

COMMUNICATIONS

CACM.ACM.ORG

OF THE

ACM

05/2020 VOL.63 NO.05

Frontiers of Fairness in Machine Learning

ACM's 2020 General Election

The Critical Role of Human
Performance in Software

Your Kindle May Be Spying on You,
But You Can't Be Sure

ACM Publications Finances

Association for
Computing Machinery



volume
01

number
01

FIRST
ISSUE
PUBLISHED

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



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



THE ACM A. M. TURING AWARD

by the community ♦ from the community ♦ for the community



ACM and Google congratulate

EDWIN CATMULL and PATRICK HANRAHAN

**For fundamental contributions to 3D computer graphics,
and the revolutionary impact of these techniques on CGI
in filmmaking and other applications.**



“Because 3-D computer graphic imagery is now so pervasive, we often forget what the field was like just a short time ago when a video game like Pong, which consisted of a white dot bouncing between two vertical white lines, was the leading-edge technology. The technology keeps moving forward, yet what Pat Hanrahan and Ed Catmull developed decades ago remains standard practice in the field today—that’s quite impressive. It’s important to recognize scientific contributions in CGI technology and educate the public about a discipline that will impact many areas in the coming years—virtual and augmented reality, data visualization, education, medical imaging, and more.”

Jeff Dean
Google Senior Fellow and SVP of Google AI
Google Inc.

Google™

For more information see <http://research.google.com/>

Financial support for the ACM A. M. Turing Award is provided by Google Inc.

Departments

- 5 **From the President**
How ACM Is Adapting in This Period of Global Uncertainties
By Cherri M. Pancake
-
- 7 **From the Chair of ACM-W**
Partnerships Can Help Drive Gender Equality
By Jodi Tims
-
- 9 **Vardi's Insights**
Efficiency vs. Resilience: What COVID-19 Teaches Computing
By Moshe Y. Vardi
-
- 11 **ACM's 2020 General Election**
Meet the candidates who introduce their plans—and stands—for the Association.
-
- 22 **Career Paths in Computing**
Launching a New Feature of Communications
By Andrew A. Chien and Mei Kobayashi
Computing enabled me to...
Automating Automation: CS at the Heart of the Manufacturing Economy
By Arquimedes Canedo
-
- 24 **BLOG@CACM**
Teaching CS Humbly, and Watching the AI Revolution
Mark Guzdial on a book that changed his thinking about teaching computer science, and Jiajie Zhang on the AI Revolution.
-
- 47 **Calendar**
-
- 53 **ACM Publications Finances**
By Jack W. Davidson, Joseph A. Konstan, and Scott E. Delman

Last Byte

- 112 **Upstart Puzzles**
Optimal Chimes
The importance of the space between the notes.
By Dennis Shasha

News



- 27 **A Proof from 'The Book'**
A decades-old conjecture about computational complexity is confirmed in just a few pages.
By Don Monroe
-
- 30 **Will RISC-V Revolutionize Computing?**
The open instruction set for microprocessors promises to reshape computing and introduce new, more powerful capabilities.
By Samuel Greengard
-
- 33 **Deceiving the Masses on Social Media**
The social media platforms like their freedom, but information gerrymandering may require legislation to fix.
By Keith Kirkpatrick

Viewpoints

- 36 **Law and Technology**
What Role for Antitrust in Regulating Platforms?
Using regulation to protect competition and innovation.
By C. Scott Hemphill
-
- 39 **Privacy and Security**
Secure Development Tools and Techniques Need More Research That Will Increase Their Impact and Effectiveness in Practice
Secure development is an important and pressing problem.
By Adam Shostack and Mary Ellen Zurko
-
- 42 **Education**
A Vision of K-12 Computer Science Education for 2030
Exploring goals, perspectives, and challenges.
By Mike Tissenbaum and Anne Ottenbreit-Leftwich
-
- 45 **Viewpoint**
Computers Do Not Make Art, People Do
The continually evolving relationship between artistic technologies and artists.
By Aaron Hertzmann
-
- 49 **Viewpoint**
When Technology Goes Awry
On engineers' obligation to tame their creations.
By Cal Newport

Practice



64

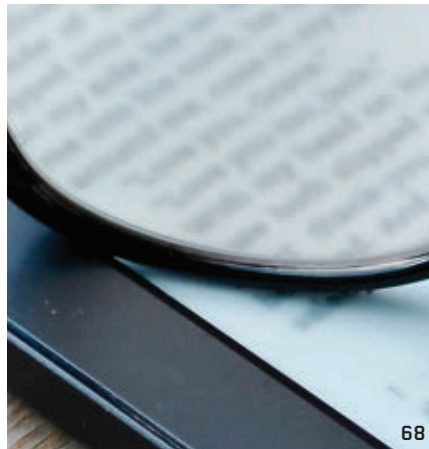
- 58 **Beyond the ‘Fix-It’ Treadmill**
The use of post-incident artifacts in high-performing organizations.
By J. Paul Reed
- 64 **Revealing the Critical Role of Human Performance in Software**
It’s time to appreciate the human side of Internet-facing software systems.
By David D. Woods and John Allspaw

Q Articles’ development led by **acmqueue**
queue.acm.org



About the Cover: Fairness and transparency is a burgeoning field of study, and this month’s cover story explores the findings of a group of experts from academia, industry, and government who assess their understanding of the nascent science of fairness in machine learning. Cover illustration by Justin Metz.

Contributed Articles



68

- 68 **Reading in the Panopticon—Your Kindle May Be Spying on You, But You Can’t Be Sure**
eBooks *may* have surveillance technologies embedded in them. Should we care?
By Stephen B. Wicker and Dipayan Ghosh
- 74 **A Bibliometric Approach for Detecting the Gender Gap in Computer Science**
Identifying female CS scientists by combining a robust bibliographic database and name filtering tools.
By Sandra Mattauch, Katja Lohmann, Frank Hannig, Daniel Lohmann, and Jürgen Teich



Watch the authors discuss this work in the exclusive *Communications* video.
<https://cacm.acm.org/videos/gender-gap>

Review Articles

- 82 **A Snapshot of the Frontiers of Fairness in Machine Learning**
A group of industry, academic, and government experts convene in Philadelphia to explore the roots of algorithmic bias.
By Alexandra Chouldechova and Aaron Roth



Watch the authors discuss this work in the exclusive *Communications* video.
<https://cacm.acm.org/videos/frontiers-of-fairness>

- 90 **Indistinguishability**
Diverse examples depict how indistinguishability plays a central role in computer science.
By Hagit Attiya and Sergio Rajsbaum

Research Highlights

- 102 **Technical Perspective**
Fake ‘Likes’ and Targeting Collusion Networks
By Geoffrey M. Voelker
- 103 **Measuring and Mitigating OAuth Access Token Abuse by Collusion Networks**
By Shehroze Farooqi, Fareed Zaffar, Nektarios Leontiadis, and Zubair Shafiq



Association for Computing Machinery
Advancing Computing as a Science & Profession



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

Darren Ramdin

Director, Office of SIG Services

Donna Cappel

Director, Office of Publications

Scott E. Delman

ACM COUNCIL

President

Cherri M. Pancake

Vice-President

Elizabeth Churchill

Secretary/Treasurer

Yannis Ioannidis

Past President

Alexander L. Wolf

Chair, SGB Board

Jeff Jortner

Co-Chairs, Publications Board

Jack Davidson and Joseph Konstan

Members-at-Large

Gabriele Kotsis; Susan Dumais; Renée McCauley; Claudia Bauzer Medeiros; Elizabeth D. Mynatt; Pamela Samuelson; Theo Schlossnagle; Eugene H. Spafford
SGB Council Representatives
Sarita Adve and Jeanna Neefe Matthews

BOARD CHAIRS

Education Board

Mehran Sahami and Jane Chu Prey

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

Jack Davidson and Joseph Konstan

Board Members

Phoebe Ayers; Nicole Forsgren; Chris Hankin; Mike Heroux; Nenad Medvidovic; Tulika Mitra; Michael L. Nelson; Sharon Oviatt; Eugene H. Spafford; Stephen N. Spencer; Divesh Srivastava; Robert Walker; Julie R. Williamson

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

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.

STAFF

DIRECTOR OF PUBLICATIONS

Scott E. Delman
cacm-publisher@cacm.acm.org

Executive Editor

Diane Crawford

Managing Editor

Thomas E. Lambert

Senior Editor

Andrew Rosenbloom

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

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 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 acmm mediasales@acm.org

Association for Computing Machinery (ACM)

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

EDITORIAL BOARD

EDITOR-IN-CHIEF

Andrew A. Chien
aic@cacm.acm.org

Deputy to the Editor-in-Chief

Morgan Denlow
cacm.deputy.to.aic@gmail.com

SENIOR EDITOR

Moshe Y. Vardi

NEWS

Co-Chairs

Marc Snir and Alain Chesnais

Board Members

Tom Conte; Monica Divitini; Mei Kobayashi;
Rajeev Rastogi; François Sillion

VIEWPOINTS

Co-Chairs

Tim Finin; Susanne E. Hambrusch;
John Leslie King

Board Members

Terry Benzel; Michael L. Best; Judith Bishop;
Lorrie Cranor; Boi Falting; James Grimmelmann;
Mark Guzdial; Haym B. Hirsch;
Richard Ladner; Carl Landwehr; Beng Chin Ooi;
Francesca Rossi; Len Shustek; Loren Terveen;
Marshall Van Alstyne; Jeannette Wing;
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; Kim Bruce; Alan Bundy;
Peter Buneman; Jeff Chase;
Yannis Ioannidis; Gal A. Kaminka;
Ben C. Lee; Igor Markov;
Lionel M. Ni; Doina Precup;
Shankar Sastry; m.c. schraefel; Ron Shamir;
Hannes Werthner; Reinhard Wilhelm

RESEARCH HIGHLIGHTS

Co-Chairs

Azer Bestavros; Shriram Krishnamurthi,
and Orna Kupferman

Board Members

Martin Abadi; Amr El Abbadi;
Animashree Anandkumar; Sanjeev Arora;
Michael Backes; Maria-Florina Balcan;
David Brooks; Stuart K. Card; Jon Crowcroft;
Alexei Efros; Bryan Ford; Alon Halevy;
Gernot Heiser; Takeo Igarashi;
Srinivasan Keshav; Sven Koenig;
Ran Libeskind-Hadas; Karen Liu; Greg Morrisett;
Tim Roughgarden; Guy Steele, Jr.;
Robert Williamson; Margaret H. Wright;
Nicolai Zeldovich; Andreas Zeller

SPECIAL SECTIONS

Co-Chairs

Sriram Rajamani, Jakob Rehof,
and Haibo Chen

Board Members

Tao Xie; Kenjiro Taura; David Padua

ACM Copyright Notice

Copyright © 2020 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.

ACM Media Advertising Policy

Communications of the ACM and other ACM Media publications accept advertising in both print and electronic formats. All advertising in ACM Media publications is at the discretion of ACM and is intended to provide financial support for the various activities and services for ACM members. Current advertising rates can be found by visiting <http://www.acm-media.org> or by contacting ACM Media Sales at (212) 626-0686.

Single Copies

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





Cherri M. Pancake

DOI:10.1145/3392777

How ACM Is Adapting in This Period of Global Uncertainties

As the world continues its efforts to slow the COVID-19 pandemic, our daily lives and routines have changed dramatically. Companies have closed their physical

locations and transitioned staff to working remotely, academic institutions—from primary school to universities—have sent students home and are turning exclusively to distance learning, and entire industries have scaled back in an effort to remain viable until the pandemic is contained. Both individually and collectively we are confronting personal, societal, and ethical dilemmas at a speed and scale few could foresee.

The global impact of computing and information technology has never been clearer than it is during this crisis. Real-time data sharing and computational modeling are central to efforts to analyze the spread of the virus, share public health strategies and experiences, and expedite development of new treatments and vaccines. At the same time, the use of social interaction platforms of all types has skyrocketed. That infrastructure is critical to keep families, friends, and colleagues connected, as well as to keep businesses, education, and government viable.

ACM's mission—to help unite the computing community, provide professional and technical information to those who need it, and advance the field—may be more relevant now than ever before. All of us at ACM are committed to continue supporting that mission throughout these difficult times. Here's an update on how ACM has responded to the crisis situation.

Like many other companies and non-profits, ACM has temporarily

closed its New York City Headquarters and is functioning as a virtual organization, but all of our units are operating and will continue to do so. Of course, recognizing the need for social distancing, we have had to postpone a growing number of ACM meetings and conferences or transition them to virtual meetings. Our volunteers and conference leaders are having to make some very difficult decisions. Fortunately, ACM and its SIGs are financially prepared to weather the storm, and we are working actively to develop technical options and practical guidance to help conference organizers modify their plans.

Our publication activities continue unabated, with all magazines, journals, conference proceedings, books, and

As a gesture of support, ACM has “unlocked” all articles in the ACM DL (through June 2020) so anyone anywhere in the world who needs ACM publications can have access to them.

newsletters being released on schedule. That includes papers accepted for ACM conferences that have not been able to meet face-to-face. I should note this “business as usual” situation is only possible because of the dedicated efforts of our volunteers, who ensure editorial boards continue to function, peer review is taking place, and editors and program committees continue to make decisions that result in important articles being delivered to you.

As an added gesture of support for the entire computing profession, ACM leadership made the decision to “unlock” all articles in the ACM Digital Library starting at the end of March to ensure anyone anywhere in the world who needs ACM publications can have ready access. This policy will continue through June 30, 2020—so it might be a good time to catch up on some of that reading you have been intending to do.

ACM's members, volunteers, authors, and readers are at the center of everything we do. Your continued support and dedication during these difficult times are inspiring, and keep reminding us how very special our ACM community really is.

Hoping that all of you will remain healthy and safe,

Cherri Pancake, ACM PRESIDENT

Cherri M. Pancake is President of ACM, professor emeritus of electrical engineering and computer science, and director of a research center at Oregon State University, Corvallis, OR, USA.

Copyright held by author/owner.

CALL FOR NOMINATIONS

AAAI Squirrel AI Award for Artificial Intelligence for the Benefit of Humanity

\$1,000,000 Prize

The AAI Squirrel AI Award for Artificial Intelligence for the Benefit of Humanity recognizes positive impacts of artificial intelligence to protect, enhance, and improve human life in meaningful ways with long-lived effects. The award will be given for the first time in 2021.

The award will be given annually at the conference for the Association for the Advancement of Artificial Intelligence (AAAI) in February, and is accompanied by a prize of \$1,000,000 plus travel expenses to the conference.

Candidates may be individuals, groups, or organizations that are directly connected with the main contribution stated in the nomination. Qualifications and technical knowledge in artificial intelligence are not requirements for nominations. The emphasis is on the significance and impact of the work.

The award is administered by AAI, with support from the European Artificial Intelligence Association (EurAI) and the Chinese Association for Artificial Intelligence (CAAI). Financial support for the award is provided by Squirrel AI.

DEADLINE for 2021 Award Nominations: May 24, 2020

<https://www.aaai.org/Awards/Squirrel-AI/>

Partnerships Can Help Drive Gender Equality

In my last editorial (Feb. 2018, p. 5), I challenged all members of the computing community to accept responsibility for the lack of gender diversity and equitable

experiences of women in our field. I also encouraged broad participation in the work necessary to realize positive change, particularly by those outside of the minority group.

Here, my focus shifts to the myriad organizations whose primary mission is to work toward an increase of women in the technology sector and/or to positively impact the experience of those women already in computing.

Each of these organizations has a unique sphere in which it works. Some seek to equip women and girls with technical skills that raise their level of interest in the field and open new educational or employment opportunities. Some work in the K–12 sector, others in higher education, and others with professional women. Some organize major events with a focus on celebrating and mentoring women while others provide local communities of support for social interaction and professional development.

All of these organizations do valuable work and I applaud those who give unselfishly of their time and talents to make that work possible. But lately I have found myself in numerous conversations that raise some important questions:

► Are we achieving all that we can or are we possibly reducing our collective impact because we are sometimes (frequently unknowingly) competing rather than cooperating for funding and volunteer effort?

► Could the large investment of time and financial resources be more

impactful if organizations identify ways in which their work intersects, and build partnership programs in those spaces?


I have seen firsthand the value of partnerships in my work within ACM-W. I will highlight two of these partnerships as prime examples of collective impact.

The Computing Research Association Committee on Widening Participation (CRA-WP) annually runs a program called Grad Cohort. This three-day workshop brings together women who are currently pursuing a master's or Ph.D. in computing with goals of encouraging persistence in the field and increasing the number of women who choose to pursue research careers. Because of funding restrictions, Grad Cohort attendees must be attending universities in the U.S. and Canada. For the past three years, ACM-W has sponsored faculty from several countries to attend the Grad Cohort workshop, with a goal of creating comparable programs internationally. CRA-WP organizers have provided valuable advice on organizing similar events. As a result, Grad Cohort workshops have been held in such locations as Greece, Ireland, India, Kuwait, Spain, and soon, Turkey.

The National Center for Women in Information Technology (NCWIT) is another U.S.-based organization that provides resources and programming for K–12, university, and professional groups. NCWIT's Aspirations in Computing (AiC) awards program honors

K–12 women, gender-queer, and non-binary students for their computing achievements and interests and encourages them to pursue their interest in technology. This past November, the ACM Canadian Celebration of Women in Computing (CAN-CWiC) ran a pilot of the AiC program. Twenty-nine nominees from across Canada attended CAN-CWiC and were recognized during the conference for their achievement. Going forward, ACM-W and NCWIT will collaborate with other non-U.S. Celebrations to launch similar programs in their regions.

These endeavors were successful because the leadership of the respective organizations came together and talked about could be leverage the mission and strengths of each individual organization could be leveraged to realize a larger impact.

It does not matter if an organization is large and globally recognized or small and working in a localized region. There is some way in which the mission and programs of each organization align with others. I challenge all who lead to be intentional in seeking out ways in which their organization can merge forces, pool resources, and work together for the greater good. Cooperation, not competition, will move the needle faster for women in computing. 

Jodi Tims (jodi.tims@northeastern.edu) is Professor of the Practice and Director of CS Programs for Northeastern University—San Francisco/Silicon Valley. She has served as chair of ACM-W since July 2017.

Copyright held by author.

Publish Your Work Open Access With ACM!

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



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



Association for
Computing Machinery



Moshe Y. Vardi

DOI:10.1145/3388890

Efficiency vs. Resilience: What COVID-19 Teaches Computing

AS I AM writing these lines, in mid-March 2020, COVID-19, the disease caused by the coronavirus virus, is spreading around the world. From a local epidemic that broke out in China in late 2019, the disease has turned into a raging pandemic the likes of which the world has not seen since the 1918 Spanish Flu Pandemic. Thousands have already died, and the ultimate death toll may be in the millions. Attempting to mitigate the pandemic, individuals are curtailing travel, entertainment, and more, as well as exercising “social distancing,” thus causing an economic slowdown. Businesses hoard cash and cut spending in order to survive a slowdown of uncertain duration. These rational actions by individuals and businesses are pushing the global economy into a recession.

Observing the economic consequences of this unexpected crisis, William A. Galston asks in a recent *Wall Street Journal*^a column: “What if the relentless pursuit of efficiency, which has dominated American business thinking for decades, has made the global economic system more vulnerable to shocks?” He goes on to argue that there is a trade-off between efficiency and resilience. “Efficiency comes through optimal adaptation to an existing environment,” he argues, “while resilience requires the capacity to adapt to disruptive changes in the environment.”

A similar point, in a different setting, was made by Adi Livnat and Christos Papadimitriou in their 2016 *Communications*^b article, “Sex as an Algorithm.”^b Computational experience has shown that Simulated Annealing, which is a lo-

cal search—via a sequence of small mutations—for an optimal solution, is, in general, superior computationally to genetic algorithms, which mimic sexual reproduction and natural selection. Why then has nature chosen sexual reproduction as the only reproduction mechanism in animals? Livnat and Papadimitriou’s answer is that sex as an algorithm offers advantages other than good performance in terms of approximating the optimum solution. In particular, sexual reproduction favors genes that work well with a greater diversity of other genes, and this makes the species more adaptable to disruptive environmental changes, that is to say, more resilient.

And yet, who have educated generations of computer scientists on the paradigm that analysis of algorithm only means analyzing their computational efficiency. As Wikipedia states: “In computer science, the analysis of algorithms is the process of finding the computational complexity of algorithms—the amount of time, storage, or other resources needed to execute them.” In other words, efficiency is the *sole* concern in the design of algorithms. (Of course, the algorithm has to meet its intended functionality). What about resilience? Quoting Galton again: “Creating resilient systems means thinking hard in advance about what could go wrong and incorporating effective countermeasures into designs.” How can we make our algorithms more resilient?

Of course, fault tolerance has been part of the canon of computing-system building for decades. Jim Gray’s 1998 Turing Award citation refers to his invention of transactions as a mechanism to provide crash resilience to databases. Leslie Lamport’s 2013 Turing Award citation refers to his work on fault tolerance in distributed systems. Neverthe-

less, I believe that computer science has yet to internalize the idea that resilience, which to me include fault tolerance, security, and more, must be pushed down to the algorithmic level. Case in point is search-result ranking. Google’s original ranking algorithm was PageRank, which works by counting the number and quality of links to a page to determine how important the website is. But PageRank is not resilient to manipulation, hence “search-engine optimization.” Today’s result-ranking algorithms are well-kept trade secrets. Indeed, adversarial machine learning, which looks at the impact of maliciously manipulated data on machine learning, is a highly active research area.

As I pointed out in an earlier column, this quest for efficiency has been single minded: “Our discipline is dedicated to reducing friction. Latency must be eliminated, bandwidth must increase, and ubiquity should be universal. Our goal is to reduce the friction of computing and communication as much as possible.” Facebook’s CEO Mark Zuckerberg speaks of “frictionless sharing” as a goal. This reduction of friction has enabled the amazing world of the Internet and the Web we have created over the past 50 years. It also provides us today with tools that enables us to work and socialize under social distancing. But we now know the imagined utopia of frictionless sharing on social media leads to filter bubbles, fake news, and extreme content.

Computing today is the “operating system” of human civilization. As computing professionals we have the awesome responsibility as the developers and maintainers of this operating system. We must recognize the trade-off between efficiency and resilience. It is time to develop the discipline of resilient algorithms. ■

a <https://on.wsj.com/3b82zcl>

b <http://bit.ly/2J0JJaQ/>

volume

01

number

01

FIRST

ISSUE

PUBLISHED

*Digital Threats:
Research and Practice*
is now available in
the ACM Digital Library



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



Association for
Computing Machinery

<https://dtrap.acm.org>

Meet the candidates who introduce their plans—and stands—for the Association.

ACM's 2020 General Election

Please take this opportunity to vote.

THE ACM CONSTITUTION provides that our Association holds a general election in the even-numbered years for the positions of President, Vice President, Secretary/Treasurer, and Members-at-Large. Biographical information and statements of the candidates appear on the following pages (candidates' names appear in random order).

In addition to the election of ACM's officers—President, Vice President, Secretary/Treasurer—five Members-at-Large will be elected to serve on ACM Council.

Electronic Balloting Procedures. Please refer to the instructions posted at <https://www.esc-vote.com/acm>.

To access the secure voting site, you will need to enter your email address (the email address associated with your ACM member record) and your unique PIN provided by Election Services Co.

Paper Ballots. Should you wish to vote by paper ballot please contact Election Services Co. to request a paper copy of the ballot and follow the postal mail ballot procedures: acmhelp@electionservicescorp.com or +1-866-720-4357.

Postal Mail Ballot Procedures. Please return your ballot in the enclosed envelope, which must be signed by you on the outside in the space provided. The signed ballot envelope may be inserted into a separate envelope for mailing if you prefer this method.

All ballots must be received by **no later than 16:00 UTC on 22 May 2020**.

The ACM Tellers Committee will validate the computerized tabulation of the ballots. Validation by the Tellers Committee will take place at 14:00 UTC on 26 May 2020.

Sincerely,

Gerald Segal

CHAIR, ACM ELECTIONS COMMITTEE

candidates for

PRESIDENT

(1 July 2020 – 30 June 2022)

**ELIZABETH CHURCHILL**

Director of User Experience
Google
Mountain View, CA
U.S.A.

Biography

Elizabeth Churchill is a Director of User Experience at Google. Her field of study is Human Computer Interaction (HCI) and User Experience (UX), with a current focus on the design of effective designer and developer tools.

Churchill has built research groups and led research in a number of well-known companies, including as Director of Human Computer Interaction at eBay Research Labs in San Jose, CA, as a Principal Research Scientist and Research Manager at Yahoo! in Santa Clara, CA, and as a Senior Scientist at the Palo Alto Research Center (PARC) and FXPAL, Fuji Xerox's Research lab in Silicon Valley.

Working across a number of research areas, she has over 100 peer reviewed top-tier journal and conference publications in theoretical and applied psychology, cognitive science, human-computer interaction, mobile and ubiquitous computing, computer-mediated communication, and social media, more than 50 patents granted or pending, and 7 academic books. Her team produces research that impacts a large number of Google's products (by shaping Google's Flutter and Material Design), influencing the work of hundreds of thousands of designers and developers globally, and thus affecting the user experience of millions of end-users. She continues to guest lecture at universities and to mentor early stage career professionals and students. In 2016, she received the Citris-Banatao Institute Athena Award for Executive Leadership.

The current Vice President of the ACM, Churchill served as ACM Secretary/Treasurer from 2016–2018, and served on the Executive Committee of the ACM's Special Interest Group on Computer-Human Interaction (SIGCHI), for 8

years, 6 years of those as Executive Vice President and two as Vice President for Chapters. She has held leadership committee positions on numerous ACM associated conferences. Churchill is a Distinguished Scientist and Distinguished Speaker of the ACM, will become an ACM Fellow in June 2020, and is a member of the SIGCHI Academy.

Churchill earned her BSc. in Experimental Psychology (1983) and her MSc. in Knowledge Based Systems (1987) from the University of Sussex in the UK, and her Ph.D. in Cognitive Science from the University of Cambridge, also in the UK (1994).

Her dissertation research focused on the design and development of Programmable User Models. After her Ph.D., she was a Postdoctoral Research Fellow at the University of Nottingham before moving to the US and into industry in 1997. She holds honorary doctorates from the University of Sussex (awarded 2018) and Stockholm University (awarded 2019) for her continued contributions to the academy in the field of HCI.

Statement

I am honored to be nominated for the position of ACM President.

As a longtime ACM member, current ACM Vice President, and an industry research leader with strong connections to academia, I believe ACM plays a leadership role in shaping not only the fields of computer science and computer engineering, but also many related disciplines. Through this connection, ACM influences industries built upon computing science and computer engineering expertise and practice.

My vision for continued ACM relevance and influence requires we focus efforts. ACM can significantly shape future research and education directions as well as future industries built on computing foundations. Toward that vision, if elected, I will be a strong voice for deepening efforts in the following areas:

- ▶ Career development: ACM provides the premier platform for career development and growth for all upcoming and established computer science and computer engineering professionals. However, we can do more. A priority area must be appealing more deeply to those in early-stage career. ACM membership currently skews toward mid-to late-stage professionals. Initiatives focused on early CS education and early professional career support will provide solid groundwork for ACM's growth and continued relevance for years to come.
- ▶ Broader promotion of our content through multi-channel platforms and diverse events: Continued investment for enabling broader access to our growing repository of high-quality, peer-reviewed content

for current and future members and beyond is critical. Making our content more accessible will ensure ACM cultivates its position as the community for all professionals associated with computing sciences.

- ▶ Enhance community networking: From our Digital Library to our many events and chapters, we provide essential community platforms for those interested in theoretical and applied computing sciences and related engineering disciplines. We can further leverage ACM's existing platforms to underscore its place as a lifelong professional network for all aspects of computer science. ACM has an unrivaled opportunity to further develop the social connectivity of its members and to create more platforms where members can share expertise with fellow members and communities beyond ACM.
- ▶ Emphasize ACM's global impact: Our membership is globally based, yet ACM is often mistaken as an 'American' association for CS academics. A focus on deepening our understanding of the needs and perspectives of our very diverse and global community will help ensure we accentuate our leadership as the premier society for computer science and engineering professionals worldwide. For all of these areas, we must address where to refresh current efforts and where to invest in new efforts.

It would be my privilege, as ACM President, to work with ACM staff, volunteers, and members to address these areas, to focus on thoughtful investments, and ensure ACM's leadership not only continues but expands and deepens.

candidates for **PRESIDENT**

(1 July 2020 – 30 June 2022)



GABRIELE KOTISIS

Full Professor in Computer Science,
Head of Department
Department of Telecooperation
Johannes Kepler University Linz
Linz, Austria

Biography

Gabriele Kotsis is Full Professor in computer science at Johannes Kepler University, Linz, Austria, and a Distinguished Member of ACM. Receiving recognition for her work from the very beginning (her master's thesis, submitted at the University of Vienna in 1991, was honored with the student sponsorship award of the Austrian Computer Society, and her Ph.D. in 1995 was honored with the highly prestigious Heinz Zemanek award) was doubtlessly a motivating factor for her and her decision to dedicate her career to research in academia and to the scientific community. In 2002, she was one of the co-founding chairs of the working group for professors in computer science within the Austrian Computer Society (OCG). From 2003 to 2007 she was President of the Austrian Computer Society, being the first female holding this position in Austria. In addition to her two-term presidency at OCG, Kotsis takes an active part in the Editorial Board of the OCG Book Series, in the working group Fem-IT (Association of Female University Professors in IT) and in the OCG award committee.

From 2007 to 2015 she served as Vice-Rector for Research at Johannes Kepler University (JKU). Her responsibilities included the

development of R&D strategies and policies within the university, coordination and interaction with national and international governmental organizations and funding bodies, and the establishment of collaborations with other research organizations and business partners. Since 2016, Kotsis has been JKU's representative in the ASEA-UNINET academic research network, which promotes cooperation among European and South-East Asian public universities. Her active involvement in this network led to her nomination and election as President for the current period, February 2019 to July 2020.

Within ACM, Kotsis has gained a reputation for organizing ACM conferences and workshops. In 2016, she received an award in appreciation of her accomplishments regarding the ACM WomEncourage conference series. Kotsis is a founding member of the ACM Europe Council, serving on the Council from 2008 to 2016. In 2014, she became an ACM Distinguished Member for her contributions to workload characterization for parallel and distributed systems, and for the founding of ACM Europe. Since 2016, she has been an elected Member-at-Large of the ACM Council.

Statement

Formal thinking and reasoning together with abstract and geometric modeling are what led me into computer science in the first place. Fascinated by the beauty and purity of binary systems in number theory, I was particularly passionate about Euclid's algorithm and prime factorization. I was captivated by the understanding of computers as machines being able to unfold the thinking condensed into algorithms. This picture has crystallized clearly over the past three decades of my active life as a computer scientist.

In our discipline, we have advanced linear (Turing-) machines to multidimensional complexity management machines, algorithmic unfolding machines to creative generative machines in artificial intelligence, and deterministic machines to true randomness in executive machine behavior in quantum computing.

These advances have opened the doors to an infinite spectrum of use domains, out of which a few are currently showing remarkable progress. Research prototypes have rapidly developed into living examples of totally autonomic machines (level 4–5 vehicles, drones, ...), very-large-scale collectives of cooperative machines (combinations of smartphones, watches, cars, homes, ...) or of self-adaptive and locally interactive machines (surgical micro-robots, personal digital agents or twins, ...).

ACM (which stands for the Association for Computing Machinery) has already reacted to significant transitions in the past by redefining and reshaping its agenda. Among the many emerging topics, for the upcoming ACM presidency I consider the following as immediately urgent:

- ▶ Computing machinery fighting the CO₂ dilemma;

- ▶ Computing machinery fertilizing medical research and health care;
- ▶ Computing machinery protecting democracy.

No other discipline or technology will have more impact on shaping our future than computer science and technology. This implies a major responsibility for our community, not only from a scientific and technical perspective in being able to provide correct solutions, but also from an ethical and societal point of view. Moreover, global problems must be addressed in a global way, independently from particular individual, national or commercial interests. My vision is that ACM, being a global organization, can and must become the platform that enables us to achieve all goals in question.

I feel honored having been nominated for the position of ACM President. ACM is a volunteer organization, and its impact depends on the help and support from all of us. We are a strong community and it will be my responsibility as President to ensure that ACM serves our needs. But we have to take a step forward. Let us work together, across the globe, not only to serve the needs of our own community, but to utilize our knowledge and expertise in "computing machinery" in order to cope with the challenges we are faced within our global society.

candidates for

VICE PRESIDENT

(1 July 2020 – 30 June 2022)

**YANNISS IOANNIDIS**

President and General Director
 "Athena" Research & Innovation Center
 Professor of Informatics & Telecom
 University of Athens
 Greece

Biography

Yanniss Ioannidis is the President and General Director of the "Athena" Research & Innovation Center in Athens, Greece (since 2011) and a Professor of Informatics & Telecom at the Univ. of Athens (since 1997). Prior to that, he was a professor of Computer Sciences at the Univ. of Wisconsin–Madison (1986–1997).

He holds a Ph.D. in Computer Sciences (Univ. of California–Berkeley, 1986), an MSc in Applied Mathematics (Harvard Univ., 1983), and a Diploma in Electrical Engineering (National Technical Univ. of Athens, 1982).

His research interests include database and information systems, data science, data and text analytics, recommender systems and personalization, and electronic infrastructures. His work is often motivated by data management problems that arise in the context of other scientific fields (Life Sciences, Cultural Heritage and the Arts, Physical Sciences). He has published over 160 articles in leading journals and conferences and holds three patents.

Ioannidis is an ACM and IEEE Fellow (essentially both "for contributions to database systems, particularly query optimization"), a member of Academia Europaea, and a recipient of several research and teaching awards, including

Presidential Young Investigator Award (1993), UW Chancellor's Teaching Award (1996), VLDB 10-Year Best Paper (2003).

An ACM member since 1983, he is the current Secretary/Treasurer of ACM and also serves on the ACM Europe Council. Previously he was a member of the SIG Governing Board Executive Committee (6 years) and the ACM Publications Board (4 years) and served 4-year terms as vice-chair and then chair of the Special Interest Group on Management of Data (SIGMOD). In 2017 he received the ACM SIGMOD Contributions Award.

Ioannidis is a vice chair of the European Strategy Forum on Research Infrastructures (ESFRI), the Greek delegate to ESFRI and a member of its Executive Board. He is also a member of the steering committee of the IEEE Int'l Conf. on Data Engineering, while in the past he has also served on the IEEE Technical Committee on Data Engineering and the VLDB Endowment Board of Trustees.

Statement

The "tetrahedron" formed by mutually interlinking *research, education, industrial innovation* (the three nodes of the well-known "triangle of knowledge"), and *policy* is a great framework for conceptualizing the essential activities of a disciplinary professional society. If honored to be elected as ACM Vice-President, I will use my experience from earlier volunteer positions, especially that of ACM Secretary/Treasurer, to help ACM further strengthen its current leadership role within our thriving community, expand its already extensive services across the four areas/nodes of the tetrahedron, and attract, embrace, and benefit all computing professionals globally. Below are some of the directions I care deeply about.

Expand footprint and become the home of all interdisciplinary areas that involve computing: Our field is now on the critical path of most scientific and societal activities, coming itself to an exciting turning point. While remaining current on the purely technological advances, ACM should form strategic alliances with peer scientific societies and have joint activities, expanding and enriching its membership with non-traditional backgrounds.

Include computing as a basic curriculum strand right from the beginning of schooling: Algorithmic thinking is a fundamental skill and ACM should continue its efforts

and intensify its involvement in shaping all levels of formal and alternative education. ACM can inspire many young people to follow a career in computing and, thus, lead to a community that is balanced on gender, geography, and age.

