COMMUNICATIONS OF THE ACM

CACM.ACM.ORG

12/2021 VOL.64 NO.12

THE HARDWARE LOTTERY

Datasheets for Datasets • Q&A with Scott Aaronson
Digital Agriculture • Speculative Taint Tracking

Association for
Computing Machinery

EICS 2022

The 14th ACM SIGCHI Symposium on

Engineering Interactive Computing Systems

Sophia Antipolis, Côte d'Azur, France

21 - 24 June, 2022

eics.acm.org/2022/

Work presented at EICS covers the full range of aspects that come into play when engineering interactive systems, such as innovations in the design, development, deployment, verification and validation of interactive systems. Authors are invited to submit original work on engineering interactive systems, including novel work on languages, processes, methods, models and tools describing and demonstrating interactive systems that advance the current state of the art.

Sponsored by

acm Association for Computing Machinery

SIGCHI special interest group computer human interaction

Inria

cnrs

ifip

IFIP WG 2.7/13.4

i3S sophia antipolis

Submission deadlines

Full papers
February 18, 2022

Late-Breaking Results, Demo Papers, Tutorials, and Tech Notes
March 11, 2022

Workshops proposal
February 4, 2022

EngageCSEdu.org

Explore Peer-reviewed Resources for Engaging Students

EngageCSEdu provides a collection of
computing resources for engaging all students

EngageCSEdu provides faculty-contributed, peer-reviewed course materials (Open Educational Resources or OERs) for all levels of introductory computer science instruction (CS0, CS1, Data Structures, and Discrete Math). Materials in the EngageCSEdu collection make clear use of evidence-based engagement practices, particularly those shown to help broaden participation in computing. EngageCSEdu promotes a framework of research-based teaching practices that support diversity and fosters a community of faculty committed to broadening participation in computing through great pedagogy. Explore the collection and consider submitting your course materials to EngageCSEdu.

https://engage-csedu.org

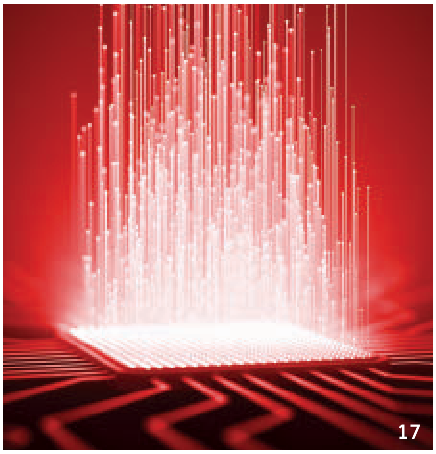
acm Association for Computing Machinery

Departments

Last Byte

Association for Computing Machinery
Advancing Computing as a Science & Profession

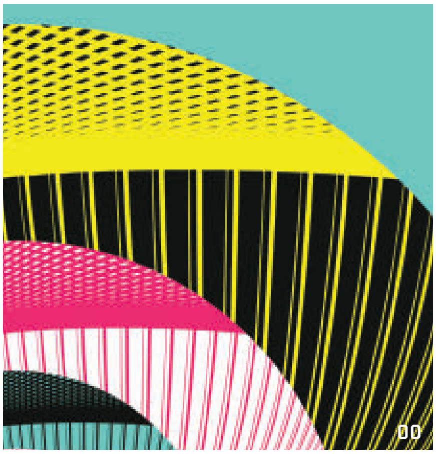
News


17

Viewpoints

IMAGE BY KTSDESIGN

Practice



Articles' development led by acmqueue
queue.acm.org

About the Cover:
Do the best AI research ideas always win? Fingers crossed—but this month's cover story (p. 58) contends that luck plays an all-too important role. The author explores the notion of a "hardware lottery," where the ability for AI to hit it big hinges on its compatibility with existing hardware and software. Cover image by Andrij Borys Associates.

IMAGE BY L.DEP/SHUTTERSTOCK

Contributed Articles

Review Articles

Research Highlights

COMMUNICATIONS OF THE ACM

Trusted insights for computing's leading professionals.

Communications of the ACM is the leading monthly print and online magazine for the computing and information technology fields. *Communications* is recognized as the most trusted and knowledgeable source of industry information for today's computing professional. *Communications* brings its readership in-depth coverage of emerging areas of computer science, new trends in information technology, and practical applications. Industry leaders use *Communications* as a platform to present and debate various technology implications, public policies, engineering challenges, and market trends. The prestige and unmatched reputation that *Communications of the ACM* enjoys today is built upon a 50-year commitment to high-quality editorial content and a steadfast dedication to advancing the arts, sciences, and applications of information technology.

ACM, the world's largest educational and scientific computing society, delivers resources that advance computing as a science and profession. ACM provides the computing field's premier Digital Library and serves its members and the computing profession with leading-edge publications, conferences, and career resources.

Executive Director and CEO
Vicki L. Hanson
Deputy Executive Director and COO
Patricia Ryan
Director, Office of Information Systems
Wayne Graves
Director, Office of Financial Services
James Schembari
Director, Office of SIG Services
Donna Cappo
Director, Office of Publications
Scott E. Delman

ACM COUNCIL
President
Gabriele Kotsis
Vice-President
Joan Feigenbaum
Secretary/Treasurer
Elisa Bertino
Past President
Cherri M. Pancake
Chair, SGB Board
Jeff Jortner
Co-Chairs, Publications Board
Joseph Konstan and Divesh Srivastava
Members-at-Large
Nancy M. Amato; Tom Crick;
Susan Dumais; Mehran Sahami;
Alejandro Saucedo
SGB Council Representatives
Sarita Adve and Jeanna Neefe Matthews

BOARD CHAIRS
Education Board
Elizabeth Hawthorne and Chris Stephenson
Practitioners Board
Terry Coatta

REGIONAL COUNCIL CHAIRS
ACM Europe Council
Chris Hankin
ACM India Council
Abhiram Ranade
ACM China Council
Wenguang Chen

PUBLICATIONS BOARD
Co-Chairs
Joseph Konstan and Divesh Srivastava
Board Members
Jonathan Aldrich; Apala Lahiri Chavan;
Tom Crick; Jack Davidson; Chris Hankin;
Mike Heroux; James Larus; Marc Najork;
Michael L. Nelson; Holly Rushmeier;
Bobby Schnabel; Eugene H. Spafford;
Bhavani Thuraisingham; Julie Williamson

DIGITAL LIBRARY BOARD
Chair
Jack Davidson
Board Members
Phoebe Ayers; Yannis Ioannidis;
Michael Ley; Michael L. Nelson;
Louiqa Raschid; Theo Schlossnagle;
Julie Williamson

STAFF
DIRECTOR OF PUBLICATIONS
Scott E. Delman
cacm-publisher@cacm.acm.org

Executive Editor
Diane Crawford
Managing Editor
Thomas E. Lambert
Senior Editor
Ralph Raiola
Senior Editor/News
Lawrence M. Fisher
Web Editor
David Roman
Editorial Assistant
Danbi Yu

Art Director
Andrij Borys
Associate Art Director
Margaret Gray
Assistant Art Director
Mia Angelica Balaquiot
Production Manager
Bernadette Shade
Intellectual Property Rights Coordinator
Barbara Ryan
Advertising Sales Account Manager
Ilia Rodriguez

Columnists
David Anderson; Michael Cusumano;
Peter J. Denning; Mark Guzdial;
Thomas Haigh; Leah Hoffmann; Mari Sako;
Pamela Samuelson; Marshall Van Alstyne

CONTACT POINTS
Copyright permission
permissions@hq.acm.org
Calendar items
calendar@cacm.acm.org
Change of address
acmhelp@acm.org
Letters to the Editor
letters@cacm.acm.org

REGIONAL SPECIAL SECTIONS
Co-Chairs
Jakob Rehof, Haibo Chen, and P J Narayanan
Board Members
Sherif G. Aly; Panagiota Fatourou;
Chris Hankin; Sue Moon; Tao Xie;
Kenjiro Taura; David Padua

WEBSITE
http://cacm.acm.org

WEB BOARD
Chair
James Landay
Board Members
Marti Hearst; Jason I. Hong;
Jeff Johnson; Wendy E. MacKay

AUTHOR GUIDELINES
http://cacm.acm.org/about-communications/author-center

ACM U.S. TECHNOLOGY POLICY OFFICE
Adam Eisgrau
Director of Global Policy and Public Affairs
1701 Pennsylvania Ave NW, Suite 200,
Washington, DC 20006 USA
T (202) 580-6555; acmpo@acm.org

COMPUTER SCIENCE TEACHERS ASSOCIATION
Jake Baskin
Executive Director

EDITORIAL BOARD
EDITOR-IN-CHIEF
Andrew A. Chien
eic@cacm.acm.org
Deputy to the Editor-in-Chief
Morgan Denlow
cacm.deputy.to.eic@gmail.com
SENIOR EDITOR
Moshe Y. Vardi

NEWS
Co-Chairs
Marc Snir and Tom Conte
Board Members
Mei Kobayashi; Rajeev Rastogi;
François Sillion

VIEWPOINTS
Co-Chairs
Tim Finin; Susanne E. Hambrusch;
John Leslie King
Board Members
Virgilio Almeida; Terry Benzel; Michael L. Best;
Judith Bishop; Lorrie Cranor;
James Grimmelmann; Mark Guzdial;
Haym B. Hirsch; Anupam Joshi; Richard Ladner;
Carl Landwehr; Beng Chin Ooi; Francesca Rossi;
Len Shustek; Loren Terveen;
Marshall Van Alstyne; Susan J. Winter

PRACTICE
Co-Chairs
Stephen Bourne and Theo Schlossnagle
Board Members
Eric Allman; Samy Bahra; Peter Bailis;
Betsy Beyer; Terry Coatta; Stuart Feldman;
Nicole Forsgren; Camille Fournier;
Jessie Frazelle; Benjamin Fried; Tom Killalea;
Tom Limoncelli; Kate Matsudaira;
Marshall Kirk McKusick; Erik Meijer;
George Neville-Neil; Jim Waldo;
Meredith Whittaker

CONTRIBUTED ARTICLES
Co-Chairs
James Larus and Gail Murphy
Board Members
Robert Austin; Nathan Baker; Kim Bruce;
Alan Bundy; Peter Buneman;
Premkumar T. Devanbu; Jane Cleland-Huang;
Yannis Ioannidis; Rebecca Isaacs;
Trent Jaeger; Somesh Jha; Gal A. Kaminka;
Ben C. Lee; Igor Markov; m.c. schraefel;
Hannes Werthner; Ryen White;
Reinhard Wilhelm; Rich Wolski

RESEARCH HIGHLIGHTS
Co-Chairs
Shriram Krishnamurthi
and Orna Kupferman
Board Members
Martin Abadi; Amr El Abbadi;
Animashree Anandkumar; Sanjeev Arora;
Michael Backes; Maria-Florina Balcan;
Azer Bestavros; David Brooks; Stuart K. Card;
Jon Crowcroft; Lieven Eeckhout;
Alexei Efros; Bryan Ford; Alon Halevy;
Gernot Heiser; Takeo Igarashi;
Srinivasan Keshav; Sven Koenig;
Ran Libeskind-Hadas; Karen Liu;
Joanna McGrenere; Tim Roughgarden;
Guy Steele, Jr.; Robert Williamson;
Margaret H. Wright; Nicholai Zeldovich;
Andreas Zeller

Association for Computing Machinery (ACM)
1601 Broadway, 10th Floor
New York, NY 10019-7434 USA
T (212) 869-7440; F (212) 869-0481

ACM Copyright Notice
Copyright © 2021 by Association for Computing Machinery, Inc. (ACM). Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and full citation on the first page. Copyright for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or fee. Request permission to publish from permissions@hq.acm.org or fax (212) 869-0481.

For other copying of articles that carry a code at the bottom of the first or last page or screen display, copying is permitted provided that the per-copy fee indicated in the code is paid through the Copyright Clearance Center; www.copyright.com.

Subscriptions
An annual subscription cost is included in ACM member dues of $99 ($40 of which is allocated to a subscription to *Communications)*; for students, cost is included in $42 dues ($20 of which is allocated to a *Communications* subscription). A nonmember annual subscription is $269.

Single Copies
Single copies of *Communications of the ACM* are available for purchase. Please contact acmhelp@acm.org.

ACM ADVERTISING DEPARTMENT
1601 Broadway, 10th Floor
New York, NY 10019-7434 USA
T (212) 626-0686
F (212) 869-0481

Advertising Sales Account Manager
Ilia Rodriguez
ilia.rodriguez@hq.acm.org

Media Kit acmmediasales@acm.org

COMMUNICATIONS OF THE ACM
(ISSN 0001-0782) is published monthly by ACM Media, 1601 Broadway, 10th Floor New York, NY 10019-7434 USA. Periodicals postage paid at New York, NY 10001, and other mailing offices.

POSTMASTER
Please send address changes to
Communications of the ACM
1601 Broadway, 10th Floor
New York, NY 10019-7434 USA

Printed in the USA.

Association for Computing Machinery

PLEASE RECYCLE THIS MAGAZINE

DOI:10.1145/3492320

Vicki L. Hanson

Communicating ACM Priorities

Having not had an opportunity to meet in person with our community since the beginning of COVID-19, I have opted for this less personal way to reach out to you. I recognize that in an organization as large and structurally complex as ACM, communication with all constituencies, while challenging, is essential. I would like to begin this series with some reflections on ACM's core values.

Events of the past two years have raised the social consciousness of us all. There is a desire by many to work in areas of computing that have the potential to improve our society. There is also a desire to be part of a professional organization whose values reflect who we are as individuals. Over time, ACM has defined a set of core values that underlie all our activities and interactions. These values are technical excellence, education and technical advancement, ethical computing and technology for positive impact, as well as diversity, equity, and inclusion.

Here I will focus on diversity, equity, and inclusion (DEI). This core value motivates ACM's five-year goal of *continuing to diversify the ACM global community, providing a welcoming community for all*. The fact that this goal is stated in terms of *continuing* to diversify the community is an acknowledgment, first, that we have made progress, and second, that our work here is not yet done.

For years, ACM's commitment to gender diversity has been deeply woven into the fabric of ACM's committees and activities. Over the past two years, however, many community members have drawn attention to the fact that racial and ethnic equity have not been similarly prioritized. As an organization, it is imperative that ACM promote equity across all our activities by re-examining existing processes that have allowed this situation to persist. We do so recognizing that the need for DEI is not only an issue of fairness, but also of excellence; research consistently finds that diverse teams produce better results than non-diverse teams. Multiple voices, bringing new perspectives, increase both the breadth and depth of ACM's important work.

Fortunately, there are some recent signs of change within the Association. Many ACM boards, councils, committees, and SIGs are actively recruiting and appointing leaders from historically underserved communities. Notably, appointments to several influential positions (for example, awards committees) are more diverse. Many SIGs and other units have developed specific programs to benefit underserved community members. There is progress, but still more to do.

There are some areas where significant change requires more time. For example, in terms of elected leadership, ACM and SIG officers serve for multiple years, and appointments to many other key positions across all volunteer units have multi-year appointments. Given the scope and scale of ACM, visible changes throughout leadership will necessarily unfold over time.

There are other cases where change, while meaningful, is not highly visible. This past year saw questioning of ACM's commitment to diversity, equity, and inclusion. This concern highlights the need to ensure appropriate processes are in place in all ACM nomination and selection processes. The ACM Awards committee, for example, has recently made process changes. In addition, through an ACM Bulletin and letters I sent to leaders of diversity-focused computing organizations, ACM seeks to create a more diverse pool of nominees from which to select award recipients. While these changes may not be highly visible, the changes are very real and directly intended to improve diversity and equity in the awards process, starting with the current 2021 Awards nomination cycle.

ACM's DEI Council is building for sustainable change throughout the organization. The Council will be focusing on increasing diversity representation across all ACM activities, re-examining ACM processes and practices to promote equity, and enhancing the culture to be more inclusive for ACM's global community. Such changes need the efforts of everyone. As a member driven society, not only ACM leadership, but all members of the community can help drive the change we need. In particular, ACM needs the participation of historically underserved members of the community to highlight where processes or activities create barriers to participation or advancement.

But while these voices are desperately needed to highlight problems, the burden for making needed change cannot rest solely on those who have been underserved. It is up to all involved in ACM to address inequities. For those reading this who already are ACM volunteers, consider how your part of the organization addresses diversity, equity, and inclusion. Are there processes or activities that need to be changed or improved? For all members, can you think of new ideas for moving beyond the status quo?

I look forward to continuing this conversation.

Vicki L. Hanson is the Chief Executive Officer of ACM. She also served as ACM President 2016–2018.

Copyright held by author/owner.

DOI:10.1145/3492903

Andrew A. Chien

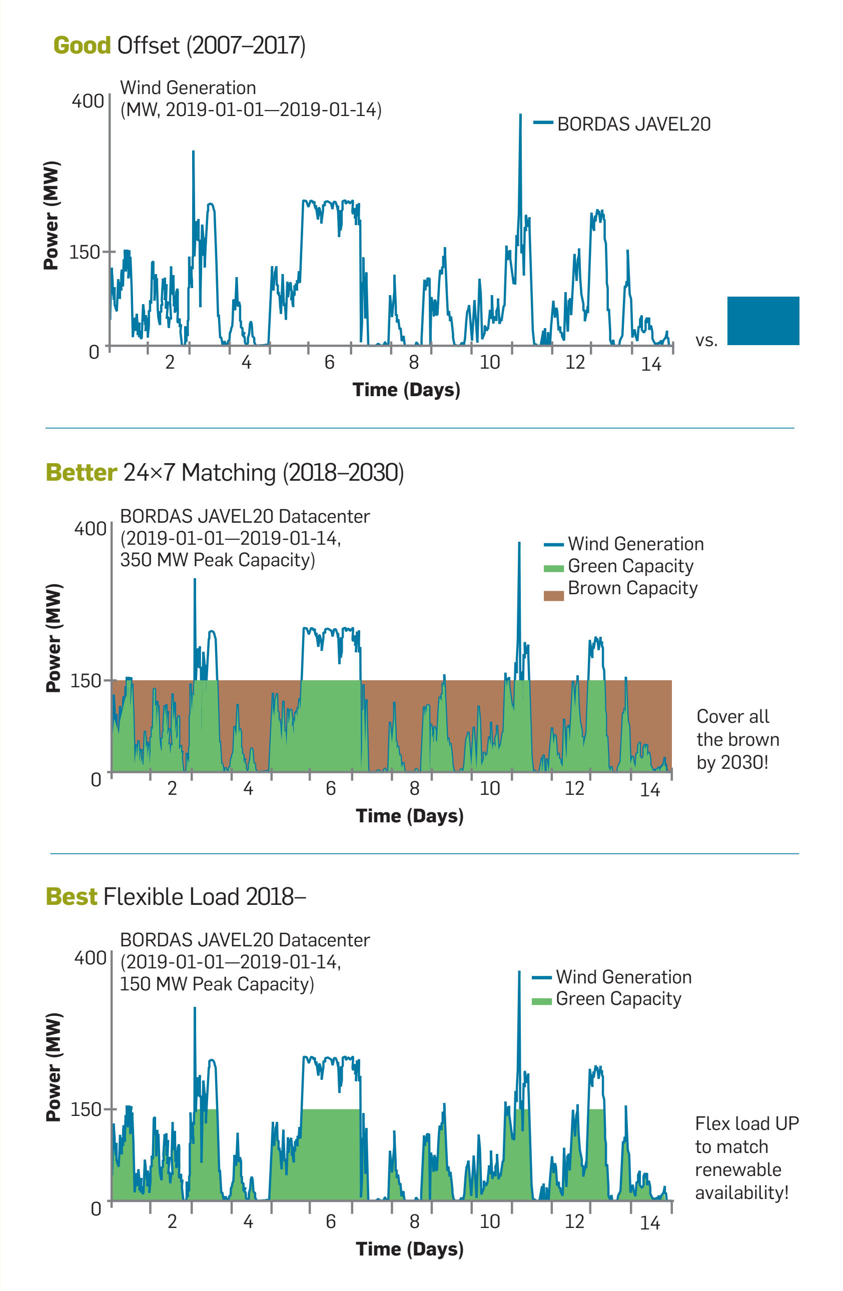
Good, Better, Best: How Sustainable Should Computing Be?

IT HAS BEEN quite a year. Increasing numbers of uncontrolled wildfires and extreme weather events have inspired new awareness and activism around climate change. This awareness has reached a broad range of computing communities: software and system developers, cloud operators, researchers and academics, policymakers, and increasingly business executives. The awakening of the community is evident in common questions: What is the problem? What can I do? Why can't we do that? And there have been especially vibrant debates around AI/ML's growing environmental impact.

As financial pressures on climate risk and reputation grow, computing industry executives have made new commitments to reduce carbon emissions and impact. They are respond-

As financial pressures on climate risk and reputation grow, computing industry executives have made new commitments to reduce carbon emissions and impact.

Models for computing's sustainability



ing to pressure and expectations of investors, peers, and customers. How much should we demand?

In Western capitalism corporations do "what they must" to secure their current and future profits. It falls to our awareness, activism, and engagement to drive the corporate profit calculus. I believe we should expect them to do heroic things—far more than the corporations "think" they can do, or "know how or can price" to do. This is essential to overcome business conservatism. And we can drive that corporate behavior with who we choose to buy from, work for, and respect. And frankly, who we consider to be a climate "greenwasher" or worse "a climate destroyer." So, my answer is that we should set the bar high and expect computing and particularly tech companies reaping billions in annual profits to not just mitigate computing's own damage, but to drive progress on climate change broadly and aggressively. Consider the spectrum depicted in the figure here:

GOOD. Carbon Neutral—companies buy renewable via long-term PPAs. This is perhaps not a sacrifice (many PPAs will save cloud companies money). This is offsetting (like buying trees). But yes, a Good commitment.

BETTER. 24x7 hourly matching—a more ambitious goal.[a] Only one cloud provider has made this commitment. This goal is match generation of renewables to datacenter power consumption synchronously. 24x7 can still be negative for grid renewable absorption and renewable fraction. So, it reflects a reduction of the environmental damage of datacenter operation. A Better commitment.

BEST. Datacenters as flexible loads. Creating a negative grid carbon footprint. By flexing loads at large-scale both temporally and geographically at dynamic ranges of 50% or even 100%, in cooperation with the grid would enable increased renewable absorption. Power grids need this help badly to achieve the radical renewable energy goals proposed for the coming decade(s). This dispatchable datacenter load approach has been studied extensively and is of growing commercial interest.[b] While promising, this approach presents significant research challenges for flexible workloads, computing systems and hardware, that we should embrace.[c]

The grid faces major challenges to decarbonize, and we can and should help accelerate the process.

I propose a call to action—let's help the grid decarbonize, shaping computing activity to accelerate the addition of renewables and more important, the effective absorption of such renewable generation! The grid faces major challenges to decarbonize, and we can and should help accelerate the process. This means research. And change in practice and goals of computing companies—because they can (have technology and resources), they should win business (greener services), and they have a vested interest to ensure their economic future.

In 2022 and beyond, let's challenge and hold ourselves and computing companies to the highest standard of "best," driving to support the accelerated decarbonization of the power grid.

Andrew A. Chien, EDITOR-IN-CHIEF

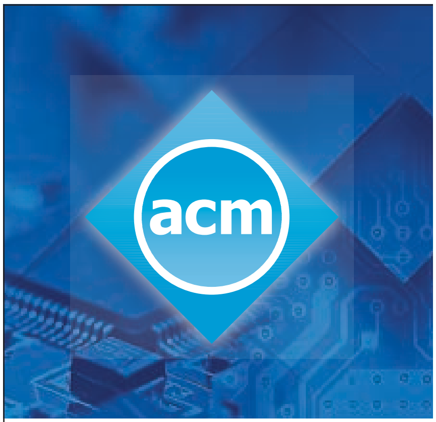
Andrew A. Chien is the William Eckhardt Distinguished Service Professor in the Department of Computer Science at the University of Chicago, Director of the CERES Center for Unstoppable Computing, and a Senior Scientist at Argonne National Laboratory.

Copyright held by author/owner.

[a] Google. The Internet is 24x7—carbon-free energy should be too. Sustainability blog, Sept. 2019.

[b] This idea also goes by "Zero Carbon Cloud" or "Dispatchable Datacenter Loads," see http://zccloud.cs.uchicago.edu/. Lancium breaks ground on First Clean Campus... (325MW), Sept. 15, 2021.

[c] Chien, A.A. Cloud computing is becoming carbon-aware; Can it become a zero-carbon cloud? Intern. Conf. Cloud Engineering, Keynote, Oct. 2021.



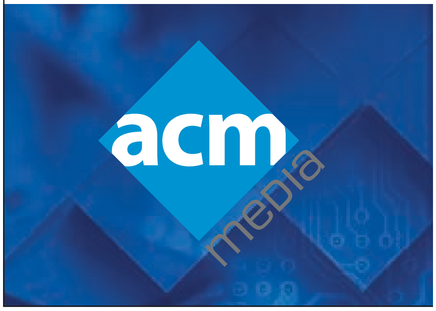
Advertise with ACM!

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
acmmediasales@acm.org



Nominees for ACM's 2022 General Election

In accordance with the Constitution and Bylaws of the ACM, the Nominating Committee hereby submits the following slate of nominees for ACM's officers. In addition to the officers of the ACM, two Members at Large will be elected to ACM Council. In addition to considering previous leadership roles both within and outside ACM, the Committee made an effort to ensure a diversity of perspectives will be represented.

The names of the candidates for each office are presented in alphabetical order below:

President (1 July 2022 – 30 June 2024):
Yannis Ioannidis, "Athena" Research Center and University of Athens
Joseph Konstan, University of Minnesota

Vice President (1 July 2022 – 30 June 2024):
Elisa Bertino, Purdue University
Chris Stephenson, Google

Secretary/Treasurer (1 July 2022 – 30 June 2024):
Theo Schlossnagel, Circonus
John West, Texas Advanced Computing Center

Members at Large (1 July 2022 – 30 June 2026):
Juan Gilbert, University of Florida
Ayanna Howard, Ohio State University
Antonio Vallecillo, University of Málaga
Michelle Zhou, Juji

The Constitution and Bylaws provide that candidates for elected offices of the ACM may also be nominated by petition of one percent of the Members who as of **1 November 2021** are eligible to vote for the nominee. Such petitions must be accompanied by a written declaration that the nominee is willing to stand for election. The number of Member signatures required for the offices of President, Vice President, Secretary/Treasurer, and Members at Large, is **745**.

The Bylaws provide that such petitions must reach the Elections Committee before **31 January 2022**. Original petitions for ACM offices are to be submitted to the ACM Elections Committee, c/o Pat Ryan, COO, ACM Headquarters, 1601 Broadway, 10th Floor, New York, NY, 10019, USA, by **31 January 2022**. Statements and biographical sketches of all candidates will appear in the May 2022 issue of *Communications of the ACM*.

The Nominating Committee would like to thank all those who helped us with their suggestions and advice.

Cherri M. Pancake, CHAIR,
Jennifer Chayes, Enrico Nardelli, Charles Isbell, Thomas Zimmermann

DOI:10.1145/3492896

Vinton G. Cerf

On Heterogeneous Computing

One of the major challenges in the development of the Arpanet was solving the problem of communication between heterogeneous computers. In the late 1960s and 1970s, there were several computer makers and their machines had varying word lengths, binary coding schemes, instruction sets and a plethora of operating systems. The underlying homogeneous network of Interface Message Processors (IMPs), which we would call "routers" or "packet switches" today, offered a uniform interface to the heterogeneous "host" computers connected to the Arpanet. The Network Working Group, led by Stephen D. Crocker, solved the problem by the invention of the Network Control Protocol (NCP) and application protocols such as File Transfer and TELNET (remote terminal access). Coping with heterogeneity is a challenge. The Internet designers tackled the problem of interconnecting heterogeneous packet-switching networks using the TCP/IP Protocol Suite.

In the computing world, the Reduced Instruction Set Computing architecture (RISC) has provided widely adopted instruction set design principles for which David A. Patterson and John L. Hennessy received the prestigious 2017 ACM A.M. Turing Award. Although I am not a hardware designer, I have been struck by the observations of others such as Margaret Martonosi, Assistant Director of the National Science Foundation for Computer, Information Systems and Engineering and Google colleague, Robert Iannucci, that heterogeneity is returning to computer design with concomitant challenges for compiler designers. In addition to RISC-based CPUs, we now see Graphical Processing Units (GPUs), Tensor Flow Processing Units (TPUs), Quantum Processing Units (QPUs), and Field Programmable Gate Arrays (FPGAs) in use or looming on the horizon. Each of these has unique properties that allow for optimal programming solutions to hard (and even NP-hard) problems.

The idea of using a mix of computing capability is by no means new. In the 1950s and early 1960s, my thesis advisor, Gerald Estrin, and his colleagues worked on what they called "Fixed Plus Variable Computing."[a] I have written about this before.[b] This time I want to focus on the challenge for compiler writers to map conventional and new programming languages into functional operation on a variety of programming platforms, bearing in mind their various results and potential parallelism must be accounted for by the compiler. Martonosi points out that testing and analysis must be applied to increase confidence that the physical devices work as intended and that the mapping of a program onto the hardware mix produces the intended computational result. Anyone familiar with the problem of numerical analysis will appreciate that details count. For example, loss of precision in large-scale floating-point computations can deliver erroneous results if inadequate attention is paid to the details of the actual computation.

Computing is an endless frontier in which we have an unending supply of new problems to confront in the search for new solutions.

Among many other considerations, a compiler writer will need to determine how data input or initial state is established for the computing unit in question. How will data be represented? How will it be advantageously transferred to other, heterogeneous computing components in the system? How will the flow of control of the computation be managed if parallel operation is anticipated? What will the "runtime" environment look like? If a computation goes awry, how can this be detected and signaled? In some sense, these are old questions demanding new answers in a more heterogeneous computing environment. Just as the Arpanet and Internet designers wrestled with interoperability, so must the designers of heterogeneous computing environments.

There is something simultaneously satisfying and unsurprising about these questions. Computing is an endless frontier in which we have an unending supply of new problems to confront in the search for new solutions. It is of vital importance to pursue these ideas. "Computational-X," for virtually all scientific values of "X," is part of a paradigm shift that began in the mid-20th century and continues unabated today. Our ability to compute on grander scales and in new ways will have significant influence on the rate at which scientific understanding progresses.

[a] Estrin, G. Organization of computer systems—The fixed plus variable structure computer. In *Proceedings of the Western Joint Computer Conf.* (San Francisco, CA, USA, May 3–5, 1960).

[b] Cerf, V.G. As we may think. *Commun. ACM 58*, 3 (Mar. 2015), 7.

Vinton G. Cerf is vice president and Chief Internet Evangelist at Google. He served as ACM president from 2012–2014.

Copyright held by author.

DOI:10.1145/3485271

Common Ails

I READ WITH INTEREST Michael A. Cusumano's October 2021 column, "Section 230 and a Tragedy of the Commons." As the author of a book about Section 230's history (*The Twenty-Six Words That Created the Internet*, Cornell University Press, 2019), I welcome discussion of this important statute. Unfortunately, Cusumano's column contains some fundamental factual errors that further muddle a debate that already has been rife with inaccuracies.

Perhaps most concerning was Cusumano's characterization of a "specific dilemma" that social media platforms face: "If they edit too much content, then they become more akin to publishers rather than neutral platforms and that may invite strong legal challenges to their Section 230 protections." This statement is false. Section 230 does not have—and never has had—a requirement for platforms to be "neutral." To the contrary, Section 230's authors were motivated by a 1995 state court ruling that suggested that online services receive less protection from liability for user content under the common law if they exercise "editorial control." As Sen. Ron Wyden, Section 230's co-author, told Vox in 2019: "Section 230 is not about neutrality. Period. Full stop. 230 is all about letting private companies make their own decisions to leave up some content and take other content down." Congress provided platforms with Section 230 protections to give them the breathing room to develop moderation technology, policies, and practices that they believe their users' demand. As the first federal appellate court to interpret Section 230 wrote in 1997, under the statute, "lawsuits seeking to hold a service provider liable for its exercise of a publisher's traditional editorial functions—such as deciding whether to publish, withdraw, postpone or alter content—are barred."

Cusumano writes that Section 230 "makes it difficult to hold these companies accountable for misinformation or disinformation they pass on as digital intermediaries." Yet Cusumano does not identify any specific types of legal claims related to misinformation that Section 230 bars. That is because the U.S. does not have a general "anti-misinformation law," nor could it due to our strong First Amendment protections. To be sure, the First Amendment permits certain causes of action related to false speech, such as defamation and false advertising, but the courts have set a high bar for these claims. For good reason, the Supreme Court has ruled that Congress cannot impose a blanket prohibition on all false speech. In recent years, some authoritarian countries have enacted anti-misinformation laws, allowing the government to determine what is permissible and what is "fake news." The U.S. would never be able to have such a law unless the courts were to radically reinterpret the First Amendment.

Cusumano argues that "the U.S. Justice Department or the U.S. Congress must amend Section 230 to reduce the blanket protections offered online platforms." Fortunately, neither the Justice Department nor any other component of the executive branch has the unilateral authority to amend a statute. Of course, Congress can amend Section 230, and in the past few years, we have seen dozens of proposals to do so. The debate about these proposals is vital to the future of the Internet. But it must occur with an accurate understanding of what Section 230 says and does.

Jeff Kosseff, Annapolis, MD, USA

With regard to Michael Cusumano's column the October issue of *Communications*, it is worth noting that legally the first definition given for "publish" is: "to make known to another or to the public generally" (see https://bit.ly/3vCUmsf). Section 230 created a legal fig leaf for the benefit of the then-embryonic social media platforms, which allowed them, for better or worse, to become what they are today. Now, though, there is no reason to ignore the plain meaning of the legal term: social media companies make content known to the public generally. They are, by the legal definition, publishers. There is no reason to allow them to evade responsibility for curating the content they publish, and very strong reason to insist, legally, that they do so. We have laws, albeit imperfect, to control damaging pollution in general and hazardous chemicals in particular. Lies about vaccines, or the integrity of the 2020 Presidential Election, are the exact equivalent in the information sphere and should be treated as such, by appropriate alterations to Section 230.

H. Joel Jeffrey, DeKalb, IL

Author's response:

I appreciate all the comments about my October 2021 Technology Strategy and Management column, both negative and positive. One observation is that lawyers, as well as managers, academics, and politicians, seem to disagree over how to interpret the legislation drafted 25 years ago. My main takeaway from reader comments is that the focus on Section 230 may be a distraction. My column is primarily about the damage already done as a result of widespread misinformation and disinformation on the Internet. And so I also argue that social media platforms need to do more to regulate the content they disseminate. This might very well mean that social media platforms not only need to start behaving more like traditional publishers but that we need to view them as such. Meanwhile, I argue that the Internet as a dependable platform for information exchange has been damaged. There is a moral hazard here if there are no consequences for the dissemination—the publication—of dangerous falsehoods and outright lies. In this sense, I believe we are facing a potential tragedy if we view the community of Internet users as a common resource.

Both right-wing and left-wing critics of social media blame at lot of these problems on Section 230, whether or not they should. I noted that ex-President Trump and others have criticized Section 230 based on the argument that social media platforms are already behaving as publishers because they censored so much content and so they should not be afforded any Section 230 protections. U.S. presidential candidate Biden also wanted to revoke Section 230 in order to *hold the social media platforms responsible for the content they disseminate. When both sides of a polarized political spectrum agree that Section 230 is problematic, then clearly we have something to fix.*

In the column, I noted that Section 230 allows platforms to set their own "terms of service" and therefore to edit or curate content that, in their view, violates those terms of service. So yes, I understand that platforms are not required by law to be neutral. In practice, though, the social media companies have behaved as if they are not responsible for the content they disseminate and they have been reluctant to edit. Only at the last moments in the recent U.S. presidential campaign did we see the main social media platforms start to ban accounts and tag content as unreliable. Why so late? In part, it seems they do not want to become embroiled in legal challenges and they are being cautious in what they censor. More importantly, perhaps, spectacular content often goes viral and generates huge advertising profits. By contrast, it is possible to hold traditional publishers accountable for the content they publish. Should we hold social media platforms accountable as well and, if so, how, given the First Amendment and other laws? Section 230, like it or not, is at the center of these debates. The Department of Justice a year ago started drafting proposed revisions of Section 230 (see https://bit.ly/30ineds). The Biden administration has been reviewing Section 230 as well and reversed a Trump executive order that empowered the Dept. of Commerce and the FCC to investigate "selective censorship" and requested the DOJ to draft legislation curtailing Section 230 protections. At the very least, the Biden administration seems intent on clarifying what the law actually says (see https://bit.ly/3BKvDUS). Of course, whether Section 230 is revoked, revised, or left alone is ultimately a decision for Congress, not the Executive Branch. But the real issues are rising mistrust in Internet content and the need for the social media platforms to take more responsibility for the misinformation and disinformation they disseminate and amplify.

Michael Cusumano, Cambridge, MA, USA

Editor-in-Chief's response:

Great to see a thoughtful and hearty debate on these issues. As computing has become a critical intermediary for discourse and really all of society, these issues are essential to the society we are becoming and hope to become! ACM should be at the heart of this debate.

Andrew A. Chien, Chicago, IL, USA

Putting the "I" in Phones

Many tinkerers, including myself, have started to independently and creatively explore the space of self-built smartphones [see "Whose Smartphone Is It?" by James Larus, Sept. 2021, p. 41—*Ed.*]. In my case, a Raspberry Pi, a matching 4-inch touchscreen display, a USB power bank, a USB headset, a Wi-Fi connection, and an account with a VoIP provider have allowed me to make phone calls and run interesting apps on a variety of Linux-based operating systems.

If you are a reader of this publication, you likely have the ability to do the same, changing the details of the system to suit your own needs and desires. For example, cellular connectivity can come from a cellular board or a USB dongle or a hotspot. Changing screens can turn a phone into a tablet, adding a keyboard can turn it into a laptop or a desktop. There is an astounding variety of hardware boards, operating systems, browsers, apps, and VoIP providers to choose from. Many apps not directly available for your OS, including WhatsApp, are available through a Web browser.

Building your own smartphone takes some time and energy, but less money than required to buy a mainstream smartphone. The more people build and use their own smartphone, the easier it will get—there will be more and better designs and hardware and software available. VoIP providers will improve their support for features such as MMS and for making voice calls through open source (rather than proprietary) SIP software.

Let's encourage everyone to build a smartphone that can be truly theirs.

E. Biagioni, Manoa, HI, USA

Author's response:

Good luck and have fun! It will be challenging to build a mobile device competitive with the highly engineered systems currently available. I would encourage readers also to support governments in their efforts to open these closed systems to permit honest and fair competition.

James Larus, Lausanne, Switzerland

Editor-in-Chief's response:

The extreme complexity and integration of modern smartphones is major challenge. Perhaps recent advances on the "right to repair" and accelerating initiatives in open source hardware will enable progress in creating modular if not ultimately "open source" smartphones.

Andrew A. Chien, Chicago, IL, USA

Lipstick on a Pig?

The article on the Frama-C Platform ("The Dogged Pursuit of Bug-Free C Programs: The Frama-C Software Analysis Platform," June 2021—*Ed.*) was interesting. While impressive work, it begs the question, why? The authors note that C is a difficult language. C has many flaws, traps, and problems. Dennis Ritchie admitted to C being quirky and flawed. C burdens programmers focusing on machine details. C is a system language, but even for system programming C exposes details that are tedious, error-prone and dangerous. John McCarthy noted the value of checking in his 1963 paper 'A Basis for A Mathematical Theory of Computation.' Checks should be integrated in languages. C has had external tools like lint. Frama-C is another addition. The Frama-C work is based on Design by Contract (DbC) by Bertrand Meyer, based on Hoare logic. Frama-C ASCL syntax looks like the lipstick of DbC stuck on a pig. While C was designed by Dennis Ritchie and Ken Thompson, C is mostly BCPL—credit should go to Strachey and Richards. C invented very little (#define is from Burroughs ALGOL—a suggestion of Don Knuth). The article also says C gives developers freedom, but in C, it is misguided freedom. 'Freedom' is spin for burden that C does not remove from programming. Easing programming is one of the major reasons for programming languages. C creates lock in—the opposite of freedom. C results in inflexible software, which is technical lock in. C culture creates mental lock in—technical criticism of C results in overheated rejection.

The article notes that formal methods can be used to address the shortcomings of C but implementing in C is difficult. Undefined behavior in C results in crashes, memory corruption, or arbitrary results. Intentional memory corruption compromises system security. This is not acceptable in modern systems. Programmers should not be able to mess with memory and addresses to undermine the execution model. Such freedom is the tool of malicious hackers. Security is the utmost problem today. Programmers should not have freedom to harm users. C culture says 'trust the programmer'—not just stupid, but criminally negligent. High-level programming deals with problem-oriented data, contents, and semantics, rather than machine-oriented memory of the container locations or access paths. C focuses on the container rather than contents. Hackers will ignore techniques for correctness and security. Computing needs fundamental fixing, not patching flawed legacy.

The article also says C is widely taught and that with Frama-C, good practices can be taught. C and C++ teach many wrong lessons—diverting students from good practices that are integrated in better languages. Learning C is about dealing with flaws and traps. Institutions should stop teaching C and C++ as foundational languages. Retrofitting DbC onto C is like adding drop-down oxygen masks to a 1920s biplane. Students should be taught languages with direct, clear, and clean support for DbC, not something hacked on to old and flawed languages as lipstick on a pig.

Ian Joyner, Sydney, Australia

Hypercriticality, Hypocriticality, and Hyperempathy

The phenomenon of harsh reviewing, sometimes called hypercriticality, has already been discussed in the *Communications*' community. I would like to address what is at stake. If unfair, or worse malevolent, criticism should be absolutely banished, we should keep in mind that hypercriticality is part of the scientific ethos, at least if the prefix hyper is understood with respect to more mundane matters. Moreover, in the age of publish or perish we need gatekeepers to avoid conferences, journals, and grant funding schemes to be flooded with questionable writings. Given that reviewers are put under pressure and loaded with many reviewing tasks with tight deadlines, the "three positives for every negative" rule of thumb is unrealistic.

Denouncing hypercriticality often takes implicitly the authors' side. What about the readers' side? The well-meaning will to avoid hypercriticality may become a refuge for hypocriticality. Instead, reviewing should be an exercise in hyperempathy with potential readers as we shall explain. Complacency with authors would be a disservice to readers, since, in accordance with the principle of communicating vessels, the less work for an author, the more work for their readers. The burden should be on authors. As a reviewer, you should make the following maxim yours: Many things that ease the life of an author are as many pebbles for readers to stumble over. Reviewing is a precarious balance, since the interests of the author and of the readers are almost always divergent, at least when these interests are superficially understood. If the reviewer must choose, they should side with readers, the silent majority. One should never disregard this silent majority, the final destination of every writing. In the academic world, the author is this ordinary hero who stands against all, with the opportunity of shining, hence they should never forget the responsibility that comes with such a lofty position. Authors should not overlook the dangers of their ego being bruised, nor overdramatize the consequences.

A final plea. Reviewers, please write reviews as if you were an authentic, self-motivated, and innocent reader. Every review should be an exercise in hyperempathy with readers, who are, after all, on the receiving end in case of publication. Don't be afraid that your hyperempathy with readers comes across as an hypercriticality against authors. What comes across as hypercriticality from the author's perspective can simply originate from an hyperempathy siding with the best interests of future readers. A lack of empathy with readers often drapes itself in the sheep's clothing of hypocriticality. Be critical but be fair. Be fair but be critical. Authors, please do not forget that writing should be an act of hyperempathy in the first place and that your reviewers are your first readers.

Anthony Bordg, Cambridge, U.K.

Verifying Verification

I would like to add some additional reality to Moshe Vardi's "Program Verification: Vision and Reality" (July, 2021). The cyber-physical systems, for example, the Boeing 737 Max-8, we are building today interact with the physical world, which includes humans who may participate in the systems' operation. By "verification," Vardi means formal verification of formal properties of formal mathematical objects. When the program to be verified, for example, a compiler, a word processor, a theorem prover, has no interaction with the physical world, the theorems to be proved *are* formal mathematical objects.

However, for cyber-physical systems, the theorems to be proved *not* formal mathematical objects. The theorems are, at their core, mathematical models of the physical world. Our experience with the sciences, from physics through psychology, says these models are never correct, but are just approximately correct. A cyber-physical system relying on the correctness of a verified one of these models cannot be relied on not to fail. In general, verified mathematical correctness for a cyber-physical system is not even meaningful. Verification in Vardi's sense would have done nothing for the Boeing 737 Max-8, because of the human failings and its incorrect assumptions about flying.

Albert Einstein said many years ago "As far as the propositions of mathematics refer to reality, they are not certain; and as far as they are certain, they do not refer to reality." (Albert Einstein; Geometry and Experience; an address on 27 January 1921 at the Prussian Academy of Sciences in Berlin, translated to English, Methuen, London, 1922).

Daniel M. Berry, Waterloo, ON, Canada

Author's response:

To a certain extent I do agree with Berry. As I wrote in the column: "In retrospect, the hope for 'mathematical certainty' was idealized, and not fully realistic, I believe." Yet, I do not agree that verification of cyber-physical systems is meaningless. I encourage Berry to read "Formally verified software in the real world," in the September 2018 of Communications.

Moshe Y. Vardi, Houston, TX, USA

Communications welcomes your opinion. To submit a Letter to the Editor, please limit your comments to 500 words or less, and send to letters@cacm.acm.org

© 2021 ACM 0001-0782/21/12 $15.00

ACM BOOKS

Collection II

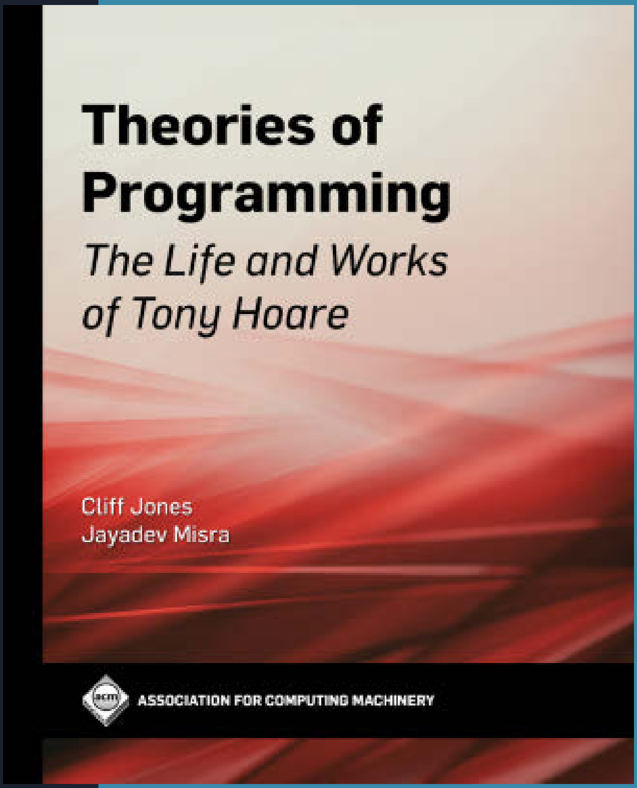
Sir Tony Hoare has had an enormous influence on computer science, from the Quicksort algorithm to the science of software development, concurrency, and program verification. His contributions have been widely recognized: He was awarded the Turing Award in 1980, the Kyoto Prize from the Inamori Foundation in 2000, and was knighted for "services to education and computer science" by Queen Elizabeth II of England in 2000.

This book presents the essence of his various works—the quest for effective abstractions—both in his own words as well as chapters written by leading experts in the field, including many of his research collaborators. In addition, this volume contains biographical material, his Turing Award lecture, the transcript of an interview and some of his seminal papers.

Hoare's foundational paper "An Axiomatic Basis for Computer Programming," presented his approach, commonly known as Hoare Logic, for proving the correctness of programs by using logical assertions. Hoare Logic and subsequent developments have formed the basis of a wide variety of software verification efforts. Hoare was instrumental in proposing the Verified Software Initiative, a cooperative international project directed at the scientific challenges of large-scale software verification, encompassing theories, tools, and experiments.

Tony Hoare's contributions to the theory and practice of concurrent software systems are equally impressive. The process algebra called Communicating Sequential Processes (CSP) has been one of the fundamental paradigms, both as a mathematical theory to reason about concurrent computation as well as the basis for the programming language occam. CSP served as a framework for exploring several ideas in denotational semantics such as powerdomains, as well as notions of abstraction and refinement. It is the basis for a series of industrial-strength tools that have been employed in a wide range of applications.

This book also presents Hoare's work in the last few decades. These works include a rigorous approach to specifications in software engineering practice, including procedural and data abstractions, data refinement, and a modular theory of designs. More recently, he has worked with collaborators to develop Unifying Theories of Programming (UTP). Their goal is to identify the common algebraic theories that lie at the core of sequential, concurrent, reactive, and cyber-physical computations.



Theories of Programming

The Life and Works of Tony Hoare

Cliff Jones
Jayadev Misra

ASSOCIATION FOR COMPUTING MACHINERY

Theories of Programming

The Life and Works of Tony Hoare

Edited by
Cliff Jones
Jayadev Misra

ISBN: 978-1-4503-8729-3
DOI: 10.1145/3477355

http://books.acm.org
http://store.morganclaypool.com/acm

BLOG@CACM

The *Communications* website, 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.

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

DOI:10.1145/3490151 http://cacm.acm.org/blogs/blog-cacm

Seeking Out Camille, and Being Open to Others

Robin K. Hill on overcoming biases against alternative views, and Carlos Baquero on his search for the elusive Camille Noûs.

Robin K. Hill
Safe Space for Alt-Views

https://bit.ly/3DfOrvN
September 27, 2021

Some of us are skeptical that recommender systems can detect their own biases and overcome them. Some of us are skeptical that either generative grammars or phrase substitution systems will ever speak any natural language fluently. Both claims challenge techno-optimism by asking why computers can't do what we do. But those challenges are not the subject of argument here. The subject is the alternative space available to such skeptics.

Claims of the power of artificial intelligence, or the success of language translation, or of the inevitable emergence of machine consciousness or volition are the premises driving much artificial general intelligence (AGI) research. Some weaknesses of those premises stand out pretty well: A program can't overcome bias unless it's programmed to look for bias in a particular attribute; a computer cannot power up itself and cannot process an interruption unless it is already checking for it. When someone (myself, for instance) raises these mundane objections, the reactions from AI boosters are often directed not against the objections per se, but rather against some form of anti-intellectualism. Skeptics are seen as propounding a religious, mystical, or magical stance. No. Far from it.

The space of alternative views is vast, not simply mud puddles where notions of soul and spirit taint the discipline of logic, but strong currents flowing every which way. Can't we allow, invite, and cultivate other paradigms, without putting up obstacles of dogma? Recall non-numeric reasoning, such as the geometric proof that an angle can be bisected with a straightedge and compass. Those methods, which do not depend on symbolic logic, preceded our systems of arithmetic and algebra, but their standing has eroded.[2] Of course, earnest attempts to transcend logic, math, and other rigorous systems encounter many pitfalls. Gödel's proofs, forced into awkward, debased, or metaphorical applications to philosophical questions, have been abused by many.[1]

To be clear, in protecting alternative views, we do not seek a particular theory, such as the Penrose-Hameroff theory of Orchestrated Objective Reduction.[3] The well-developed and quite particular theories of prominent philosophers of mind have spun off into the weeds, if it's fair to apply that figure of speech to the level of detail reflected in the discussion among, say, Jerry Fodor and his critics.[4] We want a place to refresh, a refuge for explanations of human cognitive phenomena that are novel or familiar, commonsensical or radical. Refuge from what? The Turing-computable? The digital? The discrete? The formalizable? Hard to say; hence, we avoid particularities.

All of this is meant not to close off lines of inquiry, but to illuminate the many that are open. (1) There may be an alternative other than magic. (2) There may be no alternative other than magic. (3) The alternative we now call "magic" may turn out to be something rigorous and respectable in forms that we cannot yet conceive. Or even (4) No alternative is needed; the current paradigm will work when it matures. Techno-optimism may be correct. It could turn out that there *is* a way to augment Good Old-Fashioned Artificial Intelligence, or data science, or deep learning, or the neural model, so that computers can do what we do. That way may be chemistry, or it may be quantum physics, or it may be geometry. That way may favor one of the weedy theories of philosophy of mind. Or both standard and alternative views (and more?) could play their parts in some harmonious whole. All welcome! We wish only to forestall the reaction of the Pythagoreans to the prospect of irrational numbers; that is, condemning the idea and its proponents. Let's react as did later mathematicians: they accepted the existence of numbers that could not be expressed as rationals, and dubbed them, in a stroke of brilliant unorginality, "irrational numbers."

That suggests that the alternative space could be circumscribed by giving it a name… *ubereason* (pompous), *extracomp* (unattractive). Or words from Latin such as "humilis," lowly, humble, literally "on the ground," from humus "earth," from Proto-Indo-European root *dhghem- "earth", which is also the root of "human." Or "crete," as opposed to "discrete," that is, solid as opposed to divisible. Well… this is good fun, but none of these notions are compelling. Either we are not as brilliantly prosaic as the post-Pythagoreans, or the naming exercise is premature because we cannot articulate the circumscription of "alternative" until we answer, "alternative to what?" But the very idea, the very possibility, the very question, points toward a safe space for alternatives.

The trend in computing is to subsume the humanistic in the technical. The focus and confidence of Tech sheds a glow of affirmation, which casts outer levels of interpretation into shadow. But those of us who believe in the power of AGI to triumph and make the world a better place need not treat those of us who question that belief as eccentrics. It is an inquiry, not a heresy. Let's get ready, when the time comes, to name the alternative space, declare victory, and move on.

References

1. Franzén, T. *Gödel's Theorem: An Incomplete Guide to Its Use and Abuse* (1st ed.). A K Peters/CRC Press. DOI 10.1201/9781568815008.
2. Kline, M. *Mathematics: The Loss of Certainty*. Barnes and Noble, New York, 2009 Edition by arrangement with Oxford University Press. ISBN 9781435108479.
3. Paulson, S. Roger Penrose On Why Consciousness Does Not Compute. Nautilus. *Nautilus Think, Inc.* Issue 47; May 4, 2017.
4. Rescorla, M. *The Computational Theory of Mind.* The Stanford Encyclopedia of Philosophy (Fall 2020 Edition), Edward N. Zalta (Ed.).

Carlos Baquero
We Are Camille

https://bit.ly/3kqad9y
September 20, 2021

It was a Friday afternoon and I was reading a distributed systems paper. The subject was very close to my field of research and had three authors. The first two were affiliated to a known French institution, but the third had one I have never seen before: *Laboratoire Cogitamus, France*.[5] The name was slightly odd since it is a Latin word that means We Think (*Cogitamus ergo sumus* is the plural of the *cogito ergo sum* quote from René Descartes, "I think, therefore I am").

Not too worried with the etymology of the affiliation, it was just an odd impression at the time, I decided to search for the third and last author on Google Scholar to see which other papers he or she had. (I know, I should have been reading the paper before wasting time with Googling the authors.) The scholar profile of Camille Noûs[11] shows papers since 2019, about 200 citations and an h-index of 6. More striking is that Camille, in less than two years of activity, now counts more than half a thousand papers with a wide breadth of subjects that is more typical of a renaissance author. The explanation to this inhuman productivity is quite simple: Camille Noûs does not exist, at least as a human.

Actually, Camille Noûs has existed as an idea since March 2020.[6] It was created by a French research advocacy group, RogueESR.[10] The idea is that Camille symbolizes the anonymous researcher that did not make it into the author list, but influenced and enabled the research. It could had been that five-minute talk on a coffee break that sparked an idea, or the technician that made sure the gene-sequencing machine kept working at night and the email arrived on time. The name itself, Camille, is gender-neutral, and Noûs ("we") can be seen as us, the collective. We all stand on the shoulders of giants, but until mid-2020 the giant did not have a name; maybe now it has a proper one.

Another apparent objective of the initiative is to show the fragility of evaluating author merit and production merely as a function of a few numeric metrics. It also exposes the danger of elevating individual production at the cost of collaboration. These are not fringe concerns, the DORA declaration[8] provides several guidelines to improve research evaluation and to recenter on the scientific merits of each work, and not rely exclusively on bibliometrics.

Adding Camille as an author is not without consequences; papers have been withdrawn from journals,[7] and it would be hard to argue in what capacity did a non-existing person contribute to a specific article (like this one, as a matter of fact). What can be a funny statement from a tenured professor can be a strange CV item to explain when applying for a position.

However, once an idea is created and spreads collectively, it becomes hard to stop. It is very likely that Camille Noûs will continue publishing prolifically. Paul Erdos[9] published around 1,500 papers during his lifetime. He was a generous collaborator who often visited other scientists to pick their brains with new problems and conjectures. Nowadays, scientists pay homage to Erdos by calculating the size of their authorship path to him; the closer, the better.

As Camille keeps publishing, the number of papers will not take long to pass 1,500, and the Camille collaboration graph will keep increasing and getting more tightly connected. Maybe one day the Noûs number will tell each author how close they are to the collective effort of everyone else.

References

5. Cogitamus Laboratory, www.cogitamus.fr.
6. Noûs, C. We, Camille Noûs – Research as a Common, *3 Quarks Daily*, April 19, 2021,
7. O'Grady, C. Who is Camille Noûs, the fictitious French researcher with nearly 200 papers? *Science*, March 16, 2021
8. San Francisco Declaration on Research Assessment, sfdora.org/read
9. Homan, P. Paul Erdos, Hungarian mathematician, https://www.britannica.com/biography/Paul-Erdos
10. Nous sommes RogueESR. rogueesr.fr
11. Google Scholar, Camille Noûs profile, https://scholar.google.com/citations?hl=en&user=368e0dwAAAAJ

Robin K. Hill is a lecturer in the Department of Computer Science and an affiliate of both the Department of Philosophy and Religious Studies and the Wyoming Institute for Humanities Research at the University of Wyoming. She has been a member of ACM since 1978. **Carlos Baquero** is a professor in the Department of Informatics Engineering within the Faculty of Engineering at Portugal's Porto University, and is also affiliated with INESC TEC. His research is focused on distributed systems and algorithms.

© 2021 ACM 0001-0782/21/12 $15.00

IPDPS 2022 Lyon, France

Lyon, France • May 30 - June 3, 2022

36th IEEE International Parallel and Distributed Processing Symposium

ipdps.org

ANNOUNCING 24 IPDPS 2022 WORKSHOPS

IPDPS Workshops are the "bookends" to the main conference technical program of contributed papers, invited speakers, and student programs. They provide the IPDPS community an opportunity to explore special topics and present work that is more preliminary. Each workshop has its own website and submission requirements, all later than the main conference dates. To enrich the 2022 workshops offering, eleven of them (see *) are planned as half-day events.

IPDPS 2022 WORKSHOPS MONDAY 30 MAY 2022

HCW	Heterogeneity in Computing Workshop
RAW	Reconfigurable Architectures Workshop
HiCOMB	High Performance Computational Biology
GrAPL	Graphs, Architectures, Programming, and Learning
EduPar	NSF/TCPP Workshop on Parallel and Distributed Computing Education
AsHES	*Accelerators and Hybrid Emerging Systems
APDCM	Advances in Parallel and Distributed Computational Models
HIPS	High-level Parallel Programming Models and Supportive Environments
QCCC	*Quantum Classical Cooperative Computing
CGRA4HPC	Coarse-Grained Reconfigurable Architectures for HPC
AIDO	*AI for Datacenter Operations

IPDPS 2022 WORKSHOPS FRIDAY 3 JUNE 2022

PDCO	*Parallel / Distributed Combinatorics and Optimization
JSSPP	Job Scheduling Strategies for Parallel Processing
PDSEC	Parallel and Distributed Scientific and Engineering Computing
iWAPT	*Automatic Performance Tuning
PAISE	Parallel AI and Systems for the Edge
RADR	*Resource Arbitration for Dynamic Runtimes
ScaDL	Scalable Deep Learning over Parallel And Distributed Infrastructures
ESSA	*Extreme-Scale Storage and Analysis (formerly HIPS)
ParSocial	*Parallel and Distributed Processing for Computational Social Systems
EDAML	*Electronic Design Automation and Machine Learning
COMPSYS	*Composable Systems
CORtEX	Computing using EmeRging Exotic AI-Inspired Systems
ExSAIS	*Extreme Scaling of AI for Science

Sponsored by

IEEE COMPUTER SOCIETY TCPP Technical Committee on Parallel Processing

acm In-Cooperation

ACM SIGARCH | sighpc

GENERAL CO-CHAIRS

Anne Benoit (ENS Lyon, France)
Laurent Lefèvre (Inria & ENS Lyon, France)

PROGRAM CO-CHAIRS

Yves Robert (ENS Lyon, France & Univ. of Tennessee, USA)
Bora Uçar (CNRS, Laboratoire LIP, Lyon, France)

WORKSHOPS CHAIR AND VICE-CHAIR

Ananth Kalyanaraman (Washington State University, USA)
Suren Byna (Lawrence Berkeley National Laboratory, USA)

STUDENT PROGRAMS

IPDPS 2022 will hold the traditional PhD forum of poster presentations, to provide mentoring in scientific writing and presentation skills, and to create opportunities for students to hear from and interact with senior researchers attending the conference. Visit ipdps.org for details.

IMPORTANT DATES

Conference Preliminary Author Notification

- **December 10, 2021**

Conference Final Author Notification

- **January 22, 2022**

Workshops' Call for Papers Deadlines

- **Most Fall in Late December and January**

IPDPS 2022 VENUE

In 2022, IPDPS will return to meeting in person and will do so in the European metropolis of Lyon. Located in the South-East of France, the Auvergne-Rhône-Alpes region, Lyon is 2 hours by TGV from Paris and less than 2 hours from the Mediterranean coast. The Lyon-Saint Exupéry airport connects to 115 international destinations. IPDPS 2022 is preparing for possible covid-related scenarios, including a hybrid format. Regardless of the form of presentation, in person or virtual, every peer reviewed paper accepted for the main conference and workshops will be published in the proceedings.

Science | DOI:10.1145/3490155

Don Monroe

Trouble at the Source

Errors and biases in artificial intelligence systems often reflect the data used to train them.

MACHINE LEARNING (ML) systems, especially deep neural networks, can find subtle patterns in large datasets that give them powerful capabilities in image classification, speech recognition, natural-language processing, and other tasks. Despite this power—or rather because of it—these systems can be led astray by hidden regularities in the datasets used to train them.

Issues occur when the training data contains systematic flaws due to the origin of the data or the biases of those preparing it. Another hazard is "overfitting," in which a model predicts the limited training data well, but errs when presented with new data, either similar test data or the less-controlled examples encountered in the real world. This discrepancy resembles the well-known statistical issue in which clinical trial data has high "internal validity" on carefully selected subjects, but may have lower "external validity" for real patients.

