruptible) party is willing to help the parties carry out their computation. In such a world, the parties can simply send their inputs to the trusted party, who then computes the desired function and passes each party its prescribed output. As the only action carried out by a party is that of sending its input to the trusted party, the only freedom given to the adversary is in choosing the corrupted parties' inputs. Notice that all of the described security properties (and more) hold in this ideal computation. For example, privacy holds because the only message ever received by a party is its output (and so it cannot learn any more than this). Likewise, correctness holds because the trusted party cannot be corrupted and so will always compute the function correctly.


Of course, in the "real world," there is no external party that can be trusted by all parties. Rather, the parties run some protocol among themselves without any help, and some of them are corrupted and colluding. Despite this, a secure protocol should emulate the so-called "ideal world." That is, a real protocol that is run by the parties (in a world where no trusted party exists) is said to be *secure*, if no adversary can do more harm in a real execution than in an execution that takes place in the ideal world. This can be formulated by saying that for any adversary carrying out a successful attack in the real world, there exists an adversary that successfully carries out an attack with the same effect in the ideal world. However, successful adversarial attacks *cannot* be carried out in the ideal world. We therefore conclude that all adversarial attacks on protocol executions in the real world must also fail.

More formally, the security of a protocol is established by comparing the outcome of a real protocol execution to the outcome of an ideal computation. That is, for any adversary attacking a real protocol execution, there exists an adversary attacking an ideal execution (with a trusted party) such that the input/output distributions of the adversary and the participating parties in the real and ideal executions are essentially the same. Thus a real protocol execution "emulates" the ideal world. This formulation of security is


called the ideal/real simulation paradigm. In order to motivate the usefulness of this definition, we describe why all the properties described are implied. Privacy follows from the fact that the adversary's output is the same in the real and ideal executions. Because the adversary learns nothing beyond the corrupted party's outputs in an ideal execution, the same must be true for a real execution. Correctness follows from the fact that the honest parties' outputs are the same in the real and ideal executions, and from the fact that in an ideal execution, the honest parties all receive correct outputs as computed by the trusted party. Regarding independence of inputs, notice that in an ideal execution, all inputs are sent to the trusted party before any output is received. Therefore, the corrupted parties know nothing of the honest parties' inputs at the time that they send their inputs. In other words, the corrupted parties' inputs are chosen independently of the honest parties' inputs, as required. Finally, guaranteed output delivery and fairness hold in the ideal world because the trusted party always returns all outputs. The fact that it also holds in the real world again follows from the fact that the honest parties' outputs are the same in the real and ideal executions.

We remark that in some cases, the definition is relaxed to exclude fairness and guaranteed output delivery. The level of security achieved when these are excluded is called "security with abort," and the result is that the adversary may be able to obtain output, whereas the honest parties do not. There are two main reasons why this relaxation is used. First, in some cases, it is impossible to achieve fairness (for example, it is impossible to achieve fair coin tossing for two parties¹¹). Second, in some cases, more efficient protocols are known when fairness is not guaranteed. Thus, if the application does not require fairness (and in particular in cases where only one party receives output), this relaxation is helpful.

Additional definitional parameter. Adversarial power. The informal definition of security omits one very important issue: the power of the adversary that attacks a protocol execution. As we have mentioned, the



The security of a protocol is established by comparing the outcome of a real protocol execution to the outcome of an ideal computation.




adversary controls a subset of the participating parties in the protocol. However, we have not defined what power such an adversary has. We describe the two main parameters defining the adversary: its allowed adversarial behavior (that is, does the adversary just passively gather information or can it instruct the corrupted parties to act maliciously?) and its corruption strategy (that is, when or how parties come under the "control" of the adversary?):

(1) **Allowed adversarial behavior:**


The most important parameter that must be to the actions that corrupted parties are allowed to take. There are three main types of adversaries:

- (a) Semi-honest adversaries: In the semi-honest adversarial model, even corrupted parties correctly follow the protocol specification. However, the adversary obtains the internal state of all the corrupted parties (such as the transcript of all the messages received) and attempts to use this to learn information that should remain private. This is a rather weak adversarial model, but a protocol with this level of security does guarantee that there is no inadvertent data leakage. In some cases, this is sufficient although in today's adversarial environment it is often insufficient. Semi-honest adversaries are also called "honest-but-curious" and "passive." (Sometimes, *fail-stop* adversaries are also considered; these are essentially semi-honest adversaries who may also halt the protocol execution early.)
- (b) Malicious adversaries: In this adversarial model, the corrupted parties can *arbitrarily* deviate from the protocol specification according to the adversary's instructions. In general, providing security in the presence of malicious adversaries is preferred, as it ensures that no adversarial attack can succeed. Malicious adversaries are also called "active."

- (c) Covert adversaries:¹ This type of adversary may behave maliciously in an attempt to break the protocol. However, the security guarantee provided is that if it does attempt such an attack, then it will be detected with some specified probability that can be tuned to the application. We stress that unlike in the malicious model, if the adversary is not detected, then it may successfully cheat (for example, learn an honest party's input). This model is suited to settings where some real-world penalty can be associated with an adversary being detected, and the adversary's expectation is to lose overall if it attempts an attack.
- (2) **Corruption strategy:** The corruption strategy deals with the question of when and how parties are corrupted. There are three main models:
- (a) Static corruption model: In this model, the set of parties controlled by the adversary is fixed before the protocol begins. Honest parties remain honest throughout and corrupted parties remain corrupted.
- (b) Adaptive corruption model: Rather than having a fixed set of corrupted parties, adaptive adversaries are given the capability of corrupting parties during the computation. The choice of who to corrupt, and when, can be arbitrarily decided by the adversary and may depend on its view of the execution (for this reason it is called adaptive). This strategy models the threat of an external “hacker” breaking into a machine during an execution, or a party which is honest initially and later changes its behavior. We note that in this model, once a party is corrupted, it remains corrupted from that point on.
- (c) Proactive security model:^{7,30} This model considers the possibility that parties are corrupted for a certain period of time only. Thus, honest parties may become corrupted throughout the computation



In reality, a secure multiparty computation protocol is not run in isolation; rather, it is part of a system.



(such as in the adaptive adversarial model), but corrupted parties may also become honest. The proactive model makes sense in cases where the threat is an external adversary who may breach networks and break into services and devices, and secure computations are ongoing. When breaches are discovered, the systems are cleaned and the adversary loses control of some of the machines, making the parties honest again. The security guarantee is that the adversary can only learn what it derived from the local state of the machines that it corrupted, although they were corrupted. Such an adversary is sometimes called mobile.

There is no “right” model when considering this information. Rather, the specific definition used and adversary considered depend on the application and the threats being dealt with.

Modular sequential and concurrent composition. In reality, a secure multiparty computation protocol is not run in isolation; rather, it is part of a system. Canetti⁵ proved that if you run an MPC protocol as part of a larger system, then it still behaves in the same way as if an incorruptible trusted party carried out the computation for the parties. This powerful theorem is called modular composition, and it enables larger protocols to be constructed in a modular way using secure subprotocols, as well as analysing a larger system that uses MPC for some of the computations.

One important question in this context is whether or not the MPC protocol itself runs at the same time as other protocols. In the setting of *sequential composition*, the MPC protocol can run as a subprotocol of another protocol with arbitrary other messages being sent before and after the MPC protocol. However, the MPC protocol itself must be run without any other messages being sent in parallel. This is called the stand-alone setting and is the setting considered by the basic definition of security of Canetti.⁵ The sequential modular composition theorem of Canetti⁵ states that in this setting, the

MPC protocol indeed behaves like a computation carried out by a trusted third party.

In some (many) cases, MPC protocols are run at the same time as other instances of itself, other MPC protocols, and other insecure protocols. In these cases, a protocol proven secure under the aforementioned stand-alone definition of security may not actually remain secure. A number of definitions were proposed to deal with this setting, the most popular of these is that of universal composability.⁶ Any protocol proven secure according to this definition is guaranteed to behave like an ideal execution, irrespective of what other protocols run concurrently to it. As such, this is the gold standard of MPC definitions. However, it does come at a price (both of efficiency and of assumptions required on the system setup).

Important definitional implications. The ideal model and using MPC in practice. The ideal/real paradigm for defining security actually has some very important implications for the use of MPC in practice. Specifically, in order to *use* an MPC protocol, all a practitioner needs to do is to consider the security of their system when an incorruptible trusted party carries out the computation for which MPC is used. If the system is secure in this case, then it will remain secure even when the real MPC protocols are used (under the appropriate composition case). This means that noncryptographers need not understand anything about *how* MPC protocols work, or even how security is defined. The ideal model provides a clean and easy to understand abstraction that can be utilized by those constructing systems.

Any inputs are allowed. Although the ideal model paradigm provides a simple abstraction, as described there is a subtle point that is sometime misunderstood. An MPC protocol behaves like an ideal execution; as such, the security obtained is analogous to that of an ideal execution. However, in an ideal execution, adversarial parties may input any values that they wish, and indeed there is no generic way of preventing this. Thus, if two people wish to see who earns a higher salary (without revealing any more than this one bit of

information), then nothing stops one of them from inputting the maximum possible value as their salary (and then behaving honestly in the MPC protocol itself), with the result being that the output is that they earn more. Thus, if the security of an application depends on the party's using *correct inputs*, then mechanisms must be used to enforce this. For example, it is possible to require signed inputs and have the signature be verified as part of the MPC computation. Depending on the specific protocol, this can add significant cost.

MPC secures the process, but not the output. Another subtlety that is often misunderstood is that MPC secures the process, meaning that nothing is revealed by the computation itself. However, this does not mean that the output of the function being computed does not reveal sensitive information. For an extreme example, consider two people computing the average of their salaries. It is indeed true that nothing but the average will be output, but given a person's own salary and the average of both salaries, they can derive the exact salary of the other person. Thus, just using MPC does not mean that all privacy concerns are solved. Rather, MPC secures the computing process, and the question of what functions should and should not be computed due to privacy concerns still needs to be addressed. In some cases, such as threshold cryptography, this question is not an issue (because the output of cryptographic functions does not reveal the key, assuming that it is secure). However, in other cases, it may be less clear.

Feasibility of MPC

The definition of security seems to be very restrictive in that no adversarial success is tolerated, and the protocol should behave as if a trusted third party is carrying out the computation. Thus, one may wonder whether it is even possible to obtain secure protocols under this definition, and if yes, for which distributed computing tasks. Perhaps surprisingly, powerful feasibility results have been established, demonstrating that in fact, *any* distributed computing task (function) can be securely computed, in the presence of

malicious adversaries. We now briefly state the most central of these results. Let n denotes the number of participating parties and let t denotes a bound on the number of parties that may be corrupted (where the identity of the corrupted parties is unknown):

- (1) For $t < n/3$ (that is, when less than a third of the parties can be corrupted), secure multiparty protocols with fairness and guaranteed output delivery can be achieved for any function with computational security assuming a synchronous point-to-point network with authenticated channels,¹⁸ and with information-theoretic security assuming the channels are also private.^{3,9}
- (2) For $t < n/2$ (that is, in the case of a guaranteed honest majority), secure multiparty protocols with fairness and guaranteed output delivery can be achieved for any function with computational and information-theoretic security, assuming that the parties also have access to a broadcast channel.^{18,33}
- (3) For $t \geq n/2$ (that is, when the number of corrupted parties is not limited), secure multiparty protocols (without fairness or guaranteed output delivery) can be achieved.^{18,37}

In the setting of concurrent composition described earlier, it has also been shown that any function can be securely computed.^{6,8}

In summary, secure multiparty protocols exist for any distributed computing task. This fact is what provides its huge potential—whatever needs to be computed can be computed securely! We stress, however, that the aforementioned feasibility results are *theoretical*, meaning that they demonstrate that this is possible in principle. They do not consider the practical efficiency costs incurred; these will be mentioned here later.

We conclude this section with a caveat. The feasibility results are proven in specific models, and under cryptographic hardness and/or setting assumptions. It is beyond the scope of this review to describe these details, but it is important to be aware that they need to be considered.

Techniques

Over the past three decades, many different techniques have been developed for constructing MPC protocols with different properties, and for different settings. It is way beyond the scope of this article to even mention all of the techniques, and we highly recommend reading¹⁵ for an extremely well-written and friendly introduction to MPC, such as a survey of the major techniques. Nevertheless, we will provide a few simple examples of how MPC protocols are constructed, in order to illustrate how it can work.

Shamir secret sharing. MPC protocols for an honest majority typically utilize secret sharing as a basic tool. We will therefore begin by briefly describing Shamir's secret sharing scheme.³⁴

A secret sharing scheme solves the problem of a dealer who wishes to share a secret s among n parties, so that any subset of $t + 1$ or more of the parties can reconstruct the secret, yet no subset of t or fewer parties can learn anything about the secret. A scheme that fulfills these requirements is called a $(t + 1)$ -out-of- n -threshold secret-sharing scheme.

Shamir's secret sharing scheme utilizes the fact that for any for $t + 1$ points on the two dimensional plane $(x_1, y_1), \dots, (x_{t+1}, y_{t+1})$ with unique x_i , there exists a unique polynomial $q(x)$ of degree at most t such that $q(x_i) = y_i$ for every i . Furthermore, it is possible to efficiently reconstruct the polynomial $q(x)$, or any specific point on it. One way to do this is with the Lagrange basis polynomials $\ell_1(x), \dots, \ell_t(x)$, where reconstruction is carried out by computing $q(x) = \sum_{i=1}^{t+1} \ell_i(x) \cdot y_i$. From here on, we will assume that all computations are in the finite field \mathbb{Z}_p , for a prime $p > n$.

Given this, in order to share a secret s , the dealer chooses a random polynomial $q(x)$ of degree at most t under the constraint that $q(0) = s$. (Concretely, the dealer sets $a_0 = s$ and chooses random coefficients $a_1, \dots, a_t \in \mathbb{Z}_p$, and sets $q(x) = \sum_{i=0}^t a_i \cdot x^i$.) Then, for every $i = 1, \dots, n$, the dealer provides the i th party with the share $y_i = q(i)$; this is the reason why we need $p > n$, so that different shares can be given to each party. Reconstruction by a subset of any t parties works by simply interpolating the

polynomial to compute $q(x)$ and then deriving $s = q(0)$. Although $t + 1$ parties can completely recover s , it is not hard to show that *any* subset of t or fewer parties cannot learn anything about s . This is due to the fact that they have t or fewer points on the polynomial, and so there exists a polynomial going through these points and the point $(0, s)$ for every possible $s \in \mathbb{Z}_p$. Furthermore, because the polynomial is random, all polynomials are equally likely, and so all values of $s \in \mathbb{Z}_p$ are equally likely.

Honest-majority MPC with secret sharing. The first step in most protocols for *general* MPC (that is, protocols that can be used to compute any function) is to represent the function being computed as a Boolean or arithmetic circuit. In the case of honest-majority MPC based on secret sharing, the arithmetic circuit (comprised of multiplication and addition gates) is over a finite field \mathbb{Z}_p with $p > n$. We remark that arithmetic circuits are Turing complete, and so any function can be represented in this form. The parties participating in the MPC protocol are all provided in this circuit, and we assume they can all communicate securely with each other. The protocol for semi-honest adversaries (see here for what is needed for the case of malicious adversaries) consists of the following phases:

- (1) **Input sharing:** In this phase, each party shares its input with the other parties, using Shamir's secret sharing. That is, for each input wire to the circuit, the party whose input is associated with that wire plays the dealer in Shamir's secret sharing to share the value to all parties. The secret sharing used is $(t + 1)$ -out-of- n , with $t = \frac{n-1}{2}$ (thus, the degree of the polynomial is t). This provides security against any minority of corrupted parties, because no such minority can learn anything about the shared values. Following this step, the parties hold secret shares of the values on each input wire.
- (2) **Circuit evaluation:** In this phase, the parties evaluate the circuit one gate at a time, from the input gates to the output gates. The evaluation maintains the invariant that for

every gate for which the parties hold $(t + 1)$ -out-of- n sharings of the values on the two input wires, the result of the computation is a $(t + 1)$ -out-of- n secret sharing of the value on the output wire of the gate.

- (a) **Computing addition gates:** According to the invariant, each party holds a secret sharing of the values on the input wires to the gate; we denote these polynomials by $a(x)$ and $b(x)$ and this means that the i th party holds the values $a(i)$ and $b(i)$. The output wire of this gate should be a $(t + 1)$ -out-of- n secret sharing of the value $a(0) + b(0)$. This is easily computed by the i th party locally setting its share on the output wire to be $a(i) + b(i)$. Observe that by defining the polynomial $c(x) = a(x) + b(x)$, this means that the i th party holds $c(i)$. Furthermore, $c(x)$ is a degree- t polynomial such that $c(0) = a(0) + b(0)$. Thus, the parties hold a valid $(t + 1)$ -out-of- n secret sharing of the value $a(0) + b(0)$, as required. Observe that no communication is needed in order to compute addition gates.
- (b) **Computing multiplication gates:** Once again, denote the polynomials on the input wires to the gate by $a(x)$ and $b(x)$. As for an addition gate, the i th party can locally multiply its shares to define $c(i) = a(i) \cdot b(i)$. By the properties of polynomial multiplication, this defines a polynomial $c(x)$ such that $c(0) = a(0) \cdot b(0)$. Thus, $c(x)$ is a sharing of the correct value (the product of the values on the input wires). However, $c(x)$ is of degree- $2t$, and thus, this is a $(2t + 1)$ -out-of- n secret sharing and not a $(t + 1)$ -out-of- n secret sharing. In order to complete the computation of the multiplication gate, it is therefore necessary for the parties to carry out a *degree reduction* step, to securely reduce the degree of the polynomial shared among the

parties from $2t$ to t , without changing its value at 0. Before proceeding to describe this, observe that as $t < n/2$, the shares held by the n parties do fully determine the polynomial $c(x)$ of degree $2t + 1$.

In order to compute the degree reduction step, we use an idea from Damgård and Nielsen¹² (we describe the basic idea here although Damgård and Nielsen¹² have a far more efficient way of realizing it than what we describe here). Assume that the parties all hold two independent secret sharings of an unknown random value r , the first sharing via a polynomial of degree- $2t$ denoted $R_{2t}(x)$, and the second sharing via a polynomial of degree- t denoted $R_t(x)$. Note that $R_{2t}(0) = R_t(0) = r$. Then, each party can locally compute its share of the degree- $2t$ polynomial $d(x) = c(x) - R_{2t}(x)$ by setting $d(i) = c(i) - R_{2t}(i)$. Note that both $c(x)$ and $R_{2t}(x)$ are of degree- $2t$. Next, the parties reconstruct $d(0) = a(0) \cdot b(0) - r$ by sending all of their shares to all other parties. Finally, the i th party for all $i = 1, \dots, n$ computes its share on the output wire to be $c'(i) = R_t(i) + d(0)$.

Observe that $c'(x)$ is of degree t as $R_t(x)$ is of degree t , and it is defined by adding a constant $d(0)$ to $R_t(x)$. Next, $c'(0) = a(0) \cdot b(0)$ as $R_t(0) = r$ and $d(0) = a(0) \cdot b(0) - r$; thus r cancels out when summing the values. Thus, the parties hold a valid $(t + 1)$ -out-of- n secret sharing of the product of the values on the input wires, as required. Furthermore, note that the value $d(0)$ that is revealed to all parties does not leak any information because $R_t(x)$ perfectly masks all values of $c(x)$, and in particular it masks the value $a(0) \cdot b(0)$.

It remains to show how the parties generate two independent secret sharings of an unknown random value r via polynomials of degree $2t$ and t . This can be achieved by the i th party, for all $i = 1, \dots, n$, playing the dealer and sharing a random value r_i via a degree- $2t$ polynomial $R_{2t}^i(x)$ and via a degree- t polynomial $R_t^i(x)$. Then, upon receiving such shares from each of the parties, the i th party for all $i = 1, \dots, n$ defines its shares of $R_{2t}(x)$ and $R_t(x)$ by computing $R_{2t}(i) = \sum_{j=1}^n R_{2t}^j(i)$ and $R_t(i) = \sum_{j=1}^n R_t^j(i)$. Because all parties

Over the past three decades, many different techniques have been developed for constructing MPC protocols with different properties, and for different settings.

contribute secret random values r_1, \dots, r_n and we have that $r = \sum_{j=1}^n r_j$, it follows that no party knows r .

- (3) **Output reconstruction:** Once the parties have obtained shares on the output wires, they can obtain the outputs by simply sending their shares to each other and reconstructing the outputs via interpolation. Observe that it is also possible for different parties to obtain different outputs, if desired. In this case, the parties send the shares for reconstruction only to the relevant parties who are supposed to obtain the output on a given wire.

This protocol is secure for *semi-honest adversaries* as long as less than $n/2$ parties are corrupted. This is because the only values seen by the parties during the computation are secret shares (that reveal nothing about the values they hide), and opened $d(0)$ values that reveal nothing about the actual values on the wires due to the independent random sharings used each time. Note that in order to achieve security in the presence of *malicious adversaries* who may deviate from the protocol specification, it is necessary to utilize different methods to prevent cheating. See Beerliová-Trubíniová and Hirt⁴, Chida et al.¹⁰ and Furukawa and Lindell¹⁶ for a few examples of how to efficiently achieve security in the presence of malicious adversaries.

Private set intersection. Earlier we described an approach to *general* secure computation that can be used to securely compute any function. In many cases, these general approaches turn out to actually be the most efficient (especially when considering malicious adversaries). However, in some cases, the specific structure of the function being solved enables us to find faster, tailored solutions. In this and the next section, we present two examples of such functions.


In a private set intersection protocol, two parties with private sets of values wish to find the intersection of the sets, without revealing anything but the elements in the intersection. In some cases, some function of the intersection is desired, such as its size only. There

has been a lot of work on this problem, with security for both semi-honest and malicious adversaries, and with different efficiency goals (few rounds, low communication, low computation, etc.). In this section, we describe the idea behind the protocol of Kolesnikov et al.;²³ the actual protocol of Kolesnikov et al.²³ is far more complex, but we present the conceptually simple idea underlying their construction.


A pseudorandom function F is a keyed function with the property that outputs of the function on known inputs look completely random. Thus, for any given list of elements x_1, \dots, x_n , the series of values $F_k(x_1), \dots, F_k(x_n)$ looks random. In particular, given $F_k(x_i)$, it is infeasible to determine the value of x_i . In the following simple protocol, we utilize a tool called *oblivious pseudorandom function evaluation*. This is a specific type of MPC protocol where the first party inputs k and the second party inputs x , and the second party receives $F_k(x)$, whereas the first party learns nothing about x (note that the second party learns $F_k(x)$ but nothing beyond that; in particular, k remains secret). Such a primitive can be built in many ways, and we will not describe them here.

Now, consider two parties with respective sets of private elements; denote them x_1, \dots, x_n and y_1, \dots, y_n , respectively (for simplicity, we assume that their lists are of the same size, although this is not needed). Then, the protocol proceeds as follows:

- (1) The first party chooses a key k for a pseudorandom function.
- (2) The two parties run n oblivious pseudorandom function evaluations: in the i th execution, the first party inputs k and the second party inputs y_i . As a result, the second party learns $F_k(y_1), \dots, F_k(y_n)$, whereas the first party learns nothing about y_1, \dots, y_n .
- (3) The first party locally computes $F_k(x_1), \dots, F_k(x_n)$ and sends the list to the second party. It can compute this because it knows k .
- (4) The second party computes the intersection between the lists $F_k(y_1), \dots, F_k(y_n)$ and $F_k(x_1), \dots, F_k(x_n)$, and outputs all values y_j for which $F_k(y_j)$ is in the intersection. (The party knows these values because



The aim of threshold cryptography is to enable a set of parties to carry out cryptographic operations, without any single party holding the secret key.



it knows the association between y_j and $F_k(y_j)$.)

The protocol reveals nothing but the intersection because the first party learns nothing about y_1, \dots, y_n from the oblivious pseudorandom function evaluations, and the second party learns nothing about values of x_j that are not in the intersection because the pseudorandom function hides the preimage values. This is therefore secure in the semi-honest model. It is more challenging to achieve security in the malicious model. For example, a malicious adversary could use a different key for the first element and later elements, and then have the result that the value y_1 is in the output if and only if it was the first element of the second party's list.

The most efficient private set intersection protocols today use advanced hashing techniques and can process millions of items in a few seconds.^{23, 31, 32}

Threshold cryptography. The aim of threshold cryptography is to enable a set of parties to carry out cryptographic operations, without any single party holding the secret key. This can be used to ensure multiple signatories on a transaction, or alternatively to protect secret keys from being stolen by spreading key shares out on different devices (so that the attacker has to breach all devices in order to learn the key). We demonstrate a very simple protocol for two-party RSA, but warn that for more parties (and other schemes), it is much more complex.