Identify major technology-dependent challenges connected with the UN SDGs: ACM should capture the pulse of the computing industry and establish links for better integration of industry-relevant activities. In the spirit of its co-leadership role in events such as the "AI for Good" Global Summit, ACM should coordinate with industrial innovators, establish competitions towards solving global challenges, and support winning teams of inspired community members.

Prioritize social responsibility: Today our creations "run the world" and with this comes great responsibility. ACM should promote its new Code of Ethics widely within the community and also engage with and advise policy makers on cutting-edge technologies, including on their invention without restrictions and their application within clear ethical boundaries.

candidates for

VICE PRESIDENT

(1 July 2020 – 30 June 2022)

**JOAN FEIGENBAUM**

Grace Murray Hopper Professor of Computer Science
 Professor of Economics (by courtesy)
 Yale University
 New Haven, CT, U.S.A.
 Amazon Scholar, Amazon Web Services, Inc.
 Seattle, WA,
 U.S.A.

Biography

Joan Feigenbaum is the Grace Murray Hopper Professor of Computer Science at Yale, where she also holds a courtesy appointment as Professor of Economics. She joined Yale in 2000 and served as Computer Science Department Chair from July 2014 through June 2017. Before Yale, she was at AT&T for 14 years (AT&T Bell Labs from July 1986 to December 1995 and AT&T Labs – Research from January 1996 to June 2000); there, she participated broadly in the company’s Information-Sciences agenda, e.g., by creating a research group in Algorithms and Distributed Data. Feigenbaum received her AB in Mathematics from Harvard in 1981 and her Ph.D. in Computer Science, under the direction of Andrew Chi-Chih Yao, from Stanford in 1986.

A member of ACM since grad school, Feigenbaum has served in many roles, including SIGACT Executive Committee member (2005–09) and SIGecom Vice Chair (2005–11); in SIGecom, she played a leading role in establishing the *ACM Transactions on Economics and Computation (TEAC)*. She has served as PC Chair or Co-Chair for three ACM conferences, PC member for 16 ACM conferences, and

editorial-board member for TEAC. Currently, she is on the Gödel-Prize committee and previously served on the SIGecom Test-of-Time award committee, the ACM Fellows-selection committee, and the Knuth-Prize committee. Most recently, she led the creation of the ACM Symposium on Computer Science and Law and served as General Chair for the inaugural symposium in 2019.

Feigenbaum’s research interests are in security, privacy, and anonymity; Internet algorithmics; and computational complexity. Well known for her ability to establish and explicate research priorities, she has done direction-setting work in computational accountability, authorization and trust management, distributed algorithmic mechanism design, and massive-dataset algorithmics. She is an ACM Fellow, an American Association for the Advancement of Science Fellow, a member of the Connecticut Academy of Science and Engineering, and a Connecticut Technology Council Woman of Innovation.

Statement

It is an honor to be nominated for the position of ACM Vice President at this tumultuous time in computing history. Since ACM was founded in 1947, computers have become indispensable tools in many aspects of daily life. Recently, a more profound change has begun: *Sophisticated computation* is becoming an essential component of many spheres of human activity. People who can understand and exploit computational methods and principles, rather than simply use computers as appliances, now have a decisive advantage over their less computationally astute competitors.

ACM members can address myriad threats now facing society. These threats combine sophisticated computation in critical ways with politics (as in “election hacking”), economics (as in technology-induced unemployment), journalism (as in “fake news”), law (as in mass surveillance in the name of national security), international relations (as in “cyber war”), finance (as in bitcoin speculation), and many other fields. In tackling them, computer scientists will work collaboratively with people in social sciences, law, and many disciplines besides the STEM fields with which we have collaborated for decades. My experiences as a leader of ACM’s efforts in Economics and Computation and, more recently, as a founder of its efforts in Computer Science and Law have given me the skills and

perspective needed to support ACM members in wide-ranging, interdisciplinary work.

As the premier organization of computing professionals, ACM must convince young computer scientists to join and participate. Our conference proceedings and journals are highly valued, but, with ArXiv preprints readily available and access to the Digital Library through employers, many people feel no need to join. Similarly, ACM-sponsored awards are prestigious, but non-members are eligible for many of them. If elected, I will conduct a series of structured discussions with non-members to learn their views on the proper role of a professional society.

On an optimistic note, ACM members are increasingly interested in real solutions to serious problems, such as carbon-intensive conference participation and limited access to published research. If elected, I will work to accelerate our transition to sustainable practices such as online conferences, remote participation in face-to-face conferences, and fully funded open access publication.

candidates for

SECRETARY/TREASURER

(1 July 2020 – 30 June 2022)

**JEFF JORTNER**

Principal Member of Technical Staff, Solutions
Architect
Sandia National Laboratories
Livermore, CA
U.S.A.

Biography

Jeff Jortner holds a Ph.D. in Mechanical Engineering (minor in Computer Science, Louisiana State University 1986), a MS in Mechanical Engineering (LSU, 1982), and a BS in Mechanical Engineering (LSU, 1977). His dissertation research involved the development of a new curve algorithm for Computer-Aided Design (CAD).

As a staff member at Sandia National Laboratories, Jortner has over 33 years of experience in leading, developing, evaluating, and applying novel tools for Scientific Visualization, Geospatial Analysis, Visual Analytics, and Computer-Aided Modeling.

His current interests involve multi-site collaborative VR for design and secure videoconferencing technologies. He is working on strategy development for a Unified Communication environment across the Department of Energy.

Jeff has experience in leading projects for communication

technologies in public alert and warning systems (in the aftermath of Hurricane Katrina) and image analysis and assessment of next-generation transportation security systems.

An ACM member since 1987, Jortner served on the Special Interest Group on Computer Graphics and Interactive Techniques (ACM SIGGRAPH) Executive Committee for eleven years, six of those years as President and five years as Treasurer. He has held other committee positions for a number of SIGGRAPH conferences and the

ACM SIGGRAPH organization (Registration, Networking, Panels, Information Director, Chapter Leader), and represented ACM SIGGRAPH as Treasurer at the DUX2005 Conference in collaboration with AIGA and SIGCHI.

Jortner is currently the Chair of the ACM Special Interest Group Governing Board (SGB).

Statement

I am honored to be nominated for ACM Secretary/Treasurer.

As the ACM SGB Chair, I am proud to have participated in the steps that ACM has taken as a leader in ethics, inclusivity, Open Access and in providing independent technical policy advice to the public. These efforts should be enhanced and continued.

The computing community is expanding to include a diverse set of backgrounds, locations, and skillsets. ACM is positioned through publications, conferences, chapters, and councils to be a conduit for this diverse community. Digital technologies provide ways to increase connectivity through remote engagement, online media and social networks. The technologies encourage greater participation which can only enrich our field. I look forward to helping expand digital opportunities provided by ACM.

The Digital Library (DL) is a valuable funding source for ACM that supports activities that include

the Special Interest Groups (SIGs) and Chapters. ACM is actively deploying and innovating with DL Open Access publication models. A revamped DL is in development that enhances the personal experience for research and education. I believe that we need to continue such developments, increase practitioner content, and add more reproducibility artifacts to the DL.

The SIGs interact with a significant cross-section of our community through conferences and other activities that provide a significant income stream to ACM. Continued engagement with the SIGs is crucial in the areas of volunteer recruitment, retention and development. Of equal importance is membership retention and the diversity of volunteers, speakers and attendees.

It would be my privilege to serve as Secretary/Treasurer and continue working with ACM staff and volunteers in providing services to our community.

candidates for

SECRETARY/TREASURER

(1 July 2020 – 30 June 2022)

**ELISA BERTINO**

Samuel Conte Professor of Computer Science
Computer Science Department, Purdue University
West Lafayette, IN
U.S.A.

Biography

Elisa Bertino is a professor of Computer Science at Purdue University, where she leads multidisciplinary research in IoT security, data security and privacy, 4G and 5G cellular networks and mobile systems security, analytics for security, and digital identity management. She has made pioneering contributions for over 30 years to data management and data security theory and systems and has worked to broaden participation in computing via professional leadership and mentoring. Her work in data security and privacy include context-based access control, privacy-preserving analytics, and data protection from insider threats. She led the development of Purdue Computational Research Infrastructure for Science (CRIS), released as open source software in 2016.

Previously, she was a professor in and head of the Department of Computer Science at the University of Milan, a postdoc at the IBM Research Laboratory (now Almaden), and a visiting professor at Singapore Management University and Singapore National University.

She served as editor-in-chief of *IEEE Transactions on Dependable and Secure Computing*, and coordinating co-editor-in-chief of the Very Large Database Systems (VLDB) Journal. She chaired ACM's Special Interest Group on Security, Audit and Control (SIGSAC) from 2009–2013. In 2011, she co-founded ACM's Conference on Data and Application Security and Privacy, now considered the main forum for high-quality research on data privacy and security.

Bertino is a Fellow of ACM, IEEE, and AAAS. She received the 2019–2020 ACM Athena Lecturer Award and was named to GSMA's Mobile Security Research Hall of Fame for her work on 4G and 5G cellular network security. She received the 2014 ACM SIGSAC Outstanding Contributions Award for her seminal research and outstanding leadership in data security and privacy over 25 years; the 2002 IEEE CS Technical Achievement Award for her contributions to database systems and security and advanced data management systems; and the 2005 IEEE CS Tsutomu Kanai Award for pioneering and innovative research contributions to secure distributed systems.

Statement

I have been a member of ACM for 38 years. I am honored to have been nominated as a candidate for Secretary/Treasurer of ACM.

I strongly believe that the field of computer science is today more exciting than ever.

We see fundamental advances, such as those made possible by AI, IoT systems, quantum computing, and 5G technologies, and unprecedented opportunities for novel applications. Our technologies have a fundamental role in shaping society. However key questions need to be addressed including AI and data ethics, data transparency, personal privacy versus collective security, and sustainability. Answers to those questions as well as others posed by the pervasive use of our technologies must be given by taking into account a broad multidisciplinary perspective. If elected, I will work together with the ACM Executive Committee and the many volunteers and leaders in ACM to make sure that ACM has a central

role in fostering discussions and initiatives to answer those questions as well others posed by society concerning our technologies. I will also focus on important matters, such as broadening diversity in our field, supporting younger researchers, open access to data and publications, the role of conferences vs. journals, industry engagement, large-scale research infrastructures, and last but not least making sure that ACM stays technically relevant by organizing workshops and conferences on new emerging technologies and applications.

The ACM Secretary/Treasurer also oversees ACM's finances. If elected, I will leverage my past experience as a volunteer in different roles to help ACM maintain and enhance its current financial stability, while at the same time ensuring that ACM funds are used to best serve our research community.

candidates for

MEMBERS AT LARGE

(1 July 2020 – 30 June 2024)

**ALFRED Z. SPECTOR**Chief Technology Officer, Two Sigma
New York, NY
U.S.A.

Biography

Alfred Spector is the Chief Technology Officer at Two Sigma, a firm dedicated to algorithmic approaches to a wide collection of financial optimization problems. His career has led him from innovation in large-scale, networked computing systems to broad engineering and research leadership.

Prior to Two Sigma, Spector spent nearly eight years as VP of Research at Google. Before Google, Spector held various senior-level positions at IBM, including as global VP of Services and Software Research. He previously founded Transarc Corporation, a pioneer in distributed transaction processing and wide-area file systems, and he was a tenured professor at Carnegie Mellon University. Spector was an undergraduate at Harvard and obtained a Ph.D. in computer science from Stanford.

Spector was a Hertz Fellow at Stanford and is also a Fellow of both the ACM and the IEEE. He is an active member of the National Academy of Engineering and the American Academy of Arts and Sciences, where he serves on the Council. Spector won the 2001 IEEE Kanai Award for Distributed Computing and the 2016 ACM Software Systems Award. In 2018–19, Spector lectured widely as a Phi Beta Kappa Scholar and has been a member of the ACM Turing Award Committee. As to government service, Spector was a member of the Army Science Board, and he chaired the NSF's CISE Advisory Board. He has had extensive international experience due to broad responsibilities at IBM, Google, and Two Sigma.

Recently, Spector has lectured widely on the growing importance of computer science across all disciplines based on the evocative phrase, CS+X. More recently, he has written and lectured on the societal implications of data science—both the great benefits and the unintended consequences.

Statement

Computers have become 1+ trillion times more capable since the introduction of transistors, and computer science's innovations in the analytical/algorithmic and engineering domains have been equally remarkable. Our field has also added a strong empirical dimension, both benefiting from and facilitating the vast growth of computing worldwide. Unsurprisingly, this innovation is impacting us all: as information technology professionals and as citizens, and it is affecting the ACM.

With their great reach and capability, our field has delivered amazing and sometimes insufficiently understood benefits. But society is rightly concerned with the downsides of technology, both actual, potential, and sometimes even fictional. Those concerns, if not properly addressed, run many risks, including to ourselves and the opportunities we collectively have to improve our world. Without our attention, idealistic youth might even stop flocking to our field.

If I am elected an ACM Member at Large, I will use my broad perspective to provide both strategic and operational counsel to ACM. For example, I will seek to improve the relevance of ACM to our diversifying field; to push for open access publication while ensuring a sound business model; to catalyze the publication of novel software artifacts as an intellectual pursuit; to maintain our proudly international organization in an era of increased nationalism; to accelerate the diversification of computing to all who can contribute; to thoughtfully lead discussions of policy and ethics that benefit society, our field, and our membership; and to balance the diversification of our field (e.g., the rise of data science as a quasi-parallel discipline) while promoting the core of our field. I hope to contribute to these topics and many more.

**JOHN C.S. LUI**Choh-Ming Li Chair Professor, CSE Department
The Chinese University of Hong Kong (CUHK)
Hong Kong, China

Biography

John C.S. Lui is the Choh-Ming Li Chair Professor in the CSE Dept. at The Chinese University of Hong Kong (CUHK). He received his Ph.D. in Computer Science from UCLA. He is a Fellow of ACM, Fellow of IEEE, Senior Research Fellow of the Croucher Foundation, Fellow of the HK Academy of Engineering Sciences, and was the past chair of ACM SIGMETRICS (2011–2015). After his graduation, he joined IBM Laboratory and participated in research on file systems and parallel I/O architectures. He later joined CUHK. He has been a visiting professor at UCLA, Columbia Univ., Univ. of Maryland at College Park, Purdue Univ., Univ. of Massachusetts at Amherst and Universit degli Studi di Torino in Italy. His research interests are in machine learning algorithms, mathematical analysis and design of large-scale networking/computing systems. Lui is currently the senior editor of the *IEEE/ACM Transactions on Networking*, and has been serving on the editorial board of *ACM Transactions on Modeling and Performance Evaluation of Computing Systems*, *IEEE Transactions on Network Science & Engineering*, *IEEE Transactions on Mobile Computing*, *IEEE Transactions on Computers*, *IEEE Transactions on Parallel and Distributed Systems*, and *Journal of Performance Evaluation*. He is a member of the review panel of the IEEE Koji Kobayashi Computers and Communications Award committee and has served on the IEEE Fellow Review Committee. Lui served as the chairman of the CSE Department (2005–2011), the Associate Dean of Research in the College of Engineering at CUHK (2014–2018). He is an active consultant and advisor in many high tech. companies. Lui also received various departmental teaching awards, the CUHK Vice-Chancellor's Exemplary Teaching Award and the CUHK Faculty of Engineering Research Excellence Award (2011–2012).

Statement

It is a privilege to be nominated for Member-at-Large. I've been an ACM member since I was a Ph.D. student, and have been constantly amazed by the various educational opportunities and scholarly events that ACM offers to our community, and I am equally impressed by the many dedicated ACM members for their passion of pushing forward computing technologies in all aspects of human activities.

If elected, I would like to help expand some key activities along these lines.

- ▶ Promote volunteering services from Asian countries (e.g., China, India, Japan and Singapore, etc.) and European countries (e.g., England, France, Germany, Italy, etc.).
- ▶ Promote closer collaboration between IT companies and academic researchers around the world.
- ▶ Promote more volunteering participation in organizing educational and academic events.

Information technology is crucial to our economic growth, and ACM has the responsibility of creating more public awareness of IT technologies, and how these technologies may transform our jobs and working environment. ACM also needs to have a firm commitment to relay the advantages and pitfalls of some of the latest computing technologies to different government agencies around the world. My academic training, industrial experience, and relationship with government agencies gives me the unique viewpoint of how to promote the above items. But, to be successful, I also need all ACM members to join hands and help out so that our community can make a bigger impact.

In summary, we now have the wonderful opportunity to shape a better and brighter future for our next generation. Let's join hands to create a better place for our future generations.

candidates for

MEMBERS AT LARGE

(1 July 2020 – 30 June 2024)

**TOM CRICK**

Professor of Digital Education & Policy
Computational Foundry and School of Education
Swansea University
Swansea, U.K.

Biography

While Crick's disciplinary background is in computer science and informatics, his academic interests are naturally interdisciplinary and sit at the research/policy interface, solving data-driven and computationally intensive problems across a range of domains: data science, intelligent systems, cyber security, smart cities, software sustainability and reproducibility, as well as STEM education, science/innovation policy, digital public services, and skills/infrastructure for the digital economy. His research and policy work has been funded by the UK Research Councils (EPSRC, ESRC, Innovate UK), the European Commission and the Welsh Government. He was previously the NESTA Data Science Fellow (2013–2015), developing approaches to embedding data science capability into government for more effective data-driven policy-making; a Fellow of the UK Software Sustainability Institute (2014); an HEA National Teaching Fellow (2014) for his work in computer science education; and a Science Media Fellow (2011) with the BBC. In 2017, he was appointed MBE in the Queen's Birthday Honours for "services to computer science and the promotion of computer science education."

Crick has significant experience of non-executive governance, advisory roles and influencing at senior levels in government and industry. He has chaired national curriculum reviews in the UK over recent years, especially reforming computer science, digital skills and STEM education in Wales. He is an inaugural Commissioner of the National Infrastructure Commission for Wales (2018–present), as well as a Vice-President of BCS, The Chartered Institute for IT (2017–2020). He is Vice-Chair (2019–present) of the

ACM Europe Council, having been elected in 2017, and a member of the ACM Europe Technology Policy Committee.

Statement

I am honored to be nominated as a potential Member-at-Large of ACM Council. While I have supported a variety of ACM conferences, activities, and initiatives over recent years, especially through the ACM Europe Council, I am enthusiastic about serving ACM more widely to further support and develop a diverse and impactful international computing community. It is clear we face a number of challenges—and opportunities—as a discipline and community over the coming years. There are broad social, cultural and economic imperatives; for example: the widespread impact of technology, data and computational processes on our lives; digital innovation, automation and the future of work; shifting legal, ethical and professional responsibilities; national and international collaborative research agenda (funding, industrial strategies, mobility, open access/data/research); dramatic changes to our education systems: curriculum reform, qualifications, accreditation and certification, and a range of challenges for universities; and supporting the careers and professional development of a diverse global computing/IT profession.

In a rapidly shifting political and policy landscape, much is possible—but this requires more explicit international collaboration with national academies and professional bodies, as well as with industry and the general public. Building on my previous experience and networks as an academic, policy advisor and extensive industry non-executive roles (including multibillion pound utilities and public services), I would relish the opportunity to serve as a Member-at-Large on ACM Council.

(For more information about my research/policy work, as well as my aspirations for the ACM Council Member-at-Large role, see: <https://proftomcrick.com> and @ProfTomCrick).

**SUSAN DUMAIS**

Technical Fellow and Director
Microsoft Research Labs in New England, New York
City and Montréal
U.S.A. and Canada

Biography

Susan Dumais is a Technical Fellow and Director of the Microsoft Research Labs in New England, New York City and Montréal, and an adjunct professor at the University of Washington. Prior to joining Microsoft, she was a Member of Technical Staff at Bell Labs and Bellcore.

Her research spans information retrieval, human-computer interaction and data science with a focus on algorithms and interfaces that help people more easily find and derive insights from relevant information. She is a co-inventor of Latent Semantic Analysis, a well-known dimension-reduction technique for concept-based retrieval. She has conducted research and developed systems for email spam filtering, desktop and Web search, context-aware information systems, understanding behavioral interactions, and temporal dynamics of information. She has also worked closely with Microsoft teams on search-related innovations for the desktop, enterprise and web. Her interdisciplinary research has been widely cited (~75000 citations), and she holds more than 50 patents.

Dumais is Past-Chair of ACM SIGIR (1999–2003), served as an editor for ACM ToIS (1999–2011) and ToCHI (2001–2011), was technical co-chair of ACM CHI (1994) and ACM SIGIR (2006), served on several ACM committees including Fellows, Athena Lecturer and Nominations, and has been a Member-at-Large on the ACM Council since 2016. She was elected to the CHI Academy (2005), ACM Fellow (2006), National Academy of Engineering (2011), American Academy of Arts and Sciences (2015), and received the ACM SIGIR Gerard Salton Award for Lifetime Achievement (2009), ACM SIGCHI Lifetime Research Award (2020), Tony Kent Strix Award (2014), ACM Athena Lecturer Award (2014), and Lifetime Achievement Award Indiana Univ Psychological and Brain Science (2017).

Statement

I have been an active member of ACM for my entire professional career and would appreciate the opportunity to serve as a Member-at-Large on the ACM Council. As a current Council member, I have worked on the Future of Computing Academy and new technology directions. Although ACM is widely recognized as the premier professional computer society, with strong commitment from volunteers, there are challenges ahead.

ACM has a long history of advancing computing through conferences, publications and resources for education and professional development. The digital library will have to continue to evolve to address the need for open access and the inclusion of rich media, open data, and other resources for students, practitioners and researchers. There are also new opportunities to augment conferences to better support remote participation to mitigate the need for travel and broaden participation.

Many of the challenges that the computer science community faces today will not be solved by a single discipline in isolation and would benefit from interdisciplinary perspectives. I would like to see us develop opportunities for people with different backgrounds to come together to address important technical and societal problems.

I will also work to continue to broaden participation from individuals with diverse backgrounds and to support their career development. Finally, I would like to see ACM take a more proactive role in informing policy around important issues such as personal privacy, and network and data security.

I believe that my sustained service to ACM and multidisciplinary background provide me with perspectives that would be an asset to ACM in addressing computing challenges moving forward. If elected, I would be honored to serve as a Member-at-Large on the ACM Council.

candidates for

MEMBERS AT LARGE

(1 July 2020 – 30 June 2024)

**SANJIVA PRASAD**

Professor and Head of Dept. of Computer Science & Engineering
Indian Institute of Technology Delhi
India

Biography

Sanjiva Prasad is a Professor and current Head of the Department of Computer Science and Engineering at the Indian Institute of Technology Delhi, where he has worked for 25 years. He headed the Khosla School of IT (2011–2015). Earlier, he worked at ORA Corp., Ithaca, and ECRC GmbH, Muenchen, and was a visiting Lektor at Aarhus Universitet.

Prasad earned a B. Tech. in Computer Science at IIT Kanpur (1985) and received a Ph.D. from Stony Brook University (1991). His research interests lie in formal methods and verification. In particular, he is interested in languages, types and logics for concurrent and distributed systems, and on problems such as:

- ▶ formal foundations of network routing;
- ▶ mobility and security protocols;
- ▶ analysis frameworks for secure information flow;
- ▶ operational semantics of modern architectures.

He has worked on applications of computing in fields such as:

- ▶ neurosurgery and mHealth;
- ▶ systems biology;
- ▶ eco-design;
- ▶ ICT for underserved communities.

Since 2019, Prasad is Editor-in-Chief of ACM Books. He chairs the executive committee of Association for Logic in India. He has co-chaired the Program Committee of FSTTCS twice, ICLA and ICDCIT, and served on the program committees of many conferences, e.g., POPL, LPAR, ICTAC, SEFM, ATVA, FoSSACS, APLAS, FORTE, etc. He has delivered invited lectures at many conferences and workshops and talks at leading universities across the world.

Prasad serves on several apex committees overseeing research and doctoral programs funded by the ministries of IT and Health of the Indian Government. He has designed curricula in CS, IT, and engineering for 6 universities (India, Kuwait). He has advised leading Indian hospitals (AIIMS, NEIGRIHMS), Indian Railways, and several cultural organizations.

Statement

I am honored to be nominated as a candidate for the Member-at-Large office. I joined ACM as a grad student over three decades ago, and treasure the support I received from SIGPLAN for attending a major conference.

As ACM has grown internationally, especially in Asia, it must continue to reach out: making its flagship events, activities, and publications accessible to the growing diversity (in country, gender, age, profession) of its membership. I will work toward extending ACM's culture of community, collaboration, mutual respect, inclusiveness, and spirit of volunteering to researchers and professionals across the world.

As Editor-in-Chief of ACM Books, I consider it important that young researchers and working professionals be able to access the best pedagogical and research material in a timely and affordable manner. I believe that ACM must transit soon to an open access model for its research publications that is equitable for researchers across the world. It should also provide practical technical material for working professionals in its Digital Library. I believe we are inventive enough to manage this transition without impacting ACM's other major activities.

With computing finding its way into every aspect of our lives, "advancing computing as a science and profession" means:

- ▶ imbuing in every individual (across nationality, age, gender, etc.) a computational way of thinking;
- ▶ educating society about opportunities and threats that new technologies pose;
- ▶ disseminating the ethics and ethos needed for creating responsible computing professionals.

I would like to see ACM take a proactive role in influencing international policies on important societal issues such as fairness, personal privacy, software and system safety, and network and data security.

**MEHRAN SAHAMI**

Professor (Teaching) and Associate Chair for Education
Computer Science Department
Stanford University
Stanford, CA
U.S.A.

Biography

Mehran Sahami is Professor (Teaching) and Associate Chair for Education in the Computer Science department at Stanford University. He is also the Robert and Ruth Halperin University Fellow in Undergraduate Education at Stanford. Prior to joining the Stanford faculty in 2007, he was a Senior Research Scientist at Google (2002–2007) and a Senior Engineering Manager at Epiphany (1998–2002). He is an ACM Distinguished member.

Sahami is currently Past Chair of the ACM Education Board, having completed two 2-year terms as Co-Chair, helping to initiate and oversee educational activities for ACM on a broad scale. He Co-Chaired the ACM/IEEE-CS joint task force on Computer Science Curricula 2013 (CS2013), which was responsible for creating international curricular guidelines for college programs in CS and, in 2014, received the ACM Presidential Award for his leadership of this effort. He also co-founded and served as the first General Chair of the ACM Conference on Learning at Scale, an annual meeting focused research at the intersection of learning science and computer science and was co-founder and first Chair for the annual Symposium on Educational Advances in Artificial Intelligence (EAAI).

Sahami's research interests include computer science education, computer ethics, and machine learning. He has published numerous technical papers, including the book "Text Mining: Classification, Clustering and Applications," received various awards and recognitions for his work, and has over 20 patent filings. He is currently working on a new book on ethics and technology. He received his B.S., M.S., and Ph.D. in Computer Science from Stanford.

Statement

I am honored to be nominated to run for a Member-at-Large. As Co-chair of the ACM Education Board, I've served on the Extended Executive Committee of the ACM for four years. That experience gives me a deep appreciation for the issues facing ACM and provides the opportunity to be effective from day one as a Member-at-Large.

My main goals are working to better serve the needs of the membership, specifically pursuing opportunities to push for more open access models for publication, increasing development of content relevant to practitioners, and more fully realizing ACM's mission to be a truly global association. Open access in publications is an area where ACM needs to take a stronger stance, both internally by charting a part for opening up the digital library, as well as externally by being an advocate for more open access to scholarly research in general. This is the direction the field needs to go and ACM should be leading.

Additionally, I am deeply concerned about ethical issues that have become commonplace in computing and believe that ACM needs to take a more active role in educational and policy activities to address these issues. Ethics in technology is an active area of work for me, including teaching a multidisciplinary course, for which we produced professionally written case studies on several topics (algorithmic decision-making, autonomous systems, data privacy, the power of large computing platforms/companies) and made them all freely available. While updating the ACM Code of Ethics was a good step, I believe that ACM needs to be more active in engaging with the policy conversations around how technology should be regulated for the benefit of all.

I look forward to continuing to serve ACM and I appreciate your consideration for a Member-at-Large seat.

candidates for

MEMBERS AT LARGE

(1 July 2020 – 30 June 2024)

**ALEJANDRO SAUCEDO**

Engineering Director (Machine Learning),
Seldon Technologies
Chief Scientist, The Institute for
Ethical AI & Machine Learning
London, U.K.

Biography

Saucedo actively contributes to the ACM as member of the European Technology Policy Committee through his work on Explainable AI systems and responsible machine learning development, which advocates ACM's Code of Ethics and Professional Conduct. He is the Chief Scientist at the Institute for Ethical AI & Machine Learning, where he leads the development of industry standards on machine learning bias, adversarial attacks and differential privacy. Saucedo is also the Director of Machine Learning Engineering at Seldon Technologies, where he leads large-scale projects extending and implementing open source and enterprise infrastructure for production machine learning systems that manage thousands of models. With over 10 years of software development experience, Saucedo has held technical leadership positions across hyper-growth scale-ups and has delivered multinational projects with top-tier investment banks, magic circle law firms, and global insurance companies. He has a strong track record building cross-functional departments of software engineers from scratch and leading the delivery of large-scale machine learning systems across the financial, insurance, legal, transport, manufacturing, and construction sectors (in Europe, U.S., and Latin America).

LinkedIn: <https://linkedin.com/in/axsauceado>

Twitter: <https://twitter.com/axsauceado>

GitHub: <https://github.com/axsauceado>

Website: <https://ethical.institute/>

Statement

ACM continues to represent the core values that encompass my passion for our profession, and as an organization it has managed to evolve through the decades to continuously drive our field forward while staying true to its grass-root values. I am honored to be nominated as a Member-at-Large, as that offers me the opportunity to give back directly to the great ACM community and contribute to the many initiatives that make this member-driven organization so great.

As a member of the ACM community I have been active through several practitioner, academic, and policy initiatives. My volunteering work has consisted mostly around contributions through workstreams that focus on social responsibility and emerging technologies. I am active across various communities advocating the ACM values through global open source and technology conferences and forums, as well as through my advisory roles at the Linux Foundation, the European Commission, the Royal Society, and the Institute for Ethical AI. I am continuously looking to explore ways of expanding the ACM's reach through internal and external initiatives. As an ACM Member at Large, my main objective will be to focus on the core internal member-driven initiatives that have made the ACM an organization whose members are proud to continue contributing and representing. The social, professional, and ethical responsibilities of practitioner and academic members are a key area that I am keen to continue advocating through the great resources ACM members have created. As ACM Member at Large, I will be committed to represent the ACM community and drive forward the initiatives that will help strengthen this great members-driven organization.

**NANCY M. AMATO**

Abel Bliss Professor and Department Head
of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL
U.S.A.

Biography

Nancy M. Amato is Abel Bliss Professor and Department Head of Computer Science at the University of Illinois at Urbana-Champaign. Before joining Illinois in January 2019, she was Regents Professor and Unocal Professor of Computer Science and Engineering at Texas A&M University, where she had been on the faculty since 1995. She received M.S. and Ph.D. degrees in computer science from UC Berkeley (1988) and the University of Illinois (1995), respectively, and bachelor's degrees in mathematical sciences and economics from Stanford (1986).

Her research focuses on motion planning and robotics, computational biology and geometry, and parallel computing and she has been working to broaden participation in computing for more than two decades. She has graduated 23 Ph.D. students, including 11 from underrepresented groups. She is VP for Member Activities for the IEEE Robotics and Automation Society (RAS), served as program chair for the 2015 IEEE International Conference on Robotics and Automation (ICRA) and for Robotics: Science and Systems (RSS) in 2016. She is an elected member of the Computing Research Association (CRA) Board of Directors (2014–2020), is Vice Chair of the CRA Executive Committee (2019–2020) and was co-Chair of CRA-WP (2014–2017) and of the NCWIT Academic Alliance (2009–2011).

Amato received the 2019 IEEE RAS Saridis Leadership Award in Robotics and Automation, the 2014 CRA Habermann Award, the inaugural NCWIT Harrold/Notkin Research and Graduate Mentoring Award in 2014, the 2013 IEEE Hewlett-Packard/Harriet B. Rigas Award, and Texas A&M University-level awards in teaching (2011) and research (2018). She is a Fellow of the AAAI, AAAS, ACM, and IEEE.

Statement

I have been an active member of the ACM since my days as a graduate student (30+ years!)—I am honored by this nomination and would welcome the opportunity to contribute to the ACM.

This is an extremely exciting time for our field. Advances in computing have led to breakthroughs that are changing how we solve problems in all disciplines and to new technologies and products that have impacted all sectors of the society and all aspects of daily life, providing unprecedented opportunities for computing researchers and generating tremendous interest in our field. This is in turn driving demand for and innovation in computing education.

These developments provide many opportunities for the ACM to take leadership and have impact. As the field expands, we need to ensure the ACM portfolio covers all relevant research areas, including multidisciplinary areas in which computing plays a central role (e.g., robotics), and to ensure the most important and impactful research results are published in our conferences and journals and are available to all. Additionally, there are many societal issues that ACM has an opportunity, and in many cases an obligation, to play a leading role in shaping the conversation. This includes being viewed as a trusted and honest source of information about issues related to computing, serving as an advocate for lifelong computing education, and ensuring that the entire population feels welcome to, and indeed does, engage in all aspects of our profession.

Given the myriad opportunities, ACM, and in particular its society-level bodies such as the ACM Council, needs to strategically determine where, when, and how to engage.

If elected, I would be honored to serve as a Member-at-Large of the ACM Council.



CAREER PATHS IN COMPUTING

DOI:10.1145/3391913

Launching a New Feature in *Communications*

COMPUTING IS AN extraordinarily important transformative technology that has not only grown into a major profession accounting 2.8 million jobs in the U.S. and perhaps four times that number worldwide. Our knowledge of computing technology empowers us, and enables invigorating, varied, and surprising career paths! (see “Computing Is the Secret Ingredient”^a).

To celebrate computing’s growing importance for society and commerce, increase our understanding of the broad reach of computing, and highlight a diversity of career paths and role models, *Communications* is launching a new series of one-page articles that will appear prominently in the front of the magazine, under the following charter:

To establish a high-quality, compelling feature that broadens *Communications’* presentation of the computing profession with a particular focus on a young professional audience and a breadth of career paths. Specifically, to highlight:

► remarkable computing professionals in industry (technology and beyond), government, NGO’s, and more. The collection should frame a broad view of contribution, impact, intellectual challenge, and reward—far beyond the academic and research model,

► the global ACM membership, showing career paths in all regions, including Europe, Asia, Oceania, Americas, Middle East, and Africa. In short, a global breadth and inclusive framing of culture, setting, and opportunity, and,

► the exciting range of opportunity and people that computing enables, showcasing a variety of strong role models for young professionals.

In fulfilling this charter, we hope the feature will become a “must read” serial for young computing professionals, inspiring their passion and cultivating an expansive view of contribution, impact, and possibility.

Finally, I’d like to add a personal thanks to Mei Kobayashi for her energy, advocacy, and hard work to help define and launch this new feature!

Andrew A. Chien, EDITOR-IN-CHIEF



NAME

Mei Kobayashi

BACKGROUND

Born in Tokyo, grew up in Berkeley, CA, USA

CURRENT JOB TITLE/EMPLOYER

Manager, Customer Services, NTT Communications

LAST DEGREE

Ph.D. Applied Mathematics, University of California at Berkeley

Hello! I am a computational chemist-turned-applied mathematician, who joined ACM in the latter part of my

career as my technical interests shifted toward computer science. I’ve been serving on *Communications’* News Board for several years, and recently, I had the opportunity to brainstorm with Editor-in-Chief Andrew Chien on broadening *Communications’* coverage to more accurately reflect the demographics of the ACM membership. We agreed that greater attention should be given to work by computing professionals outside of academia (for example, industry, government, NGOs, start-ups). *Communications* needed to create a new forum to encourage more participation by these members and leverage their expertise. Then, one day it clicked! The result: this new feature, which showcases careers paths of computing professionals. These personal stories will demonstrate the infinite range of possibilities of what one can do with their computing knowledge.