Because any good ML system will find the same regularities, redesigning it may not solve the problem. Therefore, researchers and companies are looking for ways to analyze and improve the underlying data, including supplying additional "synthetic" data for training.

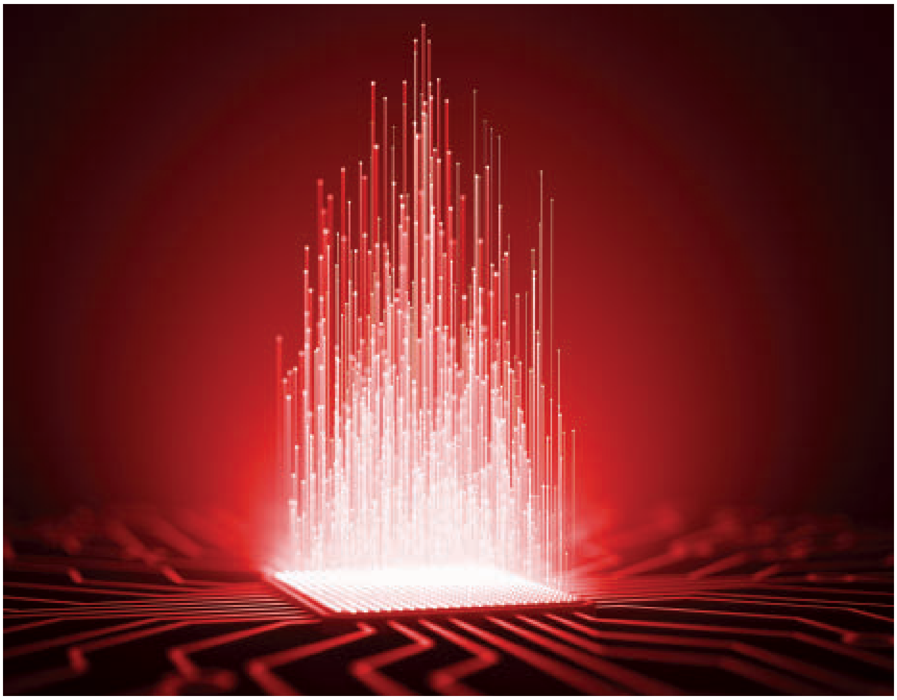


IMAGE BY KTSDESIGN

Errors and Biases

Distortions embedded in artificial intelligence systems have profound consequences for people applying for a loan or seeking medical treatment. To promote greater accuracy, as well as confidence, a growing community is demanding greater fairness, accountability, and transparency (FAT) in artificial intelligence, and holds a regular ACM conference (the ACM Conference on Fairness, Accountability and Transparency (ACM FAccT).

Many experts, however, "tend to focus innovation on models, and they forget that in some ways models are just a mirror of the data," said Aleksander Madry, a professor of computer science at the Massachusetts Institute of Technology (MIT) and director of the MIT Center for Deployable Machine Learning. "You really need to intervene on data to make sure that your model has a chance of learning the right concepts."

"Seemingly innocent things can influence how bad in bias the models are," Madry said. ImageNet, for example, a huge set of labeled images that is widely used for training, was drawn from the photo-sharing site Flickr, and its examples strongly suggest the natural habitat for a crab is on a dinner plate. More seriously, medical images showing tuberculosis frequently come from less-developed countries whose older imagers have digital signatures that systems learn to associate with the disease.

"We are just scratching the surface" in assessing how prevalent these unintentional errors are, Madry said, but there are "definitely more than we [have a right to] expect." His group also has explored how arcane details of labeling protocols can lead to surprising classifications, which machine learning tools then must reverse-engineer, along with their other goals.

These labeling problems can also reflect—and seemingly validate—the social biases of the human annotators. For example, a woman in a lab coat may be more frequently labeled as a "nurse" than as a "doctor" or "chemist." Many training sets rely on online workers from Amazon's Mechanical Turk program to label the data, which poses its own reliability and bias problems. "How do you annotate this data in a way that you are not leaking some biases just by the choice of the labels?" Madry asked. "All of this is very much an open question at this point and something that needs to be urgently tackled."

Many training sets rely on online workers from Amazon's Mechanical Turk program to label the data, which poses its own reliability and bias issues.

Audit Trails

AI systems can learn sexism and racism simply by "soaking up data from the world," said Margaret Mitchell, who was a leader of Google's Ethical AI research group until her acrimonious departure in February. In addition to being unfair, systems embodying these biases can fail at their primary goal, instead wasting resources by inaccurately ranking candidates for a loan or a job.

In addition, the most available and widely used datasets may include systematic or random errors. "Dataset creation has been very chaotic and not really well formed," Mitchell said, so it "incorporates all kinds of all things that we don't want, garbage and bias, and there isn't a way to trace back problematic sources."

Mitchell and her former Google colleagues advocate more systematic documentation at each stage of dataset assembly. This effort parallels growing mandates in other scientific fields that

Milestones

Hennessy, Patterson Receive Frontiers of Knowledge Award

The BBVA Foundation, the corporate social responsibility organization of multinational financial services company Banco Bilbao Vizcaya Argentaria, S.A., has awarded its Frontiers of Knowledge award in Information and Communication Technologies to John Hennessy of Stanford University and David Patterson of the University of California at Berkeley "for taking computer architecture, the discipline behind the central processor or 'brain' of every computer system, and launching it as a new scientific area."

The award citation described Hennessy and Patterson as "synonymous with the inception and formalization of computer architecture. Before their work, the design of computers—and in particular, the measurement of computer performance—was more of an art than a science, and practitioners lacked a set of repeatable principles to conceptualize and evaluate computer designs. Patterson and Hennessy provided, for the first time, a conceptual framework that gave the field a grounded approach toward measuring a computer's performance, energy efficiency, and complexity."

In addition to their contributions to computer architecture as researchers and educators, the two computer scientists have driven technological innovation and business development with their ideas. In the early 1980s, Hennessy and Patterson led the development of the Reduced Instruction Set Computer (RISC), the architecture on which the design of central processors is based. The Foundation noted that RISC is present today "at the heart of practically all datacenter servers, desktops, laptops, smartphones, and embedded computers (in televisions, cars, and Internet of Things devices)."

RISC is distinguished by its simple set of instructions, which were found to be able to be read as much as four times faster than its predecessor, complex instruction set computer (CISC), resulting in its popularity and ubiquity.

Hennessy offered a metaphor for the operation of RISC: "Think of an essay you are reading. Suppose the essay uses very complex words and convoluted sentence structures, it is difficult to read fast, you have to go very slowly, at a very slow pace. Now, imagine an essay that is written with clear words, simple words, a clean sentence structure, that can be read quickly. That's what RISC does."

Patterson said their work shares the determination to bring a systematic, reproducible method to their research, which enabled them to formalize the domain of computer architecture, which led them to RISC, and to the writing of their landmark textbook, *Computer Architecture: A Quantitative Approach*, which three decades later is still considered "the bible" for the discipline in universities around the world. Said Patterson, "We design processors the same way we design books, through experiment and trial."

The award citation noted that their work has had a deep and enduring impact. "They conceived the scientific field of computer architecture, motivated a systematic and quantitative design approach to system performance, created a style of reduced instruction set processors that has transformed how industry builds computer systems, and have made transformative advancements in computer reliability and in large-scale system coherence."

Hennessy and Patterson also shared the ACM A.M. Turing Award for 2017 "For pioneering a systematic, quantitative approach to the design and evaluation of computer architectures with enduring impact on the microprocessor industry."

authors deposit their code and data in a public repository. This "open science" model can improve accuracy by letting others check for reproducibility, but it is a hard sell for companies that view their data as a competitive advantage.

Computer science assistant professor Olga Russakovsky and her group at Princeton University have built a tool to help reveal biases in existing large-scale image datasets. For example, the tool can analyze the distribution of training pictures with various attributes, including "protected" attributes like gender that users may want to avoid using in models.

Designers can use this information to curate the data or otherwise compensate for biases. Although these issues are particularly important for assessing fairness, human choices have always affected performance, Russakovsky stressed. "There's very much a person component in any part of building an AI system."

Creating Balance

One approach to biased data is to include repeated copies of under-represented examples, but Russakovsky said this type of "oversampling" is not very effective. A better approach, she said, is "not just sampling from the same distribution as your training data comes from, but manipulating that distribution." One way to do this is by augmenting the training data with synthetic data to compensate for underrepresented attributes.

As an uncontroversial example, Russakovsky describes training systems to recognize people with sunglasses or hats in images in which the two features usually appear together. To help the system distinguish the features, designers can add synthetic images of faces with only sunglasses or only hats. Similarly, researchers can use a three-dimensional model to generate training images viewed from various angles.

Mitchell agreed that synthetic data can be "somewhat useful" to augment data, for example with "long-tailed" datasets that have few examples of extreme attributes. The technique is easily implemented in text processing by swapping in synonyms, she said, but "On the image side, synthetic data are not quite there yet."

"The whole premise of machine learning is to infer a model of the world from data. If you know your model of the world, why do you do machine learning to begin with?"

For assembling large datasets, however, Mitchell noted that "It doesn't make sense to have that be synthetic data because it's going to be too biased, too templatic, or not have sort of the real-world variation that you want to have." Similarly, Madry worries that "Using synthetic data as a cure for biases is a chicken-and-egg problem. The whole premise of machine learning is to infer a model of the world from data," he notes. "If you know your model of the world, why do you do machine learning to begin with?"

Synthetic data also plays a central role in one of the hottest areas of machine learning: generative adversarial networks (GANs). These systems pit neural networks against one another, one generating data and the other responding to it. "GANs escape, a little bit, the duality" associated with synthetic data, since the generating network eventually uses principles that were not built into it, Madry said. Indeed, Madry's MIT colleague Antonio Torralba has explored using GANs to improve both the fairness and interpretability of AI systems.

Machine Pedagogy

In spite of such efforts to curate it, "At the end of the day, your data is going to have biases," Russakovsky said, for which algorithms may need to compensate. "The tension there is that machine learning models are really good at learning from data. As soon as you start adding additional constraints, you're overriding what the model wants to do."

"This is an inherent problem with neural networks," Madry agreed. "If you give them a specific task: maximize my accuracy on this particular data set, they figure out the features that do that. The problem is we don't understand what features they use." Moreover, what works in one setting may fail in another, Madry said. "The kind of signals and features that ImageNet makes you develop"—for example, to distinguish friends on social networks—"will be of limited usefulness in the context of medical AI."

In the long run, developers need to consider how to present data to help systems organize information in the best way, said Patrick Shafto, a professor of mathematics and computer Science at Rutgers University-Newark. "You don't sample information randomly. You pick it to try to help them understand."

In his work, Shafto draws on notions of cooperation that are well-known in the study of language and education. For example, the "teacher" might first select data that establishes a general principle, with more subtle examples later on. Other established pedagogical techniques, such as posing questions, might also encourage better generalization by AI systems as they do for human students. "We don't want our learning to be capped at what the person teaching us has learned," he said. "In the ideal world, it would go beyond that."

Current machine learning, tuned to minimize errors on training data, is reminiscent of the much-maligned "teaching to the test," which is "not a good objective," Shafto said. "We need new objectives to conceptualize what machine learning can and should be, going forward."

Further Reading

ACM Conference on Fairness, Accountability, and Transparency (ACM FAccT), **https://facctconference.org/index.html**

Gradient Science, a blog from Aleksander Madry's lab, https://gradientscience.org/

Hutchinson, B., Smart, A., et al,
Towards Accountability for Machine Learning Datasets: Practices from Software Engineering and Infrastructure, FAccT '21, 560 (2021), https://dl.acm.org/doi/abs/10.1145/3442188.3445918

Wang, A., Narayanan, A., and Olga Russakovsky, O.,
REVISE: A Tool for Measuring and Mitigating Bias in Visual Datasets (2020), https://arxiv.org/abs/2004.07999

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

© 2021 ACM 0001-0782/21/12 $15.00

Technology | DOI:10.1145/3490317

Keith Kirkpatrick

The Road Ahead for Augmented Reality

A heads-up look at augmented reality head-up displays.

Automotive head-up displays (HUDs), systems that transparently project critical vehicle information into the driver's field of vision, were developed originally for military aviation use, with the origin of the name stemming from a pilot being able to view information with his or her head positioned "up" and looking forward, rather than positioned "down" to look at the cockpit gauges and instruments. The HUD projects and superimposes data in the pilot's natural field of view (FOV), providing the added benefit of eliminating the pilot's need to refocus when switching between the outside view and the instruments, which can impact reaction time, efficiency, and safety, particularly in combat situations.

In cars, the main concern is distracted driving, or the act of taking the driver's attention away from the road. According to the National Highway Transportation Safety Administration, distracted driving claimed 3,142 lives in 2019, the most recent year for which statistics have been published. Looking away from the road for even five seconds at a speed of 55 mph is the equivalent of driving the length of a football field with one's eyes closed.

As such, the desire to ensure that drivers keep their eyes focused on the road, instead of looking down at the gauges on the dashboard, was the impetus for the development of HUDs suitable for use in production automobiles. The first automotive HUD that was included as original equipment was found on the 1988 Oldsmobile Cutlass Supreme and Pontiac Grand Prix; both were monochromatic, and displayed only a digital readout of the speedometer.

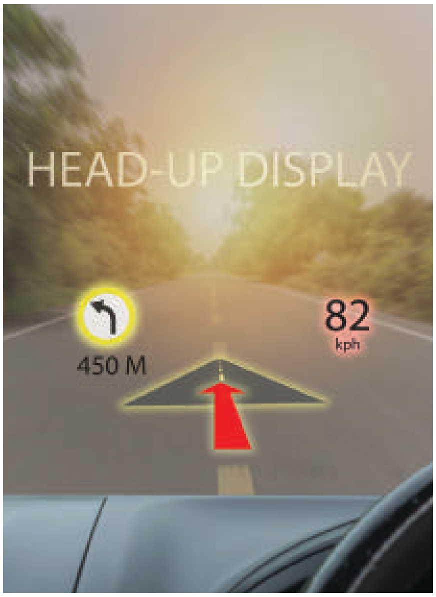


IMAGE BY MONOPOLY919

Thanks to the increasing inclusion of a variety of automotive sensors and cameras, advanced driver assistance system (ADAS) features and functions (such as automatic braking, forward collision avoidance, lane-keeping assist, and blind-spot monitoring, among others), and more powerful on-vehicle processors, automakers have been installing HUD units in commercial vehicles that provide more essential driving data, such as speed, engine RPMs, compass heading, directional signal indicators, fuel economy, and other basic information, allowing the driver to concentrate on the road instead of looking down to check the dash or an auxiliary screen.

The technology enabling most types of HUD is based on the use of a processor to generate a digital image of data coming from sensors. These images then are digitally projected from a unit located in the dash of the car onto a mirror or mirrors, which then reflect that image onto either a separate screen located behind the steering wheel, or onto the vehicle's windshield, directly in the driver's forward view. Common projection and display technologies used include liquid crystal display (LCD), liquid crystal on silicon (LCoS), digital micromirror devices (DMDs), and organic light-emitting diodes (OLEDs), which have replaced the cathode ray tube (CRT) systems used in the earliest HUDs, as they suffered from brightness degradation over time.

The HUDs that project the information onto a separate transparent screen are called combiner HUDs; these were popular because the physical space required to install the system was modest, and because the system was fully integrated, OEMs did not need to design a system that accounted for each vehicle's unique windshield angle or position. However, this type of HUD was limited by several factors; namely, the optical viewing path of a combiner HUD is shorter than looking through a windshield, and the driver's eyes must refocus slightly to the shorter visual distance when switching between looking out the windshield and checking the display. Furthermore, there is a practical limit to the size and field of vision (FOV) offered by combiner units; adding mirrors and a larger combiner screen would apper obtrusive and less elegant in a modern vehicle than simply using the windshield as a display surface.

Because HUDs were far from being standard automotive equipment in most vehicles, companies such as HUDWAY and Navdy had produced phone mounts and screens designed to allow a smartphone to operate as a head-up display. Essentially, these designs functioned as combiner systems, in that they required a separate screen on which to view the display and suffered from many of the same limitations as OEM-equipped combiner HUD systems. While Navdy went out of business in 2018, HUDWAY is still accepting orders for its HUDWAY Drive system at a cost of $229 per unit.

The technical limitations of combiner systems have driven most automotive OEMs to offer HUDs that project information directly onto the windshield and contain a far greater amount of data, known as W-type HUDs. These more-advanced systems incorporate ADAS system status information (such as displaying the status of adaptive cruise control systems, automatic braking systems, collision-avoidance technology, infrared night-vision technology, lane-keeping assist technology, and, eventually, semi-autonomous self-driving system data.

The most advanced systems include augmented reality technology, which involves superimposing specific enhanced symbols or images into the HUD onto real-world objects or roadways to provide more information, detail, and clarity to the driver. Some systems will also incorporate data from GPS navigation systems, such as clear directional graphics, street names, augmented lane markings, signposts and route numbers, and even representations of other vehicles/objects on the road. Examples of vehicles that include this technology today include the Audi Q4 E-tron, Mercedes Benz S Class, and the Hyundai IONIQ 5.

The major challenges faced by HUD designers include collimation, luminance, and clarity. Collimation refers to the aligning of light rays from the projector so they are parallel to one another, so the projected image appears to float ahead of the vehicle and seamlessly blends in with the outside world. This ensures the driver can watch the road and the display without refocusing. OEMs need to design the incorporation of each system into each vehicle so drivers of different heights can still have their eyelines in the correct position to view the HUD image properly.

There are a few different ways to accomplish this; adjusting the eyebox height by using a small motor to tilt one of the HUD mirrors up or down; applying more graphics processing combined with a driver monitoring system to compensate for the change in alignment, or designing a larger eyebox so the HUD's optical axis does not need to be adjusted and graphical alignment is maintained. Often a combination of the latter two approaches are used to provide an elegant system for providing optimal viewing angles for drivers of differing heights and seating positions.

Luminance, meanwhile, refers to the brightness of the display, which must be visible in all lighting conditions, from bright, sunny days, to overcast conditions and, of course, at night.

"The risk of cognitive overload due to screen clutter caused by displaying location-based advertising messages will need to be addressed."

Another challenge today faced by HUD designers is clarity: deciding what information should or should not be displayed on a HUD and how it should be displayed. Moreover, there is a concern about the potential for information overload. "The risk of cognitive overload due to screen clutter caused by displaying location-based advertising messages will need to be addressed," says Anuhab Grover, a mobility research analyst with market research firm Frost & Sullivan, referring to the likelihood that external partners will want to display to drivers additional messages or graphics that are not critical to driving. "It will be critical that AR is used in a minimalistic way and displays only relevant information needed to improve the driver's perception. For this, user experience and user interface design will be highly important."

Augmented reality technology can assist in this task by using graphical images that can convey information that can be easily understood by the driver, while also increasing the field of view and reducing the physical volume of the HUD system.

Some of the notable technologies that have been prototyped and are expected to make their way into production vehicles within the next five years include AR HUDs that utilize laser holographic and wave guide projection technologies. In laser holographic AR HUDs, a laser is used to beam AR content from the dashboard onto the windshield, which is embedded with holographic film that is embossed with images that

ACM Member News

BUILDING INTELLIGENT SUPPLY CHAINS

Chandrasekhar (Chandra) Narayanaswami, a Distinguished Research Staff Member at IBM's Thomas J. Watson Research Center in Yorktown Heights, NY, developed an interest in electronics as a student. "In college, I built my own computer from the ground up, and wrote my undergraduate thesis on it," he says.

Narayanaswami earned his undergraduate degree in electrical engineering from the Indian Institute of Technology Bombay, and his M.S. and Ph.D. degrees in computer and systems engineering from Rensselaer Polytechnic Institute in Troy, NY. On obtaining his doctorate, Narayanaswami joined IBM, where he has remained ever since.

However, while "there are some researchers who stick to one area for all of their career, that is not me," Narayanaswami says, explaining his career to date has been comprised of three distinct segments.

"I started off working on high-performance computer graphics, making things look good and run fast," he says. In time, Narayanaswami's interests gravitated toward mobile and wearable computing, where he became fascinated by how mobile and wearable devices could change the e-commerce experience.

His work in mobile commerce slowly pulled his interest into the back end of electronic commerce, which led to his current focus on the supply chain. Today, Narayanaswami leads supply chain research at IBM with a focus on combining artificial intelligence, blockchain, and the Internet of Things to build intelligent self-correcting supply chains.

"The pandemic has brought supply chains squarely into focus, as have the recent attacks on supply chains," says Narayanaswami. "Intelligent supply chains will be around for a while."

—John Delaney

appear with a three-dimensional effect, so that projected AR data appears to be overlaid onto real-world objects. Similarly, holographic displays that use wave guide technology, or an optical pathway through which light can travel, leverage the optical principles of diffraction, wavelength diversity, and reflection, in order to project the image onto a hologram incorporated into the windshield, instead of using a traditional DLP projector. The key benefits of these approaches include increased image sharpness and the creation of a larger eyebox that can allow users to adjust their seats or posture without losing the ability to view the AR HUD.

Grover says AR HUDs in development that utilize these types of advanced technology can provide sharper images across a wider field of view, while requiring far less physical space to install in a vehicle, a key consideration for OEMs. Today's AR HUDs typically take up about 15 to 20 liters in physical volume, or just a bit more than a typical shoe box, while newer designs using alternatives to DLP projectors and mirrors may reduce them to around one-sixth of that size.

For example, German multinational automotive parts manufacturer Continental Automotive and Sunnyvale, CA-based DigiLens, which makes holographic waveguides for augmented extended reality (XR) displays, have jointly developed a next-gen AR HUD prototype which uses a holographic waveguide projector to display visual information directly on the windshield.

Similarly, Swiss-based HUD maker WayRay, which has partnered with a wide range of automotive OEMs, uses holographic laser projector technology incorporating a holographic optical element (HOE) that is used to display the holograms, and which can be molded onto a flexible or curved surface, like a windshield. This system also uses a picture generating unit (PGU) installed on the car's dashboard comprised of a laser module, a digital light processing unit, and correction optics modules. The HOE is used to reflect the image projected by the PGU.

Meanwhile, Panasonic's AR HUD also incorporates laser holographic technology, which was developed by U.K.-based holographic technology startup Envisics, and integrates spatial AI technology developed by California's Phiar Technologies. In addition to displaying the road ahead, this system can identify and warn the driver about foreign objects, other vehicles, pedestrians, and cyclists, while also providing information such as the height of an upcoming overpass.

Tier-1 automotive parts supplier Denso is also working on an AR HUD, though the company declined to provide specifics, citing confidentiality agreements with OEMs.

Another current prototype comes from Scotland's Ceres Holographics, which is working on the development of a Holographic Transparent Display and AR Systems that utilize thin-film holographic optical elements (HOEs). The system uses a small projector embedded in the car's dashboard and a layer of holographic film laminated within the glass that makes up a windshield to project an in-plane hologram of the car's instrument cluster and navigation directly into the driver's line of sight.

According to Andy Travers, CEO of Ceres Holographics, the key challenge of incorporating the film into the windshield is rooted in the laws of physics. "The issue with holographic AR HUD, and in particular when the film is in the windshield, is with incoming light at particular directions," Travers says. "In certain circumstances, it can produce a rainbow artifact on the windshield, which is deemed distracting. So, while we and other companies try to address this issue, there is a debate in the automotive companies as to what level of artifact is acceptable or not acceptable."

The information presented on an AR HUD is the result of sensor fusion, which blends internal vehicle sensory information, including conventional HUD data with input from radar sensors and other vehicle sensor data, with digital map and GPS data to provide a virtual view of the world outside the vehicle, according to Grover. "In the future, advanced AR HUDs will be capable to project complex graphics fully integrating and adapting with the driver's environment," Grover says. "For instance, even on a foggy night using a car's thermal sensors, AR HUDs will be able to detect and display animal or human objects," improving safety in such low-light or otherwise adverse weather conditions.

According to market research firm Gartner's 2020 Priority Mix for Automotive Technologies, non-AR HUDs (those that simply project information such as speed, RPMs, or compass heading in front of the driver, but do not superimpose data onto real-world objects) are expected to become mainstream technology within two years, while AR technology is projected to become mainstream in consumer vehicles within five to 10 years. Major automotive OEMs and Tier-1 suppliers focused on developing and commercializing AR HUDs including Daimler, BMW, Jaguar, Hyundai, and others have forged partnerships, strategic alliances, or invested in technology companies, such as Porsche (WayRay), Volkswagen (SeeReal Technologies), General Motors and Hyundai Mobis (Envisics), Continental (DigiLens), and Ford (Mishor 3D).

AR HUD technology likely will help drivers stay more informed of their vehicle's operational systems, navigation directions, and infotainment systems (such as radio stations), while letting them stay focused on the road ahead. However, "prioritizing situational and contextual content will be crucial for AR HUDs," Grover says, noting that "an AI and deep learning application engine could play that role by prioritizing information [that] the driver needs to see." C

Further Reading

Ogbac, S.
What are Head-Up Displays?
And Are They Worth It?, *Motor Trend*, July 14, 2020, https://www.motortrend.com/news/head-up-display/

Augmented Reality Head-Up-Display on Audi Q4, *AirCar*, March 9, 2021, https://www.youtube.com/watch?v=4HW0WsqDP-E
Hudway: https://hudway.co/glass

Holographic Film – What is it, and where is it used?
https://nobelusuniversity.com/2017/05/05/holographic-film-what-is-it-and-where-is-it-used/

Top Head-Up Display Companies
https://www.ventureradar.com/keyword/Head-Up%20Display

Keith Kirkpatrick is principal of 4K Research & Consulting, LLC, based in New York, NY, USA.

© 2021 ACM 0001-0782/21/12 $15.00

Society | DOI:10.1145/3490165

Samuel Greengard

What Is the Cost of Living Online?

The cost of powering streaming and other rapidly growing online services will not "take down the Internet."

MODERN LIFE INCREASINGLY is defined by the activities we engage in online: Zoom meetings at work, Netflix and Xbox marathons at home, and a steady stream of YouTube, TikTok, and Facebook video clips in the nooks and crannies in between.

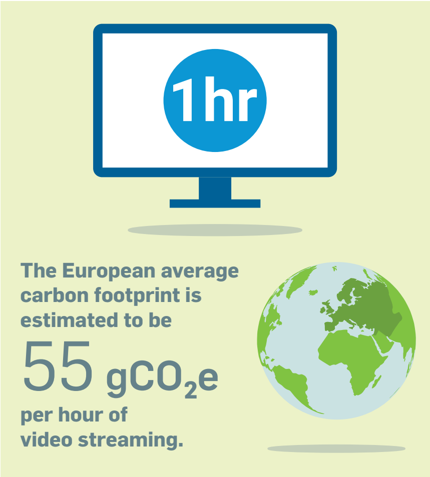
There are many benefits to life online, yet there are also undeniable social, economic, and environmental costs. While global emissions from video streaming and other digital activities comprise somewhere in the neighborhood of 3% of the total,[a] the voracious and growing appetite for bandwidth is raising concerns about sustainability—and prompting some to wonder whether it is possible to keep up with the demand.

"We're seeing the digitization of everything—work, entertainment and shopping. There's a huge shift in lifestyle and it's sharpening the focus on how all of these devices impact things," says Eric Williams, a professor of sustainability at the Golisano Institute for Sustainability of the Rochester Institute of Technology.

As bandwidth demand ticks upward and carries the demand for power with it, "There's an emerging discussion about the role of all the digital services we've come to rely on," says Mike Hazas, a professor in the Department of Information Technology at the University of Uppsala in Sweden. "It's an important discussion, because how we design and use systems will define our future."

Left to Our Devices

There's a common assumption that life online is cleaner and greener than life in the physical realm. There is near-zero cost to sending an email message or viewing a YouTube video. While it is true a Zoom meeting consumes only a fraction of the energy of a commute to work or a flight across the country, it does require bandwidth and electricity. Of course, as millions of people venture online for billions of video calls, the energy and bandwidth requirements accumulate, and can spike.



The European average carbon footprint is estimated to be 55 gCO_2e per hour of video streaming.

ALL INFOGRAPHICS FROM WHITE PAPER "CARBON IMPACT OF VIDEO STREAMING," JUNE 2021, COURTESY OF THE CARBON TRUST

The ability to click and instantly watch videos—and even autoplay them in various apps—has changed behavior in profound ways. According to networking firm Sandvine, upwards of 60% of the traffic on the Internet is now related to consumer video streaming, and sites such as Netflix, Facebook, Instagram, TikTok, and YouTube carry the bulk of this traffic, which is growing at an annual clip of about 24%.[b] The Carbon Trust, an independent U.K.-based advisory organization comprised of experts in sustainability, reports that long-form video streaming accounts for 45% of all Internet traffic.[c]

Artificial intelligence, machine learning, deep learning, cryptocurrency mining, Blockchain, and the Internet of Things (IoT) are poised to ratchet up the stakes further. "These systems will add huge volumes of traffic to the Internet, and much of this traffic is automated and not constrained by users," says Kelly Widdicks, a post-doctoral researcher at the School of Computing and Communications at Lancaster University in the U.K.

The direct use of devices, and how they draw power and bandwidth, is not the only factor in understanding how they impact things, however. About 90% of the energy a smartphone uses during its life cycle is embedded in the manufacturing process.[d] This includes collecting rare materials for batteries, fabricating devices, and recycling and disposing of components. What's more, after a smartphone handset is produced, about 90% of the energy consumption takes place off the phone, including on the network and in the datacenter.[e]

Further complicating matters: fast, persistent Internet connections modify behavior. A 2021 study conducted by a pair of researchers at the U.K.'s University of Sussex, Bernado Calderola and Steve Sorrell, found that the availability of telework may actually encourage people to move farther from their place of work and engage in additional non-work-related travel. The authors noted that such "results provide little support for the claim that teleworking reduces travel." Widdicks says there also is evidence that as people expand their social footprints online, they sometimes travel more to meet others.

Indeed, the relationship between infrastructure and streaming is complex, Hazas says. Growth in infrastructure initially made it possible to offer streaming services, which over

[a] https://arxiv.org/abs/2102.02622

[b] https://bit.ly/3izIeCV

[c] https://bit.ly/3oIGg6S

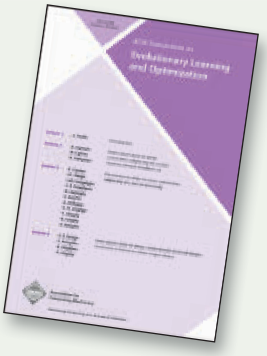
[d] https://www.en-former.com/en/digital-energy-consumption-on-the-rise/

[e] *The Impact of Teleworking on English Travel.* Bernardo Calderola and Steve Sorrell. University of Sussex. ECEEE. Publication forthcoming.

Association for Computing Machinery

ACM Transactions on Evolutionary Learning and Optimization (TELO)

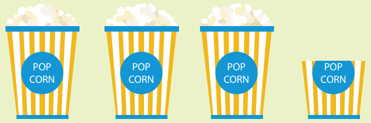
ACM Transactions on Evolutionary Learning and Optimization (TELO) publishes high-quality, original papers in all areas of evolutionary computation and related areas such as population-based methods, Bayesian optimization, or swarm intelligence. We welcome papers that make solid contributions to theory, method and applications. Relevant domains include continuous, combinatorial or multi-objective optimization.



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

time came to be expected by consumers and streaming providers. This, in turn, has contributed to further growth in online infrastructure.

For now, there's no end in sight. The total footprint from digital technology (including manufacturing gadgets) is now on par with the airline industry, at approximately 2% to 3% of global carbon emissions.[f] Industry estimates predict that streaming video as a percentage of Internet traffic could reach into the 80% range within the next few years.



The carbon emissions of an hour of video streaming are roughly 3.5× that of microwaving a bag of popcorn.

A Numbers Game

Understanding the real-world impact of streaming video, online gaming, and other digital activities is challenging. For one thing, devices continue to become more energy-efficient and smarter about how they use bandwidth. For another, "The equation varies greatly depending on the source of energy and the type of device someone uses," Williams says. Wind and solar energy create a very different cost structure than petroleum-based fuels. A 50-inch smart TV pulls 4.5 times more power than watching a video on a laptop, and 90 times more than watching it on a smartphone, according to the Carbon Trust.

For years, researchers and media outlets have inflated the carbon footprint of digital services. For example, Yahoo!, BigThink, *Gizmodo*, *The New York Post*, Phys.org, and others have reported that the emissions generated by watching 30 minutes of Netflix is the same as driving almost 4 miles (or 3.2kg CO_2e (carbon dioxide equivalents) per hour). The figure originated from the Shift Project,[g] a French think tank, which later corrected this calculation downward by a factor of eight to 400g CO_2e per hour.[h] A recent peer-reviewed study by researchers from Purdue and Yale universities and the Massachusetts Institute of Technology estimated that an hour of streaming Netflix emits 440g CO_2e.[i] Yet another study published by researchers from Simon Fraser University claimed that 35 hours of watching HD video emitted 2.68 metric tons of CO_2e, equivalent to 76kg CO_2e per hour.[j]

It turns out that all these calculations missed the mark. George Kamiya, a digital energy analyst for the International Energy Agency in Paris, exposed these flawed calculations in a blog post published in February 2020[k] and updated in December.[l] His analysis showed that, on average, one hour of streaming video consumes power that results in the emission of 30 to 80g CO_2e. "These earlier headline numbers [3.2kg CO_2e per hour or 6.1 kW] didn't make sense to me," Kamiya says. "If true, this would be the equivalent to having over 60 big-screen TVs on at the same time."

A June 2021 white paper released by the Carbon Trust (partially funded by Netflix) confirmed that an hour of streaming video in Europe resulted in the emission of 55g CO_2e, the carbon equivalent of boiling three kettles of water. In addition, using its own data, the BBC estimated that an hour on its streaming service, iPlayer, results in the emission of 33g CO_2e per hour in the U.K.[m]

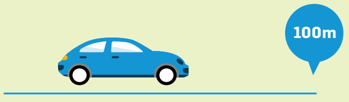
The story is also rosier at the datacenter level. In 2020, a group of researchers led by University of California, Santa Barbara professor Eric Masanet found that cloud-based datacenters account for only 1% of worldwide electricity use. Claims that giant datacenters would lead to catastrophic consequences have proved wrong. According to Google, the move to the cloud has resulted in up to seven times more computing output per datacenter than only a few years ago.[n]

Bandwidth concerns are another flash point. At the outset of the COVID-19 pandemic, Spanish telecom Telefonica, for instance, reported a

[f] https://www.atag.org/facts-figures.html
[g] https://bit.ly/3DEEIPJ
[h] https://bit.ly/3BLDm52
[i] https://bit.ly/3DGfn7V
[j] https://bit.ly/3DGfn7V
[k] https://bit.ly/3uJsHoN
[l] https://bit.ly/3vdbMvc
[m] https://bbc.in/3aFdbkB
[n] https://nyti.ms/3vecLuU

45% increase in data traffic. In fact, bandwidth demand shot up so fast that a top European Union official spoke with Netflix CEO Reed Hastings about switching to standard definition video if networks became too congested.[o] Havoc never ensued, yet fears persist that growing use of video streaming, online gaming, cryptocurrency mining, and artificial intelligence could eventually reach a tipping point and take down the Internet.

As 5G enters the picture, additional questions arise. While research firm IDC has reported that 5G delivers a 3x boost in spectrum efficiency and a 100x improvement in traffic capacity and network efficiency,[p] it's unclear what the overall impact will be, says Chris Bronk, an assistant professor in the Department of Information Technology at the University of Houston. "As you continue to add devices and introduce new technologies into the mix, the equation becomes a lot harder to figure out," he says. "There are estimates that 5G will pull 10 times as much power as LTE and increase bandwidth demands further but, in reality, nobody has a definitive answer."



The carbon emissions of an hour of video streaming are approximately 2.5× those of the average emissions of driving a distance of 100m.

A Clearer Picture

Although reports of digital devices leading to the demise of civilization have been largely exaggerated, there are growing questions about what can be done to better manage both energy and bandwidth demands. Bronk points out that many ISPs and mobile providers already cap bandwidth. Customers who exceed limits pay steep fees, or wind up throttled down in speed, or even blocked from further consumption. Others have floated the idea of a consumption tax and tighter regulations about energy sourcing.

> "Peaks are very rarely attained, even with the highest possible bandwidth services. There is absolutely no way to 'take down the Internet' with streaming."

Hazas says that one way to address growing video demand and bandwidth is to turn off the autoplay feature that streams one video after another on YouTube, Facebook, Instagram, and other platforms. Some organizations, such as the BBC, now stream video in standard definition by default, with high definition as an option for those viewing content on larger screens. Likewise, a recent study conducted at Purdue University also found that by turning off the camera during Zoom and Microsoft Teams calls, it's possible to reduce one's carbon footprint by 96%,[q] while streaming Netflix or Hulu in standard definition results in an 86% reduction.

Some, such as Dom Robinson, co-founder of consulting firm Id3as and an expert in content delivery networks (CDNs) and scaling streaming video, say that concerns over energy consumption are well-placed, but bandwidth fears are entirely misplaced. "A common myth propagated by vendors of streaming services is that there's a 'scarcity of resource.' The purpose of this is to decelerate a rapid commoditization of pricing for these services," he argues. ISPs and telecoms have enormous overhead in their networks, he says; "They are able to sustain many times peak network traffic. Peaks are very rarely attained, even with the highest possible bandwidth services. There is absolutely no way at all to 'take down the Internet' with streaming," he says.

In the end, Hazas believes it is vital to maintain a sense of perspective. Although information and communications technology will devour a greater share of electricity and bandwidth in the future, the overall carbon footprint for the sector has remained constant—and it could drop if efficiency gains continue.[r]

"The migration online is a net positive. If it comes down to Netflix or a drive to the local cinema, there's no doubt that Netflix is the better choice from an environmental perspective. … We need to pay attention to bandwidth and energy consumption and look for ways to improve efficiency and behavior. But there are much bigger problems, such as how we heat and cool our homes, use vehicles, and the foods we choose to eat."

[o] https://bit.ly/2YYeWqR
[p] https://bit.ly/3uJnTzI
[q] https://bit.ly/3aaeqby
[r] https://bit.ly/3oBpczK

Further Reading

Morley, J., Widdicks, K., Hazas, M.
(2018). Digitalisation, energy and data demand: The impact of Internet traffic on overall and peak electricity consumption. *Energy Research & Social Science,* Elsevier. 38: 128-137
https://doi.org/10.1016/j.erss.2018.01.018

Widdicks, K., Hazas, M., Bates, O., Friday, A.
(2019). Streaming, Multi-Screens and YouTube: The New (Unsustainable) Ways of Watching in the Home. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, https://doi.org/10.1145/3290605.3300696

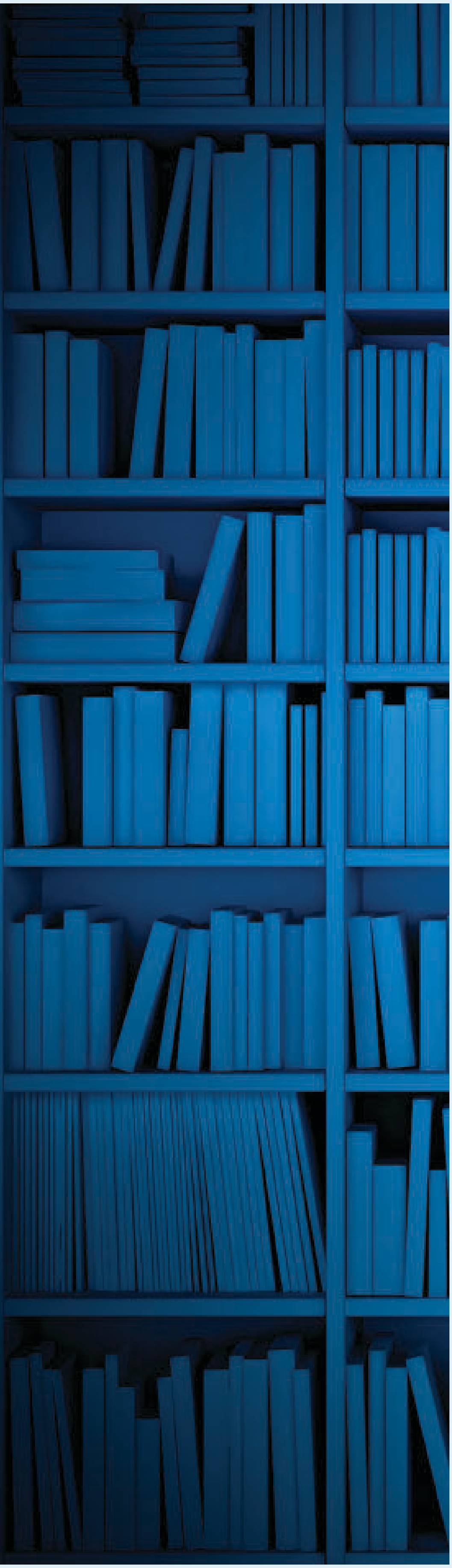
Kamiya, G.,
The carbon footprint of streaming video: fact-checking the headlines, IEA, December 11, 2020
https://www.iea.org/commentaries/the-carbon-footprint-of-streaming-video-fact-checking-the-headlines

Pelau, C., Acatrine, C.,
The Paradox of Energy Consumption Decrease in the Transition Period towards a Digital Society. *Energies*, 2019, 12, 1428; doi:10.3390, en12081428
www.mdpi.com

Carbon Trust
Carbon Impact of Video Streaming, June 2021. https://prod-drupal-files.storage.googleapis.com/documents/resource/public/Carbon-impact-of-video-streaming.pdf

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

© 2021 ACM 0001-0782/21/12 $15.00



DOI:10.1145/3488236

BY JACK W. DAVIDSON / UNIVERSITY OF VIRGINIA,
JOSEPH A. KONSTAN / UNIVERSITY OF MINNESOTA,
AND SCOTT E. DELMAN / ASSOCIATION FOR COMPUTING MACHINERY

ACM Publications Finances for 2020

WE PROVIDE HERE an annual update on ACM Publications finances, following up on our initial report published in the May 2020 issue of *Communications of the ACM* (p. 53).[a] This report summarizes income and expenses for the 2020 calendar year. Specific notes explain significant changes between 2019 and 2020.

There are several key themes that cut across the various categories:

‣ **The COVID-19 pandemic and its consequences.** 2020 is an outlier year (and 2021 will continue to be an outlier) due to the effects of the global pandemic. A significant number of ACM conferences were cancelled. ACM Headquarters shifted to remote operation and instituted a hiring freeze. And travel (for sales, development of new content, and governance) was nearly eliminated during the pandemic period. Most of these deviations are temporary, though we are still working through changes in travel to understand how much in-person events will be replaced by videoconferencing in the longer term.

‣ **Digital Library development.** 2020 shows ongoing investment into our new digital library platform. It is the first full year of operation for our new platform (with all the associated vendor expenses) and is a year with substantial ongoing development expenses. Within the scope of "digital library" we include the full range of production systems, including substantial investments (and ongoing costs) for accessibility and mobile device compatibility of published articles. These changes will continue into 2021; ACM chartered

[a] https://dl.acm.org/doi/10.1145/3389687

a new Digital Library Board that will prioritize and oversee continued improvements in the author and reader experience.

▸ **ACM Open.** In 2020, ACM committed to a path toward open access based on institutional publish-and-read agreements (referred to as "transformative agreements"). The ACM Open model offers institutions a subscription fee based on the historical number of articles published by its authors, and in return makes those articles open. Combined with author-choice OA, ACM published 8.9% of its articles open access in 2020.

▸ **Transition to ACM Open is a five-year effort (and may take longer due to the COVID-19 pandemic).** We will provide an update on the initiative in a future issue of *Communications*.

▸ **Improved Financial Reporting.** As we noted last year, part of the challenge of reporting on publications finances came from the fact that these figures combine hundreds of sources and thousands of transactions, not all of which are completely attributable to publications. As we complete this second-year report, we have been able to improve the attribution of expenses and income, and we expect to continue to do so for at least one more year.

ACM Publications Financials For Calendar Year 2020

Income	
Digital Library: Consortia, Corporates, & Govt Licenses	20,233,851
Digital Library: ACM Open Licenses	516,000
Digital Library: Articles Pay Per View	56,800
Institutional Membership Dues	318,504
Subscription Income (including SIGs); á la Carte Subscriptions	810,994
SIG Print Magazine Subscriptions (*Interactions*/*Inroads*)	153,841
Digital SIG Master Package	148,845
Advertising, including SIGs	934,732
ICPS Proceedings: Non-ACM Conference Publication Fees	349,675
Open Access Income (APCs)	88,400
All Other Publications Income: ACM Books, etc.	473,433
Total Income	**24,085,075**

2020 High-Level ACM Publications Financials

Income	**2020**	**2019**	**Favorable/ (Unfavorable)**
Subscriptions & Advertising	23,569,075	23,992,725	(423,650)
Digital Library: Open Access Licenses	516,000	N/A	516,000
Total Income	**24,085,075**	**23,992,725**	**92,350**
Expenses			
Journals	3,896,143	4,091,846	195,703
Magazines	5,313,187	5,519,977	206,790
Proceedings	5,468,671	5,631,759	163,088
Digital Library	5,888,960	5,098,667	(790,293)
Agents/Sales	2,848,053	2,747,357	(100,696)
Publications Board	60,025	211,615	151,590
Publishing Program, net	**610,037**	**691,505**	**(81,468)**

Notes:

1. **Total income.**
   Total publications income remained essentially flat from 2019, even though overall Digital Library income increased by $480K due to the introduction of ACM Open in January 2020. However, this increase was offset by a $300K decline in advertising income, which is primarily income generated by job postings on jobs.acm.org. 2020 ACM Open income comes from approximately 35 institutions in Germany, Ireland, the Netherlands, Saudi Arabia, the U.K., and the U.S.
2. **Journals Expenses.**
   An overall decrease of $195K in journal-related expenses, which is attributable to (1) decreased print and distribution related expenses because of the pandemic; (2) increased use of ACM's new XML-based production system; and (3) reduced staff costs due to ACM HQ hiring freeze and unfilled positions.
3. **Magazines Expenses.**
   An overall decrease of $206K, which is attributable to (1) decreased print and distribution related expenses as a result of the pandemic; (2) reduced staffing costs due to ACM HQ hiring freeze and unfilled positions; and (3) lower staff and volunteer-related travel costs due to the pandemic.
4. **Proceedings Expenses.**
   An overall decrease of $163K, which is attributable to reduced costs because of a reduced volume of papers (approximately 1,600 fewer than 2019) due to conference cancellations caused by the pandemic. It is expected that volume and related expenses will increase in future years.
5. **Digital Library Expenses.**
   A significant increase of $790K in overall Digital Library related expenses, which is attributable to (1) increased expenses related to the implementation of ACM's XML-based production system (TAPS); (2) increased expenses related to Digital Library platform development; and (3) increased expenses related to data-quality improvements, such as improved meta-data and publications-related data needed for ACM Open reporting. It is expected that Digital Library expenses will decline over time.
6. **Publications Board Expenses.**
   A decrease of $151K in travel-related expenses, which is attributable to the elimination of face-to-face meetings. These expenses are expected to remain lower for 2021, and then to gradually increase when it is appropriate to transition to a hybrid in-person and virtual meeting schedule.

ACM ON A MISSION TO SOLVE TOMORROW.

Dear Colleague,

Without computing professionals like you, the world might not know the modern operating system, digital cryptography, or smartphone technology to name an obvious few.

For over 70 years, ACM has helped computing professionals be their most creative, connect to peers, and see what's next, and inspired them to advance the profession and make a positive impact.

We believe in constantly redefining what computing can and should do.

ACM offers the resources, access and tools to invent the future. No one has a larger global network of professional peers. No one has more exclusive content. No one presents more forward-looking events. Or confers more prestigious awards. Or provides a more comprehensive learning center.

Here are just some of the ways ACM Membership will support your professional growth and keep you informed of emerging trends and technologies:

- Subscription to ACM's flagship publication ***Communications of the ACM***
- Online books, courses, and videos through the **ACM Learning Center**
- Discounts on registration fees to ACM Special Interest Group conferences
- Subscription savings on specialty magazines and research journals
- The opportunity to subscribe to the **ACM Digital Library**, the world's largest and most respected computing resource

Joining ACM means you dare to be the best computing professional you can be. It means you believe in advancing the computing profession as a force for good. And it means joining your peers in your commitment to solving tomorrow's challenges.

Sincerely,

Gabriele Kotsis
President
Association for Computing Machinery

acm Association for Computing Machinery
Advancing Computing as a Science & Profession

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 Association for Computing Machinery

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
acm.org/join/CAPP

DOI:10.1145/3491268

Peter J. Denning and Matti Tedre

The Profession of IT

Computational Thinking for Professionals

Professionals practice a form of computational thinking that is significantly more advanced than popular descriptions suggest.

Computational thinking, a K–12 education movement begun in 2006, has defined a curriculum to teach basic computing in pre-college schools. It has been dramatically more successful than prior computer literacy or fluency movements at convincing K–12 school teachers and boards to adopt a computer curriculum. Learning problem-solving with algorithms is seen widely as valuable for students. Hundreds of CT initiatives have blossomed around the world.

By 2010, the movement settled on a definition of CT that can be paraphrased as "Designing computations that get computers to do jobs for us." The recommended K–12 curricula were narrow in scope, designed to teach newcomers the basics of algorithms, programming, and using computers. One oft-cited overview lists nine fundamental concepts as the core of CT[2]:

- Abstraction;
- Data collection;
- Data analysis;
- Data representation;
- Algorithms and procedures;
- Problem decomposition;
- Automation;
- Parallelization; and
- Simulation.

Abstraction has been held up as the first and foremost principle. In their 2021 Turing Lecture, Alfred Aho and Jeffrey Ullman emphasized the importance of abstraction in the design of programming languages and compilers.[1]

Shortcomings of CT for Beginners

The CT story outlined above is excellent for getting young students started with programming and the powerful principles of algorithmic thinking. Unfortunately, the success of this story has also become a liability because it projects a view of computer science from which many ideas familiar to computing professionals are missing.

We call the basic story for K–12 "CT for beginners." We introduced the advanced story, "CT for professionals," for all the thinking and design practices in daily use by professional practitioners.[4] It is time to stop conflating the whole story of computational thinking into one for beginners and to expand to the whole spectrum from beginner to professional.

There are several reasons that this expansion would be healthy. First, one of the original goals of the movement was to describe how computer scientists think. Basic CT is relatively easy to learn but it reveals little of the advanced thinking of computing professionals. Most of the basic practices listed earlier are not unique to computing—they have appeared in mathematics, science,

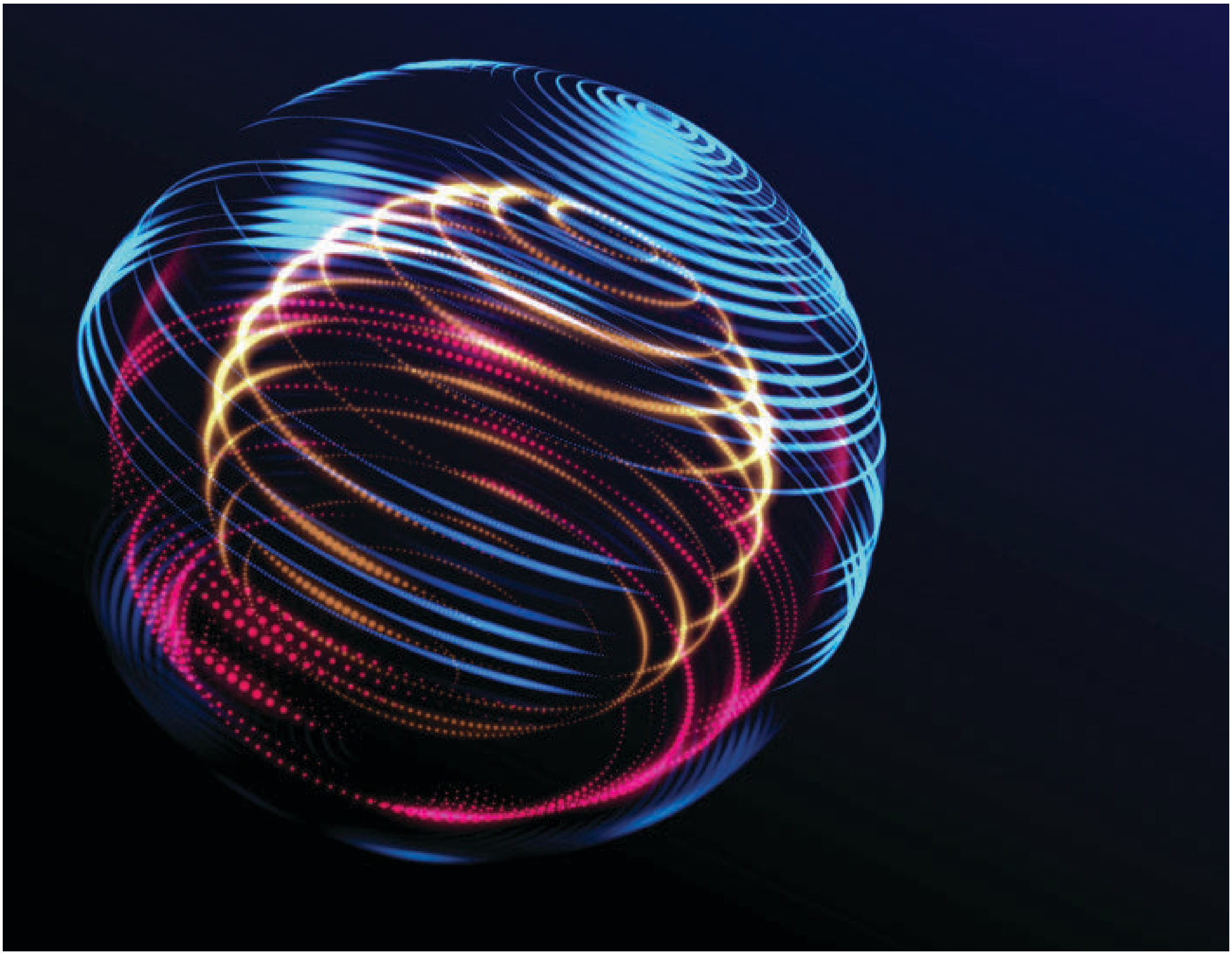


IMAGE BY THINKHUB STUDIO

and engineering for centuries.[5] Many teaching practices recommended for K–12 make heavy use of generic logic puzzles and games teasers, which intrigue children but do not illustrate the unique ways of thought and practice that make computing attractive to other disciplines. There is no good reason to use generic logic puzzles, when there are numerous examples of puzzles and games that illustrate unique features of computing, as has been done in the CS Unplugged project.[3] The current CT story is not meeting one of the original goals.

Second, Basic CT concentrates heavily on how to use the listed concepts to write good programs ("coding"). This is good: programming is central to CT. But such a heavy focus on coding reinforces a common public misconception about computing, often called "CS = programming." We fought long and hard in the 1990s to eliminate this perception. It is disconcerting to see it reviving as an unwitting consequence of the narrow view of computing built into Basic CT.

Third, Basic CT is mostly silent about the thinking and design practices for the real world of large programs, systems, networks, and user communities. It does not describe how computing professionals meet concerns, answer threats, approach problems, rise to opportunities, or communicate practices. It paints for the public—as well as school teachers, parents, and policymakers—a narrow, technical, non-humanist image of computing as a profession.

Basic CT is relatively easy to learn but it reveals little of the advanced thinking of computing professionals.

Fourth, much modern computing relies on simulation and modeling. Simulation builds information models of real-world processes, uses the computer to study their behavior, and then makes inferences about the real processes. Simulation has become a very powerful tool for professionals in computational sciences. As a result of this new emphasis, the working paraphrase definition of CT has expanded: "CT is the mental skills and practices for (1) designing computations that get computers to do jobs for us, and (2) explaining and interpreting the world as a complex of information processes."

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.

INTERACTIONS
Tech Labor
How Does AI Challenge Design Practice?
Interrupting Invisibility in a Global World
The Collateral Damage of Digitalization During Covid
A Proposal for Farmer-Centered AI Research

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

Most specifications for K–12 curricula focus on the first aspect of CT and have little to say about the second. In the remainder of this column, we will give some examples of CT for professionals and comment on why these are not implied by the beginner concepts. These advanced practices have emerged over the years as professionals grapple with the work of designing and building reliable software, evaluating the performance of systems, building distributed networks and operating systems, and designing user interfaces.

What's in CT for Professionals

Here are examples of advanced CT that professionals employ.

Neural networks are the engines of many popular AI services and tools today. Some Basic CT curricula mention that neural networks power interactive bots like Siri and Alexa, tag friends in Facebook photos, run face ID on a phone, or drive TikTok's uncanny ability to show videos each subscriber will like. Some also mention that neural networks are trained rather than programmed. But Basic CT does not mention the many challenges professionals face when trying to make neural networks reliable. Professionals design the training regimens, gather and curate the training data, test the reliability of the trained network, work to and integrate the network with other software and user interface. They understand the tensor chips that power neural networks, what kinds of errors their internal algorithms might generate, how the training algorithms might be improved, and how sensitive the network is to small changes of input. They are also involved in research on how to eliminate biases the network inherits from the data, how to explain a network's output that does not make sense, or how to evaluate how safe a network is for use in critical systems such as a driverless car. Fortunately, young learners who want to try many of these ideas at a toy scale can do with Google's Teachable Machine, without having to wait for them to be included in basic CT curricula.

Computational complexity is a unique aspect of computing. It characterizes the time and storage resources to solve a wide variety of problems on computers. Basic CT may mention that some algorithms are easier than others—for example, finding an item in a sorted pile is much easier than sorting the pile. It may discuss that some problems such as designing an optimal transportation system can be too hard for any known computer. But basic CT does not mention what the NP-complete issue is all about, what kinds of heuristics have been devised for finding approximate solutions to the hard problems, or what kinds of machines and algorithms are required to break existing public key encryption. The workings of quantum computers that might solve some of these problems in reasonable time are too advanced for Basic CT.

Software engineering takes a professional's perspective to software development. Basic CT is primarily about what might be called "CT-in-the-small"—programming methodology for small programs used by a single type of user. CT-in-the-small is suitable for beginner programmers to learn how to design and write clean code for their own programs. But designers of large software systems rely on design and project practices to manage the development of systems of many modules produced by many programmers published in many versions. Their concerns include interoperability, cross-platform portability, cross-version compatibility, bottlenecks, and understanding how

Good designers combine human insight, social pragmatics, organizational understanding, and machine control.

to design for thousands, maybe millions of users. This "CT-in-the-large" cannot be taught in Basic CT.

Operating Systems and Networks. These complex systems are the fundamental infrastructure for computing. They provide the platforms and connectivity for services such as the Cloud, Commerce, and social networking. They are also the targets of criminals who would steal data and compromise critical systems. Basic CT mentions the existence of these technologies but has little to say about their advanced concepts. Basic CT might teach that, according to abstraction and decomposition, these large systems are assemblies of modules and interfaces—where in reality they are continuously operating "societies" of cooperating dynamic processes or autonomous agents. These societies include many large subsystems, each a complex set of abstractions, and rely on their kernels for correct coordination and reliable operation. They support the Web and Cloud.

Distributed Computing and Performance Modeling. Most user jobs require service from multiple servers in a computer network. Computer systems and networks are useless if they cannot provide responses in reasonable times and keep up with a large workload of user jobs. Basic CT barely touches the structure of jobs that require multiple services and says nothing of the effects of competition and congestion at bottleneck servers. Professional performance analysts of distributed systems rely on queueing network models to answer throughput and response-time questions and to configure systems with sufficient capacity to give good service to users.

Interaction design is one of the key skills in advanced CT for professionals. Computers have no intuition or capacity to care about users. The illusion of smart systems arises from the expertise of designers at understanding how to craft software whose behavior appears "intelligent." Smart technology is a triumph of design. Take the iPhone, for instance: its success arises not only from its considerable technical prowess, but also from its support of user identities and fashion statements. Good designers combine human insight, social pragmatics, organizational understanding, and machine control. Great designers understand the social practices of communities, recognize value and worth in different social realities, listen for concerns, recognize and orchestrate emotions and moods, and identify ethical and moral problems in constructing large systems.

Despite more than a half-century of research on how to teach computing in the school, teaching computing concepts to children remains a great challenge.

Conclusion

Despite more than a half-century of research on how to teach computing in the school, teaching computing concepts to children remains a great challenge. It will keep computing education researchers busy for decades to come. We advocate that CT curriculum developers focus their attention to two things.

First, we advocate that teachers use computing's hard-earned hours in the K–12 curriculum to teach practices unique to our discipline, instead of rehashing generic brain puzzles, mathematics exercises, or perceptual reasoning problems. We are concerned that in the excited rush to develop CT curricula for schools, too many generic ideas may have been introduced at the cost of computing's own disciplinary concepts, ideas, skills, and practices. This ought to be changed.

Second, we advocate that the public face of CT be expanded to cover the rich spectrum of CT insights from beginner to professional. One of computing's perennial challenges has been the public perception of the field as little more than coding. This image of computing is harmful because it does not show the public the vast range of activities people in computing do. We curate and clean data, train neural networks, and use them to make everyday things smart. We find ways to avoid network bottlenecks to get the full power of the world's biggest computing clusters to the fingertips of smartphone users, without them ever noticing any delay. We continuously seek clever heuristic ways to circumvent the limits of computing. We build software that creates virtual worlds that seamlessly fit social communities and their practices.

Since the birth of the discipline, computing has been plagued by the public perception "computer science = programming" combined with a stereotype that only nerdy social misfits can do the programming. Both these perceptions are deeply mistaken. It is time to stop trying to explain computing with Basic CT. We need to retool the public face of CT to celebrate the professional richness of the field. C

References

1. Aho, A. and Ullman, J. Turing Lecture. 2021.
2. Barr, V. and Stephenson, C. Bringing computational thinking to K-12: What is involved and what is the role of the computer science education community? *ACM Inroads 2*, 1 (2011), 48–54.
3. Bell, T. et al. Computer science unplugged: School students doing real computing without computers. *The New Zealand Journal of Applied Computing and Information Technology 13*, 1 (2009), 20–29.
4. Denning, P.J. and Tedre, M. *Computational Thinking.* The MIT Press, Cambridge, MA, USA, 2019.
5. Denning, P.J. and Tedre, M. Computational Thinking: A disciplinary perspective. *Informatics in Education 20*, 3 (2021); https://doi.org/10.15388/infedu.2021.21
6. Grover, S. and Pea, R.D. Computational thinking in K–12: A review of the state of the field. *Educational Researcher 42*, 1 (2013), 38–43.