RSA is a public-key scheme with public-key (e, N) and private-key (d, N) . The basic RSA function is $y = x^e \bmod N$, and its inverse function is $x = y^d \bmod N$. RSA is used for encryption and signing, by padding the message and other techniques. Here, we relate to the *raw* RSA function, and show how the inverse can be computed securely amongst two parties, where neither party can compute the function itself. In order to achieve this, the system is set up with the first party holding (d_1, N) and the second party holding (d_2, N) , where d_1 and d_2 are random under the constraint that $d_1 + d_2 = d$. (More formally, the order in the exponent is $\phi(N)$ —Euler's function—and therefore the values $d_1, d_2 \in \mathbb{Z}_{\phi(N)}$ are random under the constraint that $d_1 + d_2 = d \bmod$

$\phi(N)$.) In order to securely compute $y^d \bmod N$, the first party computes $x_1 = y^{d_1} \bmod N$, the second party computes $x_2 = y^{d_2} \bmod N$, and these values are exchanged between them. Then, each party computes $x = x_1 \cdot x_2 \bmod N$, verifies that the output is correct by checking that $x^e = y \bmod N$, and if yes outputs x . Observe that this computation is correct because

$$x = y^{d_1} \cdot y^{d_2} \bmod N = y^{d_1 + d_2 \bmod \phi(N)} \bmod N = y^d \bmod N.$$

In addition, observe that given the output x and its share d_1 of the private exponent, the first party can compute $x_2 = y / y^{d_1} \bmod N$ (this is correct because $x_2 = y^{d_2} = y^{d_1 + d_2 - d_1} = y / y^{d_1} \bmod N$). This means that the first party does not learn anything more than the output from the protocol, as it can generate the messages that it receives in the protocol by itself from its own input and the output.

We stress that full-blown threshold cryptography supports quorum approvals involving many parties (for example, requiring $(t + 1)$ -out-of- n parties to sign, and maintaining security for any subset of t corrupted parties). This needs additional tools, but can also be done very efficiently; see Shoup³⁵ and references within. Recently, there has been a lot of interest in threshold ECDSA due to its applications to protecting cryptocurrencies.^{14, 17, 26, 27}

Dishonest-majority MPC. Previously, we described a general protocol for MPC that is secure as long as an adversary cannot corrupt more than a minority of the parties. In the case of a dishonest majority, including the important special case of two parties (with one corrupted), completely different approaches are needed. There has been a very large body of work in this direction, from the initial protocols of Beaver et al.², Goldreich et al.¹⁸, and Yao³⁷ that focused on feasibility, and including a lot of recent work focused on achieving concrete efficiency. There is so much work in this direction that any attempt to describe it here will do it a grave injustice. We therefore refer the reader to Evans et al.¹⁵ for a description of the main approaches, including the GMW oblivious transfer approach,^{18, 21} garbled circuits,^{2, 37} cut-and-choose,²⁸ SPDZ,¹³ TinyOT,²⁹ MPC in the head,²² and more. (We stress that for each of these

approaches, there have been many follow-up works, achieving increasingly better efficiency.)

Efficient and practical MPC. The first 20 years of MPC research focused primarily on feasibility: how to define and prove security for multiple adversarial and network models, under what cryptographic and setup assumptions it is possible to achieve MPC, and more. The following decade saw a large body of research around making MPC more and more efficient. The first steps of this process were purely algorithmic and focused on reducing the overhead of the cryptographic primitives. Following this, other issues were considered that had significant impact: the memory and communication, utilisation of hardware instructions such as AES-NI, and more. In addition, as most general protocols require the circuit representation of the function being computed, and circuits are hard to manually construct, special purpose MPC compilers from code to circuits were also constructed. These compilers are tailored to be sensitive to the special properties of MPC. For example, in many protocols XOR gates are computed almost for free,²⁴ in contrast to AND/OR gates that cost. These compilers therefore minimize the number of AND gates, even at the expense of considerably more XOR gates. In addition, the computational cost of some protocols is dominated by the circuit size, whereas in others, it is dominated by the circuit depth. Thus, some compilers aim to generate the smallest circuit possible, whereas others aim to generate a circuit with the lowest depth. See Hastings et al.¹⁹ for a survey on general-purpose compilers for MPC and their usability. The combination of these advancements led to performance improvements of many orders of magnitude in just a few years, paving the way for MPC to be fast enough to be used in practice for a wide variety of problems. See Evans et al.¹⁵ (Chapter 4) for a description of a few of the most significant of these advancements.

MPC Use Cases

There are many great theoretical examples of where MPC can be helpful. It can be used to compare no-fly lists in a privacy-preserving manner, to enable private DNA comparisons for medical

and other purposes, to gather statistics without revealing anything but the aggregate results, and much more. Up until very recently, these theoretical examples of usage were almost all we had to say about the potential benefits of MPC. However, the situation today is very different. MPC is now being used in multiple real-world use cases, and usage is growing fast.

We will conclude this article with some examples of MPC applications that have been actually deployed.

Boston wage gap.²⁵ The Boston Women's Workforce Council used MPC in 2017 in order to compute statistics on the compensation of 166,705 employees across 114 companies, comprising roughly 16% of the Greater Boston area workforce. The use of MPC was crucial because companies would not provide their raw data due to privacy concerns. The results showed that the gender gap in the Boston area is even larger than previously estimated by the U.S. Bureau of Labor Statistics. This is a powerful example demonstrating that MPC can be used for social good.

Advertising conversion.²⁰ In order to compute accurate conversion rates from advertisements to actual purchases, Google computes the *size* of the intersection between the list of people shown an advertisement to the list of people actually purchasing the advertised goods. When the goods are not purchased online and so the purchase connection to the shown advertisement cannot be tracked, Google and the company paying for the advertisement have to share their respective lists in order to compute the intersection size. In order to compute this without revealing anything but the size of the intersection, Google utilizes a protocol for privacy-preserving set intersection. The protocol used by Google is described in Ion et al.²⁰ Although this protocol is far from the most efficient known today, it is simple and meets their computational requirements.

MPC for cryptographic key protection.³⁸ As described in earlier, threshold cryptography provides the ability to carry out cryptographic operations (such as decryption and signing) without the private key being held in any single place. A number of companies are using threshold cryptography as an

alternative to legacy hardware for protecting cryptographic keys. In this application, MPC is not run between different parties holding private information. Rather, a single organization uses MPC to generate keys and compute cryptographic operations, without the key ever being in a single place where it can be stolen. By placing the key shares in different environments, it is very hard for an adversary to steal all shares and obtain the key. In this setting, the proactive model described earlier is the most suitable. Another use of MPC in this context is for protecting the signing keys used for protecting cryptocurrencies and other digital assets. Here, the ability to define general quorums enables the cryptographic enforcement of strict policies for approving financial transactions, or to share keys between custody providers and clients.

Government collaboration.³⁹ Different governmental departments hold information about citizens, and significant benefit can be obtained by correlating that information. However, the privacy risks involved in pooling private information can prevent governments from doing this. For example, in 2000, Canada scrapped a program to pool citizen information, under criticism that they were building a “big brother database.” Utilising MPC, Estonia collected encrypted income tax records and higher education records to analyze if students who work during their degree are more likely to fail than those focusing solely on their studies. By using MPC, the government was guaranteed that all data protection and tax secrecy regulations were followed without losing data utility.

Privacy-preserving analytics.⁴⁰ Machine learning usage is increasing rapidly in many domains. MPC can be used to run machine learning models on data without revealing the model (which contains precious intellectual property) to the data owner, and without revealing the data to the model owner. In addition, statistical analyses can be carried out between organizations for the purpose of anti-money laundering, risk score calculations, and more.

Discussion

Secure multiparty computation is a fantastic example of success in the long

game of research.³⁶ For the first 20 years of MPC research, no applications were in sight, and it was questionable whether or not MPC would ever be used. In the past decade, the state of MPC usability has undergone a radical transformation. In this time, MPC has not only become fast enough to be used in practice, but it has received industry recognition and has made the transition to a technology that is deployed in practice. MPC still requires great expertise to deploy, and additional research breakthroughs are needed to make secure computation practical on large data sets and for complex problems, and to make it easy to use for nonexperts. The progress from the past few years, and the large amount of applied research now being generated, paints a positive future for MPC in practice. Together with this, deep theoretical work in MPC continues, ensuring that applied MPC solutions stand on strong scientific foundations. ■

References

1. Aumann, Y., Lindell, Y. Security against covert adversaries: Efficient protocols for realistic adversaries. *J. Cryptol.* 23, 2 (2010), 281–343 (extended abstract at *TCC 2007*).
2. Beaver, D., Micali, S., Rogaway, P. The round complexity of secure protocols. In *22nd STOC* (1990), 503–513.
3. Ben-Or, M., Goldwasser, S., Wigderson, A. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *20th STOC* (1988), 1–10.
4. Beerliová-Trubíniová, Z., Hirt, M. Perfectly-secure MPC with linear communication complexity. In *TCC 2008* (2008), Springer (LNCS 4948), 213–230.
5. Canetti, R. Security and composition of multiparty cryptographic protocols. *J. Cryptol.* 13, 1 (2000), 143–202.
6. Canetti, R. Universally composable security: A new paradigm for cryptographic protocols. In the *42nd FOCS* (2001), 136–145.
7. Canetti, R., Herzberg, A. Maintaining security in the presences of transient faults. In *CRYPTO 94* (1994), Springer-Verlag (LNCS 839), 425–438.
8. Canetti, R., Lindell, Y., Ostrovsky, R., Sahai, A. Universally composable two-party and multi-party computation. In the *34th STOC* (2002), 494–503. <http://eprint.iacr.org/2002/140>.
9. Chaum, D., Crépeau, C., Damgård, I. Multi-party unconditionally secure protocols. In the *20th STOC* (1988), 11–19.
10. Chida, K., Genkin, K., Hamada, K., Ikarashi, D., Kikuchi, R., Lindell, Y., Nof, A. Fast large-scale honest-majority MPC for malicious adversaries. In *CRYPTO 2018* (2018), Springer (LNCS 10993), 34–64.
11. Cleve, R. Limits on the security of coin flips when half the processors are faulty. In the *18th STOC* (1986), 364–369.
12. Damgård, I., Nielsen, J. Scalable and unconditionally secure multiparty computation. In *CRYPTO 2007* (2007), Springer (LNCS 4622), 572–590.
13. Damgård, I., Pastro, V., Smart, N.P., Zakarias, S. Multiparty computation from somewhat homomorphic encryption. In *CRYPTO 2012* (2012), Springer (LNCS 7417), 643–662.
14. Doerner, J., Kondi, Y., Lee, E., Shelat, A. Threshold ECDSA from ECDSA assumptions: The multiparty case. In *IEEE Symposium on Security and Privacy 2019* (2019), 1051–1066.
15. Evans, D., Kolesnikov, V., Rosulek, M. *A Pragmatic Introduction to Secure Multi-Party Computation*. NOW Publishers, 2018.
16. Furukawa, J., Lindell, Y. Two-thirds honest-majority

- MPC for malicious adversaries at almost the cost of semi-honest. In the *26th ACM CCS* (2019), 1557–1571.
17. Gennaro, R., Goldfeder, S. Fast multiparty threshold ECDSA with fast trustless setup. In the *25th ACM CCS 2018* (2018), 1179–1194.
18. Goldreich, O., Micali, S., Wigderson, A. How to play any mental game – A completeness theorem for protocols with honest majority. In the *19th STOC* (1987), O. Goldreich, ed. Volume 2 of *Foundations of Cryptography – Basic Applications* (2004), Cambridge University Press, 218–229.
19. Hastings, M., Hemenway, B., Noble, D., Zdancewic, S. SoK: General purpose compilers for secure multiparty computation. In *IEEE Symposium on Security and Privacy 2019* (2019), 1220–1237.
20. Ion, M., Kreuter, B., Nergiz, E., Patel, S., Saxena, S., Seth, K., Shanahan, D., Yung, M. Private intersection-sum protocol with applications to attributing aggregate Ad conversions. *IACR Cryptology ePrint Archive*, Report 2017 (2017), 738.
21. Ishai, Y., Kilian, J., Nissim, K., Petrank, E. Extending oblivious transfers efficiently. In *CRYPTO 2003* (2003), Springer (LNCS 2729), 145–161.
22. Ishai, Y., Prabhakaran, M., Sahai, A. Founding cryptography on oblivious transfer – Efficiently. In *CRYPTO 2008* (2008), Springer (LNCS 5157), 572–591.
23. Kolesnikov, V., Kumaresan, R., Rosulek, M., Trieu, N. Efficient batched oblivious PRF with applications to private set intersection. In the *23rd ACM CCS* (2016), 818–829.
24. Kolesnikov, V., Schneider, T. Improved garbled circuit: Free XOR gates and applications. In *ICALP 2008* (2008), Springer (LNCS 5126), 486–498.
25. Lapets, A., Jansen, F., Albab, K.D., Issa, R., Qin, L., Varia, M., Bestavros, A. Accessible privacy-preserving web-based data analysis for assessing and addressing economic inequalities. In *COMPASS 2018* (2018), 48:1–48:5.
26. Lindell, Y. Fast secure two-party ECDSA signing. In *CRYPTO 2017* (2017), Springer (LNCS 10402), 613–644.
27. Lindell, Y., Nof, A. Fast secure multiparty ECDSA with practical distributed key generation and applications to cryptocurrency custody. In the *25th ACM CCS* (2018), 1837–1854.
28. Lindell, Y., Pinkas, B. An efficient protocol for secure two-party computation in the presence of malicious adversaries. In *EUROCRYPT* (2007), Springer, 52–78.
29. Nielsen, J.B., Nordholt, P.S., Orlandi, C., Burra, S.S. A new approach to practical active-secure two-party computation. In *CRYPTO 2012* (2012), Springer (LNCS 7417), 681–700.
30. Ostrovsky, R., Yung, M. How to withstand mobile virus attacks. In *10th PODC* (1991), 51–59.
31. Pinkas, B., Rosulek, M., Trieu, N., Yanai, A. SpOT-light: Lightweight private set intersection from sparse OT extension. In *CRYPTO 2019* (2019), Springer (LNCS 11694), 401–431.
32. Pinkas, B., Schneider, T., Zohner, M. Scalable private set intersection based on OT extension. *ACM T. Privacy Sec.* 21, 2:7 (2018), 1–35.
33. Rabin, T., Ben-Or, M. Verifiable secret sharing and multi-party protocols with honest majority. In the *21st STOC* (1989), 73–85.
34. Shamir, A. How to share a secret. *CACM* 22, 11 (1979), 612–613.
35. Shoup, V. Practical threshold signatures. In *EUROCRYPT 2000* (2000), Springer (LNCS 1807), 207–220.
36. Vardi, M. The long game of research. *CACM* 62, 9 (2019), 7.
37. Yao, A. How to generate and exchange secrets. In *27th FOCS* (1986), 162–167.
38. Unbound Tech. (www.unboundtech.com), Sepior (sepior.com), and Curv (www.curv.co).
39. Sharemind, <https://sharemind.cyber.ee>.
40. Duality, <https://duality.cloud>.

Yehuda Lindell (lindell@biu.ac.il) is a professor in the Department of Computer Science at Bar Ilan University, Ramat Gan, Israel, and is the CEO and co-founder of Unbound Tech.

© 2021 ACM 0001-0782/21/1 \$15.00



Watch the author discuss this work in the exclusive *Communications* video. <https://cacm.acm.org/videos/secure-multiparty-computation>

Are U.S. government employees behaving ethically when they stockpile software vulnerabilities?

BY STEPHEN B. WICKER

The Ethics of Zero-Day Exploits—The NSA Meets the Trolley Car

THE MAY 2017 WannaCry ransomware attack caused a great deal of damage across Europe and Asia, wreaking particular havoc with Britain's National Health Service.^a The attack exploited a Microsoft Windows vulnerability that had been discovered

and exploited by the U.S. National Security Agency.⁵ The NSA informed Microsoft of the vulnerability, but only after the NSA had lost control of the assets it had developed to take advantage of the vulnerability. Shortly after the attack Microsoft President and Chief Legal Officer Brad Smith characterized the NSA and CIA's stockpiling of vulnerabilities as a growing problem:

Finally, this attack provides yet another example of why the stockpiling of

vulnerabilities by governments is such a problem. This is an emerging pattern in 2017. We have seen vulnerabilities stored by the CIA show up on WikiLeaks, and now this vulnerability stolen from the NSA has affected customers around the world. Repeatedly, exploits in the hands of governments have leaked into the public domain and caused widespread damage.^b

Smith asserted that stockpiling of vulnerabilities, as opposed to immedi-

a <https://bit.ly/33dMn8d>


b <https://bit.ly/33dPi0L>

ately informing the software vendor, was wrong, in part because of its effects on Microsoft's customers. A national security operative might argue, however, that these same customers enjoyed a greater benefit through increased personal safety. As an example, this operative might point to the Stuxnet worm. Stuxnet took advantage of four Microsoft Windows vulnerabilities to attack a set of centrifuges that were critical to Iran's nuclear program.¹⁴ This highly sophisticated attack, created and delivered by agents of the U.S. and Israeli governments, may have saved the lives of potential targets of the Iranian nuclear program.


An ethical dilemma presents itself: Are U.S. government employees behaving ethically when they stockpile software vulnerabilities? To address this question, I begin by reviewing the nature of these vulnerabilities and the resulting "zero-day" exploits. I then consider whether participation in stockpiling is permissible from an ethical standpoint. This is a difficult problem, as the standard consequentialist arguments on which current policy is based are crippled from the outset by their need to cope with a great deal of uncertainty. Other complications include the alleged inability of decision makers to share the bases for their decisions with the general public, as well as a form of regulatory capture—those in a position to perform the ethical calculus are the same ones who will exploit the vulnerabilities. I argue these issues can be avoided by using a non-consequentialist approach. By creating detailed case studies for the ethical issues in play, computer scientists can develop a technically informed ethical intuition, and be in a better position to assist with policy moving forward.

Bugs, Vulnerabilities, and Exploits

Bugs have plagued computers and computer software since the six- and eight-legged varieties found their way into the electromechanical switches of UNIVAC. The problem continues today in the form of coding errors that lead to unexpected behavior on the part of computer software. Delivered code has been estimated to average from 15 to 50 errors per 1,000 lines across the industry.¹⁰ Through "cleanroom" techniques the number can be brought



Bugs rise to the level of vulnerabilities when they allow third parties to use the software in a manner that the scientist/engineer who wrote the code did not intend.



close to zero, but this is expensive, time-consuming, and usually limited to highly specialized and strictly compartmentalized government projects such as the space shuttle.

Bugs manifest themselves in a wide variety of forms, from the occasional crash to more subtle though potentially more dangerous behavior. Bugs rise to the level of vulnerabilities when they allow third parties to use the software in a manner that the scientist/engineer who wrote the code did not intend. For example, some vulnerabilities may allow a third-party to see information for which he or she is not authorized, while the worst allow a hacker to load and run malware on the machine on which the vulnerabilities reside.⁹ If the software vendor is unaware of a vulnerability in its product, the term "zero-day vulnerability" applies. "Zero-day" refers to the number of days the vendor has been aware of the vulnerability (zero), and thus the ongoing susceptibility of the software to ongoing attacks.⁹

A "zero-day exploit" is an attack that takes advantage of a zero-day vulnerability to compromise data on a target machine or to deliver and run malicious code on that machine. Zero-day exploits generally have two parts: the exploit code that gains access to a machine through a vulnerability, and an often-unrelated payload that is delivered to the machine once the exploit has gained access.⁹

Vulnerabilities in software are found through many means, but most techniques fall under three general headings: white box, gray box, and black box.¹⁷ The white box approach assumes complete access to source code, design specifications, and in some cases the programmers themselves. The black box approach takes the opposite extreme, assuming no knowledge of the internal structure of the software. As one might imagine, gray box attacks fall somewhere in between. In many cases, gray box attacks begin as black box attacks, but become increasingly gray as knowledge of the behavior of the target allows for refinement of the attack.

The most prominent example of a back/gray box attack is "fuzzing," a brute force approach in which the attacker provides overly large or otherwise unanticipated inputs to a program

and then monitors the response.¹⁷ This requires virtually no knowledge of the software beyond what constitutes an unanticipated input. Sutton, Greene, and Amini have likened this technique to “standing back, throwing rocks at the target, and waiting to hear a window break.”¹⁷ As we will see, the Eternal Blue vulnerability that led to the WannaCry attack appears to have fallen into this category.

For any given vulnerability and subsequent exploit, there are many possible timelines, some leading to problems and others not. Consider the following set of possible events and associated times for a given problem (acknowledging that, for some vulnerabilities, one or more of these events may never occur).

- ▶ T_B : Code with Vulnerability Produced
- ▶ T_T : Vulnerability Discovered by Third Party
- ▶ T_G : Vulnerability Discovered by Government Employees
- ▶ T_S : Vulnerability Discovered by the Software Vendor
- ▶ T_P : Patch Developed and Deployed by Software Vendor
- ▶ T_{CP} : All Vulnerable Computers Patched

A zero-day exploit is possible whenever the government or a third party discovers a vulnerability that the vendor of the software has yet to detect. This occurs whenever the following holds:

$$\max [(T_S - T_T), (T_S - T_G)] > 0$$

Given that the development of a patch takes a non-zero amount of time, the minimum window of vulnerability to hacking by a third party is $(T_P - T_T)$. This attack window can be shortened if T_S and thus T_P are moved up in time. It follows that there is room for the government to have a positive impact on software security if it informs the vendor of a vulnerability before the vendor discovers it on its own.^c

One can imagine a host of hypothetical situations based on who discovers what when, supporting a wide variety of arguments and assertions along the

^c The problem remains that T_{CP} may always be in the future. One may wish to refine the definition to cover only a suitable number of patched computers, along the lines of herd immunity in epidemiology.

» key insights

- **Policies that try to balance the benefits of stockpiling zero-day exploits against the threat to the general public will generally fail, as they attempt to balance objectives that are probabilistic, or even incommensurable.**
- **Non-consequentialist ethics offer a better guide to policy making, as it captures our ethical intuition regarding the public risk that many think is inherent in zero-day exploits and, more generally, cyberwarfare.**
- **Public policy that educates the public about stockpiling while reducing general risk will find more favor with the public, and will be more ethically appealing to the practitioner.**

way. Fortunately, there is data that lends credence to some of these hypothetical situations, providing us with points of focus. In a recently released RAND Corporation study, Ablon and Bogart provide a statistical analysis of several hundred actual zero-day vulnerabilities and exploits.¹ There were many interesting conclusions; but for our purposes, the following are particularly on point:

- ▶ In the RAND dataset, exploits and their underlying vulnerabilities had an average life expectancy of 6.9 years after initial discovery. Some 25% of exploits did not survive for more than a year and a half, and another 25% survived for more than 9.5 years.
- ▶ Once an exploitable vulnerability had been found, the median time required to develop a fully functioning exploit was 22 days.

▶ For a given stockpile of zero-day vulnerabilities, approximately 5.7% had been discovered by an outside entity after one year.

In an unrelated study, Bilge and Dumitras examined data from 11 million active hosts to identify files that exploited known vulnerabilities.⁴ They found that after zero-day vulnerabilities became public knowledge, the number of malware variants exploiting them increased between 183 to 85,000 times, while the number of attacks increased between 2 and 100,000 times.

In summary, vulnerabilities can last for a very long time, and the likelihood of two or more parties finding the same vulnerability is small, but **non-zero**. Further, once a vulnerability becomes known, it will be rapidly exploited by a large number of hackers.

To see how these tendencies played out in a specific case, consider the WannaCry attack in further detail. The delivery vehicle in this instance was a vulnerability that had been known and potentially exploited by the NSA but was published to the world at large by the Shadow Brokers in April 2017, and apparently from there made its way into the hands of the North Korean government.⁴ The particular vulnerability at the heart of the attack was found in a Windows transport protocol called Server Message Block (SMB). SMB operates over the Transmission Control Protocol (TCP), supporting read and write transactions between an SMB client and a server. Codenamed “EternalBlue” by the NSA, the vulnerability was probably found through fuzzing; when an SMB message request exceeds the maximum buffer size, the SMB server moves to a state in which the vulnerability can be exploited.⁷

Having learned of (or discovered) EternalBlue, the WannaCry perpetrators used the vulnerability to put target machines in the desired vulnerable state, and then issued a “request data” command that caused an encrypted viral payload to be loaded onto the target machines. The payload included ransomware as well as software that searched for other machines that had the same vulnerability. The ransomware rapidly propagated across the Internet, infecting machines that shared the EternalBlue vulnerability.

The WannaCry ransomware portion of the payload encrypted hard drives, making them inaccessible to their owners, then presented a request for a few hundred dollars in Bitcoins for reversing the operation. It has been estimated that 230,000 computers in over 150 countries were infected in the first day of the attack.² As the attack spread across Europe and Asia, it damaged Britain’s National Health Service, in part because the NHS employees were not in a position to immediately provide the requested Bitcoins. In many cases doctors were blocked from gaining access to patient files, and emergency rooms were forced to divert patients to other facilities.^d In the Essex town of Colchester, the hospital closed

^d <https://www.nytimes.com/2017/05/13/world/asia/cyberattacks-online-security-.html>

down a significant part of its facilities, only accepting patients in “critical or life-threatening situations.”^e

If the attackers learned of the EternalBlue vulnerability from the Shadow Brokers, it is noteworthy that the EternalBlue vulnerability was weaponized in a matter of weeks. Though this seems fast, the timeline is in keeping with the results of the RAND Corporation study. On the other hand, if the attackers knew of the vulnerability beforehand (having discovered it themselves), then this was an example of a “collision,” which the RAND analyses have also shown to happen with a small, but non-zero probability.

Whatever its origin, the attack highlighted a widespread failure on the part of individuals and corporations to patch their computers. Microsoft posted a security patch as part of Microsoft Security Bulletin MS17-010 (critical)^f on Mar. 14, 2017, a full two months before the first WannaCry attack. The inability of some to patch their computers in a timely manner reduces the impact of government disclosure of vulnerabilities to software vendors. We will bear this in mind when considering the balancing tests in the following section.