For this inaugural year, we invited scientists who work in large corporations as well as mavericks who have acquired unique skillsets to make innovative products and services while zig-zagging through a variety of roles. At the same time, we sought to balance geographic and demographic diversity in authorship. Despite their vastly different backgrounds, experiences, and lifestyles, all of our guests write with great passion and infectious enthusiasm, making for delightful reads. It is our hope their collective wisdom will encourage students and young scientists to realize their full potential in a job that also brings joy and fulfillment. As with all new endeavors, we appreciate constructive feedback. Thank you.

^a Chien, A. Computing is the secret ingredient (well, not so secret). *Commun. ACM* 60, 12 (Dec. 2017), p. 5.

DOI:10.1145/3391911

Computing enabled me to . . .

Automating Automation: CS at the Heart of the Manufacturing Economy

**NAME****Arquimedes Canedo****BACKGROUND**

Born and raised in Mexico, moved to Tokyo for graduate school, currently in Princeton, NJ

CURRENT JOB TITLE/EMPLOYER

Principal Key Expert Scientist, Siemens Corporate Technology

EDUCATION

Ph.D. Computer Science, University of Electro-Communications, Tokyo, Japan


A friend's birthday barbecue is coming up in a few days and we decide to surprise our friend with a new grill. Online, a manufacturer's website allows us to customize the grill. Unknown to us, the design space consists of billions of grills and we create a one-of-a-kind design that has never been produced before. Designing, procuring, producing, and delivering a unique product in a short time, at an affordable price, with minimal human intervention, requires complex interactions across software-hardware, systems, and time scales. This process, known as *lot size one* production, is only possible through the use of autonomous production systems.

Manufacturing is a cornerstone of a country's innovation pipeline and many companies are reshoring to keep manufacturing and R&D as close as possible. While most people think manufacturing is low-tech and boring, I must disagree! The extreme complexity in manufacturing demands high-tech knowledge at all job levels. For instance, just think of a modern car with 30,000 parts and 100 million lines of code. Every part—down to the screws—is intricate and has been through a logistical maze. As a computer scientist, this complexity is why I find a career in manufacturing so rewarding.

Even simple products like our grill are designed and tested digitally using design automation and simulation tools. It takes years of experience to become a good designer. This is why automation is so important. Using machine learning, I'm able to reduce the time required for designing products from months to mere hours as well as free up time for the engineers, allowing them to focus on design in terms of functions, behaviors, and structures instead.

Imagine this: Our new grill is born in a factory that produces hundreds of one-of-a-kind grills, motorcycles, and other products. Autonomous unmanned vehicles are transporting parts and tools from storage to work cells; dozens of robots are working on different parts and assembling products; and new processes are being invented and implemented on-the-fly. This is the vision for *general-purpose automation*. In pure software, automation is straightfor-

ward. However, automating hardware that operates in the real world and is subject to strict requirements (such as regulatory and safety) is extremely challenging. I spend countless hours bridging this reality gap. For example, I've been trying to digitize real-world objects into "digital twins." These digital representations of things, built through sensing and perception, can be used by computer algorithms to make informed decisions and control the system for better outcomes. As we transition these technologies and get closer to automating automation, the time from design to delivery of new products will be reduced significantly.

Automation is multidisciplinary in nature. I'm lucky enough to have the opportunity to work with practitioners who come from a variety of backgrounds, all with the common goal of navigating the complexities of the real world in order to make it a better place—or at least one where our custom-made grill is delivered within the same day. As a computer science graduate student, my career started in a conventional way. My first job at IBM's Tokyo Research Laboratory exposed me to the intricacies of the physical world, and I was captivated by the idea of software being in full control of things. Throughout my career, I've found that many people often forget about the complexity of the machines that produce the car, the grill, and so on. As automation continues to improve, I'm excited to grow my skillset and develop more complex algorithms for designing and producing products that everyone can enjoy. 

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

twitter

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

DOI:10.1145/3386312

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

Teaching CS Humbly, and Watching the AI Revolution

Mark Guzdial on a book that changed his thinking about teaching computer science, and Jiajie Zhang on the AI Revolution.



Mark Guzdial
Developing
Computational
Solutions With Humility:
Recommending
Morgan Ames'

'The Charisma Machine'

<http://bit.ly/2vvnw8ri>

February 23, 2020

Morgan Ames' book *The Charisma Machine* has influenced my thinking more than any other book I've read in the last couple years. She writes the story of the XO Laptop from the One Laptop Per Child (OLPC) project. Her summary of the book appears on her website:

The Charisma Machine chronicles the life and legacy of the One Laptop Per Child project and explains why—despite its failures—the same utopian visions that inspired OLPC still motivate other projects trying to use technology to “disrupt” education and development.

Announced in 2005 by MIT Media Lab cofounder Nicholas Negroponte, *One Laptop Per Child* promised to transform the lives of children across the Global South with a small, sturdy, and cheap laptop computer, powered by a hand

crank. In reality, the project fell short in many ways, starting with the hand crank, which never materialized. Yet the project remained charismatic to many who were enchanted by its claims of access to educational opportunities previously out of reach. Behind its promises, OLPC, like many technology projects that make similarly grand claims, had a fundamentally flawed vision of who the computer was made for and what role technology should play in learning.

The quickest possible summary of the book might be that Negroponte convinced a bunch of people to buy into his vision for the XO laptop, and he turned out to be wrong. The point of the book isn't the punchline, but is Morgan's storytelling and the empathy she has for all the participants in the narrative. She understands the big and important vision that Negroponte was promoting and why he was promoting it. She develops the construct of “charisma” to explain why people bought in to this vision. Most of all, she has empathy for the teachers and children who tried to use the XO laptop—mostly unsuccessfully.

She talks specifics about what worked and what didn't, based in part on her fieldwork in Paraguay. She found that most children who owned XO laptops didn't use them, either because the laptops broke or because the students found them boring. While XO laptops were famously rugged while closed, they were often damaged while open. Features like the mesh network worked so badly that they were quickly turned off. Teachers struggled to incorporate the laptop into their lessons because of a lack of infrastructure (classes Morgan visited had only a single power plug), broken or failing XO laptops (battery life was much shorter than expected), or deleted software. The OLPC project insisted that the computers belong to the children, not the school, so children would simply delete programs to make room for videos and games.

I didn't understand all of her story. She talks about the XO as being developed by and for “technically precocious boys.” I don't see how gender played a role here. Using “boys” diminishes the roles of Cynthia Solomon

(co-developer of Logo with Seymour Papert, Wally Feurzeig, and Danny Bobrow), Paula Bontá (co-developer of Turtle Art, one of the more popular applications on the XO), and Mary Lou Jepsen, who served as the chief technology officer for the project.

There are several important themes developed in the book. The one that most resonated for me was the lack of a human-centered development process. Negroponte famously dismissed the idea of a pilot study.

“The days of pilot projects are over. When people say, ‘Well, we’d like to do three or four thousand in our country to see how it works.’ Screw you. Go to the back of the line and someone else will do it, and then when you figure out that this works, you can join as well.”

It’s not clear there was a feedback loop from the users back to the designers. Unused or unusable features might have been removed earlier if there was. Negroponte had enormous faith in his team’s ability to design software, without ever meeting any user, that would change the user’s life. In the software engineering world, this might be called an example of the “waterfall method” of development—literally, build the technology and throw it over the proverbial wall:

“We’ll take tablets and drop them out of helicopters into villages that have no electricity and school, then go back a year later and see if the kids can read.”

It takes humility to design software that humans will use successfully. The human-computer interaction (HCI) community has developed a rich set of methods for figuring out what users need and might use, and for evaluating the potential of a new interface. To use these methods requires us to recognize our limitations—that we are unlikely to get the design right the first time and that our users know things that we don’t.

Ames doesn’t use the word *hubris* in describing the OLPC Project. Rather, she uses *charisma*—“a charismatic technology derives its power experientially and symbolically through the possibility or promise of action: what is important is not what the object is but how it invokes the imagination through what it promises to do.” Everyone *wanted* the XO laptop to succeed, for the technology to have

a powerful effect on children’s lives in the Global South. In hindsight, the challenges of the Global South are probably not solvable with laptop technology. But our imagination is still captured by the possibility. She doesn’t use the word “humility” in describing the project, but that’s probably the component that was most needed.



Jiajie Zhang
AI is to Medicine Today
What the X-ray was to
Medicine a Century
Ago, and Much More...

<http://bit.ly/2rQQ098>

December 13, 2019

Just as the X-ray machine invented more than a century ago enables doctors to see images of structures inside the human body, recent breakthroughs in artificial intelligence (AI) and machine learning are enabling doctors to not only see, but to predict, previously unidentified patterns within medical and biological data that can inform individualized disease prevention and care, as well as biomedical discovery.

For many clinical tasks, AI can often outperform—in speed and accuracy—trained clinicians. Here, I am providing only a few examples from a rapidly expanding list of medical AI applications. AI systems developed by training with massive numbers of images can recognize melanoma from photographs of the skin; diabetic retinopathy and glaucoma can be diagnosed by AI from OCT images; and endovascular thrombectomy eligibility can be determined by AI using the CT scans of stroke patients. AI systems developed from human behavioral data can detect early signs of Parkinson’s from typing movement of the hands; depression can be determined from sleep patterns tracked by mobile devices; and fall risks can be predicted through gait analysis videos. AI systems developed from longitudinal electronic health records (EHRs) can predict a multitude of health conditions such as myocardial infarction, heart failure, sepsis onset, and stroke, as well as assist in the analysis of critical quality and safety issues that include ICU mortality and

hospital readmission. In addition, AI systems utilizing EHR data can detect previously unknown drug-drug interactions, adverse drug events, and new functions of existing FDA-approved drugs. AI systems for genomic data can establish previously unknown correlations between diseases and genotypes. For clinical operations, AI algorithms can transcribe a doctor-patient conversation in real time into clinical notes and then further convert them into structured codes in EHR for clinical decision support and billing, thereby reducing the physician’s workload and facilitating more direct patient-doctor interaction.

We are in the throes of a fundamental economic and societal transformation.

The Agricultural Revolution that took place around 10,000 BC liberated people from food insecurity via farming; the Industrial Revolution that commenced 200 years ago began to free people from grueling physical labor through machines; and the Artificial Intelligence (AI) Revolution (<http://bit.ly/2TxAZOA>) occurring now is liberating people from cognitive labor through powerful computing, universal connectivity, and massive data. While AI has been disrupting and changing many industries, including information access, communication, retail, manufacturing, agriculture, entertainment, travel, finance, and education, its seismic tremor is just beginning to impact the largest industry in the U.S., which accounts for nearly one-fifth of its GDP: Healthcare.

The AI Revolution promises to be an exciting era. With virtually unlimited potential, medical AI is rapidly evolving to produce ever greater numbers of increasingly advanced clinical applications that will dramatically improve patient care, disease prevention, and biomedical discovery. It’s great to be part of that transformation!

Mark Guzdial is professor of electrical engineering and computer science in the College of Engineering, and professor of information in the School of Information, of the University of Michigan. **Jiajie Zhang** is dean, professor, and Glassell Family Foundation Distinguished Chair in Informatics Excellence at the School of Biomedical Informatics of the University of Texas Health Science Center at Houston.

© 2020 ACM 0001-0782/20/5 \$15.00

SHAPE THE FUTURE OF COMPUTING. JOIN ACM TODAY.

www.acm.org/join/CAPP

SELECT ONE MEMBERSHIP OPTION

ACM PROFESSIONAL MEMBERSHIP:

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

ACM STUDENT MEMBERSHIP:

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

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

PAYMENT INFORMATION

Name _____

Mailing Address _____

City/State/Province _____

ZIP/Postal Code/Country _____

- Please do not release my postal address to third parties

Email Address _____

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

- AMEX VISA/MasterCard Check/money order

Credit Card # _____

Exp. Date _____

Signature _____

Purposes of ACM

ACM is dedicated to:

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

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

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

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

BE CREATIVE. STAY CONNECTED. KEEP INVENTING.



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

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

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

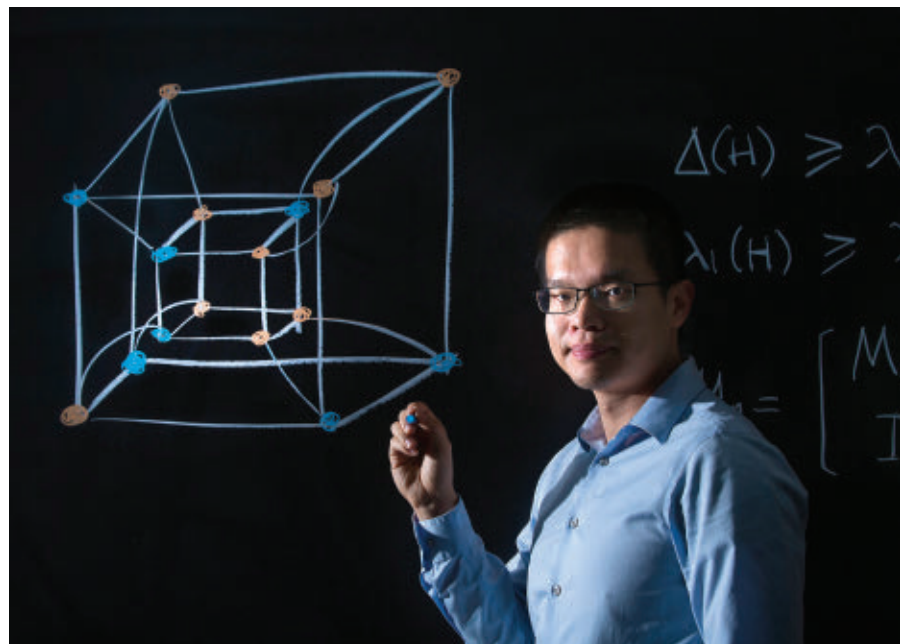
A Proof from ‘The Book’

A decades-old conjecture about computational complexity is confirmed in just a few pages.

IN 1637, THE French mathematician Pierre de Fermat scribbled in a book that he had a proof for a theorem, which the margin was too small to contain. It seems likely that he was mistaken about his famous “last theorem,” since no such proof was found for 358 years, and even then it required more than 100 pages and used mathematics that didn’t exist in his time.

By contrast, the Boolean “sensitivity conjecture” is relatively recent, but after nearly 30 years and repeated failures it seemed likely that any proof would also be a long and difficult slog. In July, 2019, however, mathematician Hao Huang of Emory University in Atlanta posted a short paper that completed its proof in a couple of pages in a way that experts found very convincing. Indeed, they immediately described it as “from the book,” a tome imagined by the great 20th-century mathematician Paul Erdős in which God records the shortest and most illuminating proof of each theorem.

The new proof closes a nagging loophole by confirming that “sensitivity” of any Boolean function is closely related to other measures of its computational complexity. “In the past, com-



Emory University mathematician Hao Huang, who developed the proof of the sensitivity conjecture.

puter scientists studied a lot of different complexity measures. All but one of them were known to be polynomially related,” Huang said, meaning that their possible values are constrained by powers of the others. “My work is basically to put the sensitivity—the last exception—into this category.”

Algorithmic complexity theory aims to provide strict upper and lower bounds on the difficulty of calculations, especially as the problems get larger. Its best-known open question, $P \neq NP$, concerns the existence of problems whose solution can be verified with computational resources that

grow as a polynomial function of the problem size but which require much more effort to solve in the first place. The new result is “not going to spring open $P \neq NP$,” cautioned Kenneth Regan of the University at Buffalo (part of the State University of New York system). Nonetheless, he noted that the sensitivity conjecture is closely related to “the tools that people have been using to try to get a handle on $P \neq NP$.”

Measures of Complexity

Boolean functions are at the heart of digital computation, producing a one-bit (zero or one) output, based on the values of a string of input bits. Sensitivity is one measure of a Boolean function’s complexity: Considering all possible input strings, the sensitivity is the largest number of input bits whose individual flipping (from zero to one or vice versa) changes the output.

Sometimes flipping any input bit is enough. For example, the parity function, which reports whether the input

has an even or odd number of ones, changes value for any input bit, for any input string. Thus its sensitivity has the maximum possible value, equal to the number of input bits. The logical “and” of all the bits also has maximum sensitivity, because, for an all-ones input, then changing any bit to zero flips the output.

Other functions are less sensitive, so that for all input strings there are some bits that cannot on their own change the answer. For example, the OR-of-AND function asks, for a bunch of non-overlapping blocks of input bits, whether any of them has all ones for inputs. Changing the answer from yes can only be done by flipping a bit in a unique all-ones block, while changing the answer from no can only be done by flipping the last remaining bit that is not zero in one of the blocks. The sensitivity is the larger of the number of bits per block or the number of blocks.

Computer scientists have explored several other measures of computational complexity. The “block sensitiv-

ity,” for example, quantifies how many blocks containing multiple input bits change the output when they are simultaneously flipped. Another measure is the function’s “degree,” which is the highest total exponent in the polynomial that reproduces the output when the inputs are zero or one.

Other measures include decision-tree depth (the minimum number of yes-or-no questions needed to guarantee knowing the output), as well as its quantum and random variants, and certificate complexity (the number of input bits needed to guarantee knowing the output). Researchers proved long ago that all these other complexity measures are closely related. Specifically, upper or lower bounds on their values, for any Boolean function, can be expressed as polynomials of the others, which is useful for proofs. “If they are polynomially related, they are roughly of the same order of magnitude,” Huang said, “so instead of looking at the more difficult ones, you can look at a simpler one.”

ACM News

The Shuttering of Corporate Datacenters

Corporate datacenters are being decommissioned rapidly. In a blog post, market research firm Gartner forecast that 80% of enterprises will have shut down their traditional datacenters by 2025.

Many companies are now migrating their in-house datacenters to the cloud. Oracle predicts 80% of enterprise workloads will move to the cloud by 2025.

According to Richard Villars, vice president for datacenter and cloud at market research firm IDC, the cloud is only one factor contributing to datacenter decommissioning. “Virtualization and converged infrastructure allowed many corporations to get a lot more capacity in their datacenters,” Villars said. “With these technologies, you could now do the same amount of work on 40 or 50% of the footprint.”

Bill Vasquez, senior vice president of Strategy & Business Development at ITRenew, which specializes in onsite datacenter decommissioning and data erasure services, agrees

decommissioning is growing at a rapid pace. One of the main growth drivers is the sheer volume of hardware deployed, he says.

Companies retain servers for four years on average, and the number of servers in a datacenter can run as high as 80,000. IDC reports enterprise computing is at near-historic highs, with next-generation workloads and advanced server innovation driving server demand at the end of 2019 to one of the highest levels in 16 years, according to a recent IDC Quarterly Server Tracker. It is the deployment of these new servers that is spurring the decommissioning of the older equipment they are replacing.

“Between the growth in data usage and storage, and the emergence of new technologies that require more and more computing power, like artificial intelligence, machine learning, augmented & virtual reality, and the Internet of Things,” Vasquez says. “Decommissioning shows no sign of slowing down.”

Considering all the data that resides on the servers from decommissioned datacenters, “The best way to verify data has been destroyed is to wipe it with 100% sector-verified erasure, and electronically capture the serial number of both the host unit and the media itself with a solution like Teraware,” Vasquez says, pointing to ITRenew’s data-wiping software. Wiped drives can be reconciled against an asset inventory system for further verification, he adds, and enterprises in industries with higher security requirements may require the units to be shredded after they have been wiped.

“How we actually destroy the media that has the information on it is where the rubber meets the road from a certitude perspective,” says Bob Johnson, CEO of the National Association of Information Destruction (NAID). “Some data centers’ internal IT staff may do their own wiping and disassembly before disposal, but in a large percentage of cases, they are

turning over their equipment and relying on a third party to perform the data destruction, as well as the equipment removal and recycling.”

NAID verifies secure data destruction companies’ services compliance with data protection laws through audits by accredited security professionals, fulfilling customers’ regulatory due diligence obligations.

The data owner is always responsible for the protection of its data, as well as for regulatory compliance. “They are not able to contract that away,” Johnson says.

Vasquez agrees the owner of the hardware is ultimately responsible for the data and its destruction, which is why it is of paramount importance for them to complete thorough due diligence when deciding on a potential data destruction partner. “Only partners who provide the most stringent security solutions should be trusted with this work,” he says.

—John Delaney is a freelance writer based in New York City, NY, USA.

Coloring Hypercubes

Some important conclusions have been mathematically expressed only in terms of sensitivity, however, and until now there had been no proof that they were also members of this club, although it was widely thought to be. Indeed, in 1992, Noam Nisan of the Hebrew University of Jerusalem and Mario Szegedy, then at AT&T Bell Laboratories, explicitly conjectured that the block sensitivity of a function could not exceed some fixed power of its sensitivity.

A critical strategy for proving this “sensitivity conjecture” was provided the same year by Craig Gotsman and Nathan Linial, both then at the Hebrew University of Jerusalem, who connected the sensitivity with the graph-theoretical properties of corners of a hypercube.

The coordinates of any corner of an n -dimensional hypercube can be written as an n -bit string of zeroes and ones. A Boolean function then corresponds to coloring the corners, say red when the function is one and white when it is zero.

If exactly half of the corners are red (and half white), they can be arranged so that no corner has a like-colored neighbor, for example by coloring them according to the parity function of their coordinates. If even one additional corner is colored red, however, it turns out that at least one of the red corners must have many red neighbors. The question is how many? The largest number, among all red corners, is called the degree of the framework (which is not the same as the degree of the function).

In this picture, the sensitivity of a function is the maximum number of white corners (opposite output) sharing an edge (one input-bit flip) with any red point. The remainder of the n edges are connected to red points, so the sensitivity is closely connected to the subgraph’s degree.

The half-page proof by Gotsman and Linial showed roughly that a bound on a subgraph’s degree, as a function of n , is equivalent to a bound on the sensitivity of a Boolean function, as a function of the degree of its polynomial. Thus, a theorem about one becomes a theorem about the other, opening the door to a proof of the sensitivity conjecture.

The proof “provides a useful addition to the toolbox of mathematics and computer science that hopefully will see more application in the future.”

Changing Signs

In spite of this clear roadmap, and decades of attempts, this promise was only realized with Huang’s proof. What Huang showed was that if even one more than half of the hypercube’s corners are red, at least one of them will have at least \sqrt{n} red neighbors, precisely what Gotsman and Linial suspected. This implies that the degree of the function is no greater than the square of its sensitivity. Together with a previous theorem that the block sensitivity is no greater than the square of the degree, this means that the block sensitivity does not exceed the fourth power of the sensitivity. This confirms the conjecture and thus connects sensitivity to all the other complexity measures.

The proof is based on something called the adjacency matrix, whose 2^n rows and 2^n columns correspond to the corners of the hypercube, and whose elements are zero unless the corners are same-colored neighbors. The critical trick is that the non-zero elements for neighbors are usually assigned a value of +1, but Huang assigned some of them a value of -1. Carefully choosing the negative values allowed him to use known matrix theorems to show that at least one row of the matrix must have at least \sqrt{n} entries.

Aaronson wrote on his blog, “How could such an elementary 1.5-page argument have been overlooked for 30 years? I don’t have a compelling answer to that, besides noting that ‘short’ and ‘elementary’ often have little to do with ‘obvious.’ Once you start looking at ... this matrix ..., the pieces snap together in precisely the right way—but how would you know to look at that?”

“For making a calculation it’s

straightforward to go through every step,” Huang noted. In contrast, “a proof is easy to verify, but it’s difficult to come up with a new proof.”

Since encountering this problem in 2012, Huang wrote in a comment on Scott Aaronson’s blog, “I revisited this conjecture every time I learned a new tool—without any success, though.”

More recently, he added, “I had been looking at other kinds of problems at the same time, and I used this adjacency matrix a lot,” he said. “I realized that it can be also applied to this particular sensitivity conjecture, and that’s how I came out with the proof.”

Regan, who had previously explored similar ideas, said the cancellations allowed by making some matrix elements negative is critical, and are reminiscent of the interference that quantum algorithms exploited.

Huang said the proof “provides a useful addition to the toolbox of mathematics and computer science that hopefully will see more application in the future,” although he noted that in math this process often takes many years. He also hoped that the work would be an inspiration for graduate students to attack long-standing unsolved problems. **■**

Further Reading

Huang, H., *Induced subgraphs of hypercubes and a proof of the Sensitivity Conjecture*, July 1, 2019 <https://arxiv.org/abs/1907.00847>

Klarreich, E., *Decades-Old Computer Science Conjecture Solved in Two Pages*, *Quanta*, July 25, 2019, <http://bit.ly/36ipHTo>

Gotsman, C., and Linial, N., *The equivalence of two problems on the cube*, *Journal of Combinatorial Theory, Series 1*, Volume 61 Issue 1, September 1992, pp. 142-146.

Blog Posts:

Sensitivity Conjecture Resolved, Scott Aaronson at *Shtetl-Optimized*, July 2, 2019.

Tools and Sensitivity, Ken Regan at *Gödel’s Lost Letter and P=NP*, July 12, 2019.

Amazing: Hao Huang Proved the Sensitivity Conjecture! Gil Kalai at *Combinatorics and More*, July 2, 2019

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

© 2020 ACM 0001-0782/20/5 \$15.00

Will RISC-V Revolutionize Computing?

The open instruction set for microprocessors promises to reshape computing and introduce new, more powerful capabilities.

MODERN COMPUTING DEPENDS ON many components to deliver fast speeds and high performance, yet few play a more integral role than a *reduced instruction set computer*, commonly known as RISC. Although the instruction set architecture (ISA) comes in different shapes and forms—and it supports numerous systems and devices—there is a common denominator. RISC allows microprocessors to operate with fewer cycles per instruction (CPI) than a complex instruction set computer (CISC).

An ISA is at the heart of computing, of course. “It is the basic vocabulary that allows hardware and software to communicate,” says Dave Patterson, professor of computer science at the University of California, Berkeley, and an ACM A.M. Turing Award recipient who essentially coined the term and developed early RISC computing models. Over the last couple of decades, two major entities, Intel and ARM, have largely controlled ISAs. Their proprietary microprocessors run everything from laptops to cloud servers, and smartphones to Internet of Things (IoT) devices. Today, it’s difficult to find a computing device without Intel or ARM processors inside.

All of this is about to change. A free, open instruction set called RISC-V, conceived by Patterson and fellow Berkeley professor Krste Asanović, along with their students, is turning the microprocessor industry upside down. The royalty-free ISA, which debuted in 2011, supports new and more specialized microprocessor designs that soon will appear in traditional computing devices as well as wearables, home appliances, robotics, autonomous vehicles and factory equipment. The appeal? “RISC-V delivers a very high level of flexibility at a much lower cost than



proprietary RISC. It allows users to produce custom chips suited to specific applications,” Asanović explains.

Following Instructions

The introduction of RISC-V coincides with other major changes in the semiconductor industry. It’s no secret that CMOS transistor scaling is slowing. Even with recent breakthroughs in design that push density and performance to new levels, Gordon Moore’s longstanding prediction of doubling transistors every two years—“Moore’s Law”—no longer holds. As semiconductor advances slow—while performance demands continue to grow—the ability to design more advanced computing devices and fuel innovation is threatened. “Moving forward, the logical path is to add extensions to the basic instruction sets on microprocessors for application domains,” Patterson explains.

The appeal of RISC-V is undeniable. A common ISA means that different implementations and use cases for

the ISA can tap the same core software stack, thus minimizing porting efforts to compilers, operating systems, and other software. The main advantage of RISC-V is not that it is a new variation or iteration of RISC, but that it is an open ISA. Hence, there’s an expectation that the model will produce the software stack needed to put RISC-V on the commercial map. Yet, at the same time, there is also a fear that giving users the ability to alter the ISA will lead to a balkanization of the RISC-V software ecosystem.

Asanović and Patterson began working on the fifth-generation RISC instruction set in Berkeley’s Parallel Computing Lab (Par Lab) in 2010. The project was born out of frustration over the lack of flexibility with proprietary ISAs. “We couldn’t do some of the research we wanted to do,” Patterson recalls. The pair took aim at a persistent industry problem: an inability to customize chips for specific purposes. The initiative was rooted in their own needs. “Since we couldn’t get permis-

sion from Intel or ARM to use or modify their proprietary instruction sets, we decided to develop our own instruction set for our own research and to help the research of fellow academics.”

The project caught the collective eye of the computing industry, and, with \$10 million in lab funding from Microsoft and Intel and additional funding from DARPA, blazed forward. “It became apparent that many people wanted something akin to a Linux operating system for microprocessors. People desired an open instruction set that would allow anyone in the world to build chips using an open and common vocabulary,” Patterson says. In 2014, RISC-V made its official public launch and, by then, the idea had garnered enough momentum to spawn the nonprofit RISC-V Foundation (riscv.org), which serves as a clearinghouse for research, standards, and industry collaboration. It now boasts more than 425 members.

Over the last couple of years, RISC-V has crept into mainstream computing. For example, Samsung announced that it will use RISC-V cores in its 2020 5G smartphones. The electronics giant also will tap RISC-V cores for artificial intelligence (AI) image sensors, security management, AI computing, and machine control systems.

Others are following suit. Western Digital, NVIDIA, and Qualcomm also have announced that they will use RISC-V for applications ranging from solid-state drives (SSD) and hard-disk drives (HDD) to graphics processing units (GPUs) used for smartphones and machine learning.

Reducing RISCs

The appeal of RISC-V is clear. “It opens entirely different possibilities through a modular design that allows users to add specific extensions based on specific computing needs,” says Calista Redmond, CEO of the RISC-V Foundation. “The design bypasses a one-size-fits-all approach with prepackaged features and capabilities that you may or may not require—and the performance and energy drain that come with them.” No less important, RISC-V wrestles control of microprocessors away from dominant industry giants Intel and ARM. “Instead, you have a diversity of suppliers and the innovations that come with them,” she adds.

“Companies can build their own core to fit their own needs. They can have greater visibility into how the core runs, and even develop their own security features.”

The result will be chips designed, built, and optimized for specific tasks. “There is nothing in the design that limits the application domain,” Asanović explains. For instance, a RISC-V chip might be used to focus on a specialized AI task such as image recognition or machine language translation, or it could be used to establish a microcontroller framework that spans generations of devices and products. This would allow a business to bypass future R&D, as well as ongoing licensing and royalty requirements. “Companies can build their own core to fit their own needs. They can have greater visibility into how the core runs and even develop their own security features,” he says.

In fact, many predict RISC-V will emerge as an industry standard. While RISC-V won’t replace proprietary RISC, its custom extensions will enable entirely new types of applications, capabilities, and perhaps even devices. Says Asanović, “No longer will businesses be forced to adapt to the features of a chip. They will create a chip that matches their specific needs.” Adds Abhi Shelat, an associate professor of computer and information science at Northwestern University, “In terms of use and cost for lower-end processors, this chip might dominate because of open-source economics. As the tool-chain becomes a standard, it will be cheaper than using proprietary alternatives for many tasks.”

Processing Change

Not surprisingly, RISC-V has doubters and naysayers. Critics argue the standard could introduce interoperability

ACM Member News

TOWARD THE TOPOLOGY OF DATA ANALYSIS



“In college, I was required to take some programming courses, and I immediately knew this was

my area,” says Tamal Dey, a professor of computer science at The Ohio State University. Several years later, while studying for his master’s degree, Dey was exposed to computational geometry, and realized this was the niche of computer science for him.

Dey graduated Jadavpur University in Kolkata, India, with a bachelor’s degree in electronics. He earned a master’s degree in computer science from the Indian Institute of Science, Bangalore, and completed his Ph.D. in computer science at Purdue University.

His career started with faculty positions at the Indian Institute of Technology, Kharagpur, and research scientist positions at the University of Illinois and the Max Planck Institute in Saarbruecken, Germany, before joining the faculty at The Ohio State University in 1999.

Dey’s research focuses on computational geometry and topology, with applications in computer graphics, geometric modeling, mesh generation, and shape analysis. His current interest is in topological data analysis, which sprung from computational topology.

“Computational topology has two aspects. One is recognizing various mathematical structures that can be extracted out of shape and data, and the other is the algorithmic issues of extracting them,” Dey says. “I am currently focusing on the algorithmic aspects because they truly offer an opportunity to marry classical mathematics with the new discoveries in algorithm designs.”

Dey feels scale, noise, dimensions, time, and algorithmic space constraints will bring new algorithmic challenges, and new mathematics will be needed to resolve them. “I wish to address these issues,” he says.

—John Delaney

challenges across different types of RISC-V devices and ecosystems. Industry fragmentation and potential interoperability issues could emerge as different versions of the ISA take shape. In addition, binary compatibility with certain types of devices, such as smartphones, could present problems. Many apps currently are coded to conform to ARM instruction sets. Likewise, the platform could encounter challenges in certain high-end cloud environments, where enormous resources are required to build systems that rival proprietary ISA designs.

There are also questions about how the instruction set will evolve—and, for now, a lack of powerful tools for managing the technology. The RISC-V Foundation is promoting advances through collaborative standards and protocols. However, success hinges heavily on ongoing cooperation. As a result, some industry players, particularly those that stand to lose the most from an open ISA range, have taken aim at the technology. For example, ARM set up an anti-RISC-V website in June 2018. It was taken down a few days after going live when staff at ARM objected to the tactic. Then ARM announced in November 2019 that it was opening up its proprietary instruction set for Cortex M cores so customers can tweak and customize instructions.

Nevertheless, RISC-V is rapidly taking shape. A November 2019 report from Semico Research Corp. predicts the market for RISC-V CPU cores will reach 62.4 billion by 2025—or about 6% of the overall CPU core business. “Companies are turning to RISC-V solutions for a wide variety of applications and to address a wide range of performance and volume requirements,” says Semico president Jim Feldhan. Communications, transportation and industrial settings are particularly hot sectors for RISC-V. “The idea of developing more innovative and efficient chips is incredibly appealing,” Feldman says.

Security could also emerge as a primary selling point for RISC-V. Presently, there’s no way to definitively know whether spyware or malicious code has been embedded at the BIOS level of a chip. “Today, microprocessor security is a black box,” Patterson says. An open-source approach offers a couple of potential advantages. First, those using RISC-V chips would know exactly

The future of RISC-V certainly looks bright. In addition to traction in the corporate world, more than two dozen universities are on board with RISC-V.

what is taking place on the microprocessor. Second, it would be possible for users to develop instruction set extensions and produce designs that focus on specific security needs. Companies and government entities could develop chips that are known to be free of embedded spyware or malware.

Powering the Future

The commercial introduction of RISC-V fills a longtime void in the computing industry, Redmond argues. Not only does it break the existing ISA duopoly of ARM and Intel, and allow users to take control of their own destiny, it establishes an open framework to fuel global collaboration and innovation. “It’s a model that has demonstrated success over the last century in many different forms—from telephones and cars to networking and software,” she says. “RISC-V represents a next logical phase of the concept and it is particularly suited to the IoT and an increasingly interconnected world.”

The future of RISC-V certainly looks bright. In addition to traction in the corporate world, more than two dozen universities are on board with RISC-V. Not only are researchers looking to develop niche and boutique RISC-V chips to aid in their studies, schools including the University of California at Berkeley, the Massachusetts Institute of Technology, Cornell University, the University of Cambridge, and Tsinghua University in Shenzhen, China, have begun to develop educational materials and instructions related to the design, engineering, and use of RISC-V. “This is planting the seeds for more widespread adoption and greater use of the framework in the future,” Redmond explains.