Peter J. Denning (pjd@nps.edu) is Distinguished Professor of Computer Science and Director of the Cebrowski Institute for information innovation at the Naval Postgraduate School in Monterey, CA, is Editor of ACM Ubiquity, and is a past president of ACM. The author's views expressed in this column are not necessarily those of his employer or the U.S. federal government.

Matti Tedre (matti.tedre@uef.fi) is Professor of Computer Science at the University of Eastern Finland.

Copyright held by authors.

DOI:10.1145/3491270 Marshall W. Van Alstyne, Georgios Petropoulos, Geoffrey Parker, and Bertin Martens

Economic and Business Dimensions
'In Situ' Data Rights

Improving on data portability.

IF WE ARE to hold platforms accountable for our digital welfare, what data rights should individuals and firms exercise? Platforms' central power stems from their use of our data so what would we want to know about what they know about us? Perhaps a reallocation of rights will rebalance the right allocation. To date, the General Data Privacy Regulation (GDPR in the E.U.) and California Consumer Protection Act (CCPA in the U.S.) grant privacy rights to individuals, including the right to know what others know about them and to control data gathering, deletion, and third-p arty use. Legislation also includes data portability rights, an individual right to download copies from and upload copies to destinations of one's choosing as protections for individuals. Neither yet covers businesses. The proposed Digital Markets Act (DMA) takes a step in that direction. The theory is that privacy empowers individuals to control what is gathered and who sees it; portability permits analysis and creates competition. By moving our data to portals that would share more value in return, we might capture more of our data value. After all, that data concerns *us*.

IMAGE BY OMELCHENKO

Data portability sounds good in theory—number portability improved telephony[9]—but this theory has its flaws.

▸ *Context:* The value of data depends on context. Removing data from that context removes value. A portability exercise by experts at the ProgrammableWeb succeeded in downloading basic Facebook data but failed on a re-upload.[1] Individual posts shed the prompts that preceded them and the replies that followed them. After all, that data concerns *others*.

▸ *Stagnation:* Without a flow of updates, a captured stock depreciates. Data must be refreshed to stay current, and potential users must see those data updates to stay informed.

▸ *Impotence:* Facts removed from their place of residence become less actionable. We cannot use them to make a purchase when removed from their markets or reach a friend when they are removed from their social networks. Data must be reconnected to be reanimated.

▸ *Market Failure.* Innovation is slowed. Consider how markets for business analytics and B2B services develop. Lacking complete context, third parties can only offer incomplete benchmarking and analysis. Platforms that do offer market overview services can charge monopoly prices because they have context that partners and competitors do not.

▸ *Moral Hazard:* Proposed laws seek to give merchants data portability rights but these entail a problem that competition authorities have not anticipated. Regulators seek to help merchants "multihome," to affiliate with more than one platform. Merchants can take their earned ratings from one platform to another and foster competition. But, when a merchant gains control over its ratings data, magically, low reviews can disappear! Consumers fraudulently edited their personal records under early U.K. open banking rules.[12] With data editing capability, either side can increase fraud, surely not the goal of data portability.

Evidence suggests that following GDPR, E.U. ad effectiveness fell,[7] E.U. Web revenues fell,[5] investment in E.U. startups fell,[6] the stock and flow of apps available in the E.U. fell,[8] while Google and Facebook, who already had user data, gained rather than lost market share[8] as small firms faced new hurdles the incumbents managed to avoid. To date, the results are far from regulators' intentions.

We propose a new *in situ* data right for individuals and firms, and a new theory of benefits. Rather than take data from the platform, or *ex situ* as portability implies, let us grant users the right to use their data in the location where it resides. Bring the algorithms to the data instead of bringing the data to the algorithms. Users determine when and under what conditions third parties access their *in situ* data in exchange for new kinds of benefits. Users can revoke access at any time and third parties must respect that. This patches and repairs the portability problems.

First, all data retains context. Prompts and replies provided by friends and family, sellers and strangers, remain intact. Yet, privacy could even improve relative to portability if data never leaves the system. Third parties need not receive anyone's personal data. By moving the algorithm to the data, not the data to the algorithm, analysis can proceed on masked data that shields identities and details. Encryption can capture context benefits without incurring privacy costs. Second, data retains freshness. All data—all stocks and all flows—is present and current. Third, data retains potency. We can use *in situ* data to make a purchase, place a post, or receive a benefit. We do not need to reconnect to reanimate. Fourth, merchants can pool their *in situ* data and context as they wish, facilitating benchmarking and analytics. Context sharing reduces monopoly hold up of business services. Fifth, merchants and consumers cannot selectively edit unflattering facts and raise the risk for others.

In situ data rights empower users to invite competition on top of the infrastructures where they already have relationships. Amazon might compete on top of Facebook to recommend books based on one's friends. Facebook might compete on top of Amazon to recommend friend groups based on one's readings. A startup could offer new apps and services of benefit to users without the threat of monopoly hold up by the platform itself. Competition to create value follows in a manner that other data rights have yet to enable. Open banking legislation has implemented one small step toward an *in situ* data right. Laws such as the E.U. Payment Services Directive II (PSD2) and the U.K.'s Open Banking Implementation Entity (OBIE) oblige banks to open access to their competitors for payment initiation services. Rather than an obligation of firms in only one sector, this should be a right of persons and firms across all sectors. This seems to work. Innovation and entry rose while fees fell in the financial sector in the E.U. and U.K. after open banking.[13] With *in situ* rights, gains could happen in other sectors too.

By moving our data to portals that would share more value in return, we might capture more of our data value.

A startup's ability to offer new apps and services *not* approved by the platform opens critical benefits like rebalancing oversight. *In situ* rights would enable price and quality comparisons, and foster competition that platforms have tried to prevent.[a] Why should platforms know everything about us, but refuse us basic knowledge about them? Have they influenced elections?[11] Reduced vaccinations?[3] Aided insurrection? Abetted balkanization?[10] Research teams received user permission to track exposure to ads, misinformation and inducements to share photos with more skin.[2] Facebook shut down access, claiming this exposed advertisers' data. Facebook persisted in denying users insights from third-party access the users had themselves invited, despite a scolding from the Federal Trade Commission. Social media platforms disseminate false statements of politicians on the basis that "users should decide," yet refuse to share the context within which those decisions are to be made. Third-party certification services could now identify fake news the platforms continue to spread. Should we not have the right to analyze data platforms push upon us? *In situ* data rights would give us that capability. C

[a] See https://bit.ly/3FFjRh4

References

1. Berlind, D. How Facebook makes it nearly impossible for you to quit. (2017); https://bit.ly/3BEe2xM
2. Dwoskin, E., Zakrzewski, C. and Pager, T. Only Facebook knows the extent of its misinformation problem. And it's not sharing, even with the White House. *Washington Post.* (Aug. 19, 2021); https://wapo.st/3v6PB9N
3. Ghaffary, S. and Heilweil, R. A new bill would hold Facebook responsible for Covid-19 vaccine misinformation. Vox. (July 22, 2021); https://bit.ly/3oTi3Lv
4. Goldberg, S., Johnson, G. and Shriver, S. Regulating privacy online: The early impact of the GDPR on European Web traffic and e-commerce outcomes (2019); *SSRN 3421731.*
5. Janssen, R. et al. GDPR and the Lost Generation of Innovative Apps. NBER Conference on the Economics of Digitization (2021); https://bit.ly/3oWKNTF
6. Jia, J., Jin, G.Z. and Wagman, L. The short-run effects of the general data protection regulation on technology venture investment. *Marketing Science.* (2021).
7. Lefrere, V. et al. The impact of the GDPR on content providers. In *The 2020 Workshop on the Economics of Information Security.* (Dec. 2020).
8. Prasad, A. and Perez, D.R. *The Effects of GDPR on the Digital Economy: Evidence from the Literature. Informatization Policy 27,* 3 (2020), 3–18.
9. Shin, D.H. A study of mobile number portability effects in the United States. *Telematics and Informatics 24,* 1 (2007), 1–14.
10. Van Alstyne, M. and Brynjolfsson, E. Electronic communities: Global village or cyberbalkans? In *Proceedings of the International Conference on Information Systems* (1996), 80–98.
11. Ward, A. 4 main takeaways from new reports on Russia's 2016 election interference. Vox. (Dec. 17, 2018).
12. Zachariadis, M. Data-sharing frameworks in financial services. *Global Risk Institute* (Aug. 2020).
13. Zachariadis, M. and Ozcan, P. The API economy and digital transformation in financial services: The case of open banking. SWIFT Institute Working Paper 2016-001. (2017); https://bit.ly/3lAJmId

Marshall W. Van Alstyne (@InfoEcon; mva@bu.edu) is a Questrom Chair Professor at Boston University where he teaches information economics. He is also a Digital Fellow at the MIT Initiative on the Digital Economy and coauthor of the international best-seller *Platform Revolution* (W.W. Norton).

Georgios Petropoulos (gpetrop@mit.edu) is a Marie Curie Skłodowska Research Fellow at MIT and Bruegel and a Digital Fellow at Stanford University.

Geoffrey Parker (geoffrey.g.parker@dartmouth.edu) is Professor of Engineering at Dartmouth College, a research fellow at the MIT Initiative on the Digital Economy, and coauthor of *Platform Revolution*.

Bertin Martens (Bertin.Martens@ec.europa.eu) is a Senior Economist in the Digital Economy group at the Joint Research Centre of the European Commission (Seville, Spain) and a Research Fellow at the Tilburg Law and Economics Centre at Tilburg University (Netherlands). The views and opinions expressed in this article do not necessarily reflect those of the Joint Research Centre or the European Commission.

The authors express their gratitude for useful feedback provided by Guntram Wolff and the *Communications* Viewpoints editorial board.

Copyright held by authors.

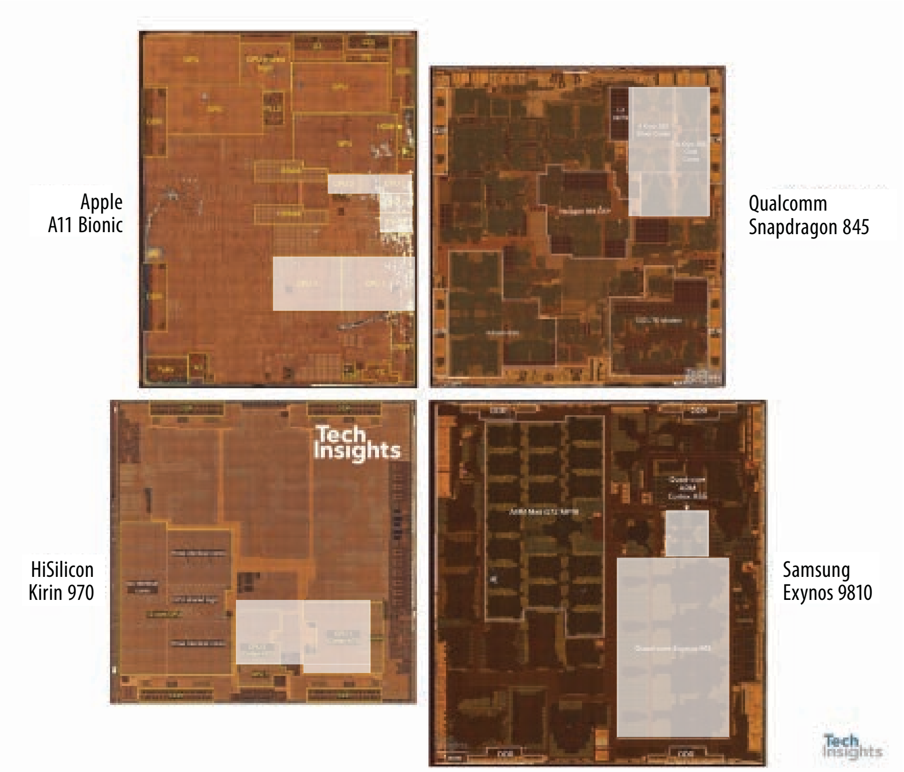
DOI:10.1145/3460970

Mark D. Hill and Vijay Janapa Reddi

Viewpoint
Accelerator-Level Parallelism

Charging computer scientists to develop the science needed to best achieve the performance and cost goals of accelerator-level parallelism hardware and software.

WHILE PAST INFORMATION technology (IT) advances have transformed society, future advances hold great additional promise. For example, we have only just begun to reap the changes from artificial intelligence—especially machine learning—with profound advances expected in medicine, science, education, commerce, and government. All too often forgotten, underlying the IT impact are the dramatic improvements in the programmable hardware. Hardware improvements deliver performance that unlocks new capabilities. However, unlike in the 1990s and early 2000s, tomorrow's performance aspirations must be achieved with much less technological advancement (Moore's Law and Dennard Scaling). How then does one deliver AR/VR, self-driving vehicles, and health wearables at costs that enable great customer value?



Modern System-on-Chip (SoC) architectures. The CPUs in modern SoCs (shown in white) occupy only a small percentage of the die area. The rest of the SoC is committed to a potpourri of different accelerators, such as the DSP, GPU, ISP, NPU, video, and audio codecs.

IMAGE BY TECHINSIGHTS AND CHIPREBEL

One approach that has emerged is to use **accelerators:** *hardware components that execute a targeted computation class faster and usually with much less energy*. An accelerator's flexibility can vary from high (GP-GPU) to low (fixed-function block). Recent work tends to focus on targeting specific application domains, such as graphics (before GPUs generalized), deep machine learning, physics simulations, and genomics. Moreover, most work on accelerators, including in articles appearing in *Communications*,[2,5,6] has focused on CPUs using a single accelerator, with one early forecast of multiple accelerator use.[1]

In our view, many future computing systems will obtain greater efficiency by employing multiple accelerators where each accelerator efficiently targets an aspect of the ongoing computation, much as a Swiss Army knife has specific tools for specific tasks. Smartphones foreshadow this future by employing many accelerators concurrently, but unlike a Swiss Army knife these accelerators often operate in parallel using separately developed software stacks.

We assert there is as yet no "science" for debating and systematically answering basic questions for how to best facilitate broad, flexible, and effective use of multiple accelerators. In this Viewpoint, we expose this opportunity (the what), but charge our readers with determining how best to address it. We review past computer system improvements exploiting levels of parallelism, and introduce Accelerator-Level Parallelism (ALP) as a way to frame new challenges, and expand on the "point" success of smartphone ALP.

Past, Present, and Future Parallelism

As technology scaling provided more and smaller transistors, computer processor architects transformed the transistor bounty into faster processing by using the transistors in parallel. Effectively using repeated transistor doubling required new levels of transistor parallelism. Figure 1 looks at the past and present, and depicts the different levels of parallelism (*y*-axis) that have emerged as computing evolved over the decades (*x*-axis).

In Figure 1, Bit-level parallelism (BLP) refers to performing basic operations (arithmetic, and so forth) in parallel. It was common in early computers and was later enhanced with larger word sizes in commodity systems. Instruction-level parallelism (ILP) is the execution of logically sequential instructions concurrently with pipelining, superscalar, and increasing speculation. Thread-level parallelism (TLP) is the use of multiple processor cores, which initially started with discrete processors and were later integrated as on-chip cores. Data-level parallelism (DLP) pertains to performing similar operations on multiple data operands via arrays and pipelines that achieved broad success via general-purpose graphics processing units (GP-GPUs).

In this Viewpoint and in Figure 1, we assert that another major parallelism level is emerging: Accelerator-Level Parallelism (ALP). We define **ALP** *as the parallelism of workload components concurrently executing on multiple accelerators*. A goal of ALP is to unlock many accelerators at the same time in a manner analogous to how ILP concurrently employs multiple functional units. ALP does not replace other parallelism levels but builds upon them, as most accelerators internally employ one or more of BLP, ILP, TLP, and DLP. Moreover, much like ILP that has been exploited at different levels of the stack, ranging from superscalar and out-of-order execution at the microarchitecture level up to instruction scheduling at the compiler level, ALP opens up many degrees of freedom for novel hardware and software design and optimization. It also opens up possibilities for new runtime resource management, which is analogous to heterogeneous scheduling across CPUs and GPUs, but with the added complexity of scheduling tasks in real time across a sea of hardware accelerators.

ALP is emerging today. Modern chipsets for mobile, edge, and cloud computing are beginning to concurrently employ multiple accelerators. We next present a case study of ALP in mobile SoCs to understand how ALP is currently used, albeit in a somewhat limited form, and then lay a foundation for future work that can exploit ALP more generally.

Mobile SoCs as Harbingers of Multiple Accelerators Using ALP

Driven by the need for extreme energy efficiency, mobile SoCs are the very early adopters of ALP. For SoCs from four major vendors—Apple, Qualcomm, Samsung, and Huawei—much less than 50% of the die is dedicated to the CPUs, as shown in the image on the first page of this Viewpoint. The majority of the area is dedicated to specialized accelerators, such as a Digital Signal Processor, Image Signal Processor, GPU, Neural Processing Unit, and Video Encoder/Decoder, as well as I/O interfaces for audio, networking, video.

It is common in smartphone SoCs for workloads to exhibit ALP with multiple accelerators in concurrent—not exclusive—use. Figure 2 shows a 4K, 60 frame-per-second video capture use case with two paths. One path goes to the display, rendering real-time content to the end user, and the

Figure 1. A snapshot of parallelism over the years, showing how the various forms of parallelism were exploited through different types of architectural mechanisms.

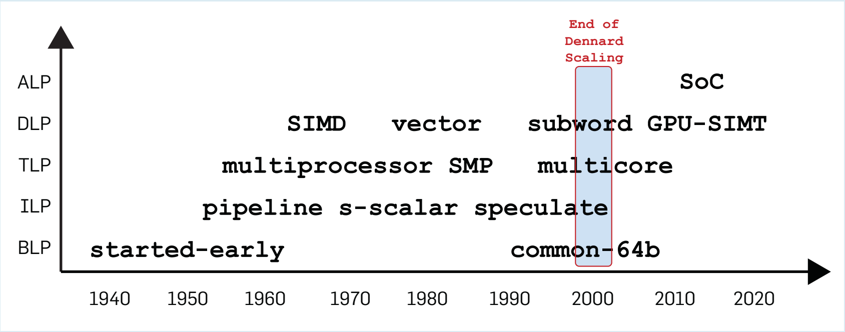
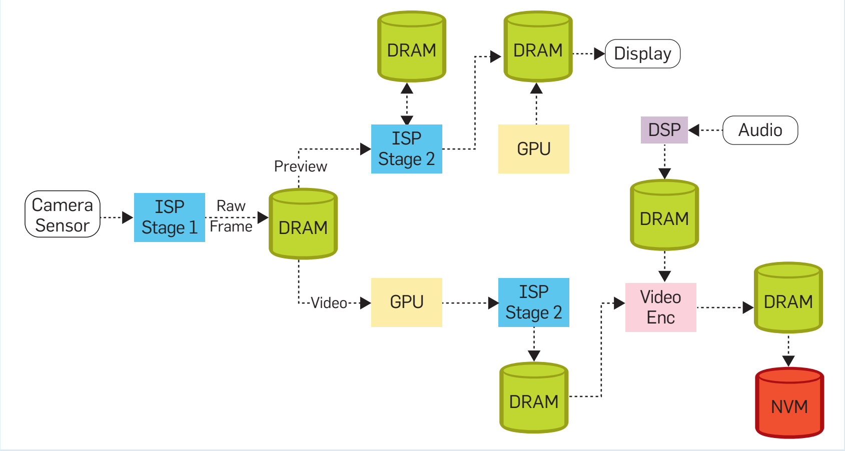


Figure 2. ALP in action in a 4K video capture use case on a smartphone.[7]



other path goes to flash storage to save the content for offline viewing. In this example, data traverses accelerators with both parallelism (two paths) and pipelining, all choreographed by CPUs (not shown). In other use cases like an interactive multiparty videoconferencing application, data flow, and CPU choreographing can be even more dynamic and complex. Nevertheless, we expect accelerators to increasingly handle "data plane" computation while CPUs retain the "control plane" tasks. Doing so will enable richer computation from a fixed power budget, valuable from smartphones to cars to the cloud.

Mobile SoCs are clearly relying on ALP for low-power and efficient execution. However, they are not yet exploiting the full potential of ALP, which we see as needed for recouping the flexibility that the CPU delivered for decades. For instance, in the above example, the dataflow and the binding between the application tasks and accelerators is fixed. The ISP cannot be programmatically repurposed for tasks aside from processing camera image inputs. To this end, we believe we need better science and engineering toward ALP utilization.

Toward a Science for Multiple-Accelerator Systems Using ALP

John Hennessy and David Patterson asserted in their 2018 Turing Award Lecture that we are upon a new golden age for computer architecture.[3] We assert that the challenge put forth by Hennessy and Patterson ought to be generalized to a new golden age for computer science and engineering and that employing multiple accelerators with ALP is an opportunity that opens up new vistas for research as accelerators are integrated into complex SoCs. We do not know all of the possibilities, but we discuss some ideas here to seed research directions.

A key challenge is developing abstractions and implementations to enable programmers to target the whole SoC and implementers to holistically design its software and hardware. We take inspiration from the Single Instruction Multiple Thread (SIMT) model that effectively abstracts GPU hardware's cornucopia of parallelism and scheduling mechanisms. SIMT both enabled GPUs to expand from graphics workloads to general-purpose DLP use and enabled software-hardware implementation improvements beneath the abstraction.

As ALP emerges, we expect new paradigms must be invented to flexibly and effectively exploit its potential. This is not the case today. In contrast to a SIMT-like holistic view, today's SoCs only exploit ALP in limited niches with each accelerator acting as a "silo" with its own programming model, and often its own (domain-specific) language, runtime, software development kit (SDK), and driver interface. While employing multiple accelerators with no abstraction can work in restricted situations (for example, for 10–20 phone use cases), it is unlikely to make ALP generally useful. How can we transcend per-accelerator software silos of different languages, SDKs, and so forth? What are abstractions and mechanisms for scheduling/sequencing accelerators or partitioning/virtualizing them (perhaps stream data flow)? What belongs in runtimes versus above/below the OS hardware abstraction layer?

Even more than previously parallel levels, ALP exploitation will likely require software-hardware co-design due to the heterogeneous nature of accelerators and ALP. Moreover, this is also likely to incentivize computer-aided design tool chain innovations to facilitate the rapid exploration of heterogeneous design spaces. ALP implementations should aspire toward globally optimal software-hardware systems, whereas much good work today focuses on making each accelerator "locally" optimal. While good accelerators are essential, a collection of locally optimal accelerators is unlikely to be globally optimal. For this reason, we need better models[4] and methods for holistically designing SoCs from accelerator, memory, and interconnect components, more like how processor cores are crafted from ALUs, register files, and buses. Analysis in both cases centers on parallel operation: ALP for SoCs and ILP for cores.

In more detail, there are many ALP questions that need better answers and better methods for systematically determining answers. For instance, from a compute perspective, we lack the fundamental science on how we must select, size, make efficient, and sometimes combine similar accelerators? Similarly, from a memory perspective, when should on-chip memory be private to accelerators or shared? When should this memory be a software-visible scratchpad or software-transparent cache? From an integration perspective, how do we best communicate data (shared memory or queues) and control (polling, interrupts, other) among accelerators? From an operational perspective, once an SoC is deployed, can we schedule heterogeneous parallel resources with (non-convex) optimization or must heuristics suffice? In sum, a more systematic approach is needed to design many accelerators as blocks to create holistic ALP systems that excel at performance and cost goals.

Conclusion

This Viewpoint has argued that employing multiple accelerators with ALP has much promise for enhancing future computing efficiency, that we do not yet know how to do it well beyond niches, and that we can work together to make this happen. We have identified *what* the opportunity is, but leave to our readers *how* best to solve it.

References

1. Borkar, S. and Chien, A.A. The future of microprocessors. *Commun. ACM 54*, 5 (May 2011), 67–77; doi: 10.1145/1941487.1941507
2. Dally, W.J., Turakhia, Y., and Han, S. Domain-specific hardware accelerators. *Commun. ACM 63*, 7 (July 2020), 48–57; doi: 10.1145/3361682
3. Hennessy, J.L. and Patterson, D.A. A new golden age for computer architecture. *Commun. ACM 62*, 2 (Feb. 2019), 48–60; doi: 10.1145/3282307
4. Hill, M.D. and Reddi, V.J. Gables: A roofline model for mobile SoCs. In *Proceedings of the High-Performance Computer Architecture (HPCA), 2019 IEEE 25th International Symposium*. 2019.
5. Jouppi, N.P. et al. A domain-specific architecture for deep neural networks. *Commun. ACM 61*, 9 (Sept. 2018), 50–59; 10.1145/3154484
6. Nowatzki, T., Gangadhar, V., and Sankaralingam, K. Heterogeneous von Neumann/dataflow microprocessors. *Commun. ACM 62*, 6 (June 2019), 83–91; 10.1145/3323923
7. Reddi, V.J., Yoon, H., and Knies, A. Two billion devices and counting. *IEEE Micro* (Jan.–Feb. 2018), 6–21.

Mark D. Hill (markhill@cs.wisc.edu) is Hardware Partner Architect at Microsoft and Professor Emeritus at the University of Wisconsin-Madison, Madison, WI, USA.

Vijay Janapa Reddi (vj@eecs.harvard.edu) is an associate professor in the John A. Paulson School of Engineering and Applied Sciences (SEAS) at Harvard University, Cambridge, MA, USA.

Mark D. Hill contributed to this work before joining Microsoft.

Copyright held by authors.

DOI:10.1145/3491297

George V. Neville-Neil

Article development led by acmqueue
queue.acm.org

Kode Vicious
Patent Absurdity

A case when ignorance is the best policy.

Dear KV,
I had been reading through a bunch of patents related to some code I am writing so I could avoid coding up something that was known to be patented. This seemed to be a good idea, but when I told my boss about it, we had to have a meeting with one of the company lawyers where I got to explain which patents I had read. I was then taken off the project and assigned some other work. I think that this was a stupid thing for my manager to do because now the person working on this feature has no clue if the patent is being violated or not. Was I wrong to try to do the requisite research before I started coding up this function?

Made Ignorant of the Law

Dear Made,
If there is one legal issue that ought to be taught to all software engineers, it is, "Don't read patents!" I am sure the company lawyer pointed out that had you not read the patent and violated it, the penalty would be much lower than if you had read the patent, and accidentally violated it. It is trivially easy to accidentally violate a software patent because, of course, lawyers write such patents to be overly broad, and thereby set traps for the unwary coder.

It is, alas, long past the moment when we could have avoided these problems by not allowing software patents at all, for, just like inviting a vampire into your home, once you invite in the lawyers, they will suck you dry. As we have seen over the past 30 years, the only people who profit from software patents are those who weaponize them for profit, and those who abet them (that is, lawyers). The real value of software patents comes not from protecting the intellectual property of "the little guy"—a fictitious character devised by patent lawyers to justify their billable hours—but from being weaponized into portfolios that various large companies can use to manipulate both the market and their competitors.

The main reason a lawyer will give for not reading a software patent is that, if you run afoul of the patent and it can be shown that you had knowledge of it, your company will incur triple the damages that they would have, had you not had knowledge of the patent. That seems like reason enough to avoid reading them, but there is an even better reason, and that is, as design or technical documents, software patents are contemptible.

KV has, on several occasions, had reason to read software patents, because one of the things KV enjoys doing is to help take down patent trolls. Of course, in these cases, KV was not involved in coding up anything relating to the patent but, instead, was looking for prior art or other things that would invalidate the troll's hold on a particular idea or concept.

In all cases where I have had cause to read such documents, my first reaction has been revulsion, but then revulsion is my first reaction to waking up in the morning as well. The revulsion to patents stems from the fact that all the software patents to which I have been exposed have had several things in common: They lay claim to obvious ideas that any software practitioners might come across in their working careers, and they are overly broad, rarely novel, and seem to be written by an infinite number of monkeys attempting to bang out a version of *Hamlet* on ancient typewriters.

All of which is to say they are not legitimate, but you cannot get up in a court of law and say that. Instead, you must spend hours meticulously deconstructing every claim. The claims are not written in either code or plain English, but in a legal code meant to protect the lawyers and their clients' intellectual property. While you might glean an understanding of what the patent intended from reading it, it is more likely you will just wonder why anyone bothered to write the patent at all.

KV often feels that the reason software developers and engineers give these documents any consideration is that they are official documents, blessed by the legal priesthood, and, as such, must have value. They do have value, but that value is not technical in nature, so, as a technologist of any sort, it is best to put down those fancy, wordy documents and let the company lawyer worry about software patents. After all, they are paid three or four times your salary to do so.

KV

Related articles on queue.acm.org

A Time and Place for Standards
Gordon Bell
https://queue.acm.org/detail.cfm?id=1028900

Logging on with KV
https://queue.acm.org/detail.cfm?id=1142039

Evolution or Revolution?
Mache Creeger
https://queue.acm.org/detail.cfm?id=1127873

George V. Neville-Neil (kv@acm.org) is the proprietor of Neville-Neil Consulting and co-chair of the ACM *Queue* editorial board. He works on networking and operating systems code for fun and profit, teaches courses on various programming-related subjects, and encourages your comments, quips, and code snips pertaining to his *Communications* column.

Copyright held by author.

DOI:10.1145/3465999

Rosalind Picard

Viewpoint
What Every Engineer and Computer Scientist Should Know: The Biggest Contributor to Happiness

Seeking the fundamental factors instrumental to happiness.

MY TEAMS AT MIT and our spin-out companies have worked for years to create technology that is both intelligent and able to improve people's lives. Through research drawing from psychiatry, neuroscience, psychology, and affective computing, I have learned some surprising things. In some cases, they are principles we have embedded into technology that interacts with people. Guess what? People like it. After one year of the COVID-19 pandemic, I realize that the principles we learned apply not only to making smart robots or software agents, but also to the people around us. They give us lessons for how to live happier lives, and happier engineers are better at solving creative problems and have more fun.

IMAGE BY MINERVA STUDIO

Researchers have studied what brings happiness in life, and what, at the end of life, people wish they had done. While many factors contribute, do you know the biggest one?

Almost never late in life do people say: "I wish I had invented a smarter or faster device," "I wish I had made more money," "I wish I had given more TED talks," "I wish I had climbed higher in my business," or "I wish I had authored more books." Even this pinnacle of achievement is not uttered: "I wish I had written an article for an ACM magazine." Instead, almost always, people wish that they had done a better job at building meaningful authentic human relationships, and spending time in those relationships.

This finding is a general one, whether studying human happiness or end-of-life reflections. They apply to hard-working, well-educated computer scientists or engineers and also to many kinds of people, different races and cultures, rich and poor, male and female, uneducated or overeducated.

All of the patents, publications, presentations, and personal technical achievements can be amazing: They can literally save lives and bring immense delight, win us world acclaim, fill our shelves with awards, tally up clicks online, and even make our resumes impressively long. However, they all pale in comparison to something that is even more joy-giving: Achieving deeply satisfying, personally-significant human relationships.

How do you engineer great relationships? Here are three helpful principles you can test in your own life and relationships. If you build AI that interacts directly with people, you can build these principles into those interactions too. I learned these principles while trying to engineer more intelligence in machines, specifically computers with skills of social-emotional intelligence. The skills derive from studies of human relationships and they apply not only when the interactions involve two people, but also when one is a computer (including chatbots, software agents, robots, and other things programmed to talk with us). The three principles below can help improve relationships, human or AI.

Principle #1: Feedback is essential to learning and improving. A machine learning algorithm can improve its performance if you give it accurate feedback such as +1 "correct" or -1 "wrong," or other corrective labels or procedural guidance; similarly, your relationship skills get better when you seek and incorporate feedback. Here are two examples how to do this, and you can invent variations:

- Ask, "Hey, at the end of this month, would you please tell me three things I did that you liked, and one thing that I could have done better?" Then mark your calendar with a time you'll meet to get the feedback. The 3:1 Good:Bad ratio acknowledges human hedonic asymmetry. It is easier to digest negative feedback if it is weighted by a greater quantity of positive feedback. Do this for 12 months and observe your relationship-improving learning gains grow.

How do you engineer great relationships?

- After a conversation or interaction that did not go well, and later when things are calm and positive, softly ask: "Would you be willing to talk about how [that prior interaction] went? I'd like to hear how you felt about it, and how I could have done my part better." If they agree, then (here's the hard part): Listen without interrupting, and practice Principle #2.

Principle #2: Conversations contain errors that need to be detected and processed intelligently. The speaker may not accurately encode their meaning into words, or you may hear their words inaccurately, or you may hear accurately, but perceive them differently than the speaker intended. Other kinds of errors can happen too. Here is a two-step way to fix errors, and you may develop others:

- Repeat back what you heard, but not exactly. Using some of the same words is good because it shows that you were listening. Also, it has been shown that people like people more when they reuse their words. Using slightly different paraphrasing of what you heard is best, as it demonstrates that you might have been thinking about what you heard. Also, you do not want to annoy the other person by sounding like a parrot or a badly programmed chatbot.[a] Showing them evidence that you were listening can be more important than listening.
- Give them a chance to adjust what they said, after you tell them what you received. They know you have an imperfect brain (sorry) and possibly also imperfect ears. You will not perfectly receive what they send. They may not realize that they also sent it imperfectly. Thus, enable them to observe what you received, so it can be fixed if needed.

This is easier to do than to describe. For example, person P might tell you,

"It made me mad when you said YZX."

Now a short quiz for you: Which reply is best?

(a) "It made you mad when I said YZX"

Feedback is essential to learning and improving.

(b) "It sounds like I made you upset when I said YZX. *Is that the case?*"

They are both good, but (b) is better. Why? Because you don't sound like a parrot and you used Principle #1 to invite them to correct you. It is wise to solicit feedback, because you might not have it right, even if you used the right words. This principle is especially vital when the message describes feelings.

Note that you could say back to them exactly what they said, and they might still say you got it wrong. "How is that rational?" you might ask. It is rational because, by hearing it reflected back from you, they may suddenly realize that what they said was not actually what they intended to communicate. The error may have been theirs, encoding their message unsuccessfully. In short, you can help engineer better communication and a better relationship by putting redundancy into the channel, and making observable to them what it is that you heard. The result is that the other person feels better understood.

Note that the shortcuts, "I understand you" or "I understand what you feel" are usually counterproductive. "I understand" does not communicate to them what you understand. It is essentially a time-wasting phrase. The communication is not complete until you tell them what it is that you understand. It can be especially productive to try to put into words what feeling it is that you think they are having, for example, "It sounds like you might be feeling frustrated …" and keep trying to get it right until they say "That's right." The process of showing you understand well enough to restate it in your own words makes for a stronger relationship, which moves you both closer to happiness.

An intriguing fact is that *none* of the principles described here involve

[a] If you *do* want to annoy them, then that case is not covered here. However, you might want to reread paragraphs one and two of this Viewpoint and reassess your life goals.

acm

ACM Transactions on Computing for Healthcare

ACM Transactions on Computing for Healthcare (HEALTH) is a multi-disciplinary 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.

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

saying anything clever, or coming up with good fixes to problems. Computer scientists and engineers are almost always trying to fix things and make them better. The method above actually does something more powerful: It gives the other person the gift of feeling understood. This gift often has the result of freeing up their cognitive-affective resources so that they can fix their own problem. Sometimes, showing understanding of feelings can make the relationship better than the cleverest "fix."

Computer scientists and engineers are almost always trying to fix things and make them better.

Principle #3: Active, constructive responses generate joy. Suppose you are really busy working on your own deadline, and your colleague interrupts with news he's super-happy about, "Hey, you know that proposal I worked on? It got selected to get complete funding!" Which of the following would be your likely response?

(a) They selected *your* ideas? Don't you realize this is going to be so much extra work—maybe they are exploiting you?

(b) "Oh? Really?" (Then look back at your screen, "Ugh, so much email.")

(c) Congratulations!

(d) Congratulations! What a nice recognition of your effort! (Perhaps you raise your hand to high-five.) How do you feel about that? (You listen and share smiles, celebrating the moment's joy.)

While none of these options takes more than a minute, which of them is likely to deepen and improve the relationship? Of all of the options, generally the strategy in (d) is best—it actively amplifies the moment's positive feelings into something even larger and greater, helping construct a better relationship. By speaking words (less than 10 seconds), you give them a gift of sharing and enlarging their joy.

While (a), (b) and (c) save a few seconds, what are seconds compared to a lifetime when a relationship is on the line? The responses in (a) and (b) leave the person's happy mood ignored or diminished—they are termed "destructive" by experts in Positive Psychology, as they take wind out of the sails of the relationship. While (c) is not destructive, it is passive. It can be easily transformed into (d), actively promoting your colleague's joy and even amplifying your own joy. A few seconds saying something like (d) can boost moods, bringing manifold benefits. A positive mood boost can expand a person's ability to solve problems efficiently, and a positive social interaction today is also associated with a reduction in stress tomorrow. That is a lot of benefit for a few seconds difference.

The three principles described here are not perfect or complete, and this Viewpoint has omitted significant details like the importance of sincerity, context, forgiveness, and authentic empathy and love. Nonetheless, they provide valuable guidelines for building better relationships. Principles 1, 2, and 3, can all be used to inform the engineering of a better chatbot, robot, AI dialogue system, or customer support system, one with more social-emotional intelligence. More importantly, the principles can be used with anyone—children, spouses, friends, bosses, colleagues, and even enemies. The best way to get rid of your enemies is to make them your friends, and these principles can help to turn any relationship in a more positive direction. Yes, these principles even apply during videoconferencing and physical distancing. With these three principles, you can make progress toward deeper and more meaningful relationships, and that is a key step toward achieving greater lifelong happiness. C

Rosalind Picard (picard@media.mit.edu) is a professor at the MIT Media Lab, Cambridge, MA, USA.

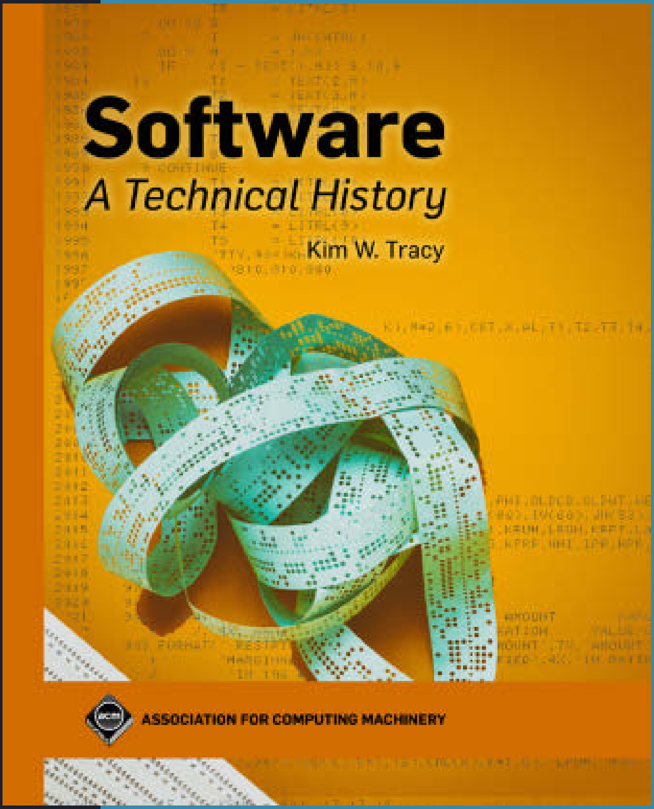
Copyright held by author.

ACM BOOKS

Collection II

Software history has a deep impact on current software designers, computer scientists, and technologists. System constraints imposed in the past and the designs that responded to them are often unknown or poorly understood by students and practitioners, yet modern software systems often include “old” software and “historical” programming techniques. This work looks at software history through specific software areas to develop student-consumable practices, design principles, lessons learned, and trends useful in current and future software design. It also exposes key areas that are widely used in modern software, yet infrequently taught in computing programs. Written as a textbook, this book uses specific cases from the past and present to explore the impact of software trends and techniques.

Building on concepts from the history of science and technology, software history examines such areas as fundamentals, operating systems, programming languages, programming environments, networking, and databases. These topics are covered from their earliest beginnings to their modern variants. There are focused case studies on UNIX, APL, SAGE, GNU Emacs, Autoflow, Internet protocols, System R, and others. Extensive problems and suggested projects enable readers to deeply delve into the history of software in areas that interest them most.



Software

A Technical History

Kim W. Tracy

ISBN: 978-1-4503-8725-5
DOI: 10.1145/3477339

http://books.acm.org
http://store.morganclaypool.com/acm

DOI:10.1145/3475161

Article development led by acmqueue
queue.acm.org

Algorithms, microcontent, and the vanishing distinction between platforms and creators.

BY LIU LEQI, DYLAN HADFIELD-MENELL, AND ZACHARY C. LIPTON

When Curation Becomes Creation

EVER SINCE SOCIAL activity on the Internet began migrating from the wilds of the open Web to the walled gardens erected by so-called *platforms* (think Myspace, Facebook, Twitter, YouTube, or TikTok), debates have raged about the responsibilities these platforms ought to bear. And yet, despite intense scrutiny from the news media and grassroots movements of outraged users, platforms continue to operate, from a legal standpoint, on the friendliest terms.

You might say today's platforms enjoy a "have your cake, eat it too, and here's a side of ice cream" deal. They simultaneously benefit from: broad discretion to organize (and censor) content however they choose; powerful algorithms for curating a practically limitless supply of user-posted microcontent according to whatever ends they wish; and absolution from almost any liability associated with that content.

This favorable regulatory environment results from the current legal framework, which distinguishes between *intermediaries* (for example, platforms) and *content providers*. This distinction is ill-adapted to the modern social media landscape, where platforms deploy powerful data-driven algorithms (so-called AI) to play an increasingly active role in shaping what people see, and where users supply disconnected bits of raw content (tweets, photos, and so on) as fodder.

Specifically, under Section 230 of the Telecommunications Act of 1996, "interactive computer services" are shielded from liability for information produced by "information content providers." While this provision was originally intended to protect telecommunications companies and Internet service providers from liability for content that merely passed through their

IMAGE BY ALICIA KUBISTA/ANDRIJ BORYS ASSOCIATES

plumbing,[1] the designation now shelters services such as Facebook, Twitter, and YouTube, which actively shape user experiences.

With the exception of obligations to take down specific categories of content (for example, child pornography and copyright violations), today's platforms have license to monetize whatever content they like, moderate if and when it aligns with their corporate objectives, and curate their content however they wish.

Antecedents in Moderation

In his 2018 book, *Custodians of the Internet*,[3] Tarleton Gillespie examines platforms through the lens of content moderation, calling into focus an apparent contradiction: Platforms constantly do (and, arguably, must) wade into the normative, making political decisions about what content to allow; and yet they operate absent responsibility on account of their purported *neutrality*.

Throughout, Gillespie is evenhanded, expressing sympathy for platforms' predicament. They must moderate, and all mainstream platforms do. Without moderation, platforms are readily taken over by harassers and robots; and yet no moderation policy is value neutral.

Flash points in the moderation debates include years-long protests over Facebook's policy of classifying (and later declassifying) breastfeeding photographs as "obscene" content; Facebook's controversial policy of taking down obscene but historically significant images, such as the Pulitzer Prize-winning "Napalm Girl" photograph notable for its role in bending public opinion on the Vietnam War; and, following the January 6 Capitol Hill riots, the wave of account suspensions that swept across Twitter, Facebook, Amazon, and even Pinterest.

In all these cases, platforms faced consequences in the marketplace, as well as brand-management challenges. From a legal standpoint, however, their autonomy has seldom been challenged.

In the end, Gillespie provokes his readers to reconsider whether platforms should be entrusted with decisions that are inevitably political and affect all of us. Analyzing platforms through the lens of moderation raises fundamental questions about the sufficiency of current regulations. The moderation lens, however, seldom forces us to question the very validity of the intermediary-creator distinction.

What Is Content Creation?

This article argues that major changes in both the technology used to curate content and the nature of user content itself are rapidly eroding the boundary between intermediaries and creators.

First, breakthroughs in machine-learning algorithms and systems for intelligently assembling the underlying content into curated experiences have given companies the power to determine with unprecedented control not only what *can be seen*, but also what *will be seen* by users in service of whatever metric a company believes serves its business objectives.

Second, unlike traditional bulletin board sites for sharing links to entire articles, or blogging platforms for sharing article-length musings, modern social media giants such as Facebook and Twitter traffic primarily (and increasingly) in microcontent—isolated snippets of text and photographs floating a la carte through their ecosystems.

Third, the largest platforms operate on such an enormous scale that their content contains nearly any assertion of fact (true or false), nearly any normative assertion (however extreme), and nearly any photograph (real or fake) floating through the zeitgeist.

Platforms now enjoy vast expressive power to create media products for their users, limited only by the available atomic content and by the power of their algorithms, both of which are advancing rapidly because of economies of scale and advances in technology, respectively.

We are not the first to suggest that curation fundamentally alters the distinction between platforms and creators. In a recently proposed amendment to Section 230, motivated by more pragmatic regulatory concerns, U.S. Representatives Anna G. Eshoo (D-CA) and Tom Malinowski (D-NJ) recently proposed to reclassify those "interactive computer service[s]" (platforms) that "used an algorithm, model, other computational process to rank, order, promote, recommend, amplify, or similarly alter the delivery or display of information" as "information content provider[s]" (creators).[2]

To be clear that the interpretation of these legal terms is faithful to the original meaning in Section 230, here is the official definition:

> The term *information content provider* means any person or entity that is responsible, in whole or in part, for the creation or development of information provided through the Internet or any other interactive computer service.

As technology advances, the murky line between curation and creation is likely to become less, not more, distinct.

Immediate legal goals aside, why target (algorithmic) content curation? At first glance, it might seem absurd that by virtue of curating content, an Internet service should assume not only some measure of responsibility, but also the very same status, vis-à-vis liability, as the creators of the underlying content. This distinction, however, may not actually be so far-fetched.

Similar debates have arisen in the arts. Who can claim responsibility for a pop song that heavily samples preexisting audio? Are the Beastie Boys the creators of *Paul's Boutique*, or do the creators of the original snippets have a sole right to that distinction? Can Jasper Johns be considered the creator of his prints and collages that repackage and juxtapose previous works of art (by himself and others)?

With such derived works, claims to creatorship, rights to the spoils, and liability need not be mutually exclusive. This precedent suggests at least one sphere of life where people appear to be comfortable with the idea that those who produce microcontent and those who assemble it into larger-scale works can share the designation of *creator*.

Of course, the line must be drawn somewhere. The DJ does not create the music in the same way that the Beastie Boys do. Art galleries do not create art in the same way that Jasper Johns does. Beneath the neat system of legal categories lies a messy spectrum of creative activities.

When Does Curation Become Creation?

Returning to the activities of Web platforms, let's consider two extremes on the curation-creation spectrum. First, let's look at the activities of a typical aggregator website such as the Drudge Report, whose content consists entirely of outbound links to full articles that exist elsewhere on the Internet. Arguably, Drudge plays the role of the DJ, creating something more like a playlist than a song.

Now, consider the typical online blogger or the typical overworked journalist of the online era offering commentary or synthesis but not original reporting. They scour the Internet for content, assembling words, phrases, whole quotes, and photographs, all of which could be found elsewhere, to produce an article or post. Most readers undoubtedly concur that this qualifies as creation. Indeed, it is creation in the same sense that Twitter and Facebook users are creators of the content they post.

Now, consider the middle ground, where someone fashions content by assembling neither whole articles, nor individual words, but instead individual sentences, drawn from the entirety of the Internet, stripped of their original context, and assembled to present any desired picture of the discourse surrounding any topic.

Legal scholars and politicians can debate whether this middle ground warrants official categorization as creation versus curation. It's difficult to deny, however, that these acts indeed constitute a spectrum, and that the curator of sentences bears greater resemblance to the curator of words than does the curator of articles.

Today's platforms have been creeping steadily along this spectrum. From the earliest days, when a comparatively puny reservoir of content was presented in reverse chronological order, to the modern era's black-box systems that power Twitter's and Facebook's news feeds, there is a shifting landscape of actors that look less and less like disinterested utilities happy to transport any content that shows up in their plumbing and more and more like active creators of a media product.

To be sure, activity along this spectrum is not uniform, even within a single platform. Take Twitter, for example: While the default news feed is indeed customized according to an opaque process, the content consists mostly of recent posts by (or retweeted by) individuals whom you follow. On the other hand, Twitter's Explore screen bears a striking resemblance to the middle-ground curator of sentences. They both present a set of hot topics, each titled according to some unknown process, and they curate, from the (often) millions of tweets on a topic, a chosen set to represent the story.

In an era where many journalistic articles appearing in traditional venues consist of curated sets of tweets loosely connected by narrative and interpretation, the line separating intermediary from creator has grown so thin as to suggest the possibility that a double standard is already at play.

Where Do We Go Next?

While the focus here is on actions that platforms take to present content, this is not the only way they influence the information a user consumes. Platforms like Twitter and Facebook regularly translate messages across languages. Image-sharing platforms, such as Instagram and Snapchat, apply algorithmic transformations to photographs.

As technology advances, the murky line between curation and creation is likely to become less, not more, distinct.

In the future, platforms might not only translate across languages, but also paraphrase across dialects[5] or provide content summaries.[7] They may move past applying cute filters and render whole synthetic images to specification.[4] Perhaps to mollify users aghast at the toxicity of the Web, Twitter and Facebook might offer features to render messages more polite.[6]

Coming up with policies that balance the competing desiderata of corporate accountability, economic vibrancy, and individual rights to free speech is difficult. This article does not presume to champion a single point on the curation-creation spectrum as the one true cutoff. Nor does it purport to offer definitive guidance on the viability of a system predicated on such a distinction in the first place.

Instead, the goal here is to elucidate that there is indeed a spectrum between curation and creation. Furthermore, technological advances provide platforms with a powerful, diverse, and growing set of tools with which to build products that exist in the gray area between "interactive computer services" and "information content providers."

Regulating this influential and growing sector of the Internet requires recognition of the essential gray-scale nature of the problem and that we eschew reductive regulatory frameworks that shoehorn all online actors into simplistic systems of categorization.

At some point, the increasing influence that modern platforms wield over user experiences must be accompanied by greater responsibilities. It is difficult to decide the precise point along the intermediary-creator spectrum at which platforms should assume liability. The bill proposed by Representatives Eshoo and Malinowski suggests that such a point has already been reached. Surely, Facebook's legal team would disagree. What is clear, however, is that today's platforms play a growing role in creating media products and that any coherent regulatory framework must adapt to this reality. C

References

1. Electronic Frontier Foundation. CDA 230: legislative history; https://www.eff.org/issues/cda230/legislative-history.
2. Eshoo, A.G. Reps. Eshoo and Malinowski introduce bill to hold tech platforms liable for algorithmic promotion of extremism, 2020; https://eshoo.house.gov/media/press-releases/reps-eshoo-and-malinowski-introduce-bill-hold-tech-platforms-liable-algorithmic.
3. Gillespie, T. *Custodians of the Internet: Platforms, Content Moderation, and the Hidden Decisions that Shape Social Media.* Yale University Press, 2018.
4. Koh, J. Y., Baldridge, J., Lee, H., Yang, Y. Text-to-image generation grounded by fine-grained user attention. In *Proceedings of the 2021 IEEE/CVF Winter Conf. Applications of Computer Vision*, 237–246; https://bit.ly/379CxFK.
5. Lewis, M., Ghazvininejad, M., Ghosh, G., Aghajanyan, A., Wang, S., Zettlemoyer, L. Pre-training via paraphrasing. *Advances in Neural Information Processing Systems 33* (2020); https://proceedings.neurips.cc/paper/2020/hash/d6f1dd034aabde7657e6680444ceff62-Abstract.html.
6. Madaan, A., Setlur, A., Parekh, T., Poczos, B., Neubig, G., Yang, Y., Salakhutdinov, R., Black, A.W., Prabhumoye, S. Politeness transfer: A tag and generate approach. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, 1869–1881; https://www.aclweb.org/anthology/2020.acl-main.169.pdf.
7. Wang, X., Yu, C. Summarizing news articles using question-and-answer pairs via learning. In *Proceedings of the 2019 Intern. Semantic Web Conf.*, 698–715; https://research.google/pubs/pub48295/.

Liu Leqi is a Ph.D. student in the Machine Learning Department at Carnegie Mellon University, Pittsburgh, PA, USA. Her research interests include AI and human-centered problems in machine learning.

Dylan Hadfield-Menell is an assistant professor of artificial intelligence and decision-making at the Massachusetts Institute of Technology, Cambridge, MA, USA. His recent work focuses on the risks of (over-) optimizing proxy metrics in AI systems.

Zachary C. Lipton is the BP Junior Chair Assistant Professor of Operations Research and Machine Learning at Carnegie Mellon University, Pittsburgh, PA, USA, and a Visiting Scientist at Amazon AI. He directs the Approximately Correct Machine Intelligence (ACMI) lab, whose research spans core machine learning methods, applications to clinical medicine and NLP, and the impact of automation on social systems. He can be found on Twitter (@zacharylipton), GitHub (@zackchase), or his lab's website (acmilab.org).

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

DOI:10.1145/3475358

Article development led by acmqueue
queue.acm.org

A user interface for querying provenance.

BY ASHISH GEHANI, RAZA AHMAD, HASSAAN IRSHAD, JIANQIAO ZHU, AND JIGNESH PATEL

Digging into Big Provenance (with SPADE)

DATA PROVENANCE DESCRIBES the origins of a digital artifact. It explains the creation of an object, as well as all the modifications and transformations that transpired over its lifetime. When the historical record is detailed, spans long periods, or both, the information collected can become voluminous. Analysis of provenance is often used even while it is continuously being extended through a series of computations that act upon it. This necessitates a framework that supports performant streaming ingestion of new elements with concurrent querying that yields responses that incorporate data as it becomes available.

Operating systems and blockchains are two of many domains where collection and analysis of *big provenance*[9] has had useful applications. In the case of operating systems, system-call information collected by a kernel's audit framework can form the basis of trustworthy provenance metadata. This facilitates tracking all activity that occurs across a machine or even a federated system. This *whole network provenance*[1] is particularly useful for applications such as malware detection and ensuring the reproducibility of computation.

Bitcoin is a blockchain-based cryptocurrency where individuals can perform transactions with each other. Each transaction between two or more users contains payment information that should be stored in the blockchain. These records form the basis for tracking the provenance of any given digital object in the blockchain. In addition to its primary purpose of tracking currency ownership, the provenance has other applications, such as detecting anomalous behavior to identify illegal activity.

Provenance metadata may be stored in a database to facilitate efficient querying. The process of interrogating the system must be intuitive and convenient to use since finding the relevant fragment in big provenance is akin to the proverbial search for a needle in a haystack. These goals must be met despite the system's use in a variety of domains, from profiling complex application workflows to performing forensic and impact analyses after attacks on a system have been uncovered.

Several interfaces exist for querying provenance. Many of them are not flexible in allowing users to select a database type of their choice. Some interfaces provide query functionality in a data model that is different from the graph-oriented one that is natural for provenance. Other interfaces have intuitive constructs for finding results but have limited support for efficiently chaining responses, as needed for *faceted search*. This article presents a user interface for querying provenance that addresses these concerns and is agnostic to the underlying database being used.

First, relevant background on data provenance is provided, along with

IMAGE BY L.DEP/SHUTTERSTOCK

how it is modeled and how the representation is realized in an open source implementation. Then the design of the query surface is presented, its core functionality outlined, illustrative use cases described, and salient aspects of the system highlighted.

Modeling Computational History

A common way of reporting data provenance is to model it as a graph structure, where vertices represent elements in a historical record, and edges represent events that relate and order the elements. A provenance graph $G(V,E)$ then contains a set of vertices, V, and edges, E. A member, v, of the set V can be an *agent*, *process*, or *artifact* that was involved in an event. Each edge e belonging to set E represents the operation that occurred and relates two vertices, v_i and v_j.

Multiple data models have been developed to represent data provenance. Notable variants are the OPM (Open Provenance Model),[11] published in 2010; the W3C PROV specification,[13] released in 2013; and the DARPA Transparent Computing program's CDM (Common Data Model),[10] finalized in 2019. They have some similarity. Each includes vertices for three categories of elements: agents or principals; processes, activities, or subjects; and artifacts, entities, or objects. They differ in detail based on their intended domain of use: OPM was designed to be domain-agnostic; W3C PROV was created to aid the publication of semantically enriched Web content; and CDM is focused on the specific domain of operating systems.

The semantics of the activity domain being monitored are captured with a custom schema. By using a property graph to represent the provenance, these details can be embedded directly. Vertices and edges are each accompanied by a (possibly empty) set, A, of domain-describing annotations, $A = a_1, a_2, \ldots, a_n$. Each annotation ai is a key-value pair—that is, $a_i = key_i : value_i$—that reports an aspect of the domain, such as `program:firefox` or `path:/etc/passwd`. In this manner, a provenance graph captures the relative order of events, as well as the salient aspects of the monitored domain.

As an example, consider a vertex representing an operating system process. This vertex could have annotations conveying information such as its name, identifier, or start time. An edge could relate the process to a file that has been read. In the case of bitcoin provenance,[7] a vertex representing a transaction would have annotations such as the hash that identifies it and the earliest time it is valid. An edge could relate unspent bitcoin to a payee with an annotation specifying the amount.

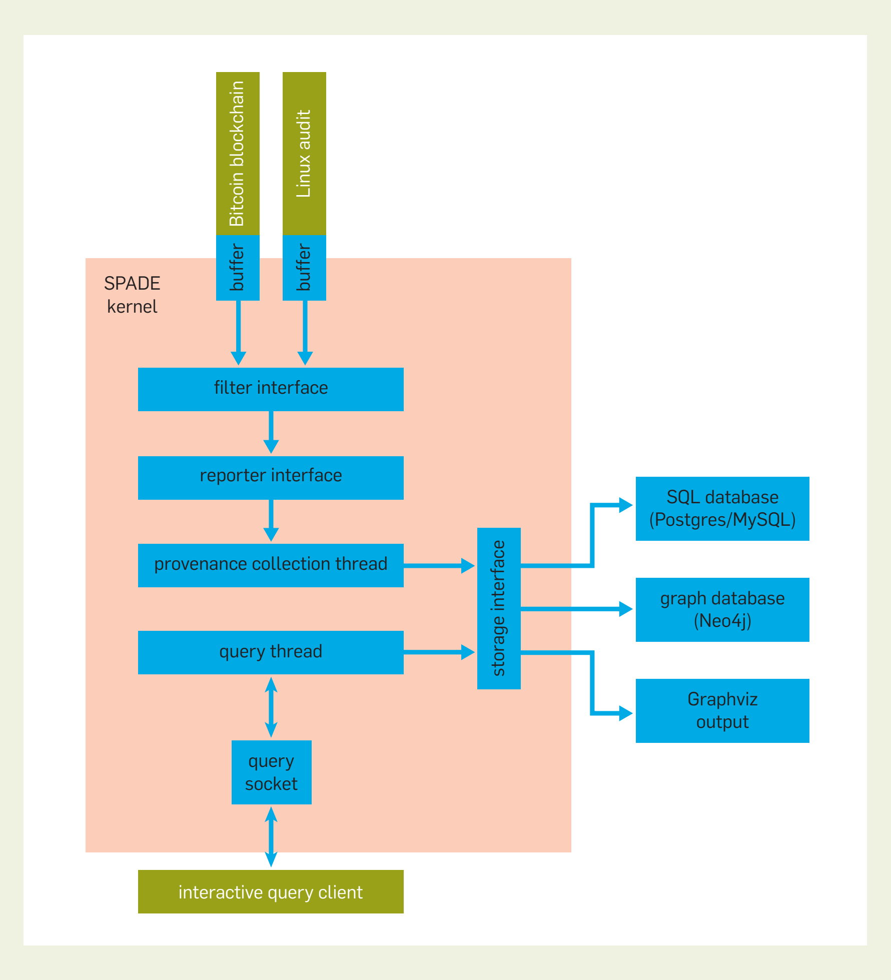
Canonical Provenance Queries

The simplest provenance query consists of searching for vertices that match a specification, defined by an expression over the annotations that describe it. Such queries are useful to locate vertices that can serve as the starting point of more complex queries, like the ones described in this article. Akin to retrieving vertices are queries for identifying individual edges. For example, a user may want to learn about all cases where the permissions of a particular file were changed. Since this is an atomic system event, the user can search for all associated provenance edges. Similarly, once an edge (or a set of them) has been located, the user can extract the endpoint vertices. In the prior example, this would allow the identification of processes that performed the action.

Among the most frequently needed functionality when operating on provenance records is support for finding the lineage of an element. In a lineage query, the ancestry of a data artifact is traced back a specified number of steps. Similar functionality that operates in the other direction is useful for identifying descendants. The ancestors or descendants of a given data artifact are found by recursively locating the parent or child vertices in the graph structure, respectively. The ancestral lineage of an item provides a picture of what transpired leading up to the creation of that item, while the descendant variant describes what was derived from it after its creation. With operating system provenance, the lineage of a file can explain how, when, and by whom that file was created. It can support the enumeration of all the system processes (and their owners) that wrote to or read from a file. In the case of bitcoin, the lineage of a payment reveals details about the participating users and all the transactions linked to them.

Another important operation consists of searching for paths between a pair of elements. Two variants of this operation often arise in practice. The first involves finding all the paths between two vertices, while the second focuses on finding the shortest path between them. A path between two vertices demonstrates how different data elements influence each other through the events that have transpired in the system. For example, finding a path between a Web browser application and a file downloaded from the Internet shows the complete set of steps required by the browser's user for bringing the file from a remote server to a local machine. In the bitcoin context, paths between two addresses can be used to find the transactions that link those two addresses. This, in turn, can be used to calculate the assets that have flowed from one user to another, even when they go through intermediaries.

Figure 1. The SPADE architecture has a kernel at its core.



Provenance System Architecture

The open source SPADE project[3] provides software for inferring, storing, and querying data provenance. It is cross-platform and can be used with diverse sources such as blockchains, online social networks, and multiple operating systems, including Linux, macOS, and Windows. The collection of provenance is done without requiring any change in the applications or the target platform. SPADE is easy to install and configure. It provides a simple mechanism for users to select from many storage formats.

The architecture of SPADE is shown in Figure 1. It is composed of multiple modules, each playing an independent role in processing provenance records. These modules are managed by the SPADE *Kernel* at the core. Provenance graph elements are inferred about activity domains and sent to the Kernel by *Reporter* modules. After being operated on by any *Filters* present, the elements are sent to *Storage* modules that insert the elements into databases configured with a custom schema. The presence of independent threads for ingestion and query processing allows clients to make provenance inquiries while the underlying graph is changing. The various types of modules are described further later.

A Reporter module acts as the producer of provenance metadata. It receives streams of events from diverse sources, extracts relevant information from them, and infers provenance relationships, constructing graph vertices and edges in the process. SPADE provides a multitude of reporters that generate provenance about diverse domains, including operating systems, blockchains, intra-application calls, and user-defined schema.