Is Stockpiling Ethical? The Consequentialist Approach

Consequentialism is a school of ethics that holds that the morality of an act follows exclusively from its consequences.^g The utilitarianism of John Stuart Mill and Jeremy Bentham, usually summarized as holding that an ethical act is one that provides the greatest happiness for the greatest number, is probably the best-known example of consequentialist ethics.³ In determining the greatest good, one must, of course have a happiness metric of some sort by which to select from among a range of actions. Bentham developed a “felicific calculus” that he claimed would determine the

amount of happiness that a given act would bring.³

The computer scientists and philosophers who have weighed in on the question of whether government stockpiling of zero-day vulnerabilities is ethical have, for the most part, adopted a consequentialist approach, and have attempted to craft zero-day policies with the goal of providing the best possible outcome (however defined).^h For example, in “Zero Days, Thousands of Nights,” Ablon and Bogart frame the debate in terms of longevity and collision rate, asserting that these factors determine whether stockpiling is desirable:

*Government agencies, security vendors, and independent researchers have each been trying to determine which zero-days to hold on to and for how long. This generally involves understanding (1) the survival probability and expected lifetime of zero-day vulnerabilities and their exploits (longevity) and (2) the likelihood that a zero-day found by one entity will also be found independently by another (collision rate). **While longevity of a vulnerability may be an obvious choice of desired metric, collision rate is also important, as the overlap might indicate what percentage of one’s stockpile has been found by someone else, and possibly the types of vulnerabilities that may be more or less desirable to stockpile.***¹ (emphasis added.)

Ablon and Bogart are using longevity and collision rate as inputs to a calculus that provides a greatest good: an optimal balance between maintaining a set of offensive capabilities and preventing attacks against one’s own people. They refine the calculus by arguing that if there are multiple vulnerabilities, then the rationale for disclosing a known vulnerability to the software vendor diminishes. They further argue that disclosure makes little sense if vulnerabilities are very hard to find.

If another vulnerability usually exists, then the level of protection consumers gain from a researcher disclosing a vulnerability may be seen as modest, and some may conclude that stockpiling

*zerodays may be a reasonable option. If zero-day vulnerabilities are very hard to find, then the small probability that others will find the same vulnerability may also support the argument to retain a stockpile.*¹

We have already noted that some users do not patch their computers in a timely manner; in the above analysis, the consequent reduced impact of disclosure would also weigh against disclosure, increasing the ethical attraction of stockpiling from a balancing perspective.

Other attempts to find the right balance have led to similar analyses. For example, in “Would a ‘Cyber Warrior’ Protect Us: Exploring Trade-Offs Between Attack and Defense of Information Systems,” Moore, Friedman, and Procaccia adopt a game-theoretic approach that yields a decision process that, they argue, would best protect the public while maintaining a satisfactory offensive capability.¹¹

The U.S. government made an effort to implement a balancing doctrine in the form of the “Vulnerability Equities Process” (VEP). Former White House Cybersecurity Coordinator Michael Daniel described VEP as follows:

*Each such agency then is responsible for designating one or more Subject Matter Experts (“SMEs”) to participate in a discussion convened by the Executive Secretary to arrive at a consensus on whether the vulnerability should be retained by the government or disclosed for patching.*⁶

Daniel asserted that the process was strongly biased toward disclosure of vulnerabilities.

In a Belfer Center discussion paper, Ari Schwartz and Rob Knake criticized the process, noting the process has apparently lapsed at least once.

*While the Obama Administration deserves credit for re-invigorating the process and for demonstrating a clear bias toward disclosure, the fact that the process fell into disuse from when it went into effect in 2010 until the Intelligence Review Group made its recommendations in 2014 is troubling.*¹⁶

Schwartz and Knake went on to recommend that the government “[m]ake public the high-level criteria that will be used to determine whether to disclose to a vendor a zero-day vulnerability in their product, or to retain the vul-

e <https://bit.ly/30hMA8n>

f <https://technet.microsoft.com/en-us/library/security/ms17-010.aspx>

g This definition is sufficient for assessing balancing considerations, and as a contrast to the non-consequentialist discussion to follow. For more details on consequentialism, see Samuel Scheffler (Ed.), *Consequentialism and Its Critics*, Oxford University Press, 1988.


h This is a separate question from that of the ethics of using the vulnerabilities. We will save the question of whether zero-day exploits are ethical for another day, noting that this involves the Pandora’s box of cyber warfare and questions of just war.

nerability for government use.” It is notable that the documents that provide an overview of the VEP were only made public through a FOIA request by the Electronic Frontier Foundation.


In summary, consequentialist arguments assert that U.S. citizens are best served by zero-day exploit policies that balance a current threat (that of identity theft, loss of data, suffering from the hacking of our critical infrastructure, among others) against the mitigation of a future threat (loss of economic dominance or an attack by a foreign power using advanced weaponry).

There are problems with this approach. For example, consider whether there is an *ethical* obligation for the state to provide aid by immediately informing software vendors of vulnerabilities. Consequentialist ethics has an “Equivalence Thesis” that holds that there is no distinction between the failure to aid and actively doing harm.⁸ This follows from the fact that consequentialists focus on outcomes, rather than intent. North Korea may have been the efficient cause of the WannaCry attack, but as Brad Smith noted, the NSA played a role. If one is to invoke balancing arguments, one must accept responsibility when the balance fails. To avoid being complicit with WannaCry-type attacks, government agencies must *immediately* inform software vendors of flaws discovered in their software. This is not necessarily a weakness in the balancing argument, but it does suggest that the practitioner should be willing to take responsibility and explain what happened when preventable attacks occur.

There is another underlying assumption of balancing arguments that is problematic; namely, that a good balance exists at all. In other words, it is assumed that some risk to the public is worthwhile given the corresponding offensive capability obtained through stockpiling vulnerabilities. It may be that no risk is acceptable—the damage done by a single WannaCry-type attack may be far greater than the potential gains from an offensive cyberattack by the U.S.. In the RAND study discussed earlier, Ablon and Bogart acknowledge that “some” may conclude that if there is *any* chance that a vulnerability may



Even if an accurate assessment of longevity and collision rate were possible, it would require a detailed knowledge of the software available only to the programmers themselves.



be found by another party, then that vulnerability should be disclosed to the vendor.

On the other hand, our analysis shows that that the collision rates for zero-day vulnerabilities are non-zero. Some may argue that, if there is any probability that someone else (especially an adversary) will find the same zero-day vulnerability, then the potentially severe consequences of keeping the zero-day private and leaving a population vulnerable warrant immediate vulnerability disclosure and patch. In this line of thought, the best decision may be to stockpile only if one is confident that no one else will find the zero-day; disclose otherwise.¹

Given that we are considering vulnerabilities in software that is in general public use, there is *always* a non-zero probability that an adversary will find a given vulnerability.

There is a further potential problem of incommensurability—there may be no acceptable basis for comparing the potential damage of a WannaCry-type attack to the added safety derived from stockpiling vulnerabilities for later use as offensive weapons. What sort of metrics can be used? Ablon and Bogart proposed the use of longevity and collision rates, but several problems arise immediately. How do we translate longevity into a utility metric? What likelihood of collision is too high? All of this assumes, of course, that longevity and likelihood of collision can actually be determined for a specific vulnerability.

Even if an accurate assessment of longevity and collision rate were possible, it would require a detailed knowledge of the software available only to the programmers themselves. It would further require a knowledge of the resources and skillset of likely attackers that would be known only to certain individuals in security agencies. There may be very little overlap between these two groups, but even if there were, the task of assigning probability metrics to longevity and collision rates for a given vulnerability would involve a great deal of guesswork.

Which brings me to a final problem with the balancing approach, namely the potential for inherent bias: those who are making the balancing decisions are generally the same people who will develop and launch the ex-

plots. The apparent lack of enthusiasm for the Vulnerability Equities Process is a case in point.

In the face of these problems, it is difficult to see how a governmental decision maker can arrive at a demonstrably ethical, well-balanced and objective stockpiling decision.

Is Stockpiling Ethical? The Non-Consequentialist Approach

There is another approach to ethical questions that may, in this instance, provide more clarity. Non-consequentialist ethics assume the rightness or wrongness of a given act cannot be based solely on the consequences of that act.⁸ Sometimes it is not ethical to choose the act that provides the greatest good for the greatest number; we must also look to prerogatives and constraints that appeal to our ethical intuition to get a more complete picture of an ethical obligation. In non-consequentialist studies, ethical intuition is developed through the study of hypothetical cases. The basic idea here is that such situations help to isolate the individual from personal details that may create a bias, providing a less cluttered focus on the ethical issues involved.

As an example, consider the famous case of the runaway trolley car.¹ Assume the driver of a trolley car has lost control, and that the car is now hurtling toward five people who are tied to the tracks. (In these scenarios we must stick to the facts at hand and not start wondering how we arrived at this situation.) A bystander finds herself next to a switch which, if thrown in time, will divert the trolley and save the five people. Unfortunately, the diversion will cause the trolley to run down a sidetrack and kill a single person who happens to be on that track. What is the bystander to do? To do nothing would entail the certain death of five people, while turning the switch

In determining the greatest good, one must have a happiness metric of some sort by which to select from among a range of actions.

would lead to the certain death of a single, otherwise safe individual.

Having given this some thought, most people agree that it would be ethical for the bystander to throw the switch and save the five at the cost of the one.¹⁵ In exploring the nature of this intuition, ethicists have developed the Doctrine of Double Effect: a foreseen, but unintended harm in pursuit of a greater good is ethically permissible, while an intended harm is not. The Doctrine of Double Effect thus acts as a constraint on intended harm, even when the harm may lead to a greater good.⁸

In order to sharpen the contrast between the foreseen and the intended, consider “the transplant case.” A noted surgeon has five patients, all of whom need transplants of various kinds if they are to survive. We may assume the surgeries will be successful and that the patients will thrive if they receive the various organs. Early one morning a strong healthy individual who has the needed organs walks into the surgeon’s office, asking for directions to the nearest fitness center. The surgeon has the skill to harvest the organs and save his five patients. Unfortunately, the harvesting of the organs will kill the strong healthy individual. Should the surgeon take the organs anyway, saving five lives at the cost of the one? Though on its surface, the ethical arithmetic appears identical to that of the trolley car case, in the transplant case most people would say that harvesting the organs is not ethically permitted. The intuition in this case rests on the fact that the death of the one was not only foreseen but was also intended.

In developing hypothetical cases to study the ethics of stockpiling, I adopt two guidelines.

- ▶ The cases must engage with the facts, while bringing those facts closer to home for those less versed in the details of computer hacking. This guideline broadens the discussion, making the issues more accessible.

- ▶ The cases should elide facts that are not ethically dispositive, but may promote bias; for example, facts that appeal to political passions.

With these guidelines in mind, I offer the following, which I call “the electrical generator case.” A manufacturer of electric generators, let’s call it Mac-

ⁱ See Philippa Foot, The Problem of Abortion and the Doctrine of the Double Effect in *Virtues and Vices*, *Oxford Review* 5, 1967. Foot was a noted British philosopher and the granddaughter of the former U. S. President Grover Cleveland. What she referred to as the “tram problem” is now a cottage industry. See, for example, F.M. Kamm, *The Trolley Problem Mysteries* (The Berkeley Tanner Lectures), Oxford University Press, 2015.

roVolt, has come up with an electric generator that is so efficient and so inexpensive that it has become the world standard for generating electricity. The MacroVolt generator is used in laboratories, test facilities, hospitals and schools throughout the world. Unfortunately, the MacroVolt generator relies on a great deal of software, and that software has bugs. A government employee has discovered a vulnerability through which any given generator can be disabled. The laboratories of a particular foreign government, for example, can be disrupted and perhaps destroyed with a few lines of code, greatly postponing the development of weapons by that government. On the other hand, if an enemy agent finds this vulnerability, he or she can cut the electric power to hospitals and other critical infrastructure in our country with equal ease. The ethical dilemma is as follows: should the government employee keep the vulnerability a secret in the hope of using it as a weapon? Or should she tell MacroVolt as soon as possible to prevent an attack by a third party? I think that the potential threat to life and limb through the failure of medical, air traffic control, and other systems that depend on electricity clearly point to the latter.

The Doctrine of Double Effect can clarify this intuition. The potential threat to life and limb caused by the MacroVolt exploit is both foreseeable *and* intended. Any exploit designed to take advantage of the vulnerability is designed to disrupt *any* generator, not just those of a foreign power.

Note the similarity to the WannaCry attack. The damage caused was foreseeable *and* intended. It is certainly true that the NSA developed the EternalBlue exploit for use against a foreign adversary, and not, presumably, the British health care service. But it is also the case that the EternalBlue exploit was intended for use against a vulnerability that the NSA knew to be shared by *all* instantiations of the target Microsoft software, whether used by adversaries, allies, or U.S. citizens. The EternalBlue exploit was intended to cripple *any* user of Microsoft software, as the attack apparently did not distinguish, say, Farsi versions of the software from that used in the U.K.

Now change the scenario slightly

and consider the “research generator case.” Suppose that MacroVolt’s generators are extremely expensive, and only used in government research environments that require a precise and stable power source. For this reason, MacroVolt’s generators are often used in nuclear weapons research, but generally not in commercial or medical environments. Once again, a government employee has discovered a vulnerability through which any given generator can be disabled. Should the government employee keep the vulnerability a secret in the hope of using it as a weapon? Now the decision to stockpile and later exploit the vulnerability seems more ethically permissible. What changed? It seems less ethically problematic that our own government research facilities are taking the risk of stockpiling upon themselves as opposed to allocating the risk to the public at large. This is an example of a core intuition in non-consequentialist ethics; namely, that individuals should not be used as a means to an end.

Non-consequentialist ethics can thus be used to hone our understanding of ethically permissible and non-permissible risk. By creating narratives that put us at a distance from the facts of a situation, we are better placed to engage our ethical intuition. One may conclude from the above that, in the case of vulnerabilities to software in general use, stockpiling is not ethically permissible. But with some efforts to mitigate the risk to the general public, stockpiling becomes permissible.

Conclusion and Further Thoughts

In this article we have taken two basic approaches to evaluating the ethics of stockpiling zero-day exploits. I have argued that the consequentialist approach has significant difficulties, primarily due to problematic underlying assumptions and a need to balance objectives that are probabilistic, or even incommensurable.

The non-consequentialist approach, on the other hand, offers more traction, capturing our ethical intuition regarding the public risk that many think is inherent in zero-day exploits in particular and cyberwarfare in general. Public policy that attempts to educate the public about stockpiling while reducing general risk will find

more favor with the public. Perhaps of equal importance, those who are involved with the stockpiling and the development of exploits will have greater cause to feel they are both defending their country and engaging in demonstrably ethical activity.

Acknowledgments. I thank the Cornell Einaudi Center for the opportunity to present these ideas at the 2018 Workshop on Privacy, Surveillance, and Civil Society. Thanks to Rebecca Slayton, Fred Schneider, and Linda Lader for their insightful comments and to Sarah Wicker for her encouragement, editorial skills, and ethical equilibrium. ■

References

1. Ablon, L. and Bogart, A. Zero Days, Thousands of Nights: The Life and Times of Zero-Day Vulnerabilities and Their Exploits. The RAND Corporation, 2017, Santa Monica, CA, USA.
2. BBC News. Cyber-Attack: Europol says it was unprecedented in scale. May 13, 2017; <http://www.bbc.com/news/world-europe-39907965>.
3. Bentham, J. *An Introduction to the Principles of Morals and Legislation*. London, 1789. Also in *Collected Works*. J.H. Burns and H. L. A. Hart, Eds. Clarendon Press, Oxford, U.K., 1970.
4. Bilge, L. and Dumitras, Y. Before we knew it: An empirical study of zero-day attacks in the real world. In *Proceedings of the 2012 ACM Conf. Computer and Communications Security*, 833–844.
5. Brewster, T. An NSA Cyber weapon might be behind a massive global ransomware outbreak. *Forbes*, May 12, 2017; <https://bit.ly/3l3qwGu>
6. Daniel, M. Heartbleed: Understanding When We Disclose Cyber Vulnerabilities. Whitehouse.gov blog, (April 28, 2014).
7. Islam, A., Oppenheim, N., Thomas, W. SMB exploited: WannaCry use of ‘EternalBlue’. *FireEye*, May 26, 2017; <https://bit.ly/3n3Z2m7>
8. Kamm, F.M. *Intricate Ethics: Rights, Responsibilities, and Permissible Harm*. Oxford University Press, 2007.
9. Libicki, M.C., Ablon, L., Webb, T. The Defender’s Dilemma: Charting a Course Toward Cybersecurity. The RAND Corporation, 2015, Santa Monica, CA, USA.
10. McConnell, S. *Code Complete: A Practical Handbook of Software Construction*. Microsoft Press, Redmond, WA, USA, 2004.
11. Moore, T., Friedman, A., and Proccaccia, A.D. Would a ‘cyber warrior’ protect us: Exploring trade-offs between attack and defense of information systems. In *Proceedings of the 2010 Workshop on New Security Paradigms*. ACM, New York, NY, 2010, 85–94.
12. Nakashima, E. The NSA has linked the WannaCry computer worm to North Korea. *Washington Post*, (June 14, 2017); <https://wapo.st/2SpSPmB>
13. Nakashima, E. and Gregg, A. NSA’s top talent is leaving because of low pay, slumping morale and unpopular reorganization. *Washington Post* (Jan. 2, 2018); <https://wapo.st/30m9Xh9>
14. Ryan, N. Stuxnet attackers used 4 Windows zero-day exploits. *ZDNet*, (Sept. 14, 2010); <http://www.zdnet.com/article/stuxnet-attackers-used-4-windows-zero-day-exploits/>
15. Sandel, M. *Justice: What’s the Right Thing to Do?* Farrar, Straus, and Giroux, New York, NY, 2009, 21.
16. Schwartz, A. and Knake, R. Government’s Role in Vulnerability Disclosure: Creating a Permanent and Accountable Vulnerability Equities Process. Discussion Paper 2016–04. Harvard University, Belfer Center, Cambridge, MA, USA, June 2016.
17. Sutton, M., Greene, A. and Amini, P. *Fuzzing: Brute Force Vulnerability Discovery*. Addison-Wesley, Boston, MA, 2007.

Stephen B. Wicker is a professor of electrical and computer engineering at Cornell University, Ithaca, NY, USA.

research highlights

P. 105

**Technical
Perspective
Deciphering Errors
to Reduce the Cost
of Quantum
Computation**

By Daniel Gottesman

P. 106

**Constant Overhead Quantum
Fault Tolerance with
Quantum Expander Codes**

By Omar Fawzi, Antoine Gropellier, and Anthony Leverrier

P. 115

**Technical
Perspective
SkyCore's
Architecture
Takes It to the 'Edge'**

By Richard Han

P. 116

**SkyCore: Moving Core
to the Edge for Untethered
and Reliable UAV-Based
LTE Networks**

By Mehrdad Moradi, Karthikeyan Sundaresan, Eugene Chai,
Sampath Rangarajan, and Z. Morley Mao

Technical Perspective

Deciphering Errors to Reduce the Cost of Quantum Computation

By Daniel Gottesman

QUANTUM COMPUTERS MAY one day upend cryptography, help design new materials and drugs, and accelerate many other computational tasks. A quantum computer's memory is a quantum system, capable of being in a superposition of many different bit strings at once. It can take advantage of quantum interference to run uniquely quantum algorithms which can solve some (but not all) computational problems much faster than a regular classical computer. Experimental efforts to build a quantum computer have taken enormous strides forward in the last decade, leading to today's devices with over 50 quantum bits ("qubits"). Governments and large technology companies such as Google, IBM, and Microsoft, as well as a slew of start-ups, have begun pouring money into the field hoping to be the first with a useful quantum computer.

However, many hurdles remain before we have large-scale quantum computers capable of the tasks described here. Whereas hardware errors are rare in classical computers, they will be a significant complication for quantum computers, in part because quantum systems are small and therefore fragile, and in part because the act of observing a quantum system collapses it, destroying the superpositions that distinguish quantum from classical. Even a single atom passing by can interact with a qubit, develop a correlation with it, and thereby eliminate the qubit's quantum coherence.

Consequently, quantum error-correcting codes are essential for building large quantum computers, along with fault-tolerant protocols that describe how to perform computations on encoded qubits. The most popular fault-tolerant protocol is based on a family of quantum codes called "surface codes." Surface codes work by arranging the qu-

bits of the computer in two dimensions and imposing local constraints so the encoded information is spread out and can't be accessed or changed without touching many qubits. Surface codes are an example of a broader class of codes known as "low-density parity check" codes, or quantum LDPC codes for short.


Surface codes have many desirable features: they can be easily laid out in two dimensions, they tolerate high error rates, and local constraints are straightforward to check during a computation. Unfortunately, they also require many extra qubits to work, so it is worthwhile to consider other codes. More general LDPC codes have local constraints like surface codes but with more complex connectivity, and some are much more efficient than surface codes. In particular, a fault-tolerant protocol based on a class of codes known as "quantum expander codes" could in principle reduce the qubit cost of fault tolerance by orders of magnitude.

However, in order for a code family to be actually useful, we need a good way of deciphering the information it gives about the errors in the system. In a well-designed quantum error-correcting code, an error will cause some of the local constraints to be violated. The list of unsatisfied constraints is known as the "error syndrome," from which it is possible to deduce the nature of the error. Possible, but not necessarily easy. Determining which error occurred is a computationally hard problem for some codes. Fault tolerance adds an additional complication, since the error syndrome itself might be faulty due to imperfect measurements while the error syndrome is being determined.

Classical LDPC codes have fast syndrome decoding algorithms, but sadly, these algorithms fail for quantum LDPC codes. The reason for this is that

quantum LDPC codes exhibit a uniquely quantum phenomenon known as "degeneracy." Multiple different errors can act the same way on the codewords, which confuses the classical algorithms. A new approach is needed, and in the following paper, the authors, building on earlier work by themselves and others, produce an algorithm that can rapidly deduce the error in a quantum expander code, even when the syndrome is partially incorrect.

The key to making the algorithm work is to consider multiple qubits at a time. Rather than treating each individual qubit separately, the algorithm looks for small groups of qubits that are part of the error; considering sets of qubits as a unit resolves the ambiguity introduced by degeneracy. The authors then use a result about percolation to show that errors appear in only small clusters, meaning many local decisions about errors can be performed independently and even simultaneously. Consequently, not only does the algorithm work but it is highly parallelizable, making it potentially even faster than the algorithms used for syndrome decoding of surface codes.

To see if expander codes are genuinely useful, much more work is needed, however. We need good codes of reasonable size and better ways of performing fault-tolerant algorithms on encoded qubits. We need to better understand how much error expander codes can tolerate and to deal with the requirement for long-range interactions. If these problems can be solved, expander codes will offer an exciting alternative to surface codes for fault tolerance in a large quantum computer. 

Daniel Gottesman is a faculty member at the Perimeter Institute in Waterloo, Ontario, and a Senior Scientist at Quantum Benchmark Inc. in Kitchener, ON, Canada.

Copyright held by author.

Constant Overhead Quantum Fault Tolerance with Quantum Expander Codes

By Omar Fawzi, Antoine Grospellier, and Anthony Leverrier

Abstract

The *threshold theorem* is a seminal result in the field of quantum computing asserting that arbitrarily long quantum computations can be performed on a *faulty* quantum computer provided that the noise level is below some constant threshold. This remarkable result comes at the price of increasing the number of qubits (quantum bits) by a large factor that scales polylogarithmically with the size of the quantum computation we wish to realize. Minimizing the space overhead for fault-tolerant quantum computation is a pressing challenge that is crucial to benefit from the computational potential of quantum devices.

In this paper, we study the asymptotic scaling of the space overhead needed for fault-tolerant quantum computation. We show that the polylogarithmic factor in the standard threshold theorem is in fact not needed and that there is a fault-tolerant construction that uses a number of qubits that is only a constant factor more than the number of qubits of the ideal computation. This result was conjectured by Gottesman who suggested to replace the concatenated codes from the standard threshold theorem by quantum error-correcting codes with a constant encoding rate. The main challenge was then to find an appropriate family of quantum codes together with an efficient classical decoding algorithm working even with a noisy syndrome. The efficiency constraint is crucial here: bear in mind that qubits are inherently noisy and that faults keep accumulating during the decoding process. The role of the decoder is therefore to keep the number of errors under control during the whole computation.

On a technical level, our main contribution is the analysis of the **SMALL-SET-FLIP** decoding algorithm applied to the family of *quantum expander codes*. We show that it can be parallelized to run in constant time while correcting sufficiently many errors on both the qubits and the syndrome to keep the error under control. These tools can be seen as a quantum generalization of the **BIT-FLIP** algorithm applied to the (classical) expander codes of Sipser and Spielman.

1. INTRODUCTION

Quantum computers are expected to offer significant, sometimes exponential, speedups compared to classical computers. For this reason, building a large, universal quantum computer is a central objective of modern science.

Despite two decades of effort, experimental progress has been somewhat slow and the largest computers available at the moment reach a few tens of physical qubits, still quite far from the numbers necessary to run “interesting” algorithms. A major source of difficulty is the extreme fragility of quantum information: storing a qubit is very challenging, but processing quantum information even more so.