All of this will likely fuel a level of disruption the semiconductor industry has not witnessed in many years. Patterson has described the introduction of RISC-V as “a new golden age for computer architecture.” Says Michael Taylor, an associate professor in the School of Computer Science and Engineering at the University of Washington in Seattle, “There are no serious technical or practical issues with RISC-V. It will eventually supplant x86 and ARM as the primary instruction set for microprocessors. It will fundamentally change the computing world.” **Q**

Further Reading

Hennessy, J.L., and Patterson, D.A. **A New Golden Age for Computer Architecture**, *Communications*, February 2019, Vol. 62 No. 2, Pages 48-60, 10.1145/3282307. <http://bit.ly/2B420R1>

Asanović, K., and Patterson, D.A. **Instruction Sets Should Be Free: The Case For RISC-V**. Electrical Engineering and Computer Sciences University of California at Berkeley. Technical Report No. UCB/EECS-2014-146. August 6, 2014. <https://people.eecs.berkeley.edu/~krste/papers/EECS-2014-146.pdf>

Lee, Y., Waterman, A., Cook, H., Zimmer, B., Puggelli, A., Jaehwa, K., Bailey, S., Blagovic, M., Chiu, P., Avizienis, R., Richards, B., Bachrach, J., Patterson, D., Alon, E., Nikolic, B., and Asanović, K. **An Agile Approach to Building RISC-V Microprocessors**, *IEEE Micro*, Volume: 36, Issue: 2, Mar.-Apr. 2016, pp. 8-20. <https://ieeexplore.ieee.org/abstract/document/7436635>

Gautschi, M., Schiavone, P.D., Traber, A., Loi, I., Pullini, A., Rossi, D., Flamand, E., Gürkaynak, F.K., and Benini, L. **Near-Threshold RISC-V Core with DSP Extensions for Scalable IoT Endpoint Devices**, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Volume: 25, Issue: 10, Oct. 2017. <https://ieeexplore.ieee.org/abstract/document/7864441>.

Pulte, C., Pichon-Pharabod, J., Kang, J., Lee, S., and Hur, C. **Promising-ARM/RISC-V: a simpler and faster operational concurrency model**, *PLDI 2019 Proceedings of the 40th ACM SIGPLAN Conference on Programming Language Design and Implementation*, Pages 1-15, Phoenix, AZ, USA — June 22 - 26, 2019. <https://dl.acm.org/citation.cfm?id=3314624>

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

Deceiving the Masses on Social Media

The social media platforms like their freedom, but information gerrymandering may require legislation to fix.

CONSIDERABLE ATTENTION HAS been paid to the impact of social media on the electoral process, given that 55% of U.S. adults now get their news from social media either “often” or “sometimes”—an 8% increase from the previous year, according to a Pew Research study published in October 2019, which was conducted in July 2019. This is concerning because, according to the Pew data, 88% of Americans understand and realize that social media companies now have at least some control over the mix of the news consumed each day, and 62% believe social media companies have far too much control over the content mix of news that is seen each day.

Much of the concern about social media companies controlling the news is visceral. However, a study published in the journal *Nature* in September 2019 identified and explained mathematically how social media companies may unwittingly become a disruptive force to the democratic process, via a concept called information gerrymandering.

In electoral gerrymandering, political district boundaries are drawn by the party or group in power to create an unfair political advantage for them. In one scenario, voting district lines are drawn so the voting power of the opposing party’s supporters are spread out across many districts, thereby leaving the party in power with a majority of voters in a single district. Another tactic used in gerrymandering is to redraw district boundary lines so the voting power of an opposing group is concentrated in one district, thereby reducing their voting power in other districts. A third tactic is when a majority party seeks to manipulate district boundaries so that in each district, the majority power will always retain a population advantage.



Researchers from the University of Pennsylvania, the University of Houston, and the Massachusetts Institute of Technology have hypothesized that a similar phenomenon may be occurring in online social networks, driven by a mismatch or inequality of viewpoints reaching users, thereby creating information networks that may skew user perceptions, unbeknownst to the members of that social media group. A social media group could be defined as the followers of a particular social media personality, people who follow a particular hashtag, or the people who read or retweet a specific news article or periodical online. In this scenario, the information gerrymandering occurs by grouping users not just by their linkages to each other (such as by whom they follow on Twitter, which hashtags they follow, or which news sources they follow), but by the information viewpoint to which they are exposed.

The researchers began studying the power of information to change opinions by placing research participants in simulated elections, then conducting experiments to see how a group of people, evenly divided on an issue, might change their opinions if exposed to additional information that was weighted to one side of an issue, or if exposed to “zealots,” humans or bots that would argue for only one side of an issue.

After conducting repeated simulated elections around a specific issue, the researchers found that people may change their views based on additional and repeated exposure to information, even if it is contrary to their long-held viewpoint. Further, people may change their views to “go along with the group,” particularly if it appears that the majority of the group is voting a certain way. They then applied their research to an analysis of Twitter users.

“So what we found in this paper is that even when two parties have the exact same number of members, and when everyone is equally influential on a social network, the network can bias the outcome of a vote in favor of one party,” says Joshua Plotkin, Walter H. and Leonore C. Annenberg Professor of the Natural Sciences at the University of Pennsylvania, a co-author of the study.

Plotkin and his team noted that the structure and opacity of how people are connected on Twitter and other online social networks may lead to users being over-exposed to news and viewpoints that don’t line up with the expected viewpoints of the hashtags, influencers, or news sources they follow; for example, if one user follows another user who constantly retweets news stories, opinion pieces, or other content that comes from sources that the first user is not familiar with or cannot verify. Repeated exposure to this information or viewpoint may be able to sway susceptible users, and they may not even be aware it’s happening, according to Plotkin.

“The basic idea is that one party can be at a disadvantage even if it has the same number of members as the other party,” Plotkin says, based on how the network is structured, the type or bias of information delivered to the group, and the frequency of that information delivery.

This is not a problem, per se; in a democracy, different viewpoints, particularly those that challenge people to think rather than simply go along with their preconceived notions, can be a good thing. The problem, according to Alexander Stewart, an assistant professor in the Department of Biology and Biochemistry at the University of Houston and a co-author of the study, is that social networks do not make clear how their algorithms connect or separate users.

This is particularly the case on Twitter, Stewart says, as that service will suggest that a user follow other users or news sources without explicitly indicating the rationale, or what the linkage may be between them. Twitter—or any other online platform, for that matter—may not be able to police or stop a user

from posting information that comes from dubious sources, or stating one’s opinion as a fact.

Stewart doesn’t believe Twitter or any of the other social networks have set up their networks to favor one side or the other of certain arguments deliberately. “I don’t think it’s plausible to suggest that there is anybody deliberately wiring networks in a way to give one side an advantage over the other,” Stewart says. “Rather, what is being reflected are naturally emerging asymmetries which reflect dominance of one side of the discussion over another which arise both due to human behavior, but also it is due to choices made by the platforms that we use. Whether or not I see your tweets depends on whether I follow you, but also on whether your tweets appear in my news feed in a more prominent or less prominent way.”

Still, he says his work around information gerrymandering is useful for citizens to consider, particularly if they are getting the majority of their news and information from tweets or their Facebook news feeds, and if they are actively participating by retweet-

Milestones

National Academy of Engineering Names Three from ACM among Newest Members

Among the 87 new members and 18 international members recently elected to the National Academy of Engineering (NAE) are three who also belong to ACM, including past president Vicki L. Hanson.

NAE membership honors those who have made outstanding contributions to “engineering research, practice, or education, including, where appropriate, significant contributions to the engineering literature” and to “the pioneering of new and developing fields of technology, making major advancements in traditional fields of engineering, or developing/implementing innovative approaches to engineering education.”

Hanson was honored with NAE membership “for contributions to the design of accessible systems, and for leadership in the computer science and engineering community.” She served as ACM president from 2016 to 2018,

was a Distinguished Professor of the Rochester Institute of Technology in the HCI and Accessibility research groups, and also professor and chair of inclusive technologies at Scotland’s University of Dundee.

Hanson is an ACM Fellow, as well as a Fellow Chartered Information Technology Professional of the British Computer Society, and a Fellow of the Royal Society of Edinburgh. She has received the Royal Society Wolfson Research Merit Award, the ACM SIGCHI Social Impact Award, the Women of Vision ABIE Award for Social Impact, and the ACM SIGACCESS Award for Outstanding Contributions to Computing and Accessibility. She was elected to the ACM SIGCHI Academy in 2017.

Also newly elected to NAE membership were James F. Kurose and Fei Fei Li.

Kurose, a distinguished professor in the College of

Information and Computer Science at the University of Massachusetts, Amherst, was honored with NAE membership “for contributions to the design and analysis of network protocols for multimedia communication.”

Kurose has been on leave from the University of Massachusetts since 2015, serving as the assistant director of the National Science Foundation for Computer and Information Science and Engineering. Kurose also co-chairs the Networking and Information Technology Research and Development Subcommittee of the National Science and Technology Council Committee on Technology.

In the course of his career, Kurose has been awarded the IEEE’s Taylor Booth Award, the IEEE’s INFOCOM Achievement Award, and the ACM SIGCOMM Special Interest Group Lifetime achievement award.

Li, a professor at Stanford University and co-director of Stanford’s Human-Centered AI Institute and the Stanford Vision and Learning Lab, was honored with NAE membership “for contributions in building large knowledge bases for machine learning and visual understanding.”

Li served as director of the Stanford Artificial Intelligence Laboratory from 2013 to 2018. In 2017, she co-founded AI4ALL, a nonprofit aimed at increasing diversity in the field of artificial intelligence.

Among the awards Li has received are the IEEE PAMI Mark Everingham Prize, the J.K. Aggarwal Prize of the International Association for Pattern Recognition, the WITI@UC Athena Award for Academic Leadership of the University of California, and the Technical Leadership Abie Award of AnitaB.org. She was named a Fellow by ACM, and one of America’s Top 50 Women in Tech by *Forbes*.

ing, forwarding, and commenting on news stories. Indeed, an August 2019 Pew Research Center study found Facebook, along with Twitter and Reddit, had the highest proportion of respective users seeking news, with 73% of Facebook users seeking news on the platform, compared with 71% of Twitter users, and 65% of Reddit users.

This greater level of engagement on social media platforms increases the potential for misinformation, compared with 20 or 30 years ago, when most people passively watched television or read newspapers and did not actively share content across a wide circle of influence. Implications within a democracy may be profound, for those who may be influencing political thoughts and viewpoints, as well as the citizens who consume, repeat, and amplify them. Indeed, information gerrymandering may create social networks or groups in which repeated exposure to a particular viewpoint is “wasted,” as the group consists primarily of like-minded individuals. It also may inadvertently create groups of people that are particularly susceptible to changing their minds on an issue, if constantly and repeatedly exposed to another point of view.

“Are these platforms skewing our conversations, not necessarily deliberately, but in ways that makes it harder for us to reach informed decisions, or engage in compromise and collective decision making?” Stewart asks. “If people are being exposed to asymmetric information, and they can be made aware of that, then you can suggest to them how they might be able to combat that” by encouraging them to consider the sources of the material they consume.

Plotkin says additional regulation of how social media sites connect people, serve up news stories, and suggest tweets, should focus on making these algorithms more transparent so users can see how they are connected and why they are being fed specific new stories or content. That said, social media companies are unlikely to share the ‘secret sauce’ behind how they match users and content, given that is how they generate revenue.

“Technology data analysis and social media companies often claim that their algorithms are intellectual property and confidential trade secrets,” says Linda Priebe, a U.S.-EU data pri-

Information gerrymandering could inadvertently create groups of people susceptible to changing their minds on an issue, if repeatedly exposed to another point of view.

vacuity/security and federal relations attorney with Dallas, TX-based law firm Culhane Meadows PLLC. As a result, the social media companies “don’t want to lose what they believe to be their competitive advantage by being transparent about what the algorithms do, how they function, what information they rely on, [and even] to what extent they’re accurate.”

Priebe also notes that many companies feel, particularly around the distribution of news or political information, it’s not the social media companies’ role to become an arbiter of thought or discussion. “When it comes to maybe what people consider to be fake news and political advertising, they feel that they have a First Amendment right, or their customers have a First Amendment right, to express their political opinion and it’s not their role to censor that,” Priebe says. “So, there’s a concern there and I think all of that needs to be balanced in some way; fair to companies, but also fair to consumers.”

It should be noted that for its part, Twitter has prohibited political advertising, though the platform still allows people to share political news and views, so long as it does not advocate for or support a specific position or candidate. As such, information gerrymandering can (and likely still does) occur by users linking to or sharing content among their connections.

Other social media platforms have taken differing approaches. Facebook’s Mark Zuckerberg has taken the

position that political content is fair game, and relies on its users to determine the veracity and accuracy of political advertisements. Google, which includes YouTube, bans misinformation (via a team of fact-checkers who compare ad content to known and verified information sources) in some ads, such as around voting procedures, but does not have a policy prohibiting politicians from running false or misleading ads. Reddit, meanwhile, continues to permit political issue ads, and allows ads from political candidates at the federal level, but bans advertising around state or local elections.

Balancing transparency against competitive needs may make enacting legislation addressing information gerrymandering via regulation of algorithms extremely challenging.

“I think transparency is really key,” Priebe says. “What the Europeans have done with GDPR [the General Data Protection Regulation] is, they’ve striven to balance the business interests and concerns with individual and consumer privacy and protection concerns.”

Indeed, GDPR mandates specific consumer information disclosures, while also protecting the trade secrets of companies. For many U.S.-based companies, however, “my sense is that they’ve been very comfortable with the fact that their internal operations are pretty much behind the scenes, and not subject to any oversight,” Priebe says. As a result, she adds, “They can generate a lot of revenue and feel that European approach would restrict their revenue streams.”

Further Reading

Stewart, A.J., Mosleh, M., Diakonova, M. et al. Information gerrymandering and undemocratic decisions. *Nature* 573, 117–121 (2019) doi:10.1038/s41586-019-1507-6 <https://www.nature.com/articles/s41586-019-1507-6>

How Social Media Sites Handle Political Ads: <https://reut.rs/2TTBz9t>

What is Information Gerrymandering: <https://www.youtube.com/watch?v=SOIUHQkdaa0>

What is GDPR: <https://gdpr.eu/>

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

▶ James Grimmelmann, Column Editor

Law and Technology

What Role for Antitrust in Regulating Platforms?

Using regulation to protect competition and innovation.

GOOGLE, FACEBOOK, AND Amazon stand accused of various tactics to thwart competition and protect their market position. The Justice Department, Federal Trade Commission (FTC), and others are investigating the platforms for possible antitrust violations. These developments call to mind the browser wars and Microsoft's legal battles in the 1990s and 2000s. They also raise an important question: What role does antitrust have to play this time around?

As I explain, antitrust law is highly relevant to some—but not all—of the critics' complaints. If Google uses customer contracts to weaken Bing, antitrust law can and should step in. Likewise if Facebook bought Instagram, back in 2012, to neutralize it as a competitive threat. More challenging are complaints about product design, such as a platform's arrangement of search results. Even further afield are concerns the platforms are copying their rivals' best ideas. If Amazon copies another seller's decision to market

a particular product, no antitrust issue is raised, but copying combined with deception would raise serious concern.

Exclusion

Platforms enhance competition in several respects. One way is by competing in adjacent businesses beyond their "home" market.¹ For example, Google's Android mobile operating system pro-

vides important competition for Apple. Platforms also level the playing field for others. Amazon's cloud computing services enable startups to scale up without investing in infrastructure, while Google's search ads help them connect with customers at low cost.

Moreover, competing hard is not an antitrust violation. For example, the leading platforms have made enormous investments in machine learning and related hardware. The result is a better algorithm, with more relevant ads and product recommendations, and an improved user interface such as digital assistants. These product improvements are competition in action, even if they make life difficult for competitors.

An important antitrust problem arises when a platform, rather than competing on the merits, excludes a rival through its customer contracts. For example, during the browser wars, Microsoft's contracts with PC manufacturers restricted their promotion of the Netscape browser. Such contracts risked depriving Netscape of a critical mass of

Contracts that weaken a rival's offering by limiting its access to customers are a major focus of platform antitrust enforcement.



customers, thereby inhibiting its development into a Windows rival. Contracts that weaken a rival's offering by limiting its access to customers are a major focus of platform antitrust enforcement.

Critics see a replay of the browser wars in Google's contracts with Android handset manufacturers, which the European Commission has condemned as unduly restricting user access to competing search engines and browsers. One challenged provision had required the manufacturers to pre-install the Google search and Chrome browser apps as a condition of access to the Google Play Store, a must-have utility for Android users seeking third-party apps. The Commission was not persuaded by the fact that downloading the Bing app today is a lot easier than browser distribution in the 1990s. A second set of contracts pertains to Google's strong position as an intermediary between advertisers and ad inventory providers. Here, the concern is that Google might have used various contracts to weaken the competitive prospects of rival intermediaries.

Platforms can also exclude rivals without using contracts, by making design choices that favor their own related businesses over rivals. For example, starting in 2007, Google introduced specialized product search results alongside the traditional "10 blue links" on the search engine results page. These so-called "product universals" pointed to third-party merchants offering the product for sale. Critics allege that by promoting these product universals, Google undermined competing comparison-shopping search engines by pushing them lower in its results. At the same time, Google arguably improved its user experience by sending users more directly to product listings rather than indirectly to another search engine.

The overall competitive effect of a design change can be difficult to assess, particularly where a change constitutes a genuine product improvement yet also has a tendency to weaken rivals. One influential approach has been to ask: Does the profitability of the change depend on its tendency to

weaken rivals? A change having both positive and negative effects is unlikely to fail this test. This lax attitude toward product design choices reflects a deep-seated reluctance to condemn product improvements.

Moreover, even where an adverse effect is likely, remedies can pose a further barrier to action. Antitrust enforcers routinely block enforcement of a problematic contract. They are much more reluctant to wade into the complex details of product design, particularly where doing so requires ongoing supervision of the design.

Mergers and Acquisitions

A second antitrust problem arises when a platform acquires its rival. For example, imagine Microsoft had bought Netscape instead of acting to exclude it by contract. (In fact, Microsoft approached Netscape about buying or licensing Netscape's browser code.) The harm would be similar, but without the conflict or complaining victim.

Platform acquisitions that thwart competition are illegal. If Facebook

tried to buy Snap, the FTC would likely move to block the deal on concerns ranging from reduced privacy to higher ad prices and slower innovation. But some platform acquisitions are less clear cut. If Microsoft had bought Netscape, would the Justice Department have perceived a sufficiently clear competitive threat to challenge the deal?

Facebook's acquisition of Instagram presents just this issue. Instagram—today a major growth engine for Facebook—was in 2012 a fast-growing mobile photo-sharing app. Had Instagram stayed independent, it might have become a full-fledged competitor to Facebook. Acquisition targets sometimes become giant-killers—consider, for example, Yahoo's missed opportunity to buy Google. Instagram in particular had ample venture capital support, and commentators at the time recognized Instagram had found and exploited Facebook's struggles in mobile photo sharing. Facebook recognized Instagram as a rising threat, and indeed, a top company official reportedly wrote colleagues that the point of the deal was to eliminate a competitive threat.

Where a powerful platform such as Facebook is concerned, protecting competition is particularly important.² The public benefits of competition are unusually great, given that Facebook faces limited competition within the market and few plausible challengers in the near term. Meanwhile, Facebook has the capacity and incentive to suppress competition, even from relatively long-shot rivals.

One broad lesson of the Microsoft litigation is that antitrust law protects “nascent, albeit unproven competitors ... particularly in industries marked by rapid technological advance and frequent paradigm shifts.”^a Netscape was a nascent competitor; so was Instagram. As applied to acquisitions, protecting platform competition requires a willingness to block a deal even when we are unsure what will happen otherwise. It also sometimes means revisiting a consummated deal after the fact. For example, suppose Facebook indeed bought In-

Antitrust enforcement is a powerful tool for blocking problematic contracts and mergers, less so for questions of platform design.

stagram in order to eliminate a competitive threat, but failed to share its true intention with the FTC at the time. An after-the-fact antitrust suit to challenge the acquisition might be appropriate, once its purpose and effects are better understood.

Copying

A third complaint is that platforms copy the products and marketing decisions of smaller rivals. For example, Amazon has a knack for spotting profitable products sold on amazon.com—Duracell batteries, say—and introducing its own version. Antitrust's first response is likely to be, so what? Private-label goods are a familiar sight in supermarkets. The copyist's strategy does not depend on weakening the copied firm. Often, copying *supports* competition and lower prices. Thus, copying is ordinarily of little interest to antitrust.

Platform copying can be more complex than the supermarket example, due to the platform's participation in multiple related businesses and its informational advantage at the center of a web of sellers and customers. For example, Amazon runs a popular Marketplace service for third-party sellers. If Amazon observes a Marketplace seller is having success selling a particular toy, it might decide to sell the same toy itself. But here, too, copying tends to intensify competition.

Though not ordinarily an antitrust issue, copying can raise important policy questions. In principle, platform appropriation might be so rapid, systematic, and effective as to discourage certain innovations. A Marketplace seller might not invest in marketing new products, lest Amazon

appropriate the benefits. If Google repackages content from the *New York Times*, while keeping a sizable fraction of the associated advertising revenues, that might harm the incentive to gather news. When Facebook copies Snap's Stories feature, a sort of temporary slideshow, this might give pause to the next would-be Snap. The problem is compounded if the copying is one-way, with the platform always the pirate, never the pirated.

Platform copying becomes an antitrust issue when combined with platform deception, such as manipulation of search results. Suppose an Amazon customer is led to believe the top search result represents the best deal, when actually it delivers the greatest profit for Amazon. The FTC has not faced a case exactly like this. However, it has challenged deception as a competitive harm on multiple occasions. And outside the antitrust sphere, the FTC has repeatedly reminded search engines of their obligation to clearly distinguish ads from “natural” search results. Manipulation of product discovery, if proven, would be a problematic distortion of the competitive process.

Conclusion

This brief discussion of platform competition issues supports several conclusions. Antitrust enforcement is a powerful tool for blocking problematic contracts and mergers, less so for questions of platform design. That does not mean that design choices, if established to harm competition, should escape scrutiny. Instead, new rules or regulators may be called for. Beyond antitrust, copying may raise important questions of innovation policy. However, the crucial underlying empirical question—to what extent is innovation really being suppressed—remains unanswered. **□**

References

1. Hemphill, C.S. Disruptive incumbents: Platform competition in an age of machine learning. *Columbia Law Review* 119, 1973 (2019).
2. Hemphill, C.S. and Wu, T. Nascent competitors. *University of Pennsylvania Law Review* (forthcoming 2020).

C. Scott Hemphill (hemphill@nyu.edu) is Moses H. Grossman Professor of Law at New York University, New York City, NY, USA.

^a *United States v. Microsoft Corp.*, 253 F.3d 34, 79 (D.C. Cir. 2001) (en banc) (per curiam).

► Carl Landwehr, Column Editor

Privacy and Security

Secure Development Tools and Techniques Need More Research That Will Increase Their Impact and Effectiveness in Practice

Secure development is an important and pressing problem.

WRITING CODE THAT is secure, and provides security without vulnerabilities, is a critical challenge to cybersecurity. Writing code without vulnerabilities has long been at least as difficult as writing code without bugs. While there are many other potential sources of security exposures in software, developing code without known classes of vulnerabilities has always seemed like a tractable goal. It relies on human developers using tools, techniques, and processes to produce software that does not have particular known types of defects.

One of the most effective approaches—research into programming languages and tools—has yielded technologies that are shown to resist categories of vulnerabilities, largely by not allowing for them. Memory safe languages that manage memory allocation and deallocation, instead of requiring the programmer to do so, make it impossible for developers to create buffer overflow vulnerabilities and some other types of exposures, from missing array bounds checks,



null pointer use, and data leakage via memory reuse. Thread-safe languages can address exposures where race conditions can be used to subvert security-related checks in the program.

Within the software development community, groups and organizations with a mission to develop software securely have incorporated tools and techniques into their software

development life cycles to include a secure development life cycle. Early high-assurance software adopted formal methods to specify the security properties of the system, and code review to use humans to find such flaws at the coding level.² Microsoft created its Security Development Lifecycle adding root cause analysis, security education, threat modeling, specific secure coding requirements, and security testing that included penetration and fuzz testing. Practices tend to be adopted based on business need, perceived security impact, and fit with established or evolving development practices.

Research that impacts what works and what could work for secure development is needed. Current research seems to play an unfortunately limited role in creating, proposing, evaluating, and proving tools, techniques, and processes that are used in practice for secure development. In particular, research is rarely brought to bear directly on tools and techniques as they are used, in the context they are used. We need more research into the effectiveness and results of secure development tools, techniques, and processes. That research can be judged on its impact on how software development works in practice. Properties of research influence how likely it is to have that impact.

Rigor in research scientific experimentation calls for a number of process requirements, including a statement of the hypothesis being tested, controlling the variables of the experiment to ensure the experiment actually tests the hypothesis, and analyzing experimental data and outcomes to mathematically prove the hypothesis (or disprove the null hypothesis). While these processes can form the basis of important foundational research in secure development, they often avoid the messy realities involved in bringing a technique into practice, precisely because those messy realities complicate experimental design.

Negative research results that fail to prove a secure development technique increases security, while important to the research field, are not likely to impact secure development in practice. An early lesson as a security developer in a large technology

company was that telling developers not to do something was almost always ineffectual, if it was not paired with the alternative that they could use to achieve the goal of the deprecated practice. “Don’t roll your own crypto” has to come with the crypto library that should be used. Additionally, finding a tool or technique experimentally ineffective in producing security does not prove it is ineffective outside of the controlled experiment, in the larger, messier, more diverse context of software development.

What are some of the things being done in research that are hopeful for practical transfer into secure development? Two current trends in security research provide some hope for secure development. One is that secure development has emerged as a topic in security research conferences, covering topics such as evaluating developers’ ability to use crypto securely and appropriately, evaluating tools to help developers avoid introducing vulnerabilities, and measuring developers’ ability to code security-relevant functionality.

The other hopeful trend is artifact evaluation. A lot of software development builds on existing software, using frameworks, libraries, and open source. Offering an artifact used to establish and validate a research idea reduces the barriers to transfer of that idea into software development. Making code available through open source, with license terms friendly to reuse, can increase its potential for use. Some research incentives are shifting to encourage artifact submission as part of the research paper submission and publication process, at

We need more research into the effectiveness and results of secure development tools, techniques, and processes.

security conferences such as USENIX Security and ACSAC.

Generalizing secure development research beyond the experiment is a challenge. A challenge of experimental research studies on secure development is the extent to which the results can generalize beyond the participants and context of the study. The challenges with increasing the similarities between an experimental study and secure development contexts may argue for an experimental approach closer to observational astronomy, medical case studies, or even public health than controlled laboratory physics experiments.

One of the aspects that complicates the design of a study evaluating a secure development process is the place of security in development tasks. As early usable security research called out, security is often not the primary goal of the user. Many of the human-centered empirical evaluation methods in use by research fit best for evaluating tools and methods in the context of explicit primary goals. In the secure development area, one of the aspects studied is avoiding the creation of vulnerabilities while coding, which is an implicit secondary goal. Prompting a coder to explicitly consider security has been shown to impact their behavior while writing code for the research study. Thus, studies that prompt the developer that way may not transfer to development contexts where coders are not told every hour to consider security for the code they are about to write, and we might guess that in the real-world developers would quickly tune out such messaging. However, remaining silent on the need for security provides less security prompting than occurs in organizations with a secure development process.

One exciting type of study balancing these concerns is the use of “Build It, Break It, Fix It” (BIBIFI) competitions³ as a different type of research context to study secure development. In BIBIFI, teams compete over several weeks to build software according to a spec, gaining points for functionality and performance. Then they compete to break each other’s software, causing the vulnerable teams to lose points. The context provides more control than a research field study,

but more ecological validity than a smaller-scope lab study. The resulting performance of each team, in terms of points, coding, and testing, can be analyzed for insights into vulnerabilities in a context that considers vulnerability-free code as only one part of the overall task. The contest may be part of curriculum requirements. Both that assignment and competition can act as motivators to keep participants engaged better and longer than most research studies.

Another complexity of research studies of secure development processes is effective recruitment of appropriate demographics. The expertise and skills of the participants potentially impact everything from what can be studied to what the limitations are on transferring the resulting findings to other contexts. Development expertise can be approximated by aspects such as years of experience in development, languages used, types of products, and types of organizations worked in. How to contextualize or measure security expertise of any particular developers, and in the general population of developers, remains an open question. Some research is emerging comparing the impact of demographic variables on the results of security task studies.

What more should be done? On the research side, there should be an explicit acknowledgment of the topic of research into the security results of secure development processes. The security research community should explicitly recognize that part of our responsibility as security researchers is to foster the full spectrum of research into better security: foundational research, practical research, and the transition of research into use (both successful and unsuccessful). A workshop venue for papers on security research and the challenges of tech transfer would be a solid step in identifying community and early work in the area.

Perhaps the largest barrier to such research is researcher access to secure development processes and their results. This requires cooperation with developers and development organizations. While each individual organization would profit from knowledge that would enable them to get the best results from their secure development process expenditure, getting there

Cross-community collaboration between researchers and development organizations is key to making progress.

would require a range of developers and organizations to cooperate with research, with no clear short-term upside. For inspiration on overcoming that, we look to near miss programs in aviation, which contribute to the safety of general aviation. One of these systems, the Aviation Safety Reporting System (ASRS) is comprised of confidential reporting, expert analysis by former pilots and air traffic controllers, publication of anonymized data, and rewards for those submitting reports. The rewards are that regulators are required to treat submission as “evidence of constructive engagement,” and will reduce penalties on that basis. The ASRS is operated by NASA, a respected scientific agency, so reports are not sent to a regulator, such as the FAA.


There are proposals¹ for a near miss database for cyber. Mirroring the structure of ASRS, a scientific agency or FFRDC would collect confidential reports, analyze them, and publish lessons. Regulatory agencies would commit to giving consideration to companies who have programs that candidly report near misses. A near miss in cyber is a place where some controls function and others do not. So a spam filter might not stop an email message containing a phishing URL, and the click on the URL might be caught by a safe-browsing list or firewall. Being able to quantify the “misses” experienced in the field is in some ways analogous to public health data, and could make a case for various types of investigations in SDP. Voluntary and rewarded near miss reporting should encounter far less industry opposition (after all, it is voluntary). Questions of scope could be examined over time by looking at the variance

between the voluntary responses. Because near misses give us data about both successes and failures, they represent a rich vein that we do not yet mine.

With such a capability, researchers could delve into the causes of near misses, and consider if the components of a SDP relate to important root causes. Root causes might be important for many reasons. They might be the most common problems, might be problems for which compensating controls are expensive, difficult, or ineffective. Researchers might argue, and have evidence for, other criteria.

Conclusion

Getting at the ability for researchers to evaluate secure development practices in context is a difficult problem, but critical for evaluating the ecological validity of practices in the wide variety of software development contexts that exist. Cross-community collaboration between researchers and development organizations is key to making progress.

The quality of software, including but not limited to security, is important to society as we become increasingly dependent on those qualities. How software development processes influence the qualities of software is thus an important societal question, worthy of study. As we improve the empirical evaluation of secure development processes as a result of these collaborations, we will benefit from a broad and deep approach to expanding our scientific inquiries. 

References

1. Bair, J. et al. That was close: Reward reporting of cybersecurity near misses. *Colo. Tech. LJ* 16 (2017), 327; http://ctlj.colorado.edu/?page_id=796#tabs-796-0-5
2. Lipner, S., Jaeger, T., and Zurko, M.E. Lessons from VAX/SVS for high-assurance VM systems. *IEEE Security and Privacy* 10, 6 (June 2012), 26–35; <https://dl.acm.org/citation.cfm?id=2420631.2420857>
3. Votipka, D. et al. Understanding security mistakes developers make: Qualitative analysis from Build It, Break It, Fix It. To appear in *Proceedings of the 29th USENIX Security Symposium (USENIX) Security* 20, 2020.

Adam Shostack (adam@shostack.org) is President of Shostack & Associates, a consultancy in Seattle, WA, USA.

Mary Ellen Zurko (mez@alum.mit.edu) is a member of the Technical Staff, MIT Lincoln Laboratory, Lexington, MA, USA.

The authors' views expressed here are not necessarily those of their employers.

Copyright held by authors.

▶ Mark Guzdial, Column Editor

Education

A Vision of K–12 Computer Science Education for 2030

Exploring goals, perspectives, and challenges.

WITH THE INCREASED prevalence of U.S. states including computer science as a required subject in K–8 education (and as an elective in 9–12), in the next decade, nearly every child in the U.S. will be taking CS classes. The rapid integration of CS into the current education system has challenged states, districts, and teacher preparation programs to revamp their current efforts considerably. As this is a relatively new innovation and challenge, it provides us with a unique opportunity to consider our agenda: What is the goal of CS education? In the K–12 context, CS is often synonymous with coding—in fact, to many educators, CS is only coding. We suggest the goal of CS K–12 education should be for K–12 students to understand CS beyond simply learning to code. The students of the next decade will become the workers, creators, policymakers, and innovators of the 2040s and beyond. Preparing these future innovators requires a reframing of CS education that deeply considers what kinds of citizens we are trying to develop. For example, what are the ethical considerations around computer science and technology use? Addressing these questions requires curricula that is more student-, community-, and equity-centered.



Making CS matter for all students: CS needs to be more than just coding. While we agree that coding is a critical part of CS education, it cannot be the sole focus. Several large-scale studies have shown that a failure to connect CS education to the lives of students, particularly young women and underrepresented minorities, is causing them to abandon CS as a career path, as it is not something for “people like them.”¹

Knowing how to code is a critical factor in women and underrepresented students succeeding and persisting in post-secondary CS education.⁵ However, we need to ensure they have relevant experiences at the K–12 levels first. One key aspect to providing successful K–12 CS experiences is recognizing that the current one-size-fits-all CS curricula being implemented, often does not actually appeal to all students. The prob-

lems students want to address in rural Illinois are likely far different than their counterparts in Chicago, even though they are separated by only a few miles. By adapting CS curricula to empower students to meaningfully connect CS to their lives, we will be more likely to show them why learning CS can, and should, matter to them.

Moving CS from the classroom to the world. The past decade has seen remarkable change due to the advent of the smartphone, dramatically changing how we connect and relate to the world around us. Over the next decade, computing will radically change again. As most current kindergarteners are expected to graduate high school in 2032, how will computing change the world around them? How do we prepare them for careers and lives with technologies we can only partially anticipate? We need to prepare them to be agile, and CS can help us do that.

These technologies increasingly offer the potential to enable what students build and create in their classrooms to be directly connected to their lived lives (for example, through smartphones and the Internet of Thing (IoT)). This in turn, allows students to see how CS operates in the world as a means to solve real-world problems. However, for students to see themselves as capable of creating these solutions, we need to develop educational initiatives that support students' development of their critical consciousness and computational identities—the idea that they are empowered citizens who can create help shape the world they live in, and why they should do so.⁴

Preparing students to solve new problems with CS. While newer ideas around CS like data science, artificial intelligence (AI), machine learning, deep learning, distributed and quantum computing are all at different levels of adoption, each will change what it means to apply computing to our daily lives. Many of these new applications will require less programming and more understanding of how to use them and their associated challenges.³

Few K–12 students will need to develop their own machine learning algorithms, but most will need to understand the critical issues around training and evaluating the algorithms, and bias and equity. For AI, students must tackle the

We need to develop educational initiatives that support students' development of their critical consciousness and computational identities.

already growing issues such as who would be held responsible for self-driving car accidents and how can historical biases affect models of future events? For the IoT, students will need to understand the importance of data privacy, sharing, and distributed systems that require critical understandings of the ecosystems that surround the code they write. For data science, students will need to understand how to collect and convert data in ways that can be useful to computer models and how to critically evaluate the results. Perhaps most important is preparing students to be able to recognize bias and identify ethical ways to use and integrate computers into their daily lives. By tackling these issues as part of a comprehensive CS curricula, we entrench CS as something that connects to the issues students increasingly face and empowers them to be engaged problem solvers.