A Filter module acts on the provenance stream emitted by a reporter. It performs a selection operation on the provenance according to programmed criteria. For example, some filters allow only the vertices and edges that match a specification to pass through. Other filters abstract or remove information in vertices or edges. The output of a filter is the processed provenance information that is meant to be persisted. Several filters can be inserted to operate sequentially on the output of each preceding one.

A Storage module takes the final output of the sequence of filters (if any are present) and stores it in one of the available databases. The module provides an abstraction over the underlying persistent store. This subsystem could be a relational database such as MySQL or Postgres, a graph database such as Neo4j, or any data store. The module provides interfaces for storing and retrieving data that are agnostic to the underlying database.

An Analyzer module provides an interface to the user for retrieving provenance records stored in SPADE. It is responsible for receiving a query, sending it to the appropriate storage, processing the information, and sharing the result with the user. The default implementation receives queries from the command line.

SPADE provides a multitude of reporters that generate provenance about diverse domains, including operating systems, blockchains, intra-application calls, and user-defined schema.

Design of the Query Surface

The first generation of SPADE and its precursors introduced support for querying a file's lineage, specified by its path and version (as of a given date and time). It included several optimizations for transferring provenance metadata across hosts[4] and accelerating cryptographic verification of such records.[5] It was not until the second generation[6] that a richer query surface was added, including support for retrieving vertices, edges, paths, and lineage.

To make querying more usable for navigating big provenance and performing faceted searches, a new surface was developed. Its name, QuickGrail, derives from the fact that its design was inspired by the Grail project[2] and initially implemented atop the Quickstep database.[12] Subsequently, SPADE added support for using QuickGrail with the Neo4j graph and Postgres relational databases.

QuickGrail provides an abstraction over the underlying database so that users can work with a uniform query language, regardless of the data model and serialization below. This allows users to focus on the provenance analysis task at hand without concern for how the queries will be translated into the native language of the database. In addition to being efficient, the surface provides a uniform mechanism for exporting responses for visualization and other external uses.

The query interface provides a collection of functions to search for provenance records and manipulate the responses retrieved from the underlying database. The databases supported are Neo4j, Postgres, and Quickstep. Each function in the query surface is implemented in an intermediate representation that is translated into the query language of the target database.

The system provides several features that facilitate *faceted search*, allowing the user to home in on information of interest to them. One such feature is the ability to assign a query response to a graph variable. Such variables can be operated upon in subsequent queries to refine the search. Another feature that facilitates efficient user interaction in the presence of big provenance is the ability to limit the size of responses. This allows users to send queries and quickly receive partial responses, which they can inspect to refine their search.

Variables. A special variable, $base, represents the entire provenance graph stored in the currently selected database. This variable serves as the universe of provenance for most queries. When a variable is used to store the *response graph* from a query, it will appear on the left-hand side of an assignment (denot-

Figure 2.

```
%firefox = name == ‘firefox’
$init_vertex = $base.getVertex(%firefox)
$firefox_lineage = $base.getLineage($init_vertex, 2, ‘ancestors’)
```

Figure 3.

```
%firefox_threads = name LIKE ‘firefox%’
$firefox_skeleton = $base.getVertex(%firefox_threads)
$firefox_process = $base.getSubgraph($firefox_skeleton)
```

ed with =), and its name must start with $. If a variable is used to define a *query constraint*, its name must start with %. Such variables are a convenience, providing a succinct way to represent constraints that may need to be passed repeatedly as arguments to queries.

Constraints. To scope the elements that match a query, a selection *constraint* can be specified. In its most basic form, it consists of an annotation key, a relational operator, and a value. (Recall that annotations were introduced earlier in this article.) The supported operators include ==, !=, <, >, <=, >=, and LIKE. The last of these facilitates matching a string (with % used as a wildcard). For example, name LIKE '/bin/%' will match vertices with an annotation-key name that has a value starting with /bin/. A constraint of uid == '0' can be used to select vertices of processes that ran as root (since its uid is 0). To simplify reuse, constraints can be stored in variables—for example, %system _ procs = name LIKE '/bin/%'.

To support more complex filtering, constraint expressions can be built by combining constituents using the logical operators AND, OR, and NOT. This allows the constraint expressions to be framed over multiple annotations. In addition, it allows the user to combine an existing constraint with a new criterion, thereby facilitating faceted search. For example, consider the constraint %proc _ python = name == 'python' used to find vertices representing python executions. The querier may be interested in the subset that ran as the root user. In this case, the querier could define another constraint %proc _ root = uid == '0'. The two constraints can then be combined into a single expression %proc _ python AND %proc _ root for use in subsequent queries.

Extracting elements. Since provenance graphs consist of vertices and edges, the most basic functions provided are getVertex and getEdge. They can be used with any existing graph variable to extract a subset of elements, as specified by a constraint. The output of these functions must be assigned to a graph variable. Note that even though these functions result in sets of vertices or edges, these sets are treated as graphs. In the following example, vertices are extracted from the special variable $base, which represents the global graph. They are constrained to the subset with an annotation type of Process:

```
%only _ processes = type == 'Process'
$all _ processes = $base.
getVertex(%only _ processes)
```

Similarly, every edge is extracted if it has an annotation of operation with value fork:

```
%all _ forks = operation == 'fork'
$fork _ edges = $base.getEdge(%all _
forks)
```

Using the set of edges just obtained, the next query extracts the processes that performed the fork operations, as well as those that were created as a result:

```
$fork _ vertices = $fork _ edges.
getEdgeEndpoints()
```

Identifying origins and impacts. Given an element in a provenance graph, a central concern is understanding what gave rise to it. Of equal importance is understanding what has been affected by a particular element. In both cases, this is done by starting with the element, finding its parents or children, respectively, and then recursing. This is supported with the getLineage function, which is generalized to operate on a set of seed elements. It takes three arguments: a set of *seed vertices*; the maximum *number of levels* to traverse from the seed vertices, which must be a positive integer; and the *direction* of traversal, which can be ancestors, descendants, or both. The example in Figure 2 extracts two levels of the ancestral lineage of vertices with a firefox annotation:

Connecting the dots. A preliminary analysis (through a faceted search using increasingly specific constraints, for example) may lead to two sets of vertices being identified: One may consist of the ingress points of network flows into the system, while the second set may have indicators of compromise, such as processes whose privilege was escalated or files whose ownership changed in a particular window of time.

Knowing whether a connection exists between two sets is a key concern. This can be ascertained with the getPath function, which takes three arguments: a set of *source vertices*; a set of *destination vertices*; and the maximum *path length* between any source and destination vertex, which must be a positive integer. The semantics of provenance imply that a path will be found only when a destination is in the provenance—that is, it is an ancestor—of a source. The following example searches for paths with a length of at most three edges between a firefox process vertex and the /etc/passwd file vertex:

```
%source = name == ‘firefox’
$firefox = $base.getVertex(%source)
%destination = path == ‘/etc/passwd’
$etc _ passwd = $base.
getVertex(%destination)
$paths = $base.getPath($firefox,
$etc _ passwd, 3)
```

If the set of paths discovered is large, it can be refined by specifying one or more sets of intermediate vertices. For example, if it is known that a $compromised_process set lies on the paths of interest from $firefox to $etc_passwd, the query can be made more specific:

```
$paths = $base.getPath($firefox,
  $compromised_process, 3, $etc_passwd,
3)
```

Filling in the missing pieces. Early in an investigation, specific agents, activities, and artifacts may be known to be of interest, but not all elements (including the relationships between them) may be known. In such cases, the analyst can define a set of interesting vertices and then ask the system to describe how they are related to each other. For example, a set of suspicious network connections and files with modified ownership may be identified in a specified timeframe. The analyst may then wish to know if and how any of these elements are related.

In a generalization, the analyst can include edges in the set to incorporate information about known provenance relations of interest. This can be effected *in toto* with the `getSubgraph` function, which takes as input a *skeleton graph*. The skeleton is a set of vertices and edges known to be of interest *a priori*. The function returns the provenance subgraph that spans all elements in the skeleton, as well as those that lie on paths between vertices and edge endpoints in the skeleton.

The example in Figure 3 shows the provenance subgraph returned will show a vertex for each thread of the several dozen that Firefox creates, each configuration and cache file that is accessed, and each socket used for interprocess communication, as well as the provenance relations between them.

Going native. Commodity databases provide diverse query surfaces. The set of primitives supported depends on factors such as the data model employed and the indexing implemented in the underlying engine. Relational databases such as Postgres and Quickstep offer an interface based on SQL (Structured Query Language). Graph databases, such as Neo4j, use Cypher, a graph-oriented declarative analog. Since each database may support custom queries that could be useful to an analyst, a facility is provided to access them. If a query is preceded with the keyword native, it will be passed unmodified to the underlying database. The response will be returned as lines of text rather than as a graph. This allows arbitrary native queries to be invoked.

As an example, consider an operating system provenance graph in OPM, with *Artifact* vertices refined by subtype, including `file`, `link`, `directory`, `block device`, `character device`, `named pipe`, `unnamed pipe`, `unix socket`, and `network socket`. In a preliminary analysis, the distribution of these elements may be of interest to identify unusual patterns. In this case, counts for each subtype can be obtained from Postgres with a user-defined function *histogram*:

```
native 'SELECT * FROM
histogram(vertex, subtype)'
```

Complex Provenance Analysis

When large data sets are analyzed, the process is often iterative. An analyst may construct numerous hypotheses, checking whether each is valid or not by querying the data. As an investigation unfolds, maintaining the workflow's efficiency requires that intermediate results are represented succinctly to avoid I/O bandwidth becoming a bottleneck. In practice, search is often faceted, with the results of one step reused in subsequent ones. It may also involve backtracking and comparing the extracted subsets of data. When results of potential interest are retrieved, visualization or other external processing may allow an analyst to obtain a broad understanding of a selected subset. The query surface has several features that address these concerns. Together, they facilitate agile exploration.

Efficient representation. SPADE models provenance as a property graph. The annotation schema is selected to ensure hashing them will produce a unique content-based identifier for each vertex and edge. When a query is executed, only the identifiers implicated in the response are associated with the graph variable used to track a response. Effectively, only a skeletal representation that consists of an adjacency list for the corresponding subgraph is constructed. The enriched representation with graph properties in the form of key-value annotations is not immediately materialized. These properties, which describe the domain about which provenance was inferred, use most of the storage needed to hold the complete graph. Their retrieval is avoided until a response is explicitly exported, either to the console or a file with the dump command, respectively. This allows an analyst to make queries that may yield large responses without disrupting their interactive workflow (as would occur if the complete response were to be materialized).

Consider the sequence in Figure 4. The graph variables $sources, $destinations, and $paths track only the identifiers of the implicated vertices and edges. Annotations of elements in the graph $paths are retrieved from the database only when dump $paths is explicitly issued, for example.

Response reuse. When the query client initiates a session, a local workspace is created to store the graph responses received. Each graph is bound to a variable name, simplifying its repeated use. Such variables can be used in one of two ways. First, since a variable represents a graph, it can be treated as the universe that will be operated upon by subsequent queries. Second, the variable can instead be passed as the argument of a query.

In Figure 5, the last query uses `$processes` instead of `$base` as the provenance universe in which to search for all process vertices that have a name

Figure 4.

```
$sources = $base.getVertex(name == 'firefox')
$destinations = $base.getVertex(path == '/etc/passwd')
$paths = $base.getPath($sources, $destinations, 3)
```

Figure 5.

```
%type_process = type == 'Process'
$processes = $base.getVertex(%type_process)
%firefox_threads = name LIKE 'firefox%'
$firefox_parents = $processes.getVertex(%firefox_threads)
```

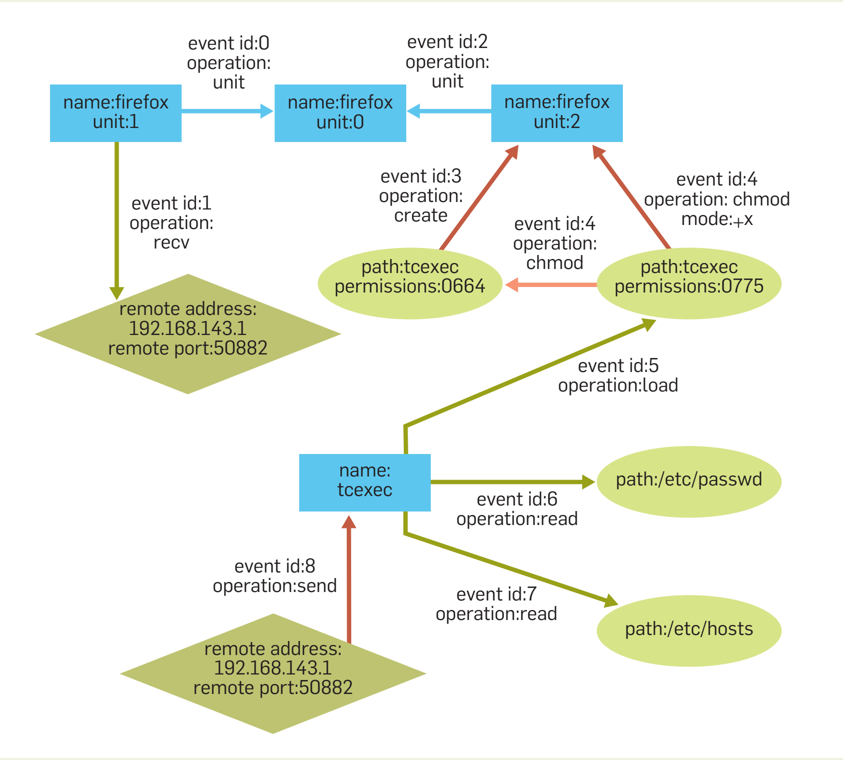
Figure 6.

```
$setuid_operations = $base.getEdge(operation == 'setuid')
$chameleons = $setuid_operations.getEdgeDestination()
$privilege_escalated = $chameleons - $chameleons.getVertex(uid != 0)
```

Figure 7.

```
$firefox_vertices = $base.getVertex(name LIKE 'firefox%')
$firefox_sample = $firefox_vertices.limit(10)
dump $firefox_sample
```

Figure 8. Provenance relations between processes and artifacts.



that starts with `firefox`.

As a session progresses, the set of currently defined variables can be identified with the command `list graph`. The `stat` command can be used to get statistics about a particular graph. For example, `stat $paths` reports the number of vertices and edges in the graph named `$paths`. The reuse of variable names is supported by destroying a binding with erase <variable name>. This eliminates the skeletal representation associated with the variable.

Set manipulation. Initial inspection of the provenance may leave an analyst with a collection of large subgraphs that require further refinement. For example, knowledge about the activity domain can be leveraged to identify subsets of the graph that are of particular interest, as described earlier. More specifically, queries framed over the domain-specific annotations can lift collections of vertices and edges from the underlying database into the workspace; these seed sets may the n be expanded through path and lineage queries.

To facilitate symbolic manipulation of the graphs, a complementary suite of operations is provided. They realize intuitive mathematical set operations in the setting of graphs. Pairs of graphs can be transformed into a union of constituents with the + operator. The result contains a vertex set that is the union of vertices in the operand graphs. Similarly, the resulting edge set is the union of edges in the operands. Alternatively, the intersection of two graphs can be calculated with the & operator. The resulting graph will contain only vertices and edges that were present in both the operand graphs. Finally, elements in a graph can be removed based on the specification of a second graph. This is effected with the *difference* operator.

Consider a situation where an analyst wishes to determine the set of processes that changed their identity during execution. First, they extract the set of all edges that report a change in identity. Next, they extract the endpoints of these edges, representing the processes that issued the `setuid()` call. The subset initially running as root, however, is not of interest in this context. Hence, such processes are removed by subtracting the corresponding set in the last step (see Figure 6).

Graph export. Since the graph that results from a query may be large, it is not immediately materialized. Instead, a graph can be used in three ways. First, it can be printed to the console in JSON (JavaScript Object Notation) format. The output is an array of vertices and edges. Each element consists of one or two identifiers—depending on whether it is a vertex or an edge—and the annotations that describe it.

In Figure 7, an analyst inspects a subset of the contents of a graph. This is done by extracting a sample (10 elements in this instance) using the `limit` function and then printing them with the `dump` command. This motif is instrumental during a faceted search, where an analyst may iteratively refine the queries based on a study of successive intermediate results.

The second way to use a graph is by exporting it to a file or pipe in JSON format. This allows it to be imported or ingested by an external tool. To affect this, an export directive is used to specify the file-system path immediately before using `dump`. For example, the graph variable `$firefox _ vertices` can be serialized to file `/tmp/firefox.json` with:

```
export > /tmp/firefox.json
dump $firefox _ vertices
```

Finally, support is provided for exporting the graph to the widely used Graphviz DOT format. This allows it to be visualized in several forms, depending on the layout tool used to render it. The mechanics are like the previous method, with an export (specifying where the DOT data should be sent) preceding use of the dump command:

```
export > /tmp/firefox.dot
dump $firefox_vertices
```

Illustrative Use Cases

This section presents use cases from two domains that were introduced earlier: an operating system and a blockchain. Provenance is queried in a post-event analysis scenario.

Operating systems. Consider a setting where provenance is inferred from system calls, as it is with SPADE's Audit Reporter on Linux, OpenBSM on macOS, and ProcMon on Windows. The resulting graph captures the interactions among users, processes, and data artifacts. As a motivating use case, consider the challenge a system administrator is faced with after a compromise. The nature and extent of the damage inflicted on the target host must be identified. This can range from determining a malware infection's source to identifying which data has been exfiltrated and which system configurations have been modified.

Now consider an example inspired by attacks seen in practice, as illustrated in Figure 8. Understanding the steps of an attack is simplified by analyzing the abstracted provenance relations between processes and artifacts in the system. Assume an application (`firefox`) accepts a malicious request via a remote connection. This exploits an existing vulnerability in the program. It causes the executing process to be hijacked, with the adversary gaining control of it. Data is written to the location of a binary (`tcexec`). The permissions of the modified file are updated to ensure it is executable. Subsequently, when this binary runs, it accesses system files and exfiltrates them to a remote host.

In Figure 9, a forensic analyst can reconstruct what transpired with a set of queries. At the outset, the analyst is assumed to know *a priori* that it was the `firefox` process that was hijacked after browsing a malicious website.

Bitcoin is used in dark Web (and other) markets.[8] Each payment is made to a specific address that denotes a user. Every successful transaction is recorded in a block that becomes part of a public ledger, the bitcoin blockchain. SPADE's Bitcoin Reporter can be used to infer the provenance graph that relates individual addresses, transactions, and blocks together. The next example assumes the blockchain has been imported into a database supported by QuickGrail. This allows forensic analysts to track the flow of funds through the bitcoin ecosystem. For example, they may wish to identify all the sources of

Figure 9.

1. Determine if a Web browser executed a file that was downloaded from a remote network connection.
   (a) Get the vertices that represent a Firefox Web browser.

```
$firefox = $base.getVertex(“command line” LIKE ‘%firefox%’)
```

(b) Get the vertices that represent a file that is world readable, writable, and executable.

```
$executableFiles = $base.getVertex(subtype
== ‘file’ AND permissions == ‘0777’)
```

(c) Get the vertices that represent network connections.

```
$networkConnections = $base.getVertex(subtype
== ‘network socket’)
```

(d) Get the paths where (1) a Firefox process reads data from a network connection, and

(2) the same Firefox process updates permissions of an executable file.

```
$potentialAttackersEntryPath
= $base.getPath($executableFiles, $firefox,
1, $networkConnections, 1)
```

(e) Get the files that were executable and written by Firefox.

```
$potentiallyExecutedFiles
= $potentialAttackersEntryPath &
$executableFiles
```

2. Determine whether any written files were executed by the Web browser.
   (a) Get the vertices that represent processes.

```
$allProcesses = $base.getVertex(type == ‘Process’)
```

(b) Get the vertices that represent processes that were started by Firefox.

```
$firefoxChildren = $base.getPath($allProcesses,
$firefox, 1).getEdgeSource()
```

(c) Get the Firefox children that accessed the written files.

```
$firefoxChildrenAccessedExecutableFile
= $base.getPath($firefoxChildren,
    $potentiallyExecutedFiles, 1).getEdgeSource()
```

3. Determine whether a process accessed sensitive system files and then sent information out through a network connection.

(a) Get the vertices that represent system files /etc/passwd, /etc/group, and /etc/hosts.

```
$systemFiles = $base.getVertex(path
== ‘/etc/passwd’ OR path == ‘/etc/group’ OR path
== ‘/etc/hosts’)
```

(b) Get the paths from network connections that were written to by Firefox children that read system files.

```
$exfiltrationPath = $base.
getPath($networkConnections,
$firefoxChildrenAccessedExecutableFile, 1,
$systemFiles, 1)
```

a particular transaction. Alternatively, they may want to check if there is a path from one bitcoin address to another.

In this example, the analysts start with a bitcoin address found on a website soliciting donations to support illegal activity. Initially, they check whether a specific address has sent any payment. The search is limited to five levels of indirection.

```
$donation _ address = $base.
getVertex(address
         == '13Pcmh4dKJE8Aqrhq4ZZ-
         wmM1sbKFcMQEE')
$payer _ candidate = $base.
getVertex(address
         == 'ZwmbK4ZdKJ3PcQEmh-
         8MEAqrhq41FcEM1s')
$paths = $base.getPath($donation _
address,
         $payer _ candidate, 5)
```

Next, the analysts retrieve all payers whose funds reached the donation address either through direct payment or via an intermediary.

```
$payers = $base.
getLineage($donation _ address, 2,
'descendants')
```

Life Cycle of a Query

Instructions to download, build, and run SPADE are available online.[3] Assuming it is running, the query client can be used interactively after it is started with the command spade query executed at the command line of a shell. It is also possible to pipe commands to it and responses from it by redirecting standard input and standard output, respectively.

The directive set `storage <name>` can be issued in the client to change the current default database. This assumes that the corresponding SPADE storage has been added previously. At this point, a session is created. Any queries made now will be sent to the selected database. A query session will continue until an exit command is issued.

Initial inspection of the provenance may leave an analyst with a collection of large subgraphs that require further refinement.

Acknowledgments

This material is based on work supported by the National Science Foundation under Grant ACI-1547467. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

1. Ahmad, R., Jung, E., de Senne Garcia, C., Irshad, H., Gehani, A. Discrepancy detection in whole network provenance. In *Proceedings of the 12th USENIX Workshop on the Theory and Practice of Provenance*; https://www.usenix.org/conference/tapp2020/presentation/ahmad.
2. Fan, J., Gerald, A., Raj, S., Patel, J. The case against specialized graph analytics engines. In *Proceedings of the 7th Biennial Conf. on Innovative Data Systems*, 2015; http://cidrdb.org/cidr2015/Papers/CIDR15_Paper20.pdf.
3. Gehani, A. SPADE; http://spade.csl.sri.com.
4. Gehani, A., Kim, M., Zhang, J. Steps toward managing lineage metadata in grid clusters. In *Proceedings of the 1st Usenix Workshop on Theory and Practice of Provenance*, 2009, 1–9; https://dl.acm.org/doi/10.5555/1525932.1525939.
5. Gehani, A., Kim, M. Mendel: Efficiently verifying the lineage of data modified in multiple trust domains, *Proceedings of the 19th ACM Intern. Symp. High Performance Distributed Computing* 2010; https://dl.acm.org/doi/abs/10.1145/1851476.1851503 227-239.
6. Gehani, A., Tariq, D. SPADE: Support for provenance auditing in distributed environments. In *Proceedings of the 13th ACM/IFIP/Usenix Middleware Conf.*; 2012; https://dl.acm.org/doi/pdf/10.5555/2442626.2442634.
7. Gehani, A., Kazmi, H., Irshad, H. Scaling SPADE to "Big Provenance." In *Proceedings of the 8th Usenix Workshop on Theory and Practice of Provenance*, 2016, 26–33; https://www.usenix.org/conference/tapp16/workshop-program/presentation/gehani.
8. Ghosh, S., Das, A., Porras, P., Yegneswaran, V., Gehani, A. Automated categorization of onion sites for analyzing the dark web ecosystem. In *Proceedings of the 23rd ACM Intern. Conf. Knowledge Discovery and Data Mining*, 2017, 1793–1802; https://dl.acm.org/doi/10.1145/3097983.3098193.
9. Glavic, B. Big data provenance: challenges and implications for benchmarking. *Revised Selected Papers of the 1st Workshop on Specifying Big Data Benchmarks 8163*, 2012, 72–80; https://dl.acm.org/doi/10.1007/978-3-642-53974-9_7.
10. Khoury, J., Upthegrove, T., Caro, A., Benyo, B., Kong, D. An event-based data model for granular information flow tracking. *Proceedings of the 12th Usenix Workshop on the Theory and Practice of Provenance*, 2020; https://www.usenix.org/biblio-4496.
11. Moreau, L. et al. The Open Provenance Model core specification. *Future Generation Computer Systems 27*, 6 (2011); https://dl.acm.org/doi/10.1016/j.future.2010.07.005.
12. Patel, J., Deshmukh, H., Zhu, J., Potti, N., Zhang, Z., Spehlmann, M., Memisoglu, H., Saurabh, S. Quickstep: A data platform based on the scaling-up approach. In *Proceedings of the VLDB Endowment 11*, 6 (2018), 663–676; https://dl.acm.org/doi/10.14778/3184470.3184471.
13. W3C Working Group. PROV-overview, 2013; https://www.w3.org/TR/prov-overview/.

Ashish Gehani is a principal computer scientist at SRI in Menlo Park, CA, USA.

Raza Ahmad is a research engineer at DePaul University. Chicago, IL,USA. He developed SPADE's QuickGrail back ends for Neo4j and Postgres.

Hassaan Irshad is a software engineer in the CS laboratory at SRI, Menlo Park, CA, USA. He maintains the SPADE framework.

Jianqiao Zhu is a software engineer at Google. He is a technical lead on the kernel execution team of the F1 Query engine. He developed SPADE's QuickGrail and its back end for Quickstep.

Jignesh Patel is a CS professor at the University of Wisconsin, Madison, WI, USA, and co-leads the Center for Creative Destruction Labs.

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

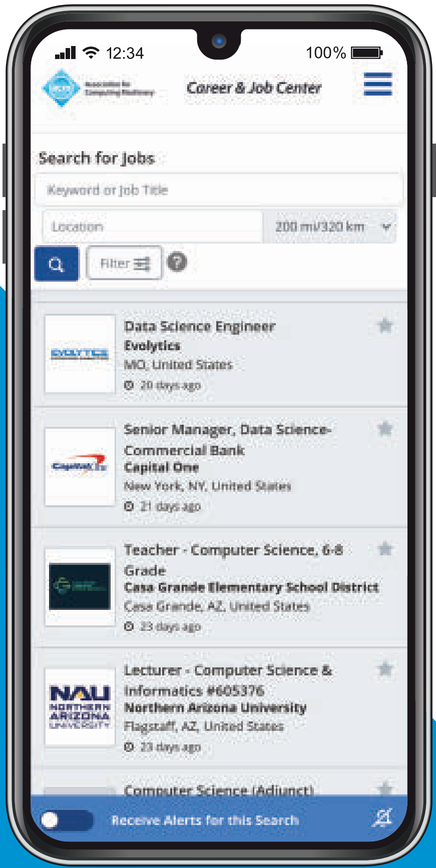
Association for Computing Machinery

Career & Job Center

The #1 Career Destination to Find Computing Jobs.

Connecting you with top industry employers.

The new ACM Career & Job Center offers job seekers a host of career-enhancing benefits, including:



12:34 100%

Career & Job Center

Search for Jobs

Keyword or Job Title

Location | 200 mi/320 km

Filter

Data Science Engineer
Evolytics
MO, United States
20 days ago

Senior Manager, Data Science-Commercial Bank
Capital One
New York, NY, United States
21 days ago

Teacher - Computer Science, 6-8 Grade
Casa Grande Elementary School District
Casa Grande, AZ, United States
23 days ago

Lecturer - Computer Science & Informatics #605376
Northern Arizona University
Flagstaff, AZ, United States
23 days ago

Computer Science (Adjunct)

Receive Alerts for this Search

Access to new and exclusive career resources, articles, job searching tips and tools.

Gain insights and detailed data on the computing industry, including salary, job outlook, 'day in the life' videos, education, and more with our new Career Insights.

Redesigned job search page allows you to view jobs with improved search filtering such as salary, location radius searching and more without ever having to leave the search results.

Receive the latest jobs delivered straight to your inbox with **new exclusive Job Flash™ emails**.

Get a free resume review from an expert writer listing your strengths, weaknesses, and suggestions to give you the best chance of landing an interview.

Receive an alert every time a job becomes available that matches your personal profile, skills, interests, and preferred location(s).

Your next job is right at your fingertips.
Get started today!

Visit https://jobs.acm.org/

DOI:10.1145/3467017

After decades of incentivizing the isolation of hardware, software, and algorithm development, the catalysts for closer collaboration are changing the paradigm.

BY SARA HOOKER

The Hardware Lottery

HISTORY TELLS US that scientific progress is imperfect. Intellectual traditions and available tooling can prejudice scientists away from some ideas and towards others.[24] This adds noise to the marketplace of ideas and often means there is inertia in recognizing promising directions of research. In the field of artificial intelligence (AI) research, this article posits that it is tooling which has played a disproportionately large role in deciding which ideas succeed and which fail.

What follows is part position paper and part historical review. I introduce the term "hardware lottery" to describe when a research idea wins because it is compatible with available software and hardware, not because the idea is superior to alternative research directions. The choices about software and hardware have often played decisive roles in deciding the winners and losers in early computer science history.

These lessons are particularly salient as we move into a new era of closer collaboration between the hardware, software, and machine-learning research communities. After decades of treating hardware, software, and algorithm as separate choices, the catalysts for closer collaboration include changing hardware economics, a "bigger-is-better" race in the size of deep-learning architectures, and the dizzying requirements of deploying machine learning to edge devices.

Closer collaboration is centered on a wave of new-generation, "domain-specific" hardware that optimizes for the commercial use cases of deep neural networks. While domain specialization creates important efficiency gains for mainstream research focused on deep neural networks, it arguably makes it even more costly to veer off the beaten path of research ideas. An increasingly fragmented hardware landscape means that the gains from progress in computing will be increasingly uneven. While deep neural networks have clear commercial use cases, there are early warning signs that the path to the next breakthrough in AI may require an entirely different combination of algorithm, hardware, and software.

This article begins by acknowledging a crucial paradox: machine-learning researchers mostly ignore hardware despite the role it plays in determining which ideas succeed. The siloed evolution of hardware, software, and algorithm has played a critical role in early hardware and software lotteries. This article considers the ramifications of this siloed evolution

» key insights

- The term hardware lottery describes a research idea that wins due to its compatibility with available software and hardware, not its superiority over alternative research directions.
- We may be in the midst of a present-day hardware lottery. Hardware design has prioritized delivering on commercial use cases, while built-in flexibility to accommodate the next generation of ideas remains a secondary consideration.
- Any attempt to avoid future hardware lotteries must be concerned with making it cheaper and less time consuming to explore different hardware/software/algorithm combinations.

IMAGE BY ANDRIJ BORYS ASSOCIATES

BONUS
TCH YOUR HARDWARE
NEW
ARE YOU A TECH WINNER?
$$$
BIGG
THE BETTE
SCALE
ODE
WILL YOUR IDEA WIN
SET NEW
30 / 30

with examples of early hardware and software lotteries. And, while today's hardware landscape is increasingly heterogeneous, I posit that the hardware lottery has not gone away, and the gap between the winners and losers will grow. After unpacking these arguments, the article concludes with thoughts on how to avoid future hardware lotteries.

Separate Tribes

For the creators of the first computers, the program was the machine. Due to both the cost of the electronics and a lack of cross-purpose software, early machines were single use; they were not expected to be repurposed for a new task (Figure 1). Charles Babbage's "difference engine" (1817) was solely intended to compute polynomial functions.[9] IBM's Harvard Mark I (1944) was a programmable calculator.[22] Rosenblatt's perceptron machine (1958) computed a stepwise single-layer network.[48] Even the Jacquard loom (1804), often thought of as one of the first programmable machines, was, in practice, so expensive to re-thread that it was typically threaded once to support a pre-fixed set of input fields.[36]

In the early 1960s, joint specialization of hardware and software went vertical. IBM was an early pioneer in the creation of instruction sets, which were portable between computers. A growing business could install a small IBM 360 computer and not be forced to relearn everything when migrating to a bigger 360 machine. Competitors Burroughs, Cray, and Honeywell all developed their own systems—compatible with their own machines but not across manufacturers. Programs could be ported between different machines from the same manufacturer, but not on competitive machines. The design itself remained siloed, with hardware and software developed jointly in-house.

Today, in contrast to the specialization necessary in computing's very early days, machine-learning researchers tend to think of hardware, software, and algorithms as three separate choices. This is largely due to a period in computer science history that radically changed the type of hardware that was produced and incentivized the hardware, software, and machine-learning research communities to evolve in isolation.

The general-purpose computer era crystallized in 1969, when a young engineer named Gordan Moore penned an opinion piece in *Electronics* magazine titled, "Cramming More Components onto Circuit Boards."[33] In it, Moore predicted that the number of transistors on an integrated circuit could be doubled every two years. The article and subsequent follow-up were originally motivated by a simple desire—Moore thought it would sell more chips. However, the prediction held and motivated a remarkable decline in the cost of transforming energy into information over the next 50 years.

Moore's law combined with Dennard scaling[12] enabled a factor of three-magnitude increase in microprocessor performance from 1980 to 2010. The predictable increases in computing power and memory every two years meant hardware design became risk averse. Why experiment on more specialized hardware designs for an uncertain reward when Moore's law allowed chip makers to lock in predictable profit margins? Even for tasks which demanded higher performance, the benefits of moving to specialized hardware could be quickly eclipsed by the next generation of general-purpose hardware with ever-growing computing power.

The emphasis shifted to universal processors, which could solve myriad different tasks. The few attempts to deviate and produce specialized supercomputers for research were financially unsustainable and short-lived. A few very narrow tasks, such as mastering chess, were an exception to this rule because the prestige and visibility of beating a human adversary attracted corporate sponsorship.[34]

Figure 1. Early computers were single use and were not expected to be repurposed. These machines could not be expected to run the variety of programs our modern-day machines do.

Treating the choice of hardware, software, and algorithm as independent has persisted until recently. It is expensive to explore new types of hardware, both in terms of time and capital required. Producing a next-generation chip typically costs $30-$80 million and takes two to three years to develop.[14] These formidable barriers to entry have produced a hardware research culture that might feel odd or perhaps even slow to the average machine-learning researcher. While the number of machine-learning publications has grown exponentially in the last 30 years, the number of hardware publications has maintained a fairly even cadence.[42] For a hardware company, leakage of intellectual property can make or break the survival of the firm. This has led to a much more closely guarded research culture.

In the absence of any lever with which to influence hardware development, machine-learning researchers rationally began to treat hardware as a sunk cost to work around rather than something fluid that could be shaped. However, just because we have abstracted hardware away does not mean it has ceased to exist. Early computer science history tells us there are many hardware lotteries where the choice of hardware and software has determined which ideas succeed and which fail.

The Hardware Lottery

The first sentence of Tolstoy's *Anna Karenina* reads, "All happy families are alike; every unhappy family is unhappy in its own way."[47] Tolstoy is saying that it takes many different things for a marriage to be happy—financial stability, chemistry, shared values, healthy offspring. However, it only takes one of these aspects to not be present for a family to be unhappy. This has been popularized as the Anna Karenina principle, "a deficiency in any one of a number of factors dooms an endeavor to failure."[32]

Despite our preference to believe algorithms succeed or fail in isolation, history tells us that most computer science breakthroughs follow the Anna

Karenina principle. Successful breakthroughs are often distinguished from failures by benefiting from multiple criteria aligning surreptitiously. For AI research, this often depends upon winning what we have named the *hardware lottery*—avoiding possible points of failure in downstream hardware and software choices.

An early example of a hardware lottery is the analytical machine (1837). Charles Babbage was a computer pioneer who designed a machine that could be programmed, at least in theory, to solve any type of computation. His analytical engine was never built, in part because he had difficulty fabricating parts with the correct precision.[25] The electromagnetic technology required to actually build the theoretical foundations laid down by Babbage only surfaced during WWII. In the first part of the 20th century, electronic vacuum tubes were heavily used for radio communication and radar. During WWII, these vacuum tubes were repurposed to provide the computing power necessary to break the German enigma code.[10]

As noted in the TV show *Silicon Valley*, often "being too early is the same as being wrong." When Babbage passed away in 1871, there was no continuous path between his ideas and modern computing. The concept of a stored program, modifiable code, memory, and conditional branching were rediscovered a century later because the right tools existed to empirically show that the idea worked.

While domain specialization creates important efficiency gains for mainstream research focused on deep neural networks, it arguably makes it even more costly to veer off the beaten path of research ideas.

The Lost Decades

Perhaps the most salient example of the damage caused by not winning the hardware lottery is the delayed recognition of deep neural networks as a promising direction of research. Most of the algorithmic components needed to make deep neural networks work had already been in place for a few decades: backpropagation was invented in 1963,[43] reinvented in 1976,[29] and then again in 1988,[39] and was paired with deep convolutional neural networks[15] in 1989.[27] However, it was only three decades later that deep neural networks were widely accepted as a promising research direction.

The gap between these algorithmic advances and empirical success is due in large part to incompatible hardware. During the general-purpose computing era, hardware such as central processing units (CPUs) was heavily favored and widely available. CPUs are very good at executing an extremely wide variety of tasks; however, processing so many different tasks can incur inefficiency. CPUs require caching intermediate results and are limited in the concurrency of tasks that can be run, which poses limitations for an operation such as matrix multiplication, a core component of deep neural-network architectures. Matrix multiplies are very expensive to run sequentially but far cheaper to compute when parallelized. The inability to parallelize on CPUs meant matrix multiplies quickly exhausted memory bandwidth, and it simply wasn't possible to train deep neural networks with multiple layers.

The need for hardware that supported tasks with lots of parallelism was pointed out as far back as the early 1980s in a series of essays titled, "Parallel Models of Associative Memory."[19] The essays argued persuasively that biological evidence suggested massive parallelism was needed to make deep neural-network approaches work.

In the late 1980s/90s, the idea of specialized hardware for neural networks had passed the novelty stage. However, efforts remained fractured due to a lack of shared software and the cost of hardware development. Without a consumer market, there was simply not the critical mass in end users to be financially viable. It would take a hardware fluke in the early 2000s, a full four decades after the first paper about backpropagation was published, for the insights about massive parallelism to be operationalized in a useful way for connectionist deep neural networks.

A graphical processing unit (GPU) was originally introduced in the 1970s as a specialized accelerator for video games and developing graphics for movies and animation. In the 2000s, GPUs were repurposed for an entirely unimagined use case—to train deep neural networks.[7] GPUs had one critical advantage over CPUs: they were far better at parallelizing a set of simple, decomposable instructions, such as matrix multiples. This higher number of effective floating-point operations per second (FLOPS), combined with clever distribution of training between GPUs, unblocked the training of deeper networks.

The number of layers in a network turned out to be the key. Performance on ImageNet jumped with ever-deeper networks. A striking example of this jump in efficiency is the now-famous 2012 Google research that required 16,000 CPU cores to classify cats; just a year later, a published paper reported the same task had been accomplished using only two CPU cores and four GPUs.[8]

Software Lottery

Software also plays a role in deciding which research ideas win and which ones lose. Prolog and LISP were two languages heavily favored by the AI community until the mid-90s. For most of this period, AI students were expected to actively master at least one, if not both. LISP and Prolog were particularly well suited to handling logic expressions, which were a core component of reasoning and expert systems.

For researchers who wanted to work on connectionist ideas, such as deep neural networks, no clearly suited language of choice existed until the emergence of MATLAB in 1992. Implementing connectionist networks in LISP or Prolog was cumbersome, and most researchers worked in low-level languages such as C++. It was only in the 2000s that a healthier ecosystem began to take root around software developed for deep neural-network approaches, with the emergence of LUSH and, subsequently, TORCH.

Where there is a loser, there is also a winner. From the 1960s through the mid-1980s, most mainstream research focused on symbolic approaches to AI. Unlike deep neural networks, where learning an adequate representation is delegated to the model itself, symbolic approaches aimed to build up a knowledge base and use decision rules to replicate the ways in which humans would approach a problem. This was often codified as a sequence of logic 'what-if' statements that were well suited to LISP and PROLOG.

Symbolic approaches to AI have yet to bear fruit, but the widespread and sustained popularity of this research direction for most of the second half of the 20th century cannot be seen as independent of how readily it fit into existing programming and hardware frameworks.

Machine-learning researchers mostly ignore hardware despite the role it plays in determining which ideas succeed.

The Persistence of the Hardware Lottery

Today, there is renewed interest in collaboration between the hardware, software, and machine-learning communities. We are experiencing a second pendulum swing back to specialized hardware. Catalysts include changing hardware economics, prompted by both the end of Moore's law and the breakdown of Dennard scaling; a "bigger is better" race in the number of model parameters;[1] spiraling energy costs;[20] and the dizzying requirements of deploying machine learning to edge devices.[50]

The end of Moore's law means we are not guaranteed more computing power and performance; hardware will have to earn it. To improve efficiency, there is a shift from task-agnostic hardware, such as CPUs, to domain-specialized hardware that tailors the design to make certain tasks more efficient. The first examples of domain-specialized hardware released over the last few years—tensor processing units (TPUs),[23] edge-TPUs,[16] and Arm Cortex-M55[2]—optimize explicitly for costly operations common to deep neural networks, such as matrix multiplies.

In many ways, hardware is catching up to the present state of machine-learning research. Hardware is only economically viable if the lifetime of the use case is longer than three years.[11] Betting on ideas that have longevity is a key consideration for hardware developers. Thus, co-design efforts have focused almost entirely on optimizing an older generation of models with known commercial use cases. For example, 'matrix multiplies' are a safe target to optimize because they are here to stay—anchored by the widespread use and adoption of deep neural networks in production systems. Allowing for unstructured sparsity and weight-specific quantization is also a safe strategy because there is wide consensus that these will enable higher compression levels.

There is still the separate question of whether hardware innovation is versatile enough to unlock or keep pace with entirely new machine-learning research directions. This question is difficult to answer; the data points here are limited, making it hard to model whether this idea would succeed given different hardware. However, despite this task's inherent challenge, there is already compelling evidence that domain-specialized hardware makes it more costly for research ideas that stray outside of the mainstream to succeed.

The authors of a 2019 published paper titled, "Machine Learning Is Stuck in a Rut"[3] consider the difficulty of training a new type of computer vision architecture called capsule networks[42] on domain-specialized hardware. Capsule networks include novel components, such as squashing operations and routing by agreement. These architecture choices aim to solve for key deficiencies in convolutional neural networks (lack of rotational invariance and spatial hierarchy understanding) but stray from the typical architecture of neural networks. As a result, while capsule network operations can be implemented reasonably well on CPUs, performance falls off a cliff on accelerators such as GPUs and TPUs, which have been overly optimized for matrix multiplies.

Whether or not you agree that capsule networks are the future of computer vision, the authors say something interesting about the difficulty of trying to train a new type of image-classification architecture on domain-specialized hardware. Hardware design has prioritized delivering on commercial use cases, while built-in flexibility to accommodate the next generation of research ideas remains a distant secondary consideration.

While hardware specialization makes deep neural networks more efficient, it also makes it far more costly to stray from accepted building blocks. It prompts the questions: how much will researchers implicitly be overfit to ideas that operationalize well on available hardware, rather than take a risk on ideas that are not currently feasible? What are the failures we still don't have the hardware to see as a success?

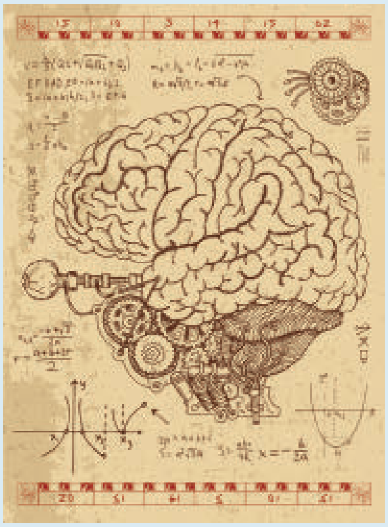
The Likelihood of Future Hardware Lotteries

It is an ongoing, open debate within the machine-learning community about how much future algorithms will differ from models such as deep neural networks. The risk you attach to depending on domain-specialized hardware is tied to your position on this debate. Betting heavily on specialized hardware makes sense if you think that future breakthroughs depend on pairing deep neural networks with ever-increasing amounts of data and computation.

Several major research labs are making this bet, engaging in a "bigger is better" race in the number of model parameters and collecting ever-more-expansive datasets. However, it is unclear whether this is sustainable. An algorithm's scalability is often thought of as the performance gradient relative to the available resources. Given more resources, how does performance increase?

For many subfields, we are now in a regime where the rate of return for additional parameters is decreasing.[46] The cost of throwing additional parameters at a problem is becoming painfully obvious. Perhaps more troubling is how far away we are from the type of intelligence humans demonstrate. Despite their complexity, human brains remain extremely energy efficient. While deep neural networks may be scalable, it may be prohibitively expensive to do so in a regime of comparable intelligence to humans. An apt metaphor is that we appear to be trying to build a ladder to the moon.

Figure 2. Our own cognitive intelligence is inextricably both hardware and algorithm. We do not inhabit multiple brains over our lifetime.



Biological examples of intelligence differ from deep neural networks in enough ways to suggest it is a risky bet to say that deep neural networks are the only way forward. While general-purpose algorithms such as deep neural networks rely on global updates to learn a useful representation, our brains do not. Our own intelligence relies on decentralized local updates which surface a global signal in ways that are not well understood.[5]

In addition, our brains can learn efficient representations from far fewer labeled examples than deep neural networks (Figure 2). Humans have highly optimized and specific pathways developed in our biological hardware for different tasks.[49] This suggests that the way a network is organized, and our inductive biases, are as important as the network's overall size.[18]

For example, it is easy for a human to walk and talk at the same time. However, it is far more cognitively taxing to attempt to read and talk.[44] Our brains are able to fine-tune and retain human skills across our lifetimes.[4] In contrast, deep neural networks that are trained upon new data often evidence catastrophic forgetting, where performance deteriorates on the original task because the new information interferes with previously learned behavior.[30]

There are several highly inefficient assumptions about how we train models. For example, during typical training, the entire model is activated for every example, leading to a quadratic explosion in training costs. In contrast, the brain does not perform a full forward and backward pass for all inputs; it simulates what inputs are expected against incoming sensory data. What we see is largely virtual reality computed from memory.[6]

The point of these examples is not to convince you that deep neural networks are not the way forward, but rather that there are clearly other models of intelligence, which suggests it may not be the only way. It is possible that the next breakthrough will require a fundamentally different way of modeling the world with a different combination of hardware, software, and algorithm. We may very well be amid a present-day hardware lottery.

The Way Forward

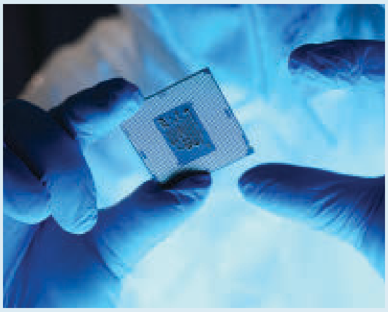
Scientific progress occurs when there is a confluence of factors that allows scientists to overcome the "stickiness" of the existing paradigm. The speed at which paradigm shifts have happened in AI research have been disproportionately determined by the degree of alignment between hardware, software, and algorithm. Thus, any attempt to avoid hardware lotteries must be concerned with making the exploration of different hardware/software/algorithm combinations cheaper and less time-consuming.

This is easier said than done. Expanding the search space of possible hardware/software/algorithm combinations is a formidable goal. It is expensive to explore new types of hardware, both in terms of time and capital required. Producing a next-generation chip typically costs \$30–\$80 million and takes two to three years to develop.[14] The fixed costs alone of building a manufacturing plant are enormous, estimated at \$7 billion in 2017.[45]

Experiments using reinforcement learning to optimize chip placement (Figure 3) may help decrease costs.[31] There is also renewed interest in reconfigurable hardware, such as field-programmable gate array (FPGAs)[17] and coarse-grained reconfigurable arrays (CGRAs).[37] These devices allow chip logic to be reconfigured to avoid being locked into a single use case. However, the tradeoff for flexibility is much higher FLOPS and the need for tailored software development. Coding even simple algorithms on FPGAs remains very painful and time-consuming.[41]

Hardware development in the short- to medium-term is likely to remain expensive and prolonged. The cost of producing hardware is important because it determines the amount of risk and experimentation hardware developers are willing to tolerate. Investment in hardware tailored to deep neural networks is assured because neural networks are a cornerstone of enough commercial use cases. The widespread profitability of downstream uses of deep learning has spurred a healthy ecosystem of hardware startups aiming to further accelerate deep neural networks and has encouraged large companies to develop custom hardware in-house.

Figure 3. Hardware design remains risk averse due to the large amount of capital and time required to fabricate each new generation of hardware.



The bottleneck will continue to be funding hardware for use cases that are not immediately commercially viable. These riskier directions include biological hardware, analog hardware with in-memory computation, neuromorphic computing, optical computing, and quantum computing-based approaches. There are also high-risk efforts to explore the development of transistors using new materials.

An interim goal is to provide better feedback loops to researchers about how our algorithms interact with the hardware we do have. Machine-learning researchers do not spend much time talking about how hardware chooses which ideas succeed and which fail. This is primarily because it is hard to quantify the cost of being concerned. At present, there are no easy-and-cheap-to-use interfaces to benchmark algorithm performance against multiple types of hardware at once. There are frustrating differences in the subset of software operations supported on different types of hardware, which prevents the portability of algorithms across hardware types.[21] Software kernels are often overly optimized for a specific type of hardware, leading to huge lags in efficiency when used with different hardware.

These challenges are compounded by an ever-more-formidable and heterogeneous landscape of hardware.[38] As the hardware landscape becomes increasingly fragmented and specialized, writing fast and efficient code will require more niche and specialized skills.[28] This means that there will be increasingly uneven gains from progress in computer science research. While some types of hardware will benefit from a healthy software ecosystem, progress on other languages will be sporadic and often stymied by a lack of critical end users.[45]

One way to mitigate this need for specialized software expertise is through the development of domain-specific languages that focus on a narrow domain. While you give up expressive power, domain-specific languages permit greater portability across different types of hardware. They allow developers to focus on the intent of the code without worrying about implementation details.[35] Another promising direction is automatically autotuning the algorithmic parameters of a program based upon the downstream choice of hardware. This facilitates easier deployment by tailoring the program to achieve good performance and load balancing on a variety of hardware.[13]

In parallel, we need better profiling tools to empower researchers with a more-informed opinion about how hardware and software should evolve. Ideally, software should surface recommendations about what type of hardware to use given the configuration of an algorithm. Registering what differs from our expectations remains a key catalyst in driving new scientific discoveries. Software needs to do more work, but it is also well positioned to do so. We have neglected efficient software throughout the era of Moore's Law, trusting that predictable gains in computing performance would compensate for inefficiencies in the software stack. This means there are many low-hanging fruit as we begin to optimize for more efficient software.[26]

Conclusion

George Gilder, an American investor, powerfully described the computer chip as inscribing worlds on grains of sand. The performance of an algorithm is fundamentally intertwined with the hardware and software it runs on. This article proposes the term 'hardware lottery' to describe how these downstream choices determine whether a research idea succeeds or fails.

Today the hardware landscape is increasingly heterogeneous. This article posits that the hardware lottery has not gone away, and the gap between the winners and losers will grow. To avoid future hardware lotteries, we need to make it easier to quantify the opportunity cost of settling for the hardware and software we have.

Acknowledgments

Thank you to many of my wonderful colleagues and peers who took time to provide valuable feedback on earlier versions of this contributed article. In particular, I would like to acknowledge the valuable input of Utku Evci, Erich Elsen, Melissa Fabros, Amanda Su, Simon Kornblith, Aaron Courville, Hugo Larochelle, Cliff Young, Eric Jang, Sean McPherson, Jonathan Frankle, Carles Gelada, David Ha, Brian Spiering, Stephanie Sher, Jonathan Binas, Pete Warden, Sean Mcpherson, Lara Florescu, Jacques Pienaar, Chip Huyen, Raziel Alvarez, Dan Hurt, and Kevin Swersky. Thanks for the institutional support and encouragement of Natacha Mainville and Alexander Popper.

References

1. Amodei, D., Hernandez, D., Sastry, G., Clark, J., Brockman, G., and Sutskever, I. AI and compute. *OpenAI* (2018), https://openai.com/blog/ai-and-compute/.
2. ARM. Enhancing AI performance for IoT endpoint devices. (2020), https://www.arm.com/company/news/2020/02/new-ai-technology-from-arm
3. Barham, P. and Isard, M. Machine learning systems are stuck in a rut. In *Proceedings of the Workshop on Hot Topics in Operating Systems (HotOS '19)*, (Bertinoro, Italy), ACM, New York, NY, USA, 177–183. https://doi.org/10.1145/3317550.3321441
4. Barnett, S. and Ceci, S. When and where do we apply what we learn? A taxonomy for far transfer. *Psychological Bulletin 128*, 4 (2002), 612–37.
5. Bi, G. and Poo, M. Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type. *Journal of Neuroscience 18*, 24 (1998), 10464–10472. https://doi.org/10.1523/JNEUROSCI.18-24-10464.1998 arXiv:https://www.jneurosci.org/content/18/24/10464.full.pdf
6. Bubic, A., Cramon, D., and Schubotz, R. Prediction, cognition, and the brain. *Frontiers in Human Neuroscience 4* (2010), 25. https://doi.org/10.3389/fnhum.2010.00025
7. Chellapilla, K., Puri, S., and Simard, P. High performance convolutional neural networks for document processing. *Tenth International Workshop on Frontiers in Handwriting Recognition* (2006).
8. Coates, A., Huval, B., Wang, T., Wu, D., Catanzaro, B., and Andrew, N. Deep learning with COTS HPC systems. In *Proceedings of the 30th Intern. Conf. on Machine Learning* (2013), Sanjoy Dasgupta and David McAllester (Eds.). PMLR, Atlanta, GA, USA, 1337–1345. http://proceedings.mlr.press/v28/coates13.html
9. Collier, B. Little engines that could've: The calculating machines of Charles Babbage. *Garland Publishing, Inc.* (1991) USA.
10. Computer history 1949-1960: Early vacuum tube computers overview. *Computer History Archives Project* (2018), https://www.youtube.com/watch?v=WnNm_uJYWhA
11. Dean, J. The deep learning revolution and its implications for computer architecture and chip design. *IEEE International Solid-State Circuits Conference* (2020), 8–14.
12. Dennard, R., Gaensslen, F., Yu, H., Rideout, V., Bassous, E., and LeBlanc, A. Design of ion implanted MOSFET's with very small physical dimensions. *IEEE Journal of Solid-State Circuits 9*, 5 (1974), 256–268.
13. Dongarra, J., Gates, M., Kurzak, J., Luszczek, P., and Tsai, Y. Autotuning numerical dense linear algebra for batched computation with GPU hardware accelerators. In *Proceedings of the IEEE 106*, 11 (2018), 2040–2055.
14. Feldman, M. The era of general-purpose computers is ending. *The Next Platform* (2019), https://bit.ly/3hP8XJh
15. Fukushima, K. and Miyake, S. Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in position. *Pattern Recognition 15*, 6 (1982), 455–469. http://www.sciencedirect.com/science/article/pii/0031320382900243
16. Gupta, S. and Tan, M. EfficientNet-EdgeTPU: Creating accelerator-optimized neural networks with AutoML. *Google AI Blog* (2019), https://ai.googleblog.com/2019/08/efficientnet-edgetpu-creating.html
17. Hauck, S. and DeHon, A. *Reconfigurable Computing: The Theory and Practice of FPGA-Based Computation*. (2017), Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
18. Herculano-Houzel, S., et al. The elephant brain in numbers. *Frontiers in Neuroanatomy 8* (2014).
19. Hinton, G. and Anderson, J. *Parallel Models of Associative Memory*. (1989), L. Erlbaum Associates Inc., USA.
20. Horowitz, M. Computing's energy problem (and what we can do about it). In *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*. 10–14.
21. Hotel, H., Johansen, H., Bernholdt, D., Héroux, M., and Hornung, R. Software productivity for extreme-scale science. *U.S. Department of Energy Advanced Scientific Computing Research* (2014).
22. Isaacson, W. Grace Hopper, computing pioneer. *The Harvard Gazette* (2014). https://news.harvard.edu/gazette/story/2014/12/grace-hopper-computing-pioneer/
23. Jouppi, N., et al. In-datacenter performance analysis of a tensor processing unit. *SIGARCH Comput. Archit. News 45*, 2 (June 2017), 1–12. https://doi.org/10.1145/3140659.3080246
24. Kuhn, T. *The Structure of Scientific Revolutions*. (1962), University of Chicago Press, Chicago.
25. Kurzweil, R. *The Age of Intelligent Machines*. (1990), MIT Press, Cambridge, MA, USA.
26. Larus, J. Spending Moore's dividend. *Commun. ACM 52*, 5 (May 2009), 62–69. https://doi.org/10.1145/1506409.1506425
27. LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., and Jackel, L. Backpropagation applied to handwritten zip code recognition. *Neural Computation 1*, 4 (1989), 541–551. https://doi.org/10.1162/neco.1989.1.4.541
28. Lee, H., Brown, K., Sujeeth, A., Chafi, H., Rompf, T., Odersky, M., and Olukotun, K. Implementing domain-specific languages for heterogeneous parallel computing. *IEEE Micro 31*, 5 (2011), 42–53.
29. Linnainmaa, S. Taylor expansion of the accumulated rounding error. *BIT Numerical Mathematics 16* (1976), 146–160.
30. McClelland, J., McNaughton, B., and O'Reilly, R. Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review 102* (Aug. 1995), 419–57. https://doi.org/10.1037/0033-295X.102.3.419
31. Mirhoseini, A., et al. A graph placement methodology for fast chip design. *Nature 594* (June 9, 2021), 207-212, https://www.nature.com/articles/s41586-021-03544-w.
32. Moore, D. The Anna Karenina Principle applied to ecological risk assessments of multiple stressors. *Human and Ecological Risk Assessment: An International Journal 7*, 2 (2001), 231–237. https://doi.org/10.1080/20018091094349
33. Moore, G. 1965. Cramming more components onto integrated circuits. *Electronics 3*8, 8 (April 1965). https://www.cs.utexas.edu/~fussell/courses/cs352h/papers/moore.pdf
34. Moravec, H. When will computer hardware match the human brain. *Journal of Transhumanism 1* (1998).
35. Olukotun, K. Beyond parallel programming with domain specific languages. *SIGPLAN Not. 49*, 8 (Feb. 2014), 179–180. https://doi.org/10.1145/2692916.2557966
36. Posselt, E.A. *The Jacquard Machine Analyzed and Explained: The Preparation of Jacquard Cards and Practical Hints to Learners of Jacquard Designing*. (1888).
37. Prabhakar, R., et al. Plasticine: A reconfigurable architecture for parallel patterns. In *2017 ACM/IEEE 44th Annual Intern. Symposium on Computer Architecture (ISCA)*. 389–402.
38. Reddi, V., et al. MLPerf inference benchmark. In *2020 ACM/IEEE 47th Annual Intern. Symposium on Computer Architecture* (2020), 446–459.
39. Rumelhart, D., Hinton, G., and Williams, R. Learning representations by back-propagating errors. *MIT Press* (1988), 696–699.
40. Sabour, S., Frost, N., and Hinton, G. Dynamic routing between capsules. (2017), 3856–3866. http://papers.nips.cc/paper/6975-dynamic-routing-between-capsules.pdf
41. Shalf, J. The future of computing beyond Moore's law. *Philosophical Transactions of the Royal Society A, 378* (2020).
42. Singh, V., Perdigones, A., Garcia, J., Cañas, I., and Mazarrón, F. Analyzing worldwide research in hardware architecture, 1997-2011. *Commun. ACM 58* (January 2015), 76–85. https://doi.org/10.1145/2688498.2688499.
43. Steinbuch, K. and Piske, U.A.W. Learning matrices and their applications. *IEEE Transactions on Electronic Computers EC-12*, 6 (1963), 846–862.
44. Stroop, J. Studies of interference in serial verbal reactions. *J. of Experimental Psychology 18*, 6 (1935), 643. https://doi.org/10.1037/h0054651
45. Thompson, N. and Spanuth, S. The decline of computers as a general purpose technology: Why deep learning and the end of Moore's Law are fragmenting computing. (November 2018).
46. Thompson, N., Greenewald, K., Lee, K., and Manso, G. The computational limits of deep learning. *arXiv e-prints*, Article arXiv:2007.05558 (July 2020), arXiv:2007.05558 pages. arXiv:2007.05558 [cs.LG]
47. Tolstoy, L. and Bartlett, R. *Anna Karenina*. Oxford University Press (2016), https://books.google.com/books?id=1DooDwAAQBAJ
48. Van Der Malsburg, C. Frank Rosenblatt: Principles of neurodynamics: Perceptrons and the theory of brain mechanisms. *Brain Theory* (1986), 245–248.
49. Von Neumann, J., Churchland, P.M., and Churchland, P.S. *The Computer and the Brain*. Yale University Press (2000), https://books.google.com/books?id=Q30MqJjRv1gC
50. Warden, P. and Situnayake, D. *TinyML: Machine Learning With TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers*. (2019), O'Reilly Media, Inc. https://books.google.com/books?id=sB3mxQEACAAJ

Sara Hooker (shooker@google.com) is a research scholar at Google Brain, Mountain View, CA, USA, with a focus on deep learning.

This work is licensed under a Creative Commons Attribution-NonCommercial International 4.0 License.

Watch the author discuss this work in the exclusive *Communications* video. https://cacm.acm.org/videos/the-hardware-lottery contributed articles

DOI:10.1145/3450409

3D heart modeling and AI bring new cardiac surgery to remote and less-developed regions.

BY XIAOWEI XU, HAILONG QIU, QIANJUN JIA, YUHAO DONG, ZEYANG YAO, WEN XIE, HUIMING GUO, HAIYUN YUAN, JIAN ZHUANG, MEIPING HUANG, AND YIYU SHI

AI-CHD: An AI-Based Framework for Cost-Effective Surgical Telementoring of Congenital Heart Disease

CONGENITAL HEART DISEASE (CHD), the most common congenital birth defect, has long been known as one of the main causes of infant death during the first year of life.[1] More than one million of the world's approximately 135 million newborns are born each year with CHD.[21] Over the last century, cardiac surgery has been an effective approach to tackling CHD; its remarkable advance has decreased the mortality rate of newborns with CHD.[10]

However, that lower mortality rate is mostly observed in developed countries rather than developing ones. Surgical treatment of CHD requires highly skilled surgeons along with complex infrastructures and equipment. While developed countries have perfected their treatment of CHD for more than 50 years, developing countries are still in the early stages. It is estimated that the number of congenital cardiac surgeons needs to increase by 1,250 times to satisfy only the basic needs of CHD treatment worldwide,[16] and most of those surgeons reside in developed countries. As a result, the mortality rate in developing countries is currently at 20%, strikingly higher than the 3% to 7% in developed countries,[16] not to mention the fact that mortality rates in developing countries are likely underreported due to the lack of proper diagnosis.