Any physical implementation of a quantum computer is unavoidably imperfect because qubits are subject to decoherence and physical gates can only be approximately realized. In order to compute the outcome of an ideal circuit C using imperfect qubits and gates, the idea is to transform C into another circuit C' , which gives the same outcome with high probability, even if its components are noisy. It is common to refer to the gates or wires of the circuit C as *logical* gates or wires and to those of C' as the *physical* ones.

1.1. Fault-tolerant classical computation

The idea of constructing reliable circuits from unreliable components goes back to von Neumann²⁵ and we briefly sketch the construction he proposed. Given an ideal classical circuit C computing a Boolean function, we construct C' by duplicating each wire and each gate m times. For example, suppose we have an AND gate between wires w_1 and w_2 in C . Then, we will associate to the logical wires w_b in C , m physical wires $w_{b,i}$ for $i \in \{1, \dots, m\}$, and the logical AND will be implemented by m physical AND gates between wires $w_{1,i}$ and $w_{2,i}$. Then, the output of C' is defined as the majority applied to the m wires corresponding to the output of C . If the components of C' are perfect, we can see C' as a version of C where each wire is encoded in a simple error-correcting code: the m -repetition code. If the components of C' are now noisy, then the m wires will generally take different values. As each gate can potentially propagate errors, it is important to correct for errors regularly. If we could apply perfect gates, this would be easy: we simply apply a majority vote among the m wires. Interestingly, von Neumann showed the existence of a circuit that reduces errors even with noisy gates and he called it a “restoring organ.” This is done by applying majorities not on all the m wires but on well-chosen subsets using a concentrator; see Pippenger¹⁶ for details. As the probability that the majority of a block of m wires takes the wrong value is exponentially small in m , it is sufficient to

The original version of this paper was published in FOCS 2018.

choose $m = O(\log s)$ to ensure that all the components of the circuit work as expected with high probability. Here, s is the number of gates in the original circuit C . Thus, starting with a circuit C with s gates, the circuit C' has $O(s \log s)$ gates.

It is very natural to ask at this point whether this logarithmic overhead to construct a fault-tolerant circuit is best possible. Instead of using a simple repetition code, we might try to encode our computation using an error-correcting code with better parameters. In fact, it is well-known since Shannon's work¹⁸ that instead of encoding only one bit in m wires, we could encode a number of bits that is linear in m while keeping a comparable error probability. The first difficulty when using more complicated codes is the implementation of gates. This was particularly simple for the repetition code as described earlier: to implement a logical AND gate, it suffices to apply m AND physical gates between disjoint wires. Using the standard terminology used in quantum fault tolerance, we say that the repetition code has a transversal AND. The important property here is that the circuit to implement a logical and gates uses a physical circuit of constant depth and so errors cannot propagate too much. The second difficulty is to design an error reduction procedure using noisy gates for such general codes. In fact, it turns out that this logarithmic overhead is unavoidable as shown in Pippenger et al.¹⁷

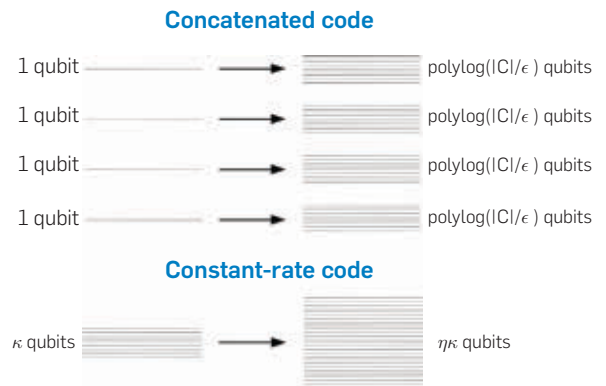
We finally note that for classical computers, fault tolerance is not needed in practice because with the development of the transistor, errors almost never occur.

1.2. Fault-tolerant quantum computation

On the other hand, for quantum computers, fault tolerance is really necessary. For this reason, immediately after Shor discovered his famous factoring quantum algorithm,¹⁹ the search for methods to reduce the effect of decoherence started. Shor himself showed that, perhaps contrary to what one could infer from the quantum no-cloning principle, quantum error-correcting codes do exist²⁰ and he made some steps toward fault tolerance.²¹ A few years later, the celebrated *threshold theorem* was proved. It states that upon encoding the logical qubits within the appropriate quantum error-correcting code, it is possible to transform an arbitrary quantum circuit C into a fault-tolerant one C' , such that even if the components of the circuit C' are subject to noise, below some threshold value it computes the same function as C .¹

Naturally, the fault-tolerant circuit C' will be larger than C . In particular, a number of additional qubits are required and the space overhead, that is, the ratio between the total number of qubits of the fault-tolerant circuit C' and the number of qubits of the ideal circuit C , scales polylogarithmically with the number of gates involved in the original computation. The depth and size overhead are also polylogarithmic, but we focus here on the space overhead. The polylogarithmic factor comes for a reason that is similar to the logarithmic factor in von Neumann's construction for the classical case. The main technique that is used to protect logical qubits is to use concatenated codes. In order to guarantee an overall failure probability ϵ for a circuit C

Figure 1. A natural idea to save on the memory overhead is to encode multiple qubits in the same block.



acting on κ qubits with $|C|$ locations,^a the fault-tolerant version of the circuits needs $O(\log \log (|C|/\epsilon))$ levels of encoding, which translates into a $\text{polylog}(|C|/\epsilon)$ space overhead (see Figure 1).

Although this might seem like a reasonably small overhead, this remains rather prohibitive in practice. As an example, an application of Shor's algorithm to factorize numbers of cryptographic interest would require a few thousand logical qubits, but tens of millions of physical qubits with the best fault-tolerant schemes currently available; see for example, Fowler et al.⁶ and Gidney and Ekerå.⁹ Given the extreme difficulty of controlling a large number of qubits, it is absolutely crucial to try to reduce the overhead of quantum fault tolerance as much as possible. From a computational point of view, it is also a very natural question to determine the optimal overhead required to achieve fault tolerance. As a classical computation is a special case of quantum computation, the previously mentioned logarithmic lower bound for fault-tolerant classical space overhead applies.¹⁷ However, in this context, it is natural to treat classical computations and quantum computations differently. In fact, it is very well motivated in practice to assume that classical computations are error-free but that quantum gates are noisy and to ask what is the minimal possible space overhead that can be achieved in this setting. In this model, building on Gottesman's framework,¹¹ we prove that quantum fault tolerance is possible with constant overhead (see Theorem 1). The main tool that we introduce here in order to achieve this goal is a class of quantum error correcting with good properties. These codes are called *quantum expander codes* and they are constant-rate low-density parity-check quantum codes with a decoding algorithm that can correct typical errors very efficiently even when the syndrome is noisy (see Theorem 3). Before introducing quantum expander codes, we give an overview

^a A location is any point in the circuit that could have an error, so it refers to a quantum gate, the preparation of a qubit in a given state, a qubit measurement, or a wait location if the qubit is not acted upon at a given time step.

of Gottesman's fault-tolerant scheme to motivate the desired properties of the quantum codes.

1.3. Gottesman's scheme

The natural approach to overcome the polylogarithmic barrier had been contemplated for a while, namely to rely on quantum error-correcting codes that encode multiple logical qubits within a block. Ideally, we would like to encode the κ logical qubits needed for the computation within a single quantum error-correcting code of length n with n linear in κ (see Figure 1) and then perform the gates corresponding to the computation within this code and regularly correcting (or more precisely reducing) the errors. However, turning this idea into a full-fledged scheme required much more work. The two main difficulties are to implement fault-tolerantly the logical gates and to correct the errors in a fault-tolerant way. In a breakthrough paper, Gottesman was able to overcome the first difficulty and partially the second one: he showed that polynomial-time computations could be performed with a noisy circuit with only a *constant* overhead provided that a family of quantum codes with good decoding properties was available.¹¹ In fact, this overhead can even be taken arbitrarily close to 1 provided that the physical error is sufficiently small.

We start by briefly describing how Gottesman's construction dealt with the difficulty of implementing the logical gates. One special gate that is used at the beginning of the computation is a preparation gate that prepares a fixed logical qubit state. We need to be able to apply this gate in a fault-tolerant way, that is, such that the number of qubits having an error is under control. In fact, using the technique of gate teleportation, once we are able to fault-tolerantly prepare a small number of fixed logical states, we can implement any logical gate in a fault-tolerant way. In order to achieve this fault-tolerant state preparation, Gottesman uses techniques based on code concatenation. But to keep the associated memory overhead small, we cannot prepare all the κ logical qubits in one shot. Instead, the κ logical qubits of the circuit C are partitioned into $\text{polylog}(\kappa)$ blocks of $\frac{\kappa}{\text{polylog}(\kappa)}$ qubits each and each block is encoded using a constant rate code. Then, the logical circuit C is "serialized" in such a way that a single gate is applied at each time step. In this way, at a given time step, only one gate is applied that acts on at most two logical qubits. Thus, at most two of the blocks are active and the overhead used for applying this gate is polylogarithmic in $\frac{\kappa}{\text{polylog}(\kappa)}$ and thus still linear in κ .

The error correction part of the fault-tolerant scheme is more relevant for the present work. The standard error correction procedure for a quantum error-correcting code is to perform a measurement that outputs a syndrome σ (this is in direct analogy with classical error-correcting codes) and then the decoding algorithm is a classical algorithm taking as input σ and returning an error E that is consistent with this syndrome. This error E is then undone by acting on the quantum systems. We refer to Section 2.2 for formal definitions of quantum error-correcting codes. If the quantum components used for this measurement are noisy, the obtained syndrome will in general be incorrect. One class of codes for which the number of errors in the syndrome stays

under control is low-density parity-check (LDPC) codes. This property is crucial as it ensures that the syndrome measurement circuit is of constant depth and thus errors cannot propagate too much.

Another property that the quantum code needs to have is that it can correct typical errors of size linear in the block-length n . This means that the minimum distance of the code should at least grow with n . And constant rate LDPC codes with minimum distance growing with n are quite difficult to construct. The situation is indeed much more involved than in the classical case where good LDPC codes (constant rate and linear minimum distance) can be found by picking a sparse parity-check matrix at random. In the quantum case, by contrast, the best known constructions display a minimum distance barely above the square-root of the length \sqrt{n} .⁷ But it is not sufficient to have quantum codes with large minimum distance: the decoding algorithm needs to be efficient. In fact, efficient decoding is crucial in the context of fault tolerance: while the decoding algorithm is running, the quantum circuit is waiting for the output of the decoding algorithm and thus errors keep accumulating. Thus, ideally, we would want the decoding to run in constant time that is independent of the number of qubits of the circuit. In addition to the efficiency, another important property that the decoding algorithm should have is that it should come with guarantees even if the observed syndrome σ is itself noisy. In fact, recall that the syndrome measurement circuit will be faulty and so its outcome will have a certain number of errors.

In the present work, we consider *quantum expander codes* introduced in Leverrier et al.¹⁵ obtained by taking the hypergraph product²⁴ of classical expander codes.²² We show that the SMALL-SET-FLIP decoding algorithm introduced in Leverrier et al.¹⁵ does satisfy all these properties. Namely, this algorithm can, in a constant number of time steps, reduce the size of a typical error by a constant fraction even if the observed syndrome is noisy.

We obtain the following general result by using our analysis of quantum expander codes in Gottesman's generic construction.¹¹

THEOREM 1. *For any $\eta > 1$ and $\varepsilon > 0$, there exists $p_\tau(\eta) > 0$ such that the following holds for sufficiently large κ . Let C be a quantum circuit acting on κ qubits, and consisting of $f(\kappa)$ locations for an arbitrary polynomial. There exists a circuit \tilde{C} using $\eta\kappa$ physical qubits, depth $\mathcal{O}(f(\kappa))$, and number of locations $\mathcal{O}(\kappa f(\kappa))$ that outputs a distribution, which has total variation distance at most ε from the output distribution of C , even if the components of C are noisy with an error rate $p < p_{th}$.*

2. QUANTUM EXPANDER CODES

In this section, we first review the construction of classical and quantum expander codes. We then discuss models of noise that are relevant in the context of quantum fault tolerance. We finally introduce the SMALL-SET-FLIP decoding algorithm for quantum expander codes.

2.1. Classical expander codes

A linear classical error-correcting code C of

dimension κ and length n is a subspace of \mathbb{F}_2^n of dimension κ . Mathematically, it can be defined as the κ -dimensional kernel of an $m \times n$ matrix H , called the parity-check matrix of the code: $C = \{x \in \mathbb{F}_2^n : Hx = 0\}$. The minimum distance d_{\min} of the code is the minimum Hamming weight of a nonzero code word: $d_{\min} = \min\{|x| : x \in C, x \neq 0\}$. Such a linear code is often denoted as $[n, \kappa, d_{\min}]$, and a code family has a *constant encoding rate* when $\kappa = \Theta(n)$. An important property for a linear code is the sparsity of H : the code is a *low-density parity-check* (LDPC) code when the rows and columns of H have a weight bounded by a constant.⁸ This is particularly attractive because it allows for efficient decoding algorithms, based on message passing for instance.

An alternative description of a linear code is *via* a bipartite graph known as its *factor graph* $G = (V \cup C, \mathcal{E})$ and defined as follows. The sets V of bits and C of check-nodes have cardinality n and m , respectively, and an edge is present between $v \in V$ and $c \in C$ whenever $H_{c,v} = 1$. In particular, any bipartite graph of constant maximum degree gives rise to an LDPC code. Depending on the description, an error is either a binary word $e \in \mathbb{F}_2^n$ or a subset $E \subseteq V$ whose indicator vector is e . Its corresponding *syndrome* is then either $\sigma(e) := He \in \mathbb{F}_2^m$ or the subset $\sigma(E) := \bigoplus_{v \in E} \Gamma(v) \subseteq C$ corresponding to the odd neighborhood of E in the graph. Here, $\Gamma(v) \subseteq C$ is the set of neighbors of v and the operator \oplus is interpreted as the symmetric difference of sets.

The codes that we will rely on for quantum fault tolerance are the quantum generalization of expander codes, which are the classical codes associated with *expander graphs*, and first considered by Sipser and Spielman.²²

DEFINITION 2 (EXPANDER GRAPH). *Let $G = (V \cup C, \mathcal{E})$ be a bipartite graph with left and right degrees bounded by d_v and d_c , respectively. We say that G is (γ, δ) -expanding if for any subset $S \subseteq A$ (with A is equal to either V or C) with $|S| \leq \gamma|A|$, we have $|\Gamma(S)| \geq (1 - \delta)d_A|S|$.*

Observe that we are requiring two-sided expansion for the graph. Even though only one-sided expansion is required for analyzing classical expander codes, the definition asks for two-sided expansion as this is used for the analysis of quantum expander codes. We note that the existence of (γ, δ) bipartite expanders can be shown via the probabilistic method provided that $d_A > \delta^{-1}$ and γ is a sufficiently small constant. Remarkably, classical expander codes come with an efficient decoding algorithm, BIT-FLIP, that can correct *arbitrary* errors of weight $\Omega(n)$, provided that $\delta < \frac{1}{4}$.²² The strategy behind the BIT-FLIP decoding algorithm is as simple as it can get: given some observed syndrome $\sigma(E)$, simply go through the bits $v \in V$ and flip any bit v if this decreases the syndrome weight, that is, if $|\sigma(E \oplus \{v\})| < |\sigma(E)|$. For a sufficiently expanding factor graph, and provided that the error weight is below γn , it is possible to show that there exist *critical bits* satisfying the condition above, and in fact, the number of such critical bits is linear in the size of E . Going through all the bits once will therefore decrease the syndrome weight by a constant fraction, and decoding will be achieved with logarithmic depth if the algorithm is suitably parallelized. In the context of fault tolerance, where

the syndrome is potentially noisy, the goal changes a little bit because it is not possible in general to correct all errors. In that case, it is sufficient to keep the error weight under control, and this can possibly be achieved by performing a constant number of rounds instead of a logarithmic one. Our present aim is to generalize these results to the quantum setting.

2.2. Quantum error-correcting codes

A quantum error-correcting code encoding κ logical qubits into n physical qubits is a subspace of $(\mathbb{C}^2)^{\otimes n}$ of dimension 2^κ . The *stabilizer* formalism developed by Gottesman¹⁰ allows one to describe a code as the kernel of a linear operator, exactly as in the classical case. A stabilizer group is an Abelian group $\langle g_1, \dots, g_m \rangle$ of n -qubit Pauli operators (n -fold tensor products of single-qubit Pauli operators $X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$, $Y = ZX$, $Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$ and $I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ with an overall phase of ± 1 or $\pm i$) that does not contain $-I$. The associated *stabilizer code* is defined as the common eigenspace of the generators g_1, \dots, g_m with eigenvalue ± 1 . If the generators are independent, then $\kappa = n - m$.

Devising good codes is significantly more complex in the quantum case because of the commutation requirement for the generators. A convenient way to enforce this condition is via the CSS construction,^{3, 23} where the stabilizer generators are either products of single-qubit X -Pauli matrices or products of Z -Pauli matrices. Commutativity should then only be verified between X -type generators (corresponding to products of Pauli X -operators) and Z -type generators, and this can be obtained directly by considering two classical linear codes C_x and C_z of length n with parity-check matrices H_x and H_z satisfying $H_x \cdot H_z^T = 0$. The generators of the stabilizer are of the form g_i^X , and g_i^Z is defined as

$$g_i^X = \bigotimes_{j: \text{s.t. } (H_z)_{ij}=1} X_j, \quad g_i^Z = \bigotimes_{j: \text{s.t. } (H_x)_{ij}=1} Z_j,$$

where X_j denotes the X Pauli operator applied to the j th factor, and where identity operators are omitted. The resulting quantum code has length n and encodes $\kappa = \dim C_x + \dim C_z - n$ logical qubits. Its minimum distance d_{\min} is defined in analogy with the classical case as the minimum Hamming weight of a Pauli operator mapping a code word to an orthogonal one. For the CSS code, one has $d_{\min} = \min(d_x, d_z)$ where $d_x = \min\{|E| : E \in C_x \setminus C_z^\perp\}$ and $d_z = \min\{|E| : E \in C_z \setminus C_x^\perp\}$, where the dual code C_x^\perp consists of words orthogonal to all words of C_x . Note that d_x can be larger than the minimum distance of the classical code C_x as we only consider the weight of code words in C_x that are not in C_z^\perp . In fact, for quantum LDPC codes, the minimum distance of the classical C_x will be bounded by a constant because the condition $H_x \cdot H_z^T = 0$ implies that the rows of H_z , which have a constant weight by the LDPC condition, are in C_x . As such, to construct interesting quantum LDPC codes, it is crucial to use the condition $E \notin C_z^\perp$. The reason the bistrings in C_z^\perp should not be considered as errors is that the corresponding X -type Pauli operators are in the stabilizer group and thus do not affect the state. Two Pauli X -type operators (e.g., errors) that are related by a Pauli X -type operator whose support is given by

an element in C_z^\perp are called *equivalent*. We say that $CSS(C_x, C_z)$ is a $[[n, \kappa, d_{\min}]]$ quantum code.

Even if the CSS framework simplifies matters a little bit, it remains nontrivial to find interesting codes subjected to the condition $H_x \cdot H_z^T = 0$. The hypergraph product code construction introduced by Tillich and Zémor gives a general method to turn a pair of *arbitrary* linear codes into a quantum CSS code.²⁴ In particular, starting with a classical code C with parity-check matrix H and a biregular (γ, δ) -expanding factor graph with vertex set $A \cup B$ (of size $n_A + n_B$) and left and right degrees d_A and d_B (satisfying $d_A \leq d_B$), one obtains a CSS code called *quantum expander code* with parity-check matrices H_x and H_z given by

$$H_x = (I_{n_A} \otimes H, H^T \otimes I_{n_B}),$$

$$H_z = (H \otimes I_{n_A}, I_{n_B} \otimes H^T).$$

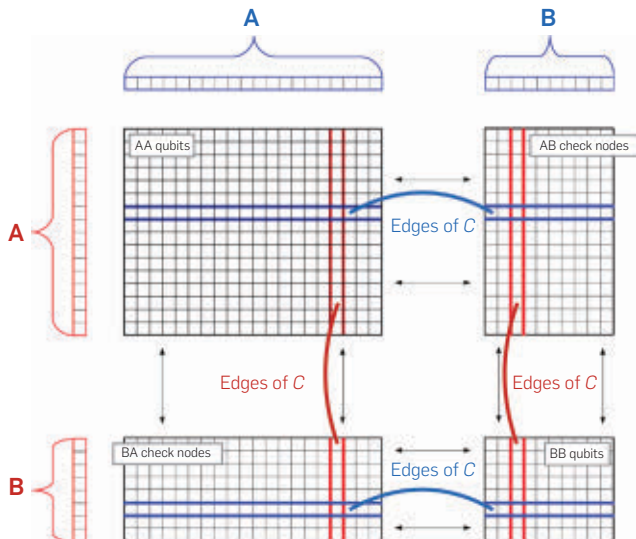
We illustrate this construction in Figure 2.

Quantum expander codes are LDPC with generators of weight $d_A + d_B$ and qubits involved in at most $2d_B$ generators, and they admit parameters $[[n, \kappa, d_{\min}]]$ with $\kappa = \Theta(n), d_{\min} = \Theta(\sqrt{n})$, provided that the expansion satisfies $\delta < \frac{1}{2}$.

2.3. Noise models

In the context of quantum fault tolerance, we are interested in modeling noise occurring during a quantum computation. In the circuit model of quantum computation, the effect of noise is to cause faults occurring at different locations of the circuit: on the initial state and ancillas, on gates (either active gates or storage gates) or on measurement gates. We refer to this model as *basic model* for fault tolerance. The main idea to perform a computation in a fault-tolerant manner is then to encode the logical qubits with a quantum

Figure 2. An illustration of quantum expander codes. Starting with a bipartite expander graph between the vertex sets A and B , the quantum expander code is defined by two bipartite graphs: H_x between the set of qubit nodes $(A \times A) \cup (B \times B)$ and the check nodes $A \times B$ and H_z between the qubit nodes $(A \times A) \cup (B \times B)$ and the check nodes $B \times A$.



error-correcting code, replace the locations of the original circuit by gadgets applying the corresponding gate on the encoded qubits, and interleave the steps of the computation with error correction steps. In general, it is convenient to abstract away the details of the implementation and consider a *simplified model* of fault tolerance where one is concerned with only two types of errors: errors occurring at each time step on the physical qubits, and errors on the results of the syndrome measurement. The link between the basic and the simplified models for fault tolerance can be made once a specific choice of gate set and gadgets for each gate is made. This is done for instance in Section 7 of Gottesman.¹¹ In other words, the simplified model of fault tolerance allows us to work with quantum error-correcting codes where both the physical qubits and the check nodes are affected by errors.

As usual in the context of quantum error correction, we restrict our attention to Pauli-type errors acting on the set V of qubits because the ability to correct all Pauli errors of weight t implies that arbitrary errors of weight t can be corrected. In particular, one only needs to address X - and Z -type errors because a Y -error corresponds to simultaneous X - and Z -errors. Therefore, we think of an error pattern on the qubits as a pair (E_x, E_z) of subsets of the set of qubits V . This should be interpreted as Pauli error X on all qubits in $E_x \setminus E_z$, error Y on $E_x \cap E_z$ and error Z on $E_z \setminus E_x$. In the case of a CSS code, the syndrome associated to this error pattern should be $(\sigma_x(E_x), \sigma_z(E_z))$ but errors will also affect the syndrome extraction, leading to an observed syndrome (σ_x, σ_z) given by

$$\sigma_x := \sigma_x(E_x) \oplus D_x, \quad \sigma_z := \sigma_z(E_z) \oplus D_z,$$

where the error on the syndrome consists of two classical strings (D_x, D_z) , which are subsets of the sets C_x and C_z of check nodes, whose values have been flipped.

How to properly model the effect of noise in a quantum computer is a delicate question. In particular, the assumption of independence of errors affecting distinct qubits is not well justified because the topology of the quantum circuit will generally create correlations between errors. For this reason, a particular reasonable approach suggested by Gottesman consists in only making the assumption that the probability of an error decays exponentially with its weight.¹¹ The relevant error model for the pair (E_x, D_x) is the *local stochastic noise model* with parameters (p, q) defined by requiring that for any $F \subseteq V$ and $G \subseteq C_x$, the probability that F and G are part of the qubit and syndrome errors, respectively, is bounded as follows:

$$\mathbb{P}[F \subseteq E_x, G \subseteq D_x] \leq p^{|F|} q^{|G|}.$$

The error model is exactly the same for the pair (E_z, D_z) . Note that, as the decoding algorithm we use does not take into account correlations between X and Z errors, the joint distribution between (E_x, D_x) and (E_z, D_z) will not affect the analysis.

2.4. The SMALL-SET-FLIP decoding algorithm

If the syndrome extraction is noiseless, a decoder is given the pair (σ_x, σ_z) of syndromes and should return a pair of

errors (\hat{E}_X, \hat{E}_Z) such that $E_X \oplus \hat{E}_X \in \mathcal{C}_Z^\perp$ and $E_Z \oplus \hat{E}_Z \in \mathcal{C}_X^\perp$. In that case, the decoder outputs an error equivalent to (E_X, E_Z) , and we say that it succeeds.