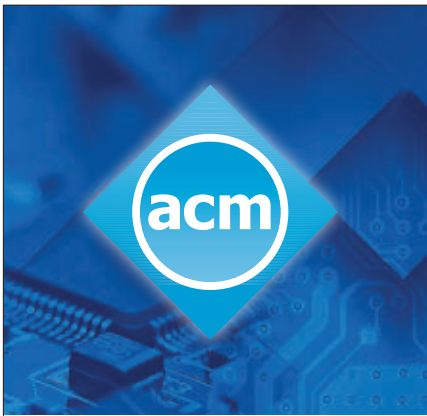
Using CS as an interdisciplinary tool. Over the next decade, we anticipate CS education will be integrated into a range of other disciplines (for example, physics, economics, civics, environmental sciences) rather than just a stand-alone subject. Integrating CS (not just computational thinking) into these subjects, offers the potential for making subjects more personally meaningful. Students can develop their own sensors to capture and report on environmental factors in their communities for science class, or design their own data models to visualize economic conditions at critical moments in history (for example, the great depression). We are only scratching the surface of researching the potential of these interdisciplinary approaches to CS. Success-

fully integrating CS into multiple domains will likely require individual subjects to develop introductory CS lessons specific to their domain that extend what they learn in their CS classes.

Framing computational literacy as a fundamental literacy. We believe K–12 students' understanding of the role of computing in non-computing disciplines will be critical for future work success. By advocating for a vision of CS that extends beyond just programming, we get closer to the vision of computing as a fundamental literacy that can be used by all. In their future jobs, it is unlikely that all students in the next decade will be programmers or data scientists. However, the ability to understand how computing (and by extension coding) can be applied to their jobs and lives enables them to be truly computationally literate. Managers, designers, and team members will need to understand what types of computing solutions are possible and “speak the language of programming” to be a part of effective work teams. A basic understanding of computing and computational thinking must be a foundational part of all students' education to prepare them to contribute solutions to the problems of their futures.

Challenges associated with student-centered CS. This vision of the next decade of CS education is not without significant challenges. We must develop new forms of formative and summative CS assessment that move beyond assessing students' ability to write and debug code, toward those that capture changes students' attitudes toward computing, their computational identities, their sense of digital empowerment, and their ability to engage in computational design thinking processes.⁴ While many researchers are already looking at this from a research perspective, these outcomes are equally important in teachers' classroom evaluations. Additionally, if CS education is truly student centered, how do we develop assessments that properly attend to the individual, is contextualized to their specific learning, and allows for the necessary variation that comes out of this kind of learning?

If we are serious about creating relevant K–12 CS curricula that provides students opportunities to connect to their lives and communities, we need



Advertise with ACM!

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



Request a media kit with specifications and pricing:

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



We are often not placing enough attention on what the broader outcomes of CS should be, and what kind of citizens we are preparing the next decade of students to be.

to find ways for these communities to buy in to these initiatives. A failure to do so will reinforce ideas that computing designs *at* rather than *for* these communities. We urge the field to examine ways to use student- and human-centered pedagogical approaches that will enable students to select problems that are personally meaningful and use computing to help solve them.

We want to challenge those involved in CS education to create and use more open-ended, student-centered, and real-world K–12 CS curricula. This will require more intensive curriculum planning and increased teacher program development. Many states are implementing new CS teaching certification and licensure programs that focus primarily on programming and linear approaches to CS education. We are at a critical juncture in supporting the next decade of CS teachers. We must ensure these student-, community-, and ethics-focused pedagogies are deeply integrated into the licensure process, or risk duplicating the challenges faced by math and sciences, which has largely failed to show students how these disciplines apply to their lives and futures.²

We also must ensure there is recognition of the importance of these non-programming focused CS pedagogies throughout the K–12 pipeline. Superintendents, principals, policymakers, and parents all need to recognize the value in moving beyond a pure programming-focused approach to CS. A failure to get them to understand the importance of this approach will result in small pockets of innovation within a

sea of traditional CS practices—the fate of many innovative learning initiatives.

Next steps in CS education. We are at a pivotal moment in the future of CS education. In the rush to implement large-scale CS education, we are often not placing enough attention on what the broader outcomes of CS should be, and what kind of citizens are we preparing the next decade of students to be. We need to look at how we prepare these students to not only succeed, but also thrive in a landscape in which they may not need to program, but will need to deeply understand how computing (and programming) can influence and shape their work and lives.

For a successful next decade, CS education needs us to immediately confront digital inequities, not only in access to technology, but in whose problems are being solved by that technology. If we create educational initiatives that reinforce existing power dynamics, we will lose out on the unique contexts, issues, and needs of underrepresented students. This will result in failed efforts to truly create *CS for all*. NCWIT's newest messaging platform conveys the importance of empowering every student: “The idea you don’t have is the voice you haven’t heard.” With the broad sweeping changes to both K–12 CS curriculum and teacher training/licensures, we must ensure they incorporate digital equity and student-centered pedagogies to prepare all students to computationally solve the problems of their futures. **□**

References

1. Carlone, H.B. and Johnson, A. Understanding the science experiences of successful women of color: Science identity as an analytic lens. *Journal of Research in Science Teaching* 44, 8 (Aug. 2007), 1187–1218.
2. Chapman, A. and Feldman, A. Cultivation of science identity through authentic science in an urban high school classroom. *Cultural Studies of Science Education* 12, 2 (Feb. 2017), 469–491.
3. Shapiro, B. and Tissenbaum, M. New programming paradigms. *Cambridge Handbook of Computing Education Research*. Cambridge University Press, 2019, 606–636.
4. Tissenbaum, M., Sheldon, J., and Abelson, H. From computational thinking to computational action. *Commun. ACM* 62, 3 (Mar. 2019), 34–36.
5. Weston, T.J., Dubow, W.M., and Kaminsky, A. Predicting women’s persistence in computer science- and technology-related majors from high school to college. *ACM Transactions on Computing Education (TOCE)* 20, 1 (Jan. 2019), 1–16.

Mike Tissenbaum (miketissenbaum@gmail.com) is an assistant professor at the University of Illinois at Urbana Champaign, Illinois, USA.

Anne Ottenbreit-Leftwich (left@indiana.edu) an associate professor at Indiana University–Bloomington, IN, USA.

Copyright held by author.

Viewpoint

Computers Do Not Make Art, People Do

The continually evolving relationship between artistic technologies and artists.

WE LIVE IN an age of amazing new visual art created with artificial intelligence (AI) technology. The recent wave began with neural stylization apps and the trippy, evocative DeepDream. Many fine artists now work with neural network algorithms, creating high-profile works appearing in major venues.¹

Together with these new developments comes the hype: technologists who claim that their algorithms are artists and journalists who suggest that computers are creating art on their very own. These discussions usually betray a lack of understanding about art, about AI, or both.

This column explains why today's technologies do not create art; they are tools for artists. This is not a fringe viewpoint; it reflects mainstream understanding of both art and computer science. There is a long tradition of computer-driven procedural art, and all of it is ultimately made by people, even when they use software branded as AI. It is possible this could change someday, that our software gets so good that we assign it authorship of its own works. As I will explain, I believe this is unlikely.

The First Artistic Machine

Art has a long history of evolving in response to new technologies. In the past century, many of these technologies have led to debates and misunder-



Generative artwork by Tom White, made with aesthetic considerations and to fool image recognition algorithms. Most current algorithms identify this image as “trombone.”

standings about the role of the artist. Tools that seemed at first to make artists irrelevant actually gave them new expressive opportunities.

The invention of photography is, ar-

guably, the most important invention in the modern history of art, but it did not seem so positive at first. The lessons from this story repeat periodically when new artistic technologies arise.

Important steps in fine art technology. (a) Oil paint technology, (b) Early artistic photography, mimicking conventional painting, (c) Rule-based, procedural computer-generated art, (d) Evolutionary computer-generated art, (e) Current neural-network GAN art.



(a)



(b)



(c)



(d)



(e)

Artworks:

- (a) "The Emperor Napoleon in His Study at the Tuileries," by Jacques-Louis David, 1812.
- (b) "Fading Away" by Henry Peach Robinson, 1858.
- (c) "Homage á Paul Klee" by Frieder Nake, computer-generated graphic, 1965, China ink on paper.
- (d) "Electric Sheep" by Scott Draves, individual "sheep" by BrothaLewis. CC BY 3.0 US. <https://creativecommons.org/licenses/by/3.0/us/>
- (e) "AmI Dali Yet?" by Helena Sarin, 2018. (collection of Jeremy Howard).

In 1839, Louis-Jacques-Mandé Daguerre described the first practical photographic technique.⁹ Public interest was immediate and widespread; practical applications of photography were immediately evident, and new developments and uses quickly followed in the subsequent decades. The first major impact of photography was on portraiture, where it soon became dominant, largely replacing portrait painting, silhouette cutouts, and printmaking.

The status of photography as art was more controversial. In one of the early presentations of the Daguerreotype in 1839, classical painter Paul Delaroche was quoted as saying "From today, painting is dead."

To understand why, it is helpful to

imagine the time before photography, when realistic images of the world could only be produced by skilled artists. Today we are so swamped with images that it is hard to imagine just how special and unique it must have felt to see a well-executed, realistic painting. The technical skills of realism were inseparable from other creative skills in making images. This changed when photography mechanized the task of producing images of the real world.

Some critics, like the poet Charles Baudelaire, saw photography as a corrupt and dangerous attack on true artistic genius. On the other side, photographers developed and advocated for their own art form. They argued that the artist's ability to control image creation to express their vision made it an art form

in itself. Some early photographers, like Henry Peach Robinson, tried to elevate their work by mimicking the themes and composition techniques of painting. Subsequent generations explored the unique qualities of photography. By 1910, after years of work by photographers and critics, mainstream museums and other institutions began to recognize photography as an art form in its own right. Today, major photographers are included in the art history canon, and photography continues to be a significant art form.

When the camera was first invented, it looked like a machine that automated the creation of art. It required no skill and would destroy high-quality art. What actually happened? A new art form was invented, with its own unique qualities. Portraiture technologies became largely obsolete and portrait artists did need to learn the new technology. Image-making became more available to hobbyists; nowadays, anyone with a mobile phone can take a picture.

Unexpectedly, photography had a profoundly positive effect on painting. As cameras became cheaper, lighter, and easier to use, realistic photographs became commonplace by the end of the 19th century. If photorealism could be reduced to a mechanical process, then what is the artist's role? Many artists of the era, including Whistler, Munch, and Van Gogh, wrote that true art was not about reproducing reality, because that was "just photography"⁹—true art was about something beyond realism. It seems the Modern Art movement came about because of photography. Rather than killing painting, photography spurred decades of innovation in painting.

The Artist Is the Mastermind

In the Modern Art era that followed, the definition of art broadened significantly. Marcel Duchamp's "Fountain" was a landmark: he (or possibly a friend of his) found a urinal, flipped it over, signed it, and submitted it as a sculpture in 1917. Later artists, like Robert Smithson and Yves Klein, removed the requirement that the artwork be an object at all. This, and other work like it, ultimately set a precedent that an artist creates an artwork simply by naming it as such.

Throughout the arts, the artist is the mastermind behind the work, no matter how much or how little they contributed to its actual execution (see the figure images in this column). Since auteur theory, the director is credited as author of a film, despite the recognized contributions of many other artists and artisans. An architect is credited for their buildings, even when large teams of artisans, technicians, and builders all contribute. A DJ who samples and remixes sounds is the artist behind a new track. (Copyright law treats ownership differently, but that is an entirely separate topic.)

The same applies to software art, of which there is now a long tradition. The first software art was created in the 1960s, by artists including A. Michael Noll, Georg Nees, and Frieder Nake.⁷ Harold Cohen's AARON software generates paintings based on a set of randomized procedural rules.¹ Karl Sims¹⁰ and Scott Draves¹⁴ evolutionary artworks involve automatic creation of images, virtual creatures, and procedural animations from user feedback. Many artists have created lovely abstract interactive artworks that respond to the viewer's movements, including the work of Daniel Rozin, Camille Utterback, and Golan Levin. New Media arts programs typically have entire courses of study around software and interactive art.

In each of these cases, the practice of creating artistic software—or making art with software—involves iteration, experimentation, and refinement. The artist does not simply write a program and let it go. The artist writes a piece of software and then tinkers and refines the algorithms over a long period of time, continually judging and evaluating the imagery produced by the system. The final results we see exhibited come from many hundreds of hours of hard work from the artist.

In popular art, computer animation is widely accepted as an art form. But progress in computer animation frequently renews the old fears. For example, in the 1980s, Ed Catmull and Alvy Ray Smith made many trips to Disney to convince them to fund computer graphics research.⁸ The animators always resisted, afraid that computers would take their jobs, and management was too conservative to take the

risk. So Catmull and Smith ultimately founded Pixar instead, and created an environment where artists and technologists could develop the art form together. Now, computer-animated films employ hundreds of animators and other kinds of artists; high-quality computer animation is both dependent on human creativity and is extremely labor-intensive.

Even though software, crowdworkers, and/or artisans may have executed on the artist's instructions, the artist is the person who instigated and coordinated the work. None of these software systems is called “an artist.”

Current AI-Based Artwork

All of this leads to the inevitable conclusion that AI-based artwork is still artwork made by a human. Our current “AI” software is just software,² despite the fancy branding, and there is a long precedent of art made with software.

Earlier computer-generated artwork, such as AARON and Sims' evolved virtual creatures, employed classic AI methods, that is, handcoded rules and numerical optimization. The same goes for procedural image stylization algorithms.

The recent neural stylization algorithms use data-fitting as one step, but they are each ultimately the result of a human writing software, and then experimenting with and improving the software algorithms, parameters, and training data until they get results they like.

We can get a window into some of today's Generative Adversarial Network (GAN) artists' experimentation via their Twitter feeds. Artists like Mario Klingemann (@quasimondo) and Helena Sarin (@glagolista) regularly tweet about their experiments using the latest image transformation software to create art. They experiment with code, parameters, and datasets, tinkering with the tools until they get great results. Assigning authorship of their art to software is perverse, dismissing the value of the artists' own hard work and creativity.

In a few cases, computer scientists have claimed their software is (possibly) the artist. In each case, they are writing the code, running an optimization, tuning the algorithm and the optimization, and selecting the favored results—just as in all previous artworks,

Calendar of Events

At press time, scheduled conferences were significantly impacted by COVID-19, often requiring cancellation of the event. Please check with the hosting SIG regarding the conferences listed here.

April

Apr. 27–30

EuroSys '20: 15th EuroSys Conference 2020, Heraklion, Greece, Co-sponsored: ACM/SIG, Contact: Markatos Evangelos, Email: markatos@ics.forth.gr

May

May 5–7

I3D '20: Symposium on Interactive 3D Graphics and Games, San Francisco, CA, USA Sponsored: ACM/SIG, Contact: Eric A Haines, Email: erichaines@gmail.com

May 11–13

CF '20: Computing Frontiers Conference, Catania, Italy, Sponsored: ACM/SIG, Contact: Maurizio Palesi, Email: Maurizio.palesi@dieci.unict.it

May 23–29

ICSE '20: 42nd International Conference on Software Engineering, Seoul, Korea, Co-sponsored: ACM/SIG, Contact: Gregg E. Rothermel, Email: gerother@ncsu.edu

May 23–24

ICSSP '20: International Conference on Software and System Processes, Seoul, Korea, Co-sponsored: ACM/SIG, Contact: Liguang Huang, Email: lghuang@smu.edu

May 25–26

TechDebt '20: International Conference on Technical Debt, Seoul, Korea, Co-sponsored: ACM/SIG, Contact: Clemente Izurieta, Email: clemente.izurieta@montana.edu

like those listed here. Perhaps these authors claim this out of ignorance of the history of computer-generated art, or perhaps based on a desire to be provocative. If these software packages are artists, then so is the font rendering and page layout package that renders your Word or PowerPoint documents, and so is a game engine that beautifully renders procedural 3D environments.

People sometimes talk about the possibility of “collaborating” with an AI. We do collaborate with our tools in that they can automate tasks, produce unexpected results, and push us in ways that we would not have otherwise considered. But “collaboration” also often implies co-ownership and joint high-level decision-making. In this sense of the word, one does not collaborate with software any more than one collaborates with watercolor paints or Photoshop.

But the lessons from history are ultimately positive: new technology gives new tools to artists, who in turn invent new visual styles and new artistic media. The infusion of new technology into art is one of the main ways that art remains vital. New AI technology will continue to invigorate art and empower artists in the future.

Will An AI Ever Be An Artist?

The definition of art changes over time.¹¹ In the distant future, will we start to accept software systems as themselves artists, independent of their creators?

To think about this question, one may first ask why we create art. Evolutionary theory provides a compelling answer, saying that artmaking is the product of our evolution.⁵ According to this theory, art emerged as a way for our Pleistocene ancestors to strengthen their social ties and social status. For example, art can serve as gifts, as fitness signals for mating, and as displays of status and tribal affiliation. In each role, the fundamental purpose of art is to affect peoples’ relationships with each other, where the relationships are themselves important. We have many behaviors for establishing and maintaining personal and group relationships, like gifts, competition, conversation, games, and romance, and making art is one of these behaviors.

Hence, I hypothesize that art can only be created by people (or other

independent actors) capable of these kinds of social relationships. In contrast, while we can get emotionally attached to our computers and other possessions, we feel no real empathy for their emotions, no ethical duty toward them, and no need to demonstrate our feelings toward them. This means computers cannot be credited as artists until they have some kind of personhood, just as people do not give gifts to their coffeemakers or marry their cars. If there is ever such a thing as human-level AI, with thoughts, feelings, and moral status comparable to ours, then it would be able to create art. But “human-level AI” is pure science fiction right now, and we are nowhere near achieving it. We do seem open to the idea that animals (such as our pets) could create art—it is just that, while they have social relationships, they lack interest in creative objects or performance.

Perhaps someday we will believe that social, shallow AIs are artists. There are many anecdotes of people being fooled into thinking that chatbots are real people, including the recent plague of Twitter bots. But, once the veil is lifted, it is clear that these chatbots do not exhibit real intelligence, and we feel cheated if we had thought they were “real.” Maybe agents like a Siri or Alexa will someday be treated like junior members of the family, who answers questions, raises children, and makes artwork. This seems unlikely.

Some people say that only humans can create art because art requires intent, or, it must express something, such as an emotion. However, it would be easy to build artificial systems that do this. For example, in many artworks, the intent can be summarized by a short sentence: “depict a specific beautiful landscape,” or “convey an experience of the horrors of war.” It would be straightforward to build systems that generate “intents” like these, and then create artwork from them. But most people would probably agree that this system is not “an artist;” it is still a human-engineered system, and the authorship really belongs to the author and/or user of the system.

Even if we could someday develop an algorithm that autonomously produces an endless stream of artworks that are original, beautiful, surpris-

ing, provocative, expressive, and culturally relevant, as long as we understand the software as just executing the instructions it has been given, it will continue to be a dumb machine, and not an artist.

Conclusion

I do not believe any software system in our current understanding could be called an “artist.” Art is a social activity, and our “AI” software is still just software, mechanically following the instructions we give it.

Moreover, calling a software system an artist is irresponsible, because it is misleading: it could make people think that the software has human-like intelligence, autonomy, and emotions.

Art maintains its vitality through continual innovation, and technology is one of the main engines of that innovation. We are lucky to be alive at a time when artists can explore ever-more powerful tools. Today, through GitHub and Twitter, there is an extremely rapid interplay between machine learning researchers and artists; it seems like, every day, we see tinkerers and artists tweeting new creative experiments with the latest neural networks. Seeing an artist create something wonderful with new technology is thrilling, because each piece is a step the evolution of new forms of art. As artists’ tools, AI software will surely transform the way we think about art in thrilling and unpredictable ways. □

References

1. Bailey, J. The tools of generative art, from Flash to neural networks. *Art in America* (Jan 8, 2020).
2. Brooks, R. The seven deadly sins of AI predictions. *Technology Review* (Oct. 6, 2017).
3. Cohen, H. The further exploits of AARON, painter. *Stanford Humanities Review* 4, 2 (1995).
4. Draves, S. The electric sheep screen-saver: A case study in aesthetic evolution. In *Proceedings of EvoWorkshops*. (2005).
5. Dutton, D. *The Art Instinct: Beauty, Pleasure, and Human Evolution*. Bloomsbury Press, 2009.
6. Hertzmann, A. Can computers create art? *Arts* 7, 2 (2018).
7. Nake, F. Computer art: A personal recollection. In *Proceedings of CGC*. ACM, New York, NY, USA, (2005), 54–62; <https://doi.org/10.1145/1056224.1056234>
8. Paik, K. *To Infinity and Beyond! The Story of Pixar Animation Studios*. Chronicle Books, 2007.
9. Scharf, A. *Art and Photography*. Penguin, 1968.
10. Sims, K. Artificial evolution for computer graphics. In *Proceedings of SIGGRAPH*. ACM, New York, NY, USA, (1991), 319–328. <https://doi.org/10.1145/122718.122752>
11. Warburton, N. *The Art Question*. Routledge, 2002.

Aaron Hertzmann (hertzman@dgp.toronto.edu) is a Principal Scientist at Adobe Research in San Francisco, CA, USA, and an ACM Fellow.

Copyright held by author.

Viewpoint

When Technology Goes Awry

On engineers' obligation to tame their creations.

I BEGIN MY BOOK, *Digital Minimalism*,² by quoting an essay by the journalist Andrew Sullivan. “An endless bombardment of news and gossip and images has rendered us manic information addicts,” he wrote. “It broke me. It might break you, too.”⁵

When I talk to people about their relationship with their digital devices, many report experiences that echo Sullivan. Many people look at screens constantly; not just for work, but while at home, with their children, while in bed, or even in their bathrooms. Some users jump from Hacker News, to email, then over to Twitter to share a take no one requested, then back to email. At best, it is needlessly distracting; at worse, it might break some of you, too.

So I wrote a book that attempted to untangle the forces that pushed many of people toward this place of diminished autonomy, and then provide ideas about how we might reduce this bombardment of our attention. Given the *Communications* readership, however, it seems to me the details of *what* is in this book are less important than the question of *why* someone like me—a computer science professor who primarily studies the theory of distributed systems—is tackling these comparably woolier, public-facing issues in the first place. My answer not only provides insight into my specific path, but more importantly underscores a critical need for engineers in general to get more involved in resolving the increasingly thorny issues gen-



erated at the intersection of technology and culture.

To better articulate my call to action for engineers, some brief historical background will prove useful. As an area of inquiry, the philosophy of technology has a long pedigree that stretches from Aristotle’s *Physics*, through Francis Bacon’s *New Atlantis*, to, much more recently, Kevin Kelly’s *What Technology Wants*. The field seems to have coalesced into a more consistent area of inquiry around the late industrial revolution,

and in the century and half since, it has produced two competing approaches for understanding the role of tools in human affairs: technological determinism and technological instrumentalism. Roughly speaking, the former philosophy believes the features and properties of a given technology can drive human behavior and culture in directions that are often unplanned and unforeseen, while the latter believes tools are neutral, and what matters in understanding their impact is the cultural

context and motivations of the people that develop and use them for specific purposes.

The determinist philosophy received a lot of attention in the second half of the 20th century when a loosely organized group of philosophers, historians, and critics, including Lewis Mumford, Jacques Ellul, Lynn White Jr., William Ogburn, and Neil Postman were publishing big-think idea books about ways in which technology sparks surprising and powerful consequences. A famous example of this thinking is the historian Lynn White Jr.'s 1962 classic, *Medieval Technology & Social Change*,⁶ which argues that the arrival of the horse stirrup in medieval Europe accidentally sparked the rise of feudalism. (In case you are wondering how this connection works, it goes something like this: The stirrup made it possible to put armored knights on horses, as they kept knights in their saddle after absorbing the blows of lance strikes; this new class of armored shock troops provided an immense warfare advantage that once introduced was necessary to maintain power, but they were also expensive and complicated to support; the division of land into feudal fiefdoms, each supporting a small number of knights, proved to be an efficient economic configuration to solve this problem.)

In recent years, however, the pendulum of power in the formal study of philosophy of technology, especially within academia, has swung in favor of the technological instrumentalists. This shift is well captured by the rise to prominence of a theory known as the Social Construction of Technology (often abbreviated as SCOT), an instrumentalist philosophy that understands technologies' development and impact primarily from the perspective of the underlying social forces influencing the technologists. One of the most well-cited examples of this approach—in some sense, the constructivist response to Lynn White's armored knights standing in their stirrups—is a careful study by Trevor Pinch and Wiebe Bijker of the shifting cultural trends that helped the safety bicycle become more popular than the big-wheeled penny farthing that preceded

it.⁴ To the SCOT theorist, technology is not so interesting on its own: like the physicist studying iron filings displaced by a magnet, technology should mainly be observed to help highlight the underlying power dynamics these theorists believe matter more. (For a more nuanced take on these duals frameworks, I point the interested reader toward Doug Hill's excellent 2016 survey book, *Not So Fast: Thinking Twice About Technology*.)¹

I am reviewing this split because I have come to believe the shift toward instrumentalism, though intellectually interesting and often quite illuminating, is ill-suited on its own to tackle some of the more pressing issues we face in our current moment of rapid technological innovation. As I will describe, to prevent the onslaught of technology (especially in computing) from diminishing our lives and culture, we should be willing in some circumstances to deploy a more determinist view of these tools—a move that will require engineers to get involved.

Engineers are instinctually skeptical of technological determinism. The idea of our tools acting autonomously from human intention seems suspiciously mystical, and given our love of optimization, there is great appeal in the instrumental notion that if a tool is impacting you negatively, it is because you are using it wrong. Based on my close study of these issues, however, I think we of-

We should be willing in some circumstances to deploy a more deterministic view of these tools—a move that will require engineers to get involved.

ten hubristically overestimate our degree of control when dealing with certain innovations.

To provide an illustrative example that I have written about before, consider the introduction of an internal email system to IBM in the early 1980s.^a Because computing power was expensive, the team tasked with introducing this service first conducted a study to determine how much employees were already communicating through memos and phone calls, with the idea being the bulk of this messaging would be moved to email once it was introduced. Based on their findings, they provisioned a \$10 million mainframe that should have had no trouble handling the expected load. Almost immediately, the mainframe overloaded.

“Thus—in a mere week or so—was gained and blown the potential productivity gain of email,” joked Adrian Stone, an engineer who was part of the original IBM email team.^b When I interviewed Stone about these events, he told me the mere presence of this new tool radically changed how people worked. Not only did they send more messages than they ever had before, they began cc'ing messages to many more people. Within days, the workflow at IBM had transformed from one of occasional messaging to constant communication.

The technological instrumentalist would try to find a social force that explains this change—some group, for example, that realized they could gain advantage by pushing for more frequent communication—but Stone remembers this shift in behavior as much more haphazard, and more recent research backs up this assessment. In her careful study of interactions in the Boston Consultant Group, for example, Harvard Business School professor Leslie Perlow documented a process she calls the “cycle of responsiveness,” in which a culture of non-stop emailing emerged from an unstable feedback loop, in

a I previously cited this example here: C. Newport, “A Modest Proposal: Eliminate Email,” *Harvard Business Review Online*, February 18, 2016; <https://bit.ly/33w0Uus>

b See Adrian Stone's response, posted June 27, 2014, in the following Quora thread: <https://bit.ly/399Naac>

which fast responses engendered even faster responses, until the consultants blindly converged to a set of organizational norms for email that *no one* liked.³ When Perlow introduced new policies that tamed these norms, employee satisfaction and productivity, as measured by surveys, increased significantly.

This is a useful case study of technological determinism: the properties of low-friction digital communication destabilized the social dynamics surrounding communication, leading to a new style of work—ceaseless electronic chatter—that no one planned, and that ended up making employees less happy and less productive. When Perlow interviewed the consultants she was studying, they assumed that *someone* must have intentionally introduced the culture of hyper-connectivity under which they suffered, but as with the IBM example, no one had. The technology, in some sense, made the decision for them.

To provide a more grandiose example consider the impact of the social media “like” button. Facebook was the first major social media platform to add this so-called feature. As the engineers who developed it reported in contemporaneous blog posts, their goal was to solve a simple technical problem. Many Facebook posts were attracting large numbers of comments that offered generic positive approval: “nice!,” “great!,” “beautiful!.” The engineers worried these short comments were displacing more interesting longer comments, so the “like” button was conceived as a way for users to demonstrate basic approval without needing to leave a comment.

This simple optimization, however, generated an unexpected and profound effect: people began looking at their accounts *much more* than ever before.^c The “like” button, it turns out, transformed the social media ex-

The “like” button added something new: an incoming stream of social approval indicators.

perience. In their original incarnation, these platforms provided an easy way for you to post things about yourself and occasionally check on things your friends posted. The “like” button added something new: an incoming stream of social approval indicators. Now you had a reason to keep tapping on the Facebook app throughout the day: to check in on this stream of evidence that other people are thinking about you—a reward that’s significantly more appealing than simply catching up on your friends’ activities. To make matters worse from the perspective of the user’s attention, this stream of indicators is unpredictable: sometimes when you check you receive a lot of feedback, and sometimes you receive very little. As the behavioralists uncovered in their famed experiments of animals pressing levers to dispense food, this style of *intermittent* reinforcement fosters compulsion.

This small change help spark a massive transformation of not only the social media experience but our relationship with our smartphones. We used to check social media websites occasionally when bored and deployed our smartphones for specific uses, such as looking up directions or playing music while we walked across town (I am ignoring here the early business power users who were already addicted to email on their Blackberries at this point—a different phenomenon). In the post-“like” world, our phones became constant companions that we check incessantly throughout the day, craving the next hit of reward as we become conditioned to fear any downtime. Though I am obviously eliding

Distinguished Speakers Program

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

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

speakers.acm.org



Association for
Computing Machinery

^c For more on the ways in which the “like” button was developed and its consequences, I recommend the following two resources: Victor Luckerson, “The Rise of the Like Economy,” *The Ringer*, February 15, 2017, <https://bit.ly/33xL9Dy>; and Alter, Adam. *Irresistible: The Rise of Addictive Technology and the Business of Keeping Us Hooked*, Penguin Press, New York, 2017.

some other relevant details in this story,^d it is reasonable to claim that much like the horse stirrup accidentally sparking the rise of feudalism, a small tweak meant to improve the quality of social media comments significantly altered the daily routines of hundreds of millions of people.

We can now return to my proposal that engineers get more involved in our culture's ongoing struggle to react to technological change. In the examples here, tools that were introduced for narrow, often bland purposes—such as making memos more efficient or consolidating comments—ended up creating major impacts that caught many people off guard and did not necessarily serve their best interests. I call these impacts *complex side effects*, as they are often best understood through the lens of complex system theory: the interaction between humans and machines is complex, and seemingly small changes, like eliminating the friction in intra-office communication through the introduction of email, can create large and hard to predict shifts in the system's behavior. My examples focus on my narrow area of expertise in the study of technology and culture: network systems and their impact on personal and professional productivity. These side effects, however, are relevant to many different topics within this general space, such as AI and automation, data privacy, and algorithmic bias—all subjects where new tools have the potential to create unexpected consequences.

Complex side effects are not well handled by the current academic emphasis on technological instrumentalism. When we view these impacts through the lens of social construction, we are either reduced to the role of the detached observer, or face the daunting challenge of somehow re-engineering social dynamics, an effort that historically sways uneasily between condescension and authoritarianism.

^d I provide a more detailed accounting of this transformation in Chapter 1 of *Digital Minimalism*. In this richer account, the “like” button helped Facebook learn that economic value of transforming their service into a source of social approval indicators, after which, in more instrumentalist fashion, they invested heavily in optimizing this effect (a process called “attention engineering”).

I do not mean to disparage the contributions of existing social scientists thinking about technology and society.


When we instead adopt the perspective of technological determinism, these side effects are stripped of their implicative power, and can become yet another aspect of performance that needs to be measured and addressed as needed. It is here that engineers have a role to play. We are the ones who build these systems, and once deployed, we evaluate them on factors such as their efficiency and security. When shortcomings are revealed, we iterate, either trying to improve the system or propose a new approach. Complex side effects should be included in this iterative engineering process.

This applies to systems we directly help create. If you were an engineer on the IBM team that introduced internal email in the 1980s, the fact that your servers created wild and sudden swings in user behavior should have been just as much a concern as lagging performance or dropped packets. This approach also applies to systems created by others. The engineers who introduced the “like” button at Facebook would have had a difficult time trying to tame the excesses it instigated as those excesses turned out to be highly profitable to their employers, but there was nothing stopping engineers outside of Facebook from highlighting the negatives of this complex side effect and suggesting alternative ways to build these systems. (Indeed, this is what former Google engineer Tristan Harris did when he appeared on *60 Minutes* in 2017, held up a smartphone, and told Anderson Cooper: “this is a slot machine.”^e The

^e Tristan Harris, CBS “60 Minutes” interview with Anderson Cooper: <https://cbn.ws/2vzncip>

non-profit he subsequently co-founded, The Center for Humane Technology, proposes design principles that better respect user attention—see <https://humanetech.com/>).

I do not mean to disparage the contributions of existing social scientists thinking about technology and society. However, given the accelerating rate and increasing impact of technological change, and the antipathy toward technological determinism in the fields that traditionally study these issues, engineers need to join this conversation. Our systems often create powerful complex side effects that are independent of specific human intentions, and we are particularly well situated to rapidly notice and address them. Meticulously researched SCOT analyses are not sufficient by themselves to tame the consequences of the momentous technological innovations that define our current moment.

To return to where I began this Viewpoint, my colleagues and mentors have often wondered why I maintain “two careers” as a writer and engineer, but I no longer see it that way. Exploring complex side effects in my writing is as integral to my scientific obligation as proving theorems about these systems. To adapt the message Samuel Morse prophetically sent during his public introduction of the telegraph, engineers should keep asking, “What have we wrought?,” then add the crucial follow-up prompt: “And what should we do about it?” 

References

- Hill, D. *Not So Fast: Thinking Twice About Technology*. University of Georgia Press, Athens, GA, 2016.
- Newport, C. *Digital Minimalism: Choosing a Focused Life in a Noisy World*. Portfolio, New York, 2019.
- Perlow, L. *Sleeping with Your Smartphone: How to Break the 24/7 Habit and Change the Way You Work*. Harvard Business Review Press, Boston, MA, 2012.
- Pinch, T. and Bijker, W. The social construction of facts and artifacts: Or how the sociology of science and the sociology of technology might benefit each other. In W.E. Bijker, T. P. Hughes, and T. Pinch, Eds. *The Social Construction of Technological Systems: New Directions in the Sociology and History of Technology*. MIT Press, Cambridge, MA, 1987, 17–50.
- Sullivan, S. I used to be a human being. *New York* (Sept. 18, 2016); <https://nym.ag/2UjArw6>
- White Jr., L. *Medieval Technology & Social Change*. Oxford University Press, London, 1962.

Cal Newport (cnewport@cs.georgetown.edu) is Provost's Distinguished Professor in the Department of Computer Science at Georgetown University, Washington, D.C., USA.

Copyright held by author.

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

ACM Publications Finances

OVER THE PAST YEAR, ACM has developed and started promoting the ACM Open business model, a model we expect to lead to full open access for ACM publications. As part of these discussions with our institutional subscribers, we were asked for a clearer picture of the finances of ACM's publications program. We know that members have also often asked questions about both the costs and revenues associated with our publications. We are writing this article to share what we have found with ACM's membership.