Remote Surgery

Remote surgery has been an active field for decades, enabling experienced surgeons to remotely instruct robots (telerobotics) or guide less-experienced surgeons (surgical telementoring).[8] It enables high-quality surgical expertise to be passed from surgeons in developed countries to those in developing ones, or from high-end urban medical centers to rural hospitals inside developing nations.

» key insights

- **Congenital heart disease (CHD), the most common congenital birth defect, is usually treated with heart surgery. However, such procedures require highly skilled surgeons, who are especially scarce in remote and less-developed regions.**
- **Surgical telementoring enables an expert surgeon to remotely guide a less-experienced surgeon. The main challenge is high costs brought on by inefficient planning based on low-quality images.**
- **Using AI to automatically construct accurate 3D models of the heart from low-quality images can help during remote surgical planning, via 3D printing or virtual reality (VR) technology, and can improve operational efficiency during surgery.**

IMAGE COLLAGE BY ANDRIJ BORYS ASSOCIATES, USING SHUTTERSTOCK BY SERGEY NIVENS, AND MARKO ALIAKSANDR

Telerobotics enables surgeons to remotely control robots in a master-slave relationship. Stable camera systems are implemented in both sites. At the robot site, multiple cameras construct a virtual image of the operative field, which is provided to the surgeon site. At the surgeon site, multiple cameras and 3D imaging systems capture the surgeon's hand movements, which are sent to the robot site and emulated by the robot operating on the patient. Images and movements are required to be transmitted in real time, but large latency may affect the surgeon's performance or even lead to surgical failure. Telerobotics can also let surgeons sit comfortably while performing some delicate operations requiring fine movements. However, due to the highly demanding nature and potential risks, telerobotics has very limited clinical application. Only a few systems, including da Vinci[14] and Zeus,[13] are approved for use. At present, telerobotics is still in its early stages.

Surgical telementoring, on the other hand, consists of an expert surgeon remotely guiding a less-experienced surgeon. Such guidance is achieved with real-time audio and video transmission. Thus, the two surgeons can discuss the procedure in real time, and the expert surgeon can deliver precise guidance based on the real-time video streaming of the surgery process. Surgical telementoring can be performed in rural areas and austere environments, not only for difficult surgeries but also for surgery education. Like telerobotics, surgical telementoring requires real-time data transmission. Fortunately, 4G and 5G communication technologies have made this possible across great distances. Surgical telementoring has been widely adopted and explored in clinical use thanks to improved transmission quality, and the technology carries less potential risk compared with telerobotics.[11]

Though also in its early stage, surgical telementoring is still more mature than telerobotics, with its relatively lower cost and lower technical complexity. However, surgical telementoring for CHD still faces challenges. First, cardiac surgery for the treatment of CHD is rather complex, generally regarded as the jewel in the crown of surgery. As such, CHD diagnosis and surgical planning are usually time-consuming and costly; delivering this expertise to developing countries or rural areas can often be time- and cost-prohibitive. For example, examining the medical image of a CHD patient for diagnosis takes even a very experienced radiologist several hours, whereas that time is usually on the order of minutes for common heart diseases. Second, the machine quality and operator skill in developing countries or rural areas may be limited, leading to issues such as low imaging quality under non-ideal settings.

Figure 1. Surgical telementoring: Technical assistance and guidance via real-time video and audio streams.

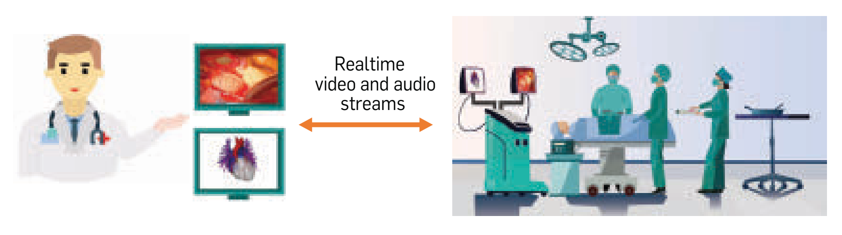
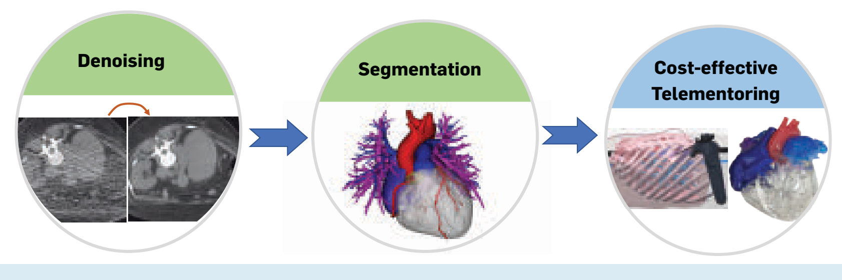


Figure 2. Overall flow of AI-CHD.



AI-CHD

One potential solution to reduce costs is to use artificial intelligence (AI) to automatically construct accurate 3D models of the heart from medical images, a critical yet otherwise time-consuming process in CHD surgical telementoring. Before surgery, this model can help during remote surgical planning and discussions via 3D printing or virtual reality (VR) technologies. During surgery, viewing the model via a 2D screen can enhance operational efficiency by fostering communication between the surgeon and the novice.

Our novel solution, an AI-based framework called AI-CHD, aims to construct accurate and efficient heart models for surgical telementoring of CHD based on 3D computed tomography (CT) images. Considering that the artifact type and pattern in CT images acquired with medium- and low-end machines or by users with limited skills may be different from those in standard training sets, the framework first exploits a weakly supervised way to remove artifacts in a CT image, which does not require a prior training set. Further, considering that hearts in CHD exhibit large variations in structure and/or great vessel connections without local tissue changes, the framework then deploys hybrid deep-learning and graph analysis to tackle the model construction. We evaluate each step with collected datasets and the overall system with a case study.

Medical image artifact reduction. Medical images exhibit various types of artifacts, with different patterns and mixtures that depend on many factors, including scan setting, machine condition, patient size and age, surrounding environment, etc. This problem is even worse on middle-of-the-road and low-end CT imaging machines—often operated by less-skilled technologists—which are common in rural areas in developing countries. On the other hand, existing deep learning-based artifact-reduction methods for medical images are restricted by the specific training data that contains predetermined artifact types and patterns, which can hardly capture all possibilities exclusively. Accordingly, they can only work well under the scenarios defined by the training data. In this step, we exploit the power of deep learning but without using pre-trained networks for medical artifact reduction. Specifically, at test time we train a lightweight, image-specific, artifact-reduction network using data synthesized from the input image. Without requiring any prior training data, our method can work with almost any medical images that contain varying or unknown artifacts.

The main flow of artifact reduction contains two modules: artifact synthesis and artifact removal.[3] In the first module, radiologists need to annotate a total of 10–20 regions of interest (RoI) from a 3D-input CT image. These RoI are further used to train a lightweight, five-layer synthesis network, which synthesizes a large number of paired patches. Note that as different medical images have different ranges of pixel values, we normalize pixel values so that each has a value between 0 and 1. With synthesized paired patches, theoretically any existing CNN-based artifact-reduction networks can be trained. However, a key issue here is that we perform the task on each input image. Deep and complex networks may need many data pairs and long training times. Smaller networks, on the other hand, may not attain desired performance. Thus, in the second module, we resort to an *artifact-removal network*—a compact, attentive generative network architecture that can pay special attention to artifacts and train them adversarially for faster convergence. It is formed via a two-step attentive-recurrent network followed by a 10-layer contextual autoencoder to reduce artifacts and restore the information obstructed by them. Once trained, the artifact-removal network is applied to all slices of the 3D volume for artifact reduction.

Fundamental to our method is the fact that artifacts in most medical images exhibit localized patterns—that is, they do not cover the entire image uniformly. It is almost always possible to identify "clean" regions (with few artifacts) and "dirty" regions (with significant artifacts) within an image. This makes it possibile to synthesize paired, dirty-clean training patches from an image with artifacts. In addition, as visual entropy inside a single image is much smaller than in a general external collection of images,[30] the synthesized training data does not need to be big, and the associated artifact-reduction network can be compact and converge quickly.

AI-CHD is an accurate, AI-based framework for surgical telementoring of CHD developed through deep collaborations between computer scientists, radiologists, and surgeons.

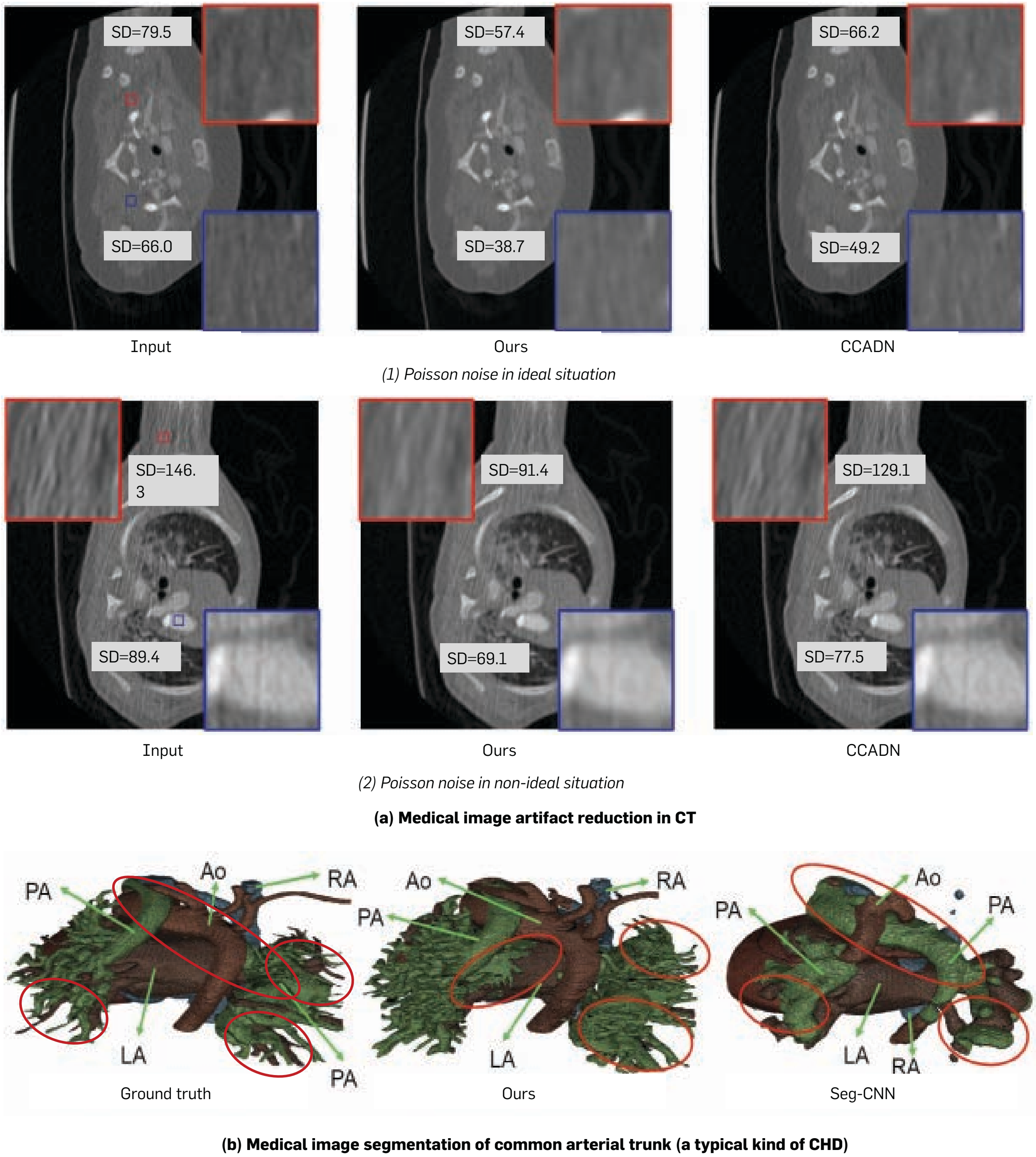
We evaluate the performance of this step with CT images containing different levels of Poisson noise collected by our wide-detector, 256-slice MDCT scanner with 8 cm of coverage, using the following protocol: collimation, (96–128)×0.625 mm; rotation time, 270 ms, which corresponds to a 135-ms standard temporal resolution; slice thickness, 0.9 mm; and reconstruction interval, 0.45 mm. Adaptive-axial z-collimation was used to optimize the cranio-caudal length. Data was obtained at 40%–50% of the RR interval, using a 5% phase tolerance around the 45% phase. All CT images are qualitatively evaluated by our radiologists on structure-preservation and artifact levels. For quantitative evaluation, due to the lack of ground truth, we follow most existing works[24,28] and select the most homogeneous area in regions of interest chosen by radiologists. Standard deviation (artifact level) of the pixels in the area should be as low as possible.

Our method trains and tests on each individual patient's image, and no pre-training is involved. We compare our method with state-of-the-art, deep learning-based, medical-image artifact-reduction methods on a cycle-consistent adversarial denoising network (CCADN),[9] which is trained following the exact same reported settings. The CT training data set for the CCADN contains 100,000 image patches. We consider both the ideal situation, where test images only contain Poisson noise levels such as those in the training set, and non-ideal situations, where test images have higher noise levels. Figure 3(a) shows the results. Qualitatively, our method and CCADN preserve structures well for both ideal and non-ideal situations. Our method outperforms CCADN even at noise levels that the CCADN is trained to reduce—that is, the regions in Figure 3(a)(1). Quantitatively, our method beats CCADN in both ideal and non-ideal situations, achieving up to 29.2% lower standard deviation and 18.6% on average.

Though our method is trained and tested on each test image, it has almost the same execution time compared with CCADN, due to the significantly reduced network complexity and faster convergence brought by the internal visual entropy.

Medical image segmentation. CHD usually comes with significant variations in heart structures and great-vessel connections, which renders general whole-heart and great-vessel segmentation methods[18,22] in normal anatomy ineffective. Most existing segmentation methods are only dedicated to CHD target blood pool and myocardium.[23,29] Recently, semi-automated segmentation in CHD has also been explored,[17] requiring users to locate an initial seed. However, fully automated segmentation of whole-heart and great-vessel segmentation in CHD remains a missing piece in the literature. Inspired by the success of graph matching in several applications with large variations,[12] we propose to combine deep learning[25] and graph matching for fully automated whole-heart and great-vessel segmentation in

Figure 3. Step-by-step performance of AI-CHD.



(a) Medical image artifact reduction in CT

(b) Medical image segmentation of common arterial trunk (a typical kind of CHD)

Mean and standard deviation (SD) of Dice score of the state-of-the-art method, Seg-CNN,[18] and our method (in %) for seven substructures of whole-heart and great-vessel segmentation.

Method		LV	RV	LA	RA	Myo	Ao	PA	Average
Seg-CNN	Mean	67.3	65.0	70.2	76.0	71.5	63.0	52.3	66.5
[9]	SD	±13.9	±12.0	±7.8	±7.5	±8.3	±13.3	±12.3	±10.7
Our	Mean	82.4	77.6	78.6	82.7	77.3	82.2	67.1	78.3
method	SD	±10.5	±14.3	±7.4	±7.5	±8.3	±8.1	±19.8	±10.8

CHD.[26] Particularly, we leverage deep learning to segment the four chambers and myocardium followed by blood pool, where variations are usually small and accuracy can be high. We then extract the vessel connection information and apply graph matching to determine the categories of all the vessels.

The overall flow for whole-heart and great-vessel segmentation—left ventricle (LV), right ventricle (RV), left atrium (LA), right atrium (RA), myocardium (Myo), aorta (Ao), and pulmonary artery (PA)—contains two modules: whole-heart segmentation and great-vessel segmentation. In whole-heart segmentation, RoI cropping is first presented to extract the area that includes the heart and its surrounding vessels. We resize the input image to a low resolution of 64×64×64 and then adopt the same segmentation-based extraction as in Payer et al.[18] to get the RoI. The RoI are then resized to 64×64×64 and fed into a 3D U-net[5] for segmentation.

In great-vessel segmentation, blood-pool segmentation is conducted on each 2D slice of the input using a 2D U-net[19] with an input size of 512×512. Note that to detect the blood-pool boundary for easy graph extraction in graph matching later, we add another class: blood-pool boundary in the segmentation. With the high-resolution blood segmentation, whole-heart segmentation achieves chamber and myocardium refinement by refining chamber and myocardium boundaries. By removing the blood pool corresponding to the low-resolution, whole-heart segmentation, great-vessel segmentation obtains the blood pool corresponding to the great vessels and adopts graph matching to identify Ao, PA, and anomalous vessels.

For evaluation, we collected a dataset composed of 68 3D CT images captured by a Siemens Biograph 64 machine. The ages of the associated patients range from one month to 21 years, with the majority between one month and two years. The size of the images is 512 × 512×(130–340), and the typical voxel size is 0.25×0.25×0.5 mm^3. The dataset covers 14 types of CHD, including six common types—atrial septal defect (ASD), atrio-ventricular septal defect (AVSD), patent ductus arteriosus (PDA), pulmonary stenosis (PS), ventricular septal defect (VSD), and co-arctation (CA)—plus eight less-common ones—Tetrology of Fallot (ToF), transposition of the great arteries (TGA), pulmonary artery sling (PAS), anomalous drainage (AD), common arterial trunk (CAT), aortic arch anomalies (AAA), single ventricle (SV), and pulmonary atresia (PuA).

All labeling was performed by experienced radiologists, and the labeling time per image was two to three hours. The labels include seven substructures: LV, RV, LA, RA, Myo, Ao, and PA. For easy processing, venae cavae (VC) and pulmonary vein (PV) are also labeled as part of RA and LA respectively, as they are connected, and their boundaries are relatively hard to define. Anomalous vessels are also labeled as one of the above seven substructures based on their connections. The comparison with Seg-CNN[18] is shown in the Table. Our method can achieve a mean Dice score between 5.8% and 19.2% higher across the seven substructures (12% higher on average) with almost the same standard deviation. The highest improvement is achieved in Ao, due to its simple graph connection with successful graph matching. The smallest improvement is obtained in myocardium because it is not well considered in the high-resolution blood-pool segmentation. Figure 3(b) shows visualization of CAT segmentation using our method and Seg-CNN. Our method can clearly segment Ao and PA with some slight mis-segmentation between PA and LA. However, Seg-CNN segments the main part of Ao as PA, since pixel-level segmentation by U-net is based only on the surrounding pixels, and the connection information is not well exploited.

Real-Time Video and Audio Transmission

Real-time video and audio transmission is also a critical part of surgical telementoring. Such transmission needs high data rates and low latency so that important decisions can be made in real time to avoid potential complications. In addition, wireless transmission is always preferred over wired transmission in an operating room. The most common wireless transmission method so far, 4G, offers about 50-ms latency and a 10-Mbps average data rate. Such moderate transmission quality and speeds can support less-complex telemonitoring procedures, such as addiction management[20] and training.[2]

5G wireless communication is emerging to offer about 10-ms latency (1 ms in special cases) and higher data rates of 50+ Mbps on average. Such high transmission quality has enabled many complex procedures, including intestinal tumor procedure,[15] liver removal of a laboratory test animal,[6] laparoscopic cholecystectomy,[27] and gall bladder surgery.[4] However, surgical telementoring has not been reported in cardiac surgery applications for CHD treatment, possibly due to its extremely high complexity and high risk.

AI-CHD Case Study

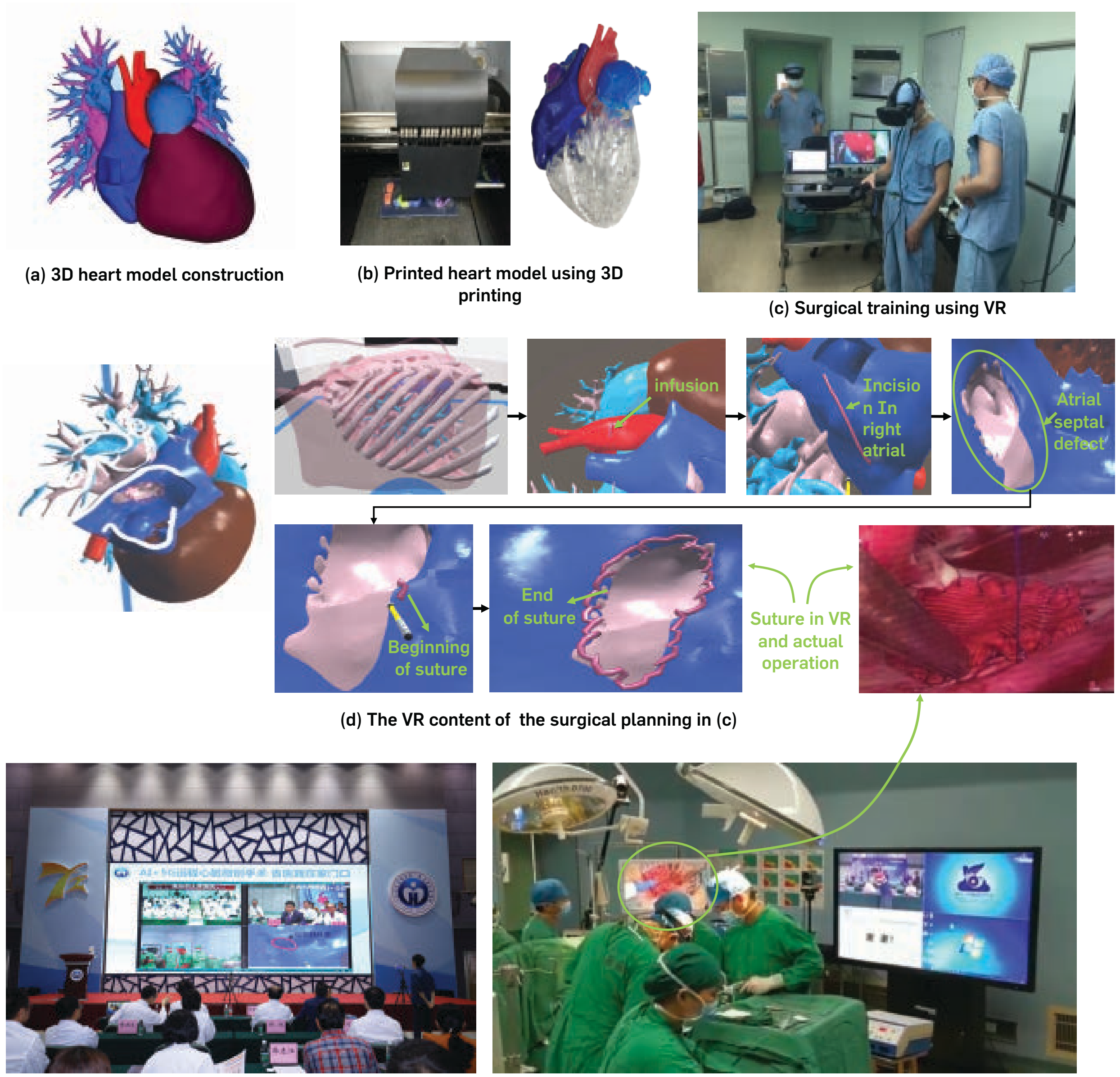
On April 3, 2019, we used AI-CHD to perform China's first 5G-based heart telementoring procedure, with collaboration between Guangdong Provincial People's Hospital (GPPH) and the People's Hospital of Gaozhou (PHG). Guangdong Mobile Communications Group and Huawei provided 5G capabilities. AI-CHD produced an accurate heart model of the patient, which was used for pre-surgical planning and training, and for real-time guidance during the surgery.

The 41-year-old patient, Ms. Green (alias), from Gaozhou, Guangdong Province, was experiencing shortness of breath, chest pain, difficulty walking, and insomnia. She was diagnosed with atrial septal defect (ASD) with tricuspid insufficiency, severe pulmonary hypertension, and cardiac failure at PHG. ASD is a moderate type of CHD that can be treated through a relatively easy, low-risk operation at a young age; however, her condition had not been detected due to lack of screening availability in the rural area where she lived. And, the untreated ASD brought about other conditions, such as severe pulmonary hypertension and heart failure. Thus, her condition changed from a relatively simple CHD to a complex one. Only surgery could save her heart.

Being in a less-developed region, PHG surgeons only have experience in conventional open-heart surgery, a higher-risk procedure, especially for a long-suffering patient such as Ms. Green. A minimally invasive, lower-risk surgery was preferred. However, Ms. Green would have had to travel 250 miles to GPPH, one of China's largest cardiac medical centers, and the near-

Figure 4. AI-CHD case study.

Telementoring for a patient with atrial septal defect is performed between Guangdong Provincial People's Hospital (GPPH) and the People's Hospital of Gaozhou (PHG) on April 3, 2019. The two hospitals are 250 miles away from each other. The 3D heart model obtained from AI-CHD (a) is used for heart-model printing (b) and surgical training (c). In surgical training, surgeons can practice the surgery in VR (d), which is also based on the 3D heart model. Telementoring is successfully performed, and some on-site photos of the event are shown (e).



(a) 3D heart model construction

(b) Printed heart model using 3D printing

(c) Surgical training using VR

(d) The VR content of the surgical planning in (c)

(e) The telementor in Guangdong Provincial People's Hospital (GPPH) (left) and the heart surgery operation in the People's Hospital of Gaozhou (PHG) (right).

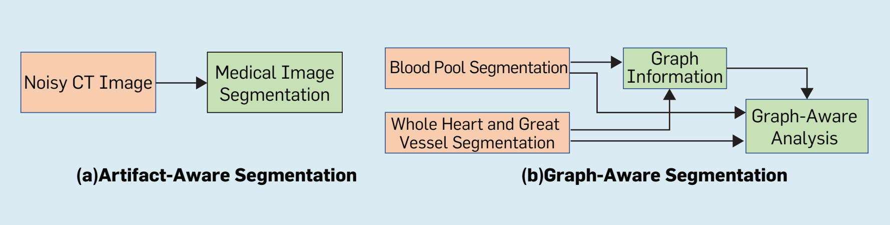
est hospital with surgeons capable of performing such a complex procedure. Considering Ms. Green's weakened condition, the journey was not feasible. Therefore, telementoring was the most suitable approach. With comprehensive discussion and analysis, Dr. Huiming Guo, GPPH's chief physician of cardiac surgery, agreed to serve as telementor in this surgery. Dr. Guo brought much expertise with minimally invasive surgeries to the table and is well known internationally in the surgical treatment of CHD.

Before Ms. Green's surgery, we first collected her 3D cardiac CT images from PHG. AI-CHD then produced her 3D heart model, see Figure 4(a), with a clinically acceptable accuracy of 0.81 (Dice score) for the surgery. Runtime was less than two minutes, much faster than manual segmentation—which could take two to three hours or even longer—thus significantly reducing the cost. With the 3D heart model, surgical planning and training were then performed. As shown in Figure 4(b), the heart model was first printed out with a 3D printer (thin vessels of pulmonary artery and pulmonary vein were removed, as they were not related to the surgery). The printed model provided a straightforward view of the heart's appearance and structure and showed where the problem was. Dr. Guo used the printed model to discuss the surgical plan with other members of his team and the remote team at PHG. Once the plan was set, virtual surgical training was carried out via VR, shown in Figure 4(c).

As seen in Figure 4(d), VR enables doctors to enlarge and shrink specific heart structures, including its inner structures, and to perform virtual operations such as infusion and suture as a practice. In this way, Dr. Guo could get a comprehensive understanding of the surgery to be telemonitored and establish the best operations/parameters for effective guidance.

After Dr. Guo confirmed the surgical plan and the detailed process with the help of AI-CHD, telementoring of Ms. Green's surgery began at 9:35 AM on April 3, 2019. Figure 4(d) shows pictures of the guidance team at GPPH and the actual surgery at PHG. The procedure featured four real-time video streams: the view of the surgeon, the corresponding VR view of the heart, the telementor (Dr. Guo), and the operating room. Since this was China's first CHD telementoring surgery, doctors from the cardiac surgery, cardiac ultrasound, and cardiac imaging departments all showed up to observe the event, as seen in the telementor view.

Figure 5. Future work of AI-CHD on medical image-artifact reduction and segmentation: Artifact-aware segmentation (a) that directly performs segmentation on noisy images rather than artifact reduction first and then segmentation, and graph-aware segmentation (b) that takes graph information among thin vessels into consideration for accurate segmentation.


Noisy CT Image
Medical Image Segmentation
(a)Artifact-Aware Segmentation
Blood Pool Segmentation
Graph Information
Whole Heart and Great Vessel Segmentation
Graph-Aware Analysis
(b)Graph-Aware Segmentation

Based on the real-time view of the surgeon and the corresponding VR scene, Dr. Guo could easily recognize the current view of the heart and offer immediate guidance via a real-time audio stream. For example, when determining the opening point at the pericardium, the surgeon asked, "Should the pericardium be opened here?" Dr. Guo answered instantaneously, "Move up three centimeters." For the suture in the operation, which is also a key part of the surgery, Dr. Guo reminded the surgeon, "Do not be too close to the Koch's triangle when stitching; otherwise, it is easy to cause myocardial injury and block the heart-beating rhythm conduction." Koch's triangle was drawn in VR video and shown to surgeons in the operating room 250 miles away in real time.

Throughout the telemonitoring, the transmission rate for video streams is stabilized at around 25 Mbps with a latency of 30 ms. The surgery went smoothly and finished at 1:00 PM. The heart was sutured, Ms. Green's heart resumed beating, and after a week in recovery, she was discharged. The postoperative review showed that pulmonary artery pressure and mitral regurgitation were within the normal range. The patient remains in good health as of the writing of this article (Dec. 1, 2019). Our case study has received extensive media coverage in some of the biggest and most influential news organizations in China, including *Xinhua*[28] and *Global Times*.[7]

Looking Forward

This is the third year of collaboration between computer scientists, cardiac surgeons, and radiologists in our team. The original work of artifact reduction and segmentation of cardiac CT images has gradually evolved into a holistic system of 3D heart-model construction. In the future, in addition to further optimizing AI-CHD, we plan to explore the following four promising areas for automatic and cost-efficient treatment of CHD:

Artifact-aware segmentation. The current approach involves two steps: artifact reduction and segmentation. However, it may be possible to improve efficiency by performing segmentation directly on noisy images in just one step, as shown in Figure 5(a). We believe this is promising for two reasons: First, artifacts typically display some patterns, making it possible to capture and remove them and segment targets jointly in one neural network. Second, while it may be a problem to obtain the training dataset, we can use our artifact reduction method or other existing methods to get clean images and then perform manual labeling.

Graph-aware segmentation. Our method and other existing methods are still challenged to correctly segment thin vessels, especially thin PA vessels. The main reason is that the rich connection information among these vessels is not well exploited. As shown in Figure 5(b), we may extract the graphs of these thin vessels from blood-pool segmentation results and whole-heart and great-vessel segmentation results to represent connection information. Then, graph-aware analysis that takes both segmentation and connection information into consideration can be performed to obtain more accurate segmentation results.

Automatic diagnosis. Accurate diagnosis of CHD is more significant compared with artifact reduction and segmentation. The lack of CHD diagnosis experience in developing regions means many cases are not diagnosed correctly and miss timely treatment.[16] To gain the expertise to be able to make such a diagnosis, radiologists require more than 10 years of training, which can be time-consuming and costly. Even experienced radiologists may need up to a half-hour to diagnose a patient with CHD. Thus, automatic CHD diagnosis is preferred, for its ability to provide large-scale, high-quality, cost-efficient medical care. To be clinically acceptable, automatic CHD diagnosis also needs to report the features or reasons for the diagnosis with an accompanying confidence score. Radiologists could more easily verify the results; low confidence scores would denote cases in need of manual diagnosis.

Automatic surgery planning. Due to the large structure variations in CHD, dozens of surgical procedures exist, each containing parameters such as opening point, incision size, direction. Currently, surgeons plan based on their experience, which may or may not be the optimal choice in terms of prognosis. We will further extend AI-CHD to enable accurate automatic surgical planning for optimal treatment.

Conclusion

AI-CHD is an accurate, AI-based framework for surgical telementoring of CHD. It is developed through deep collaborations between computer scientists, radiologists, and surgeons. The technology enables cost-effective and timely model construction of hearts in CHD, which assists radiologists and surgeons with performing efficient surgical planning and training in CHD surgery, as demonstrated by the case study. AI-CHD can reduce costs while improving the quality of CHD surgery telementoring in developing countries and regions.

Acknowledgments

This work was approved by the Research Ethics Committee of Guangdong General Hospital, Guangdong Academy of Medical Science under Protocol No. 20140316. It was supported by the National Key Research and Development Program of China (2018YFC1002600), the Science and Technology Planning Project of Guangdong Province, China (No.2017B090904034, 2017B030314109, 2018B090944002, 2019B020230003), Guangdong Peak Project (DFJH201802), and the National Natural Science Foundation of China (No.62006050).

References

1. Bernier, P-L., Stefanescu, A., Samoukovic, G., and Tchervenkov, C.I. The challenge of congenital heart disease worldwide: Epidemiologic and demographic facts. In *Seminars in Thoracic and Cardiovascular Surgery: Pediatric Cardiac Surgery Annual 13*, Elsevier (2010), 26–34.
2. Bogen, E.M., Augestad, K.M., Patel, H.R.H., and Lindsetmo, R-O. Telementoring in education of laparoscopic surgeons: An emerging technology. *World Journal of Gastrointestinal Endoscopy 6*, 5 (2014), 148.
3. Chen, Y-J., Chang, Y-J., Wen, S-C., Shi, Y., Xu, X., Ho, T-Y., Jia, Q., Huang, M., and Zhuang, J. Zero-shot medical image artifact reduction. In *2020 IEEE 17th Intern. Symp. on Biomedical Imaging (ISBI)*, 862–866.
4. Chinese surgeons conduct remote surgery using 5G technology. *The Times of India* (June 11, 2019), https://timesofindia.indiatimes.com/world/china/chinese-surgeons- conduct-remote-surgery-using-5g-technology/articleshow/69742530.cms.
5. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., and Ronneberger, O. 3D U-Net: Learning dense volumetric segmentation from sparse annotation. In *Intern. Conf. on Medical Image Computing and Computer-Assisted Intervention*. Springer (2016), 424–432.
6. Cuthbertson, A. Surgeon performs world's first remote operation using 5G surgery on animal in China. *Independent* (Jan. 17, 2019), https://www.independent.co.uk/life-style/gadgets-and-tech/news/5g-surgery-china-robotic-operation-a8732861.html.
7. First AI+5G surgery completed with Huawei's technological support. *Global Times* (2019), http://www.globaltimes.cn/content/1144600.shtml.
8. Huang, E.Y., Knight, S., Guetter, C.R., Davis, C.H., Moller, M., Slama, E., and Crandall, M. Telemedicine and telementoring in the surgical specialties: A narrative review. *The American Journal of Surgery 218*, 4 (2019), 760–766.
9. Kang, E., Koo, H.J., Yang, D.H., Seo, J.B., and Ye, J.C. Cycle consistent adversarial denoising network for multiphase coronary CT angiography. (2018), arXiv preprint arXiv:1806.09748.
10. Kempny, A., Dimopoulos, K., Uebing, A., Diller, G-P., Rosendahl, U., Belitsis, G., Gatzoulis, M.A., and Wort, S.J. Outcome of cardiac surgery in patients with congenital heart disease in England between 1997 and 2015. *PLoS One 12*, 6 (2017), e0178963.
11. Lacy, A.M., Bravo, R., Otero-Piñeiro, A.M., Pena, R., De Lacy, F.B., Menchaca, R., and Balibrea, J.M. 5G-assisted telementored surgery. *British Journal of Surgery 106*, 12 (2019), 1576–1579.
12. Lajevardi, S.M., Arakala, A., Davis, S.A., and Horadam, K.J. Retina verification system based on biometric graph matching. *IEEE Transactions on Image Processing 22*, 9 (2013), 3625–3635.
13. Marescaux, J. and Rubino, F. The ZEUS robotic system: Experimental and clinical applications. *Surgical Clinics 83*, 6 (2003), 1305–1315.
14. Nifong, L.W., Chu, V.F., Bailey, B.M., Maziarz, D.M., Sorrell, V.L., Holbert, D., and Chitwood, Jr., W.R. Robotic mitral valve repair: Experience with the da Vinci system. *The Annals of Thoracic Surgery 75*, 2 (2003), 438–443.
15. Nita, R. World's first 5G-powered surgery: Dr. Antonio de Lacy. *World Record Academy* (2019), https://www.worldrecordacademy.org/medical/worlds-first-5g-powered-surgery-dr-antonio-de-lacy-219142.
16. Ntiloudi, D., Giannakoulas, G., Parcharidou, D., Panagiotidis, T., Gatzoulis, M.A., and Karvounis, H. Adult congenital heart disease: A paradigm of epidemiological change. *International J. of Cardiology 218* (2016), 269–274.
17. Pace, D.F., Dalca, A.V., Brosch, T., Geva, T., Powell, A.J., Weese, J., Moghari, M.H., and Golland, P. Iterative segmentation from limited training data: Applications to congenital heart disease. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*. Springer (2018), 334–342.
18. Payer, C., Štern, D., Bischof, H., and Urschler, M. Multi-label whole heart segmentation using CNNs and anatomical label configurations. In *Intern. Workshop on Statistical Atlases and Computational Models of the Heart*. Springer (2017), 190–198.
19. Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Intern. Conf. on Medical Image Computing and Computer-Assisted Intervention*, Springer (2015), 234–241.
20. Sagi, M.R., Aurobind, G., Chand, P., Ashfak, A., Karthick, C., Kubenthiran, N., Murthy, P., Komaromy, M., and Arora, S. Innovative telementoring for addiction management for remote primary care physicians: A feasibility study. *Indian Journal of Psychiatry 60*, 4 (2018), 461.
21. Van Der Linde, D., Konings, E.E.M., Slager, M.A., Witsenburg, M., Helbing, W.A., Takkenberg, J.J.M., and Roos-Hesselink, J.W. Birth prevalence of congenital heart disease worldwide: A systematic review and meta-analysis. *Journal of the American College of Cardiology 58*, 21 (2011), 2241–2247.
22. Wang, C., MacGillivray, T., Macnaught, G., Yang, G., and Newby, D. A two-stage 3D Unet framework for multi-class segmentation on full resolution image. (2018), arXiv preprint arXiv:1804.04341.
23. Wolterink, J.M., Leiner, T., Viergever, M.A., and Išgum, I. Dilated convolutional neural networks for cardiovascular MR segmentation in congenital heart disease. In *Reconstruction, Segmentation, and Analysis of Medical Images*. Springer (2016), 95–102.
24. Wolterink, J.M., Leiner, T., Viergever, M.A., and Išgum, I. Generative adversarial networks for noise reduction in low-dose CT. *IEEE Transactions on Medical Imaging 36*, 12 (2017), 2536–2545.
25. Xu, X., Liu, Q., Yang, L., Hu, S., Chen, D., Hu, Y., and Shi, Y. Quantization of fully convolutional networks for accurate biomedical image segmentation. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition* (2018), 8300–8308.
26. Xu, X., Wang, T., Shi, Y., Yuan, H., Jia, Q., Huang, M., and Zhuang, J. Whole heart and great vessel segmentation in congenital heart disease using deep neural networks and graph matching. In *Intern. Conf. on Medical Image Computing and Computer-Assisted Intervention*, Springer (2019), 477–485.
27. Yamei. 5G remote surgery conducted in central China. *Xinhua* (2019), http://www.xinhuanet.com/english/2019-06/11/c_138134223.htm.
28. Yang, Q., Yan, P., Zhang, Y., Yu, H., Shi, Y., Mou, X., Kalra, M.K., Zhang, Y., Sun, L., and Wang, G. Low dose CT image denoising using a generative adversarial network with Wasserstein distance and perceptual loss. *IEEE Transactions on Medical Imaging* (2018).
29. Yu, L., Yang, X., Qin, J., and Heng, P-A. 3D FractalNet: Dense volumetric segmentation for cardiovascular MRI volumes. In *Reconstruction, Segmentation, and Analysis of Medical Images*. Springer (2016), 103–110.
30. Zontak, M. and Irani, M. Internal statistics of a single natural image. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference*, 977–984.

Meiping Huang (huangmeiping@163.com) is a chief physician at Guangdong Provincial People's Hospital, Guangdong Academy of Medical Sciences.

Xiaowei Xu is an assistant professor at Guangdong Provincial People's Hospital, Guangdong Academy of Medical Sciences.

Hailong Qiu is a Ph.D. student at Guangdong Provincial People's Hospital, Guangdong Academy of Medical Sciences.

Qianjun Jia and **Haiyun Yuan** are deputy chief physicians at Guangdong Provincial People's Hospital, Guangdong Academy of Medical Sciences.

Zeyang Yao and **Wen Xie** are master's students at Guangdong Provincial People's Hospital, Guangdong Academy of Medical Sciences.

Huiming Guo and **Jian Zhuang** are chief physicians at Guangdong Provincial People's Hospital, Guangdong Academy of Medical Sciences.

Yiyu Shi is an associate professor at University of Notre Dame.

© 2021 ACM 0001/0782/21/12

DOI:10.1145/3454008

Smart Farming with technologies such as IoT, computer vision, and AI can improve agricultural efficiency, transparency, profitability, and equity for farmers in low- and middle-income countries.

BY RANVEER CHANDRA AND STEWART COLLIS

Digital Agriculture for Small-Scale Producers:

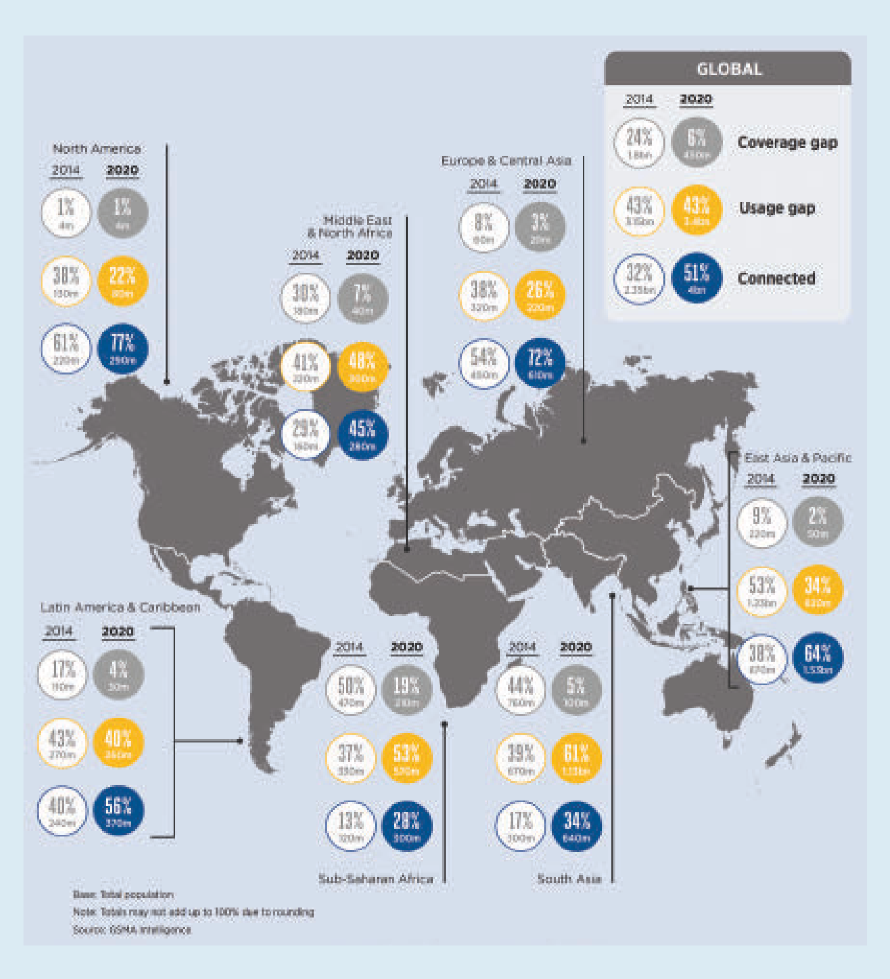
Challenges and Opportunities

THE FOOD AND Agriculture Organization of the United Nations (FAO) reports that, compared to 2010 levels, global food production needs to increase by 70% prior to 2050 to feed the world's growing population, which is expected to reach between 9.4 and 10.2 billion by then.[12] We need to achieve this goal in spite of the fact that the amount of arable land is not increasing, diets are changing, water demand is rising, the climate is changing, and both the environment and soil health are under pressure. These problems are most alarming in low- and middle-income countries (LMICs), which are expected to see the highest population increases,[19] leading to a growing demand for food and more diversified diets. In many LMICs, most of the population is rural, and more than 70% of farmers are small-scale producers (SSPs).[16] As has been seen in more developed nations, economic

» key insights

- 72% of the world's 570 million farms operate on less than 1 hectare.
- The average smallholder in sub-Saharan Africa lives on less than $2 USD per day.
- Digital solutions can improve access to finance, advisory, insurance, and market services for millions of smallholders.
- Only 13% of SSPs in sub-Saharan Africa have registered for digital services and less than 5% are active.
- Innovations in connectivity, sensors, AI, digital infrastructure, and usability are needed to drive the adoption of digital agriculture for smallholder farmers.

Figure 1. Mobile Internet connectivity by region 2020.[27]



Base: Total population
Note: Totals may not add up to 100% due to rounding
Source: GSMA Intelligence

growth in LMICs can reduce population growth and potentially improve livelihoods. LMICs need an agricultural transformation to help grow their economies, a daunting task made more difficult by such enormous obstacles.

Digital agriculture promises to help address many of these global challenges. Digitization of the food system can enable greater efficiency, transparency, profitability, and equity. The use of digital technology has seen rapid growth and investment, which has spurred many new innovations in Smart Farming.[3] These include sensors, Internet of Things (IoT), automation, Blockchain, artificial intelligence (AI), and computer vision. However, most innovation of this kind has been designed for high-income countries (HICs) and large commercial farming systems. Few digital innovations are designed specifically for LMICs and SSPs. Additionally, while digital solutions can potentially improve the lives of millions of rural poor, there are fundamental barriers to adoption. For example, mobile Internet for SSPs is not available to all, with sub-Saharan Africa and South Asia seeing some of the largest gaps.[17] In Figure 1, *coverage gap* represents the population living in areas with no mobile broadband. *Usage gap* represents people who live in areas with mobile coverage but do not have access, often due to handset or subscription costs, digital skills, literacy, trust, and safety, which are even bigger barriers to adoption amongst women.[26] Such barriers must be overcome to realize the potential of digital agriculture for SSPs.

In this article, we discuss how digital technologies can benefit small-scale producers, smallholder farmers, and small livestock operators. We highlight how digital agriculture innovations must be developed differently for these users, and, consequently, the need for new approaches to make those innovations feasible for LMIC value chains. Investing in and focusing on the right technologies for smallholder farmers could have enormous impact at scale. We contend that the increased use of digital technologies can not only help address global food-production challenges, but also improve the livelihoods of millions of SSP households.

The Promise of Digital Agriculture

Digital agriculture—that is, using digital technology and data to drive agricultural processes and decisions—can help the entire agricultural sector be more efficient, transparent, productive, profitable, and responsible. Sensors in the field, combined with automated farm equipment and data from drones and satellites, can provide new insights and better advisories to farmers.[9] While today it may just be an aspiration for SSPs, the entire operation of the farm can be automated. Better data and understanding of farm operation risks can help improve targeting of inputs and tailoring of finance and insurance products, as well as give buyers, off-takers, and commodity traders the insights they need to invest in agriculture.[7] Some of the key trends in digital agriculture, based on recent reports on agricultural investments, include:[3]

▸ **Sensors:** Existing sensors measure weather or basic soil properties. New sensor types are being developed, such as for nutrients or more accurate sensing of plants and livestock. There is also new research on in-plant sensing.

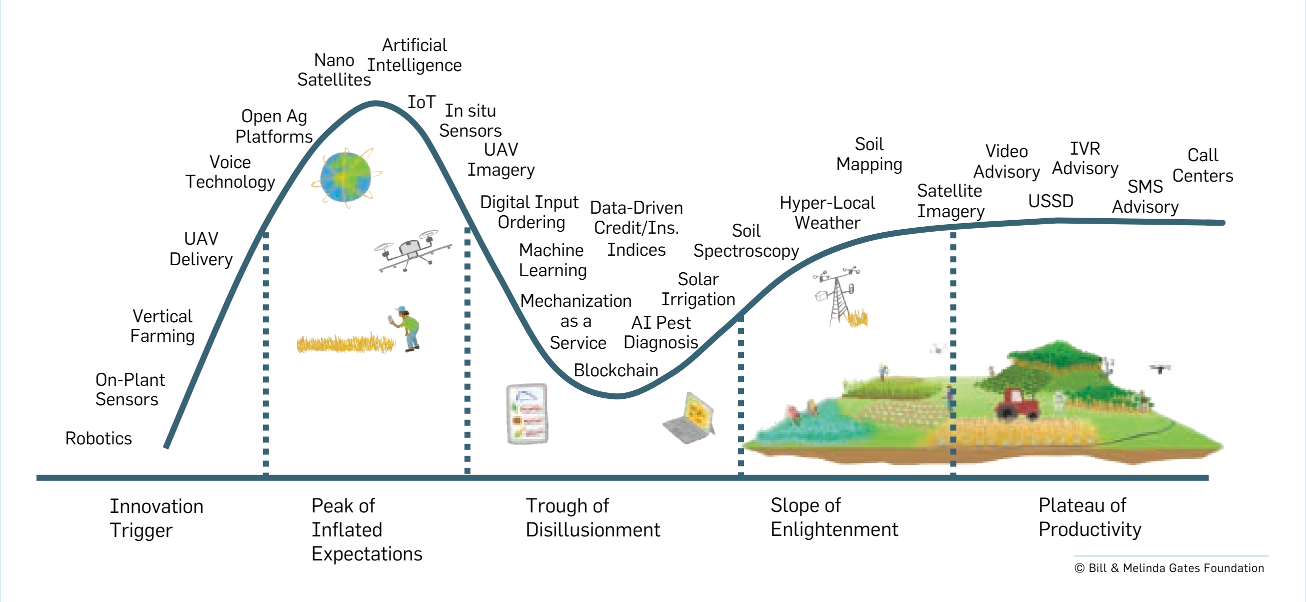
▸ **IoT:** The connection of small, cheap, and disposable sensors throughout IoT platforms enables real-time monitoring and cloud or edge computing, providing greater visibility and traceability of food throughout the supply chain. For example, if fresh produce exceeds temperature thresholds at any point during transit this can be recorded and flagged in real time. There is also a greater demand to learn where food comes from. Several startups are using IoT to build technologies that trace and monitor produce in storage units from the farm to the retail store.

▸ **Automation:** Robotic milling stations already exist for dairy and autonomous tractor applications. Startups are developing new applications around sowing, chemical application, irrigation, and weeding. Autonomous vehicles equipped with high-resolution cameras continually monitor crops as they grow through a process known as rapid phenotyping, which promises to increase the pace of new crop variety development.

- **Imagery:** Remote sensing, which in agriculture typically refers to analyzing images from satellite data, has been around for several decades. New trends center on either using low Earth orbit (LEO) satellites to access images more frequently or using higher-resolution imagery from drones—flying beneath clouds and available on demand—to monitor crop health or pest and disease outbreaks.
- **Blockchain:** Smart contracts and traceability can be ensured with secure blockchain technology, so that sellers, buyers, and consumers can be certain that information and data about the source and transit of food products and shipments is trustworthy and has not been compromised.
- **Artificial intelligence (AI):** As new data sources become available in larger quantity, resolution, timeliness, and quality, new techniques using distributed computing are required to process that data and convert it into actionable information. AI holds promise and machine learning (ML) is already being used to some degree—from crop prediction to chatbots—and will continue to be applied in more sophisticated ways. Augmented intelligence may be even more feasible to support SSPs' decision-making in the near-term.
- **Computer vision:** Algorithms are rapidly improving to take advantage of the larger amounts of available imagery and photos. An AI model analyzing a photograph of a leaf, for example, can quickly identify specific plant pests and diseases, and make real-time treatment recommendations. Farm field boundaries can also be automatically extracted from satellite images.

These technologies are in various stages of readiness for agriculture, and many are still at the peak of expectations for precision agriculture[20] in HICs. The Gartner Hype Cycle reflects the visibility of technologies as they mature. Technologies at the peak indicate a lack of clear, evidence-based applications—that is, they may be overhyped and confusing the landscape. We present a version of the Gartner Hype Curve for SSPs (see Figure 2), adapted and updated from Rakestraw et al.[20] and Gray et al.,[13] because the same assumptions about readiness are not necessarily consistent across HIC and LMIC contexts. We also propose that an improved innovation assessment framework may be required, such as the Technology Readiness Levels created by NASA[6] and previously adopted for crop research[23] or another innovation-scaling framework.[14] Regardless, the advantages of

Figure 2. Digital agriculture hype curve for SSPs.



© Bill & Melinda Gates Foundation

Figure 3. Smallholder farmer typologies.[2]

		Importance of farm sales (Agricultural sales/total agricultural income)		
		Low	Medium	High
Importance of non-farm income (Non-farm income/total household income)	Low	Subsistence farms	Pre-commercial farms	Specialized commercial farms
	High	Transitioning farms		Diversified commercial farms

digitization and application of the right technologies are likely to bring SSPs enormous benefit. We describe some of these in subsequent sections.

Small-Scale Producers and Farming in the Developing World

The agriculture sector is one of the primary employers in LMICs, employing more than 80% of the rural population in some countries.[15] The majority of the world's farmers are smallholders, with 72% of the world's 570 million farms operating on less than 1 hectare.[16] Agriculture is a significant driver of economic growth in these countries and critical to overall gross domestic product (GDP).[18] Many smallholder farmers are producing for their own consumption, but for agriculture to contribute to economic growth and help bridge looming food gaps, farmers must invest in their operations, produce more, and sell into markets.

For agriculture to contribute to economic growth and help bridge looming food gaps, smallholder farmers must invest in their operations, produce more, and sell into markets.

Who are smallholder farmers? According to AGRA's 2017 Africa Agriculture Status Report,[2] smallholder farmers provide livelihoods for more than two billion people and produce about 80% of the food in sub-Saharan Africa and Asia. While a generalized definition of a smallholder farmer is based on area (less than 1 hectare or 2.5 acres), a more nuanced typology is based on the share of crop production value that is sold, and the share of non-farm income to total-household income, as shown in Figure 3. Based on this definition, the general categories are:

a. Subsistence-oriented small farms, which sell less than 5% of their agricultural output and obtain less than 33% of their total income from non-farm sources.

b. Transition farms, which obtain 33% or more of their income from non-farm sources and sell up to 50% of their crop output.

c. Pre-commercial small farms, which sell 5% to 50% of their production and earn less than 33% of their income from non-farm sources.

d. Commercial small farms, which sell 50% or more of their production. Commercial small farms sub-divide further, into specialized commercial farms, if their non-farm income share is less than 33%, or into diversified commercial farms otherwise.

Across the groups that have farming as their main source of income—that is, subsistence, pre-commercial, and specialized commercial—the average annual farm income is around $780 per year. To be able to design solutions for them, it is critical to understand this in terms of the scale required and the resources available to smallholder farmers to access and use technology.

Small-scale producer farm families lead a challenging existence. The average smallholder farmer in sub-Saharan Africa lives on less than $2 USD per day. They are often growing food for their own consumption and when they do have access to markets, they often don't receive competitive market prices. With no way of storing produce post-harvest, most farmers in a region are selling simultaneously, driving prices down. Smallholders often don't have access to the latest seed, inputs, and advice for maximizing yields—they are forced to rely on older, traditional, or counterfeit varieties and inputs, and to use antiquated practices. If farmers do have access to a government extension agent, visits are often infrequent, with agent-to-farmer ratios as low as one agent to 5,000 farmers.[25]

This means their crops are more vulnerable to the risks of climate change, weather variability, pests, and disease. Recent outbreaks in East Africa of fall armyworm and locusts have caused significant losses to production. Smallholders have limited ability to absorb these shocks because they have little or no access to affordable credit or insurance due to lack of collateral, ineffective land tenure systems, or the fact that they are landless contractors. With these challenges and barriers, and the looming persistence of poverty in LMICs, improvements to the livelihoods of the rural poor are difficult to realize. It is in this context that digital solutions tailored to smallholder farmers need to be designed.

Digitally Enabling Agricultural Transformation for Smallholder Farmers

Data about farms, coupled with insights from the latest advances in technology, can help transform the livelihoods of smallholder farmers and SSPs. Digitization of smallholder agriculture can reduce risk, improve productivity, increase incomes, empower women, and help solve the impending challenge of producing enough nutritious food for the world's rapidly growing population. Some promising trends are enabling digital technology adoption, including:

▸ Smartphone costs are rapidly decreasing. A Jio phone in India, for example, costs less than $10 USD and data plans are becoming more affordable. More countries are reaching data costs of less than 2% of monthly income.[27]

▸ Social media usage is rapidly growing. Social media and tools such as WhatsApp and Facebook are seeing rapid adoption in LMIC's.

▸ Mobile money has enabled growth of digital services. MPESA in Kenya, for example, has been attributed to a 2% decrease in poverty.[23]

▸ Mobile Internet coverage is increasing, enabling smartphone use, access to information, app usage, and two-way data flows. LMIC 4G coverage increased from 30% in 2014 to 75% in 2018.[27]

▸ Digitization of services, such as access to inputs; access to finance, insurance, and advisory services; and connections to markets, is taking place.[10]

▸ Voice and conversational AI, in the form of voice assistants or chatbots, shows potential for providing farmers with automated advice and more streamlined access to information and services.

Furthermore, there is some evidence that the application of digital technologies for relevant use cases results in higher smallholder farmer productivity and income, particularly when those services are bundled. Figure 4, from the Digitalization of African Agriculture Report,[10] shows the multiplier effect of bundling services. These use cases are outlined in Table 1 and summarized below:

Agricultural advisory services. If the data from a farm is captured, whether from in-field sensors or remote-sensed by satellite or drone, it can be combined with agronomic science to create digital advisories for:

▸ Planning: What crop to grow in a particular season based on expected weather, crop prices, and market demand.

▸ Planting: When to sow seed based on crop type and predicted weather.

▸ Management: When to irrigate, fertilize, and apply pesticide.

▸ Harvest: When to harvest the crop based on market prices, predicted weather, and drying and storage costs.

And, by digitizing extension and development agents, even farmers who do not directly interact with digital tools themselves can benefit from more accurate and timely information through such digitally enabled intermediaries.

Market linkages. Digital tools can help connect farmers to markets. On the input side, digital ordering services can help connect farmers to certified seeds and fertilizers—for example, DigiFarm and iProcure in Kenya. On the output (harvest) side, access to current market prices and digital connection to transport and aggregation centers—for example, Loop in India—can help farmers reduce costs and increase profit. Similarly, milk producers in India can receive instant

Figure 4. Digital agriculture impact on smallholder farmers.[10]

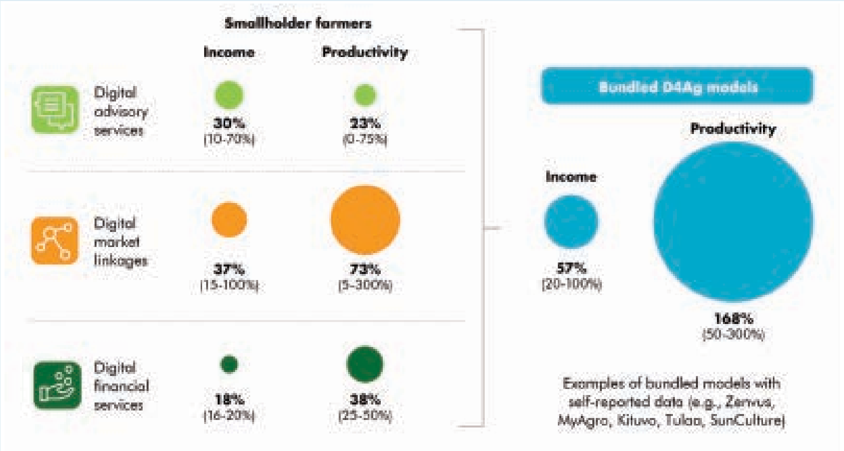
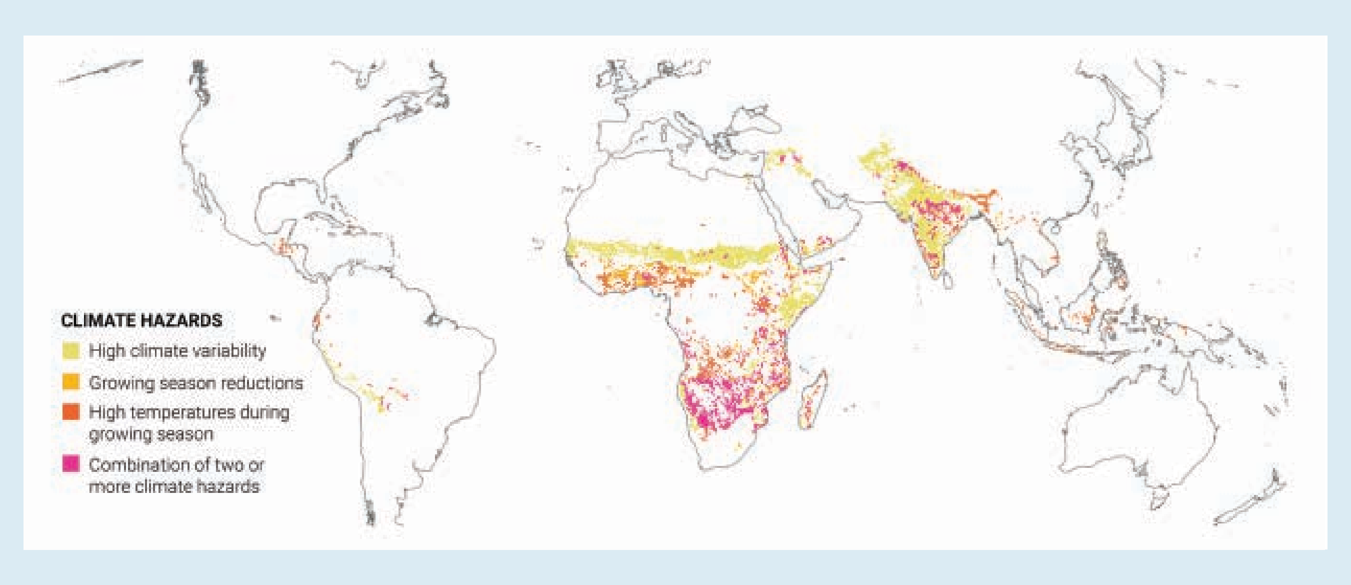


Table 1. Digital services for smallholder producers, and technology needs, including a few examples.