A natural approach to perform error correction (in the noiseless syndrome case) would be to directly mimic the classical BIT-FLIP decoding algorithm analyzed by Sipser and Spielman, that is try to apply X -type (or Z -type) correction to qubits when it leads to a decrease of the syndrome weight. Unfortunately, in that case, there are error configurations of constant weight that cannot be corrected in this way. This led Leverrier et al.¹⁵ to introduce the SMALL-SET-FLIP strategy that we describe next.

Focusing on X -type errors for instance, and assuming that the syndrome $\sigma = \sigma_X(E)$ is known, the algorithm cycles through all the X -type generators of the stabilizer group (i.e., the rows of H_Z), and for each one of them, it determines whether there is an error pattern contained in the generator that decreases the syndrome weight. Assuming that this is the case, the algorithm applies the error pattern (choosing the one maximizing the ratio between the syndrome weight decrease and the pattern weight, if there are several). The algorithm then proceeds by examining the next generator. Because the generators have constant weight $d_A + d_B$, there are $2^{d_A + d_B} = \mathcal{O}(1)$ possible patterns to examine for each generator.

Before describing the algorithm more precisely, let us introduce some additional notations. Let \mathcal{X} be the set of subsets of V corresponding to X -type generators: $\mathcal{X} = \{\text{Supp}(g_i^X) : i \in [m]\} \subseteq \mathcal{P}(V)$, where $\mathcal{P}(V)$ is the power set of V . Here, m denotes the number of X -type generators, and $\text{Supp}(g_i^X)$ denotes the subset of qubits on which g_i^X acts nontrivially. The indicator vectors of the elements of \mathcal{X} span the dual code \mathcal{C}_Z^\perp . The condition for successful decoding of the X -type error E is that E equivalent to the output of the decoding algorithm \hat{E} , i.e., there exists a subset $X \subset \mathcal{X}$ such that $E \oplus \hat{E} = \bigoplus_{x \in X} x$. At each step, the SMALL-SET-FLIP algorithm tries to flip a subset of $\text{Supp}(g_i^X)$ for some generator g_i^X , which decreases the syndrome weight $|\sigma|$. In other words, it tries to flip some element $F \in \mathcal{F}_0$ such that $\Delta(\sigma, F) > 0$ where:

$$\begin{aligned} \mathcal{F}_0 &:= \{F \subseteq g_i^X : i \in [m]\}, \\ \Delta(\sigma, F) &:= |\sigma| - |\sigma \oplus \sigma_X(F)|. \end{aligned} \quad (1)$$

The SMALL-SET-FLIP decoding algorithm consists of two iterations of Algorithm 1 below: it first tries to correct X -type errors by examining the corresponding syndrome $\sigma_X(E_X)$, and then, it is applied a second time (exchanging the roles of X and Z) to correct Z -type errors. The idea of applying the same decoder twice, to correct first X -type errors, and then Z -type errors, is particularly natural when considering a CSS code. Note that this is a suboptimal strategy in general because both types of errors could be correlated, but this will be sufficient for our purpose and this significantly simplifies the exposition.

Algorithm 1: SMALL-SET-FLIP for noiseless syndrome.

INPUT: a syndrome $\sigma = \sigma_X(E) \subseteq C_X$, corresponding to an unknown X -type error pattern $E \subseteq V$

OUTPUT: $\hat{E} \subseteq V$, a guess for the error pattern

SUCCESS: if $E \oplus \hat{E} = \bigoplus_{x \in X} x$ for $X \subseteq \mathcal{X}$, i.e., E and \hat{E} are equivalent errors

```

 $\hat{E}_0 = 0; \sigma_0 = \sigma; i = 0$ 
while  $(\exists F \in \mathcal{F}_0 : \Delta(\sigma_i, F) > 0)$  do
     $F_i = \arg \max_{F \in \mathcal{F}_0} \frac{\Delta(\sigma_i, F)}{|F|}$ 
     $\hat{E}_{i+1} = \hat{E}_i \oplus F_i$ 
     $\sigma_{i+1} = \sigma_i \oplus \sigma_X(F_i)$  //  $\sigma_{i+1} = \sigma_X(E \oplus \hat{E}_{i+1})$ 
     $i = i + 1$ 
end while
return  $\hat{E}_i$ 

```

Leverrier et al.¹⁵ studied the decoding algorithm SMALL-SET-FLIP and showed that it corrects arbitrary *qubit* errors of size $\mathcal{O}(\sqrt{n})$ for quantum expander codes (when the syndrome extraction is noiseless) provided that the expansion of the graph satisfies $\delta < \frac{1}{6}$.

This analysis was extended to the case of random errors (either independent and identically distributed, or local stochastic) provided that the syndrome extraction is performed perfectly and under a stricter condition on the expansion of the graph.⁵ More precisely, for quantum expander codes with an expansion $\delta < \frac{1}{8}$, there exist a probability $p_0 > 0$ and constants C, C' such that if the noise parameter on the qubits satisfies $p < p_0$, the SMALL-SET-FLIP decoding algorithm described above runs in time linear in the code length and corrects a random error with probability at least $1 - Cn(\frac{p}{p_0})^{C'\sqrt{n}}$.

The analysis of the decoding algorithm is inspired by the work of Kovalev and Pryadko¹⁴ who studied the behavior of the maximum likelihood decoding algorithm (that has exponential running time in general). We represent the set of qubits as a graph $\mathcal{G} = (V, \mathcal{E})$ called *adjacency graph* where the vertices correspond to the qubits of the code and two qubits are linked by an edge if there is a stabilizer generator that acts on the two qubits. The approach is then to show that provided the vertices E corresponding to the error do not form large *connected* subsets, the error can be corrected by the decoding algorithm. How large the connected subsets are allowed to be is related to the minimum distance of the code for the maximum-likelihood decoder or to the maximum size of correctable errors for more general decoders. This naturally leads to studying the size of the largest connected subset of a randomly chosen set of vertices of a graph. This is also called site percolation on finite graphs and is a well-studied topic.

In order to analyze the efficient SMALL-SET-FLIP decoding algorithm for quantum expander codes, a slightly more complex notion of connectivity turns out to be relevant. Namely, instead of studying the size of the largest connected subset of E , one studies the size of the largest connected α -subset of E . We say that X is an α -subset of E if $|X \cap E| \geq \alpha|X|$. Note that for $\alpha = 1$, this is the same as X is a subset of E . Then, one shows that, if the probability of error of each qubit is below some threshold depending on α and the degree of \mathcal{G} , then the probability that a random set E has a connected α -subset of size $\Omega(\sqrt{n})$ vanishes as $e^{-\Omega(\sqrt{n})}$. As SMALL-SET-FLIP can correct errors of size $\mathcal{O}(\sqrt{n})$, one concludes that random errors of linear size are corrected with high probability. The

key property of SMALL-SET-FLIP that is used here is its “locality”: at each step, errors on distant qubits are decoded independently. We refer the reader to Fawzi et al.⁵ for the details of the analysis.

3. DECODING WITH A NOISY SYNDROME

In the quantum fault tolerance setting, the syndrome extraction cannot be assumed to be noiseless anymore, and we must consider that the decoding algorithm is fed with noisy syndromes of the form

$$\sigma_x := \sigma_x(E_x) \oplus D_x, \quad \sigma_z := \sigma_z(E_z) \oplus D_z, \quad (2)$$

described by a local stochastic noise model of parameters p and q . As before, we focus on correcting X -type errors so we write E for E_x and D for D_x .

In the case where $D = \emptyset$, we saw in the previous section that the SMALL-SET-FLIP decoding algorithm succeeds in outputting \hat{E} that is equivalent to E provided E is local stochastic with a sufficiently small parameter. In the noisy case $D \neq \emptyset$, the success condition for the decoding algorithm is different. We cannot hope to entirely correct the error because any single qubit error cannot be distinguished from a well-chosen constant weight syndrome bit error. Perhaps surprisingly, we will be using the same SMALL-SET-FLIP decoding algorithm for this noisy case: we keep flipping sets F that decrease the syndrome weight until we cannot do so anymore. In this case, we end up with a final syndrome that is in general not empty, but instead, we prove in Theorem 3 that when $\delta < \frac{1}{16}$, the correction provided by the SMALL-SET-FLIP algorithm leads to a residual error that is local stochastic with controlled parameters.

Before stating the theorem, we note that the fact that we use the same decoding algorithm even with a noisy syndrome is a remarkable feature of SMALL-SET-FLIP for quantum expander codes. In fact, for many other codes such as surface codes, it is necessary not only to change the decoding algorithm but also to repeat the syndrome measurement several times and to apply a more complicated decoding algorithm that depends on all of these outcomes. This property of the SMALL-SET-FLIP algorithm is called *single-shot* in the fault-tolerant quantum computation literature.²

THEOREM 3 (INFORMAL). *There exist constants $p_0 > 0, p_1 > 0$ such that the following holds. Consider a bipartite graph with sufficiently good expansion and the corresponding quantum expander code. Consider random errors (E, D) satisfying a local stochastic noise model with parameter $(p_{\text{phys}}, p_{\text{synd}})$ with $p_{\text{phys}} < p_0$ and $p_{\text{synd}} < p_1$. Let \hat{E} be the output of the SMALL-SET-FLIP decoding algorithm on the observed syndrome. Then, except for a failure probability of $e^{-\Omega(\sqrt{n})}$, the remaining error $E \oplus \hat{E}$ is equivalent to E_{ls} that has a local stochastic distribution with parameter $p_{\text{synd}}^{\Omega(1)}$.*

In the special case where the syndrome measurements are perfect, that is, $p_{\text{synd}} = 0$, the statement guarantees that for a typical error of size at most $p_0 n$, the SMALL-SET-FLIP algorithm finds an error that is equivalent to the error that occurred. If the syndrome measurements are noisy, then

we cannot hope to recover an equivalent error exactly, but instead we can control the size of the remaining error $E \oplus \hat{E}$ by the amount of noise in the syndrome measurements. In particular, for any qubit error rate below p_0 , the decoding operation reduces this error rate to be $p_{\text{synd}}^{\Omega(1)}$ (our choice of p_0 will be such that $p_{\text{synd}}^{\Omega(1)} \ll p_0$). This criterion is sufficient for fault-tolerant schemes as it ensures that the size of the qubit errors stay bounded throughout the execution of the circuit. The proof of this theorem consists of two main parts: analyzing arbitrary errors of weight $\mathcal{O}(\sqrt{n})$ and then exploiting percolation theory to analyze stochastic errors of linear weight.

3.1. Sketch of the analysis

The SMALL-SET-FLIP decoding algorithm proceeds by trying to flip small sets of qubits so as to decrease the weight of the syndrome, and the main challenge in its analysis is to prove the existence of such a small set F . In the case where the observed syndrome is error free, Leverrier et al.¹⁵ and Fawzi et al.⁵ relied on the existence of a “critical generator” to exhibit such a set of qubits. This approach, however, only yields a *single* such set F , and when the syndrome becomes noisy, nothing guarantees anymore that flipping the qubits in F will result in a decrease of the syndrome weight and it becomes unclear whether the decoding algorithm can continue. Instead, in order to take into account the errors on the syndrome measurements, we would like to show that there are *many* possible sets of qubits F that decrease the syndrome weight. In order to establish this point, we consider an error E of size below the minimum distance and we imagine running the SMALL-SET-FLIP decoding algorithm *without errors on the syndrome*. The algorithm gives a sequence of small sets $\{F_i\}$ to flip successively in order to correct the error. In other words, we obtain the following decomposition of the error, $E = \bigoplus_i F_i$ (note that the sets F_i might overlap). The expansion properties of the graph guarantee that there are very few intersections between the syndromes $\sigma(F_i)$. This ensures that a linear number of these F_i 's can be flipped to decrease the syndrome weight at the current step. More formally, one can prove the following statement.

PROPOSITION 4. *There exist constants c_1, c_2, γ_0 such that the following statement holds. Suppose the current error E satisfies $|E| \leq \gamma_0 \sqrt{n}$ and let $\tilde{\sigma} = \sigma_x(E) \oplus D$, then there exists $\mathcal{F}^* \subseteq \mathcal{F}_0$ such that:*

1. $\Delta(\tilde{\sigma}, F) \geq (\frac{1}{2} - 8\delta) |\sigma_x(F)|$ for all $F \in \mathcal{F}^*$,
2. $\sum_{F \in \mathcal{F}^*} |\sigma_x(F)| \geq c_1 |\sigma_x(E)| - c_2 |D|$.

With this, provided that the syndrome of the current error is still large compared to the number of errors D on the syndrome, there will remain some $F \in \mathcal{F}^*$ that can be flipped in order to decrease the syndrome weight and the SMALL-SET-FLIP algorithm can continue. This guarantees then when running the algorithm, the size of the residual error $E \oplus \hat{E}$ can be upper bounded by $c|D|$, for some constant c .

In order to analyze random errors of linear weight, we use percolation theory for α -connected sets similar to the

noiseless syndrome case described in the previous section. The main difference is that we use the *syndrome adjacency graph* of the code, which is similar to the adjacency graph except that we also include check nodes as vertices. This is in order to ensure the “locality” of the decoding algorithm with respect to this graph, implying that each cluster of the error is corrected independently of the other ones. Using the fact that clusters are of size bounded by $\mathcal{O}(\sqrt{n})$, the result on low weight errors shows that the size of $E \oplus \hat{E}$ is controlled by the syndrome error size. In order to show that the error after correction is local stochastic, a more delicate analysis is needed. For this, we introduce the notion of witness to assign residual qubit errors to neighboring syndrome errors. We refer to Fawzi et al.⁴ for details.

3.2. Parallelizing SMALL-SET-FLIP

We established that at each step of Algorithm 2.4, there are many possible flips F that decrease the syndrome weight. We already exploited this property to handle a noisy syndrome, but it can also be used to parallelize the decoding algorithm. In fact, we can now flip several of these small sets F *simultaneously*. However, we have to pay attention to the fact that the sets $\sigma_X(F)$ could intersect. In order to avoid that, we introduce a coloring of the X -type generators: if g_1 and g_2 have the same color, then for any $F_1 \subseteq \Gamma_X(g_1)$ and $F_2 \subseteq \Gamma_X(g_2)$: $\sigma_X(F_1) \cap \sigma_X(F_2) = \emptyset$. It is simple to show that the set C_Z of all the X -type generators can be partitioned using a constant number χ of color classes $C_Z = \cup_{k=1}^{\chi} C_Z^k$.

This leads to Algorithm 2 that is a parallelized version of Algorithm 1 where we flip all the small sets that decrease the syndrome weight sufficiently and that have the same color. Let us discuss the stopping condition for this parallelized decoding algorithm. The natural stopping condition (which is not exactly the one used in Algorithm 2) here would be similar to the sequential version: when no more flips decrease the syndrome weight. As one can show that the syndrome weight decreases by a constant fraction at each step, the number of steps for this algorithm would be of order and $\mathcal{O}(\log n)$ we obtain the same result as in Theorem 3: the residual error is local stochastic with parameter only depending on p_{synd} and not on the size of the initial error. Instead, in Algorithm 2, we apply a fixed number of steps f_0 , where f_0 is a well-chosen constant that depends on the degrees and expansion parameters of the expander graph. This allows the decoding algorithm to run in constant time, which is important for fault tolerance if we do not assume that classical computations are instantaneous. But the price to pay is that the residual error will not only depend on syndrome error rate p_{synd} but also on the qubit error rate p_{phys} . In particular, even if the syndrome was perfect, this algorithm would only reduce the size of the error but not completely correct it. This is however good enough in the context of fault tolerance. We refer the reader to Grospellier¹² for more details.

Algorithm 2: Parallel SMALL-SET-FLIP decoding algorithm.

INPUT: $\tilde{\sigma} \subseteq C_X$ a syndrome where $\tilde{\sigma} = \sigma_X(E) \oplus D$ with $E \subseteq V$ an error on qubits and $D \subseteq C_X$ an error on the syndrome

OUTPUT: $\hat{E} \subseteq V$, a guess for the error pattern

```

 $\hat{E}_0 = \emptyset$ ;  $\tilde{\sigma}_0 = \tilde{\sigma}$ 
for  $i \in \llbracket 0; f_0 - 1 \rrbracket$  do //  $f_0$  is a parameter
   $\kappa = i \bmod \chi$  // current color
  in parallel for  $g \in C_Z^\kappa$  do
    if  $\exists F \subseteq \Gamma_Z(g), \Delta(\tilde{\sigma}_i, F) \geq (\frac{1}{2} - 8\delta) |\sigma_X(F)|$  then
       $F_g =$  arbitrary such  $F$ 
    else
       $F_g = \emptyset$ 
    end if
  end parallel for
   $F_i = \bigoplus_{g \in C_Z^\kappa} F_g$ 
   $\hat{E}_{i+1} = \hat{E}_i \oplus F_i$ 
   $\tilde{\sigma}_{i+1} = \tilde{\sigma}_i \oplus \sigma_X(F_i)$  //  $\tilde{\sigma}_{i+1} = \sigma_X(E \oplus \hat{E}_{i+1}) \oplus D$ 
end for
return  $\hat{E}_i$ 

```

THEOREM 5. *There exist constants $p_0 > 0, p_1 > 0$ such that the following holds. Suppose the pair (E, D) satisfies a local stochastic noise model with parameter $(p_{\text{phys}}, p_{\text{synd}})$ where $p_{\text{phys}} < p_0$ and $p_{\text{synd}} < p_1$. Then, there exists an event succ that has probability $1 - e^{-\Omega(\sqrt{n})}$ and a random variable E_{ls} that is equivalent to $E \oplus \hat{E}$ such that conditioned on $\text{succ}, E_{\text{ls}}$ has a local stochastic distribution with parameter $p_{\text{ls}} = p_0^2$.*

Note that there is nothing special about the square in the expression p_0^2 , and this can be replaced by p_0^c for any $c > 1$. When c increases, the local stochastic parameter p_{ls} of the remaining error gets better but at the cost of a larger number of steps, f_0 .

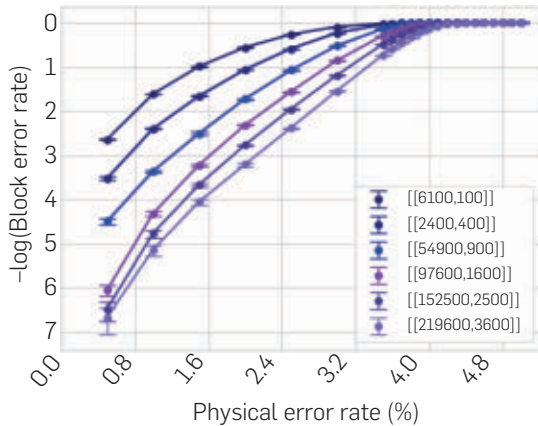
4. CONCLUSION

In this work, we have designed a very efficient decoding algorithm for quantum expander codes that has multiple good properties that are particularly suited for fault-tolerant quantum computation with a small memory overhead. This work should be seen as a theoretical proof of principle and we now mention some limitations of this work and avenues for future research.

A first limitation is that the statements we obtain here are asymptotic in the limit of very large computation. In particular, even though the value of the threshold (i.e., the tolerated error rate) we obtain is a constant, its value is extremely small to be of practical use: an estimate gives 10^{-58} . Part of the explanation is due to the very crude bounds that we obtain via percolation theory arguments. In this work, we have not tried to optimize the value of the threshold and have instead tried to simplify the general scheme as much as possible. As shown in Figure 3, numerical simulations¹³ suggest nevertheless that the threshold value for expander codes could be comparable to the best constructions based on concatenating surface codes.

Another limitation is in the geometry of quantum expander codes. Measuring the syndrome is simple in the sense that one needs to act on a small number of qubits, but the qubits will in general not be *geometrically* local. Performing gates that are not geometrically local may be significantly harder than nearest neighbor gates for many


Figure 3. Logical error rates after decoding quantum hypergraph product codes of various blocklengths with the SMALL-SET-FLIP algorithm, as a function of the physical error rate for i.i.d. X-Pauli errors. These simulations were done with perfect syndrome (figure from Grouès et al.¹³).



quantum computing architectures. Note that this is in contrast to the surface code for which the syndrome bits can be obtained by performing an operation on four neighboring qubits on a two-dimensional lattice. One interesting (architecture-dependent) question for future research is to quantify to which extent a gain in the encoding rate justifies the additional difficulty to perform gates that are not geometrically local.

A third limitation is that for our analysis to apply, we need bipartite expander graphs with a large (vertex) expansion. One issue is that there is no known efficient algorithm that can deterministically construct such graphs. Although algorithms to construct graphs with large *spectral* expansion are known, they do not imply a sufficient vertex expansion for our purpose. Random graphs will display the right expansion (provided their degree is large enough) with high probability, and it is not known how to check efficiently that a given graph is indeed sufficiently expanding.

ACKNOWLEDGMENTS

We would like to thank Benjamin Audoux, Alain Couvreur, Anirudh Krishna, Vivien Londe, Jean-Pierre Tillich, and Gilles Zémor for many fruitful discussions on quantum codes as well as thank Gottesman for answering questions about his paper.¹¹ OF acknowledges support from the ANR through the project ACOM. AG and AL acknowledge support from the ANR through the QuantERA project QCDA. 

References

1. Aharonov, D., Ben-Or, M. Fault-tolerant quantum computation with constant error rate. *SIAM J. Comput.* 4, 38 (2008), 1207–1282.
2. Bombin, H. Single-shot fault-tolerant quantum error correction. *Phys. Rev. X* 3, 5 (2015), 031043.
3. Calderbank, A.R., Shor, P.W. Good quantum error-correcting codes exist. *Phys. Rev. A* 2, 54 (1996), 1098.
4. Fawzi, O., Grospellier, A., Leverrier, A. Constant overhead quantum fault-tolerance with quantum expander codes. In *Proceedings of the 2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)* (2018), IEEE, 743–754.
5. Fawzi, O., Grospellier, A., Leverrier, A. Efficient decoding of random errors for quantum expander codes. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing* (2018), ACM, 521–534.
6. Fowler, A.G., Mariantoni, M., Martinis, J.M., Cleland, A.N. Surface codes: Towards practical large-scale

- quantum computation. *Phys. Rev. A* 3, 86 (2012), 032324.
7. Freedman, M.H., Meyer, D.A., Luo, F. Z₂-symplectic freedom and quantum codes. *Mathematics of Quantum Computation*. Chapman & Hall/CRC, 2002, 287–320.
8. Gallager, R. Low-density parity-check codes. *IRE Trans. Inform. Theor.* 1, 8 (1962), 21–28.
9. Gidney, C., Ekerå, M. How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits. *arXiv preprint arXiv:1905.09749* (2019).
10. Gottesman, D. *Stabilizer codes and quantum error correction*. PhD thesis, California Institute of Technology (1997).
11. Gottesman, D. Fault-tolerant quantum computation with constant overhead. *Quant. Inform. Comput.* 15–16, 14 (2014), 1338–1372.
12. Grospellier, A. *Constant time decoding of quantum expander codes and application to fault-tolerant quantum computation*. PhD thesis, Inria Paris (2019).
13. Grouès, L., Grospellier, A., Krishna, A., Leverrier, A. Combining hard and soft decoding for hypergraph product codes. *arXiv preprint arXiv:2004.11199* (2020).
14. Kovalev, A.A., Pryadko, L.P. Fault tolerance of quantum low-density parity check codes with sublinear distance scaling. *Phys. Rev. A* 2, 87 (2013), 020304.
15. Leverrier, A., Tillich, J.-P., Zémor, G. Quantum expander codes. In *Proceedings of the 2015 IEEE 56th Annual Symposium on Foundations of Computer Science (FOCS)* (2015), IEEE, 810–824.
16. Pippenger, N. On networks of noisy gates. In *Proceedings of the 26th Annual Symposium on Foundations of Computer Science (SFCS 1985)* (1985), IEEE, 30–38.
17. Pippenger, N., Stamoulis, G.D., Tsitsiklis, J.N. On a lower bound for the redundancy of reliable networks with noisy gates. *IEEE Trans. Inform. Theory* 3, 37 (1991), 639–643.
18. Shannon, C.E. A mathematical theory of communication. *Bell Syst. Tech. J.* 3, 27 (1948), 379–423.
19. Shor, P.W. Algorithms for quantum computation: Discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science* (1994), IEEE, 124–134.
20. Shor, P.W. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A* 4, 52 (1995), R2493.
21. Shor, P.W. Fault-tolerant quantum computation. In *Proceedings of 37th Conference on Foundations of Computer Science* (1996), IEEE, 56–65.
22. Sipser, M., Spielman, D.A. Expander codes. *IEEE Trans. Inform. Theory* 6, 42 (1996), 1710–1722.
23. Steane, A.M. Error correcting codes in quantum theory. *Phys. Rev. Lett.* 5, 77 (1996), 793.
24. Tillich, J.-P., Zémor, G. Quantum LDPC codes with positive rate and minimum distance proportional to the square root of the blocklength. *IEEE Trans. Inform. Theory* 2, 60 (2014), 1193–1202.
25. Von Neumann, J. Probabilistic logics and the synthesis of reliable organisms from unreliable components. *Autom. Stud.*, 34 (1956), 43–98.

Omar Fawzi (omar.fawzi@ens-lyon.fr), Univ Lyon, ENS de Lyon, CNRS, UCBL, LIP Lyon, France.

Antoine Grospellier and Anthony Leverrier ([antoine.grospellier, anthony.leverrier@inria.fr]@inria.fr), Inria, Paris, France.