Why this article? As far as we know, this is the first time ACM has reported this level of detailed and comprehensive cross-departmental financial information relating to publications revenues and expenses to our entire membership. ACM keeps detailed financial records as a non-profit, however, we have historically reported financials by department in our annual reports. Over the past year, it has become

increasingly important to identify all of the costs related to our publications, not just in our publications department, but across all ACM departments involved with and expenses related to ACM publications, such as information services and technology, marketing, finance, membership, legal, office space, travel, utilities, and management, and to report these expenses in a clear and transparent way for our membership and institutional Digital Library customers.

To build a more comprehensive view of publications finances, we worked together with ACM's finance staff, its COO, and with the leaders of the various units of ACM that overlap publishing. We should note that any financial reporting involves certain assumptions and decision-making. We outline some of the most important ones here:

- ▶ We chose to look at the 2019 calendar year rather than an ACM fiscal year (which starts July 1) because most of our revenue comes from institutional subscriptions (mostly from university libraries) that run on calendar years. Our reporting to these subscribers involves counts of publications on a calendar year basis, so we had those figures to work from.

- ▶ We chose to lump together “small items”—typically revenue and expense items under \$5000—so as to help us better focus on the big picture.

- ▶ We had to develop a cost base and rate allocation for ACM's indirect expenses (also known as overhead). This effort involved tallying all expenses not directly in the budgets of core business units (e.g., HR, space, utilities, marketing and communications, management, etc.) and allocating these appropriately across the core business units. This allocation was a best-effort estimate that will likely be refined over time.

- ▶ We chose not to allocate membership fee funds to publications; we know that it is common to consider a part of membership as a “subscription fee” (e.g., to *Communications of the ACM*), but ACM's membership dues are set independently of publication costs and have therefore been held level even as we've invested substantial new resources into *CACM*.

- ▶ While we have total revenue and

expense figures, we had to make a decision on the level of detail to which we wanted to break these down. We chose to focus our per-article analysis on the three core lines of publications (journals, magazines, and conference proceedings). ACM has many smaller lines of business (ACM Books, the International Conference Proceedings Series (ICPS), partnership publishing and hosting arrangements with various societies and other organizations, and many others), but we knew it would take substantial time to tease apart the costs for these programs (most of which are often handled by the same staff and systems as our core lines), and decided to aggregate these smaller lines together for reporting. Similarly, we did not allocate any costs to maintaining the archive of past articles, instead dividing those costs among the current year's published works (as this is the model our subscribers use, and will be the model that future open access models will use for financing publications).

- ▶ Where we have indicated expenses related to direct and overhead staff in terms of full time equivalents (FTEs), please note these expenses include benefits and staff related expenses, such as healthcare, insurance, retirement benefits, etc. and should not be interpreted as purely compensation.

- ▶ Finally, given the goal of understanding our costs and how they relate to a transition to the ACM Open model, we limited our per-article analyses to full-length articles (full research papers in conference and journals, full-sized articles in magazines) rather than allocate costs to poster abstracts, panel abstracts, job board postings, books, and other types of content in ACM's various publications.

To put it succinctly, the data below represent our best effort to get a picture of publications revenues and expenses. We recognize that the data are imperfect, in part because they were never collected and tagged with the intent of supporting these analyses. We expect that future years' reports will be increasingly robust as we are able to better allocate expenditures to their function at the time of expense.

ACM Financials At A Glance

As Table 1 shows, ACM had 2019 publications revenue of almost \$24 million and expenses of \$23.3 million, for a net surplus of just under \$700 thousand (all figures U.S. dollars). The lion's share of ACM's income comes from subscriptions; just over \$22 million from institutional subscriptions from 2,700 university libraries (which often subscribe through consortia that negotiate pricing together), 100 corporations, individual ACM member subscriptions, and individual publication subscriptions. ACM has approximately \$1.2 million in advertising revenue, primarily through the sale of classified advertising on ACM's Job Board and Magazines. There is some additional revenue associated

Table 1.
**ACM Publications Financials:
At A Glance**

Calendar Year 2019

Income	
Digital Library: Consortia, Corporates, & Govt Licenses	20,270,144
Digital Library: Articles Pay Per View	91,705
Institutional Membership Dues	313,376
Subscription Revenue (including SIGs); A la Carte Subscriptions	883,907
SIG Hardcopy Magazine Subscriptions (<i>Interactions/Inroads</i>)	119,892
Digital SIG Master Package	179,563
Advertising, including SIGs	1,240,972
ICPS Proceedings: Non-ACM Conference Publication Fees	362,883
Open Access Revenue (APCs)	177,400
All Other Publications Revenue: ACM Books, etc.	352,884
Total Income	23,992,725
Expenses	
Journals	3,966,813
Magazines	5,519,977
ACM Conference Proceedings	5,541,257
ICPS, Books, Hosted Content	215,535
Digital Library	5,098,668
Cost of Sales (Agents, etc.)	2,747,357
Publications Board	211,615
Total Expenses	23,301,223
Total ACM Publications Net	691,503

with book sales, with fees we collect for certain purposes (e.g., ICPS publication fees), etc. The \$177K of revenue listed from open access article processing charges is less than 25% of the total paid by authors; our anti-double dipping policy results in crediting most of our received APC revenue back to our subscribers.

We divide expenses into the key publication lines (journal, magazine, proceedings), other publications, the costs of operating and serving the ACM Digital Library, cost of sales, and the costs of the ACM Publications Board. We break down the four largest expenses in the next section; only the smaller ones are discussed here.

- ▶ **Cost of Sales**—ACM has a small staff (4 FTE) selling subscriptions to approximately 2800 universities and companies around the world. Rather than hire a large worldwide sales staff, we contract with various sales agents with expertise in selling scholarly publication subscriptions to libraries in different regions of the world.

- ▶ **ICPS, Books, Hosted Content**—We know that this figure is an underestimate of true costs. First, in 2019 we had been in a contract where expenses associated with ACM Books were covered by a partner prior to distributing revenue (that is no longer our business model). Second, as noted above, we have not yet adequately allocated staff time and share of resources to these activities. Expect this category to show an increased share of costs (and somewhat increased total costs) in future years.

- ▶ **Publications Board**—The volunteer management expense that includes meetings and projects in areas including: developing and reviewing new publication proposals; regular review of publications and their editors; editorial searches; handling accusations of plagiarism and other publications misconduct; technology and practice advances to promote data deposit, artifact review, reproducibility and replicability; efforts to increase the reviewer volunteer pool; and other projects designed to ensure the long-term health of publications. All positions are volunteer, but funds are expended on meetings, workshops, and contracts for services (e.g., conducting surveys).

ACM Financials: Deeper Dive

Table 2 shows cost breakdowns for the main expense categories; we review each of them here. We note that these expenses are those allocated specifically to individual categories of publication (including their share of ACM's overall indirect/overhead costs). Costs associated with the ACM Digital Library and subscription sales are broken out separately.

Journals. ACM published approximately 2,700 journal articles in approximately 59 journals last year with total journal-specific expenses of \$3.96M. The expenses break down into:

- ▶ **Submissions**—\$190K for subscriptions to submission and journal management software.

- ▶ **Production**—\$1.2M for composition and copyediting (about 75% of the total), plagiarism detection software, DOI and copyright registration, etc.

- ▶ **Printing and Distribution**—\$170K (some of this represents individual subscribers who still prefer paper journals; others are for archival copies in certain libraries or library systems).

- ▶ **Direct and Support Staff**—\$1M for 6.46 FTE including journal managers, acquisitions editors, production staff, and others who directly support ACM's journals. We should note that staff costs are not simply salaries, but the full cost of positions including medical benefits, social security taxes, retirement, and other benefit costs.

- ▶ **Overhead Staff and Expenses**—\$1.37M including rent, utilities, outside legal and accounting costs, and 4.87 FTE of in-house management, marketing, accounting, HR, legal, and other functions.

Magazines. ACM Publishes seven magazines, including *Communications of the ACM*, *Interactions*, *XRDS*, *InRoads*, *Ubiquity*, *eLearn*, and *Queue*, which together published 675 articles in 2019. The total cost of publishing these magazines is approximately \$5.5M, which is a significant expense in the context of ACM's overall publications budget. The expenses break down into:

- ▶ **Production**—\$1.28M includes a bit over \$500K for freelance articles (CACM) and contractors (*Queue*); over \$300K in graphic art and design; \$100K in contracted editorial support, and additional funds for soft-

ware, copyediting, website contractors, travel, and other publication support (plagiarism detection, DOI registration, etc.).

- ▶ **Printing & Distribution**—\$1.4M, most of which is associated with printing and mailing 60,000 copies of *Communications of the ACM* each month to over a hundred countries worldwide.

- ▶ **Direct and Support Staff**—\$1.3M for 8.65 FTE including magazine editorial managers, business staff, production staff, and others who directly support ACM's magazines.

- ▶ **Overhead Staff and Expenses**—\$1.37M including rent, utilities, outside legal and accounting costs, and 6.52 FTE of in-house management, marketing, accounting, HR, legal, and other functions.

Magazines are labor intensive, requiring significant hands-on editorial and graphic design work, as well as the real-world cost of printing and distributing four of these magazines that exist in both print and digital formats. At the same time, magazine content is by far the most widely read content ACM publishes, and the predominant venue for reaching our practitioner members. Based on DL download records, our seven magazines accounted for 33% of all downloads in 2019 (with our journal articles accounting for 11% and ACM conference papers for 49%; the rest are downloads of ICPS papers and other content). A slightly different analysis looks at usage. We have 9582 magazine article downloads in 2019 for each new article published (of course some of those downloads are from old articles, and some of this year's articles will be downloaded in the future). Comparable numbers for journal and conference articles are 778 and 703 respectively. And these figures arguably understate the impact of magazines given that they ignore the copies read in print.

We also note that the largest part of the cost of our magazine program is *Communications of the ACM*. Printing and mailing costs alone are approximately \$1.4M per year. At the same time, membership surveys have shown that receiving CACM, and in particular *receiving it in print* is important to our members.

Table 2.
ACM Publications Financials:
The Deeper Dive
 Calendar Year 2019

Income	
Publications & Advertising	23,872,833
Revenue from Other Sources	119,892
Total Income	23,992,725
Expenses	
Journals	
Submissions	190,567
Production	1,233,073
Printing & Distribution	170,668
Direct and Support Staff	1,002,078
Overhead Staff and Expenses	1,370,427
Journals Total	3,966,813
Magazines	
Production	1,279,533
Printing & Distribution	1,404,663
Direct and Support Staff	1,292,344
Overhead Staff and Expenses	1,543,437
Magazines Total	5,519,977
ACM Proceedings	
Production	4,775,040
Direct and Support Staff	295,890
Overhead Staff and Expenses	470,326
ACM Proceedings Total	5,541,257
Hosted Content	
Production	54,691
Direct and Support Staff	61,222
Overhead Staff and Expenses	99,622
Hosted Content Total	215,535
Digital Library	
Infrastructure	2,646,617
Value Added Services	520,240
Content Preservation	18,181
Direct and Support Staff	898,833
Overhead Staff and Expenses	1,014,797
Digital Library Total	5,098,668
Cost of Sales/Agents	2,747,357
Publications Board/Volunteers	211,615
Total Expenses	23,301,223
Total ACM Publications Net	691,503

Conference Proceedings. Conferences are one of ACM's largest activities with ~200 events each year; conference proceedings included 13,511 papers in 2019 plus many more abstracts, videos, and other materials. Unlike journals and magazines, the cost of conferences is mostly borne outside the publications budget by ACM's Special Interest Groups (SIGs). The ACM Council (ACM's Board of Directors) decided years ago to distribute part of ACM's DL subscription revenues to the SIGs to compensate them for the cost of producing the content that fills the DL, and to compensate them as well for the loss of revenue from subscription packages for conference proceedings. The publications expenses we report include this distribution, though we do not attempt to break down how SIGs actually spend that distribution. The expenses associated with conference proceedings break down into:

- **Production**—\$4.78M includes \$3.5 million distributed to the SIGs; just under \$600K for conference reviewing software systems (we support several of the most popular ones for our conferences); \$660K for services that compile and produce proceedings; and just under \$40K on plagiarism detection, DOI registration, and other publications support. We have invested heavily in improving tools to reduce the cost of proceedings production and expect to see that cost continue to decline in future years.

- **Direct & Support Staff**—\$296K for 2.26 FTE, primarily focused on production and rights management for proceedings.

- **Overhead Staff and Expenses**—\$470K including rent, utilities, outside legal and accounting costs, and 1.7 FTE of in-house management, marketing, accounting, HR, legal, and other functions. The low level of overhead staff represents avoiding duplication of overhead staff already covered as part of the SIG operations budget.

Digital Library. ACM's Digital Library is the vehicle through which ACM's published content is distributed worldwide, and also includes a set of services to ensure indexing, content preservation, etc. In 2019, we replaced our aging home-grown in-

frastructure (official cut-over was the start of 2020). These changes have been accompanied by other improvements for mobile access, accessibility by users with disabilities, search, and other features. The expenses associated with the DL break down into:

- **Infrastructure**—\$2.65M includes \$1.3M for the DL platform itself (partly ongoing, partly capital investment); \$730K for the data center and core database systems; \$217K for email distribution, commenting, and notification systems; \$176K for statistics and analytics; \$144K for the content delivery network, caching, and security; and various other smaller amounts for components that enable the core functionality.

- **Value Added Services**—\$520K includes a variety of metadata services (to regularize, link references, provide useful exports, etc.) totaling \$288K, plus profiling and a variety of other services targeted at libraries, external indexers, and individual users.

- **Content Preservation**—\$18K includes service contracts to keep DL content available in the event of disruptions to ACM or its servers. This is an anomalously low annual figure due to timing of payments and is more typically closer to \$35K.

- **Direct & Support Staff**—\$899K for 3.65 FTE, primarily technical staff developing and supporting DL operations.

- **Overhead Staff and Expenses**—\$1.01M including rent, utilities, outside legal and accounting costs, and 1.7 FTE of in-house management, marketing, accounting, HR, legal, and other functions.

Analysis: Per-Article Cost

Table 3 shows the per-article cost of publishing ACM content. For this purpose we include Digital Library costs and Cost of Sales divided evenly per article (including distributing these costs to the more than 9,000 articles from other sources, predominantly ICPS). We did not allocate Publications Board costs across the categories because we did not feel we yet had an accurate basis on which to do so. If allocated evenly, it would add \$8 per article. We believe, however, that we spend more than half our time and effort on journals which would

Table 3.
ACM Publications Financials: Article-Level Expenses

Calendar Year 2019

Expenses at the Article level	Magazines	Journals	ACM Proceedings	Digital Library and Cost of Sales
2019 Cost	\$5,519,977	\$3,966,813	\$5,541,257	\$7,846,025
# of articles/published works	675	2,693	13,511	26,224
Publishing cost per article	\$8,178	\$1,473	\$410	\$299
DL cost per article	\$299	\$299	\$299	
Total cost per article	\$8,477	\$1,772	\$709	

make the closer to \$4 for non-journal articles and \$40 per journal article. We make the following observations:

- As expected, magazine articles are by far the most expensive to produce and publish and conference proceedings articles are the cheapest (mostly due to lack of copy-editing and typesetting, and due to the lower level of staff support).

- These figures represent cost. It would take a separate analysis to accurately represent value. As noted earlier, magazine articles get substantially higher usage per article. Journal articles tend to be longer (and have more content per article). Proceedings articles overall represent the largest percentage of digital library downloads (though less than half). We hope to explore the cost/value relationship in a future article.

- The cost per proceedings article (\$709) is very close to ACM's open access article processing charge for ACM member conference papers. We recognize, however, that pricing should probably consider both costs and revenues associated with each article.

Looking Forward: What Changes Lie Ahead?

We foresee a few changes with expenses and substantial model changes associated with revenue. On the expense side, most are changes associated with our new Digital Library infrastructure, with the scale of production, and with expanded services and publication types:

- Looking at total costs and staffing in relation to ACM's total number of articles published, we find that ACM is an efficient publisher; our staffing levels per article are significantly

lower than those we know from peer societies and commercial publishers. We are publishing increasing numbers of articles in journals, in proceedings, and in our auxiliary products (including ICPS and hosted content). While costs do increase somewhat with volume, we have previously seen economies of scale with production increases and expect to do so into the future.

- The last two years and the present year reflect the costs of migration to a new commercial digital library platform. In addition to start-up costs, we are seeing significant transition costs associated with converting content to new formats, reviewing and fixing metadata, and other changes that will improve the quality and accessibility of the digital library. As this project winds down, we hope to see savings associated with a stable DL.

- One area where significant investment is needed is in preserving digital research artifacts (e.g., code and data) and presentation artifacts (e.g., slides, video presentations) alongside publications. We are exploring a variety of solutions and vendors but with the expectation that core DL costs will increase as we both increase the media content and the capacity to preserve and make useful code and data within the DL. These efforts are led by volunteer committees committed to better supporting our mission within reasonable costs.

- One additional area where we have been investing is in publications designed to bridge research and practice. These "*Research and Practice*" journals (the first two are *Digital Threats: Research and Practice* and *Digital Government: Research and Practice*) will have production costs


somewhere in between journals and magazines as we find the appropriate level of support to help authors bridge the two communities. We hope to see more such R&P publications as warranted.

Revenues. On the revenue side, we are preparing for much bigger changes. While this article is not focused on the ACM Open model per se (we will be writing more about that in the future), we wanted to share its key features:

- ACM Open shifts the basis for digital library subscription from readership-based to authorship-based. Today's subscriptions are largely based on the size of an institution and the number of computer science researchers, professionals, and students who may be accessing content. ACM Open changes that basis to the number of articles published by authors at that institution. Indeed, a key feature of the ACM Open model is that "subscribers" are not paying for their own access to the DL, but instead for the entire world to access the works they have published.

- We are working with high-publication universities right now (many have already committed) to lead the transition to this "Publish and Read" model of subscription. As the high-publication institutions sign up, we will be able to drop prices lower and lower for institutions with few publications, and eventually to flip the entire digital library to open access (where author institutions underwrite open access for all readers). Our goal is to make the flip within five years, but this will depend heavily on how quickly universities adopt ACM's new model.

- Financially, we've designed this model to produce the same revenue, though that revenue will come from a smaller number of authoring institutions. We have also been able to build that model in a manner that provides average per-article subscription fees well below our current individual APC rates.

We hope you have found this article informative. We are committed to providing such information annually and we welcome your feedback, including suggestions for information to include in future reports. Please send your feedback to <cacmfeedback@acm.org.> 

The use of post-incident artifacts in high-performing organizations.

BY J. PAUL REED

Beyond the ‘Fix-It’ Treadmill

OF ALL THE traits the technology industry is known for, self-reflectivity and historical introspection do not rank high on the list. As industry legend Alan Kay once famously quipped, “The lack of interest, the disdain for history is what makes computing not-quite-a-field.” It is therefore somewhat cognitively dissonant, if not fully ironic, that the past few years have seen renewed interest in the mechanics of retrospectives and how they fit into the daily practice of our craft.

Of course, retrospectives are not new, in software development at least. For more than 15 years capital-A Agile software development methods have been extolling the virtues of a scheduled, baked-in reflection period at the end of each development sprint. (Whether these actually occur in organizations practicing Agile remains an open question.) Those same 15 years have also seen a tectonic shift in the way software is delivered: the general industry trend has sharply moved from packaging up those bits and bytes

into boxes to be shipped to users to “operate” themselves toward deploying it on massive server installations that *we* are responsible for maintaining, operating the software we have developed *for* users.

This shift has made the practice of software operations, and thus the study of how to do it and do it well, of interest to industry practitioners and spectators alike. As a part of the practice of software operations, there is renewed examination into the role played by operational retrospectives—more commonly referred to in an industrial context as *postmortems*. In short, looking back at the past to improve the future has become front-of-mind for many companies, precisely because the cost of not doing so in the development phase of software can be nebulous to measure, but the cost of not doing so in software operations is very apparent: Service-impacting incidents can be (and often are) easily translated to eye-popping dollars of lost revenue or service-level agreement penalties.

Think back to the last incident post-mortem in which you participated (or if you have never had the opportunity to participate in one, take a moment and imagine what might occur there). It probably looks something like this: A few days after the incident, a group of people meet for an hour. (It’s always an hour.) The size of the group (and how many managers are present) is directly proportional to how *important*—code for *visible* or *costly*—the incident was. The discussion kicks off by going through the details of the incident, often starting with the specifics of exactly how costly or how visible the outage was. Next up, what “actually happened” during the incident is discussed: how it started, who did (or didn’t) do what, and perhaps how the teams interacted with each other to address the problem. Maybe this discussion is aided by a timeline compiled beforehand (or maybe this timeline is put together at the meeting); logs and other metrics might be presented.

Conversations might tend toward

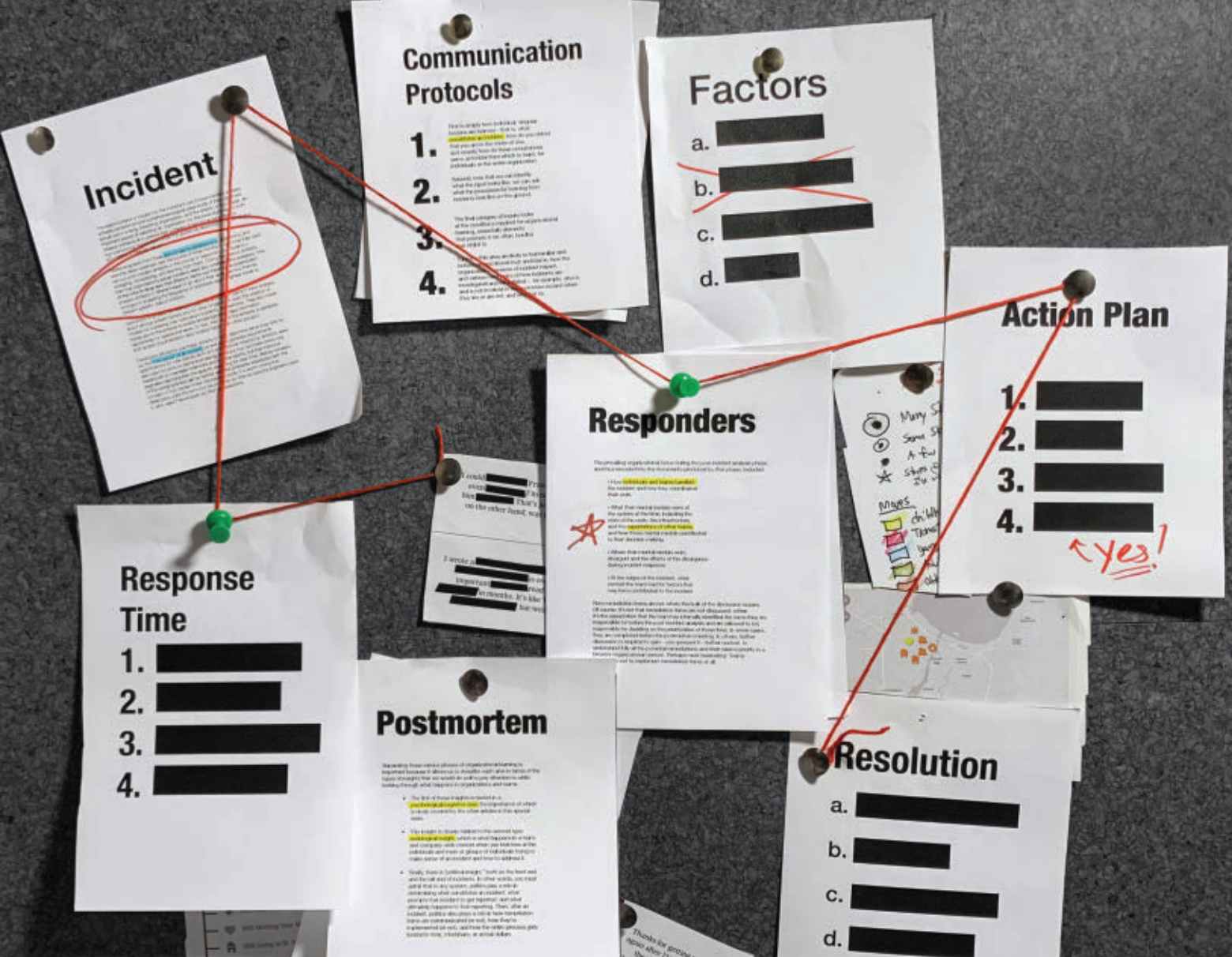


IMAGE BY ANDRÉJ BORYS ASSOCIATES

tense, and depending on a number of organizational factors, blame might be flung around the room. Or maybe it's someone's job to remind everyone they are all blameless. Maybe they believe it. Maybe whether or not they believe it depends on who is in the room. At some point, either to defuse a tumultuous situation, because someone notices there are 10 minutes remaining in the hour, or just to change a topic that no one wants to dive too deeply into, the discussion shifts to remediation items. The question is asked, "What are we doing to 100% *make sure this never happens again?*" The group brainstorms a list of remediation items. They range from low-cost, high-value items—"We already implemented *those*," one engineer proudly reports—to high-cost, questionably valuable items, which would otherwise be laughed at but in this specific setting everyone quietly nods their head

in agreement. Someone writes down those remediation items or takes a picture of the whiteboard where they are written. And the team disperses.

Maybe the suggested remediation items get entered into a ticket-tracking system. Maybe the company has a team whose sole purpose is to chase down these items and ensure each development and infrastructure team completes every item on that list in some (maybe discussed, maybe agreed upon, maybe neither) time frame. Maybe the team completes a large number of the items on the remediation list in the next two or three sprints; hopefully, the organization feels pretty good about that. Or maybe the importance of that work, once thought so critical, gets lost in the shuffle to meet the continuing onslaught of other goals, like a promised new feature or a big platform migration. Or maybe another critical incident—possibly related?—takes

up all the mindshare available for "do something" about the earlier incident.

If this pattern feels familiar, it should. Most operational retrospective and incident-analysis processes in technology companies look more or less like this. Some organizations are more experienced at the practice than others, some foster a "healthier" environment for it than others, and some value it more in the calculus of how they deliver software to and operate it for their customers. But the model, and its expected outputs, are generally the same, which leads to an important question: Are we missing anything in this prevalent rinse-and-repeat cycle of how the industry generally addresses incidents that could be helpful?

Put another way: As we experience incidents, work through them, and deal with their aftermath, if we set aside incident-specific, and therefore fundamentally static, remediation

items, both in technology and process, are we learning anything else that would be useful in addressing and responding to incidents? Can we describe that knowledge? And if so, how would we then make use of it to leverage past pain and improve future chances at success?

What Is Meant by “Learning”?

The topic of organizational learning has been of long-standing interest to the safety sciences, and researchers have been observing how it works in the context of industries from aviation to healthcare to maritime shipping for almost 90 years. Organizational learning has been deconstructed into three distinct categories of inquiry, following an evolution not dissimilar to the operation of Web-scale infrastructure and software:

► First is simply how individual, singular lessons are learned—that is, what constitutes an *incident*, how do you detect that you are in the midst of one, and exactly how do these occurrences serve as fodder from which to learn, for individuals or the entire organization.

► Second, now that we can identify what the input looks like, we can ask what the processes for learning from incidents look like on the ground. Much of the focus of organizational learning is on this specific facet, because it gets into the details of how real-world teams identify lessons to be learned and go about implementing them in their systems (or don’t).

► The final category of inquiry looks at the conditions required for organizational learning, essentially elements that promote it (or, often, hurdles that inhibit it). Topics in this area are likely to feel familiar and include organizational trust and blame, how the organization conceives of incident impact, and various mechanics of how incidents are investigated and remediated—for example, who is and is not involved in these processes (and when they are or are not, and why that is).

Types of Insights

Separating these various phases of organizational learning is important because it allows us to describe each area in terms of the types of insights that we would do well to pay attention to while

looking through what happens in organizations and teams.

► The first of these insights is rooted in a psychological/cognitive view, the importance of which has been covered in recent articles in the Practice section.

► This insight is closely related to the second type: sociological insight, which is what happens in a team- and company-wide context when you look less at the individuals and more at groups of individuals trying to make sense of an incident and how to address it.

► Finally, there is “political insight,” both on the front end and the tail end of incidents. In other words, you must admit that in any system, politics play a role in determining what constitutes an incident, what prompts that incident to get reported, and what ultimately happens to that reporting. Then, after an incident, politics also plays a role in how remediation items are communicated (or not), how they’re implemented (or not), and how the entire process gets funded in time, mindshare, or actual dollars. (Or not.)

These frameworks for investigating organizational learning have been applied to numerous industries. (A personal favorite delved into how Swedish rail workers learn from incidents, versus how the rail company *thinks* they learn, versus how the rail company *itself* “learns.”) Only in the past five or so years, however, have software operations been brought under the same lens, which necessarily drags software *development* along with it under the microscope (in an interesting twist, this is missing from other industries the safety sciences *have* studied).

A focus of these inquiries in the technology industry has been impactful or visible site/service outages, precisely because there are a set of practices that engineers and companies engage in during and after an event, but they are highly variable and not well described in the literature. (I aimed to change that in research conducted in late 2017 and recently published as a master’s thesis.)

Post-Incident Analysis Artifacts

Even the most nascent of incident postmortem processes produce *something* as an output. Common examples

include a postmortem report, remediation item tickets (relating to the software, the infrastructure, or both), updated documentation or runbooks, or distilled communications for other groups such as customers or executives. My deep dive into organizational learning in software development and operations organizations focuses specifically on these outputs, beginning with the various forms they take. All of the other details about the incident—the incident itself, what happened during the retrospective, and even how those artifacts came to be created—were considered to be a black box.

The study of these artifacts began at a broader, industry-wide level, by soliciting retrospective and postmortem templates via survey. These templates were then analyzed for structural elements in order to find commonalities (examples include an incident summary, basic timeline, and action items were the top three structures observed in postmortem templates), as well as the more unique structures. (Among the least common elements: document version/last modified date, a reminder to template users that root cause does not exist, and broad organizational findings.)

Perhaps the most notable finding from analyzing these various postmortem templates was that different template archetypes are used within the industry, each with a different focus and serving a different purpose. Three were apparent from the industry samples:

► *The Record-keeper*. This is the most common industry template and what most practitioners think of when they think of a postmortem report: It serves to provide additional prompts and “hints” to facilitate the running of post-incident analysis processes. These can include questions the organization wants asked during postmortem meetings or reminders to participants about the cultural ethos the organization values (blamelessness, for example) or otherwise wants highlighted to participants or facilitators during these processes.

► *The Facilitator*. While similar in structure to the record-keeper, the facilitator includes additional prompts and “hints” to facilitate the running of post-incident analysis processes. These can include questions the or-


organization wants asked during post-mortem meetings or reminders to participants about the cultural ethos of the organization values (blamelessness, for example) or otherwise wants highlighted to participants or facilitators during these processes.

► *The Signpost*. This template archetype can be aptly described as a pointer: It can provide either a reporting function, to be distributed to the larger organization for training or information purposes, or serve as a shorthand “itemized receipt,” pointing to additional data sources, usually various organizational systems of record, regarding the incident. In either case, it is marked by a lightweight treatment of the incident and the analysis outcomes and, as such, is typically used as a means of broad organizational communication regarding (especially impactful) incidents.


These three template archetypes do not preclude the existence of others; if more industry templates were collected and analyzed, other commonalities with enough uniquely identifiable elemental structures could define additional archetypes. In fact, as the practice of incident analysis evolves within the industry, so too should these archetypes.

Artifact Usage in the Production Environment

The second phase of inquiry into the industry’s use of post-incident analysis artifacts centered around a phenomenological case study of their observed actual use in a living, breathing organization, and the effects of that usage. An important aspect of selecting an organization for the case study was it both develop software *and* operate that software. It had to be considered a high-performing organization under the guidelines described in the 2016 and 2017 State of DevOps reports. Twelve engineers from three distinct teams (development, operations, and security) were observed over the course of three months to see how they used various post-incident artifacts in the course of responding to incidents—analyzing, remediating, and learning from them. During this period, artifacts from the organization’s actual incidents were also collected and analyzed.



Looking back at the past to improve the future has become front-of-mind for many companies, precisely because the cost of not doing so in the development phase of software can be nebulous to measure, but the cost of not doing so in software operations is very apparent.



One of the initial findings was that different teams use these same post-incident analysis artifacts in different ways to go about their work. Various themes emerged in analyzing the frequency of references each engineer made to different specific uses of artifacts. Operations engineers, for example, used the artifacts to perform trend analysis about various system factors and for other longer-term uses (the creation of models for bucketing their company’s incidents, for example). They also made heavy use of the artifacts to create knowledge base-type information repositories for operational work. (In fact, their use of the artifacts to generate and update documentation was notably higher than other groups.)

Developers tended to use these artifacts to help determine (what they refer to as) the “root cause” of an incident, as well as to generate requirements specifications for new feature work and architectural refactoring. Artifacts were also used to justify or clarify engineering decisions that had been previously made both to new team members and to other teams, but that individual engineers had forgotten the specific reasoning for over time. (Astute followers of the safety sciences will be familiar with the problems associated with the concept of root cause; those discussions aside, it is worth noting that developers used the term *root cause* twice as often as security engineers used it, who used it twice again as often as operations engineers, who seldom used it at all.)

Finally, security engineers used the artifacts more than other teams as one of the primary tools to drive their work. In the context of responding to security incidents, this makes intuitive sense: Security engineers need to respond to real-world threats they are seeing against production systems, so they use past incidents as a way of getting stronger signals indicating where they should plan their efforts and focus for the future. This includes guiding the generation and distribution of security-related documentation and driving internal security product roadmaps.

Taken together, these various uses add up to more than the sum of their parts. In today’s modern distributed systems, it is neither novel nor contro-

versial to point out that engineers work in complex systems. In the safety sciences, the term *complex socio-technical system* is usually used to point out that systems are an amalgam of not only code, compute, network, and storage, but also of people and teams. These people naturally have competing priorities, preferences, incentives, and goals, and they are often confronted with situations where they have to make critical decisions under extreme time and stress pressures, where all these factors consciously (and subconsciously) weigh into their decisions and actions.

One of the most important findings about the uses of these post-incident artifacts is that actors use them to help create and update mental maps of the emergent, complex socio-technical systems that they are responsible for engaging with. Because these Web-scale complex software and infrastructure systems constantly evolve, both in terms of technology and the teams behind that technology, individuals', teams', and even the organization's mental maps of how systems work can degrade over time. Anyone who has been frustrated at finding four architectural diagrams on the internal wiki, none of which is current, has experienced this. Incident artifacts provide, in effect, "patches" to these maps, allowing engineers and teams to update their above-the-line representations of the system and to discuss with each other where their cross-boundary (team or system) mental models were mismatched, inaccurate, or otherwise hampered their work.

This updating of the map of the organization's complex socio-technical systems was observed in a couple of ways. First, the artifacts provided evidence of a linkage between seemingly disparate, unconnected components of the wider system. There were many technical examples of this ("This microservice, in a particular failure mode, will call this other microservice that it used to rely on, but that dependency was thought to be removed; however, the dependency actually still exists, but only in this specific error condition"). But this effect also identified unknown and missing linkages *between people and teams* in the system. The most prominent example

was a team that turned out to be fielding a large number of security issues. They were located in a different state and focused on customer support, so they had no way to contact security engineers who could help them; because of this, a security incident occurred, and one of the updates to the *socio* part of the socio-technical system map was, "These people need to be introduced to those people, and an ongoing channel of communication needs to be established between them." Part of this included a need for training, which was eventually rolled out to a series of teams.

The second way this artifact usage was observed was as a way to identify hot spots within the socio-technical system. The old adage, "Where there's smoke, there's fire," is apt here, and post-incident analysis artifacts give engineers a sense of whether the smoke is from a small grease fire that set off the kitchen smoke detector for a few seconds, or if the smoke is visible from four blocks away and potentially more attention should be paid. Again, this provides input into mapping the *terrain* of the complex socio-technical system on which not only operations engineers are operating, but also developers are updating and changing, and security engineers are defending from external attack. This "smoke" can be indicative of (again, both technical and social) areas the organization has neglected and needs to invest more in, but it can also highlight entirely emergent areas that need to be addressed merely because the complex system has evolved in some unconceived way.