	Sensors and IoT	Satellite Imagery	Cameras	Data and Compute Platforms	ICT and Notifications	Drones and Robots
Advisories Planning, farm management	Monitor weather, soil moisture, temperature, nutrient equipment	Crop health, farm field boundary	Pest detection, livestock monitoring	Secure data sharing, ground truth data, scalable AI	SMS/IVR, P2P video education	Auto-spraying, scouting, seeding
Market Linkages Logistics, buyers, inputs	Monitor storage (temperature, humidity), truck monitoring	Logistics planning	Detect crop quality, bid livestock	Smart contracts, crop area, price prediction	Connect input suppliers, buyers, markets	AVs, drone delivery
Financial Services Insurance, loans	Farm monitoring, credit scoring	Credit score, index insurance	Livestock risk adaptation	Biometric ID, mobile money	Damage and law enforcement	Damage and loss assessment
Sustainability GHG estimation, climate adaptation	Carbon, nitrogen, water sensors, climate advisory	Verification of farm management	NIR, carbon sensors	Carbon verification, carbon exchange	Incentives for regenerative ag, climate alters	Robotic weed removal vs. tilling

© Microsoft Corporation. All rights reserved.

Figure 5. Areas of high agricultural risk for different climate hazards in vulnerable areas.[1]


CLIMATE HAZARDS
High climate variability
Growing season reductions
High temperatures during growing season
Combination of two or more climate hazards

feedback on quality through a digital scale and testing service and get rewarded for higher quality.

Traceability of agricultural products from farm to consumer for food safety and transparency can also unlock opportunities to enable small-scale producers to participate in regional and global marketplaces.

Financial services and insurance. Farming businesses get a return on their investment once they sell their produce to a buyer. This could be 90 days (from seed to harvest) after they obtain capital to purchase seeds, fertilizer, pest and disease treatments, and hire labor during the season for harvesting and transportation. Credit can be obtained from banks and other creditors if farmers have collateral, such as land or other assets. However, many smallholder farmers do not own the land they are farming and don't own other forms of collateral. Farmers may not be identifiable, or their land ownership cannot be verified, making it hard to get credit at reasonable rates.

More data about the farmer and farm can provide financial services and insurance providers with valuable information to assess risk and provide smallholder farmers with more-tailored credit products, loans, and insurance policies. There are companies in Africa and South Asia—for example, Pula, Oko, Acre, Farmdrive, and Skymet—working to assess risk using satellite imagery and weather data to create credit-scoring algorithms and insurance risk assessments. If more data about a farmer and the farm field could be integrated to create trusted and verifiable credit scores and risk profiles, this could unlock finance and risk-mitigation tools that smallholder farmers need to invest in their farms to try new technologies.

Agricultural research and development. In the field of plant science, plant phenotyping refers to the set of methodologies and protocols used to accurately measure plant growth, architecture, and composition at different scales. Next-generation sequencing technology has greatly accelerated functional genomics, allowing for the identification of important genes and agronomic traits. The understanding of how genetics interact with the environment (GxE) through phenotyping is an essential part of crop breeding. Rapid phenotyping is now possible with digital technology—for example, high-resolution, multi-spectral imaging; drone imagery; in-ground sensors; and data platforms—enabling precise measurement, analytics, and digital twins to be generated. These solutions, however, are not yet adopted in LMICs, and development is still required, especially for below-ground measurement of root structures.[28]

Sustainability and climate. Climate change is one of the greatest threats facing humanity.[1] For agriculture, climate change could depress crop yields up to 30% by 2050 without adaptation. Small farms around the world will be most affected (see Figure 5). Digital technologies will be essential to implementing the three action plans identified by the commission: increase research on agricultural climate adaptation; expand access to climate-informed digital advisory services; and expand small-scale food producers' access to insurance, finance, markets, adaptive technologies, and agroecological practices.

Long-term sustainability of farming systems is also dependent on improving and maintaining soil health. Unfortunately, soil health is in decline in many regions, limiting the soil's ability to support higher productivity. Without investment in soil health, crop yields would decline. Digital soil spectroscopy sensors combined with satellite imagery can produce soil property maps and provide the information needed for tailored agronomic advisory. Similar application of technology to measure soil carbon could also help open new revenue streams for smallholder farmers, which many proponents argue may be possible through carbon markets.

Barriers to Adoption and Research Opportunities

The technologies discussed in the previous section, such as sensors, imagery, or automation, still aren't widely adopted in smallholder farmer systems.[22] Only 13% of SSPs in sub-Saharan Africa have registered for digital services and even fewer are active.[10] There are several reasons for this. For example, despite improvements in mobile Internet coverage, there is still a staggering usage gap of 3.3 billion people who live in areas covered by mobile broadband but are not using mobile Internet services[27]—and the usage gaps are significant in sub-Saharan Africa and South Asia (Figure 1). These figures tell us that the factors creating the digital divide go beyond technology, such as:

Connectivity and access. Many farms and rural areas do not have quality Internet access. The ITU estimates that around half of the world's population does not have Internet access, and many of them are in the developing world. The situation is improving; at the end of 2018, the mobile coverage gap represented 10% of the world population compared to 24% in 2014, but most of this gap exists in the marginal, rural areas of LMICs.[27]

Even with good connectivity, there are still adoption challenges to overcome. Significant digital gender gaps exist across LMICs—313 million fewer women than men use mobile Internet, representing a gender gap of 23%.[26]

What is needed? New technology solutions need to be developed to provide low-cost Internet access. To connect cameras or tractors, and support video, this connectivity needs to be broadband, which in the U.S. is defined as 25 Mbps downlink, and 3 Mbps uplink. We note that for precision agriculture technologies, uplink capacity needs to be greater than downlink, since most data is sent to the cloud. To address gender gaps and usage issues more broadly, there should be an emphasis on human-centered design that considers social context. Women, for example, may share a mobile handset, in which case separate accounts for personal access to services may be important.

Affordability. Smallholder farmers are financially constrained. Consequently, digital solutions for smallholder farmers need to be affordable. Existing on-farm solutions are cost-prohibitive. Sensors cost a few hundred dollars and can be prone to theft in these geographies. Drones cost a few thousand dollars, have limited battery life, and access to energy is scarce. Furthermore, tractors are often not a viable option for small-scale producers due to the high capital cost. Remote-sensing solutions require the processing of large amounts of satellite imagery in the cloud. AI techniques on this data require GPUs and further increase the cost of providing digital solutions.

What is needed? New techniques need to be invented to accomplish the same task as expensive devices, but at a low cost. In some cases, technology can be replaced by using less-costly manual components or by adopting creative service-delivery business models.

Literacy and skills. Many farmers in LMICs are not literate and technology skills are low. The GSMA Gender Report[26] cites literacy and skills (linguistics and technology) as the highest barriers to digital-device adoption.

What is needed? Technologies to translate insights and to make them usable by smallholder farmers. In some cases, this may also require a new UI for smartphones or research into voice technology. For farmers with older-feature phones, new methods are needed to convey digital insights and design to meet users where they are on the digital and language-literacy spectrum.

Timely and relevant information. Relevance is one of the barriers to adoption of services. One way to ensure

Table 2. Research problems that inhibit the deployment of technologies for smallholder producers.

Open research problems in various areas of computer science that are inhibiting the deployment of different technologies for smallholder producers. Green, yellow, and red indicate the need—less, needed, or critical, respectively—for research in that area for that technology to be adopted by smallholder producers.

	Hardware and Architecture (Affordability)	**Vision, Speech, ML/AI** (Relevant Data)	**Systems, Security** (Connectivity Data Platforms)	**Human Interface** (Usability)
🌡	🟨 Low-cost sensing (RF, audio), low-power sensors, sensing roots/carbon	🟩 Surrogate sensing Microclimate prediction Advisories, for example, water Livestock health, estrus	🟨 Low-cost IoT networks Secure data ingestion Reliable sensor system Data sharing	🟩 Display alerts Automated diagnosis Fault recovery
🛰	🟨 High-res optical cams Satellites SAR probes LEO constellations	🟨 Cloud removal, SAR AI/ML for yield/disease/etc. Accurate super -res imagery	🟩 Satellite downlink speed Merging IoT + remote sensing Timely analysis	🟨 Visualizing form imagery and insights
🚁	🟥 Battery life Wireless charging Low-cost robots	🟥 Localization below canopy ML without labeled data automation in mixed crops	🟨 Edge compute Shared robots Large data transfer Secure sharing	🟥 Low-tech operation Interpretation of results
📷	🟥 Low-power cams Low-cost multispectral	🟨 Livestock stress detection Cow health Pest detection	🟨 Broadband Edge compute Federated ML	🟨 Ease of use Theftproof
💬	🟨 Low-cost devices with rich UI	🟨 Local speech -to-text, digitizing knowledge	🟩 Internet systems (e.g., IFTTT) with phones	🟩 Geospatial insights on SMS, new apps
☁	🟨 Secure edge Low-cost sovereign data centers	🟨 Obtain ground truth data, AI on unlabeled data, multimodal data	🟥 Secure data sharing AI on encrypted data Data collaboratives Models	🟨 Ease of data sharing Awareness of misuse

© Microsoft Corporation. All rights reserved.

information services are more relevant is to base them on timely and spatially relevant data. With little in-field observations from the farm itself, existing solutions often rely on remotely sensed data from satellites. However, the small size of farms challenge existing solutions that are typically designed for farms in HICs, which span tens of acres and more. In LMICs, the farms are typically less than 1 hectare, and producers often plant multiple crops in that plot. Since each small farm is only a few pixels in a satellite image, it is difficult to extract intelligence for each crop in the field.

What is needed? Higher-resolution and more timely data, and information and insights at an affordable cost, are required to enable smallholder farmers to be more precise in their operations and more resilient to climate-induced and other shocks.

Data trust and security. Despite the need to obtain more data about smallholder farmers, there are stakeholders collecting information about agriculture in LMICs. Theoretically, if this information could be shared and aggregated, it could benefit the entire agriculture industry, including SSPs. However, one of the biggest barriers to sharing data is trust about data usage and consumer protection. Farmers, input suppliers, buyers, traders, financial companies, governments, and other entities all have varying incentives to collect and share data, yet there are few effective mechanisms for securely sharing data. Additionally, data breaches can have dramatic repercussions, yet securing data can be costly and complex.

What is needed? Research into secure data platforms, peer-to-peer and privacy-preserving data-sharing models, and data marketplaces will provide stakeholders with choices about how to share data with appropriate protections against misuse, and within privacy and consumer protection laws.

Open Research Problems

For the digital technologies presented in the previous section to be adopted by small-scale producers, we need research in different areas of computer science. For example, to enable the adoption of sustainable agricultural practices, computer scientists need to develop low-cost carbon sensors; new AI techniques to estimate carbon, nitrogen, and water use from satellite imagery and other sources; and new user interfaces to convey the insights to smallholder farmers—all using low-cost techniques. We highlight some of these challenges for various digital agriculture scenarios listed in Table 2.

There is still a staggering usage gap of 3.3 billion people who live in areas covered by mobile broadband but are not using mobile Internet services.

Hardware and architecture innovation: Making solutions more affordable. Research is needed to design low-cost hardware architecture that is as functional as existing solutions. Smallholder producers can benefit from various types of hardware, including ground sensors that measure soil moisture and temperature; weather stations that measure abiotic properties; motion sensors that monitor livestock; cameras that identify pests and diseases; and drone imagery that helps to identify crop stress, predict yield, and recommend action to bridge the yield gap.

A few desired features of the hardware for smallholder farmers are:

- **Lower cost:** Technologies to sense or convey information at a lower price point.
- **Low power:** Sensors or cameras should operate for long periods of time with low and intermittent power, perhaps using renewable energy sources or low-cost energy-storage solutions.
- **Ruggedized:** Technologies on a farm must operate in harsh conditions, including adverse weather, floods, and in the presence of wild animals.
- **Theft-proof:** Devices that can operate out of sight or can trigger an alarm may be more usable.

Achieving all the beneficial features is difficult. A few promising approaches are being explored by startups and academia. The Chameleon soil moisture sensor[8] developed by CSIRO uses low-cost components to display different-colored lights, instead of sending values to the cloud, to indicate soil moisture content. Surrogate sensing is another promising approach for reducing cost. MEMS and audio technology can help build low-power sensors to measure agricultural parameters. Recent work leverages Wi-Fi chipsets on smartphones as a sensor for soil moisture and electrical conductivity.[11] GroGuru has developed sensors that can be placed deep in the soil and are not visible to the human eye.

Other promising research has explored alternative imaging solutions. Plantix uses a smartphone's camera to detect pests and diseases. TYE[24] uses a camera (or smartphone) mounted on a helium balloon and transported by a human on a tractor or a bicycle as an aerial imaging alternative to drones. Further research, either on the use of low-cost components or alternative-sensing methods, can help make sensors and cameras more affordable.

These technology innovations can also lead to potentially new business models. Instead of deploying multiple sensors, a farmer could manually move a sensor to different parts of the farm to determine the soil map—for example, for nutrients. Alternatively, crop insurance, finance providers, or the government could subsidize the cost of hardware on the farm.

AI/ML, speech, and computer vision research: Deriving relevant data. Advances in computer vision and AI have already brought significant benefits to agriculture. Commercial tools, such as Climate Fieldview, Farmers Edge, and Land O'Lakes R7, can detect crop stress; predict weather, yields, and outcomes; and provide natural language and voice interfaces to growers. However, additional research is needed to realize these benefits for smallholder farmers.

Key research challenges include:

▸ **AI on low-resolution satellite data:** Each pixel in satellite imagery is a few meters, which is often too coarse for small farms.

▸ **Multi-cropping systems:** Several smallholder producers practice subsistence farming. They plant multiple crops in the same farm, which leads to challenges in the ways different crops interact and how digital systems can isolate the performance of each crop to provide appropriate advisories.

▸ **AI on sparse data:** There is a dearth of good, labeled data from small farms. Satellite data is limited, sensors are also sparsely deployed, and voice data is collected for the most-spoken languages. New AI techniques need to be developed to label the available data and augment data streams to train models for smallholder producers. While AI holds promise, more immediate solutions could leverage an augmented intelligence approach.

▸ **Downscaled climate models:** Accurate weather forecasts are available in HICs but less so in LMICs due to cost and complexity. Localized forecasts supported by low-cost sensors would improve a small-scale producer's ability to adapt to climate change and be more resilient. High-resolution seasonal forecasts in the tropics are important for seed selection, insurance, subsidy programs, and food security.

There are a few promising approaches to address the sparse data challenges noted above. However, they have not yet been applied at scale for agriculture. For example, generational adversarial networks (GANs) can help generate new data to train the AI models. This can also be generated from 3D simulations of farms. Recent advances in graph neural networks can help incorporate human knowledge to overcome the limitations of sparse data. Self-supervised learning schemes can combine different data sources to generate more labeled data. Applying these and other AI techniques in agriculture can help bring the benefits of the latest in AI to smallholder farmers.

Networking, systems, and security research: Connectivity, edge, data platforms. Farmers need low-cost Internet connectivity on their devices. The sensors, cameras, and other devices need connectivity on farms. However, existing solutions are very expensive. A key challenge is how to make high-speed Internet more affordable.

To deliver broadband, an Internet service provider (ISP) typically uses a mixture of technologies, such as satellite communications, fixed-wireless, or fiber-optic cables. Wired technologies work best in densely populated areas, such as cities. For remote locations, satellite connectivity, in conjunction with some terrestrial technology, is more cost-effective. Among terrestrial technologies, fixed wireless in mmWave works well for high-speed, short-range access. Lower frequencies propagate further and are more suited to provide broadband in rural areas.

Long-range, affordable broadband access can be provided using unlicensed spectrum and open-source technologies, and can bring down the cost of Internet access. A few promising technologies include:

▸ **TV white spaces:** Unused TV spectrum can be leveraged to opportunistically send and receive data. Since most TV channels are unused in rural areas, this provides large amounts of available capacity for broadband connectivity.

▸ **LEO satellite-based Internet access**, such as from SpaceX, is another promising technology. It is relatively inexpensive to launch a swarm of cubesats, such as those being launched by SpaceX, which can then provide Internet access worldwide.

▸ **Private LTE:** Private cellular networks, such as Endaga, use community cellular networks to bring connectivity to rural areas.

There has been other recent work on leveraging airplanes[4] and Google's Loon balloons for delivering connectivity. However, any connectivity technology solution must be accompanied by innovative business models to make them economically viable for smallholder farmers. For example, an ISP might provide digital agricultural advisories in addition to Internet access to offset the cost of deploying a communication tower and infrastructure.

Another promising technology is edge computing. Not all data needs to be sent to the cloud. Instead, large amounts of it can be processed on a computing device closer to the farm. The edge could gather imagery, perform analytics, and provide a digital advisory service, which can also run offline. However, edge compute devices need to be low-power, ruggedized, and may, in some cases, need to run an app store for completely disconnected operation.[5]

A privacy-preserving, secure, and sovereign data-sharing platform can help aggregate data, both for creating market linkages and for delivering AI-based advisories. However, sharing geolocation attributes, such as farm parameters, weather, or farm management, might reveal the producer's identity. Technologies such as Confidential Compute Framework and homomorphic encryption are promising, but need to be adapted for geospatial data.

Computer human interfaces: Improving usability. To reach producers who are not very tech savvy, who typically have a feature phone, and who have literacy challenges, we need to look beyond a graphical user interface (GUI).[15] There has been work with using non-GUI interfaces, such as those based on speech, haptics, and gestures. The two most common non-GUI user interfaces for agriculture that have been explored are speech and haptics/gestures. The latter is useful for robotics and autonomous vehicles, while speech has been studied as an interface for farmers in India and Africa.

Startups use different user interfaces to reach smallholder producers. Digital Green uses video to educate growers about farming techniques. Awaaz De uses audio, missed calls, and text messages to send notifications and alerts to producers. Many others use either SMS messages, voice calls, or both.

The latest advances in voice technology, which automates and personalizes messaging apps; bots; and speech-based agents, such as Alexa, Cortana, Siri, or Google Voice, can help reduce barriers to digital service access. However, these speech AI models must be customized to learn local dialects and mannerisms. This is not trivial; there might not be enough data to train AI models. Research challenges include:

▸ **Support for regional languages:** Growers in rural communities typically only know their regional languages; therefore, the user interface must be trained for every region.

▸ **Two-way interactions:** Several interactions, such as price discovery and digital advisories, require two-way communication between the grower and the service provider. Text messages are too short, while speech is often used only to convey information to the grower.

▸ **Geo-spatial insights over text messages or speech:** A digital advisory might have a geo-spatial component, such as the geo-coordinates of crop stress, or a micro-scale nitrogen map of the farm. In the developed world, this insight is conveyed using 2D or 3D maps. Conveying the same information over text messages or speech is challenging.

Summary

This paper highlights the innovations that are needed across different areas of data science and computer science to enable digital agriculture for small-scale producers. This includes research across various ACM Special Interest Groups (SIGs), such as SIGCOMM, SIGOPS, SIGARCH, SIGCHI, SIGGRAPH, and many more, to devise affordable hardware for cloud and networking solutions, with innovations in AI to handle sparse and incomplete data.

As technology solutions emerge, it is also important to understand the path to market and scalable adoption. As such, we are working on a Digital Agriculture Technology Readiness Index[6] and calculator to evaluate the maturity of different digital technologies for small-scale agriculture in Table 2. The output of this index will guide our efforts towards accelerating research in technologies that are not yet ready for adoption and investing in technologies as they mature. We are excited to work with the computer science community to address some of the hardest problems, so that we may help accelerate the adoption of digital agriculture for smallholder producers, improve the livelihoods of millions of families, and ensure greater global food security.

References

1. Adapt now: A global call for leadership on climate resilience. *Global Center on Adaptation* (2020), https://cdn.gca.org/assets/2019-09/GlobalCommission_Report_FINAL.pdf.
2. Hazell, P.B. Why an inclusive agricultural transformation is Africa's way forward. In *AGRA (Ed.), The Business of Smallholder Agriculture in Sub-Saharan Africa 5* (2017), 10. Nairobi, Kenya: Alliance for a Green Revolution in Africa (AGRA).
3. Agrifoodtech investment report. *AgFunder* (2021), https://research.agfunder.com/2021/2021-agfunder-global-report.pdf
4. Ahmad, T., Chandra, R., Kapoor, A., Daum, M., and Horvitz, E. Wi-Fly: Widespread opportunistic connectivity via commercial air transport. In *Proceedings of the 16th ACM Workshop on Hot Topics in Networks* (2017), 43–49. https://doi.org/10.1145/3152434.3152458.
5. Ahmad, T. et al. GreenApps: A platform for cellular edge applications. In *Proceedings of the 1st ACM SIGCAS Conference on Computing and Sustainable Societies*, Ellen W. Zegura (Ed.). (2018), 45:1–45:5; https: //doi.org/10.1145/3209811.3212704.
6. Banke, J. Technology readiness levels demystified. NASA (2010), https://www.nasa.gov/topics/aeronautics/features/trl_demystified.html.
7. Bronson, K. and Knezevic, I. The digital divide and how it matters for Canadian food system equity. *Canadian J. of Communication 44* (2019), https://doi.org/10.22230/cjc.2019v44n2a3489.
8. Chameleon soil water sensor. *CSIRO.* https://www.csiro.au/en/Research/AF/Areas/Food-security/Chameleon-soil-water-sensor
9. Chandra, R. FarmBeats: Automating data aggregation. *Farm Policy Journal 15* (August 2018), 7–16. https://www.microsoft.com/en-us/research/ publication/farmbeats-automating-data-aggregation/.
10. Digitalization of African agriculture report. *CTA* (2018–2019). https://www.cta.int/en/digitalisation-agriculture-africa
11. Ding, J. and Chandra, R. Towards low-cost soil sensing using WiFi. In *ACM MobiCom* (2019), https://www.microsoft.com/enus/research/publication/towards-low-cost-soil-sensing-using-wi-fi/.
12. Global agriculture towards 2050. *FAO High Level Expert Forum—How to Feed the World in 2050*, Rome, Italy (2009), http://www.fao.org/fileadmin/templates/wsfs/docs/Issues_papers/HLEF2050_Global_Agriculture.pdf.
13. Gray, B. et al. Digital farmer profiles: Reimagining smallholder agriculture (2018), https://www.usaid.gov/sites/default/files/documents/15396/Data_Driven_Agriculture_Farmer_Profile.pdf.
14. Insights on scaling innovation. *Intern. Development Innovation Alliance* (2017), https://static1.squarespace.com/static/5b156e3bf2e6b10bb0788609/t/5b1717eb8a922da5042cd0bc/1528240110897/Insights+on+Scaling+Innovation.pdf.
15. Kortum, P. *HCI Beyond the GUI.* Elsevier, (2008).
16. Lowder, S., Skoet, J., and Raney, T. The number, size, and distribution of farms, smallholder farms, and family farms worldwide. *World Development 87* (2016), 16–29. https://doi.org/10.1016/j.worlddev.2015.10.041.
17. Mehrabi, Z. et al. The global divide in data-driven farming. *Nature* (2020), https://doi.org/10.1038/s41893-020-00631-0.
18. Mellor, J. *Agricultural Development and Economic Transformation*. Palgrave Macmillan, Cham (2017), https://link.springer.com/book/10.1007/978-3-319-65259-7.
19. Our world in data. (2020), https://ourworldindata.org/.
20. Rakestraw, R. Precision Ag innovation hype curve. *Precision Ag Vision Conference* (2016), https://drive.google.com/file/d/0B0w0TilL5ROMHA5bm1HS3pycVU/view.
21. Sartas, M., Schut, M., Proietti, C., Thiele, G., and Leeuwis, C. Scaling readiness: Science and practice of an approach to enhance impact of research for development. *Agricultural Systems* 183 (2020), 102874; https: //doi.org/10.1016/j.agsy.2020.102874.
22. Scaling up disruptive agricultural technologies in Africa. *World Bank Group* (August 2019), https://olc.worldbank.org/content/executive-summary-scaling-disruptive-agricultural-technologies-africa.
23. Suri, T. Mobile money. *Annual Review of Economics 9*, 1 (2017), 497–520; https://doi.org/10.1146/annurev-economics-063016-103638 arXiv; https://doi.org/10.1146/annurev-economics-063016-103638.
24. Swamy, V., et al. Low-cost aerial imaging for smallholder farmers. In *ACM Compass* (2019), https://www.microsoft.com/en-us/research/publication/low-cost-aerial-imaging-for-small-holder-farmers/.
25. Swanson, B. et al. In-depth assessment of the public agricultural extension system of Ethiopia and recommendations for improvement. *International Food Policy Research Institute* (2010), https://core.ac.uk/download/pdf/6237665.pdf.
26. The mobile gender gap report. *GSMA* (2020), https://www.gsma.com/r/ gender-gap/.
27. The state of mobile Internet connectivity 2020. *GSMA* (2020), https://www.gsma.com/r/somic.
28. Yang, W., et al. Crop phenomics and high-throughput phenotyping: Past decades, current challenges, and future perspectives. *Molecular Plant 13*, 2 (2020), 187–214. https: //doi.org/10.1016/j.molp.2020.01.008

Ranveer Chandra is the managing director of Research for Industry and leads Networking Research at Microsoft Research in Redmond, WA, USA. His research has shipped in multiple Microsoft products, including Xbox, Azure, and Windows.

Stewart Collis is senior program officer for Digital Agriculture Solutions at the Bill and Melinda Gates Foundation where he focuses on digital farmer services, smart farming, and digital support systems for small-scale crop and livestock producers in low- and middle-income countries.

© 2021 ACM 0001/0782/21/12

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

acm
Association for
Computing Machinery

DOI:10.1145/3458723

Documentation to facilitate communication between dataset creators and consumers.

BY TIMNIT GEBRU, JAMIE MORGENSTERN, BRIANA VECCHIONE, JENNIFER WORTMAN VAUGHAN, HANNA WALLACH, HAL DAUMÉ III, AND KATE CRAWFORD

Datasheets for Datasets

DATA PLAYS A critical role in machine learning. Every machine learning model is trained and evaluated using data, quite often in the form of static datasets. The characteristics of these datasets fundamentally influence a model's behavior: a model is unlikely to perform well in the wild if its deployment context does not match its training or evaluation datasets, or if these datasets reflect unwanted societal biases. Mismatches like this can have especially severe consequences when machine learning models are used in high-stakes domains, such as criminal justice,[1,13,24] hiring,[19] critical infrastructure,[11,21] and finance.[18] Even in other domains, mismatches may lead to loss of revenue or public relations setbacks. Of particular concern are recent examples showing that machine learning models can reproduce or amplify unwanted societal biases reflected in training datasets.[4,5,12] For these and other reasons, the World Economic Forum suggests all entities should document the provenance, creation, and use of machine learning datasets to avoid discriminatory outcomes.[25]

Although data provenance has been studied extensively in the databases community,[3,8] it is rarely discussed in the machine learning community. Documenting the creation and use of datasets has received even less attention. Despite the importance of data to machine learning, there is currently no standardized process for documenting machine learning datasets.

To address this gap, we propose *datasheets for datasets*. In the electronics industry, every component, no matter how simple or complex, is accompanied with a datasheet describing its operating characteristics, test results, recommended usage, and other information. By analogy, we propose that every dataset be accompanied with a datasheet that documents its motivation, composition, collection process, recommended uses, and so on. Datasheets for datasets have the potential to increase transparency and accountability within the machine learning community, mitigate unwanted societal biases in machine learning models, facilitate greater reproducibility of machine learning results, and help researchers and practitioners to select more appropriate datasets for their chosen tasks.

After outlining our objectives, we describe the process by which we developed datasheets for datasets. We then provide a set of questions designed to elicit the information that a datasheet for a dataset might contain, as well as a workflow for dataset creators to use when answering these questions. We conclude with a summary of the impact to date of datasheets for datasets and a discussion of implementation challenges and avenues for future work.

Objectives. Datasheets for datasets are intended to address the needs of two key stakeholder groups: dataset creators and dataset consumers. For dataset creators, the primary objective is to encourage careful reflection on the process of creating, distributing, and maintaining a dataset, including any underlying assumptions, potential risks or harms, and implica-

tions of use. For dataset consumers, the primary objective is to ensure they have the information they need to make informed decisions about using a dataset. Transparency on the part of dataset creators is necessary for dataset consumers to be sufficiently well informed that they can select appropriate datasets for their chosen tasks and avoid unintentional misuse.[a]

Beyond these two key stakeholder groups, datasheets for datasets may be valuable to policy makers, consumer advocates, investigative journalists, individuals whose data is included in datasets, and individuals who may be impacted by models trained or evaluated using datasets. They also serve a secondary objective of facilitating greater reproducibility of machine learning results: researchers and practitioners without access to a dataset may be able to use the information in its datasheet to create alternative datasets with similar characteristics.

Although we provide a set of questions designed to elicit the information a datasheet for a dataset might contain, these questions are not intended to be prescriptive. Indeed, we expect that datasheets will necessarily vary depending on factors such as the domain or existing organizational infrastructure and workflows. For example, some the questions are appropriate for academic researchers publicly releasing datasets for the purpose of enabling future research, but less relevant for product teams

» key insights

- **There are currently no industry standards for documenting machine learning datasets.**
- **Datasheets address this gap by documenting the contexts and contents of datasets: from their motivation, composition, collection process, and recommended uses.**
- **Datasheets for datasets can increase transparency and accountability within the machine learning community, mitigate unwanted biases in machine learning models, facilitate greater reproducibility of machine learning results, and help researchers and practitioners to choose the right dataset.**
- **Datasheets enable dataset creators to be intentional throughout the dataset creation process.**
- **Iterating on the design of datasheets with practitioners and legal experts helped improve the questions.**
- **Datasheets and other forms of data documentation are increasingly commonly released along with datasets.**

[a] We note that in some cases, the people creating a datasheet for a dataset may not be the dataset creators, as was the case with the example datasheets that we created as part of our development process.

IMAGE BY GOODSTUDIO

creating internal datasets for training proprietary models. As another example, Bender and Friedman[2] outline a proposal similar to datasheets for datasets specifically intended for language-based datasets. Their questions may be naturally integrated into a datasheet for a language-based dataset as appropriate.

We emphasize that the process of creating a datasheet is not intended to be automated. Although automated documentation processes are convenient, they run counter to our objective of encouraging dataset creators to carefully reflect on the process of creating, distributing, and maintaining a dataset.

Development Process

Here, we refined the questions and workflow provided over a period of approximately two years, incorporating many rounds of feedback.

First, leveraging our own experiences as researchers with diverse backgrounds working in different domains and institutions, we drew on our knowledge of dataset characteristics, unintentional misuse, unwanted societal biases, and other issues to produce an initial set of questions designed to elicit information about these topics. We then "tested" these questions by creating example datasheets for two widely used datasets: Labeled Faces in the Wild[16] and Pang and Lee's polarity dataset.[22] We chose these datasets in large part because their creators provided exemplary documentation, allowing us to easily find the answers to many of the questions. While creating these example datasheets, we found gaps in the questions, as well as redundancies and lack of clarity. We therefore refined the questions and distributed them to product teams in two major U.S.-based technology companies, in some cases helping teams to create datasheets for their datasets and observing where the questions did not achieve their intended objectives. Contemporaneously, we circulated an initial draft of this article to colleagues through social media and on arXiv (draft posted Mar. 23, 2018). Via these channels we received extensive comments from dozens of researchers, practitioners, and policy makers. We also worked with a team of lawyers to review the questions from a legal perspective.

We incorporated this feedback to yield the questions and workflow provided in the next section: We added and removed questions, refined the content of the questions, and reordered the questions to better match the key stages of the dataset life cycle. Based on our experiences with product teams, we reworded the questions to discourage yes/no answers, added a section on "Uses," and deleted a section on "Legal and Ethical Considerations." We found that product teams were more likely to answer questions about legal and ethical considerations if they were integrated into sections about the relevant stages of the dataset lifecycle rather than grouped together. Finally, following feedback from the team of lawyers, we removed questions that explicitly asked about compliance with regulations, and introduced factual questions intended to elicit relevant information about compliance without requiring dataset creators to make legal judgments.

Questions and Workflow

In this section, we provide a set of questions designed to elicit the information that a datasheet for a dataset might contain, as well as a workflow for dataset creators to use when answering these questions. The questions are grouped into sections that approximately match the key stages of the dataset lifecycle: motivation, composition, collection process, preprocessing/cleaning/labeling, uses, distribution, and maintenance. This grouping encourages dataset creators to reflect on the process of creating, distributing, and maintaining a dataset, and even alter this process in response to their reflection. We note that not all questions will be applicable to all datasets; those that do not apply should be skipped.

To illustrate how these questions might be answered in practice, we produced an appendix that includes an example datasheet for Pang and Lee's polarity dataset.[22] (The appendix is available online at https://dl.acm.org/doi/10.1145/3458723.) We answered some of the questions with "Unknown to the authors of the datasheet." This is because we did not create the dataset ourselves and could not find the answers to these questions in the available documentation. For an example of a datasheet that was created by the creators of the corresponding dataset, please see that of Cao and Daumé.[6,b] We note that even dataset creators may be unable to answer all the questions provided here. We recommend answering as many questions as possible rather than skipping the datasheet creation process entirely.

Motivation. The following questions are primarily intended to encourage dataset creators to clearly articulate their reasons for creating the dataset and to promote transparency about funding interests. The latter may be particularly relevant for datasets created for research purposes.

1. **For what purpose was the dataset created?** Was there a specific task in mind? Was there a specific gap that needed to be filled? Please provide a description.
2. **Who created the dataset (for example, which team, research group) and on behalf of which entity (for example, company, institution, organization)?**
3. **Who funded the creation of the dataset?** If there is an associated grant, please provide the name of the grantor and the grant name and number.
4. **Any other comments?**

Composition. Dataset creators should read through these questions prior to any data collection and then provide answers once data collection is complete. Most of the questions here are intended to provide dataset consumers with the information they need to make informed decisions about using the dataset for their chosen tasks. Some of the questions are designed to elicit information about compliance with the EU's General Data Protection Regulation (GDPR) or comparable regulations in other jurisdictions.

Questions that apply only to datasets that relate to people are grouped together at the end of the

[b] See https://github.com/TristaCao/into_inclusivecoref/blob/master/GICoref/datasheet-gicoref.md.

section. We recommend taking a broad interpretation of whether a dataset relates to people. For example, any dataset containing text that was written by people relates to people.

5. **What do the instances that comprise the dataset represent (for example, documents, photos, people, countries)?** Are there multiple types of instances (for example, movies, users, and ratings; people and interactions between them; nodes and edges)? Please provide a description.

6. **How many instances are there in total (of each type, if appropriate)?**

7. **Does the dataset contain all possible instances or is it a sample (not necessarily random) of instances from a larger set?** If the dataset is a sample, then what is the larger set? Is the sample representative of the larger set (for example, geographic coverage)? If so, please describe how this representativeness was validated/verified. If it is not representative of the larger set, please describe why not (for example, to cover a more diverse range of instances, because instances were withheld or unavailable).

8. **What data does each instance consist of?** "Raw" data (for example, unprocessed text or images) or features? In either case, please provide a description.

9. **Is there a label or target associated with each instance?** If so, please provide a description.

10. **Is any information missing from individual instances?** If so, please provide a description, explaining why this information is missing (for example, because it was unavailable). This does not include intentionally removed information, but might include, for example, redacted text.

11. **Are relationships between individual instances made explicit (for example, users' movie ratings, social network links)?** If so, please describe how these relationships are made explicit.

12. **Are there recommended data splits (for example, training, development/validation, testing)?** If so, please provide a description of these splits, explaining the rationale behind them.

13. **Are there any errors, sources of noise, or redundancies in the dataset?** If so, please provide a description.

Datasheets for datasets have the potential to increase transparency and accountability within the ML community, mitigate unwanted societal biases in ML models, facilitate greater reproducibility of ML results, and help researchers and practitioners select more appropriate datasets for their chosen tasks.

14. **Is the dataset self-contained, or does it link to or otherwise rely on external resources (for example, websites, tweets, other datasets)?** If it links to or relies on external resources, a) are there guarantees that they will exist, and remain constant, over time; b) are there official archival versions of the complete dataset (that is, including the external resources as they existed at the time the dataset was created); c) are there any restrictions (for example, licenses, fees) associated with any of the external resources that might apply to a dataset consumer? Please provide descriptions of all external resources and any restrictions associated with them, as well as links or other access points, as appropriate.

15. **Does the dataset contain data that might be considered confidential (for example, data that is protected by legal privilege or by doctor–patient confidentiality, data that includes the content of individuals' non-public communications)?** If so, please provide a description.

16. **Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety?** If so, please describe why.

If the dataset does not relate to people, you may skip the remaining questions in this section.

17. **Does the dataset identify any subpopulations (for example, by age, gender)?** If so, please describe how these subpopulations are identified and provide a description of their respective distributions within the dataset.

18. **Is it possible to identify individuals (that is, one or more natural persons), either directly or indirectly (that is, in combination with other data) from the dataset?** If so, please describe how.

19. **Does the dataset contain data that might be considered sensitive in any way (for example, data that reveals race or ethnic origins, sexual orientations, religious beliefs, political opinions or union memberships, or locations; financial or health data; biometric or genetic data; forms of government identification, such as social security numbers; criminal history)?** If so, please provide a description.

20. **Any other comments?**

Collection process. As with the questions in the previous section, dataset creators should read through these questions prior to any data collection to flag potential issues and then provide answers once collection is complete. In addition to the goals outlined earlier, the following questions are designed to elicit information that may help researchers and practitioners to create alternative datasets with similar characteristics. Again, questions that apply only to datasets that relate to people are grouped together at the end of the section.

21. **How was the data associated with each instance acquired? Was the data directly observable (for example, raw text, movie ratings), reported by subjects (for example, survey responses), or indirectly inferred/derived from other data (for example, part-of-speech tags, model-based guesses for age or language)?** If the data was reported by subjects or indirectly inferred/derived from other data, was the data validated/verified? If so, please describe how.

22. **What mechanisms or procedures were used to collect the data (for example, hardware apparatuses or sensors, manual human curation, software programs, software APIs)?** How were these mechanisms or procedures validated?

23. **If the dataset is a sample from a larger set, what was the sampling strategy (for example, deterministic, probabilistic with specific sampling probabilities)?**

24. **Who was involved in the data collection process (for example, students, crowdworkers, contractors) and how were they compensated (for example, how much were crowdworkers paid)?**

25. **Over what timeframe was the data collected?** Does this timeframe match the creation timeframe of the data associated with the instances (for example, recent crawl of old news articles)? If not, please describe the timeframe in which the data associated with the instances was created.

26. **Were any ethical review processes conducted (for example, by an institutional review board)?** If so, please provide a description of these review processes, including the outcomes, as well as a link or other access point to any supporting documentation.

The process of creating a datasheet is not intended to be automated. Although automated documentation processes are convenient, they run counter to our objective of encouraging dataset creators to carefully reflect on the process of creating, distributing, and maintaining a dataset.

If the dataset does not relate to people, you may skip the remaining questions in this section.

27. **Did you collect the data from the individuals in question directly, or obtain it via third parties or other sources (for example, websites)?**

28. **Were the individuals in question notified about the data collection?** If so, please describe (or show with screenshots or other information) how notice was provided, and provide a link or other access point to, or otherwise reproduce, the exact language of the notification itself.

29. **Did the individuals in question consent to the collection and use of their data?** If so, please describe (or show with screenshots or other information) how consent was requested and provided, and provide a link or other access point to, or otherwise reproduce, the exact language to which the individuals consented.

30. **If consent was obtained, were the consenting individuals provided with a mechanism to revoke their consent in the future or for certain uses?** If so, please provide a description, as well as a link or other access point to the mechanism (if appropriate).

31. **Has an analysis of the potential impact of the dataset and its use on data subjects (for example, a data protection impact analysis) been conducted?** If so, please provide a description of this analysis, including the outcomes, as well as a link or other access point to any supporting documentation.

32. **Any other comments?**

Preprocessing/cleaning/labeling. Dataset creators should read through these questions prior to any preprocessing, cleaning, or labeling and then provide answers once these tasks are complete. The questions in this section are intended to provide dataset consumers with the information they need to determine whether the "raw" data has been processed in ways that are compatible with their chosen tasks. For example, text that has been converted into a "bag-of-words" is not suitable for tasks involving word order.

33. **Was any preprocessing/clean-**

ing/labeling of the data done (for example, discretization or bucketing, tokenization, part-of-speech tagging, SIFT feature extraction, removal of instances, processing of missing values)? If so, please provide a description. If not, you may skip the remaining questions in this section.

34. **Was the "raw" data saved in addition to the preprocessed/cleaned/labeled data (for example, to support unanticipated future uses)?** If so, please provide a link or other access point to the "raw" data.

35. **Is the software that was used to preprocess/clean/label the data available?** If so, please provide a link or other access point.

36. **Any other comments?**

Uses. The following questions are intended to encourage dataset creators to reflect on the tasks for which the dataset should and should not be used. By explicitly highlighting these tasks, dataset creators can help dataset consumers to make informed decisions, thereby avoiding potential risks or harms.

37. **Has the dataset been used for any tasks already?** If so, please provide a description.

38. **Is there a repository that links to any or all papers or systems that use the dataset?** If so, please provide a link or other access point.

39. **What (other) tasks could the dataset be used for?**

40. **Is there anything about the composition of the dataset or the way it was collected and preprocessed/cleaned/labeled that might impact future uses?** For example, is there anything that a dataset consumer might need to know to avoid uses that could result in unfair treatment of individuals or groups (for example, stereotyping, quality of service issues) or other risks or harms (for example, legal risks, financial harms)? If so, please provide a description. Is there anything a dataset consumer could do to mitigate these risks or harms?

41. **Are there tasks for which the dataset should not be used?** If so, please provide a description.

42. **Any other comments?**

Distribution. Dataset creators should provide answers to these questions prior to distributing the dataset either internally within the entity on behalf of which the dataset was created or externally to third parties.

43. **Will the dataset be distributed to third parties outside of the entity (for example, company, institution, organization) on behalf of which the dataset was created?** If so, please provide a description.

44. **How will the dataset be distributed (for example, tarball on website, API, GitHub)?** Does the dataset have a digital object identifier (DOI)?

45. **When will the dataset be distributed?**

46. **Will the dataset be distributed under a copyright or other intellectual property (IP) license, and/or under applicable terms of use (ToU)?** If so, please describe this license and/or ToU, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms or ToU, as well as any fees associated with these restrictions.

47. **Have any third parties imposed IP-based or other restrictions on the data associated with the instances?** If so, please describe these restrictions, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms, as well as any fees associated with these restrictions.

48. **Do any export controls or other regulatory restrictions apply to the dataset or to individual instances?** If so, please describe these restrictions, and provide a link or other access point to, or otherwise reproduce, any supporting documentation.

49. **Any other comments?**

Maintenance. As with the previous questions, dataset creators should provide answers to these questions prior to distributing the dataset. The questions in this section are intended to encourage dataset creators to plan for dataset maintenance and communicate this plan to dataset consumers.

50. **Who will be supporting/hosting/maintaining the dataset?**

51. **How can the owner/curator/manager of the dataset be contacted (for example, email address)?**

52. **Is there an erratum?** If so, please provide a link or other access point.

53. **Will the dataset be updated (for example, to correct labeling errors, add new instances, delete instances)?** If so, please describe how often, by whom, and how updates will be communicated to dataset consumers (for example, mailing list, GitHub)?

54. **If the dataset relates to people, are there applicable limits on the retention of the data associated with the instances (for example, were the individuals in question told that their data would be retained for a fixed period of time and then deleted)?** If so, please describe these limits and explain how they will be enforced.

55. **Will older versions of the dataset continue to be supported/hosted/maintained?** If so, please describe how. If not, please describe how its obsolescence will be communicated to dataset consumers.

56. **If others want to extend/augment/build on/contribute to the dataset, is there a mechanism for them to do so?** If so, please provide a description. Will these contributions be validated/verified? If so, please describe how. If not, why not? Is there a process for communicating/distributing these contributions to dataset consumers? If so, please provide a description.

57. **Any other comments?**

Impact and Challenges

Since circulating an initial draft of this article in March 2018, datasheets for datasets have already gained traction in a number of settings. Academic researchers have adopted our proposal and released datasets with accompanying datasheets.[7,10,23,26] Microsoft, Google, and IBM have begun to pilot datasheets for datasets internally within product teams. Researchers at Google published follow-up work on *model cards* that document machine learning models[20] and released a *data card* (a lightweight version of a datasheet) along with the Open Images dataset.[17] Researchers at IBM proposed *factsheets*[14] that document various characteristics of AI services, including whether the datasets used to develop the services are accompanied with datasheets. The Data Nutrition Project incorporated some of the questions provided in the previous section into the latest release of their Dataset Nutrition Label.[9] Finally, the Partnership on AI, a multi-stakeholder organization focused on studying and formulating best practices for developing and deploying AI technologies, is working on industry-wide documentation guidance that builds on datasheets for datasets, model cards, and factsheets.[c]

These initial successes have also revealed implementation challenges that may need to be addressed to support wider adoption. Chief among them is the need for dataset creators to modify the questions and workflow provided earlier based on their existing organizational infrastructure and workflows. We also note that the questions and workflow may pose problems for dynamic datasets. If a dataset changes only infrequently, we recommend accompanying updated versions with updated datasheets.

Datasheets for datasets do not provide a complete solution to mitigating unwanted societal biases or potential risks or harms. Dataset creators cannot anticipate every possible use of a dataset, and identifying unwanted societal biases often requires additional labels indicating demographic information about individuals, which may not be available to dataset creators for reasons including those individuals' data protection and privacy.[15]

When creating datasets that relate to people, and hence their accompanying datasheets, it may be necessary for dataset creators to work with experts in other domains such as anthropology, sociology, and science and technology studies. There are complex and contextual social, historical, and geographical factors that influence how best to collect data from individuals in a manner that is respectful.

Finally, creating datasheets for datasets will necessarily impose overhead on dataset creators. Although datasheets may reduce the amount of time that dataset creators spend answering one-off questions about datasets, the process of creating a datasheet will always take time, and organizational infrastructure and workflows—not to mention incentives—will need to be modified to accommodate this investment.

Despite these implementation challenges, there are many benefits to creating datasheets for datasets. In addition to facilitating better communication between dataset creators and dataset consumers, datasheets provide an opportunity for dataset creators to distinguish themselves as prioritizing transparency and accountability. Ultimately, we believe that the benefits to the machine learning community outweigh the costs.

Acknowledgments

We thank P. Bailey, E. Bender, Y. Bengio, S. Bird, S. Brown, S. Bowles, J. Buolamwini, A. Casari, E. Charran, A. Couillault, L. Dauterman, L. Dodds, M. Dudík, M. Ekstrand, N. Elhadad, M. Golebiewski, N. Gonsalves, M. Hansen, A. Hickl, M. Hoffman, S. Hoogerwerf, E. Horvitz, M. Huang, S. Kallumadi, E. Kamar, K. Kenthapadi, E. Kiciman, J. Krones, E. Learned-Miller, L. Lee, J. Leidner, R. Mauceri, B. Mcfee, E. McReynolds, B. Micu, M. Mitchell, S. Mudnal, B. O'Connor, T. Padilla, B. Pang, A. Parikh, L. Peets, A. Perina, M. Philips, B. Place, S. Rao, J. Ren, D. Van Riper, A. Roth, C. Rudin, B. Shneiderman, B. Srivastava, A. Teredesai, R. Thomas, M. Tomko, P. Tziachris, M. Whittaker, H. Wolters, A. Yeo, L. Zhang, and the attendees of the Partnership on AI's April 2019 ABOUT ML workshop for valuable feedback.

[c] https://www.partnershiponai.org/about-ml/

References

1. Andrews, D., Bonta, J., and Wormith, J. The recent past and near future of risk and/or need assessment. *Crime & Delinquency 52*, 1 (2006), 7–27.
2. Bender, E. and Friedman, B. Data Statements for Natural Language Processing: Toward Mitigating System Bias and Enabling Better Science. *Trans. of the Assoc. for Computational Linguistics 6* (2018), 587–604.
3. Bhardwaj, A. et al. DataHub: Collaborative data science & dataset version management at scale. *CoRR* abs/1409.0798 (2014).
4. Bolukbasi, T., Chang, K., Zou, J., Saligrama, V., and Kalai, A. Man is to computer programmer as woman is to homemaker? Debiasing Word Embeddings. In *Advances in Neural Information Processing Systems* (2016).
5. Buolamwini, J. and Gebru, T. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Proceedings of the Conf. on Fairness, Accountability, and Transparency* (2018). 77–91.
6. Cao, Y. and Daumé, H. Toward gender-inclusive coreference resolution. In *Proceedings of the Conf. of the Assoc. for Computational Linguistics* (2020). abs/1910.13913.
7. Cao, Y. and Daumé, H. Toward gender-inclusive coreference resolution. In *Proceedings of the Conf. of the Assoc. for Computational Linguistics* (2020).
8. Cheney, J., Chiticariu, L., and Tan, W. Provenance in databases: Why, how, and where. *Foundations and Trends in Databases 1*, 4 (2009), 379–474.
9. Chmielinski, K. et al. The dataset nutrition label (2nd Gen): Leveraging context to mitigate harms in artificial intelligence. In *NeurIPS Workshop on Dataset Curation and Security*, 2020.
10. Choi, E. et al. QuAC: Question answering in context. In *Proceedings of the 2018 Conf. on Empirical Methods in Natural Language Processing*.
11. Chui, G. Project will use AI to prevent or minimize electric grid failures, 2017.
12. Dastin, J. Amazon scraps secret AI recruiting tool that showed bias against women, 2018; https://reut.rs/3imOH4d.
13. Garvie, C., Bedoya, A., and Frankle, J. *The Perpetual Line-Up: Unregulated Police Face Recognition in America.* Georgetown Law, Center on Privacy & Technology, Washington, D.C., 2016.
14. Hind, M. et al. Varshney. Increasing trust in AI services through supplier's declarations of conformity. *CoRR* abs/1808.07261 (2018).
15. Holstein, K., Vaughan, J., Daumé, H, Dudík, M., and Wallach, H. Improving fairness in machine learning systems: What do industry practitioners need? In *Proceedings of 2019 ACM CHI Conf. on Human Factors in Computing Systems*.
16. Huang, G., Ramesh, M., Berg, T., and Learned-Miller, E. Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments. Technical Report 07-49. University of Massachusetts Amherst, 2007.
17. Krasin, I. et al. OpenImages: A public dataset for large-scale multi-label and multi-class image classification, 2017.
18. Lin, T. The new investor. *UCLA Law Review 60* (2012), 678.
19. Mann, G. and O'Neil, C. Hiring Algorithms Are Not Neutral, 2016; https://hbr.org/2016/12/hiring-algorithms-are-not-neutral.
20. Mitchell, M. et al. Model cards for model reporting. In *Proceedings of the Conf. on Fairness, Accountability, and Transparency* (2019). 220–229.
21. O'Connor, M. How AI Could Smarten Up Our Water System, 2017.
22. Pang, B. and Lee, L. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Assoc. for Computational Linguistics*. 2004, 271.
23. Seck, I., Dahmane, K., Duthon, P., and Loosli, G. Baselines and a datasheet for the Cerema AWP dataset. *CoRR* abs/1806.04016 (2018). http://arxiv.org/abs/1806.04016
24. Doha Supply Systems. Facial Recognition, 2017.
25. World Economic Forum Global Future Council on Human Rights 2016–2018. How to Prevent Discriminatory Outcomes in Machine Learning; 2018. https://www.weforum.org/whitepapers/how-to-prevent-discriminatory-outcomes-inmachine-learning.
26. Yagcioglu, S., Erdem, A., Erdem, E., and Ikizler-Cinbis, N. RecipeQA: A challenge dataset for multimodal comprehension of cooking recipes. In *Proceedings of the 2018 Conf. on Empirical Methods in Natural Language Processing*.

Timnit Gebru is founder of DAIR Institute, Palo Alto, CA, USA.

Jamie Morgenstern is an assistant professor at the University of Washington, Seattle, WA, USA.

Briana Vecchione is a Ph.D. student at Cornell University, Ithaca, NY, USA.

Jennifer Wortman Vaughan is Senior Principal Researcher at Microsoft Research, New York, NY, USA.

Hanna Wallach is Partner Research Manager at Microsoft Research, New York, NY, USA.

Hal Daumé III is Senior Principal Researcher at Microsoft Research and a professor at the University of Maryland, College Park, MD, USA.

Kate Crawford is Senior Principal Researcher at Microsoft Research, and Research Professor at USC Annenberg, CA, USA.

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

Watch the authors discuss this work in the exclusive *Communications* video. https://cacm.acm.org/videos/datasheets-for-datasets

research highlights

DOI:10.1145/3491194

To view the accompanying paper, visit doi.acm.org/10.1145/3491196

Technical Perspective
Cooking Up a Solution to Microwave Heat Distribution

By Fadel Adib

WHAT IS COMMON between microwaved popcorn and WiFi networks? They both require microwave signals with frequencies around 2.45GHz. Indeed, research in the wireless networking community has demonstrated that microwave ovens, when turned on, can and do interfere with nearby WiFi networks and degrade their throughput. Thankfully, over the past two decades, the shielding of microwave ovens has improved, and WiFi networks have become smarter in combatting interference. So, it seemed the competition for airspace between microwave ovens and WiFi is from a bygone era until recently, when microwave ovens made a comeback to the wireless and mobile networking community.

Surprisingly, this time around, microwave ovens did not come back to interfere with WiFi networks. Rather, researchers figured out a way to program microwave ovens to heat food more evenly. We have all faced situations when a microwaved meal turned out unevenly heated, undercooked, or overcooked. Microwave oven designers try to combat this by using turntables and placing reflectors inside the oven to distribute the microwave energy when heating food more evenly. However, these approaches are inherently limited when it comes to meals that combine different types of food (vegetables, rice, meat), each of which requires different amounts of time to warm up without burning or overcooking.

So how can wireless research help? Over the past two decades, the wireless networking community has made significant strides in understanding the physical layer of WiFi networks by playing around with software-defined radios. This understanding has resulted in highly effective solutions, which have made it to state-of-the-art WiFi standards, including WiFi 6 (802.11ax). Much of these improvements are driven by the ability of a WiFi sender and receiver to exchange information about the wireless channel to better direct their energy toward each other, thus driving throughput gains through a feedback mechanism (specifically, by sharing channel state information). Contrast this to microwave ovens that have no feedback mechanism and still operate blindly with respect to the food in the oven.

> **By measuring the amount of glow and its distribution across each tiny lamp, the microwave oven can automatically sense heat gradients and control its turntable to heat food more evenly.**

The following paper makes an exciting leap by introducing a similar feedback mechanism to microwave heating through a technique the authors call software-defined cooking. At the core of this technique is an ability to directly measure the amount of heat delivered at different areas in a microwave oven and feed this information back to a turntable controller. To do this, the authors place an array of tiny neon lamps on the turntable; these lamps power up using the electromagnetic energy emitted by the microwave and glow in proportion to this energy. By measuring the amount of glow and its distribution across each of these tiny lamps, the microwave oven can automatically sense heat gradients and control its turntable to more evenly heat food placed in it. The authors built on this core idea to realize a system that can monitor temperature gradients even when the neon lights are in non-line-of-sight with respect to an external camera. They demonstrated how one can use their technique to heat food more evenly or heat different regions of a meal according to a recipe.

While there are many challenges before a moonshot idea like the one proposed in this paper makes it to a commercial product, the authors present an exciting direction to close the loop on microwave oven heating. It also opens the door to adopting various sophisticated techniques from wireless networking—such as MIMO beamforming, precoding, and sensing—to software-defined cooking. Perhaps future microwave ovens may even fuse sensed information from antennas and cameras inside the oven to execute highly sophisticated recipes. For now, I invite you to munch on a perfectly popped bag of popcorn as you read this paper and envision how learnings from wireless networking can help improve the design of microwave ovens. C

Fadel Adib is an associate professor at MIT, Cambridge, MA, USA, as well as Doherty Chair in Ocean Utilization and Founding Director of the Signal Kinetics Research Group.

Copyright held by author.

DOI:10.1145/3491196

Software-Defined Cooking Using a Microwave Oven

By Haojian Jin, Jingxian Wang, Swarun Kumar, and Jason Hong

Abstract

Despite widespread popularity, today's microwave ovens are limited in their cooking capabilities, given that they heat food blindly, resulting in a nonuniform and unpredictable heating distribution. We present software-defined cooking (SDC), a low-cost closed-loop microwave oven system that aims to heat food in a software-defined thermal trajectory. SDC achieves this through a novel high-resolution heat sensing and actuation system that uses microwave-safe components to augment existing microwaves. SDC first senses the thermal gradient by using arrays of neon lamps that are charged by the electromagnetic (EM) field a microwave produces. SDC then modifies the EM-field strength to desired levels by accurately moving food on a programmable turntable toward sensed hot and cold spots. To create a more skewed arbitrary thermal pattern, SDC further introduces two types of programmable accessories: A microwave shield and a susceptor. We design and implement one experimental test bed by modifying a commercial off-the-shelf microwave oven. Our evaluation shows that SDC can programmatically create temperature deltas at a resolution of 21°C with a spatial resolution of 3 cm without the programmable accessories, and 183°C with them. We further demonstrate how an SDC-enabled microwave can be enlisted to perform unexpected cooking tasks: Cooking meat and fat in bacon discriminatively and heating milk uniformly.

1. INTRODUCTION

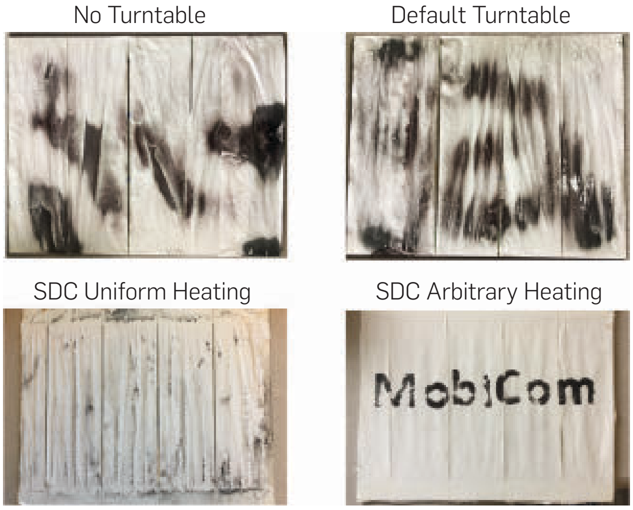
Since the introduction of microwaves to the consumer market in the 1970s, they have seen widespread adoption and are today the third most popular domestic food heating method (after baking and grilling).[13] Indeed, the original patents for the microwave by Raytheon Inc. in the late 1940s envisioned a universal food cooking instrument for all kinds of food ranging from meat to fish.[1] While microwaves have revolutionized the kitchen since their inception, today's consumer microwaves are mainly used as blunt heating appliances (e.g., reheating pizzas) rather than precise cooking instruments (e.g., cooking steak). The potential of microwave cooking is limited by the fact that today's microwaves heat food blindly, resulting in a nonuniform and unpredictable heating distribution.[18]

We present software-defined cooking (SDC), a novel low-cost closed-loop system that can augment existing consumer microwaves to sense and control heating at a fine-grained resolution, all using microwave-safe components within the chamber. SDC's design can unlock numerous programmable heating opportunities (see Figure 1). For example, when microwaving liquids (e.g., milk, baby formula), one need not worry about uneven heating that may scald the mouth or destroy nutrients—a reason why microwaves should not be used to heat formula despite their convenience. Further, SDC can enable fine-grained forms of cooking. For example, the Maillard reaction occurs when searing meats and pan-frying vegetables are an important part of the flavor of these cooked foods. With SDC, it is possible to execute a computer-generated Maillard reaction pattern that discriminatively heats the food surface.

At the heart of SDC is a novel approach both to sense and control heat at different points in space within the microwave chamber. SDC senses heat using the same phenomenon that produces heat in the first place: The electromagnetic (EM) field. SDC aims to measure the amplitude of the EM field at any given point. Although battery-powered sensors exist to sense both heat and EM fields, SDC must do so using only microwave-safe components, which excludes typical commercial batteries.[6]

SDC achieves this by relying on the fact that the EM field within the microwave is a natural source of energy. This means that one can simply power the sensor of the EM field by the EM field itself. SDC uses tiny radio frequency (RF)-powered

Figure 1. Results of microwaving wet thermal paper (black spots indicate high heat). (Top) A traditional microwave without/with a turntable. A turntable can mitigate uneven heating, but cold/hot spots remain. (Bottom) SDC, for uniform heating (fewer black spots show heat is spread uniformly) or heating to write "MobiCom." We use patterned susceptors and let SDC ensure the text area has been heated in hot spots.


No Turntable

Default Turntable

SDC Uniform Heating

SDC Arbitrary Heating

The original version of this paper was published in *Proceedings of the 25th Annual International Conference on Mobile Computing and Networking*, 2019, ACM.

neon lights (see Figure 2) that glow in response to the EM field within the microwave. Specifically, the oscillating microwave results in a potential difference (of a few 100 V to a few kV) between two electrodes within each light bulb. Due to the potential difference, electrons are accelerated away from the cathode and give rise to collisions with the neon gas atoms or molecules, which will emit a characteristic glow in proportion to the amplitude of the field. Neon lights are inexpensive, compact, and produce minimal disruption to the EM field itself—meaning that they can be tightly packed at key locations around the chamber to sense the EM-field amplitude at high accuracy. Given that the neon lights may be obstructed from view due to the food placed in the microwave, we run optical fibers made of microwave-safe glass that carry the light signals outside the chamber to be sensed by a camera (see Figure 3). Section 4 describes our framework to fuse measurements from this hardware with infrared (IR) cameras to estimate current and future spatial temperature distributions of the food.

Figure 2. We place neon lights with different lengths of wire extensions (red: 0 mm, green: 7.5 mm, and blue: 15 mm) under the turntable. We then measure the percentage of glowing time to quantify the sensitivity.