Copyright held by authors/owners. Publication rights licensed to ACM.

Technical Perspective

SkyCore's Architecture Takes It to the 'Edge'

By Richard Han


THE FOLLOWING SKYCORE paper addresses an exciting use case for Unmanned Aerial Vehicles (UAVs) or drones in which UAVs can act as mobile base stations for the cellular network, flying to areas in the cellular network in order to improve wireless connectivity in those areas. This adaptive capability to patch the network capacity on demand would be useful to address hotspots, such as at sporting venues or other temporary events that have insufficient network capacity, and/or emergency scenarios when parts of the cellular network are incapacitated. The paper uses the Long-Term Evolution (LTE) standard as a case study for providing on-demand adaptive cellular connectivity via UAVs.

The challenge of this work is how to adapt the existing LTE standard to support the concept of UAV-based mobile base stations, especially in the presence of multiple UAVs. In current cellular networks, base stations employ a Radio Access Network (RAN) to communicate with clients, for example, cell phones. Packets are then routed over a high-speed wired network of gateways comprising the Evolved Packet Core (EPC) network to the Internet. The paper observes that current cellular operators typically deploy UAV base stations with an architecture in which the UAVs contain the RAN while the EPC is ground-based. In order to connect the UAV-based RAN to the EPC, the UAV is either tethered via wire to the UAV base stations (limiting their mobility and range) or connected wirelessly to the UAV, exposing EPC communication to the unreliability of the wireless link.

Instead, the authors propose a novel Edge-EPC network architecture called SkyCore in which EPC functionality is pushed into the extreme edge, namely, the UAV itself. This avoids the tethering and wireless unreliability problems noted earlier but introduces two new challenges. First, the UAVs have limited computational

resources. Second, the hierarchical nature of the standard EPC network provides a global view that can manage hand-off of a mobile client from one base station to the next, whereas UAV-based EPC functionality will not have a global view. The authors propose novel solutions to these problems respectively: software refactoring to reduce the EPC's footprint on the UAV; and proactive inter-UAV communication for EPC agents via a new software-defined networking (SDN) control-data interface.

The paper makes the following contributions: First, it builds a real-world prototype of the Skycore system consisting of a two UAV LTE network that seamlessly works with commercial off-the-shelf RANs and mobile LTE clients. Second, the paper shows the feasibility of SkyCore UAVs acting as adaptive LTE-hotspots, providing improved on-demand network capacity to clients. Third, the paper demonstrates the feasibility of Skycore to act as an independent ad hoc LTE network, connecting geographically separated clients through two different UAVs, while also allowing for seamless hands-off. Fourth, the work experimentally shows that when compared to a generic Edge-EPC architecture, SkyCore's enhanced Edge-EPC features lower control plane latencies by an order of magnitude, and lower CPU utilization by a factor of five.

The SkyCore system introduces a new edge-centric cellular network architecture that opens up the possibility for efficiently supporting mobile drone-based base stations in future hotspot and emergency scenarios. Skycore's Edge-EPC architecture also has the virtue that it is not limited to LTE networks and can be generalized to 5G cellular networks and beyond. 

Richard Han is a professor in the Department of Computer Science at the University of Colorado, Boulder, CO, USA.

Copyright held by author.



Association for
Computing Machinery

2018 JOURNAL IMPACT
FACTOR: 6.131

ACM Computing
Surveys (CSUR)

ACM Computing Surveys (CSUR) publishes comprehensive, readable tutorials and survey papers that give guided tours through the literature and explain topics to those who seek to learn the basics of areas outside their specialties. These carefully planned and presented introductions are also an excellent way for professionals to develop perspectives on, and identify trends in, complex technologies.



For further information
and to submit your
manuscript,
visit csur.acm.org

SkyCore: Moving Core to the Edge for Untethered and Reliable UAV-Based LTE Networks

By Mehrdad Moradi, Karthikeyan Sundaresan, Eugene Chai, Sampath Rangarajan, and Z. Morley Mao

Abstract

The advances in unmanned aerial vehicle (UAV) technology have empowered mobile operators to deploy LTE (long-term evolution) base stations (BSs) on UAVs and provide on-demand, adaptive connectivity to hotspot venues as well as emergency scenarios. However, today's evolved packet core (EPC) that orchestrates LTE's radio access network (RAN) faces fundamental limitations in catering to such a challenging, wireless, and mobile UAV environment, particularly in the presence of multiple BSs (UAVs). In this work, we argue for and propose an alternate, radical *edge* EPC design, called SkyCore that pushes the EPC functionality to the extreme edge of the core network—collapses the EPC into a single, lightweight, self-contained entity that is colocated with each of the UAV BS. SkyCore incorporates elements that are designed to address the unique challenges facing such a distributed design in the UAV environment, namely the resource constraints of UAV platforms, and the distributed management of pronounced UAV and UE mobility. We build and deploy a fully functional version of SkyCore on a two-UAV LTE network and showcase its (i) ability to interoperate with commercial LTE BSs as well as smartphones, (ii) support for both hotspot and stand-alone multi-UAV deployments, and (iii) superior control and data plane performance compared to other EPC variants in this environment.

1. INTRODUCTION

Mobile LTE (long-term evolution) networks that are ubiquitous today are deployed after sufficient RF planning in a region. However, the static nature of LTE base station (BS) deployments limits their ability to cater to certain key 5G use cases—surging traffic demands in hotspots (e.g., stadiums and event centers), as well as their availability in emergency situations (e.g., natural disasters), where the infrastructure could itself be compromised. Providing an additional degree of freedom for base stations, namely *mobility*, allows them to break away from such limitations.

UAV driven mobile networks. Advances in unmanned aerial vehicle (UAV) technology have empowered operators to take on-demand, outdoor connectivity to another level, by allowing their base stations to be deployed aurally on UAVs (Figure 1), thereby offering complete flexibility in their deployment and optimization. Mobile operators such as AT&T and Verizon have both conducted trials with LTE base stations mounted on UAVs^{9,8} (helicopter and fixed-wing aircraft, respectively). AT&T also provided LTE network services from its UAV in the aftermath

of hurricane Maria in Puerto Rico last year.³ Further, with the availability of shared access spectrum such as CBRS² in 3.5 GHz, this also opens the door for smaller, greenfield operators to deploy and provide on-demand, private LTE connectivity services without the heavy cost associated with spectrum and deployment.

Limitations of the legacy EPC. A typical mobile cellular network requires the deployment of two essential components: a radio access network (RAN) consisting of multiple base stations (BSs) that provide wide-area wireless connectivity to clients (UEs) and a high-speed, wired core network of gateways (evolved packet core, EPC) that sits behind the RAN and is responsible for all the mobility, management, and control functions, as well as routing user traffic to/from the Internet. Realizing a multi-UAV-driven RAN (BSs deployed on UAVs) with an EPC on the ground or in the cloud is one way to directly apply today's EPC architecture to the UAV environment (as shown in Figure 2). Based on publicly available information,^{3,8,9} this has been the case with the current operator-driven UAV efforts. However, this faces significant

Figure 1. UAV-based LTE networks.

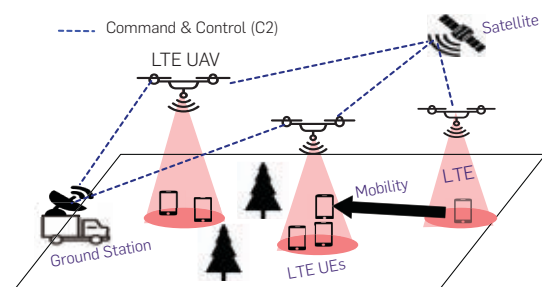
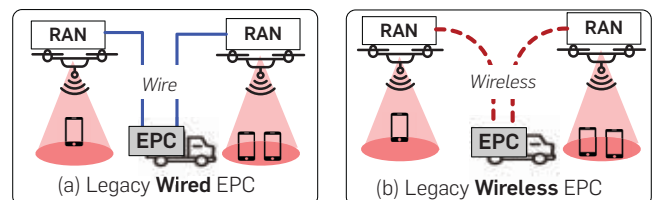


Figure 2. Legacy EPC variants for UAV-based LTE networks.



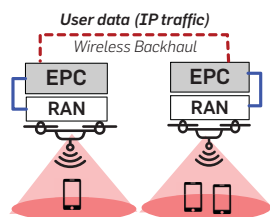
The original version of this paper was published in the *Proceedings of the 2018 ACM MobiCom Conference*.

limitations in delivering real value to this challenging environment. Specifically, although a **tethered setup** (EPC-UAV link being wired, Figure 2a) significantly *limits the UAV's mobility and ability to scale to multiple UAVs*, a **wireless setup** (EPC-UAV link being wireless/mobile, Figure 2b) incurs all the *vagaries of the wireless channel*. For the latter, the choice of the wireless technology becomes critical given that the EPC is responsible for setting up, routing, and tearing down all voice/data bearers. It is essential for the EPC to *reliably reach all the UAVs* wirelessly, such as those that are potentially far away in the presence of non-line-of-sight conditions (e.g., buildings, foliage, etc.). Further, it must deliver sufficient capacity to support the traffic demands in the RAN. It is extremely challenging for a wireless technology, be it lower frequency (sub-6 GHz such as LTE, WiFi, etc.) or higher frequency (mmWave, satellite), to simultaneously satisfy the *needs of range, reliability/robustness, and capacity* that the UAV environment demands from the critical EPC-RAN link.

Core at the edge. Given the fundamental limitations in deploying an EPC on the ground or in the cloud to support a multi-UAV RAN, we advocate for a radical, yet standards-compliant redesign of the EPC, namely the *Edge-EPC* architecture, to suit the UAV environment. As the name suggests, we aim to push the *entire* EPC functionality to the extreme edge of the core network, by collapsing and locating the EPC as a single, lightweight, self-contained entity on each of the UAVs (BSs) as shown in Figure 3. Being completely distributed at the very edge of the network, such an architecture completely eliminates wireless on the critical EPC-RAN path and hence the crippling drawbacks faced by the legacy architecture in this environment.

Although definitely promising at the outset, realizing this radical design is not without its own set of challenges that are unique to the UAV environment. In particular, (i) *Resource-challenged environment*: The compute resources consumed by the numerous network functions in the EPC are appreciable and become a concern when all the EPC functionality is placed into a single node and deployed directly on a UAV platform—the latter being highly resource-challenged to begin with. This could significantly affect both the UAV's operational lifetime and the processing (control and data plane) latency of its traffic (see Figure 4), thereby resulting in a reduced traffic capacity. (ii) *Mobility management*: The hierarchical nature of the legacy EPC architecture, gives a single network gateway (such as the mobility management entity, the MME) a consolidated view of multiple BSs, thereby allowing it to efficiently manage handoffs during

Figure 3. Edge-EPC for UAV-based LTE networks.

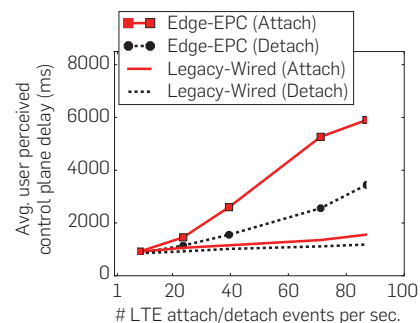


mobility of active UEs as well as tracking/paging mobile UEs that are in idle mode. Mobility of both active (handoffs) and idle UEs (tracking/paging) becomes a critical challenge, when the entire EPC is located at each of the UAVs, thereby restricting their view of events to only those that are local to the UAV. The frequency of such events is further exacerbated by the mobility of the UAVs.

Our proposal: SkyCore. Toward our vision of building *untethered yet reliable UAV-based mobile networks*, we present our novel EPC design, SkyCore. SkyCore embodies the Edge-EPC architecture while introducing two key pillars in its design to address the associated challenges: a complete *software refactoring of the EPC* for compute-efficient deployment on a UAV and a new *inter-EPC communication interface* to enable fully functional operation in a multi-UAV environment. Through *software refactoring*, SkyCore eliminates the distributed EPC interfaces and collapses all distributed functionalities into a single logical entity (agent) by transforming the latter into a series of switching flow tables and associated switching actions. It also reduces control plane signaling and latency by precomputing and storing (in-memory) several key attributes (security keys, QoS profile, etc.) for UEs that can be accessed quickly in real time without any computation. To ensure complete EPC functionality, SkyCore manages mobility right at the edge of the network—it enables a new control/data interface through *software-defined networking (SDN)* to realize efficient *inter-EPC signaling and communication* directly between UAVs. This allows the SkyCore agents on each UAV to *proactively* synchronize their states with each other, thereby avoiding the real-time impact of wireless (UAV-UAV) links on critical control functions—results in fast and seamless handoff of active-mode UEs as well as tracking of idle-mode UEs across multiple UAVs.

Real-world prototype: We have built a complete version of SkyCore on a single board server with a small compute and energy footprint and deployed it on DJI Matrice 600 Pro rotary-wing drones to create a two-UAV LTE network. To the best of our knowledge, this is the first realization of a self-contained Edge-EPC solution that can support a multi-UAV network and is a direct affirmation of SkyCore's design. SkyCore's feasibility and functionality are validated by seamless integration and operation with a commercial LTE RAN (BS) from ip.access and off-the-shelf UEs (Moto G and Nexus smartphones). We demonstrate SkyCore UAVs

Figure 4. Edge-EPC has high overheads on UAV works.



to operate both as LTE hotspots that allow for better UE connectivity to the Internet by extending coverage of a terrestrial LTE network, as well as stand-alone LTE networks for connectivity of geographically separated UEs through two different UAVs (e.g., first responders in emergency scenarios), whereas also allowing for handoffs. Our real-world evaluations of SkyCore and its comparison with a state-of-the-art software EPC (OpenEPC⁶) on UAV clearly showcase SkyCore’s superior performance and scalability—SkyCore provides an order of magnitude lower control plane latencies, incurs 5× lower CPU utilization, and provides data plane rates that currently scale up to a Gbps.

Our two key contributions in this work include the following:

- A novel Edge-EPC solution, SkyCore that can *reliably and scalably* support a multi-UAV LTE network deployment that was not possible earlier
- A real-world implementation and evaluation that showcase both its feasibility and its superior performance

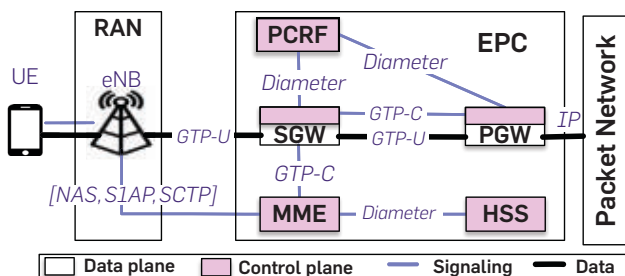
Broader implications: SkyCore’s underlying design is driven by the observation that when connectivity between core network functions, which are on the critical path, is unreliable (wireless and mobile), the merits of pushing functionality to the edge of the network significantly outweigh the associated drawbacks. Hence, although designed for a multi-UAV environment, SkyCore’s design can also benefit other deployments, where distributed critical functions have to communicate over unreliable links (e.g., distributed enterprise RANs). Further, adopting an SDN-based design, SkyCore is equally applicable to future RAN technologies such as 5G and 6G.

2. SKYCORE: DESIGN OVERVIEW

2.1. Background on legacy EPC

Evolved packet core (EPC, Figure 5) is a distributed system of different nodes, each consisting of diverse network functions (NFs) that are required to manage the LTE network. The EPC consists of data and control data planes: the data plane enforces operator policies (e.g., DPI, QoS classes, and accounting) on data traffic to/from the user equipment (UE), whereas the control plane provides key control and management functions such as access control, mobility, and security management. eNodeBs or eNBs (RANs) are grouped into logical serving areas and connected to serving gateways (SGWs). The SGW is connected to an external packet network (PGW). The SGW is connected to an external packet

Figure 5. Legacy EPC architecture.



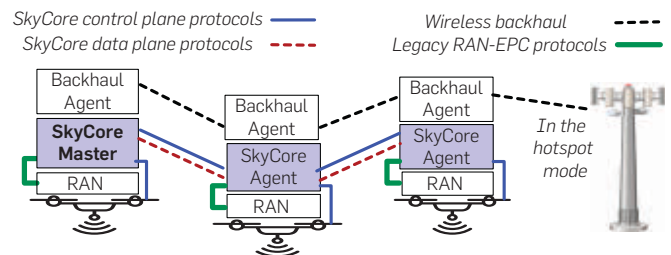
network (e.g., the Internet) via the packet data network gateway (PGW). The PGW enforces most of the data plane policies (e.g., NAT and DPI) and may connect the core to other IP network services (e.g., Web servers). The EPC forwards each UE’s data traffic between the eNodeB and PGW using a separate GTP-U (GPRS tunneling protocol) tunnel. The mobility management entity (MME) is responsible for access control and enforcement, as well as security and mobility functions (e.g., attach/detach and paging/handoff) in conjunction with the HSS (home subscriber server) database and PCRF (policy and charging rules function).

In contrast to legacy EPC, SkyCore adopts the Edge-EPC architecture as shown in Figure 6. SkyCore collapses the entire EPC and pushes it to the edge of our network, namely at each of the UAVs themselves, where it is colocated with the RAN. Although this completely eliminates wireless from the critical path between the EPC and RAN, to address the challenges associated with the Edge-EPC architecture, SkyCore introduces two novel design components, which are briefly explained here:

Software refactoring of the EPC functionality: To reduce its compute footprint on the UAV, SkyCore adopts a software refactoring approach to eliminate distributed EPC interfaces and collapse all distributed functionalities (Figure 5) into a single logical entity. It realizes this by transforming the distributed data plane functions into a series of switching flow tables and associated switching actions (corresponding to functions such as GTP-U encapsulation/decapsulation, charging, etc.). It also reduces control plane signaling and latency by precomputing and storing (in-memory) several key attributes relating to security keys, QoS profile, etc. for UEs that can be accessed locally in real time without any computation.

Efficient inter-EPC communication: With every UAV now running its own EPC agent, even a simple eNB-eNB handoff of an active UE across two UAVs now becomes an inter-MME handoff, which needs to be accomplished across two different EPC agents. SkyCore enables a new control/data interface that allows agents on different UAVs to *proactively* (in the background) synchronize the state of UEs. This bypasses the real-time impact of wireless (UAV-UAV links) on critical control path functions, allowing for seamless handoffs and tracking of idle-mode UEs right at the edge. The HSS equivalent in each SkyCore agent maintains the location (anchoring SkyCore agent) of all UEs in the network. Hence, when an agent sends a UE location update, the agents in other UAVs update their HSS

Figure 6. The SkyCore network architecture.



accordingly. Thus, whenever traffic needs to be sent from a SkyCore agent to a UE located at another UAV, the HSS will reveal the destination SkyCore agent at which the UE is anchored and to whom the traffic has to be routed. The actual routing path taken by the traffic on the mesh backhaul is then determined by SkyCore, with the underlying backhaul topology information made available by a backhaul agent that resides on the UAV.

2.2. Software refactoring of EPC

Each SkyCore agent has a minimalist and UAV-aware SDN-based architecture (Figure 7), consisting of a controller that executes the control functions to process UEs' signaling traffic and to coordinate with other agents, and a switch that processes user data traffic. In the following, we describe six high-level steps that we take to refactor and extend the EPC functionality onto our agent architecture.

Step 1. Decoupling the EPC control and data plane pipelines. One of the main reasons behind the high complexity and overhead of the EPC is its nodes performing mixed control and data plane functions. To make the EPC functionality suitable for UAVs, we first decouple the EPC control and data planes. Among the EPC nodes, the MME, PCRF, and HSS are pure control nodes. Hence, our decoupling does not affect these elements, and only affects the SGW and PGW. The resulting control components from the decoupling are the PGW-C, SGW-C, MME, PCRF, and HSS, and the data elements include the SGW-D and PGW-D (C stands for control and D for data). Although the benefits of decoupling control and data planes have been articulated before,¹⁹ we apply it in the context of UAV networks and enhance it substantially with the following mechanisms.

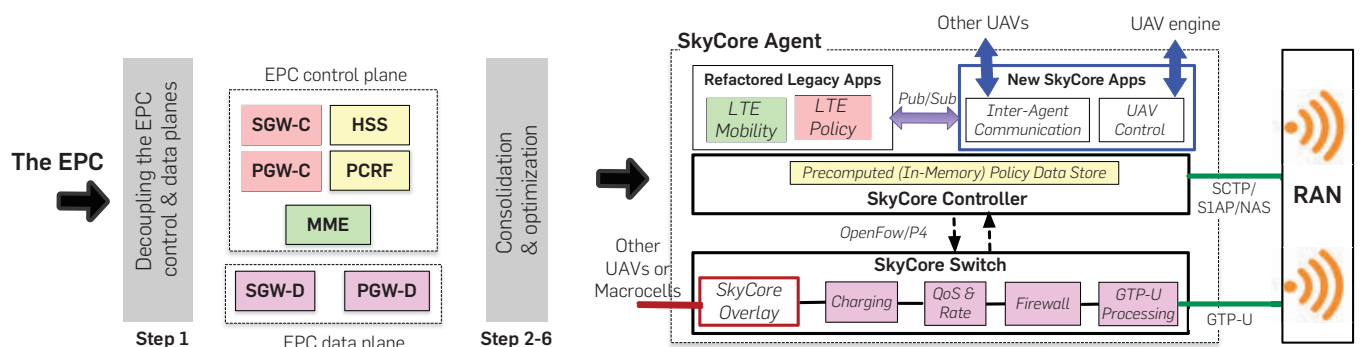
Step 2. Categorizing the functionality of the EPC control plane. Next, we categorize the EPC control nodes based on their high-level functionality. In our decoupled EPC, there are three types of nodes: (1) the SGW-C and PGW-C are responsible for managing QoS policy enforcement on and routing of user data traffic, (2) the MME exchanges signaling traffic with UEs and eNBs, and (3) the PCRF and HSS dynamically generate network security and QoS policies for the other nodes. To compress the EPC functionality, we consolidate the nodes in each category on top of our agent controller and remove the EPC-distributed protocols as follows.

Step 3. Collapsing the SGW-C, PGW-C, and MME into lightweight SDN applications. We extract the internal functions in the SGW-C and PGW-C and refactor them into a single SDN application, LTE Policy Application, on top of the controller. We do the same process for the MME and transform it into LTE Mobility Application. One notable aspect of this consolidation is that we naturally eliminate the complex GTP-C protocol, its six interfaces, and continuous control messages from the core network (Figure 5). This makes the SDN applications extremely lightweight and extensible without hurting their original functionality. Note that these applications still exchange information with each other but through simple local publish-subscribe mechanisms.

Step 4. Eliminating the HSS and PCRF from the core and replacing them with a precomputed policy data store. Next, we focus on the HSS and PCRF that are known to be the source of today's signaling storms in cellular networks.^{5,7} The HSS stores hundreds of database tables containing different UEs' states often on disk. Moreover, it acts as a proxy between the MME and these tables, and performs different types of complex security and location tracking computations. The PCRF often accesses a logical database (sometimes implemented in the HSS) and dynamically generates different QoS and charging policies for UEs. In SkyCore, we completely eliminate these two nodes from our agents and show that dynamic policy generation can be carefully replaced with a precomputed in-memory policy data store (see Figure 9). Precomputation combined with in-memory transactions substantially minimizes the overhead of the core on resource-challenged UAVs. This also removes the complex Diameter protocol (Figure 5) from the core.

Step 5. Adding UAV-specific SDN applications to the core. One of the key differences between SkyCore and the traditional EPC is in its continuous interaction with the UAV hardware and its APIs. In particular, we advocate for two new applications on top of our agents. Each SkyCore agent runs UAV Control Application that listens to flight change events from UAV and remaining battery resources on the UAV. This is necessary for our agents to properly handoff UEs to each other, for example, when a UAV needs to immediately leave the network for recharging. Such use cases clearly show the potential of our SDN-based

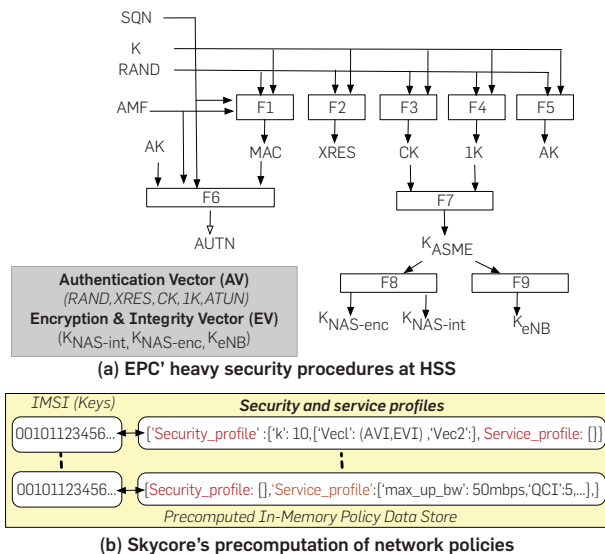
Figure 7. SkyCore refactors the EPC functionality into a lightweight software-driven agent having new interfaces for interaction with the local UAV and other UAVs.



UAV-aware architecture. In addition, we design an inter-agent (UAV) communication application (Section 2.3) that exchanges control plane messages with its neighbor agents to synchronize states *proactively*, thereby enabling seamless mobility (active and idle). The legacy EPC applications and new SkyCore core applications that need to exchange information with each other do so through our local publish-subscribe protocols.