As an example of this effect, a security engineer disabled a particular set of options available to engineers via the use of a company-wide networking library; this improved the company's security posture. Some days later, a team went to deploy a new version of their microservice, and the deployment prompted an outage. After the issue was detected and remediated, one of the "smoky" issues the incident analysis raised, via distribution of the post-incident artifacts, was that the security team did not have any data on which versions of their library were in use across the company.

This was not neglect in terms of the organization focusing on other priori-

ties; rather, it was the system had evolved in terms of microservice- and software-dependency complexity to such a point that such data was now worth collecting and could highlight other potential problems, where a factor is teams using older versions that had been assumed to have been deprecated. This resulted in both a technical solution (starting to track library version use) *and* a social solution (that team now regularly engages other teams which the data shows are continuing to use old versions of the library to see why they have not migrated, if they can help them migrate, and if they need any new features before they do so).

A Move Toward Dynamic Remediation Items

Industry survey data indicates that 91% of respondents consider collection and recording of remediation items to be the core purpose of their post-incident analysis meetings and the artifacts produced from those meetings. Spending three months watching how a high-performing organization used their artifacts differently, however, sheds light on another approach: a focus on collecting, understanding, and sharing deeper, richer context about the technical state of a subsystem and the priorities, preferences, incentives, and constraints of the team responsible for operating and maintaining it. In this organization's environment, static lists of remediation items took a back seat to the search for and promulgation of this rich context.

The prevailing organizational focus during the post-incident analysis phase, and thus encoded into the documents produced by that phase, included:

- ▶ How individuals and teams handled the incident and how they coordinated their work.
- ▶ What their mental models were of the system at the time, including the state of the code, the infrastructure, and the expectations of other teams, and how those mental models contributed to their decision making.
- ▶ Where their mental models were divergent and the effects of this divergence during incident response.
- ▶ At the edges of the incident, what context the team had for factors that

may have contributed to the incident (that is, what other pressures, incentives, or circumstances the team faced with that may have made their local environment more prone to promoting factors identified as related to the incident).

Rote remediation items are not where the bulk of the discussion occurs. Of course, it's not that remediation items are not discussed; rather, it's the expectation that the team has internally identified the items they are responsible for *before* the post-incident analysis and are (allowed to be) responsible for deciding on the prioritization of those fixes. In some cases, they are completed before the post-mortem meeting. In others, further discussion is required to gain—you guessed it—further context, to understand fully all the potential remediations and their relative priority in a broader organizational context.

Perhaps most fascinating: Teams can decide *not* to implement remediation items at all. They may determine that taking a series of small outages that they believe can be remediated quickly enough is the right decision, given the other priorities the organization has tasked them with. This works in their organization because it is recognized that the development, operations, and security teams are closest to the systems they operate, and therefore are trusted to make the right decisions, given their local rationality and the context they have gathered from the other teams and systems around them. If that decision results in further outages that impact the rest of the organization or customers, then the exchange of context flows the other way between the involved teams—not a list of remediation items for a specific incident—and drives a more resilient, flexible resolution. One engineer aptly describes this model as “strategic accountability more than tactical accountability.”

This sharing of context has another benefit: It promotes the concept of blamelessness. The idea of the blameless postmortem has been bandied about in the industry for quite a while and has been met with some skepticism. With outages that have the potential to cost millions (or even pose an existential threat to the company—just

ask Knight Capital), it is entirely understandable to wonder if blamelessness can ever exist when the tempo is high and the consequences are very real. But because this search for and exchange of the context of the various subcomponents of the socio-technical system are valued higher than remediation items alone, in the aftermath of incidents the first step to understanding what happened is “share the context for why whatever happened, happened.” This is a marked departure from an approach that begins with the question, “What did you do?” and then seeks to hold a group referendum on whether or not that was the “correct” action to have taken.


Early Times, Exciting Times

The technology industry loves to hold aviation as the gold standard in incident and accident investigation, but it was not always that way. One of the biggest contributions to improved aviation safety was the introduction of crew resource management (CRM) in the 1980s. The insight that brought CRM to the fore of the aviation industry was not based on a set of remediation items from a specific accident, but rather from a holistic view of a series of accidents and looking for commonalities across companies, situations, equipment, and people. It was born not of a focus on piecemeal fixes but on a realization that improving how people go about doing their work, interacting with each other and their equipment, and effectively communicating about and responding to changes in their complex socio-technical environment is a place where some of the biggest discoveries of “hot spots” can be and where the biggest safety wins can emerge.

Given that humanity's study of the sociological factors in safety is almost a century old, the technology industry's post-incident analysis practices and how we create and use the artifacts those practices produce are all still in their infancy. So don't be surprised that many of these practices are so similar, that the cognitive and social models used to parse apart and understand incidents and outages are few and cemented in the operational ethos, and that the byproducts sought from post-incident analyses are far-and-away fo-

cused on remediation items and prevention (often with varying degrees of blame sprinkled in, whether we want to admit it or not).

But it doesn't have to stay this way. The industry is prime for a renaissance, but we must get past the notion the only value of post-incident analysis is in the list of static remediation items that so many of those processes are modeled, even optimized, to produce. Disavowing this notion requires becoming comfortable with moving away from the (admittedly comforting) assumption that if all the items on that list are implemented—we “100% remediate the incident!”—then it won't happen again.

Getting past that (admittedly tall) hurdle can create the cognitive and social space needed to explore all the various lessons an impactful, even painful, incident is trying to impart. Organizations can begin to approach solutions not from a list of tasks and bug fixes that try to address a situation that may never happen again, but instead from a place of moving toward broader solutions that address factors which tend to create situations where such incidents can occur. And this, ultimately, will push incident-analysis processes to evolve from such a laser-focus on the specific event that resulted in our Bad Day, toward what that Bad Day reveals about the true nature of our practices, processes, incentives, local contexts, the complex systems we operate every day, and perhaps most valuably: each other. 

Related articles on queue.acm.org

Postmortem Debugging in Dynamic Environments

David Pacheco

<https://queue.acm.org/detail.cfm?id=2039361>

The Network is Reliable

Peter Bailis, Kyle Kingsbury

<https://queue.acm.org/detail.cfm?id=2655736>

Why SRE Documents Matter

Shylaja Nukala and Vivek Rau

<https://queue.acm.org/detail.cfm?id=3283589>

J. Paul Reed is a senior applied resilience engineer on Netflix's CORE team in San Francisco, CA, where he focuses on incident analysis, systemic risk identification and mitigation, resilience engineering, and the human factors expressed in company's various socio-technical systems.

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

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

It's time to appreciate the human side of Internet-facing software systems.

BY DAVID D. WOODS AND JOHN ALLSPAW

Revealing the Critical Role of Human Performance in Software

FOUR ARTICLES, PUBLISHED across the March through May issues of *Communications*, highlight how people are the unique source of the adaptive capacity essential to incident response in modern Internet-facing software systems. While it's reasonable for software engineering and operations communities to focus on the intricacies of technology, there is not much attention given to the intricacies of how people do their work. Ultimately, it is human performance that makes modern business-critical systems robust and resilient.

As business-critical software systems become more successful, they necessarily increase in complexity.

Ironically, this complexity makes these systems inherently messy so that surprising incidents are part and parcel of the capability to provide services at larger scales and speeds.¹³ Studies in resilience engineering^{2,12} reveal that people produce resilient performance in messy systems by doing the cognitive work of anomaly response; coordinating joint activity during events that threaten service outages; and revising their models of how the system actually works and malfunctions using lessons learned from incidents. People's resilient performance compensates for the messiness of systems, despite constant change.

Thus, incidents that threaten service outages are endemic as an emergent side effect of the increasing complexity of the interdependencies required to provide valuable services at scale. Incidents will continue to present challenges that require resilient performance, regardless of past reliability statistics. It is the *cognitive work*, *coordination across roles*, and *adaptive capacity* of people that resolve anomalies as they threaten to grow into service outages.⁴ To be more specific: modern business-critical systems work as well as they do because of the adaptive capabilities of *people*; and without the *cognitive work* that people engage in with each other, all software systems eventually fail (some with increasingly catastrophic impact, given the criticality of the services they provide).¹

Human Performance and Software Engineering

Richard Cook connects human performance to software tooling through his insightful "Above the Line/Below the Line" diagram.⁵ Cook points out that discussions focused solely on the technology miss what is actually going on in the operations of Internet-facing applications. Figure 1 in Cook's article reveals the cognitive work and joint activity that go on above the line and places the technology and tooling for development and operations below the line. The "line" here is the line of represen-



tation. No one can directly inspect or influence the processes running below the line; all understanding and action are mediated through *representations*.

Below the line are the facilities engineers use to develop, change, update, and operate software that enables valuable services. This includes all the components needed to create the value that businesses provide to customers: the technology stack, code repositories, data sources, and a host of tools for testing, monitoring, deployment, and performance measurement, as well as the various ways of delivering these services.

The above-the-line area in the diagram includes the people who are engaged in keeping the system running and extending its functionality. They are the ones preparing to deploy new code, monitoring system activities, and re-architecting the system. These people ask questions such as:

What's it doing now? Why is it doing this? What's it going to do next? This cognitive work—observing, inferring, anticipating, planning, and intervening, among others—is done by interacting, not with the things themselves, but with *representations* of them. Interestingly, some representations (for example, dashboards) are designed by (and for) software engineers and other stakeholders.

Notice all the above-the-line actors have mental models of what is *below* the line. These models vary depending on people's roles and experience, as well as on their individual perspectives and knowledge. Notice that the actors' mental models are different. This is because there are general limits on the fidelity of models of complex, highly interconnected systems.¹¹ This is true of modern software systems and is demonstrated by studies of incident response; a common statement heard

during incidents or in the postmortem meetings afterward is, "I didn't know it worked that way."¹² Cook's concept and diagram reframes how Internet-facing systems function and is utilized by the other articles in the set.

Systems Are Messy

Systems are developed and operate with finite resources, and they function in a constantly changing environment. Plans, procedures, automation, and roles are inherently limited; they cannot encompass all the activities, events, and demands these systems encounter. Systems operate under multiple pressures and virtually always in degraded mode.¹³

The adaptive capacity of complex systems resides in people. It is *people* who adapt to meet the inevitable challenges, pressures, trade-offs, resource scarcity, and surprises that occur. A slang term from World War II captures

both the state of the system and the acceptance of the people who made things work: SNAFU (situation normal, all fouled up). With this term, soldiers were acknowledging that this is the usual status and their jobs were to make the flawed and balky parts work. If SNAFU is normal, then SNAFU catching is essential—resilient performance depends on the ability to adapt outside of standard plans, which inevitably break down.

However technologically facilitated, *SNAFU catching* is a fundamentally human capability that is essential for viability in a world of change and surprise. Some people in some roles provide the essential adaptive capacity for SNAFU catching, though the catching itself may be local, invisible to distant perspectives, or even conducted out of organizational view.⁹

Surprises in complex systems are inevitable. Resilience engineering enhances the *adaptive capacity* needed for response to surprises. A system with adaptive capacity is poised to adapt. It has some readiness to change how it currently works—its models, plans, processes, behaviors—when it confronts anomalies and surprises.¹¹ Adaptation is the potential to modify plans to continue to fit changing situations. NASA’s Mission Control Center in Houston is a positive case study for this capability, especially how Space Shuttle mission controllers developed skill at handling anomalies, expecting that the next anomaly they would experience was unlikely to match any of the ones from the past that they had practiced or experienced.¹⁰

IT-based companies exist in a presurized world where technology, competitors, and stakeholders change. Their success requires scaling and transforming infrastructure to accommodate increasing demand and build new products. These factors add complexity (for example, having to cope with incident response involving third-party software dependencies) and produce surprising anomalies.^{1,12} Knowing they will experience anomalies, IT-based companies, organizations, and governments need to be fluent at change and poised to adapt.¹³

Anomaly Response

Marisa Grayson describes her results

from examining a key function above the line by studying the cognitive work of anomaly response as people respond to an evolving incident.⁶ Grayson focuses on the general function of *hypothesis exploration* during anomaly response.¹⁴ Hypothesis exploration begins with recognition of an anomaly (that is, a difference between what is observed and the observer’s expectations). Those expectations are derived from the observer’s model of the system and the specific context of operations. Anomaly recognition in large, interconnected, and partially autonomous systems is particularly difficult. Sensemaking is challenging when monitoring a continuous flow of changing data about events that might be relevant. This is the norm for many Internet-facing business systems: Data streams are wide and fast flowing; normal variability is high; alert overload is common; operations and observations, as well as technology, are highly distributed. To make matters worse, the representations typically available require long chains of inference rather than supporting direct visualization of anomalous behaviors.

Grayson’s results show how practitioners generate, revise, and test potential explanations that could account for the unexpected findings. She developed a method to diagram and visualize hypothesis exploration based on the above-the-line/below-the-line framework.

Her charts reveal the typical flow of exploration where multiple hypotheses are generated to account for the anomalies, and the hypotheses in this set change over time. As response teams converge on an assessment of the situation, they frequently revise what are deemed candidate hypotheses and their relative confidence across the possibilities. In her study, Grayson found that sometimes a hypothesis that had been considered confirmed was overturned as new evidence came to the fore.

In hindsight, people focus on the answer that resolved the incident. The quality of anomaly response, however, is directly related to the ability to generate and consider a wide range of hypotheses and to revise hypotheses as the situation changes over time—for example, when interventions to resolve

problems end up producing additional unexpected behavior.

Controlling the Costs of Coordination in Joint Activity

Laura Maguire’s article⁸ expands the above-the-line frame by examining what coordination across multiple roles looks like when events threaten service outages, especially how people adapt to control the costs associated with coordinating the joint activities needed to resolve the situation. The value of coordination across roles and perspectives is well established as people wrestle with uncertainty and risk during an incident response. Handling anomalies in risky worlds such as space mission operation centers is one example.¹⁰ But studies of joint activity also reveal the costs of coordination can offset the benefits of involving multiple people and automation in situation management.⁷

However, this earlier research looked at anomaly response anchored in physical control rooms where responders were collocated in open workspaces. Internet-facing software systems are managed differently, as the norm is for responders to be physically distributed. People connect via ChatOps channels, unable to observe each other. The cognitive costs of coordination are greater for geographically distributed groups. Maguire’s article describes how this both enables and constrains joint activity. For example, growth has led to third-party software dependencies that require coordination across organization (and company) boundaries during anomaly response.

In her research, Maguire asks the question: What do practitioners do to control the costs of coordination as they carry out anomaly response under uncertainty, risk, and pressure? Her results are based on studying how software engineers experience these “costs” across a set of incident response cases. They highlight the shortcomings of traditional ways of coordinating roles and managing the costs of coordination (for example, incident commander, disciplined procedure-following (based on an incident command system), and efforts to use IT prosthetics such as bots). Maguire’s work reveals how people adapt when

the costs of coordination become larger. Understanding these adaptations can help in the design of effective tools, alter roles, and build organizational frameworks that enhance joint activity and reduce the costs of coordination during incident response.

Learning What Makes Incident Response Work

There is a significant gap between how we imagine incidents occur (and are resolved) and how they *actually* occur.³ J. Paul Reed considers how organizations learn to close this gap (see his article on p. 58 of this issue). He broadens the perspective to reveal the factors that affect how learning from incidents can be narrow and reactive or broad and proactive. Broad, proactive learning keeps pace with change, continually recharging the sources of adaptive capacity that lead to resilient performance.

Reed's research highlights an important but often invisible driver of work above the line—the ways people capture lasting memories of past incidents and how these memories are used by those not present or involved with handling the incidents at the time. How do people come to understand what happened? How do they share attributions about why it happened? Why do some incidents attract more organizational attention than others?

Organizations usually reserve limited resources to study events that have resulted in (or come close to) significant service degradation. Social, organizational, and regulatory factors constrain what learning is possible from such events. In contrast, *proactive* learning about resilient performance and adaptive capacities focuses on how cognitive work usually goes well despite all of the difficulties, limited resources, trade-offs, and surprises. The data and analyses in previous reports illustrate the potential insights to be gained from in-depth examination of the cognitive work of incident response.^{2,12}

In this piece, and for the others in the set, a theme repeats: incidents are opportunities to update and revise models of the ways organizations generate and sustain adaptive capacities to handle surprising challenges as IT sys-

tems grow and operate at new scales. If you take the view that systems are up, working, and successful because of the adaptive capacity that people have, then incidents can be reframed as ongoing opportunities to update and revise mental models as the organization/technology/infrastructure changes, grows, and scales.⁴

Conclusion

Together, the four articles provide a sketch of what is happening above the line of representation, especially during incident response. These activities are essential to building, fielding, and revising the modern information technology on which our society increasingly depends. Understanding how people detect anomalies, work together resolving incidents, and learn from those experiences is essential for having more resilient systems in the future.

The intimate relationship between human expertise and the technological components of modern systems defies linear decomposition. As Cook shows, there is really only one system here—how the system works and evolves depends on an awareness of how people's capacity to adapt is sometimes facilitated and at other times frustrated by the technology. The articles by Grayson, Maguire, and Reed demonstrate how looking at incidents through the lens of cognitive work, joint activity, and proactive learning provides new insights about how this human-technology system really works. Incidents are challenges that reveal the system doesn't work the way it has been imagined. The experience of the incident and post-incident inquiry offer learning opportunities highlighting where mental models need revision.

The articles go further, though. Together they highlight how everyone's mental models of Internet-facing software systems are in need of significant revision. Human cognitive, collaborative, and adaptive performance is central to software engineering and operations. As the scale and complexity of the software systems necessary to provide critical services continue to increase, what goes on above the line will remain central to all stories of growth, success, precariousness, and breakdown.

Understanding, supporting, and sustaining the capabilities above the line require all stakeholders to be able to continuously update and revise their models of how the system is messy and yet usually manages to work. When organizations value openness to continually reexamining how the system really works, they can follow the tangible paths these articles provide to learn how to learn from incidents. **C**

References

- Allspaw, J. 2016. Human factors and ergonomics practice in web engineering and operations: navigating a critical yet opaque sea of automation. *Human Factors and Ergonomics in Practice*. S. Shorrock and C. Williams, eds. CRC Press (Taylor & Francis) Boca Raton, FL, 2016, 313–322.
- Allspaw, J. Trade-offs under pressure: Heuristics and observations of teams resolving Internet service outages. Master's thesis, 2015. Lund University, Lund, Sweden.
- Allspaw, J. Incidents as we imagine them versus how they actually are. PagerDuty Summit 2018. YouTube; <https://www.youtube.com/watch?v=8DtzmVljiyQ>.
- Allspaw, J. and Cook, R.I. SRE cognitive work. *Seeking SRE: Conversations about Running Production Systems at Scale*. D. Blank-Edelman, ed. O'Reilly Media, 2018, 441–465.
- Cook, R.I. Above the line, below the line. *Comm. ACM* 63, 3 (Mar. 2020), 43–46.
- Grayson, M.R. Cognitive work of hypothesis exploration during anomaly response. *Comm. ACM* 63, 4 (Apr. 2020), 97–103.
- Klein, G., Feltoich, P.J., Bradshaw, J.M. and Woods, D.D. Common ground and coordination in joint activity. *Organizational Simulation*. W. Rouse and K. Boff, eds. Wiley, 2005, 139–184.
- Maguire, L.M.D. Managing the hidden costs of coordination. *Comm. ACM* 63, 4 (Apr. 2020), 90–96.
- Perry, S.J. and Wears, R.L. 2012. Underground adaptations: cases from health care. *Cognition, Technology & Work* 14, 3 (2012), 253–260; doi:10.1007/s10111-011-0207-2.
- Watts-Perotti, J. and Woods, D.D. Cooperative advocacy: a strategy for integrating diverse perspectives in anomaly response. *Computer Supported Cooperative Work: The Journal of Collaborative Computing* 18, 2 (2009), 175–98.
- Woods, D.D. Four concepts of resilience and the implications for resilience engineering. *Reliability Engineering and Systems Safety* 141 (2015), 5–9; doi:10.1016/j.res.2015.03.018.
- Woods D.D. Stella Report from the SNAFUcatchers Workshop on Coping with Complexity, 2017; <https://snafucatchers.github.io/>.
- Woods, D.D. Resilience is a verb. *IRGC Resource Guide on Resilience (vol. 2): Domains of Resilience for Complex Interconnected Systems*. B.D. Trump, M.-V. Florin, and I. Linkov, eds. EPFL International Risk Governance Center, Lausanne, Switzerland, 2018. https://www.researchgate.net/publication/329035477_Resilience_is_a_Verb.
- Woods, D.D. and Hollnagel, E. *Joint Cognitive Systems: Patterns in Cognitive Systems Engineering*. CRC Press (Taylor & Francis), Boca Raton, FL, 2006.

David D. Woods is Professor of Integrated Systems Engineering at Ohio State University. He developed resilience engineering on the dangers of brittle systems and the need to invest in sustaining sources of resilience beginning in 2000–2003 as part of the response to several NASA accidents.

John Allspaw has worked in software systems engineering and operations for over 20 years. Previously, he served as CTO of Etsy. His 2009 Velocity talk with Paul Hammond—10+ Deploys per Day: Dev and Ops Cooperation—helped start the DevOps movement.

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

DOI:10.1145/3376899

eBooks may have surveillance technologies embedded in them. Should we care?

BY STEPHEN B. WICKER AND DIPAYAN GHOSH

Reading in the Panopticon— Your Kindle May Be Spying on You, But You Can't Be Sure

The building *circular*—A cage, glazed—a glass lantern about the Size of *Ranelagh*—The prisoners in their cells, occupying the circumference—The officers in the centre. By *blinds* and other contrivances, the inspectors concealed from the observation of the prisoners: hence the sentiment of a sort of omnipresence—The whole circuit reviewable with little, or if necessary, without any, change of place. *One* station in the inspection part affording the most perfect view of every cell.

—Jeremy Bentham, 1798^a

JEREMY BENTHAM PROPOSED the panopticon as a new form of prison, one that would emphasize surveillance and rehabilitation as opposed to retribution and punishment. The panopticon was to have cells arranged in a circle about a centrally placed watchtower. The cells were lit from behind, outside the circle, so that guards

^a Proposal for a New and Less Expensive mode of Employing and Reforming Convicts (London, 1798); <http://bit.ly/35osJoG>



» key insights

- Amazon has patented eBook surveillance technology that may dramatically compromise anonymous reading.
- Any data collected by eBook providers is readily available to the U.S. government, bypassing Fourth Amendment protections.
- Given the potential impact of eBook surveillance, Amazon and other eBook providers have an obligation to clearly describe the data being collected, and to give readers the opportunity to opt out.



in the watchtower could observe the prisoners, but the prisoners could not see the guards. The panopticon thus created a surveillance regime in which the prisoners never knew when they were being observed, but the sense of being watched was always present. Bentham failed to get the necessary funding for his prison, and it was never built,^b but the underlying concept has

lived on as a metaphor for the perception of omnipresent surveillance.

Michel Foucault obtained the most traction from the concept, ignoring the more liberal aspects of the scheme to focus on the potential for the application of power.^c In *Discipline and Punish*, he characterized the panopticon, as illustrated in Figure 1, as inducing in the inmate “a state of conscious and per-

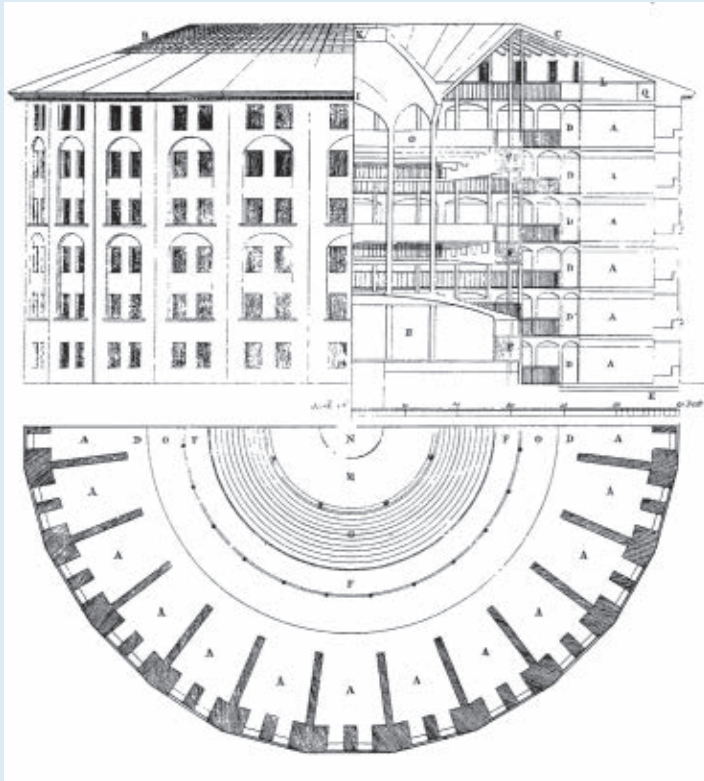
manent visibility that assures the automatic functioning of power.” Foucault then proceeded to find panopticons in various aspects of modern society, as have many scholars since.^d More recently the notion has been applied to virtually all forms of electronic surveillance; the authors and others, for example, have pointed to cellular net-

b J. Semple, *Bentham's Prison: A Study of the Panoptic Penitentiary*. Oxford University Press, 1993.

c J. Semple, Foucault and Bentham: A Defence of Panopticism, *Utilita I*, 1 (May 1992).

d M. Foucault, *Discipline and Punish*, Vintage, 1995, (Surveiller et punir: Naissance de la Prison, 1975).

Figure 1. The plan for Jeremy Bentham’s panopticon prison. This iconic blueprint was drawn by Willey Reveley in 1791.



works as forming panopticons: cellular technology tracks user movements, creating a detailed personal history that is available to law enforcement, advertisers, and hackers, but is invisible and inaccessible to the user herself.

In this article, we extend the panoptic metaphor to surveillance technologies that may be built into our eBooks. We choose the words “may be” with great care; our studies of Amazon’s patents indicate the potential for extensive surveillance, but when we asked Amazon to confirm or deny their use of these technologies, we received what can best be described as a non-answer.^e It follows that Kindle users do not know that the surveillance technologies described here are actually in use, only that they are available for use. And that, of course, reflects the under-

lying power of the panopticon.

Having described Amazon’s patented Kindle surveillance technology, we turn to the question of why we should care. Using case law and common sense, we suggest that anonymous reading is connected to free expression. Surveillance has a chilling effect on one’s choice of reading material, which in turn limits what one has to contribute to the marketplace of ideas. We conclude with a brief discussion of possible policy solutions.

Kindle Surveillance

To any avid reader, the Kindle/Kindle app is a truly wonderful technology. One can pack for a conference trip without worrying whether one will be in the mood for reading Turing, Kierkegaard, or Calvin and Hobbes. Whatever one chooses to read, it will be readily at hand. The Kindle user will see immediate evidence, however, that his or her reading is under some form of surveillance. Statements of the form “703 passages have been highlighted 6,855 times” greet the reader when opening a new book. Several questions arise, such as “How do they know this?” and

“What else do they know?” It was in trying to answer the latter question that this research project was born.

We began with the specifications that one sees when shopping for a Kindle on Amazon. The information is technically limited and straightforward. For example, some Kindles include GPS sensors, while most have an accelerometer. Both have benign uses; for example, GPS can be used to enforce copyright restrictions that may vary from country to country. The accelerometer can be used to sense the rotation of the display.

The “Kindle Store Terms of Use” proved more interesting. The terms begin with a clear statement that eBooks are licensed, not sold: Amazon states that “unless specifically indicated otherwise, you may not sell, rent, lease, distribute, broadcast, sublicense, or otherwise assign any rights to the Kindle Content or any portion of it to any third party.” The distinction is important as it allows the Kindle Store to avoid the “First-Sale Doctrine,” an aspect of copyright law that allows one to buy a book, and then turn around and resell it to a used book store or some other third party.^f The First-Sale Doctrine is based on the legal notion of “exhaustion”—the author’s interest in a specific copy of a book is exhausted after the first sale, and no royalties or similar forms of authorial control apply after that. Used bookstores may thus resell copies of books without having to compensate the author. There are limitations, of course; one is not allowed to make dozens of copies of a newly purchased book and sell the copies to one’s friends, but one may otherwise use, abuse, sell or destroy the single copy that was purchased.^g

Though Amazon’s positioning of its service within the laws of copyright is not strictly a matter of data collection, it does allow Amazon to call upon the full weight of a country’s judicial system should anyone choose to treat their eBooks like, say, a physical book. We will return to this point when we suggest policy solutions.

^e “Thank you for reaching out. I can share that some basic app, device, and usage data are logged in order to ensure the performance of our Kindle products and services and to improve the customer experience. We’re not in the business of selling customer information to others. You can read more about our practices in the Amazon Privacy Policy.” Kindle PR, email to the first author, July 12, 2018.

^f A. Perzanowski and J. Schultz, *The End of Ownership: Personal Property in the Digital Economy*. MIT Press, Cambridge, MA, 2016.

^g Copyright law creates a distinction between the author’s right to create copies of his or her text and the author’s rights with regard to a particular copy. It is the latter that is said to be exhausted after the first sale.

The “Kindle Store Terms of Use” further acknowledge data collection by the Kindle: “The Software will provide Amazon with information about use of your Reading Application and its interaction with Kindle Content and the Service (such as last page read, content archiving, available memory, up-time, log files, and signal strength). ... We will handle any information we receive in accordance with the Amazon.com Privacy Notice.” Note the use of “such as” in the parenthetical clause.

We were unable to find an explicit list of what Amazon was collecting in any publicly available article or notice; and as already noted, Amazon was unwilling to tell us what it was collecting. There is, however, one place where Amazon is apparently willing to advertise its capabilities: its patents. Patents are legal documents that are based on a *quid pro quo*: in return for a clear description of the invention, made available to all in the public domain, the inventor or inventors receive the right to prevent others from using their invention for a limited period of time.^h We wish to be clear—there is no guarantee that Amazon uses the technology described in its patents. What is instead provided is an indication as to what Amazon *can* do with its Kindle technology, an indication that is both informative and unsettling.

Amazon filed the application that became U.S. Patent No. 7,748,634 in 2006, a year before the first Kindle was released. This patent provides a general overview of an early eBook reader. Figure 2 of the patent (reproduced here) depicts the reader as a general-purpose computer with a few extras, including a “page-turn detector” and “communication connection(s).” There is little here that is remarkable from a surveillance standpoint. The communication connections, for example, are clearly necessary for obtaining eBooks—the eBook reader must somehow ingest books if the owner of the eBook reader is to have something to read.

The application that matured into U.S. Patent No. 9,390,402 (henceforth the ‘402 patent) was filed on June 30, 2009. This patent is much more interesting from a surveillance standpoint,

as it describes the collection of “annotation information, such as annotations made by users. Annotations can be in the form of notes, highlights, bookmarks, etc.”ⁱ Amazon may also collect the “location during access,” such locations including “venues such as airplanes, night clubs, restaurants, etc., specific geolocation such as 48.93861.degree. N 119.435.degree. W, or both.”^j Further, Amazon may collect “data derived from other sensor inputs, such as an accelerometer or ambient light sensor. For example, accelerometer input may provide data indicating the user reads while walking. In another example, ambient light input in conjunction with other [Content Access Information] may indicate that users have a greater rate of abandonment when reading in low light levels.”^k

An example may put this into context: through Kindle surveillance, Amazon potentially knows that one is reading a particular novel in a specific nightclub, that the lights are low, and that one’s reading is degrading over time.

Finally, Amazon may be evaluating one’s intelligence, at least as one’s intelligence is evinced by one’s preferred reading material. The ‘402 patent points to the Flesch-Kincaid Readability score as a means for evaluating the complexity of a given eBook. Amazon may thus track “a preferred maximum complexity level. For example, the user prefers content items not exceeding a grade 16 reading level.”^l And, of course, any desire to hide one’s interest in romance novels is lost—Amazon will know one’s “preferred genre of content items, such as mystery, science fiction, biography, horror, reference, etc.”^m

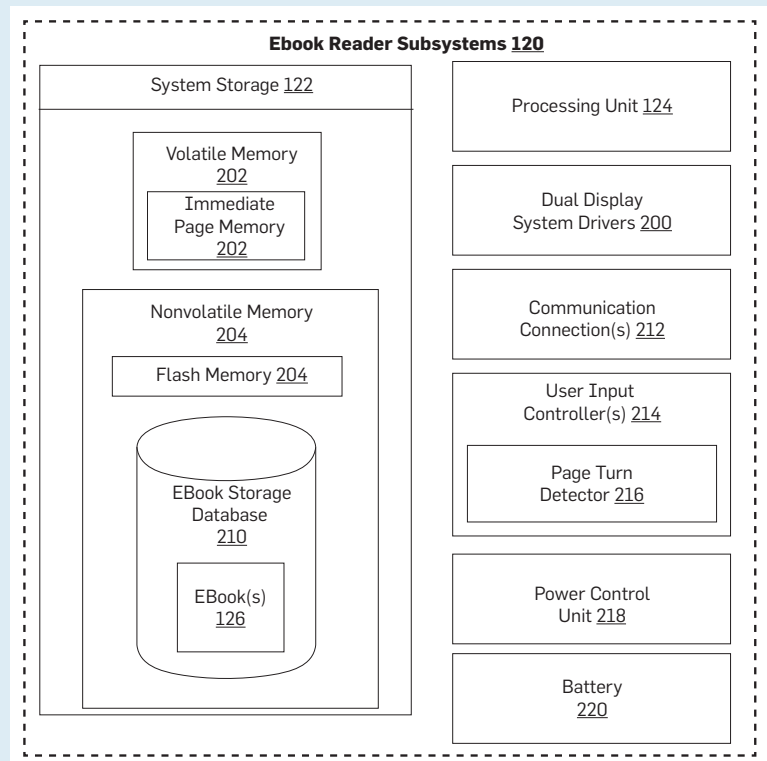
The ‘402 patent contains a paragraph that sums up the extent of the potential data collection quite nicely. It contains a few acronyms (CAE is a content access event), but we think the point is clear:

For example, the CAE collection module 316 may gather a set of CAEs from access device 104(1) indicating that the “Illustrated History of the Fork” was last displayed on screen two months ago for a period of ten minutes in a landscape presentation mode while on an airplane

i U.S. Patent No. 9,390,402, 10:63–65
 j U.S. Patent No. 9,390,402, 10:40–42
 k U.S. Patent No. 9,390,402, 10:49–55

l U.S. Patent No. 9,390,402, 11:8–17
 m U.S. Patent No. 9,390,402, 9:14–15

Figure 2. A reproduction of Amazon’s Kindle patent.



h In the U.S., patent validity extends for 20 years after the filing of the application. See 35 U.S. Code § 154.

at an altitude of 31,000 feet and speed of 513 miles per hour. Furthermore, the user only accessed seven pages of material during that time, and at the conclusion of the access, unloaded the content item from local storage on the access device 104(1). All of these factual data points may be captured as CAEs.

That is data collection indeed.