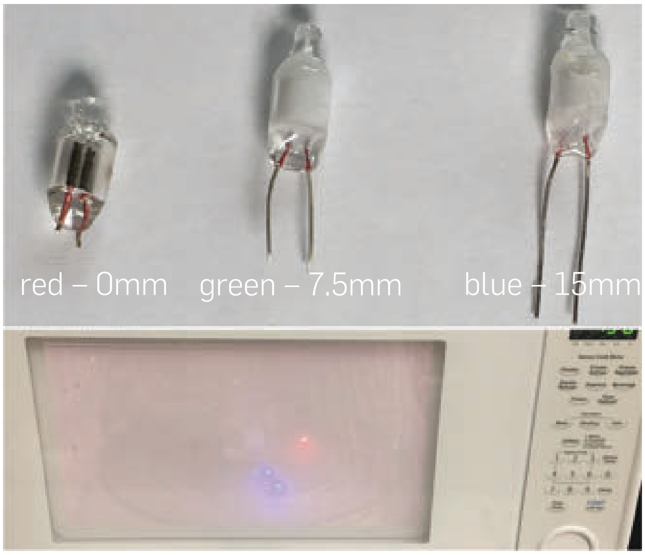
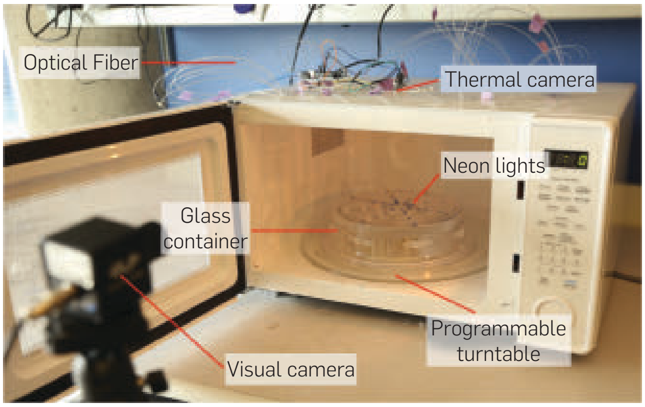


Figure 3. SDC's Hardware.



Upon sensing food heating, SDC controls heat according to the user-specified thermal recipe by building microwave shields that protect regions of the food that must not be overcooked (e.g., for meat). SDC achieves this through small metallic spheres placed within the microwave at key locations. Although conventional wisdom says that one must not place metal in a microwave, RF propagation is more nuanced. Specifically, metallic surfaces within the microwave only produce energetic sparks at sharp edges, found in most kitchen utensils and bowls. Metallic spheres by definition do not have edges and are thus microwave-safe.[16] SDC carefully packs metallic spheres at specific regions of the microwave to minimize RF energy transfer at these regions. Section 5 shows how SDC accurately rotates the turntable to guide parts of the food that must not be overcooked toward these regions.

We implement a prototype of SDC by modifying a commercial Microwave oven (Sharp SMC1441CW). We place neon lamp arrays inside the microwave oven cavity and use a camera outside the cavity to monitor lamp flashes conducted via fiber-optic cables. We replace the coarse turntable motor with a step motor controllable via an Arduino board. Given a desired heating distribution pattern, SDC recommends the initial position where the user should place the food. During the heating, SDC continuously senses the real-time EM-field strength around the food and adjusts the actuation plan. Figure 3 illustrates the basic hardware setup of SDC. We conducted detailed experiments to evaluate SDC's sensing, uniform heating, and planned heating capabilities. Our experiments reveal:

- SDC can improve thermal heating uniformity by 633% compared to commercial microwaves with turntables.
- SDC can create an arbitrary temperature delta of 183°C with a resolution of 3 cm.
- We demonstrate SDC in performing two unconventional cooking tasks: Cooking bacon and warming milk for an infant.

Contributions: SDC is a novel redesign of the microwave oven that both senses and actuates the EM field at fine-grained spatial resolution. SDC introduces programmable RF-powered neon lights whose signals are conducted by optic fibers that are microwave-safe to sense the EM-field distribution. SDC then adjusts the spatial heat distribution within the microwave chamber by moving food carefully around hot and cold spots predesigned using microwave-safe accessories. We built a prototype implementation of SDC by modifying an existing commercial microwave, and found that it had an accuracy of 7°C–10°C with respect to a heating recipe.

2. MICROWAVE HEATING 101

In a microwave, water, fat, and other electric dipoles in the food will absorb energy from the microwaves in a process called dielectric heating.[21] Namely, when an electric field is applied, the bipolar molecules tend to behave like microscopic magnets and attempt to align themselves with the field. When the electrical field changes millions of times per second (e.g., 2450 million times per second for 2.45 GHz microwave signals), these molecular magnets are unable to keep up in the presence of forces acting to slow them. This resistance to the rapid movement of the bipolar molecules creates friction and results in heat dissipation in the material exposed to the microwave radiation.

Although strong direct microwave radiation can burn human body tissue as well as electronic devices, the cooking chamber works as a Faraday cage to significantly attenuate waves escaping the microwave chamber. The US federal emission standard[17] limits the amount of microwave leakage from an oven throughout its lifetime to 5 mW per square centimeter at approximately 2 inches from the oven surface (a safety factor of 10,000 or more below levels that may harm people[10]).

Once the microwave signals enter the metal cavity, they are effectively reflected by the metallic walls. Original and reflected waves resonate in the cavity and form standing waves,[19] which produce antinodes (heating hot spots) and nodes (heating cold spots). The EM fields are weak at nodes and therefore nothing cooks there. In contrast, EM fields at antinodes alternate at maximum amplitude to produce maximum heating. This is also the reason why microwaves have a rotating turntable so that the turntable moves food in and out of the hot spots to cook more uniformly.

3. SDC HEATING RECIPE

At its most basic, "cooking" means applying heat to food,[12] which can be specified in three main variables:

- The most important variable in cooking is the **temperature** of food, which will trigger different chemical reactions (e.g., protein denaturation, Maillard reaction, and caramelization). For instance, if we want to cook a steak at least rare, we need to heat the meat to a temperature between 55°C (the highest survival temperature for most bacteria) and 65.5°C (the denaturing temperature for the protein actin).
- **Time** is an important factor in both cooking food accurately and killing bacteria. For example, the standard food safety rule[4] provided by the Food and Drug Administration (FDA) states various time and temperature combinations: Heating at 55°C for 89 min can achieve a similar effect as heating at 62.2°C for 5 min to reduce Salmonella.
- **Space:** Different parts of food (e.g., meat vs. fat, egg white vs. yolk) may need to be cooked with different specifications to obtain optimal tasting food. SDC therefore aims to specify heating requirements for each spatial "pixel" of the food surface.

We envision that the future microwave heating recipes specify the desired thermal trajectory, that is, temperature vs. time, for each "pixel" of the food (see Figure 4 for an example steak recipe). This is precisely SDC's input, with its performance dictated by how closely it follows this specification.

We note that across all three common heating methods (convection, conduction, and radiation), food is cooked from the outside in, that is, the outer portions will warm up faster, and the heat conduction will heat the inner parts over time. So SDC focuses on controlling the surface temperature of food.

4. SDC'S HEAT SENSING

SDC's heat sensing aims to capture both the current temperature of food as well as the intent to heat. At first blush, one may assume that heat sensing can be readily achieved using a thermal camera that captures the current temperature of the food. By measuring thermal camera readings over time, one can make predictions about how food will heat in the future. Yet, thermal cameras have important limitations that limit a design that relies exclusively on them for heat sensing. First, the food thermal properties evolve on a time scale of seconds, and the carryover in cooking will continue heating even if the food is removed from the source of heat. However, thermal cameras often have limited refresh rates (<9 Hz) and modest accuracy (± 2°C). Therefore, thermal cameras only measure the effect of heating after-the-fact and cannot prevent undesired heating, often until the damage is already done. Thermal cameras are also limited to measuring heating on the surface of the food in direct line-of-sight. To mitigate this, SDC complements a thermal camera that senses current temperature with microwave-safe sensors that estimate future expected temperature (see Figure 5).

4.1. Sensing hardware design

SDC places an array of neon lights, each with a 5-mm diameter and 13-mm length, inside the microwave chamber to sense EM fields. A neon light (see Figure 2) is a miniature gas discharge lamp, which consists of a small glass capsule that contains a mixture of neon and other gases at low pressure as well as two electrodes (an anode and a cathode). During microwaving, the electrodes will couple with the EM field and act as antennas. The oscillating microwave applies a potential difference between two electrodes. Due to the potential

Figure 4. SDC heating recipe is a progression of desired temperature vs. time per pixel of food. For example, if we want to cook a steak at least rare, the ideal temperature is between 55°C and 65.5°C, so the process kills bacteria but avoid denaturing actin protein. Meat becomes tough and dry when cooked to higher temperatures.[12]

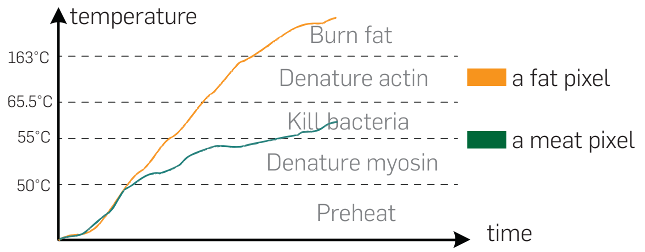
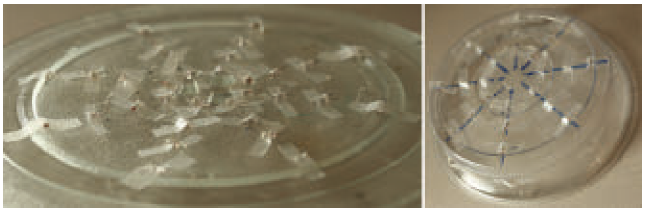


Figure 5. A turntable with 32 neon lights (left) and a plate cover with 32 neon lights (right).



difference, electrons are accelerated away from the cathode and give rise to collisions with the neon gas molecules, which will emit a characteristic glow. The brightness of the lamp is proportional to the EM-field strength at the placed location, and SDC leverages that brightness to measure the EM-field strength.

The glow of the light is sensed by a visible light camera outside the chamber to capture real-time EM-field strength (see Figure 6). However, neon lights are often blocked from direct view of the camera due to obstructions such as food on the turntable. To mitigate this, SDC conducts the light from the neon lamps to the camera using optic fibers (see Figure 7).

Is SDC's hardware microwave safe? Neon lights are microwave-safe because the metal electrodes are encapsulated with a glass capsule, and because the gas glow discharge can avoid energy accumulation. Each neon light consumes minimal microwave energy (≈19.5 mW), producing negligible interference to existing EM-field patterns. Glass optical fibers are also microwave-safe.

The cameras used in SDC are not affected by the microwave because they are placed outside the microwave oven. The holes created for the optical fibers and the thermal camera are smaller than the 1/20 wavelength of 2.4 GHz radio, so the chamber remains a Faraday cage. The leakage through holes is negligible. Indeed, many commercial microwaves have holes of similar dimensions to support the turntable or stirrer fan.

Programming EM sensitivity: Much like EM fields in radio communication, it is important to tune neon lights to the correct range of sensitivity to accurately sense EM-field strength. We define sensitivity of the neon light as the change in brightness for a given change in EM-field strength. A highly sensitive neon light may be saturated by a strong EM field and burn the antennas, whereas a poorly sensitive neon lamp may not light up under a low EM field.

Figure 6. The turntable inside a running oven. The brightness of a neon light is proportional to the EM-field strength at the placed location.

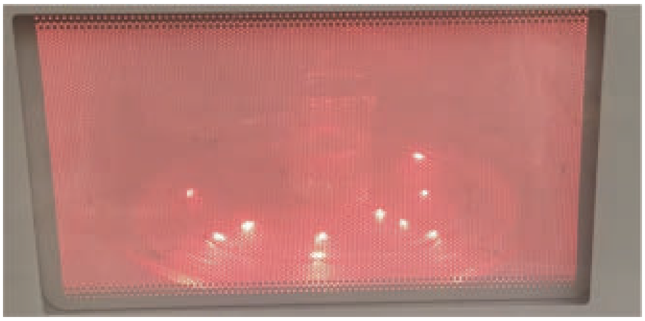
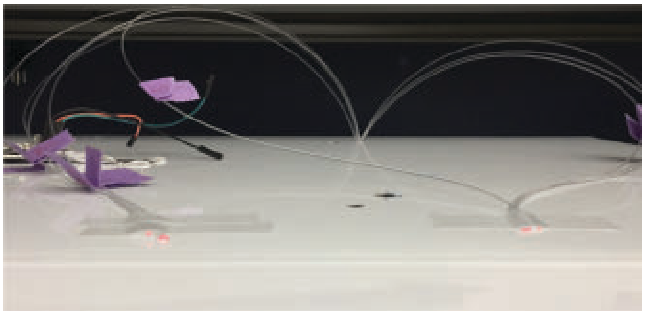


Figure 7. The optical fiber carries the signal outside of the chamber.



To find the right level of sensitivity, we tested neon lights with several different lengths of wire extensions (see Figure 2). Specifically, depending on the EM-field strength, a neon light may experience one of the following three states when running the microwave: Consistently off, flashing at various intervals, and consistently on. The flashing (flashing frequency) and consistently on (brightness) states offer more fine-grained resolution of the EM field than consistently off. An ideal neon light sensitivity would ensure that a good percentage of the neon lights in the oven are in the flashing and consistently on states. To tune the optimal antenna length (i.e., the wire extension length), we empirically tested various types of loads in the microwave and found that a wire extension of 8 mm achieves the desired sensitivity.

4.2. Modeling heat over time and space

Next, we describe how we measure the brightness and flashing frequency of strategically placed neon lamps, and use this data to model the current and future temperature of food over 3D space and time.

Creating a spatial heatmap: The visual camera, placed at the front of the microwave oven, captures the brightness of neon lights in a real-time video stream. SDC measures the brightness of the lamps every 0.1 seconds (i.e., 12 frames of a 120fps video stream). For each frame, SDC converts the image to grayscale, finds the pixels around the neon light or at the end of an optical fiber, and sums up the pixel values as the brightness score. Because the locations of the neon lights are known *a priori*, we interpolate the brightness at remaining locations using cubic-spline interpolation. We then map neon light brightness and flashing frequencies to EM-field strength empirically by comparing results from colocated neon lamps. This, coupled with spatial interpolation, allows us to generate a 3D EM-field intensity view within the microwave chamber. SDC can therefore estimate the EM-field strength, given a specific location at fine-grained spatial resolution.

Modeling temperature over space-time: As mentioned previously, SDC can measure the current temperature of the food surface by placing a thermal camera on the roof of the microwave oven to sense the food surface in a top-down view. However, using a thermal camera exclusively to model temperature has two limitations: (1) the camera only measures temperatures on the food surface in its direct field-of-view; (2) the camera only measures the current temperature and not future expected temperature.

SDC estimates future heat by integrating measurements of the EM field obtained from the neon lamps. Specifically, the heating of any given point in space of the food is proportional to the EM-field intensity of that location. This means that integrating the observed field intensity, while accounting for the rotation of food over time, can provide a robust estimate of its future temperature. Yet, two challenges remain in making this mapping accurate: (1) first, the temperature of the food for the very same EM field may change owing to the material composition of the food itself; (2) second, integrating EM-field intensities over time may cause errors to build up progressively as well.

SDC mitigates the limitations of both the thermal camera system and the EM-field estimation by combining them and obtaining the best of both worlds. Specifically, at each point in time we reconcile the integration of EM-field estimates with those of the thermal camera over space. We then use this to refine our model for mapping EM field to temperature. We repeat this process over time to continuously avoid any drift of our EM field to temperature mapping, as well as accounting for material properties.

5. SDC'S HEAT ACTUATION

Today's microwave ovens actuate the heating process in a crude manner. The oven turntable rotates the food blindly without any precise control. The magnetron, the heating engine of the microwave oven, achieves power control by periodically turning itself on and off. SDC augments the existing blind actuation hardware into a closed-loop control system by incorporating the results of heat sensing.

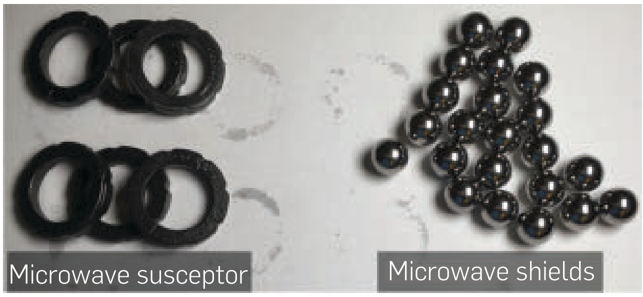
5.1. Actuation hardware

Smart turntable: We modify the default turntable inside a commercial microwave oven. More specifically, we replaced the motor with a low-cost stepper motor, and 3D printed a plastic coupler between the motor head and the glass platform to enable precise control. We also connected the magnetron to an Arduino and programmatically sent pulse-width modulation signals to control the ON/OFF of the magnetron. Rotating the food around can manipulate the heat pattern mildly (e.g., uniform heating), but it is insufficient to create an arbitrary heat pattern.

Programmable accessories: To achieve a more skewed heating pattern, we developed programmable accessories (see Figure 8) that leverage the reflective property of microwave heating, redirecting energy toward desired locations and shielding undesired locations, to achieve an arbitrary heating capability.

To minimize RF energy transfer at specific regions, we installed a horizontal glass plane above the turntable and carefully packed metallic spheres (see Figure 8 right) correspondingly. Metallic spheres by definition do not have edges and are therefore microwave-safe. These metallic spheres effectively form a microwave mirror to reflect microwave energy at the specific region.

Figure 8. Programmable microwave accessories.



The most common dielectric dipole in the food is water, so microwave heating rarely achieves temperature beyond the boiling point of water. However, some important food chemical reactions occur well above water's boiling point, such as Maillard reactions and caramelization. We introduce microwave susceptors to address this limitation. Materials like silicon carbide (see Figure 8 left) can effectively absorb microwave energy inside the oven and reach 200+°C within 1 min microwaving. Attaching silicon carbide to the food surface can then trigger desired high-heat reactions.

5.2. Recipe and actuation representation

Having developed programmable actuation hardware, this section formally states the actuation optimization problem that attempts to heat the food in accordance with the input heating recipe.

Heating recipe: An SDC heating recipe will specify the desired temperature trajectory and duration for each part at different temperatures. Mathematically, we can formulate the recipe as follows. Let us imagine that the food surface is divided into a set of discrete pixels. Given *n* pixels $B = \{B_1, B_2, \ldots B_n\}$ on the surface of the food, the 3D coordination of the pixels is $\{x_i, y_i, z_i\}$, where $i \in \{1, 2, \ldots n\}$. The recipe is a mapping function *f* that maps the pixels and the timestamps to desired temperatures throughout the *D* min cooking journey:

$$f(B_i, j) = p_{ij}, \qquad i \in \{1,2,\ldots n\} \quad 0 < j < D \tag{1}$$

where *j* denotes the timestamp since start of the cooking process, and p_{ij} refers to the desired temperature for *i*-th pixel at the timestamp *j*.

SDC's optimization problem: Our goal of the smart turntable is to find a rotation plan $S*$ that can move food in and out of these hot and cold spots as needed to cook food according to the desired heat trajectory *P*, which contains collection of desired temperatures p_{ij} across the space and time. SDC defines a rotation plan *S* as a sequence of angle-duration and magnetron on-off-duration tuples:

$$S = \begin{bmatrix} \{\theta_1 : d_{\theta 1}\}, \{\theta_2 : d_{\theta 2}\}, \{\theta_3 : d_{\theta 3}\}, \ldots \\ \{o_1 : d_{o1}\}, \{o_2 : d_{o2}\}, \{o_3 : d_{o3}\}, \ldots \end{bmatrix} \tag{2}$$

$$D = \sum \{d_{\theta 1}, d_{\theta 2}, d_{\theta 3}, \ldots\} = \sum \{d_{o1}, d_{o2}, d_{o3}, \ldots\} \tag{3}$$

where $\{\theta_k : d_{\theta k}\}$ indicates that the turntable will stay at the absolute offset angle θ_k for a duration of deu, $\{o_k : d_{ok}\}$ describes the duration d_{ok} for keeping the magnetron on or off (o_k). Based on these definitions, we now formulate SDC's core optimization problem as follows:

$$S^\star = \arg \min_S \sum || \overline{P}(S) - P ||^2 \tag{4}$$

where *P(S)* denotes the temperature trajectory for the n pixels using a rotation plan *S* over time.

5.3. Actuation algorithm

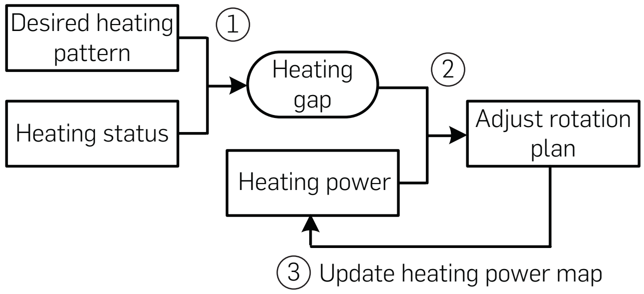
Solving this optimization problem is challenging for two reasons. First, microwaves heat the food through a standing wave, so they cannot heat individual pixels independently. Heating one pixel will inevitably heat other pixels as well. To achieve the desired heat pattern, we need to select a set of heat patterns whose union is equivalent to the target heat pattern. Second, the heat pattern, the result of the EM-ingredient coupling, is nonstatic and unpredictable. The EM-field distribution changes gradually when the turntable rotates the food and when the food heats up. SDC cannot predict the output heat pattern until the food is heated.

The stochastic knapsack: At a high level, this problem is a variant of the stochastic knapsack problem,[11] a classic resource allocation problem of selecting a subset of items to place into a knapsack of given capacity. Placing each item in the knapsack consumes a random amount of the capacity and provides a stochastic reward,[11] which is only observable after the item is placed. Due to the intrinsic uncertainty of stochastic knapsack problems, adaptive and closed-loop strategies often perform better than open-loop ones in which the items chosen are invariant of the remaining time budget.[2]

Approximation algorithm: We propose a greedy approximation algorithm to determine the immediate rotation plan on-the-fly based on the sensing result in section 4. Our greedy strategy is as follows: "at each step of the journey, heat at the rotation angle whose temperature gradient is most similar to the current heating gap".

Figure 9 illustrates the workflow of the greedy algorithm. In SDC, the thermal camera continuously senses the current food temperature at the n pixels: $C = \{c_1, c_2, \ldots, c_n\}$. SDC then compares the desired heating pattern $f(B_i, j)$ with the observed thermal distribution, and computes the real-time heating gap $G = \{g_1, g_2, \ldots, g_n\}$. When running, SDC continuously computes the similarity between the temperature gradient at θ and the current heating gap using the cosine of an angle between these two vectors. Once computed, SDC rotates the turntable to θ^*, which has the most well-aligned temperature gradient. Simultaneously, SDC updates the dictionary of temperature gradients based on the EM field and real-time observation of temperature from the IR camera at each rotation angle θ.

Figure 9. Workflow of SDC's heat actuation: ① SDC first computes the heating gap by comparing the desired heating pattern with the current status from the thermal camera. ② SDC then adjusts the actuation plan accordingly and ③ updates the distribution continuously.



6. IMPLEMENTATION AND EVALUATION

We implement SDC by modifying a commercial Microwave oven (Sharp SM1441CW) and conduct experiments with a variety of food types, including meat and rice, to evaluate SDC's heat sensing and actuation.

Ground truth: To obtain ground-truth temperature data, we use a noncontact IR thermometer (Etekcity Lasergrip 630), which provides ±2°C resolution from -50°C to 580°C, as well as the thermal camera we used in the SDC.

6.1. Uniform heating

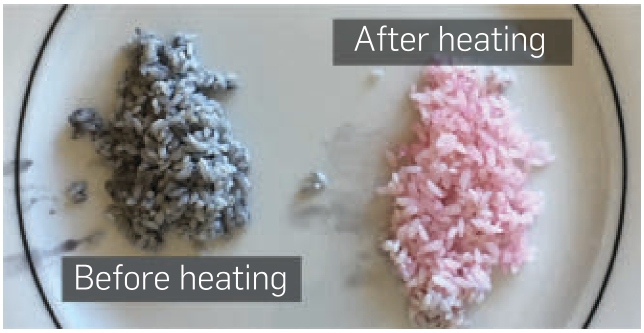
Nonuniform heating is a major drawback associated with today's microwaves,[18] which not only affects the quality of the food but also compromises food safety when the microorganisms may not be destroyed in the cold spots. This experiment evaluates SDC when provided with a Uniform heating plan, a common input thermal trajectory provided to SDC, in which we aim to heat all the pixels to the same temperature at a uniform pace.

Method: We conduct our evaluation by heating raw rice grains using SDC. To visualize the heat pattern, we color the grains with thermal-chromatic pigment (see Figure 10), which will turn into pink as the temperature increases. We use the thermal-chromatic pigment approach because it can provide a rich and analog temperature visualization, while thermal cameras have a limited resolution and the final output images are based on interpolation.

We begin our experiment at a room temperature of 20°C. We create a uniform heating recipe that requires that the food be heated uniformly from 20°C to 60°C over 2 min over space as per the recipe provided by the thick blue line in Figure 11. We note that the thermal trajectory is identical across all pixels of the food surface. However, the temperature increase is not designed to be linear over time, instead mimicking the smoothed average temperature trajectory for raw rice within a microwave under normal microwave operation.

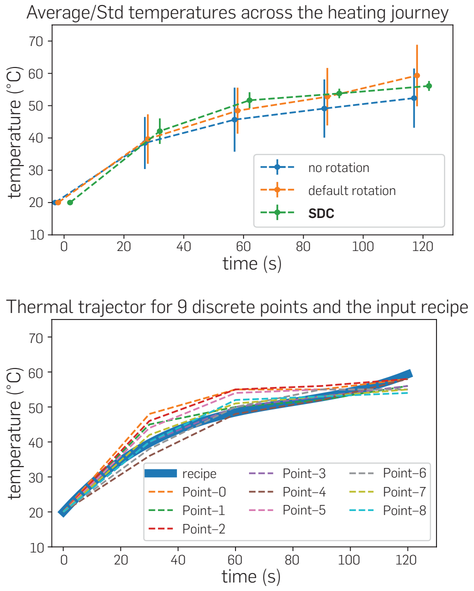
To characterize the benefits of SDC, we use two cases of blind microwave heating as the baselines: The same microwave oven (1) with and (2) without turntable rotation. To collect the immediate temperature during heating, we take out the food every 30 sec to measure ground truth.

Figure 10. We color rice grains with thermal-chromatic pigment, which turn pink in a predictable manner as their temperature increases.



Results: Figure 12 shows a visualization of the thermalchromatic pigment, which changes color at 31°C and progresses to darker shades of pink with increased temperature. The rice colored (dull white) regions denote spots of food that remain below 31°C. We observe that SDC achieves a uniform pink hue that darkens over time, whereas the baselines (no rotation or default rotation) continue to have cold spots through time. Note that SDC visually appears to have the deepest shade of pink vs. the baselines at t=120 sec as it achieves more spatially uniform temperature relative to the baselines. In actuality, there are also some hot spots of the baseline schemes that achieve even higher temperatures (over 70°C), whereas SDC achieves uniform temperature closer to 60°C as desired.

Figure 11. (above) Temperature variance of SDC uniform heating is the lowest; (below) points on food closely follow heating recipe.



Validating heat sensing and actuation: Figure 11 (bottom) shows the trajectory of the temperature over time for nine discrete uniformly spaced points of the food using SDC. Note that all points closely follow the recipe over time, which demonstrates SDC's high accuracy in modeling the temperature gradient. Figure 11 (top) compares the average and standard deviation of the trajectory across the same discrete points measured over multiple experiments vs. time. Of particular interest here is the standard deviation of the temperature of the food where one can clearly observe that SDC achieves a lower spatial variance in temperature when compared to the baseline schemes. This validates our findings that although microwaves heat food blindly and nonuniformly, SDC can achieve significant uniformity in heating.

Our results validate the correctness of both heat sensing and actuation, both of which must operate correctly to achieve the desired heating objective.

6.2. Arbitrary heating

In real-world cooking, different ingredients often require to be cooked at different temperatures. SDC can support these activities computationally by specifying thermal trajectories for different surface pixels. In this section, we aim to stress test SDC by exploring the maximum heating resolution, that is, the maximum temperature difference that can be created in a fine-grained spatial resolution.

Method: We create an imaginary recipe that heats a unique thermal pattern (depicted in Figure 13 left). The recipe sets the target temperature for the inner ring area at 500°C and the rest area at 50°C. We deliberately set an unachievable goal of 500°C for SDC to stress test the system and evaluate how well SDC can approximate to the targets. We conduct two independent experiments with and without the help of microwave accessories. To characterize the performance, we focus on the peak-to-peak temperature differences ΔP (i.e., the difference

Figure 12. Visualization of heating of rice as a function of time for no rotation, default rotation, and SDC. The dull white regions denote spots of undercooked rice. SDC results in the most uniform heating.

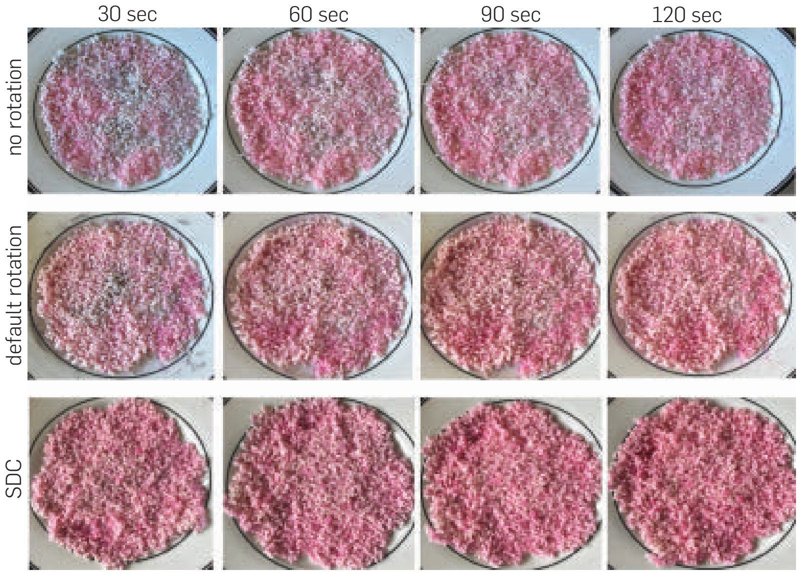


Figure 13. Left: An input recipe for stress test. Middle: SDC without susceptors. Right: SDC with susceptors. Susceptors can help build more skewed thermal distributions.

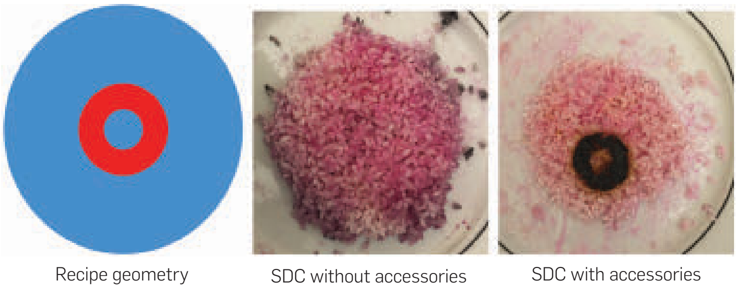
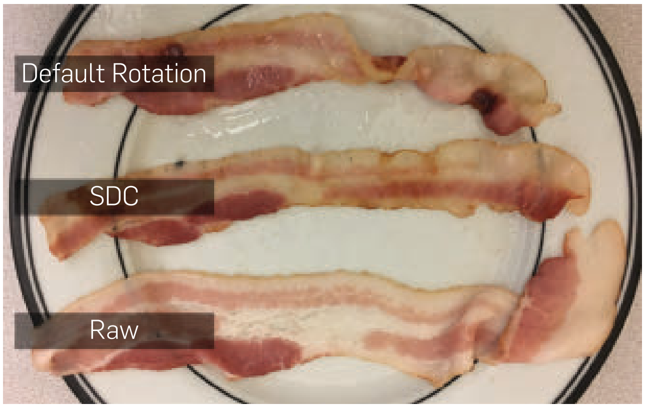


Figure 14. Mean (ΔP) and standard deviation (σ_p) of thermal delta for arbitrary heating. The final column $\left(\frac{\Delta P}{d}\right)$ denotes the temperature gradient per unit distance that can be achieved.

Scheme	ΔP	σ_P	$\frac{\Delta P}{d}$
without accessories	21°C	99°C	3°C/cm
with accessories	183°C	42°C	61°C/cm

Figure 15. The raw bacon and slices of bacon cooked with SDC and the original turntable. These three slices of bacon are from the same package, so the original fat patterns are nearly identical. Heated meat and fat will shrink. SDC applies heat to meat and fat discriminatively, so the fat shrinks more than the meat.



between the maximum and the minimum temperature).

Results: Figure 13 (right) shows a visualization of the thermal-chromatic pigment, which aligns well with the desired pattern (Figure 13 left), without and with microwave accessories. As expected, we observe that the presence of accessories helps improve the contrast between the high temperature and low temperature rings. This is precisely why accessories are needed to improve SDC's performance during tasks such as searing, where extreme temperature gradients are needed on the food.

Figure 14 on the right summarizes the mean ΔP and standard deviation σ_p of the maximum temperature difference between the inner and outer rings achieved in SDC with and without microwave accessories. We also note that with microwave accessories, SDC can cause extremely high temperature gradients (up to 61°C per centimeter) at very fine spatial resolution.

6.3 App—Cooking meat (Bacon)

In this section, we evaluate SDC's performance in cooking with advanced thermal recipes. We consider bacon with different heating requirements for the meat and fat.

Method: We use the heating recipe required per pixel for the meat vs. the fat as shown in Figure 4. Based on the package instruction, we set the heating time to 1 min. We place strips of bacon on the microwave plate across multiple experiments. We measure the continuous ground-truth temperature using the thermal camera and the final temperatures at discrete points using the noncontact IR thermometer (Etekcity Lasergrip).

Results: Figure 15 depicts the initial and final product over the cooking process. Note that although default rotation heats the bacon unevenly (resulting in the uneven shape), SDC heats the microwave more uniformly while differentiating between the heat applied to meat and fat. Indeed, we observe that the cooking process does not overcook/burn the meat, while at the same time avoiding colder spots that may pose a health hazard.

7. RELATED WORK

Related work falls under three broad categories:

Sensing in microwaves: There has been much past related work on improving the heat sensing within microwave ovens. For example, advanced FISO Microwave Work Stations (MWS)[3] used in food research have special microwave-safe fiber-optic sensors to collect real-time fine-grained direct measurements inside the cavity, but cost $80k+. Researchers[8] have also used software radios to monitor the signal strength of the microwave leakage and recognize the type of food. However, many variables, such as food type, quantity, temperature, and food location inside the oven impact microwave leakage unpredictably.[19] In contrast to these systems, SDC estimates both current and future temperature distributions by directly placing low-cost microwave-safe sensors within the cavity and modeling EM propagation.

Actuation in microwaves: The most widely adopted microwave actuation is the turntable and the stirrer blade[22] that attempt to spread radiation uniformly. However, these blind actuation approaches cannot eliminate hot/cold spots, due to the inherent unpredictability of the EM-field distribution. More recent advances in the microwave generators, such as an RF solid-state cooker, adjust the transmitter's real-time power, frequency, and phase to move the hot/cold spots around, albeit at high cost (∼$10,000[9]) and complexity. SDC also draws inspiration from many microwave accessories that have been developed to cook certain foods in a microwave—for example, Corning Ware Microwave Browners, Microwave egg boilers, or the susceptors in popcorn bags. Unlike this past work, SDC provides a generalized framework for heat actuation as per a user-specified thermal trajectory as well as the sensing results, without being tied to specific types of food or adding expensive components.

Computational fabrication and heating: Designing computational fabrication techniques[15] for digital gastronomy is an emerging topic.[23] The most relevant approach is laser cooking,[5] which uses a computer-controlled laser cutter to heat a sequence of small spots of the food surface. While innovative, the rolling pixel-by-pixel heating process is known to be highly time-consuming. SDC overcomes the slow production time of laser heating while allowing for a high degree of flexibility in the numerous heating patterns produced.

8. CONCLUSION

In this work, we presented Software-Defined Cooking (SDC), a novel next-generation microwave oven that both senses and actuates heating at fine-grained spatial resolution. Among all three common heating methods (convection, conduction, and radiation), radiation is the only one that can redirect energy toward the desired location. This redirectable feature makes the microwave oven an ideal platform to experiment with SDC, as it can effectively program energy transferring without physical hardware changes.

Meanwhile, although there has been a great deal of past work on novel RF applications for communication,[14] sensing,[7] and energy harvesting,[20] novel actuation mechanisms using RF signals are less explored so far. SDC is designed to innovate in this space.

Our current prototype already demonstrates a promising result in controlled RF actuation. With more sophisticated power transfer mechanism development (e.g., inexpensive ways of performing high-power RF beamforming), we think that future well-engineered RF cooking devices can be the ultimate cooking equipment for every family, achieving cooking quality and convenience at the same time.

References

1. Ackerman, E. A brief history of the microwave oven - IEEE spectrum. https://spectrum.ieee.org/tech-history/space-age/a-brief-history-of-the-microwave-oven, 2016. (Accessed on 02/10/2019).
2. Dean, B.C., Goemans, M.X., Vondrák, J. Approximating the stochastic knapsack problem: The benefit of adaptivity. *Math. Oper. Res. 33*, 4 (2008), 945–964.
3. FISO Technologies Inc. Mws microwave work station - product datasheet, 2018. http://tech-quality.com/images/stories/pdf/OT/mws.pdf. (Accessed on 11/21/2018).
4. Food Safety and Inspection Service, United States Department of Agriculture. Appendix a to compliance guidelines, 1999. https://www.fsis.usda.gov/Oa/fr/95033f-a.htm?redirecthttp=true. (Accessed on 02/21/2019).
5. Fukuchi, K., Jo, K., Tomiyama, A., Takao, S. Laser cooking: A novel culinary technique for dry heating using a laser cutter and vision technology. In *Proceedings of the ACM Multimedia 2012 Workshop on Multimedia for Cooking and Eating Activities* (2012), ACM, New York, NY, USA, 55–58.
6. Howell, E. Things you shouldn't cook in a microwave | microwave safety, 2013. https://www.livescience.com/34406-microwave-safety-experiments.html. (Accessed on 03/17/2019).
7. Jin, H., Yang, Z., Kumar, S., Hong, J.I. Towards wearable everyday body-frame tracking using passive rfids. *Proc. ACM Interact. Mobile Wearable Ubiquitous Tech. 1*, 4 (2018), 145.
8. Kawahara, Y., Bian, X., Shigeta, R., Vyas, R., Tentzeris, M.M., Asami, T. Power harvesting from microwave oven electromagnetic leakage. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp'13* (2013), ACM, New York, NY, USA, 373–382.
9. Manz, B. Rf energy is finally cooking, 2017. https://www.mwrf.com/community/rf-energy-finally-cooking. (Accessed on 03/15/2019).
10. Osepchuk, J.M. A history of microwave heating applications. *IEEE Trans. Microwave Theory Tech. 32*, 9 (1984), 1200–1224.
11. Pike-Burke, C., Grunewalder, S. Optimistic planning for the stochastic knapsack problem. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*. A. Singh and J. Zhu, eds. Volume 54, Proceedings of Machine Learning Research. PMLR, Fort Lauderdale, FL, USA, 2017, 1114–1122.
12. Potter, J.. *Cooking for Geeks: Real Science, Great Hacks, and Good Food*. O'Reilly, 2016.
13. Sloan, A.E. Demographic redirection. *Food Tech. 67*, 7 (2013), 38–49.
14. Talla, V., Kellogg, B., Gollakota, S., Smith, J.R. Battery-free cellphone. *Proc. ACM Interact. Mobile Wearable Ubiquitous Tech. 1*, 2 (2017), 25.
15. Torres, C., Chang, J., Patel, A., Paulos, E. Phosphenes: Crafting resistive heaters within thermoreactive composites. In *Proceedings of the 2019 on Designing Interactive Systems Conference, DIS'19* (2019), ACM, New York, NY, USA, 907–919.
16. United States Department of Agriculture. Microwave ovens and food safety, 2013. https://www.fsis.usda.gov/wps/portal/fsis/topics/food-safety-education/get-answers/food-safety-fact-sheets/appliances-and-thermometers/microwave-ovens-and-food-safety/ct_index. (Accessed on 02/14/2019).
17. U.S. Food & Drug Administration. Microwave oven radiation safety standard, 2017. https://www.fda.gov/radiation-emittingproducts/resourcesforyouradiationemittingproducts/ucm252762.htm. (Accessed on 02/10/2019).
18. Vadivambal, R., Jayas, D. Non-uniform temperature distribution during microwave heating of food materials - A review. *Food Bioprocess Tech. 3*, 2 (2010), 161–171.
19. Vollmer, M. Physics of the microwave oven. *Phys. Educ. 39*, 1 (2004), 74.
20. Wang, J., Zhang, J., Saha, R., Jin, H., Kumar, S. Pushing the range limits of commercial passive rfids. In *16th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19)* (2019), USENIX Association, Boston, MA, 301–316.
21. Wikipedia. Dielectric heating - Wikipedia, 2018. https://en.wikipedia.org/wiki/Dielectric_heating. (Accessed on 09/05/2018).
22. Yu, H.-S. *Microwave Oven with a Turntable and Mode Stirrers*, US Patent 5,877,479, 2 1999.
23. Zoran, A., Cohen, D. Digital konditorei: Programmable taste structures using a modular mold. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (2018), ACM, New York, NY, USA, 400.

Haojian Jin and Jason Hong ({haojian@cs., jasonh@cs.}cmu.edu), Human-Computer Interaction Institute, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA.

Jingxian Wang and Swarun Kumar ({jingxian@, swarun@}cmu.edu), Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, USA.

Copyright held by authors/owners.

This work is licensed under a https://creativecommons.org/licenses/by/4.0/

DOI:10.1145/3491198

To view the accompanying paper, visit doi.acm.org/10.1145/3491201 rh

Technical Perspective
A Recipe for Protecting Against Speculation Attacks

By Timothy Sherwood

THERE HAS BEEN a great deal written about the threat posed by Spectre and Meltdown style attacks to our computing infrastructure. The authors of "How to Live in a Post-Meltdown and -Spectre World" (*Communications*, Dec. 2018, p. 40) rightly note that "Meltdown and Spectre were particularly difficult to patch" and that "the scope of vulnerabilities such as Meltdown and Spectre is so vast that it can be difficult to address." There are many nuances to such an attack (see "Spectre Attacks: Exploiting Speculative Execution" (*Communications*, July 2020, p. 93), but part of the reason they are so problematic is they really describe a new *recipe* for attacks. Specifically, they show how to use a fundamental aspect of machine operation, speculation, against the memory read protections enforced by that very same machine. While any given instance of the attack might rely on the peculiarities of a specific memory hierarchy or software organization, this recipe is surprisingly general.

Many new solutions to these attacks have been proposed since the vulnerability was disclosed, but most of them only address specific instances of the vulnerability rather than the underlying problem. They can block a specific set of attacks, but not *new* instances of the recipe. A simple tuning of parameters, changing of exfiltration paths, or use of other micro-architectural conflicts can defeat many of these approaches. Unlike a bug or a bit-flip error, an adversary will purposefully and intelligently find new unprotected paths to work around a countermeasure. An approach capable of providing long-term protection needs to speak to the fundamental issues at the heart of this new class of attacks. While the following paper is not the end of the speculation-based attacks, it might be a beginning of an end.

Nearly everything in a machine short of commit (typically the very last stage of the processor pipeline) happens speculatively and "make the common case fast" is a mantra etched into the mind of everyone up and down the hardware stack. So, do we have to give up speculation and timing variation entirely to address speculation-based attacks? One would certainly hope not because the performance impact will be dire. Instead of abandoning speculation, the following paper proposes a new computer architecture that intelligently limits the impact of speculation in very specific ways such that it simultaneously allows enough predictive execution that reasonable performance can be maintained while also guaranteeing the memory hierarchy can't be used as part of such an attack. The key idea is it is "safe to execute and selectively forward the results of speculative instructions" that read data you wish to keep secret as long as you can also "prove that the forwarded results do not reach potential covert channels." This is easy to write, but harder to realize, and then even harder to prove. While I leave it to the authors to describe how the technique works, I find it satisfying both as an engineer (in that I find it very implementable) and a scientist (because it is formally grounded in a way that gets to a fundamental aspect of computer security). The authors provide one of the very first solutions that is a real *recipe* for *fixing* this problem more generally. There is absolutely a reason why this work has already been honored with a "Best Paper Award" at the IEEE/ACM International Symposium on Microarchitecture and selected for IEEE Micro Top Picks of 2019.

An approach capable of providing long-term protection needs to speak to the fundamental issues at the heart of this new class of attacks.

Looking forward I cannot help but look back. The threat posed by timing channels, albeit in a different context, is described in this very publication 48 years ago in "A Note on the Confinement Problem" (*Communications*, Oct. 1973, p. 613). Butler Lampson notes that a "service can transmit information which a concurrently running process can receive by observing the performance of the system" and goes on to warn us the "communication channel thus established is a noisy one, but the techniques of information theory can be used to devise an encoding which will allow the information to get through reliably no matter how small the effects the service on system performance are, provided they are not zero." This statement remains as true today as it was then. The fact that a channel is noisy, slow, unpredictable, or seemingly difficult to set up, is little comfort when one understands that our field has spent decades developing robust means of communication under just such conditions. Two options appear to remain—we accept that arbitrary system state (for example, memory values) will just always leak over time and learn to live in that new reality, or we develop new software and hardware techniques, like the ones proposed here, that thoughtfully and even provably limit information flows. Perhaps a mix of these two options is inevitable, but hardware/software systems that can maintain our existing abstractions while simultaneously providing real clarity on where information can and cannot escape is a noble objective. This paper gives me real hope that such an objective is truly achievable. C

Timothy Sherwood is a professor in the Department of Computer Science at the University of California, Santa Barbara, CA, USA. He is also the co-founder of Tortuga Logic.

Copyright held by author.

DOI:10.1145/3491201

Speculative Taint Tracking (STT): A Comprehensive Protection for Speculatively Accessed Data

By Jiyong Yu, Mengjia Yan, Artem Khyzha, Adam Morrison, Josep Torrellas, and Christopher W. Fletcher

Abstract

Speculative execution attacks present an enormous security threat, capable of reading arbitrary program data under malicious speculation, and later exfiltrating that data over microarchitectural covert channels. This paper proposes speculative taint tracking (STT), a high security and high performance hardware mechanism to block these attacks. The main idea is that it is safe to execute and selectively forward the results of speculative instructions that read secrets, as long as we can prove that the forwarded results do not reach potential covert channels. The technical core of the paper is a new abstraction to help identify all microarchitectural covert channels, and an architecture to quickly identify when a covert channel is no longer a threat. We further conduct a detailed formal analysis on the scheme in a companion document. When evaluated on SPEC06 workloads, STT incurs 8.5% or 14.5% performance overhead relative to an insecure machine.

1. INTRODUCTION

Speculative execution attacks such as Spectre[15] have opened a new chapter in hardware security. In these attacks, malicious speculative execution causes doomed-to-squash instructions to *access* and later *transmit* secrets over *microarchitectural covert channels* such as the processor cache.[26]

Consider "Spectre V1" (Figure 1) as an example. On modern processors, branch directions are predicted early in the processor pipeline to enable subsequent instructions to be fetched before the branch's predicate resolves. In a speculative execution attack, the attacker "mistrains" the branch predictor to predict "taken" even if the branch predicate eventually resolves to "not taken." This means that in between branch prediction and resolution, the program speculatively executes down the taken (incorrect) path: accessing a value `secret` potentially outside the bounds of `array1` and passing that value as the address to a second load reading `array2`. For the remainder of the paper, we will consider such speculatively accessed data to be secret.

In the context of Figure 1, the second load forms a microarchitectural covert channel. Specifically, on modern processors, loads result in address-dependent (and by extension `secret`-dependent) hardware resource usage due to the presence of hardware structures such as cache. Thus, an attacker that can monitor the load's hardware resource usage, or the program's execution time, can use that information to infer `secret`.

Figure 1. Spectre Variant 1 assuming a 64-byte cache line size. Variables carrying potentially secret data are colored green. If the `if` condition is predicted as true, then the cache line of array2 indexed by `secret` is loaded into the cache (Line 3) even though both loads are eventually squashed.

```
1  if (off < array1_size) {             // mispredicts
2    secret = array1[off];              // secret accessed
3    y = array2[64 * secret]; }         // secret transmitted
```

Making matters worse, an attacker that can freely control `off` can repeat the attack with different `off` to leak different secret values in the victim's memory. Further, although the above example covered Spectre V1, there are many other ways to leak secret data using similar principles. For example, by accessing secret information through other types of processor misspeculation, or by exfiltrating those secrets through other microarchitectural covert channels.

1.1. This paper's defense approach

A secure, but conservative, way to block *all* speculative execution attacks—regardless of source of misspeculation or choice of microarchitectural covert channel—is to delay executing all instructions that can access a secret until such instructions become nonspeculative. In nearly all attacks today, this would imply blocking all loads until they are nonspeculative, which would be tantamount to disabling speculative execution.

This paper proposes a principled, high-performance mechanism that achieves the same security guarantee as the above conservative scheme. The key idea is that *speculative execution is safe unless speculatively accessed data (secrets) reaches a covert channel*. In many cases, speculative instructions either do not have access to secrets or do not form covert channels, and so can execute freely under speculation. For example, the first load in Spectre V1 (Figure 1) forms a covert channel, but that channel only leaks the attacker-selected address `&array1[off]`—not the secret data stored at that address. Thus, this load's execution need not be protected. Likewise, many instructions (e.g., simple arithmetic) do not form covert channels even if their operands are secret

The original version of this paper was published in the *Proceedings of the 52nd International Symposium on Microarchitecture* (October 2019).

values. It is only when the secret is passed to a covert channel (e.g., the second load in Figure 1) that protection must be applied.

To implement this idea, we present speculative taint tracking (STT), a framework that tracks the flow of speculatively accessed data through in-flight instructions (similar to dynamic information flow tracking/DIFT[21]) until it is about to reach an instruction that may form a covert channel. STT then delays the forwarding of the data until it becomes a function of nonspeculative state or the execution squashes due to misspeculation. To be secure and efficient, we address two key challenges.

- **Identifying what is a covert channel.** First, we develop an abstraction that indicates how and when instructions can form covert channels, so as to stall data forwarding only when it becomes unsafe.
- **Identifying what is a secret.** Second, we develop a microarchitecture that determines the earliest time when data should no longer be considered secret, so as to re-enable data forwarding as soon as it becomes safe.

We now describe these two components in more detail.

1.2. New abstractions for describing microarchitectural covert channels

Covert channels come in different shapes and sizes. For example, attackers can monitor how loads interact with the cache,[15] the timing of SIMD units,[20] execution pipeline port contention,[4] branch predictor state,[1] and more. To comprehensively block information leakage through these different channels, it is necessary to understand their common characteristics.

To address this challenge, the paper proposes a new abstraction through which the covert channels on speculative microarchitectures can be viewed, discovers new points where instructions can create covert channels, and discovers a new class of covert channels. We find that all covert channels are one of two flavors, which we call explicit and implicit channels (related to explicit and implicit information flow,[19,22] respectively). In an *explicit channel*, data is directly passed to an instruction whose execution creates operand-dependent hardware resource usage and that resource usage reveals the data. For example, how a load impacts the cache depends on the load address,[15] as in Line 3 of Figure 1. In an *implicit channel*, data indirectly influences how (or whether) an instruction(s) execute, and these changes in resource usage reveal the data. For example, the instructions executed after a branch reveal the branch predicate.[4, 20] The paper further defines subclasses of implicit channel, based on when the leakage occurs and based on the nature of the secret-dependent condition that forms the channel.

Key advance: safe prediction. Through its investigation of implicit channels, the paper makes a key advance by showing *how to use hardware predictors safely*. Spectre attacks were born from attackers mistraining predictors to leak secrets. Through its abstraction for implicit channels, *STT enforces a policy that prevents arbitrary predictor mistraining from leaking any secret data over any microarchitectural covert channel*. The paper shows how this enables existing predictors to stay enabled without leaking privacy, dramatically improving performance. In the future, we expect the idea of safe prediction to enable further innovation, that is, by enabling the design of new predictors without fear of opening new security holes. Indeed, our follow-on work uses this idea to safely improve the performance of instructions that create explicit channels.[28]

1.3. Mechanisms to quickly and safely disable protection

Once we have mechanisms to block secret data from reaching covert channels, the next question is when and how to disable that protection, if speculation turns out to be correct. This is crucial for performance, as delaying data forwarding longer than necessary increases the chance that later instructions are, themselves, delayed.

STT tackles this problem with a safe but aggressive approach, *by re-enabling data forwarding as soon as data becomes a function of nonspeculative state*. For example, in Figure 1, this corresponds to the moment when the branch predicate resolves. This represents the earliest safe point but is nontrivial to determine in hardware, in general. For example, a delayed instruction's operand(s) may be the result of a complex dependency chain across many control flow and speculative operations. Intuitively, determining that data is a function of nonspeculative state would require retracing a backward slice of the program's execution, which is costly to do quickly.

Despite the above challenges, STT proposes a simple hardware mechanism that can disable protection/re-enable forwarding for an arbitrary instruction in a single cycle, using hardware similar to traditional instruction wake-up logic. The key idea is that to determine whether data is a function of nonspeculative state, it is sufficient to determine whether the *youngest* load, whose return value influences the data, has become nonspeculative. Checking this condition is akin to tracking a single extra dependency for each instruction, as opposed to performing complex backward slice tracking.

1.4. Security guarantees and formal analysis

Alongside the main paper, we formally prove that STT enforces a novel form of noninterference[9] with respect to speculatively accessed data. In a nutshell, we show that, with STT, hardware resource usage patterns over time are independent of data that eventually squashes. We released a companion technical report[29] with detailed formal analysis and a security proof for this property on a processor model implementing STT.

1.5. PUTTING IT ALL TOGETHER

Putting everything together, STT provides both high security and high performance. It does not require partitioning or flushing microarchitectural resources, and does not require changes to the cache/memory subsystem or the software stack. When evaluated on SPEC06 workloads, STT incurs 8.5% or 14.5% performance overhead (depending on the threat model) relative to an insecure machine.

2. BACKGROUND

We now provide additional details about processor microarchitecture. Also see Section 1 for basics on Spectre attacks.

Out-of-order execution. Dynamically scheduled processors execute instructions in parallel and out of program order to improve performance.[11, 23] Instructions are fetched and decoded in the processor *frontend*, *dispatched* to *reservation stations* for scheduling, *issued* to execution (functional) units in the processor *backend*, and finally *retired* (at which point they update architected system state). Instructions proceed through the frontend, backend, and retirement stages in order, possibly out of order, and in order, respectively. In-order retirement is implemented by queuing instructions in a hardware structure called the reorder buffer (ROB)[13] in instruction-fetch order, and retiring a completed instruction when it reaches the ROB head. Instructions are referred to by their age in the ROB, that is, if I_1 precedes I_2 in fetch order, then I_1 is *older* than I_2.

Speculative execution. Speculative execution improves performance by executing instructions whose validity is uncertain instead of waiting to determine their validity. If such a speculative instruction turns out to be valid, it is eventually retired; otherwise, it is *squashed* and the processor's state is rolled back to a valid state. (As a byproduct, all instructions younger than the point of misspeculation also get squashed.)

There are multiple types of speculation in modern processors, associated with different instructions and events. For example, to enable immediate fetching of instructions after a branch, that is, before the branch's predicate resolves, modern processors employ branch prediction. Branch predictors are (typically) stateful structures in the processor frontend that predict the direction of the branch based on information such as the branch's program counter and whether the branch historically has been taken/not taken. If the processor backend later resolves the branch predicate and determines the prediction to be incorrect, all subsequently fetched instructions are squashed and control flow is diverted to the correct path.

3. ATTACKER MODEL AND PROTECTION SCOPE

Attacker model. STT assumes a powerful attacker that can monitor any microarchitectural covert channel from anywhere in the system and induce arbitrary speculative execution to access secrets and create covert channels. For example, the attacker can monitor covert channels through the cache/memory system,[15] data-dependent arithmetic,[10] port contention,[4] branch predictors,[1] etc.

We note that the above attacker is very strong, perhaps even unrealistic. The goal is that through defending against such an attacker, we will by extension defend against weaker, more realistic attackers.

Scope: protecting speculatively accessed data. A speculative execution attack consists of two components.[14, 20] First, an instruction that reads a potential secret into a register, making it accessible to younger instructions. We call this instruction the *access instruction*.[14] Second, a younger instruction or instructions that exfiltrate the secret over a microarchitectural covert channel. The access instruction is almost always a load,[15, 24] but some attacks use a privileged register read.[5]

We distinguish attacks based on whether the access instruction is doomed-to-squash (*transient*) or bound to retire (*nontransient*). STT's goal is to block attacks involving doomed-to-squash access instructions, as shown in Figure 2. These attacks can access data that a correct (not misspeculated) execution would never access, which often results in being able to read from any location in memory. Attacks involving bound-to-retire access instructions are out of scope. They can only leak *retired (or bound-to-retire) register file state*, not arbitrary memory, and their leakage can be reasoned about by programmers or compilers and blocked using complementary techniques (e.g., Data-oblivious ISAs[27]).

4. ABSTRACTION FOR COVERT CHANNELS

STT proposes a novel abstraction for covert channels (Figure 3). In our abstraction, covert channels are broken into two classes: *explicit* and *implicit* channels. An *explicit channel*, related to explicit flow in information flow,[19, 22] is one where data (e.g., a secret) is *directly* passed to an instruction whose execution creates operand-dependent hardware resource usage and that resource usage reveals the data. An example is a load instruction's changes to the cache state. An *implicit channel*, related to implicit flow,[19, 22] is one where data *indirectly* influences how (or whether) an instruction or several instructions execute, and these changes in resource usage reveal the data. An example is a branch instruction, whose outcome determines subsequent instructions and thus whether some functional unit is used.

We further find new ways that implicit channels can leak, and find entirely new classes of implicit channels. Figure 4 gives examples of "traditional" (Figure 4(a)) and new (Figure 4(b) and (c)) channels. We denote the value being revealed through the channel as secret. The examples assume the attacker can monitor the cache-based covert channel, that is, the program's memory access pattern. We note that in many cases (e.g., Figure 4(a) and (b)), the load can be replaced by any instruction; in particular, not necessarily

Figure 2. STT's scope is to protect speculatively accessed data from leaking over any microarchitectural covert channel. Protecting values that are accessed nonspeculatively is outside of scope.

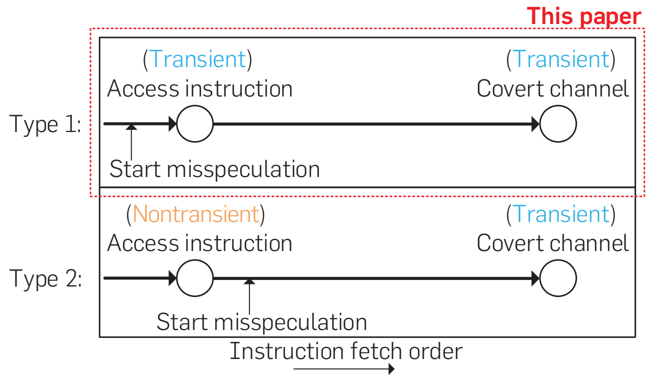


Figure 3. STT's new classification schema for microarchitectural covert channels.

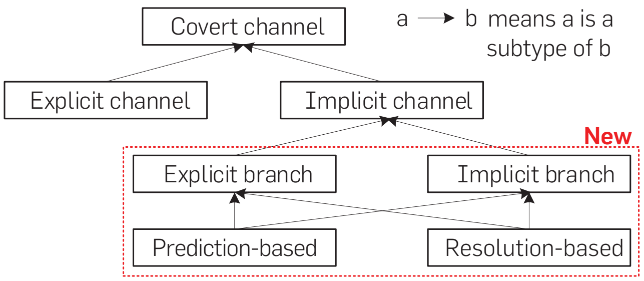
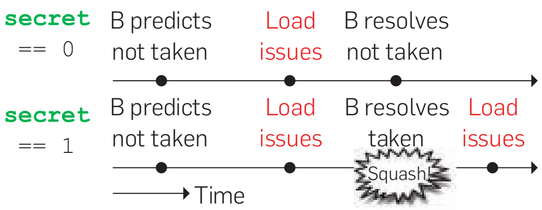


Figure 4. Examples of implicit covert channels revealing `secret`. Assume an older speculative access instruction has already read `secret` into a register, for example, Line 2 in Figure 1. The attacker can see the sequence of load addresses sent to the memory system. For stores, we assume address translation and other address-dependent actions occur when the store retires. `rX`, `rY`, and `rZ` are registers. Each of these covert channels can be "plugged into" existing attacks as the "Covert channel" in Figure 1. For example, we can replace Line 3 with one of (a)–(c) above.

(a) Control dependency:	(b) Squash dep. (new):	(c) Alias dep. (new):

`if (secret)`
`  load rX <- (rY)`
	`if (secret)`
	`  rX += 64`
	`load rY <- (rZ)`
		`store rX -> (secret)`
		`load rY <- (rZ)`

Figure 5. Resolution-based implicit channel for Figure 4(b) due to secret-dependent pipeline squashes. When the branch (B) resolves, it leaks the secret based on whether a squash occurs, as this causes the younger load to execute once or twice. There is an analogous case when the branch is predicted taken.



one that forms an explicit channel. Case in point, secret is not passed directly as the load address in any of the examples, yet still leaks.

4.1. Prediction- versus resolution-based leakage

First, we find that implicit channels can leak at two points: when a control-flow prediction is made (if any) and when that prediction is resolved. Recall, branch prediction and resolution occur in the processor frontend and backend, respectively (Section 2). This creates new types of leakage depending on the attacker's capability. In the following, consider a branch whose predicate depends on a secret.

At *prediction time*, the sequence of instructions fetched after this branch is fetched (after branch prediction but before resolution) leaks secrets if the predictor structures were updated based on secret information at some time in the past. For example, if an attacker runs repeated experiments and the branch predictor is updated speculatively based on how the branch resolves, the branch predictor "learns" the secret and will make future predictions based on the secret.

At *resolution time*, the branch can also leak the secret *even if the predictor state has not been updated based on secret data*, because incorrect predictions will cause pipeline squashes. See the code snippet in Figure 4(b), whose timing is shown as a function of the secret in Figure 5. If the attacker knows the branch will predict not taken (e.g., by priming it beforehand[15]), a squash means the branch was actually taken. The attacker can observe the squash through different effects, for example, program timing or the fact that the load issues twice.

4.2. Explicit versus implicit branches

Second, we find that implicit channels can feature either an *explicit* or an *implicit* branch. For example, in Figure 4(c), there is no explicit control-flow instruction and the load address seemingly does not depend on secret data.