Step 6. Replacing the hierarchical data plane gateways with a compact SDN switch. Because SkyCore is a flat architecture, it eliminates the need for hierarchical gateways on each UAV. To further make our agents compact, we refactor the SGW-D and PGW-D functionality into a single software switch. Each data plane function in S/PGW-D is implemented as a separate Match+Action table in this software switch. Each table performs a lookup on a subset of user's data traffic fields and applies the actions corresponding to the first match. Users' traffic travels through these tables before leaving or entering the UAV. In particular, our software switch performs UL/DL data rates enforcement, stateful firewall operations, and QoS control by transport-level mechanisms (e.g., setting DiffServ) based on QoS class identifier (QCI) associated with each UE. Although the legacy EPC tunnels each UE's traffic into two tunnel segments across the RAN, PGW-D, and SGW-D, SkyCore departs from this approach and terminates GTP-U tunnels inside our agent switch (decapsulates GTP-U header from uplink packets sent by the eNB and encapsulates downlink packets to the eNB into a proper GTP-U header) for two reasons. First, per-UE tunnels do not scale in LTE UAV networks as UEs are mobile and these tunnels are subject to frequent changes. Second, our consolidation already eliminates the need for complex GTP-U tunnels between the SGW-D and PGW-D functionality.

Figure 8. SkyCore's precomputation of network policies not only makes the core resource-efficient but also minimizes network access delay.

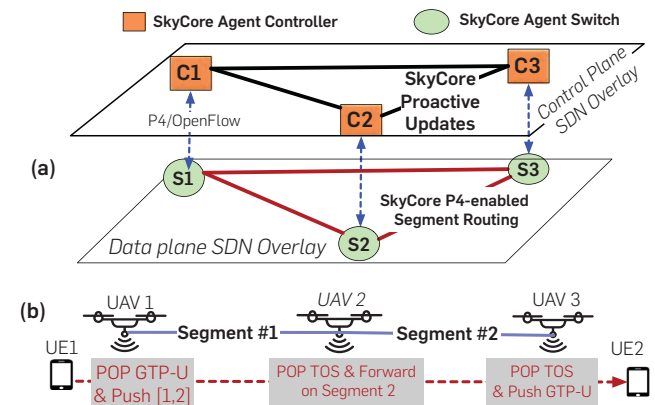


2.3. Efficient interagent communication

Scalable SDN control and data overlays. SkyCore agents seamlessly exchange control and data traffic with each other, a functionality that is lacking among today's EPC instances. Rather than relying on distributed and multi-hop wireless routing protocols, we choose to adopt an SDN approach in the design of SkyCore to support the traffic exchange between agents. SDN enables us to perform global optimization (e.g., multipath traffic engineering) and offer fine-grained programmability (e.g., to effectively support different QoS classes), which are necessary tools to instantly and efficiently reconfigure the core in response to network dynamics (e.g., UAV departures and arrivals) in our environment. In particular, we leverage SDN overlays to create two virtualized network layers (slices) on top of the physical UAV network (Figure 9a). One of these network slices is used for control plane traffic between SkyCore agents and the other is for data traffic. Our separation of the control and data traffic ensures time-critical control plane traffic is not affected when the network is saturated. To form the overlays, we use traffic tunneling technologies but depart from existing approaches used in the EPC and SDN-based datacenter (DC) networking^{15, 16} as they require frequent changes to the network configuration (will be discussed shortly). We adopt a novel variant of segment-based routing in SkyCore and propose a design for its optimization based on the most advanced capability in SDN, that is, P4 language.¹² P4 allows us to define new packet headers and packet processing actions for the SDN switch inside our agents to minimize the packet header overhead on inter-UAV links, which is caused by forming the overlays.

Proactive stateless mobility support. SkyCore replaces the notion of centralized HSS and PCRF with a precomputed policy data store replicated at different agents. Hence, it is essential that the UE states and policies be consistent across different agents, particularly during UE mobility. Reactive approaches to consistency management, for example, distributed hash table (DHT), put

Figure 9. (a) SkyCore's network-wide control and data plane connectivity for LTE UAV networks. (b) Example of our segment routing for data traffic from UE1 to UE2.



wireless (inter-UAV links) on the critical path of control functions. SkyCore avoids this real-time dependence by adopting a *proactive synchronization* of state between agents—each agent proactively broadcasts its changes to UE policies and states to other agents in the network. Such an approach (i) minimizes the control plane delay between agents, particularly in mobility scenarios as the destination agent already knows the latest information about the mobile UE; (ii) enables seamless handoff of active UEs to a neighboring UAV, when the current UAV goes down for a recharge; and (iii) is scalable because the amount of control plane traffic that is broadcasted on interagent backhaul links is negligible compared to user data plane traffic among agents (Section 4).

A SkyCore agent needs to send only three types of broadcast update messages in the network to build up a consistent network-wide view: (i) security update to notify other agents that it has used one of the security vectors precomputed for a UE and to request other agents to invalidate the vectors, (ii) location update to inform other agents that a particular UE has attached to its UAV, and (iii) policy update to communicate its local changes to the precomputed QoS and charging profile of a UE.

3. IMPLEMENTATION

SkyCore prototype. We prototyped a complete version of SkyCore that involved extensive engineering effort. Our prototype has three notable features: (1) seamlessly works with commercial LTE RANs and off-the-shelf UEs (SIM cards are programmed to connect to SkyCore) by exchanging signaling and data traffic with them; (2) is fully virtualized and can manage multiple LTE UAVs out of the box by forming a wireless network of SkyCore agents; and (3) fully adheres to our proposed designs both for a single agent (Figures 7 and 8) and across agents (interagent communication) (Figures 6 and 9). Each SkyCore agent consists of a controller enforcing control plane policies and a switch processing user data traffic. We developed a high-performance multithreaded controller in C++ and built our SkyCore switch on top of OVS²² software switch in the kernel space. We substantially instrumented and optimized OVS as it does not support our custom flow tables and switch actions (e.g., our P4-enabled tunneling scheme and GTP-U tunnel encapsulation/decapsulation operations). Because our baseline (Edge-EPC based on OpenEPC⁶—will be described shortly) operates in the user space, we developed another variant of the SkyCore switch in the user space on top of Lagopus software switch.⁴ This ensures that our comparisons are at the architecture level and independent of a particular packet forwarding technology.

UAV experiments. We conduct three kinds of experiments. (1) *Outdoor small-scale: 2 UAV, few UEs.* We deploy the SkyCore prototype on two advanced DJI Matrice 600 Pro drones (Figure 10). We securely install two machines on each drone. One of the machines (platform P1) is a low-end single-board 4-core server with 8GB of RAMs and 1.9GHz CPU that executes SkyCore and Edge-EPC. It is also equipped with a wireless network card to support our interagent communication. The other machine

is a commercial LTE small cell (ip.access S60 eNB) supporting LTE UEs (50Mbps downlink rate per UE) and connects through an Ethernet cable to platform P1. (2) *Outdoor large-scale: 2 UAV, tens of UEs.* To stress test SkyCore’s control and data planes in the presence of a large number of UEs, we replace the eNB on the drone with another single-board server that runs a unified RAN/UE emulator (emulates both an eNB and activity of a large number of UEs). The emulator interacts with the LTE core similar to real UEs. (3) *Emulating powerful UAV platforms.* To understand SkyCore’s performance with more powerful UAVs, we emulate the latter by replacing platform P1 with a high-end server (platform P2)—an Intel Xeon E5-2687W processor operating at 3.0GHz with 12 CPU cores and 128GB of RAM. Because it is not possible to fly our current drone with such a server, these experiments are conducted in the lab (results available in Moradi et al.¹⁸).

Baseline. We focus on comparisons between the Edge-EPC architecture (a standard EPC on each LTE UAV) and SkyCore. We implement the Edge-EPC using OpenEPC⁶ as it is the most complete open-source implementation of the 3GPP EPC architecture that can work with commercial devices (e.g., LTE eNBs and smartphones).

Metrics. We study four performance metrics under different network saturation levels: (1) UE-perceived control delay in network access (LTE attach/detach), (2) UE-perceived service disruption time in LTE active/idle-mode mobility, (3) CPU usage on our resource-constrained UAVs, and (4) supported data plane rate for user traffic.

4. EVALUATION

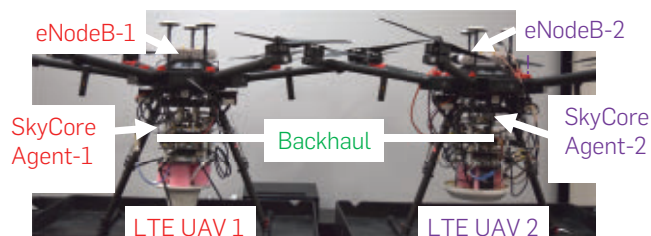
We first show the basic functionality and potential of SkyCore in realizing hotspot and stand-alone LTE UAV networks. We then demonstrate that SkyCore is more efficient and lightweight than the Edge-EPC architecture both in small- and large-scale experimental settings.

4.1. Small-scale on-drone evaluation

We form a two-drone LTE network (Figure 11), each in the partial line of sight (affected by one building) of a single mobile UE on the ground. Each drone covers a region with the diameter of 650 feet. The drones operate in a small overlapping area for our mobility experiments.

Basic functionality—LTE hotspots use case. Forming on-demand hotspots is an important use case for LTE UAV as well as 5G networks. In a single-drone experiment, we

Figure 10. Multi-UAV SkyCore prototype.



show this functionality by connecting one of our drones to the Internet through a terrestrial LTE network not accessible to our UEs on the ground (see Figure 11a). Next, we turn on a Moto G phone on the ground, which sends an LTE attach request to the SkyCore agent through the on-drone eNB. SkyCore agent successfully completes the LTE attach process by quickly accessing its precomputed policy data store. Then, we visit CNN.com and watch a 4K Youtube video on the phone. Finally, we take the Moto G into the airplane mode, causing the UE to properly detach from our agent. Figure 12 shows this basic functionality by depicting the data traffic exchanged between the UE and the Internet.

Basic functionality—stand-alone LTE use case. Next, we demonstrate SkyCore’s ability to create stand-alone LTE networks (e.g., between first responders across an impassable mountain). To emulate such a scenario, we

Figure 11. Our UAV-based setup for the basic functionality experiments. In Edge-EPC (the baseline), each SkyCore agent is replaced by a 3GPP EPC (OpenEPC).

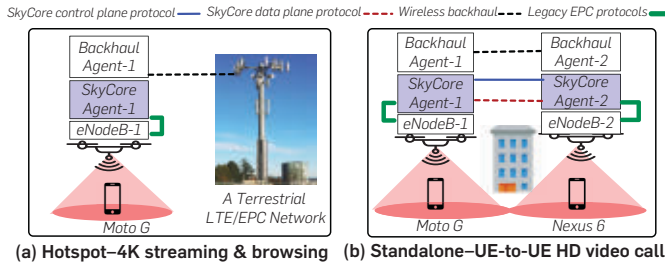


Figure 12. Hotspot UAV-based LTE network: exchanged data traffic and control events over time.

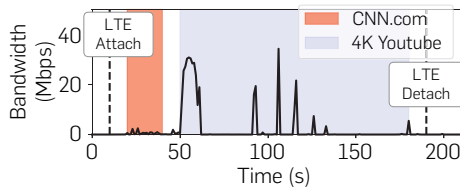
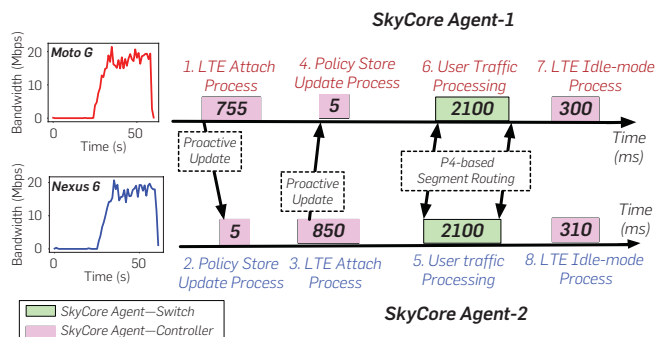


Figure 13. Stand-alone UAV-based LTE network: UE-to-UE HD video call enabled by SkyCore’s efficient interagent communication scheme. Control and data plane processing times and traffic exchanges inside and between the SkyCore agents on the two drones.



establish a direct video call between our two UEs across a building, each connected to a separate drone, through our interagent data plane overlay (see Figure 11b). Figure 13 shows the timeline of control and data plane traffic exchanges between the two SkyCore agents. We again turn on a Moto G phone in the area covered by the first drone. Its SkyCore agent handles the LTE attach process and sends a background SkyCore update message to the other drone’s agent. The message consists of location, policy, and security updates as described in Section 2. After the second agent processes this update, we turn on a Nexus 6 phone in the area covered by the second drone, triggering a similar SkyCore update message to the first agent in the background. Finally, we establish a 35-s HD video call from the Nexus 6 to the Moto G. Owing to SkyCore’s proactive background updates, the agent corresponding to the Nexus 6 does not have to wait to discover the location of the other UE. Based on our segment-based tunneling scheme, it immediately pushes the correct label stacks on its egress user data traffic and forwards it to the other agent. A similar process manifests in the reverse direction. In this two-UAV enabled video call, 7.5K video packets were successfully exchanged between the UEs.

Performance benefits of refactoring. Using the same setting, we demonstrate that SkyCore is significantly more lightweight than Edge-EPC. For a fair comparison with Edge-EPC, we employ SkyCore’s user space version here. We sample and average the LTE attach/detach delay and uplink/downlink bandwidth for the Moto G in the area covered by the first drone at 40 locations. As Figure 14 and Table 1 show, SkyCore on average reduces the network control plane delay (spent in the core) by 69–90% and the UE-perceived control plane delay by 40–60%. In addition, it doubles the uplink/downlink rates measured for the UE. Further, SkyCore lowers the avg. CPU usage on the machine running the core network by 25% in the LTE attach/detach events. These savings come

Figure 14. Breakdown of the network access delay in the core.

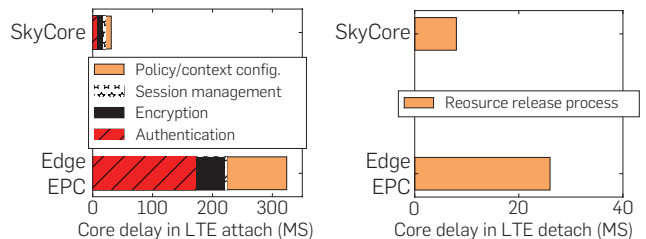
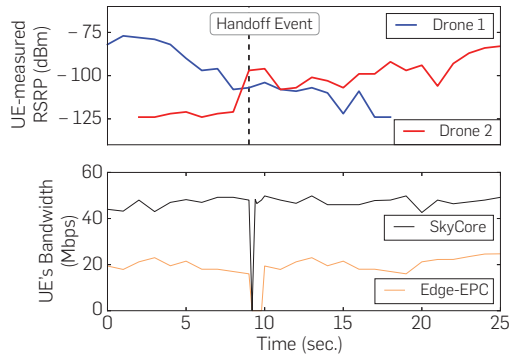


Table 1. Benefits of SkyCore’s software refactoring of the EPC functionality on UE-perceived QoS.

	Avg. data plane bandwidth (Mbps)		Avg. UE-perceived control delay (ms)	
	Downlink	Uplink	Attach	Detach
SkyCore	48.2	17.8	921	300
Edge-EPC	21.7	10.9	1545	750

Figure 15. Benefits of SkyCore’s interagent communication scheme: SkyCore provides seamless active-mode mobility support, whereas Edge-EPC causes severe connection drops.



from our precomputation of network policies and consolidation of the EPC functionality onto our compact SDN-driven agents.

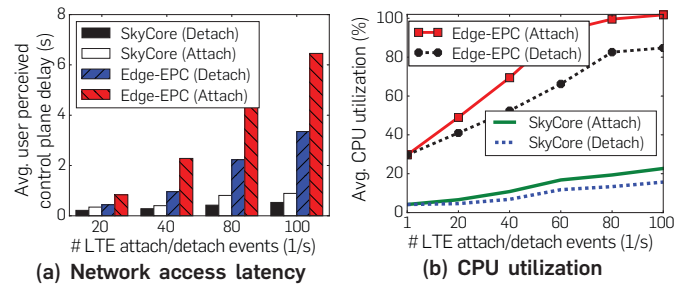
Efficient interagent communication—handoff. Unlike Edge-EPC, SkyCore supports seamless UE mobility, owing to its efficient interagent communication scheme. In this experiment, we measure the service disruption experienced by a mobile UE moving between the regions covered by our two drones and triggering a handoff event. Figure 15 depicts the signal strength received from the two drones on the UE and its continuous bandwidth measurements using iPerf3. The RAN on the first drone collects UE-measured RSRP values and sends a Handoff Required message to its local SkyCore agent when the RSRP values from the second drone become higher. Because SkyCore agents on the drones are already synced, the UE gets migrated to the second drone within a minimal 140 ms (incurred in the interagent coordination). In contrast, Edge-EPC does not support mobility of the UE and thus forces the UE to go through the detach process with the EPC on the first drone, followed by the heavy attach process with the EPC on the second drone. The entire process results in 2 s of disconnection time, significantly impacting mobile application performance.

4.2. Large-scale on-drone evaluation

Using the same two-drone experimental setting, we replace the ip.access eNB with a RAN/UE simulator on each drone to test SkyCore and Edge-EPC under large-scale network access workloads (mobility workload results are available in Moradi et al.¹⁸).

Handling signaling storms. Our RAN/UE emulator on the first drone emulates a flash crowd event with a large number of users entering the region (*attach storm*) covered by a drone. Similarly, the emulator creates an LTE detach storm by having many users gracefully disconnect from the drone. During this process, we sample the CPU utilization of the LTE core machine and measure the average control plane delay perceived by the UEs. In Figure 16a, we observe that the UEs experience exponentially larger delays when the attach/detach load on Edge-EPC increases. In particular, when the number of attach

Figure 16. SkyCore substantially reduces network access time and CPU utilization to handle large-scale network access requests.



requests per sec. reaches 100, the UEs must wait by up to 6 s before connecting to the network, thereby degrading QoE. With the EPC being a complex system, we observe from Figure 16b that Edge-EPC quickly uses its available CPU resources on the drone and thus faces performance bottlenecks, leading to larger latencies. In contrast, we notice that the network access delay is below 1 s when the drone employs SkyCore owing to its software refactoring of the EPC functionality.

5. RELATED WORK


SDN/NFV-based EPC. Recently, the wireless networking community has proposed several software-defined EPC solutions. SoftMoW¹⁹ enhances the programmability of the EPC by decoupling its control and data planes. KLEIN²³ and SCALE¹⁰ optimize the placement of the EPC nodes on geo-distributed DCs. ECHO²⁰ deals with EPC-node failure in unreliable public clouds. PEPC²⁴ scales the EPC data plane by creating a per-UE EPC-in-box. Although there are some similarities between SkyCore and this proposal, the differences are significant. These prior designs are customized for highly-reliable, often hierarchical DC infrastructure, where over provisioning and reactive network updates are inexpensive. In contrast, SkyCore operates in an unreliable and resource-constrained wireless environment, where such approaches scale poorly.

SDN control and data planes. There is a rich literature in distributed SDN control plane designs with hierarchical and flat structures (e.g., ONOS¹¹). Most of the schemes are designed for DC networks and operate based on a centralized data store or complex consensus algorithms, which are ill-suited for our unreliable multi-UAV environment.

RAN optimization for LTE UAVs. DroneNet¹⁴ extends the coverage of existing LTE cells by creating WiFi on-drone hotspots. Some recent works^{17,25} investigate the theoretical optimization of a UAV trajectory for certain mobile users on the ground (e.g., maximize the minimum average rate among all user). These RAN efforts are predominantly for a single UAV and complementary to SkyCore that focuses on the EPC design for multi-UAV LTE networks.

6. CONCLUSION

We presented the design and implementation of a novel edge-EPC architecture—SkyCore, supporting the untethered and reliable operation of multi-UAV LTE networks. SkyCore’s

SDN-based design is equally applicable to future RAN (e.g., 5G and 6G) technologies. Further, even in the context of terrestrial networks, where EPC-RAN communication is often reliable, operators can leverage Edge-EPC designs such as SkyCore to move the EPC functionality to their edge clouds or cell towers and realize ultralow latency as required by many 5G use cases. Such deployments are motivated by operators' push toward mobile edge computing (MEC)^{13, 21} and their efforts in deploying white box switches at cell towers.¹ 

References

1. AT&T is deploying white box hardware in cell towers to power mobile 5G era, 2017. <https://goo.gl/snRW6M>.
2. CBRS Spectrum, 2017. <https://goo.gl/3zbYyo>.
3. Flying COW connects Puerto Rico, 2017. <https://goo.gl/NEq1HA>.
4. Lagopus: SDN switch, 2017. <http://www.lagopus.org/>.
5. LTE signaling storm, 2017. <http://goo.gl/qk6Bp9>.
6. OpenEPC, 2017. <http://www.openepc.com/>.
7. Oracle communications LTE diameter signaling index, 2017. <https://goo.gl/6BZ8Fo>.
8. Verizon trials drones as flying cell towers, 2017. <https://goo.gl/q9YjNv>.
9. When COWs fly: AT&T sending LTE signals from drones, 2017. <https://goo.gl/9u33qC>.
10. Banerjee, A., Mahindra, R., Sundaresan, K., Kaser, S., Van der Merwe, K., Rangarajan, S. Scaling the LTE control-plane for future mobile access. In *Proceedings of the ACM CoNEXT*, 2015.
11. Berde, P., Gerola, M., Hart, J., Higuchi, Y., Kobayashi, M., et al. ONOS: towards an open, distributed sdn os. In *Proceedings of the ACM SIGCOMM Workshop on HotSDN*, 2014.
12. Bosshart, P., et al. P4: Programming protocol-independent packet processors. *ACM CCR*, 2014.
13. Cho, J., et al. ACACIA: Context-aware edge computing for continuous interactive applications over mobile networks. In *Proceedings of the ACM CoNEXT*, 2016.
14. Dhekne, A., et al. Extending cell tower coverage through drones. In *Proceedings of the ACM HotMobile*, 2017.
15. Hong, C.-Y., Kandula, S., Mahajan, R., et al. Achieving high utilization with software-driven WAN. *ACM CCR*, 2013.
16. Jain, S., Kumar, A., Mandal, S., et al. B4: Experience with a globally-deployed software defined WAN. *ACM CCR*, 2013.
17. Lin, X., Yajnanarayana, V.,

- Muruganathan, S.D., et al. The sky is not the limit: LTE for unmanned aerial vehicles. *arXiv preprint arXiv:1707.07534*, 2017.
18. Moradi, M., Sundaresan, K., Chai, E., Rangarajan, S., Mao, M. Skycore: Moving core to the edge for untethered and reliable UAV-based LTE networks. In *Proceedings of the ACM MobiCom*, 2018.
19. Moradi, M., Wu, W., Li, L.E., Mao, Z.M. SoftMoW: Recursive and reconfigurable cellular wan architecture. In *Proceedings of the ACM CoNEXT*, 2014.
20. Nguyen, B., Zhang, T., Radunovic, B., et al. MSR Technical Report. *A Reliable Distributed Cellular Core Network for Hyper-Scale Public Clouds*, 2018.
21. Patel, M., et al. Mobile-edge computing introductory technical white paper. *White Paper, Mobile-edge Computing (MEC) Industry Initiative*, 2014.
22. Pfaff, B., Pettit, J., Koponen, T., et al. The design and implementation of open vswitch. In *Proceedings of the USENIX NSDI*, 2015.
23. Qazi, Z.A., Krishna, P., Sekar, V., Gopalakrishnan, V., Joshi, K., Das, S.R. Klein: A minimally disruptive design for an elastic cellular core. In *Proceedings of the ACM SOSR*, 2016.
24. Qazi, Z.A., Walls, M., Panda, A., et al. A high performance packet core for next generation cellular networks. In *Proceedings of the ACM SIGCOMM*, 2017.
25. Wu, Q., Zeng, Y., Zhang, R. Joint trajectory and communication design for multi-UAV enabled wireless networks. *IEEE/ACM TON*, 2018.

Mehrdad Moradi and Z. Morley Mao ([moradi, zmao]@umich.edu), University of Michigan, Ann Arbor, MI, USA.

Karthikeyan Sundaresan, Eugene Chai and Sampath Rangarajan ([karthiks, eugene, sampath]@nec-labs.com), NEC Laboratories America, Princeton, NJ, USA.

© 2021 ACM 0001-0782/21/1 \$15.00

Semantic Web for the Working Ontologist

Effective Modeling for Linked Data, RDFS, and OWL

**Dean Allemang
James Hendler
Fabien Gandon**

THIRD EDITION

ISBN: 978-1-4503-7617-4
DOI: 10.1145/3382097
<http://books.acm.org>
<http://store.morganclaypool.com/acm>



 **ACM BOOKS**
Collection II

Baylor University *Endowed Chair in Data Science*

The McCollum Family Endowed Chair in Data Science is a research-focused position in the Baylor University Computer Science and Informatics Department. Data Science is one of the five Signature Academic Initiatives in Baylor's strategic plan Illuminate (Illuminate - Data Science) and is involved in key research for the University (Data Science Research). This transformative, endowed position is a visionary investment in the future of Data Science research and education across the university (Endowment Details).

The Department: Computer Science and Informatics is one of three departments in the School of Engineering and Computer Science. It offers a B.S. in Informatics with majors in Data Science and Bioinformatics, B.S. and B.A. degrees in Computer Science, and a B.S. in Computing with a major in Computer Science Fellows. On location M.S. and Ph.D. degrees in Computer Science are offered, as well as an online M.S. program which started Fall 2020. The Department has 17 full-time faculty, over 280 undergraduates, and over 25 graduate students. Departmental website: Informatics.

The University: Baylor University is a private Christian university and a nationally ranked research institution, consistently listed with highest honors among The Chronicle of Higher Education's "Great Colleges to Work For." Baylor seeks faculty who share in our aspiration to become a tier-one research institution while strengthening our distinctive Christian mission. As the world's largest Baptist University, Baylor offers over 40 doctoral programs and has over 17,000 students from all 50 states and more than 85 countries.

Qualifications: The University invites applications for this tenure-track position at the rank of full Professor beginning in the Fall 2021 semester. An ideal candidate will help shape a comprehensive, university-wide strategic plan for Data Science. This will be done through leadership, collaboration, and growth of infrastructure and interdisciplinary research. Applicants should have a Ph.D. in Data Science or a related discipline; Baylor is recruiting new faculty with a deep commitment to excellence in teaching, research, and scholarship. Other qualifications include an established history of extramural funding, high impact academic artifacts, and graduate student mentorship. A viable applicant should demonstrate excellent potential as an individual researcher and collaborator across multiple disciplines.

Appointment Date: Fall 2021. For full consideration, applications must be received by December 31, 2020.

Application Procedure: To apply, please submit a letter of application, a 1-2 page research

plan, a 1-2 page teaching philosophy, a copy of an official transcript showing the highest degree conferred (if the Ph.D. is in progress, a copy of the official transcript of completed Ph.D. hours should also be submitted), and the names and email addresses of three persons willing to provide letters of recommendation as a single PDF file through this Interfolio link Application Link Finalists for this position will be required to submit official transcripts for the doctoral degree in advance of a campus visit. Inquiries about the position can be sent to CSSearch@Baylor.edu.

Baylor University is a private not-for-profit university affiliated with the Baptist General Convention of Texas. As an Affirmative Action/Equal Opportunity employer, Baylor is committed to compliance with all applicable anti-discrimination laws, including those regarding age, race, color, sex, national origin, marital status, pregnancy status, military service, genetic information, and disability. As a religious educational institution, Baylor is lawfully permitted to consider an applicant's religion as a selection criterion. Baylor encourages women, minorities, veterans and individuals with disabilities to apply.

Boston College *Tenure Track Assistant Professor of Computer Science*

The Computer Science Department of Boston College seeks a tenure-track Assistant Professor beginning in the 2021-2022 academic year. Successful candidates for the position will be expected to develop strong research programs that can attract external funding in an environment that also values high-quality undergraduate teaching. Outstanding candidates in all areas of Computer Science will be considered, with a preference for those who demonstrate a potential to contribute to cross-disciplinary teaching and research in conjunction with the planned Schiller Institute for Integrated Science and Society at Boston College.

A Ph.D. in Computer Science or a closely related discipline is required. See cs.bc.edu and <https://www.bc.edu/bc-web/centers/schiller-institute.html> for more information. Application review is ongoing.

Applicants should submit a cover letter, a detailed CV, and teaching and research statements. Arrange for three confidential letters of recommendation to be uploaded directly to Interfolio. To apply go to: <https://apply.interfolio.com/79609>.

Boston College conducts background checks as part of the hiring process. Information about the University and our department is available at bc.edu and cs.bc.edu.

Boston College is a Jesuit, Catholic university that strives to integrate research excellence with a foundational commitment to formative liberal arts education. We encourage applications from

candidates who are committed to fostering a diverse and inclusive academic community. Boston College is an Affirmative Action/Equal Opportunity Employer and does not discriminate on the basis of any legally protected category including disability and protected veteran status. To learn more about how BC supports diversity and inclusion throughout the university, please visit the Office for Institutional Diversity at <http://www.bc.edu/offices/diversity>.

California State University San Bernardino (CSUSB) *School Director with Tenure at the rank of Full or Associate Professor*

California State University San Bernardino (CSUSB), a comprehensive university of The California State University, one of the largest and most widely-recognized institutions of higher education in the nation, invites applications for an academic administrative leader with a collaborative and inspiring vision for the position of Director of School of Computer Science and Engineering (CSE). The successful candidate should be eligible for appointment at the level of Professor or Associate Professor with tenure to begin in August 2021.

As one of the largest department/school in the College of Natural Sciences, the School of Computer Science and Engineering (CSE) has 12 tenure-track faculty with a variety of research interests and approximately 1000 students with a diverse backgrounds. CSE offers 4 undergraduate and 1 graduate programs, i.e., B.S. in Computer Science (ABET accredited), B.S. in Computer Engineering (ABET accredited), B.S. in Bioinformatics, B.A. in Computer Systems, and M.S. in Computer Science.

The School Director reports to the Dean of the College of Natural Sciences and is a 12-month 0.75 position. The director will provide strong academic leadership in the planning and administration of graduate and undergraduate programs in computer science and engineering, assist the entire faculty in developing new initiatives and a viable strategic vision, teach courses, maintain an active research program involving undergraduate and/or graduate students, work with the CSUSB Office of Advancement in fundraising, and maintain and extend our existing strong relationship with industry and government agencies. The overall responsibilities of the Director position is described in FAM 641.65, which is available at: <https://www.csusb.edu/faculty-senate/fam/600-675-personnel/640-644-recruitment-appointment-responsibilities-related>

The preferred candidate should meet the following qualifications

- ▶ Ph.D. in Computer Science or Computer Engineering discipline.
- ▶ Candidates should be eligible for appointment at the level of Professor or Associate Professor with tenure.

- ▶ Demonstrated administrative experience as a department chair/school director.
- ▶ Excellent leadership, communication and interpersonal skills
- ▶ Excellent record of teaching at undergraduate and graduate level
- ▶ Excellent record of publication and research funding
- ▶ Excellent record of leadership in ABET Accreditation

For more information on how to apply, please visit <https://www.csusb.edu/cse>. Formal review of applications will begin **February 1, 2021** and continue until the position is filled.

If you are interested in this opportunity, we invite you to apply by using this CSU Recruit hyperlink at: <https://secure.dc4.pageuppeople.com/apply/873/gateway/Default.aspx?c=apply&sJobIDs=497961&SourceTypeID=803&sLanguage=en-us&lApplicationSubSourceID=11248>

Max Planck Institutes in Computer Science Tenure-Track Openings at Max Planck Institutes in Computer Science

The Max Planck Institutes for Informatics (Saarbruecken), Software Systems (Saarbruecken and Kaiserslautern), and Security and Privacy (Bochum), invite applications for tenure-track faculty in all areas of computer science. We expect to fill several positions.

A doctoral degree in computer science or related areas and an outstanding research record

are required. Successful candidates are expected to build a team and pursue a highly visible research agenda, both independently and in collaboration with other groups.

The institutes are part of a network of over 80 Max Planck Institutes, Germany's premier basic-research organisations. MPIs have an established record of world-class, foundational research in the sciences, technology, and the humanities. The institutes offer a unique environment that combines the best aspects of a university department and a research laboratory: Faculty enjoy full academic freedom, lead a team of doctoral students and post-docs, and have the opportunity to teach university courses; at the same time, they enjoy ongoing institutional funding in addition to third-party funds, a technical infrastructure unrivaled for an academic institution, as well as internationally competitive compensation.

We maintain an international and diverse work environment and seek applications from outstanding researchers worldwide. The working language is English; knowledge of the German language is not required for a successful career at the institutes.

Qualified candidates should apply on our application website (apply.cis.mpg.de). To receive full consideration, applications should be received by December 15th, 2020.

The Max Planck Society wishes to increase the number of women in those areas where they are underrepresented. Women are therefore explicitly encouraged to apply. The Max Planck Society is also committed to increasing the number of employees with severe disabilities in its workforce.

Applications from persons with severe disabilities are expressly desired.

The initial tenure-track appointment is for five years; it can be extended to seven years based on a positive midterm evaluation in the fourth year. A permanent contract can be awarded upon a successful tenure evaluation in the sixth year.

University of Central Missouri Assistant Professor in Computer Science - Tenure Track

The School of Computer Science and Mathematics at the University of Central Missouri is accepting applications for one tenure-track position in Computer Science at the rank of Assistant Professor. The appointment will begin August 2021. We are looking for faculty excited by the prospect of shaping our school's future and contributing to its sustained excellence.

The Position: Duties will include teaching undergraduate and graduate courses in computer science and/or cybersecurity and developing new courses depending upon the expertise of the applicant and school needs, conducting research which leads toward peer-reviewed publications and/or externally funded grants, and program accreditation/assessment. Faculty are expected to assist with school and university committee work and service activities, and advising majors.

Required Qualifications:

- ▶ Ph.D. in Computer Science or Software Engineering by August 2021
- ▶ Research expertise and/or industrial experiences in Cybersecurity or Software Engineering
- ▶ Demonstrated ability to teach existing courses at the undergraduate and graduate levels
- ▶ Ability to develop a quality research program and secure external funding
- ▶ Commitment to engage in curricular development/assessment at the undergraduate and graduate levels
- ▶ A strong commitment to excellence in teaching, research, and continued professional growth
- ▶ Excellent verbal and written communication skills

The Application Process: To apply online, go to <https://jobs.ucmo.edu>. Apply to position #997516. The following items should be attached: a letter of interest, a curriculum vitae, a teaching and research statement, copies of transcripts, and a list of at least three professional references including their names, addresses, telephone numbers and email addresses. Official transcripts and three letters of recommendation will be requested for candidates invited for on-campus interview. For more information, contact:

Dr. Songlin Tian, Search Committee Chair
School of Computer Science and
Mathematics
University of Central Missouri
Warrensburg, MO 64093
(660) 543-4930
tian@ucmo.edu

Initial screening of applications begins November 30, 2020 and continues until position is filled.

Full PhD and Postdoc scholarships at one of Germany's leading Digital Engineering PhD Schools

Research School

"Service-Oriented Systems Engineering"

The research school "Service-Oriented Systems Engineering" is active in research areas such as system design, analysis, and modeling; adaptability; component-based development and application integration; business process management; cyber security; software engineering; and programming technology.

Research School

"Data Science and Engineering"

The research school "Data Science and Engineering" unites top PhD students and researchers in all areas of data-driven research and technology, including scalable storage, stream processing, data cleaning, machine learning and deep learning, text processing, data visualization, digital health and more.

The Hasso Plattner Institute (HPI) is Germany's university excellence center for Digital Engineering, covering the research areas of systems engineering, data science, cybersecurity, and digital health.

Its location in Potsdam, right on the border to Berlin, offers a perfect living and working environment for young researchers. Each year we provide 18 full PhD and postdoctoral scholarships in our two PhD programs. Both programs have an interdisciplinary and international structure. They interconnect all research groups at HPI as well as its branches at the University of Cape Town, Technion, Nanjing University, and UC Irvine.

The Hasso Plattner Institute offers:

- Full research scholarships, travel funds, and no tuition
- Cutting edge research projects
- An outstanding research environment
- Close mentorship by professors and postdocs
- Excellent graduate and undergraduate students
- Cooperation with many partners in academia and industry

Applications now open until 1 February to start in April. Or apply until 15 August to start in October.
www.hpi.de/research-school



AA/EEO/ADA. Women and minorities are encouraged to apply.

UCM is located in Warrensburg, MO, which is 35 miles southeast of the Kansas City metropolitan area. It is a public comprehensive university with about 11,000 students. The School of Computer Science and Mathematics offers undergraduate and graduate programs in Computer Science, Cybersecurity and Software Engineering with approximately 1000 students. The undergraduate Computer Science and Cybersecurity programs are accredited by the Computing Accreditation Commission of ABET.

Vanderbilt University
20+ Tenure-Track Faculty Positions in
Computer Science

The Department of Electrical Engineering and Computer Science (EECS) is launching a multi-year faculty recruitment and hiring process in Computer Science for 20 tenure-track positions at the Assistant, Associate, and Full Professor levels, but with preference at early-career appointments. This year, the initiative will support at least six new faculty positions. Destination-CS is part of the university's recently launched Destination Vanderbilt, a \$100 million university excellence initiative to recruit new faculty. Over the next two to four years, the university will leverage the investment to recruit approximately 60 faculty who are leaders and rising stars in their fields.

We seek exceptional candidates in broadly defined areas of computer science that enhance our research strengths in areas that align with the following investment and growth priorities of the Vanderbilt University School of Engineering (<http://vu.edu/destination-cs>):

1. Autonomous and Intelligent Human-AI-Machine Systems and Urban Environments
2. Cybersecurity and Resilience
3. Computing and AI for Health, Medicine, and Surgery
4. Design of Next Generation Systems, Structures, Materials, and Manufacturing

Our priorities are designed to ensure the strongest positive impact on computer science and cross-disciplinary areas at all five academic departments in the School of Engineering and other colleges and schools across campus. The hiring initiative builds on these strengths and aspires to propel the Vanderbilt computer science program to one of the leading academic programs nationally and beyond. Successful candidates are expected to teach at the undergraduate and graduate levels and to develop and grow vigorous programs of externally funded research.

Vanderbilt University is a private, internationally renowned research university located in vibrant Nashville, Tennessee, and with the adjoining Vanderbilt University Medical Center, is the largest employer in the region. Its 10 schools share a single cohesive campus that nurtures interdisciplinary activities. The School of Engineering is on a strong upward trajectory in national and international stature and prominence, and has built infrastructure to support a significant expansion in faculty size. In the rankings of graduate engineering programs by U.S. News & World Report, the school ranks in the top 20 private, research-extensive engineering schools. Five-year

average T/TK faculty funding in the EECS Department is above \$800k per year. All junior faculty members hired during the past 15 years have received prestigious young investigator awards, such as NSF CAREER and DARPA CSSG.

With a metro population of approximately 1.9 million people, Nashville has been named one of the 15 best U.S. cities for work and family by Fortune magazine, was ranked as the #1 most popular U.S. city for corporate relocations by Expansion Management magazine, and was named by Forbes magazine as one of the 25 cities most likely to have the country's highest job growth over the coming five years. The top major industries by employment include trade, transportation and utilities; education and health services; professional and business services; government; and leisure and hospitality. Other industries include manufacturing, financial activities, construction, and information. Long known as a hub for health care and music, Nashville is emerging as a tech-

nology center with a considerable pool of health care, AI, and defense-related jobs available.

In recent years, the city has experienced an influx of major office openings by some of the largest global tech companies and prime Silicon Valley startups.

Vanderbilt University has a strong institutional commitment to recruiting and retaining an academically and culturally diverse community of faculty. Minorities, women, individuals with disabilities, and members of other underrepresented groups, in particular, are encouraged to apply. Vanderbilt is an Equal Opportunity/Affirmative Action employer.

Applications should be submitted on-line at: <http://apply.interfolio.com/80624>. For more information, please visit our web site: <http://vu.edu/destination-cs>. Applications will be reviewed on a rolling basis beginning December 15, 2020 with interviews beginning January 1, 2021. For full consideration, application materials must be received by January 31, 2021.



ADVERTISING IN CAREER OPPORTUNITIES

How to Submit a Classified Line Ad:

Send an e-mail to acmm mediasales@acm.org. Please include text, and indicate the issue/or issues where the ad will appear, and a contact name and number.

Estimates:

An insertion order will then be e-mailed back to you. The ad will be typeset according to CACM guidelines. NO PROOFS can be sent. Classified line ads are NOT commissionable.

Deadlines:

20th of the month/2 months prior to issue date. For latest deadline info, please contact:

acmm mediasales@acm.org

Career Opportunities Online:

Classified and recruitment display ads receive a free duplicate listing on our website at:

<http://jobs.acm.org>

Ads are listed for a period of 30 days.

For More Information Contact:

ACM Media Sales

at 212-626-0686 or

acmm mediasales@acm.org



Dennis Shasha

DOI:10.1145/3434645

Upstart Puzzles Stay in Balance

No tipping.

TWO PLAYERS, EACH with a collection of weights, sit in front of a plank that weighs three kilograms with two supports at -1 and $+1$. Each player wants to get rid of his or her weights, by placing them on integral markers, at most one weight per integral marker. So, for example, player A may not place a weight above A's weights or player B's weights. Further, neither player is allowed to place a weight at $-1, 0$, or $+1$.

In a turn, a player must put at least one weight somewhere on the plank, but may put several weights on the plank, if they are at consecutive integer marks. The goal of each player is to be the first to place all of his or her weights without causing the plank to tip (by having a strictly negative torque on the right support or a strictly positive torque on the left support).

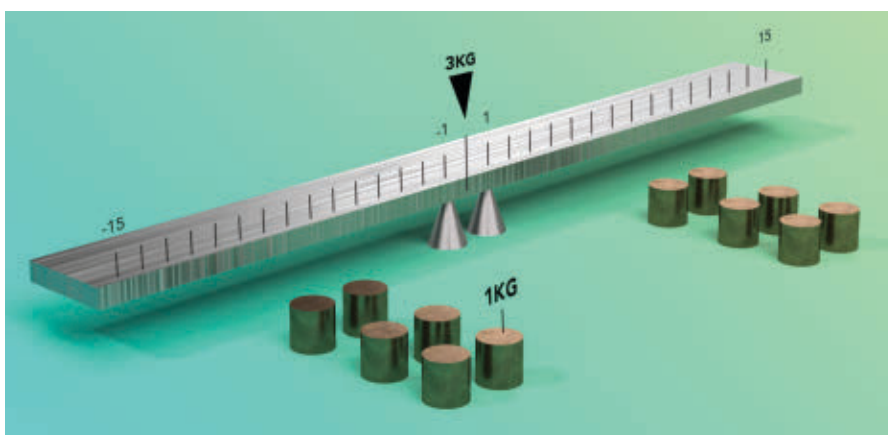
Warm-Up: Let's say all weights weigh one kilogram and each player has six weights. Suppose each player always places as many weights as possible in each move (a "greedy" strategy). Which player will win?

Solution to Warm-Up: First player can put a weight at $+2$ and $+3$. So torque on left support is $-(3*1 + 1*3 + 1*4)$ and on right support is 0 . So player 2 can put weights at $-2, -3, -4, -5$. Now torque on right support is $(1*6 + 1*5 + 1*4 + 1*3 + 3*1) - (1*1 + 1*2) = 18$. The torque on the left support is now 0 . Therefore the first player can add weights at $+4, +5, +6, +7$ and win.

So if we have just six weights of one kilogram and each is greedy, then the first player wins.

Question: If each player has seven weights of one kilogram and each is greedy, then which player wins?

Solution: Each player plays as before. After the first player adds weights at $+4, +5, +6, +7$, the torque on the



Three-kilogram plank. "How can I be the first to place all my weights on the plank without causing the plank to tip?"

left support is $-(5 + 6 + 7 + 8) = -26$ the second player can put his or her last weight at -6 and win.

Question: Could the above change if the plank weighs more?

Solution: Yes, for example, if the plank weighed much more, then the first player could put all his weights on one side.

In non-greedy play, each player must put at least one weight somewhere on the plank during his or her turn. If multiple weights, they must be consecutive.

In the following upstarts, assume the same rules as before: the plank weighs three kilograms, weights can be put only on integral markers, and never on top of another weight.

Upstart 1: Is there any way the player who moves first can guarantee to win if he or she can choose the 10 integral weights that each player starts with provided each weight weighs at least one kilogram and all the weights are distinct with the further restrictions that the first player must play greedily but his or her opponent need not.

Upstart 2: One player is the chooser and given an n must choose exactly n distinct weights and they must each weigh an integral number of kilograms ≥ 1 . The non-chooser decides whether to go first or second. Both must play greedily. For which values of n can the chooser guarantee to win?

Upstart 3: As in Upstart 2, the chooser, given an n , must choose exactly n distinct weights that must each weigh an integral number of kilograms ≥ 1 . The chooser decides whether to go first or second, but must play greedily. The non-chooser need not play greedily. For which values of n can the chooser guarantee to win?

Dennis Shasha (dennishasha@yahoo.com) is a professor of computer science in the Computer Science Department of the Courant Institute at New York University, New York, NY, USA, as well as the chronicler of his good friend the omniheurist Dr. Ecco.

All are invited to submit their solutions to upstartpuzzles@cacm.acm.org; solutions to upstarts and discussion will be posted at <http://cs.nyu.edu/cs/faculty/shasha/papers/cacmpuzzles.html>

Copyright held by author.

volume
01

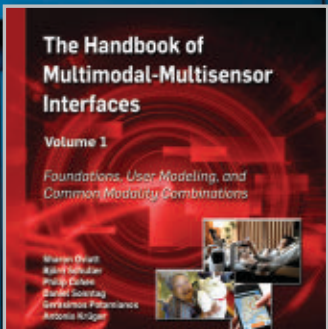
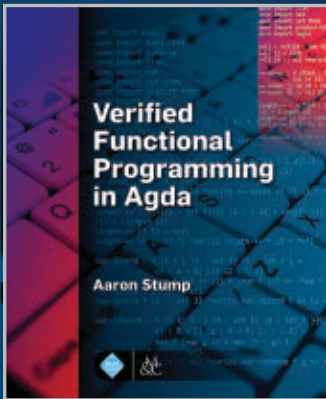
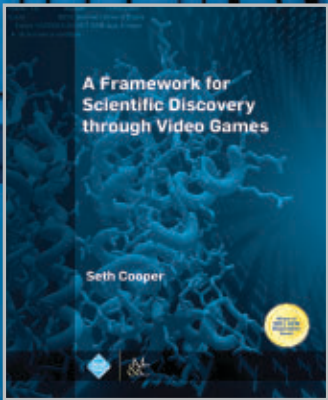
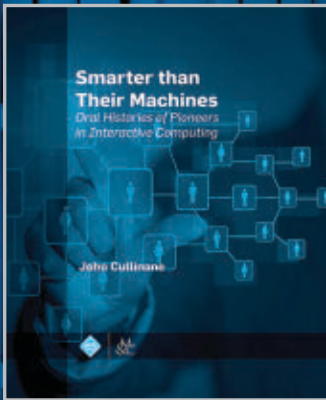
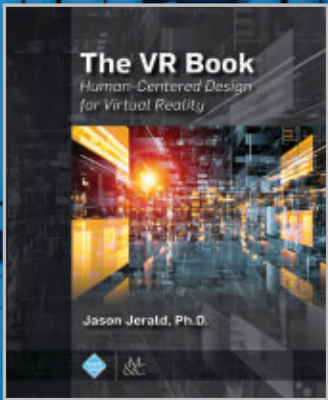
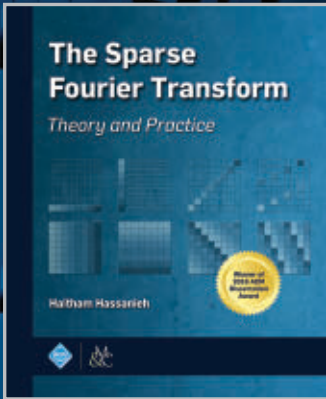
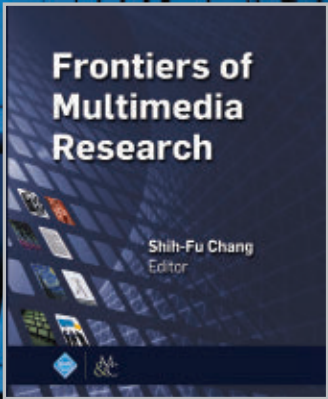
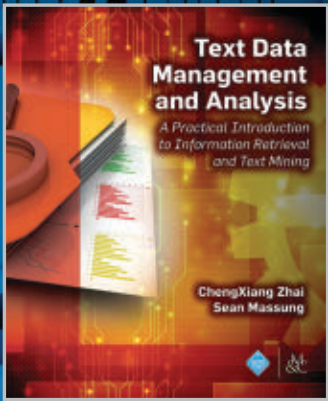
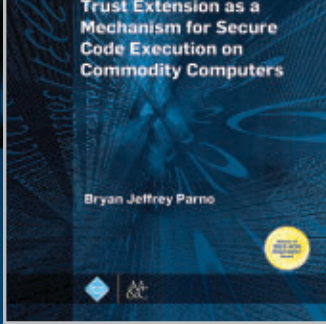
number
01

FIRST
ISSUE
PUBLISHED

ACM Transactions on Internet of Things
is now available in
the ACM Digital Library



ACM Transactions on Internet of Things (TIOT) publishes novel research contributions and experience reports in several research domains whose synergy and interrelations enable the IoT vision. TIOT focuses on system designs, end-to-end architectures, and enabling technologies, and on publishing results and insights corroborated by a strong experimental component.



In-depth. Innovative. Insightful.

Inspired by the need for high-quality computer science publishing at the graduate, faculty, and professional levels, ACM Books are affordable, current, and comprehensive in scope.

**Full Collection | Title List
Now Available**

For more information, please visit
<http://books.acm.org>



Association for Computing Machinery

1601 Broadway, 10th Floor, New York, NY 10019-7434, USA

Phone: +1-212-626-0658 Email: acmbooks-info@acm.org