Yet, there may still be an implicit channel. For example, consider a machine that performs store-to-load forwarding. With this optimization, the processor can forward data (**rX**) directly from the older in-flight store to the younger load's output register (**rY**), as opposed to waiting for the store to retire and accessing the cache, if the store/load addresses alias, that is, if secret==**rZ**. Store-to-load forwarding thus creates an implicit channel, as whether a cache access is performed depends on the secret.

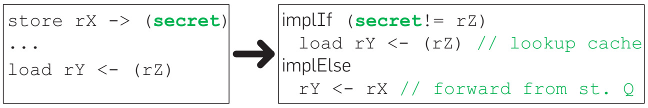
Another common technique with similar implications is memory-dependence speculation.[18] This optimization allows a load to (speculatively) read from cache even if older in-flight stores have unresolved addresses, that is, it speculates that store-to-load forwarding will not be needed. In our example, if the older store address later resolves and we have that secret==**rZ**, the load and younger instructions will squash, causing a similar pipeline disturbance as discussed in Section 4.1. (Note, this is not the already known Spectre Variant 4 (SSB) attack.[12, 25] In that attack, an *access instruction* reads stale data through a store bypass. Our attack is concerned with store bypass used as a covert channel.)

An important observation is that hardware optimizations such as those above can be modeled as *implicit branches*, whereas explicit control-flow instructions such as branches can be viewed as *explicit branches*. That is, the store-load pair in Figure 4(c) can be rewritten as shown in Figure 6, where the "implicit branch" direction is predicted if secret has not yet resolved. In this sense, implicit branches may also leak at prediction and/or resolution time (Section 4.1). For example, memory-dependence speculation is sometimes implemented with a stateful predictor called a store set predictor,[6] which tries to guess when store-load pairs will address alias, which can similarly "learn" functions of secret data.

4.3. Insights from analysis of implicit channels

Since it was proposed in the paper, the classification for

Figure 6. Rewriting a store-load pair as an implicit branch. implIf reveals a potential covert channel as a function of memory aliasing to the older store. This occurs if the microarchitecture supports store-to-load forwarding or memory-dependence speculation.



```
store rX -> (secret)
...
load rY <- (rZ)
```

```
implIf (secret!= rZ)
  load rY <- (rZ) // lookup cache
implElse
  rY <- rX // forward from st. Q
```

implicit channels has proven to be a robust and useful way to represent and pinpoint the root cause of microarchitectural attack vulnerabilities. For example, in the NetSpectre attack,[20] a secret branch predicate conditionally causes a SIMD instruction to be issued, which triggers a SIMD unit power-on event. A common misconception is that the attack root cause is SIMD unit power-on time. STT's abstraction shows, however, that the root cause is an explicit branch and that "fixing" the SIMD unit does not prevent the attack.

Even more subtly, the abstraction demonstrates and provides cases where implicit flow and privacy leakage *do occur* despite not occurring according to program semantics. For example, at the software level, neither Figure 4(b) nor (c) would be flagged as creating covert channels. Figure 4(b) would not be considered a channel because the load is control-and data-independent of the branch. Likewise, Figure 4(c) would not be considered a channel because, although there is possible information flow from **rX** to **rY** due to address aliasing, this information flow does not (seemingly) impact the memory access pattern. Generally speaking, the analysis shows that in advanced processors, subtle microarchitectural decisions that are orthogonal to program semantics must be taken into account to reason about possible microarchitectural covert channels.

Finally, the abstraction applies to a large set of microarchitectural optimizations. For example, the representation of store-to-load forwarding and memory-dependence speculation (Figure 6) also captures the behavior of memory consistency speculation,[8] value prediction,[16] and other optimizations. For reference, Table 1 specifies the channel types for existing attacks and a variety of hardware optimizations. As we will see in the next sections, being able to represent different optimizations as *predictions on implicit branches* will enable STT to apply a uniform mechanism to block leakage through a variety of structures (e.g., branch, store set, etc., predictors).

Table 1. Classifying existing attacks and covert channel-creating hardware structures.

Channel	Spectre PoC?	Type	Branch type
Cache timing[17, 26]	Spectre V1[15]	Exp	–
Execution unit timing[3, 10]	–	Exp	–
SIMD utilization	NetSpectre[20]	Imp	Exp
Port contention[2]	SmotherSpectre[4]	Imp	Exp
Store-load forwarding	–	Imp	Imp
Mem. dep. prediction[18]	–	Imp	Imp
Mem. consist. speculation[8]	–	Imp	Imp
Value prediction[16]	–	Imp	Imp

A channel's *Type* can be either Explicit (Exp) or Implicit (Imp), c.f. Section 4. An implicit channel's *Branch Type* is likewise Exp or Imp, c.f. Section 4.2. Attacks utilizing implicit channels may be either prediction- or resolution-time (Section 4.1); thus, we leave that field out.

5. STT: DESIGN

STT "taints" secret (speculatively accessed) data as it flows through the pipeline in a manner similar to dynamic information flow tracking (DIFT).[7, 21] The STT framework (Section 5.1) defines which data should be tainted, which instructions might leak it and thus should be protected, and when protection can be disabled. STT tracks the flow of tainted data between instructions in the ROB and automatically "untaints" data once the instruction that produces it becomes nonspeculative (Section 5.2), in contrast to conventional DIFT schemes. Based on taint information, STT applies novel protection mechanisms to block both explicit and implicit covert channels (Section 5.3).

5.1. Framework and concepts

STT requires that the microarchitect defines what instructions write secrets into registers (*access instructions*, mainly loads), what instructions can form explicit channels (*transmitters*), and what instructions form implicit channel branch predicates (for both explicit and implicit branches). Finally, the architect must define the *Visibility Point*, after which speculation is considered safe (e.g., at the point of the oldest unresolved branch, or at the head of the ROB). If the Visibility Point refers to an instruction older than an access instruction, we call the access instruction *unsafe*; otherwise, it is considered *safe*.

We provide guidelines for microarchitects to identify access and transmit instructions. An instruction should be classified as an access instruction if it has the potential to return a secret. Except for loads, there are only a handful of such instructions, which can be identified manually.

An instruction should be classified as a transmit instruction if its execution creates operand-dependent resource usage that can reveal the operand (partially or fully). Identifying implicit branches is similar: the architect must analyze whether the resource usage of some in-flight instruction changes as a function of *some other* instruction's operand. This definition can be formalized by analyzing (offline) how information flows in each functional unit at the SRAM-bit and flip-flop levels to determine whether resource usage depends on the input value, in the style of the OISA[27] or GLIFT[22] formal frameworks. Automatically performing such analysis is important future work.

5.2. Taint and untaint propagation

Conceptually, in each clock cycle, STT applies the following taint rules to instructions in the ROB:

- **The output register of an access instruction** is tainted if and only if the access instruction is unsafe.
- **The output register of a nonaccess instruction** is tainted if and only if at least one of its input operands is tainted.

In the implementation, taint propagation is piggybacked on the existing register renaming logic in an out-of-order core. Tainting is therefore fast. By contrast, it is difficult to propagate "untaint," to all dependencies of an access instruction that becomes safe, in a single cycle. We address this with a single-cycle implementation for untaint in Section 6.

Unlike prior DIFT schemes,[21] STT does not require tracking taint in any part of the memory system or across store-to-load forwarding. The reason is that because loads are access instructions, the taint of their output is determined only based on whether they have reached the Visibility Point. That is, the output of an unsafe load is always tainted.

5.3. Blocking covert channels

Given STT's rules for tainting/untainting data and its abstraction for covert channels, STT blocks all covert channels by applying a uniform rule across each type.

Blocking explicit channels. STT blocks explicit channels by delaying the execution of any transmit instruction whose operands are tainted until they become untainted. This scheme imposes relatively low overhead because it only delays the execution of transmit instructions if they have tainted operands. For example, a load that only returns a secret but does not have (transmit) a secret operand—such as the load on Line 2 in Figure 1—executes without delay. The load on Line 3, however, will be delayed and eventually squashed, thereby defeating the attack.

Blocking implicit channels. STT blocks implicit channels by enforcing an invariant that the sequence of instructions fetched/executed/squashed never depends on tainted data. That is, *STT makes the program counter independent of tainted data*. To enforce this invariant efficiently, without needing to delay execution of instructions following a tainted branch, we introduce two general principles to neutralize the sources of implicit channels:

- **Prediction-based implicit channels** are eliminated by preventing tainted data from affecting the state of any predictor structure.
- **Resolution-based implicit channels** are eliminated by delaying the effects of branch resolution until the (explicit or implicit) branch's predicate becomes untainted.

The above principles can be applied to efficiently make *any* hardware predictor impossible to exploit as a covert channel for leaking speculatively accessed data.

Conceptually, the protection mechanism does not need to reason about whether an implicit channel is caused by an explicit or implicit branch: both types have a predicate, and the policy with respect to the predicate is the same in both cases. The implementation, however, must identify the predicate. We illustrate this by showing how the STT microarchitecture handles explicit branches.

Applying Principle #1 (prediction-based channels). STT requires that every frontend predictor structure be updated based only on untainted data. This makes the execution path fetched by the frontend unaffected by the output of unsafe access instructions. Specifically, STT passes a branch's resolution results to the direct/indirect branch predictors only after the branch's predicate and target address become untainted; if the branch gets squashed before this, the predictor will not be updated.

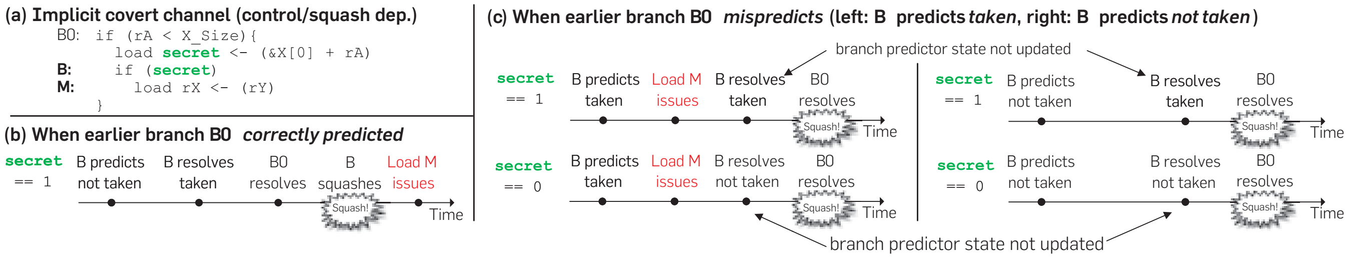
Figure 7(c) demonstrates the effect of STT on a speculative execution of the code snippet in Figure 7(a), in which the branch `B0` is mispredicted as taken. No matter how many experiments the attacker runs, the predicted direction of the branch **B** will not be a function of `secret`, because the branch predictor is not updated when **B** resolves. As a result, the execution path does not depend on `secret` (top vs. bottom)—it only depends on the predicted branch direction (left vs. right).

Applying Principle #2 (resolution-based channels). STT delays squashing a branch that resolves as mispredicted until the branch's predicate becomes untainted. As a result, a doomed-to-squash branch with a tainted predicate (such as the branch **B** in Figure 7(c)) will never be squashed and re-executed, preventing the implicit channel leak discussed in Section 4.3. As Figure 7(c) shows, the doomed-to-squash branch **B** is eventually squashed once an older (mispredicted) branch with an untainted predicate squashes. Thus, the squash does not leak any information about the branch's resolution. Importantly, it is safe to resolve a branch *as soon as* its predicate becomes untainted, even if an *older branch with a tainted predicate* has not yet resolved.

STT only increases the latency of *recovering* from a *tainted branch* misprediction. For example, in Figure 7(b), the load does not execute immediately after B resolves. Fortunately, tainted branch mispredictions are only a small fraction of overall branch mispredictions, which are infrequent in the first place because successful speculation requires accurate branch prediction.

Implicit branches. The paper applies STT's principle to secure several common microarchitectural optimizations

Figure 7. STT executing the code in (a), which includes an untainted branch B0, an access instruction reading `secret`, and an implicit channel (due to branch B).



(a) Implicit covert channel (control/squash dep.)

```
B0:  if (rA < X_Size){
        load secret <- (&X[0] + rA)
B:      if (secret)
M:        load rX <- (rY)
     }
```

(b) When earlier branch B0 *correctly predicted*

(c) When earlier branch B0 *mispredicts* (left: B predicts *taken*, right: B predicts *not taken*) that can be formulated as implicit branches, namely: store-to-load forwarding, memory-dependence speculation, and memory consistency speculation. In the process, the paper details various optimizations and cases which arise when dealing with implicit channels. In particular, whether the explicit/implicit branch has a prediction step can be resolved early or can be optimized in some other way. For example, because store-to-load forwarding can only result in two observable outcomes (issue the load or forward from a prior store), we hide which one occurs by unconditionally accessing the cache.

6. STT: IMPLEMENTATION

We previously assumed untaint information propagated along data dependencies instantly. This is difficult to implement in hardware because a word of tainted data may be a function of complex dependency chains involving many access instructions.

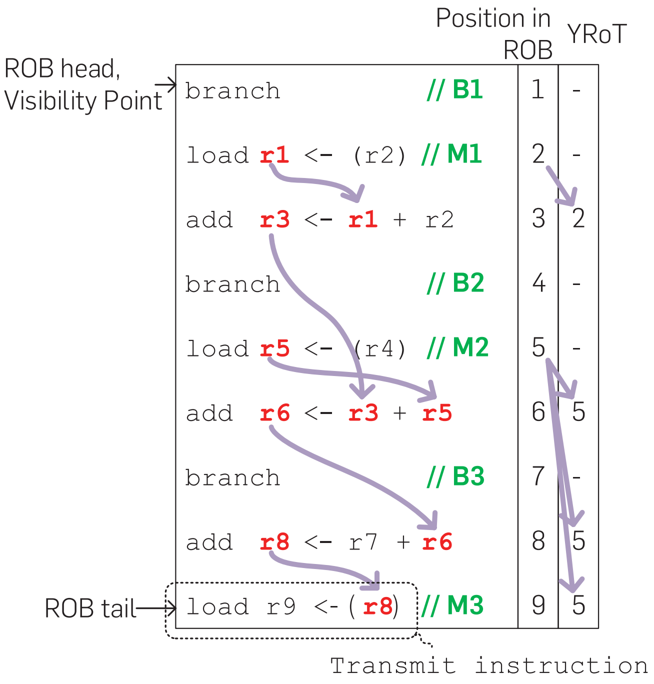
A tainted register needs to be untainted once all the access instructions on which it depends reach the Visibility Point, that is, become safe. Our key observation is that it suffices to track only when the *youngest access instruction* becomes safe, because instructions become nonspeculative in program order in the processor reorder buffer (ROB). We call this youngest access instruction the *youngest root of taint (YRoT)*.

Determining the YRoT is done through modifications to rename logic in the processor frontend. Specifically, the YRoT for an instruction X being renamed is given by the max of (1) the YRoT(s) of the instruction(s) producing the arguments for X, if those instructions are not access instructions; or (2) the ROB index of the instruction(s) producing the arguments for X, otherwise. (By convention, we assume the ROB index increases from ROB head to tail.) After renaming, the YRoT is stored alongside the instruction in its reservation station and is conceptually an extra dependency for that instruction. When the Visibility Point changes, its new position is broadcast to in-flight instructions, akin to a normal writeback broadcast, and instructions whose YRoT is less than the Visibility Point's new position are allowed to execute (assuming their other dependencies are satisfied). The entire architecture requires modest changes to the frontend rename logic, storage in reservation stations for the YRoT, and logic to compare the YRoT to the Visibility Point which is comparable to normal instruction wakeup logic.

Figure 8 shows an example. Assume the Spectre attack model, that is, the Visibility Point will be set to the ROB index of the oldest unresolved branch. The ROB contains 3 unresolved branches (`B1–B3`) and a transmit instruction (`M3`) whose operand/address `r8` is a function of the return value of two access instructions (`M1` and `M2`). `M3` is a transmit instruction (because it is a load) and can potentially leak secrets because misspeculations on branches `B1` and `B2` can influence the data returned by loads `M1` and `M2`, which in turn contribute to the address of `M3` through data dependencies.

On the one hand, the data dependency chain from load `M1` all the way to load `M3` is quite complex. That is, the instruction at ROB index 6 depends on index 5 and index 3, index 8 depends on 6, etc. Re-traversing this dataflow graph to propagate untaint, akin to tracing backwards slices, would be expensive. On the other hand, the YRoT dependency chain is relatively simple. Each instruction just tracks whichever is the youngest load that contributes to its dependency chain (e.g., load `M2` for instructions 6, 8 and 9). When branches `B1` and `B2` resolve, the Visibility Point advances to point to branch `B3` (ROB index 7). As 7 is greater than 5 (the YRoT for the transmit instruction `M3`), `M3` is allowed to execute at this point. Note, the dependency chain could have been more complex, with additional branches and arithmetic dependencies separating load `M2` and load `M3`, but this would not change the moment that it is safe to execute load `M3`.

Figure 8. Example showing YRoT tracking showing a snapshot of ROB state. Addition (add) instructions are used to represent arithmetic (non-loads). If the YRoT is set to '-', it means the instruction's youngest dependent access instruction is a part of retired state.



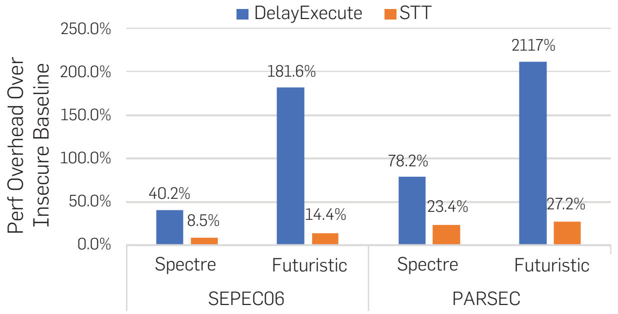
	Position in ROB	YRoT
branch // B1	1	-
load r1 <- (r2) // M1	2	-
add r3 <- r1 + r2	3	2
branch // B2	4	-
load r5 <- (r4) // M2	5	-
add r6 <- r3 + r5	6	5
branch // B3	7	-
add r8 <- r7 + r6	8	5
load r9 <- (r8) // M3	9	5

Importantly, the above scheme is only secure after applying STT's mechanisms to block *both* explicit and implicit channels (Section 5). That is, the scheme requires that `r8` is not a function of speculative data *at the exact moment load M2 becomes nonspeculative*. This requires that branch `B3` not be influenced by speculative data (achieved by protections for implicit channels) and that other intervening instructions that can cause explicit channels not execute until they are likewise safe (achieved by protections for explicit channels).

7. FORMAL ANALYSIS/SECURITY PROOF

We formally prove in a companion document[29] that STT enforces a novel notion of noninterference: at each step of the execution, the value of a *doomed* register—a register written to by a bound-to-squash access instruction—does

Figure 9. Performance evaluation on SPEC06 and PARSEC benchmark suites. STT outperforms the baseline secure scheme (DelayExecute) with much smaller performance overhead, for both Spectre and Futuristic attacker models.



not influence future visible events in the execution. This applies to all microarchitectural timing and interference-based attacks. For instance, the property ensures that the program's completion time and hardware resource usage—for all hardware structures such as cache, branch predictor, etc.—are completely independent of doomed values.

The key challenge in the analysis is how to avoid "looking into the future" to determine if an instruction is doomed to squash. We address this by running the STT machine alongside a nonspeculative in-order processor, which allows us to verify the STT machine's branch predictions and determine whether a prediction leads to misspeculation or not.

8. EVALUATION RESULTS

We evaluate STT on 21 SPEC and 9 PARSEC workloads. The results are shown in Figure 9. Relative to an insecure machine, STT adds only 13.0%/18.2% overhead (averaged across both SPEC and PARSEC benchmarks) depending on whether the attack model considers only control-flow speculation (Spectre) or all types of speculation (Futuristic). Compared to the baseline secure scheme (DelayExecute) described in Section 1, STT reduces overhead by 4.0× in the Spectre model and 10.5× in the Futuristic model, on average. This indicates that defending against stronger attack models is viable with STT without sacrificing much performance.

Acknowledgments

This work was funded in part by NSF under grant CNS-1816226, Blavatnik ICRC at TAU, ISF under grant 2005/17, and by an Intel Strategic Research Alliance (ISRA) grant. We thank Joel Emer, Sarita Adve, and Shubu Mukherjee for very helpful discussions. We would especially like to thank our colleagues at Intel who contributed significant feedback throughout the project's development, in particular Fangfei Liu, Matthew Fernandez, Frank McKeen, and Carlos Rozas.

References

1. Aciicmez, O., Seifert, J.-P., Koc, C.K. Predicting secret keys via branch prediction. In *IACR'06* (2006).
2. Aldaya, A.C., Brumley, B.B., ul Hassan, S., García, C. P., Tuveri, N. Port contention for fun and profit. In *IACR'18* (2018).
3. Andrysco, M., Kohlbrenner, D., Mowery, K., Jhala, R., Lerner, S., Shacham, H. On subnormal floating point and abnormal timing. In *S&P'15* (2015).
4. Bhattacharyya, A., Sandulescu, A., Neugschwandtner, M., Sorniotti, A., Falsafi, B., Payer, M., Kurmus, A. SMoTherSpectre: Exploiting speculative execution through port contention. In *CCS'19* (2019).
5. Canella, C., Bulck, J.V., Schwarz, M., Lipp, M., von Berg, B., Ortner, P., Piessens, F., Evtyushkin, D., Gruss, D. A systematic evaluation of transient execution attacks and defenses. In *USENIX Security'19* (2019).
6. Chrysos, G.Z., Emer, J.S. Memory dependence prediction using store sets. In *ISCA'98* (1998).
7. Dalton, M., Kannan, H., Kozyrakis, C. Raksha: A flexible information flow architecture for software security. In *ISCA'07* (2007).
8. Gharachorloo, K., Gupta, A., Hennessy, J. Two techniques to enhance the performance of memory consistency models. In *ICPP'91* (1991).
9. Goguen, J.A., Meseguer, J. Security policies and security models. In *1982 IEEE Symposium on Security and Privacy* (1982).
10. Großschädl, J., Oswald, E., Page, D., Tunstall, M. Side-channel analysis of cryptographic software via early-terminating multiplications. In (2009).
11. Hennessy, J.L., Patterson, D.A. *Computer Architecture: A Quantitative Approach*, 6th edn. Morgan Kaufmann Publishers Inc., 2017.
12. Intel. Q2 2018 speculative execution side channel update, 2018. https://www.intel.com/content/www/us/en/ security-center/advisory/intel-sa-00115.html.
13. Johnson, M. *Superscalar Microprocessor Design*. Prentice Hall Englewood Cliffs, New Jersey, 1991.
14. Kiriansky, V., Lebedev, I.A., Amarasinghe, S.P., Devadas, S., Emer, J. DAWG: A defense against cache timing attacks in speculative execution processors. In *MICRO'18* (2018).
15. Kocher, P., Genkin, D., Gruss, D., Haas, W., Hamburg, M., Lipp, M., Mangard, S., Prescher, T., Schwarz, M., Yarom, Y. Spectre attacks: Exploiting speculative execution. In *S&P'19* (2019).
16. Lipasti, M.H., Wilkerson, C.B., Shen, J.P. Value locality and load value prediction. In *ASPLOS'96* (1996).
17. Percival, C. Cache missing for fun and profit. In *Proceedings of BSDCan 2005* (2005).
18. Reinman, G., Calder, B. Predictive techniques for aggressive load speculation. In *MICRO'98* (1998).
19. Sabelfeld, A., Myers, A.C. Language-based information-flow security. *IEEE J. Sel. Areas Commun. 21*, 1 (Jan. 2003), 5–19.
20. Schwarz, M., Schwarzl, M., Lipp, M., Gruss, D. Netspectre: Read arbitrary memory over network. In *ESORICS'19* (2019).
21. Suh, G.E., Lee, J.W., Zhang, D., Devadas, S. Secure program execution via dynamic information flow tracking. In *ASPLOS'04* (2004).
22. Tiwari, M., Wassel, H.M., Mazloom, B., Mysore, S., Chong, F.T., Sherwood, T. Complete information flow tracking from the gates up. In *ASPLOS'09* (2009).
23. Tomasulo, R.M. An efficient algorithm for exploiting multiple arithmetic units. *IBM J. Res. Dev. 11*, 1 (1967), 25–33.
24. Van Bulck, J., Minkin, M., Weisse, O., Genkin, D., Kasikci, B., Piessens, F., Silberstein, M., Wenisch, T.F., Yarom, Y., Strackx, R. Foreshadow: Extracting the keys to the Intel SGX kingdom with transient out-of-order execution. In *USENIX Security'18* (2008).
25. Yan, M., Choi, J., Skarlatos, D., Morrison, A., Fletcher, C.W., Torrellas, J. InvisiSpec: Making speculative execution invisible in the cache hierarchy. In *MICRO'18* (2018).
26. Yarom, Y., Falkner, K. Flush+Reload: A high resolution, low noise, L3 cache side-channel attack. In *USENIX Security'14* (2014).
27. Yu, J., Hsiung, L., Hajj, M.E., Fletcher, C.W. Data oblivious ISA extensions for side channel-resistant and high performance computing. In *NDSS'19*. https://eprint.iacr.org/2018/808.
28. Yu, J., Mantri, N., Torrellas, J., Morrison, A., Fletcher, C.W. Speculative data-oblivious execution: Mobilizing safe prediction for safe and efficient speculative execution. In *ISCA'20*.
29. Yu, J., Yan, M., Khyzha, A., Morrison, A., Torrellas, J., Fletcher, C.W. *Speculative Taint Tracking (STT): A Formal Analysis*. Technical report, University of Illinois at Urbana-Champaign and Tel Aviv University, 2019. http://cwfletcher.net/Content/Publications/Academics/TechReport/stt-formal-tr_micro19.pdf.

Jiyong Yu, Josep Torrellas, and Christopher W. Fletcher, University of Illinois at Urbana-Champaign, IL, USA.

Mengjia Yan, Massachusetts Institute of Technology, Cambridge, MA, USA.

Artem Khyzha and Adam Morrison, Tel Aviv University, Israel.

© 2021 ACM 0001-0782/21/12 $15.00

California State University, Sacramento
Department of Computer Science
Tenure-Track Assistant Professor

Two tenure-track assistant professor positions to begin with the Fall 2022 semester. Applicants specializing in all areas of computer science will be considered. However, those with knowledge/skill in computer graphics and game engine design OR computer architecture/hardware are especially encouraged to apply as the department has an urgent need to meet student demand in these two particular areas. Ph.D. in Computer Science, Computer Engineering, or closely related field required by the time of the appointment. For detailed position information, including application procedure, please see https://careers.csus.edu/en-us/listing/.

Screening will begin December 1, 2021, and remain open until filled. AA/EEO employer. Clery Act statistics available. Mandated reporter requirements. Criminal background check will be required.

Georgia Institute of Technology
Tenure-Track Faculty 2021-2022

The School of Computational Science and Engineering (CSE) in the College of Computing at the Georgia Institute of Technology invites applications for multiple openings at the Assistant Professor level (tenure-track); exceptional candidates at the Associate Professor and Professor level also will be considered. CSE focuses on foundational research of an interdisciplinary nature that enables advances in science, engineering, medical, and social domains. Applicants are expected to develop and sustain a research program in one or more of our core areas: high-performance computing, scientific and numerical computing, modeling and simulation, discrete algorithms, and large-scale data analytics (including machine learning and artificial intelligence).

All areas of research will be considered, especially scientific artificial intelligence (AI methods unique to scientific computing), urban computing (enabling effective design and operation of cities and urban communities), application-driven post-Moore's law computing, and data science for fighting disease. Applicants must have an outstanding record of research and a commitment to teaching.

Applicants are expected to engage in substantive research with collaborators in other disciplines. For example, current faculty have domain expertise and/or collaborations in computational chemistry; earth sciences; biomedical and health sciences; urban systems and smart cities; social good and sustainable development; materials and manufacturing; and others.

For more information, including how to apply, go to: https://academicjobsonline.org/ajo/jobs/19677.

For full consideration, applications are due by December 1, 2021. To be considered for the Edenfield Early Career Professorship, submit by November 1, 2021

Georgia Tech is organized into six Colleges. The School of Computational Science and Engineering resides in the College of Computing along with the School of Computer Science, the School of Interactive Computing and the School of Cybersecurity and Privacy. Joint appointments with other Schools in the College of Computing as well as Schools in other Colleges will be considered.

Georgia Tech provides equal opportunity to all faculty, staff, students, and all other members of the Georgia Tech community, including applicants for admission and/or employment, contractors, volunteers, and participants in institutional programs, activities, or services. Georgia Tech complies with all applicable laws and regulations governing equal opportunity in the workplace and in educational activities. Georgia Tech prohibits discrimination, including discriminatory harassment, on the basis of race, ethnicity, ancestry, color, religion, sex (including pregnancy), sexual orientation, gender identity, national origin, age, disability, genetics, or veteran status in its programs, activities, employment, and admissions. This prohibition applies to faculty, staff, students, and all other members of the Georgia Tech community, including affiliates, invitees, and guests.

Rutgers University at New Brunswick
Tenure-Track Positions in Computer Science,

The Computer Science Department at Rutgers University invites applications for multiple tenure-track/tenured positions at the Assistant Professor and Associate Professor levels. We will consider outstanding candidates at the Professor level as well.

We invite applications in all areas of computer science. We are especially interested in the areas of trustworthy computing and human-centered computing. Broadly defined, trustworthy computing includes topics from cryptography and formal methods to system security, policy and privacy. Human-centered computing is also broadly defined, including and not limited to intelligent interaction (e.g., HRI), fairness and transparency in AI systems, AI for social good, as well as economics and computation.

Founded in 1966, the Department of Computer Science at Rutgers is a center for research, innovation and education both at the undergraduate and graduate levels. Strong research groups exist in areas of AI and robotics, foundations of computer science, scientific computing, and systems. Rutgers, the State University of New Jersey, stands among America's highest-ranked, most diverse public research universities: the oldest, largest, and top-ranked public university in the New York/New Jersey metropolitan area. Located on the Northeast Corridor train line, Rutgers is 50 minutes from New York City and in close proximity to the Philadelphia metropolitan area. The region features excellent public high schools and offers multiple opportunities for collaboration with corporate research and development centers.

We are strongly committed to increasing the diversity of our faculty. We welcome applications from all qualified candidates who want to join our community, including candidates with non-traditional career paths who have taken time off (e.g., to care for children or a family member in need) or who have achieved excellence in careers outside academia (e.g., in industry or government research).

Responsibilities will include research, supervision of Ph.D. students, and teaching undergraduate- and graduate-level courses in computer science. Pursuit of external research funding is expected.

Requirements: Successful completion of a Ph.D. or equivalent in computer science or a closely related field is required by the start date.

Timeline: The appointment will start September 1, 2022. Applications received by January 3, 2022, will be given priority.

How to Apply: Applicants should submit their cover letter, CV, a research statement addressing both past and future work, a teaching statement, and contact information for at least three references at https://jobs.rutgers.edu/postings/144584. Contact Info: hiring-committee@cs.rutgers.edu.

Rutgers Policies: Rutgers subscribes to the value of academic diversity and encourages applications from individuals with varied experiences, perspectives, and backgrounds. Women, minorities, and persons with disabilities are encouraged to apply. Rutgers is an affirmative action/equal opportunity employer. Offer is contingent upon successful completion of all pre-employment screenings.

San Diego State University
Tenure-Track Assistant Professor Position

The Department of Computer Science is seeking to hire a tenure-track assistant professor in computer systems and their applications beginning Fall 2022.

The candidates should have a PhD degree in Computer Science or a closely related field. Position details and instructions to apply can be found at https://apply.interfolio.com/96191.

Questions about the position may be directed to COS-CS-SA-Search2022@sdsu.edu.

University of Illinois at Chicago - College of Engineering

Open Rank - Multiple Tenure Track Faculty / Computer Science

Located in the heart of Chicago, the UIC CS department anticipates hiring multiple tenure track faculty at all ranks starting from Fall 2022 (with preference to candidates at the Assistant and Associate Professor ranks). Outstanding candidates in all areas who could complement and enhance current department strengths will be considered. Candidates working in Artificial Intelligence, Machine Learning, Data Science, Systems and Software, Computer Graphics, and related areas are especially encouraged to apply. Candidates should have a PhD in Computer Science, Data Science, Information Systems, or closely related fields, and the potential for excellence in teaching and research.

Applications must be submitted at https://jobs.uic.edu/, and must include a curriculum vitae, teaching and research statements, and names and addresses of at least three references. Links to a professional website such as Google Scholar or Research Gate are recommended. Applicants may contact the faculty search committee at cs-tt-search@uic.edu for more information. For fullest consideration, applications must be submitted by December 10, 2021. Applications will be accepted until the positions are filled.

The Department of Computer Science at UIC, which will be hiring between 15 and 30 new faculty in the next 5 years, has 40 tenure-system faculty, 4 research faculty with strong and broad research agendas, and 19 clinical/teaching faculty. The department is committed to building a diverse faculty preeminent in its missions of research, teaching, and service to the community. Candidates who have experience engaging with a diverse range of faculty, staff, and students, and contributing to a climate of inclusivity are encouraged to discuss their perspectives on these subjects in their application materials.

UIC is a major public research university (Carnegie R1) with about 3,100 faculty and 34,000 students. UIC is committed to increasing access to education, employment, programs, and services for all. UIC is committed to supporting the success of dual-career couples.

Chicago epitomizes the modern, livable, vibrant, and diverse city. World-class amenities like the lakefront, arts and culture venues, festivals, and two international airports make Chicago a singularly enjoyable place to live. Yet the cost of living, whether in an 88th floor condominium downtown or on a tree-lined street in one of the nation's finest school districts, is remarkably affordable.

Duties:
Teach, Conduct Research, Mentor Students

Qualifications:
PhD in Computer Science, Data Science, Information Systems or closely Related Field and the Potential for Excellence in Teaching and Research.

The University of Illinois at Chicago is an affirmative action, equal opportunity employer. All qualified applicants will receive consideration for employment without regard to race, color, religion, sex, gender identity, sexual orientation, national origin, protected veteran status, or status as an individual with a disability.

Offers of employment by the University of Illinois may be subject to approval by the University's Board of Trustees and are made contingent upon the candidate's successful completion of any criminal background checks and other pre-employment assessments that may be required for the position being offered. Additional information regarding such pre-employment checks and assessments may be provided as applicable during the hiring process.

The University of Illinois System requires candidates selected for hire to disclose any documented finding of sexual misconduct or sexual harassment and to authorize inquiries to current and former employers regarding findings of sexual misconduct or sexual harassment. For more information, visit https://www.hr.uillinois.edu/cms/One.aspx?portalId=4292&pageId=1411899

University of Illinois faculty, staff and students are required to be fully vaccinated against COVID-19. This employment offer is contingent on your timely submission of proof of your vaccination. If you are not able to receive the vaccine for medical or religious reasons, you may seek approval for an exemption in accordance with applicable University processes.

Penn Engineering
UNIVERSITY of PENNSYLVANIA

Open Faculty Positions in ESE
Multiple Faculty Positions

The School of Engineering and Applied Science at the University of Pennsylvania is growing its faculty by 33% over a five-year period. As part of this initiative, the Department of Electrical and Systems Engineering is engaged in an aggressive, multi-year hiring effort for multiple tenure-track positions at all levels. Candidates must hold a Ph.D. in Electrical Engineering, Computer Engineering, Systems Engineering, or related area. The department seeks individuals with exceptional promise for, or proven record of, research achievement, who will take a position of international leadership in defining their field of study and who will excel in undergraduate and graduate education. Leadership in cross-disciplinary and multi-disciplinary collaborations is of particular interest. We are interested in candidates in all areas that enhance our research strengths in:

1. Nanodevices and nanosystems (nanoelectronics, MEMS/NEMS, power electronics, nanophotonics, nanomagnetics, quantum devices, integrated devices and systems at nanoscale);

2. Circuits and computer engineering (analog, RF, mm-wave, digital circuits, emerging circuit design, computer engineering, IoT, beyond 5G, and cyber-physical systems);

3. Information and decision systems (control, optimization, robotics, data science, machine learning, communications, networking, information theory, signal processing).

Diversity candidates are strongly encouraged to apply. Interested persons should submit an online application by following the links above and include curriculum vitae, research, teaching, and diversity statements, and at least three references. Review of applications will begin on January 4, 2021.

https://apptrkr.com/2522477

Department of Electrical and Computer Engineering
Graduate School of Engineering and Management
Air Force Institute of Technology (AFIT)
Dayton, Ohio

Faculty Position

The Department of Electrical and Computer Engineering at the Air Force Institute of Technology is seeking applications for a tenured or tenure-track faculty position. All academic ranks will be considered. Applicants must have an earned doctorate in Electrical Engineering or a closely affiliated discipline by the time of their appointment (anticipated 1 September 2022).

We are particularly interested in applicants specializing in one or more of the following areas: radar cross section analysis, low observables, electromagnetic scattering analysis, computational electromagnetics, antennas and propagation, or microwave theory and measurements. Applicants having experience in the electromagnetic survivability community are highly desired. This position requires teaching at the graduate level as well as establishing and sustaining a strong Department of Defense relevant externally funded research program with a sustainable record of related peer-reviewed publications.

The Air Force Institute of Technology (AFIT) is the premier Department of Defense institution for graduate education in science, technology, engineering, and management, and has a Carnegie Classification as a High Research Activity Doctoral University. The Department of Electrical and Computer Engineering offers accredited M.S. and Ph.D. degree programs in Electrical Engineering, Computer Engineering, and Computer Science as well as an MS degree program in Cyber Operations.

For more information on the position and how to apply, please visit *https://www.usajobs.gov/GetJob/ViewDetails/jobadnumber*. Be sure to include

- A letter of application to include the USA Jobs announcement number jobadnumber.
- Your curriculum vitae (no photographs please).
- Transcripts for all degrees listed on curriculum vitae (official copies must follow).
- A statement of your research plans (limited to one page) and a statement of your teaching philosophy at the graduate level (limited to one page).
- A list of three professional references including name, complete mailing address, email address, and phone number.

Applicants must be U.S. citizens and currently hold or be able to obtain a security clearance. More information on AFIT and the Department of Electrical and Computer Engineering can be found at *http://www.afit.edu/ENG/*. Review of applications will begin on January 3, 2022. The United States Air Force is an equal opportunity, affirmative action employer.

University of Illinois at Chicago - College of Engineering

Open Rank – Multiple Non-Tenure Track Faculty / Computer Science

The Computer Science Department at the University of Illinois Chicago (UIC) seeks one or more full-time teaching faculty members to fill one of two possible positions – Lecturer or Clinical Professor. Candidates would work alongside 19 full-time teaching faculty with over 150 years of combined experience and 12 awards for excellence. Standard teaching load is one to three undergraduate courses per term, depending on enrollment. Areas of interest include introductory programming, data structures, computer organization/systems, web development, data science, software engineering, and machine learning.

The Lecturer track is a long-term career track that starts with the Lecturer position and offers opportunities for advancement to Senior Lecturer. Minimum qualifications include an MS in Computer Science or a closely related field.

The Clinical Professor track is a long-term career track that starts with the Clinical Assistant Professor position and offers advancement to Clinical Associate and Clinical Full Professor. Minimum qualifications include a PhD in Computer Science or a closely related field. Candidates interested in Computer Science Education research are encouraged to apply.

The department seeks candidates dedicated to teaching. Candidates for either position must have either (a) demonstrated evidence of effective teaching or (b) convincing argument of future dedication and success in the art of teaching.

UIC is one of the top-ten most diverse universities in the US (US News and World Report), a top 25 public and top 10 best value (Wall Street Journal and Times Higher Education), and a Hispanic-serving institution. Chicago epitomizes the modern, livable, vibrant city. Located on the shore of Lake Michigan, Chicago offers an outstanding array of cultural, culinary, recreational, and sporting experiences. In addition to the lakefront, theater, and many ethnic districts, Chicago boasts one of the world's tallest and densest skylines, an 8100-acre park system, professional teams in all major sports, and extensive public transit and biking networks.

Submit applications online at https://jobs.uic.edu. Include a curriculum vitae, names & addresses of at least three references, a statement providing evidence of effective teaching, a statement describing past experience in activities that promote diversity and inclusion (or plans to make future contributions), recordings of recent teaching activities either in-person or online, and recent teaching evaluations. For more information, send e-mail to cs-ntt-search@uic.edu. For fullest consideration, apply by 11/1/2021. Applications will be accepted and reviewed until the positions are filled.

The University of Illinois at Chicago is an affirmative action, equal opportunity employer. All qualified applicants will receive consideration for employment without regard to race, color, religion, sex, gender identity, sexual orientation, national origin, protected veteran status, or status as an individual with a disability.

Offers of employment by the University of Illinois may be subject to approval by the University's Board of Trustees and are made

VIRGINIA TECH™

FACULTY POSITIONS

Department of Computer Science

The Department of Computer Science at Virginia Tech invites applications for eight (8) tenure-track or tenured faculty positions at all ranks (Assistant, Associate, or Full Professor) in all areas of computer science. The department is in a period of rapid growth and expanding opportunity. We are seeking candidates motivated to contribute to a collegial, interdisciplinary community with a strong tradition of both fundamental and applied research. We embrace Virginia Tech's motto, *Ut Prosim* ("That I May Serve"): we are committed to research, education, service, and inclusivity that makes a positive difference in the lives of people, communities, and the world.

The department currently has 67 faculty members, including 56 tenured or tenure-track faculty, 17 early career awardees, and numerous recipients of faculty awards from IBM, Intel, AMD, Microsoft, Google, Facebook, and others. CS faculty members direct several interdisciplinary research centers, including the Center for Human-Computer Interaction and the Sanghani Center for Artificial Intelligence & Data Analytics. The department is home to over 1,400 undergraduate majors and over 600 graduate students and is located in the College of Engineering, whose undergraduate program ranks 13th and graduate program ranks 31st among all U.S. engineering schools *(USN&WR)*. The Mission of the College of Engineering is to educate and inspire our students to be critical thinkers, innovators and leaders. Our core values are inclusiveness, excellence, integrity, perseverance and stewardship.

Virginia Tech's main campus is located in Blacksburg, VA, in an area consistently ranked among the country's best places to live. Our program in the Washington, D.C., area is also expanding rapidly, with Virginia Tech's exciting new Innovation Campus in Alexandria, VA, slated to open in 2024. Candidates for faculty positions at the Innovation Campus are encouraged to apply to separate announcements for those opportunities.

The successful candidate will have a Doctoral degree in computer science or a closely related field at the time of appointment, a rank appropriate record of academic accomplishments and a proven ability to work collaboratively; a commitment to interdisciplinary research and instruction and a willingness to expand disciplinary boundaries to address complex technical and societal challenges. Tenured and tenure-track faculty are expected to initiate and develop independent research that is internationally recognized for excellence, conscientiously mentor research-oriented graduate students, teach effectively at both graduate and undergraduate levels, and serve the university and their professional communities. The successful candidate will be required to have a criminal conviction check as well as documentation of COVID-19 vaccination or receive approval from the university for a vaccination exemption due to a medical condition or sincerely held religious belief. The positions require occasional travel to professional meetings.

Applicants must apply online at **jobs.vt.edu** (job number **517689**): application materials include a cover letter; curriculum vitae; statements discussing teaching and research goals; a statement on contributions to advancing diversity, equity, and inclusion; and contact information for at least three references. Review of applications will commence on November 20, 2021 and continue until the positions are filled. Questions regarding the positions should be directed to Dr. Ali R. Butt at **facdev@cs.vt.edu**.

The department fully embraces Virginia Tech's commitment to increase faculty, staff, and student diversity; to ensure a welcoming, affirming, safe, and accessible campus climate; to advance our research, teaching, and service mission through inclusive excellence; and to promote sustainable transformation through institutionalized structures. Virginia Tech does not discriminate against employees, students, or applicants on the basis of age, color, disability, sex (including pregnancy), gender, gender identity, gender expression, genetic information, national origin, political affiliation, race, religion, sexual orientation, or veteran status, or otherwise discriminate against employees or applicants who inquire about, discuss, or disclose their compensation or the compensation of other employees or applicants, or on any other basis protected by law. If you are an individual with a disability and desire an accommodation, please contact Joan Watson at **jmwatson@vt.edu** during regular business hours at least 10 business days prior to the event.

contingent upon the candidate's successful completion of any criminal background checks and other pre-employment assessments that may be required for the position being offered. Additional information regarding such pre-employment checks and assessments may be provided as applicable during the hiring process.

The University of Illinois System requires candidates selected for hire to disclose any documented finding of sexual misconduct or sexual harassment and to authorize inquiries to current and former employers regarding findings of sexual misconduct or sexual harassment. For more information, visit https://www.hr.uillinois.edu/cms/One.aspx?portalId=4292&pageId=1411899

University of Illinois faculty, staff and students are required to be fully vaccinated against COVID-19. This employment offer is contingent on your timely submission of proof of your vaccination. If you are not able to receive the vaccine for medical or religious reasons, you may seek approval for an exemption in accordance with applicable University processes.

University of Mary Washington

Professor of Computer Science

The University of Mary Washington (UMW) invites applications for a tenure-track position at the assistant or associate professor level in computer science to begin August 15, 2022.

Our close-knit team of ten faculty specializes in working with eager undergraduate students in small classes and one-on-one mentoring settings. Candidates for the position must either have earned a Ph.D. in Computer Science or a closely related field or have made substantial progress toward completing their dissertation. An earned terminal degree will be required for continuation in the tenure-track position. Passionate teachers in all subdisciplines of computer science will be considered. Successful applicants will demonstrate an enthusiasm for teaching at the undergraduate level, the ability to teach a variety of computer science topics, an interest in involving undergraduates in research, and demonstrate dedication to effective teaching and high-impact learning in an inclusive environment that embraces diverse talents and backgrounds. The normal teaching load is three sections per semester in small classes.

Faculty are expected to engage in scholarly/professional development, perform service in support of the students, the program, and the institution, and be committed to the continuous enhancement of a diverse and inclusive curriculum in response to changing needs and expectations of our student body.

UMW is a public liberal arts and sciences university dedicated to effective teaching and the integration of undergraduate students in research. The Department offers majors in Computer Science and Cybersecurity, as well as minors in Computer Science, Cybersecurity and Data Science. Recently recognized as a "Great College to Work For" by the Chronicle of Higher Education, UMW is centrally located between Washington, DC, and Richmond, VA, close to many government labs and private engineering firms offering opportunities for collaborative research and student internship opportunities.

Candidates are required to complete a Commonwealth of Virginia application. They are also asked to electronically submit a letter of application, curriculum vitae, statement of teaching philosophy, research statement, copies of graduate transcripts and the contact information for three references.

The preferred deadline for applications is December 31, 2021. The University of Mary Washington accepts only completed online applications. Faxed, mailed, or e-mailed applications will not be considered. Please apply at https://careers.umw.edu/.

In a continuing effort to enrich its academic environment and provide equal educational and employment opportunities, the University of Mary Washington actively encourages women, minorities, disabled individuals, and veterans to apply.

For questions regarding the position, please feel free to contact Dr. Jennifer Polack, Search Committee Chair, Department of Computer Science, College of Arts and Sciences at polack@umw.edu.

ADVERTISING IN CAREER OPPORTUNITIES

How to Submit a Classified Line Ad: Send an e-mail to *acmmediasales@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 by 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:
acmmediasales@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
acmmediasales@acm.org

University of Michigan – Dearborn

Assistant Professor in Computer and Information Science

The Department of Computer and Information Science (CIS) at the University of Michigan - Dearborn invites applications for one tenure-track Assistant Professor position. Applicants in areas related to AI Systems (e.g., intelligent agents, natural language processing, AI software development, AI-powered analytics, etc.) or Visual Computing (e.g., computer graphics, game design, visualization, virtual and augmented reality, etc.) will be considered. The expected starting date is September 1, 2022. Although candidates at the Assistant Professor rank are preferred, exceptional candidates may be considered for the rank of Associate Professor depending upon experience and qualifications. We offer competitive salaries and start-up packages.

The CIS Department offers several B.S. and M.S. degrees, and a Ph.D. degree. The current research areas in the department include artificial intelligence, computational game theory, computer graphics, cybersecurity, data privacy, data science/management, energy-efficient systems, game design, graphical models, machine learning, multimedia, natural language processing, networking, service and cloud computing, software engineering, and health informatics. These areas of research are supported by several established labs and many of these areas are currently funded by federal agencies and industries.

The department and College of Engineering of Computer Science value a culture of diversity, equity, and inclusion. We are committed to the development of diverse and culturally intelligent faculty who thrive and contribute to a positive and inclusive environment.

Qualifications:
Qualified candidates must have earned a Ph.D. degree in computer science or a closely related discipline by September 1, 2022. Candidates will be expected to do scholarly and sponsored research, as well as teaching at both the undergraduate and graduate levels.

Applications:
Applicants should send a cover letter; curriculum vitae; statements of teaching, research interests, and diversity; evidence of teaching performance (if any); and a list of three references through Interfolio at:

http://apply.interfolio.com/93511 for the position.

Review of applications will start on January 15, 2022, but applications will be accepted until the position is filled.

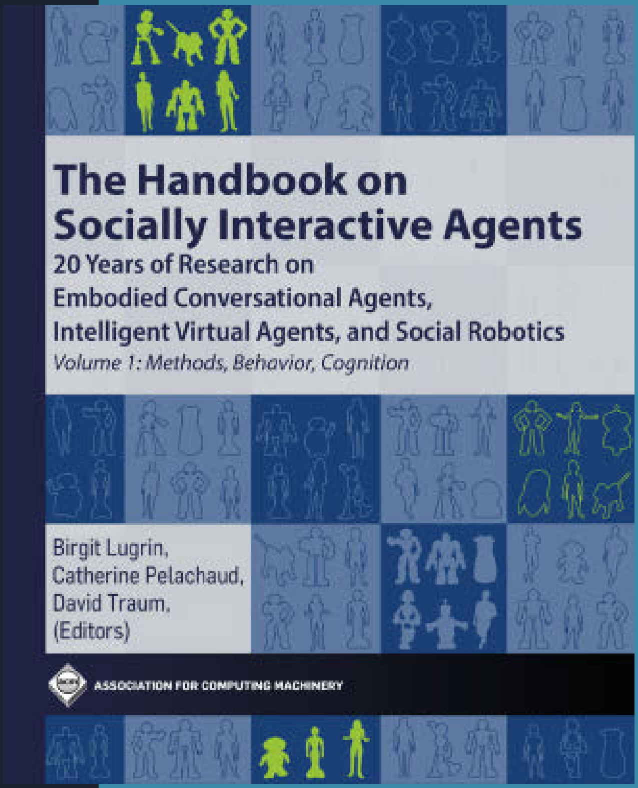
The University of Michigan-Dearborn is an equal opportunity/affirmative action employer.

ACM BOOKS
Collection II

The Handbook on Socially Interactive Agents provides a comprehensive overview of the research fields of Embodied Conversational Agents, Intelligent Virtual Agents, and Social Robotics. Socially Interactive Agents (SIAs), whether virtually or physically embodied, are autonomous agents that are able to perceive an environment including people or other agents, reason, decide how to interact, and express attitudes such as emotions, engagement, or empathy. They are capable of interacting with people and one another in a socially intelligent manner using multimodal communicative behaviors, with the goal to support humans in various domains.

Written by international experts in their respective fields, the book summarizes research in the many important research communities pertinent for SIAs, while discussing current challenges and future directions. The handbook provides easy access to modeling and studying SIAs for researchers and students, and aims at further bridging the gap between the research communities involved.

In two volumes, the book clearly structures the vast body of research. The first volume starts by introducing what is involved in SIAs research, in particular research methodologies and ethical implications of developing SIAs. It further examines research on appearance and behavior, focusing on multimodality. Finally, social cognition for SIAs is investigated using different theoretical models and phenomena such as theory of mind or pro-sociality. The second volume starts with perspectives on interaction, examined from different angles such as interaction in social space, group interaction, or long-term interaction. It also includes an extensive overview summarizing research and systems of human–agent platforms and of some of the major application areas of SIAs such as education, aging support, autism, and games.



The Handbook on Socially Interactive Agents
20 Years of Research on Embodied Conversational Agents, Intelligent Virtual Agents, and Social Robotics
Volume 1: Methods, Behavior, Cognition

Birgit Lugrin,
Catherine Pelachaud,
David Traum,
(Editors)

ASSOCIATION FOR COMPUTING MACHINERY

The Handbook on Socially Interactive Agents

20 Years of Research on Embodied Conversational Agents, Intelligent Virtual Agents, and Social Robotics

Edited by
Birgit Lugrin
Catherine Pelachaud
David Traum

ISBN: 978-1-4503-8721-7
DOI: 10.1145/3477322

http://books.acm.org
http://store.morganclaypool.com/acm

Introducing *ACM Transactions on Human-Robot Interaction*

Now accepting submissions to ACM THRI

In January 2018, the *Journal of Human-Robot Interaction* (JHRI) became an ACM publication and was rebranded as the *ACM Transactions on Human-Robot Interaction* (THRI). It will continue to be open access, fostering the widest possible readership of HRI research and information. All issues will be available in the ACM Digital Library.

ACM THRI aims to be the leading peer-reviewed interdisciplinary journal of human-robot interaction. Publication preference is given to articles that contribute to the state of the art or advance general knowledge, have broad interest, and are written to be intelligible to a wide range of audiences. Submitted articles must achieve a high standard of scholarship. Accepted papers must: (1) advance understanding in the field of human-robot interaction, (2) add state-of-the-art or general information to this field, or (3) challenge existing understandings in this area of research.

2017
Volume 6, Number 1

ACM Transactions on
Human-Robot Interaction

Article 1 O. C. Jenkins, S. Sabanovic — Editorial Introduction: Impact, Sustainability, and Inclusion for JHRI

Article 2 F. Levillain, E. Zibetti — Behavioral Objects: The Rise of the Evocative Machines

Article 3 H. Admoni, B. Scassellati — Social Eye Gaze in Human-Robot Interaction: A Review

Article 4 G. Briggs, T. Williams, M. Scheutz — Enabling Robots to Understand Indirect Speech Acts in Task-Based Interactions

Association for Computing Machinery
Advancing Computing as a Science & Profession

ACM THRI encourages submission of well-written papers from all fields, including robotics, computer science, engineering, design, and the behavioral and social sciences. Published scholarly papers can address topics including how people interact with robots and robotic technologies, how to improve these interactions and make new kinds of interaction possible, and the effects of such interactions on organizations or society. The editors are also interested in receiving proposals for special issues on particular technical problems or that leverage research in HRI to advance other areas such as social computing, consumer behavior, health, and education.

The inaugural issue of the rebranded *ACM Transactions on Human-Robot Interaction* has been published and can be found in the ACM Digital Library.

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

Association for
Computing Machinery

[CONTINUED FROM P. 120]

You have since gone on to produce groundbreaking results in quantum supremacy.

Around 2008 or 2009, I got interested in just how hard quantum computations can be to simulate classically. Forget whether the quantum computer is doing anything useful; how strong can we make the evidence that a quantum computation is hard to simulate? It turns out—and there were others who came to the same realization around the same time—if that is your goal, you can get enormous leverage by switching attention from problems like factoring, which have a single right answer, to sampling problems, where the goal of your computation is just to output a sample from some probability distribution over strings of N bits.

There are certain probability distributions that a quantum computer can easily sample from.

Not only that, but a pretty rudimentary quantum computer. If a classical computer could efficiently sample the same distribution in polynomial time, then the polynomial hierarchy would collapse to the third level, which we use as a kind of standard yardstick of implausibility.

But if you want to do an experiment to demonstrate quantum supremacy, it's not enough to have a distribution that's hard for a classical computer to sample exactly. Any quantum computer is going to have a huge amount of noise, so the most you could hope for is to sample from some distribution that's close to the ideal one. And how hard is it for a classical computer to approximately sample from the same distribution?

To answer that, we realized you needed a quantum computation where the amplitudes (related to probabilities) for the different possible outputs are what's called "robustly #P-complete," meaning that if you could just approximate most of them, then you could solve any combinatorial counting problem.

And bosons fit that bill perfectly.

Fermions and bosons are the two basic types of particles in the universe. If you have a bunch of identical fermions, and you want to know their quantum amplitude to get from an input state to an output state, then you have to calculate the determinant of a matrix. If they're bosons, then it becomes the permanent of the matrix. Now, the determinant and the permanent are two of the most famous functions in theoretical computer science, for reasons having nothing to do with physics. The determinant is easy to compute, but the permanent is #P-complete.

> **“We're in an era now where it's a real fight, on these special sampling benchmarks, beween the state-of-the-art quantum computers and the largest classical supercomputers."**

I remembered that from a talk that Avi Widgerson gave in 2000, where he'd said, "Isn't it unfair that the bosons have to work so much harder than the fermions to figure out what they're going to be doing?"

The joke stuck with me, but I knew that the permanent had exactly the right sort of properties, being robustly #P-complete. And I said, "Well, what if we could design a quantum computation where the amplitudes would be permanents of matrices? Well, I guess we would use bosons. Now I'd better learn what these bosons are and how physicists model them."

Eventually, you and your student, Alex Arkhipov, put forward a new conjecture that approximate boson sampling would be hard for a classical computer to simulate.

We came at it purely for theoretical reasons. But afterward, we gave talks about it, and the experimental physicists started getting all excited. Especially the quantum optics people, because photons are bosons, and they are dealing all the time with generating a bunch of identical photons, sending them through a network of beamsplitters, and measuring them.

And so there was a race to do it.

The first paper, in 2013, was from a group in Australia that did it with just three photons and confirmed experimentally that the amplitudes were permanents of three-by-three matrices.

Now, no classical computer is going to break a sweat calculating a three-by-three permanent. Basically, if I want to calculate the permanent of an N-by-N matrix, the difficulty is going to grow roughly two to the power of the number of photons using the best classical algorithms that we know. If you want to outperform any classical computer on earth, you're going to want 50 or 60 photons. And now you're talking about a pretty hard experiment, because there's no photon source on earth that will generate one photon on demand, exactly when you want it.

But then there was an incredibly interesting interaction between the theorists and the experimentalists—a huge effort to meet in the middle with something that the experimentalists could actually do and that we would agree seems hard for a classical computer.

In 2019, Google's Quantum AI lab announced it had demonstrated quantum supremacy with 53 superconducting qubits, and you wrote a paper with Lijie Chen that gave some theoretical evidence to support the claim.

Yes, we developed the theory of those experiments. The best classical simulations we know take exponential time, and we gave some evidence that this is inherent. But it's still debated. IBM and others have said they could simulate Google's results with a classical computer a lot faster than Google thought they could be simulated, albeit still not breaking the exponential barrier.

We're now in the era where it's a real fight, on these special sampling benchmarks, between the state-of-the art quantum computers and the largest classical supercomputers. I expect the quantum computers will pull ahead soon. If there is a real fight today, it suggests that in a few years' time, there's not going to be a real fight anymore.

Leah Hoffmann is a technology writer based in Piermont, NY, USA.

© 2021 ACM 0001/0782/21/12

DOI:10.1145/3490319

Leah Hoffmann

Q&A
Exploring the Promise of Quantum Computing

ACM Computing Prize recipient Scott Aaronson discusses his work in quantum complexity.

WE HAVE NOT yet have realized—or, perhaps, even fully understood—the full promise of quantum computing. However, we have gotten a much clearer view of the technology's potential, thanks to the work of ACM Computing Prize recipient Scott Aaronson, who has helped establish many of the theoretical foundations of quantum supremacy and illuminated what quantum computers eventually will be able to do. Here, Aaronson talks about his work in quantum complexity.

Let's start with your first significant result in quantum computing: your work on the collision problem, which you completed in graduate school.

The collision problem is where you have a many-to-one function, and your task is just to find any collision pair, meaning any two inputs that map to the same output. I proved that even a quantum computer needs to access the function many times to solve this problem.

It's a type of problem that shows up in many different settings in cryptography. How did you come to it?

When I entered the field, in the late 1990s, I got very interested in understanding quantum computers by looking at how many queries they have to make to learn some property of a function. This is a subject called query complexity, and it's actually the source of the majority of what we know about quantum algorithms. Because you're only counting the number of accesses to an input, you're not up against the P vs. NP problem. But you *are* fighting against a quantum computer, which can make a superposition of queries and give a superposition of answers. And sometimes quantum computers can exploit structure in a function in order to learn something with exponentially fewer queries than any classical algorithm would need.

So, what kind of structure does your problem need before a quantum computer can exploit it to get this exponential speed-up?

That's exactly what we've been working on for the past 30 years. In 1995, Peter Shor showed that quantum computers are incredibly good at extracting the period of a periodic function. Others showed that, if you're just searching for a single input that your function maps to a designated output, then quantum computers give only a modest, square-root improvement. The collision problem was interesting precisely because it seemed poised between these two extremes: it had less structure than the period-finding problem, but more structure than the "needle in a haystack" problem.

When my advisor, Umesh Vazirani, told me that the collision problem was his favorite problem in quantum query complexity, I said, "Okay, well, I'll know not to work on that one, because that one's too large." But it kept coming up in other contexts that I cared about. I spent a summer at Caltech and I decided to try to attack it.

I had a colleague, Andris Ambainis, who had invented this amazing lower bound technique—what's now called the Ambainis adversary method—a couple years prior. I didn't know it at the time, but he had actually invented it to try to solve the collision problem, though he was not able to make it work for that. But he could solve some problems that I couldn't solve using this method that I understood really well, called the polynomial method. I started trying to use Ambainis' method to attack the collision problem. I worked on it probably more intensely than I've worked on anything before or since, and what I finally realized was that the polynomial method would work to prove a lower bound for the problem and show that even a quantum computer needs at least $N^{1/5}$ queries to solve it, where N is the number of outputs. Shortly afterward, Yaoyun Shi refined the technique and was able to show, first, that you need $N^{1/4}$ queries, and then that you need $N^{1/3}$.

[CONTINUED ON P. 119]

PHOTO BY SASHA HAAGENSEN PHOTOGRAPHY, COURTESY OF THE UNIVERSITY OF TEXAS AT AUSTIN

Crossroads The ACM Magazine for Students
XRDS
Combating Fake News and Misinformation
The Bad News Game
Detecting Deepfakes
How to Tame Your Chatbot

Crossroads The ACM Magazine for Students
XRDS
View Source
Be a new source
Unlocking the Black Box
Libera Tu Radio
Wikipedia's Editathons Experience

Crossroads The ACM Magazine for Students
XRDS
Changing Healthcare
Digitization in the Healthcare Industry
Thermalytix—Using AI to Save Lives

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

acm
Association for
Computing Machinery

MATLAB SPEAKS

MACHINE LEARNING

With MATLAB® you can use clustering, regression, classification, and deep learning to build predictive models and put them into production.

mathworks.com/machinelearning

MathWorks®

©2021 The MathWorks, Inc.