Why We Should Care

Amazon may have an immense trove of personal data collected from those who enjoy the convenience of the Kindle or the Kindle App. The question naturally arises as to why the general reader should care. To begin with, any information that one provides to Amazon may also be available to hackers and advertisers. The information is almost certainly available through subpoena to the federal government. In the 1976 case of *United States v. Miller*, the U.S. Supreme Court established that U. S. citizens have no reasonable expectation of privacy in information voluntarily given to third parties.ⁿ Law enforcement in the *Miller* case, for example, did not need a warrant to obtain copies of Mr. Miller's checks, information that was subsequently used to convict Miller of tax evasion. In *Smith v. Maryland*, this "third-party doctrine" was applied to the numbers dialed on a telephone; Mr. Smith had no reasonable expectation of privacy in the data he freely provided to the telephone company.^o

In the recently decided *Carpenter v. United States*, the U. S. Supreme Court held that a warrant was needed to obtain historical cell site data, but based its opinion on the "exhaustive chronicle of location information casually collected by wireless carriers today."^p The Court did not overturn *Miller* or *Smith*, and made it clear that exceptions to the warrant requirement would even hold for historical cell site data. One must assume that the data collected by Amazon would be available to the U.S. government upon issuance of a subpoena and would not require a warrant.^q

eBook surveillance is thus potentially part of a larger trend in which data



eBook surveillance is potentially part of a larger trend in which data collection that would be illegal if performed by a state actor has become a common business practice of a private actor.



collection that would be illegal if performed by a state actor has become a common business practice of a private actor. At least in the U.S., the difference to the surveilled individual is *de minimis*, as the government has ready access to the data. Given that there was a time when government surveillance of one's reading interests was a matter of personal safety, this should be a serious concern.^r

But one need not imagine a McCarthyesque set of hearings and the threat of prison to see that surveillance of the act of reading can have a negative impact. There is a substantial body of First Amendment jurisprudence that connects the right to read anonymously to freedom of expression. The 1965 case of *Lamont v. Postmaster General* provides an excellent example.^s Corliss Lamont was an American scholar, a former head of the American Civil Liberties Union, and an instructor at Cornell University. In the 1950s, Lamont was called before Senator Joseph McCarthy's senate subcommittee and questioned about his leftist inclinations. Lamont testified that he had never been a member of the Communist Party but invoked his First Amendment rights when questioned about his political opinions. He was cited for contempt but fought back in Federal Court and had the charges dismissed. He was a wealthy man, and well-placed to defend himself.

Our present interest in Lamont rests with his reading matter, and in particular, his subscription to the *Peking Review*. In 1962, Congress passed the Postal Service and Federal Employees Salary Act, section 305(a) of which required that the Postmaster General detain unsealed foreign mailings that contained "communist political propaganda," delivering it only upon the addressee's specific request. Upon receiving said propaganda, the post office would forward a card to the addressee. The addressee had to check a box indicating a desire to receive the material, and then return the card

^r For example, in 1953 senate investigator Roy Cohn interrogated Langston Hughes as follows: Q. You mean to say you have no familiarity with communism?

A. No, I would not say that, sir. I would simply say that I do not have a complete familiarity with it. I have not read the Marxist volumes. I have not read beyond the introduction of the *Communist Manifesto*.

^s *Lamont v. Postmaster General*, 381 U.S. 301 (1965)

ⁿ *United States v. Miller*, 425 U.S. 435 (1976)

^o *Smith v. Maryland*, 442 U.S. 735 (1979)

^p *Carpenter v. United States*, No. 16-402, 585 U.S. (2018)

^q A subpoena is much easier to obtain than a warrant.

to the post office. Lamont received such a card, and instead of checking and returning, he filed suit, insisting the affirmative act of checking the box violated his First Amendment rights to free expression. The Supreme Court agreed, writing in a unanimous opinion that the required request was “an unconstitutional abridgment of the addressee’s First Amendment rights.”

The Court’s logic in this case is particularly interesting—the justices concluded the requirement that the addressee request his material from the post office interfered with the addressee’s right to read anonymously. The Court found the interference took the form of a deterrent, or chilling effect on what the individual read.

The addressee carries an affirmative obligation which we do not think the Government may impose on him. This requirement is almost certain to have a deterrent effect, especially as respects those who have sensitive positions. Their livelihood may be dependent on a security clearance. Public officials like schoolteachers who have no tenure might think they would invite disaster if they read what the Federal Government says contains the seeds of treason. Apart from them, any addressee is likely to feel some inhibition in sending for literature which federal officials have condemned as “communist political propaganda.”

The “deterrent effect” in turn places limits on speech: what one does not take in, one cannot use in expressing new ideas. With this in mind, the Court found that section 305(a) was in conflict with the “uninhibited, robust, and wide-open” debate and discussion that are contemplated by the First Amendment. In short, the free and uninhibited collection of information is a critical element in the free expression of opinions—a cornerstone of democracy.

In *Kleindienst v. Mandel* (1972) Justice Thurgood Marshall reinforced the point, explicitly connecting input (in this case auditory) and output (speech):

The freedom to speak and the freedom to hear are inseparable; they are two sides of the same coin. But the coin itself is the process of thought and discussion. The activity of speakers becoming listeners and listeners becoming speakers in the vital interchange of thought is the

“means indispensable to the discovery and spread of political truth.”^u

Amazon’s surveillance capability and the subsequent chilling effect goes far beyond that of the post office in *Lamont*. To explore Marxism, sexuality, or addiction on one’s Kindle, one must allow Amazon to not only know that we may read the given material, but to know when, where, how much, and with which fellow Kindle consumers one is reading the material.

Policy Considerations and Conclusion

One may argue the Kindle user agrees to such surveillance by choice, and that we are free to walk away from our Kindles and resort to old-fashioned physical books that do not have the ability to monitor our reading habits. This is certainly a reasonable argument, but one last element must be brought into consideration. As we have seen, Amazon enjoys the benefit of U.S. Copyright laws. The U.S. is one of the few countries that considers copyright violation to be a criminal offense. Amazon may not only sue you; Amazon can have you put in jail.

To see the extent of Amazon’s protection, consider the *Digital Millennium Copyright Act* (DMCA), an act that makes it illegal to tamper with a technology that “controls access to a work” such as an eBook. The relevant language is reproduced here, with the most relevant parts underlined:

17 USC §1201–Circumvention of Copyright Protection Systems

(2) No person shall manufacture, import, offer to the public, provide, or otherwise traffic in any technology, product, service, device, component, or part thereof, that—

(A) is primarily designed or produced for the purpose of circumventing a technological measure that effectively controls access to a work protected under this title;^v

If one attempts to bypass or disable the surveillance technology in the Kindle, then one arguably runs afoul of the DMCA. Our Kindles thus provide us with a take-it-or-leave-it deal

^u *Kleindienst v. Mandel*, 408 U.S. 753,

^v 17 USC §1201, Circumvention of Copyright Protection Systems

in which, in return for the opportunity to read eBooks, we consent to surveillance that is open ended, undefined, and enforced by the full weight and power of the Federal Government.

What is to be done? Given the extent of its government protection and the ease of government access to the collected data, it seems reasonable to expect the eBook industry to accept some modest regulation. For example, at a minimum, readers should know precisely what data is being collected. It is not enough to be provided with examples (“such as”); readers need to know the full extent of data collection so that they may make fully informed choices when selecting an eBook reader, and when choosing a book to read with that device. We note there is evidence that privacy is becoming a marketable commodity and part of the business ethic of some companies (Apple is a notable example).

At the next level of regulation, one can imagine readers being given the ability to opt out of such data collection, perhaps for an added fee. As many readers will not bother to opt out, the provider should still have ample data on which to base its marketing schemes.

In yet another step, the public might insist that users have access to the data that has been collected. Such a regime is already in place in Europe. Note that Jeremy Bentham called for public inspection of his panopticon, requiring transparency of management to insure the welfare of the inhabitants. Cannot we ask for as much?

The publishing industry has a great deal of influence with legislators,^w so these forms of regulation may not be possible. The other approach is to publicize the data collection and hope that a market emerges for a surveillance-free eBook reader. Until then we must accept that our eBook readers are capable of a wide range of surreptitious surveillance. Your eBook provider may be watching. Or it may not. ■

^w J. Litman. Copyright and Compromise. *Digital Copyright*, Maize Books, 2017

Stephen B. Wicker is a professor of electrical and computer engineering at Cornell University, Ithaca, NY, USA.

Dipayan Ghosh is co-director of the Digital Platforms & Democracy Project and Shorenstein Fellow at the Harvard Kennedy School, Cambridge, MA, USA.

^t *Lamont v. Postmaster General*, 381 U.S. 301 (1965)

DOI:10.1145/3376901

Identifying female CS scientists by combining a robust bibliographic database and name filtering tools.

BY SANDRA MATTAUCH, KATJA LOHMANN, FRANK HANNIG, DANIEL LOHMANN, AND JÜRGEN TEICH

A Bibliometric Approach for Detecting the Gender Gap in Computer Science

WOMEN ARE UNDERREPRESENTED in the fields of science, technology, engineering, and mathematics (STEM) in most countries, including Germany and the U.S.^{29,32} This was demonstrated in several surveys investigating the proportion of women in the STEM fields for specific populations. Some of these studies, for example, investigated the number of enrolled students^{10,30} or the percentage of female professors at universities. Other studies analyzed the disparities in research funding.²³ Nearly all these surveys selected a particular population of women in consideration of their university degree

or their nationality.^{11,34} Like many other studies investigating the gender gap and its reasons in science, these surveys are usually based on data records from several kinds of registrations or enrollments, for example, the enrollment as student or doctoral student, the registration of finished doctoral theses or the membership as professor in a certain country.^{1,14,16,28} However, researchers at the postdoctoral level or industrial researchers are often not registered and unfortunately drop out of the surveys.

Bibliometric approaches are widely used to detect the gender gap and to determine possible reasons for it,^{4,12,15,33} for example, the research performance or collaboration behavior^{1,2,4,18} or different cognitive or sociocultural determinants.^{9,13,16} In this study, we use a method to detect the gender gap in the group of scientifically active researchers regardless of the limitations mentioned and focused to a certain scientific field. The group of interest comprises scientists that are currently active in doing research and publishing their findings—regardless of their university degree, nationality, gender, age, or origin and irrespective of their employment level in university or industry. As a case study, we measured the gender gap in the scientific field of the Transregional Research Centre 89 Invasive Computing (CRC/Transregio 89),^a which investigates a novel paradigm for the design and programming of future parallel computing systems and covers research from diverse domains of computer science and

a <http://www.invasic.de>

» key insights

- The bibliometric approach allows to estimate the proportion of scientifically active women in CS, regardless of their degree, employment level, nationality, age, or origin.
- The percentage of women contributing to 19 representative conferences in CS within the last six years is, on average, below 10%.
- The percentage of women shows only small variations over individual years and conferences.

Table 1. Selected conferences.

Conference name and abbreviation

- International Conference on Applied Cryptography and Network Security (ACNS)
- International Conference on Architecture of Computing Systems (ARCS)
- International Conference on Application-specific Systems, Architectures and Processors (ASAP)
- Asia and South Pacific Design Automation Conference (ASP-DAC)
- International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)
- International Conference on Compilers, Architectures, and Synthesis for Embedded Systems (CASES)
- International Conference on Compiler Construction (CC)
- International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS)
- Design Automation Conference (DAC)
- Design, Automation and Test in Europe (DATE)
- International European Conference on Parallel and Distributed Computing (Euro-Par)
- European Conference on Computer Systems (EuroSys)
- International Conference on Parallel Computing (ParCo)
- Symposium on Operating Systems Principles (SOSP)
- USENIX Annual Technical Conference (USENIX)
- International Conference on Virtual Execution Environments (VEE)
- Conference on Design and Architectures for Signal and Image Processing (DASIP)
- International Conference on Humanoid Robots (Humanoids)
- Network and Distributed System Security Symposium (NDSS)

Table 2. List of Excluded Classes.

Onomastic Classes

- Hong Kong
- China
- Taiwan
- Republic of Korea
- Viet Nam
- Democratic People's Republic of Korea

electrical engineering, such as computer engineering, operating systems, programming languages, security, robotics, and high-performance computing. To ensure only scientifically active scientists are taken into account, we decided to collect data of researchers that successfully published their results in proceedings of international conferences within the last six years.

Conferences and the appropriate conference proceedings are the common publication medium in computer science and have a much higher impact than journal papers. For this purpose, and for working with representative and high-quality data, we used the DBLP Computer Science Bibliography,⁸ which lists the major computer science journals and conference proceedings, as our database. Table 1 presents a summary and selection of the 19 most relevant conferences for different disciplines of our CRC/Transregio 89. Based on this selection, we developed a Perl script extracting the author names by the given

constraints (conference name and a period of six years). Based on the filtered results, we subsequently determined the country of origin and the gender of each author by NamSor Applied Onomastics.²⁰ We finally verified this approach by random sampling and manual classification of the sampled names. The extracted information was then used to detect the gender gap in the field of the CRC/Transregio 89 Invasive Computing.

Methods

Extraction of author names from the DBLP Computer Science Bibliography.

To gather the original population of all scientifically active researchers within the scientific field described, we extracted the names of authors contributing to most relevant conferences (Table 1) within the last six years from the DBLP Computer Science Bibliography.⁸

The DBLP Computer Science Bibliography provides bibliographic information on all major computer science journals and proceedings. This open-data service indexes more than 4 million articles, published by more than 2.1 million authors.⁸

To pull the author names from the DBLP database, we created a Perl script: This script, which is publicly available under MIT license^b extracts all author names—regardless of the

^b <https://github.com/luhsra/venueauthor>

order of authors—for all papers published at a certain conference. The conference is defined by the input variables *venue* and *year*. The *venues* are the acronyms of the conferences as listed in Table 1. For *years*, we chose the 2012 to 2017. The script displays a list with the authors' first and last name, and the conference name and year. The resulting population comprises of 18,116 authors. Some 242 authors used abbreviations instead of first names, so these names were excluded from the analyses, resulting in an original population of 17,874 names.

Data handling. The extracted author names from the DBLP database were subsequently classified by NamSor Applied Onomastics, a name recognition software provided by a private start-up company.²⁰ The specialized data mining software also recognizes the linguistic or cultural origin of each personal name in any alphabet/language and allocates an onomastic class and the gender to each author name. The innovative machine learning algorithm provides unmatched accuracy at a fine-grained level, with flexibility and integration capability, to filter through large databases and extract names. It recognizes which language or culture stands behind a given name.²⁰ It is already known that the cultural context and origin are important for the determination of gender by name. Therefore, some names cannot be clearly defined without the origin. The name *Andrea*, for example, is a male name in Italy, but a female name in Spain. Some more examples are Jean, Joan, Laurence, Sascha, and Maria. To ensure a high degree of accuracy in the classification of the author names and to take the cultural context and the origin into account, we decided to use NamSor Origin API first, followed by NamSor Gender API.

Determination of the likely country of origin of a name by NamSor Origin API. NamSor Origin API allows determining the likely country of origin of each author, based on the sociolinguistics of the name (language, culture). The anthroponomical classification can be summarized as follows: Judging from the name only and the publicly available list of all 150k Olympic athletes since 1896 (and other similar lists of names), for which national team would

the person most likely run? Here, the U.S., Australia, among others are typically considered as a melting pot of other cultural origins (Ireland, Germany, among others) and not as an onomastic class on its own.^{25,27}

Based on the NamSor Origin API algorithm, the basic population of 17,874 authors was classified into 71 onomastic classes. The 20 proportionally largest classes represent 82.3% of the basic population. 16 onomastic classes have less than 20 authors listed and represent together under 1% of the basic population. The classification of cultural and geographical provenience of the author names by the NamSor Origin API algorithm shows that our data set is reasonably diverse and shows an acceptable variability with respect to the origin.

Determination of the likely gender of a name by using NamSor Gender API. For this task, we used the NamSor Gender API. The software predicts the gender of a personal name on a -1 (male) to +1 (female) scale and covers the U.S., European, Indian, African, Chinese, Hebrew, Russian/Slavic/Cyrillic, and Arabic names. In this step, the software combines two algorithms to maximize accuracy. First, a unique global name sociolinguistics algorithm that recognizes the origin of the couple first name and last name and infers whether the name sounds male or female in that particular culture. Second, a query in a massive database (800,000 names), which contains statistical information about baby names in each country of the world.¹⁹ Nevertheless, NamSor recommends passing additional geography/local context to the names to improve the accuracy of classification.¹⁹ The reliability of this method was already investigated in several publications.^{6,25-27,31}

Figure 1 reveals that 67.7% of the author names are classified as male and only a small proportion of 9.9% are classified as female names. Some 22.4% of the names in the basic population are unclassified (scale 0). These not classified names mainly have two reasons: names like Kerry, Jean, or Maria that are not strongly correlated to gender, and the structure and usage of Asian names.

Removal of Asian names. In most countries and cultures, the method of onomastics is very accurate, with a

precision in the range of 95%–99%—but we should pay attention to the structure of Asian names. The used Perl script generates a list of authors with first name and family name. In Asia, the family name comes first, followed by the first name. Although there are currently over 4,000 Chinese surnames, only 100 surnames still make up over 85% of China’s 1.3 billion citizens. In fact, just the top three Wang, Li, and Zhang cover more than 20% of the population.²² The situation is aggravated by the fact that a lot of Chinese names are not strongly correlated with gender. Moreover, if they were transliterated in Latin characters, even more information gets lost. The automatic determination of gender from Asian names with sufficient accuracy is not within the bounds of possibility of this work.³⁵ The analysis shows that 96.3% of the unclassified names come from these six onomastic classes. For these reasons, we decided to exclude all these Asian names from the onomastic classes listed in Table

2. Removal of these names reduces the population by 4,773 to 13,101 names. After the removal of Asian names, 149 unclassified names are remaining.

In Figure 2, the distribution of male, female, and unclassified authors after the removal of Asian names is shown. The percentage of female authors increases slightly to 11.3%, but the number of unclassified names has been reduced to 1.1%. The number of male authors has increased accordingly to 87.5%.

Validation of name sorting. After applying the procedure described earlier, we ended up with a population of 13,101 names (basic population): 1,486 names were classified as female names, 11,466 as male names. To test whether the names classified as female names really belong to women and—vice versa—those classified as male names really belong to men, we randomly selected samples from the basic population of men and women. The minimal sample sizes n of women and men is calculated using the following formula:

Figure 1. Distribution of female, male, and unclassified names as assorted by NamSor Gender API in the original population.

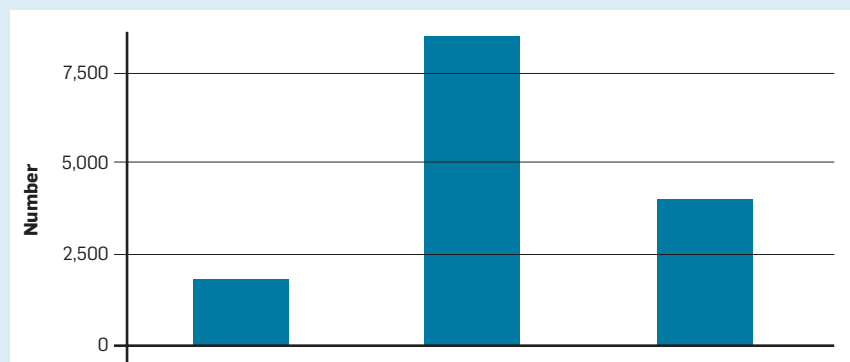
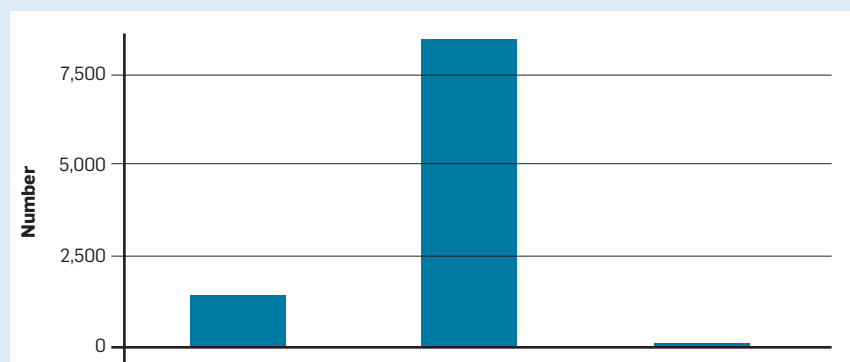


Figure 2. Distribution of female, male, and unclassified names as assorted by NamSor Gender API in the population when disregarding Asian names.



$$n \geq \frac{N}{1 + \frac{(N_1) \cdot e^2}{z^2 \cdot P \cdot (1 - P)}}$$

In Equation (1), N is the number of elements in the stock population, e the margin of error (5%), z is the z-score (1.96 for a confidence level of 95%), and P the prior judgment of the correct distribution (0.5, no prior judgment).

This gives us a sample size of 306 for the group of female names and 372 for the group of male names. The gender of scientists from these sample groups was manually verified by searching them on the Internet—assuming scientifically active persons to have an Internet presence. We determined the gender of the scientists by photos and the usage of gender-specific keywords (he, she, him, her, among others) on the personal homepages, on platforms like LinkedIn¹⁷ or ResearchGate²⁴ or pages referring to the scientist, for instance, as authors.

The results are shown in Figure 3. The estimation of the likely gender of a person by “NamSor Gender API” works quite well for male scientist but noticeably not as good for the group of female scientists: In the group of men, 84% were correctly verified to be male, only 0.3% were female, and 15.6% could not be verified due to no Internet presence. In the group of women, only 70% were correctly verified as female, yet 17.6% were male and 12% could not be found on the Internet.

In addition to the determination of the likely gender on the basis of the country of origin, we evaluated the gender classification accuracy when alternatively using the affiliation country extracted from the Scopus database. Scopus, Elsevier’s abstract and citation database generates precise citation search results and automatically updates researcher and institution profiles, unlike the DBLP database. The Python script we developed to extract the affiliation country of authors is publicly available under MIT license.^c To compare the classification accuracy of both approaches, the hand-verified set of 314 male and 269 female names serves as input. The percentage of true classifications for the first method, for example, is obtained as the number of correctly classified samples (528) in re-

lation to the total of 583 samples. We could show that there is a difference of less than 1% in classification accuracy when either using the country of origin or the affiliation country as input for the NamSor Gender API (see Figure 3).

Based on these random experiments, we decided to correct the automatically determined number of female and male authors accordingly using the following term:

$$F_{corr} = F \cdot corr_{ff} + M \cdot corr_{fm} \quad (2)$$

$$M_{corr} = M \cdot corr_{mm} + F \cdot corr_{mf} \quad (3)$$

In Eqs. (2) and (3), F_{corr} and M_{corr} denote the corrected numbers of women and men, F and M are the original values obtained from the name-sorting procedure, and $corr_x$ are the correction factors estimated from the results of the verification of name sorting:

$$corr_{ff} = \text{females in female group} = 0.70$$

$$corr_{mm} = \text{males in male group} = 0.84$$

$$corr_{fm} = \text{females in male group} = 0.003$$

$$corr_{mf} = \text{males in female group} = 0.17$$

The results shown next present corrected percentages of female and male researchers using Eqs. (2) and (3).

Case Study

For the 19 representative computer science conferences selected for our analysis as shown in Table 1, we extracted from the DBLP Computer Science Bibliography a total of 18,116 names of authors contributing to these conferences within the last six years and removed 242 authors that used initials instead of the full first names (original population). The names were then classified by origin and gender using the NamSor Applied Onomastics. From the original population, 4,773 author names assigned to Hong Kong, China, Taiwan, Republic of Korea, Viet Nam, and the Democratic People’s Republic of Korea were removed due to the infeasibility of automatic classification. A small number of 149 names (0.8%) were left unclassified for unknown reasons.

After applying the presented stochastic sampling of this population and subsequently applying the correction according to Eqs. (2) and (3) on the resulting basic population of 13,101 names, we could finally estimate that the percentage of women contributing

to the 19 conferences within the last six years is, on average, below 10% (as illustrated in Figure 4). On a per year basis, the percentage of female authors shows only small variations between 8.68% in 2012 and 10.1% in 2016.

Our approach now allows us to have a closer look at the proportion of scientifically active women in different individual conferences, and thus areas of computer science and not only to calculate the overall proportion of women in computer science as a whole. To illustrate the percentage of female authors in individual conferences, we picked out three of them: The International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), the Design, Automation and Test in Europe (DATE) and the International Conference on Compiler Construction (CC). The percentage of female authors varies here between 6.2% for the CC and 11.7% for the CODES+ISSS conference. For the DATE conference, the percentage of female authors amounts to an average value of 9.6%.

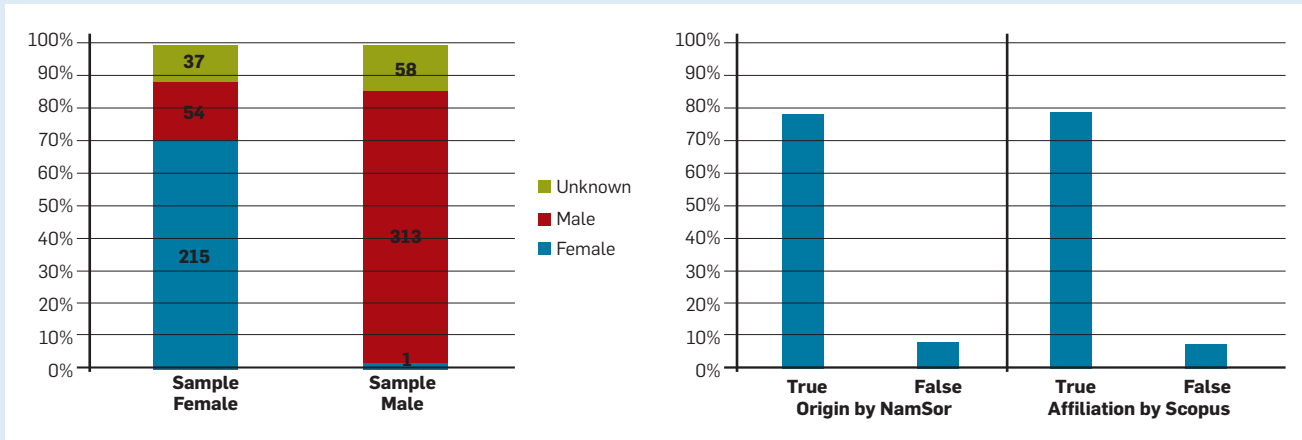
A closer look at the participation of women in all 19 conferences finally reveals a nearly symmetrical distribution. Five of the investigated conferences have a percentage of female authors above 10%, and five conferences have a proportion of female authors below 8.0% (see Table 3).

Table 3. Percentage of female authors in the examined conferences.

Conference	Percentage of female authors
CODES+ISSS	11.73
ACNS	11.51
Humanoids	10.98
DAC	10.41
CASES	10.26
ASP-DAC	9.94
Euro-Par	9.80
VEE	9.72
DATE	9.63
NDSS	9.61
DASIP	9.39
PARCO	8.68
EuroSys	8.48
USENIX	8.32
ASPLOS	7.96
SOSP	7.00
ARCS	6.72
ASAP	6.70
CC	6.15

^c <https://github.com/luhsra/venueauthor>

Figure 3. Results of manual verification of gender classification using samples of names classified as female, respectively male (left) and comparison of accuracy when either using country of origin determined by NamSor or affiliation country extracted from Scopus as an alternative (right).



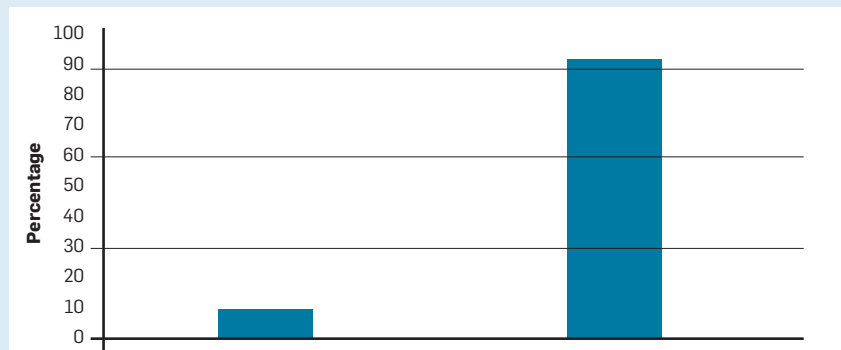
Discussion

In this work, we used a bibliometric approach to estimate the proportion of scientifically active women in the specific scientific field of computer science. In contrast to previous studies in the STEM fields that refer to limited data records, our method provides a more general approach with reduced limitations:

- We make sure to take all authors with publication activity in the last six years into account—independent of their university degree. Along with professors and postdoctoral, and industrial researchers, the examined group includes senior lecturers, doctoral students, career changers, and even employees without an academic degree like technicians or qualified IT specialists. Our approach allows us to exclude researchers that are not scientifically active anymore, for example, due to a change in their scientific field or job. Also, researchers active in administration or management are omitted, as well as students at the beginning of their studies. We cannot exclude that our results are partly influenced by an imbalance in research activity between female and male researchers, as has been shown for other scientific areas.^{5,15,21,28} However, since we consider the number of authors and not the number of publications, we assume this influence is relatively small.

- We generate our population independent of the origin of the authors. On the selected international conferences, one can find conference delegates from all over the world. As expected, we found author names from 71 different on-

Figure 4. Final distribution of female and male names for 19 conferences in computer science and electrical engineering after removal of Asian and unclassified names, and correction using stochastic samples and applying Eqs. (2) and (3).



mastic classes on our list, reflecting the likely country of origin of the authors. Our approach also provides the possibility to generate a population of authors only for national conferences or for individual conferences.

Compared to many previous studies searching for female scientists in computer science, our approach makes it possible to focus the analysis to a single conference further, a set of conferences representative for a specific scientific field, or to limit the data to a certain period of time. For the case study presented here, we examined representative conferences suggested by the researchers of the CRC/Transregio 89 Invasive Computing, which covers computer engineering, operating systems, programming languages, security, and the field of application including robotics and high-performance computing. By the selection of conferences, it would be pos-

sible to investigate other scientific fields or to limit further the scientific area (for example, to operating systems or computer security).

Despite these advantages of the method, we are not able to directly extract the gender or origin of the authors from the DBLP Computer Science Bibliography, one reason being that DBLP does not list these properties. By applying NamSor Applied Onomastics, we were able to determine the gender of the authors automatically. Yet, after testing the accuracy of this fully automatic classification on random samples from the group of men and women, we found out that although only one man was wrongly classified, 17.6% of those classified as women were in fact men. A more thorough inspection indicated that 24.1% of wrongly classified women were from India. These differences in accuracy between men and women through verification by random

sampling is not explained by NamSor Gender API. Indeed, they do not provide any information about the classification of names from India. To take the wrong classifications into account, we determined corrective factors.


The most significant disadvantage and a potential source of error of our approach is the removal of names classified as Asian names. The excluded group comprises a total of 4,773 names, which amounts to 26.7% of all names in the original population obtained from the DBLP Computer Science Bibliography. The removal of these names may distort the results. However, there is no evidence so far that the proportion of women in the group of removed Asian names is significantly higher than in the investigated group. In fact, several studies on women in the STEM disciplines in Asia indicate that the proportion of female students is even lower than in other parts of the world.^{3,30} For the approach introduced in this study, there was no possibility to determine the gender on the basis of an Asian name, as explained in detail previously. The use of the *Chinese Name Gender Guesser*⁷ or other software platforms was not taken into consideration because these take the traditional Chinese characters of the name to classify the gender.

For our analysis, we also removed 391 additional names of unknown gender due to missing information. For example, 242 authors submitted only a single character as the first name. There is obviously no way to determine the gender by one letter. However, there is no evidence that there is a disproportionate percentage of women in this group. These names reflect 2.2% of the entire population and were therefore neglected.

Another assumption taken in this study is the Internet presence of the authors for the estimation of the correction factors in Eqs. (2)-(3). This assumption, however, turned out not to be critical since the percentages of authors not found on the Internet are in the same range for female and male authors.

In conclusion, we are presenting a bibliometric method to capture and classify female scientists that are currently active in research and in each a specific field of computer science. The group of female authors we captured with our method includes those female scientists successfully publishing their

research findings in peer-reviewed publications and, thus, having an impact on their scientific community. The data was collected regardless of the university degree and irrespective of whether the scientist is employed at a university or industry. The data provided by the presented method is closing the gap of postdoctoral researchers in industry and university existing in many other surveys of women in science. The method allows estimating the number of female candidates suitable for recruiting them as high-potential postdocs or professors and could also be used to address other questions of interest in the area of gender research as well as in a more general context of university research.

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - Project number 146371743 - TRR 89: Invasive Computing. 

References

1. Abramo, G., D'Angelo, C.A., and Caprasecca, A. The contribution of star scientists to sex differences in research productivity. *Scientometrics* 81, 1 (2009), 137–156; doi: 10.1007/s11192-008-2131-7.
2. Abramo, G., D'Angelo, C.A., and Murgia, G. Gender differences in research collaboration. *J. Informetrics* 7, 4 (2013), 811–822; doi: 10.1016/j.joi.2013.07.002.
3. Association for Academics and Societies in Sciences in Asia. *Women in Science and Technology in Asia*. Panmun Education Co., Ltd, Aug. 2015; http://bit.ly/38ZfIU.
4. Araújo, T. and Fontainha, E. The specific shapes of gender imbalance in scientific authorships: A network approach. *J. Informetrics* (Feb. 2017), 88–102; doi: 10.1007/s11192-011-0369-y.
5. Arensbergen, P., van der Weijden, I. and van den Besselaar, P. Gender differences in scientific productivity: A persisting phenomenon? *Scientometrics* 93, 3 (2012), 857–868; doi: 10.1007/s11192-012-0712-y.
6. Carsenat, E. What's the gender gap in the European Union Whoiswho? Sept. 2014; http://blog.namsor.com/2014/09/09/whats-the-gender-gap-in-the-european-union-whoiswho/.
7. Chinese Name Gender Guesser. July 2018; http://www.chinesetools.com/tools/gender-guesser.html.
8. *dblp computer science bibliography*. July 2018; http://dblp.uni-trier.de/db/.
9. De Paola, M. and Scoppa, V. Gender discrimination and evaluators' gender: Evidence from Italian academia. *Economica* (Dec. 2013), 211–213; doi: 10.1111/ecca.12107.
10. Denisco, A. *The state of women in computer science: An investigative report*. Sept. 2017; https://tek.io/2QcvKCm.
11. European Commission. *She Figures 2015—Gender in Research and Innovation*; doi: 10.2777/744106.
12. Holman, L., Stuart-Fox, D., and Hauser, C.E. The gender gap in science: How long until women are equally represented? *PLoS Biology*, 2018.
13. Hyde, J.S., Fennema, E., and Lamon, S.J. Gender differences in mathematics performance: A meta-analysis. *Psychological Bulletin* 107 (Apr. 1990), 139–55; doi: 10.1037/0033-2909.107.2.139.
14. Jonathan, C. and Zuckerman, H. The productivity puzzle. *Advances in Motivation and Achievement* (Jan. 1984), 217–258.
15. Larièvre, V., Ni, C., Gingras, Y., Cronin, B., and Sugimoto, C. Bibliometrics: Global gender disparities in science. *Nature* 504 (Dec. 2013), 211–213; doi: 10.1038/504211a.
16. Larièvre, V., Vignola-Gagné, E., Villeneuve, C., Gélinas, P., and Gingras, Y. Sex differences in research funding, productivity and impact: An analysis of Québec university professors. *Scientometrics* 87, 3 (June 2011), 483–498; doi: 10.1007/s11192-011-0369-y.
17. LinkedIn. July 2018; url: https://www.linkedin.com.

18. Mihaljevic-Brandt, H., Santamaria, L. and Tullney, M. The effect of gender in the publication patterns in mathematics. *PLoS ONE* 11 (Oct. 2016), e0165367; doi: 10.1371/journal.pone.0165367.
19. NamSor Gender API. Sept. 2015; http://blog.namsor.com/api/.
20. NamSor Origin API. Sept. 2015; http://blog.namsor.com/name-recognition-software/.
21. O'Brien, K.R. and Hapgood, K.P. The academic jungle: Ecosystem modelling reveals why women are driven out of research. *Oikos* 121, 7 (2012), 999–1004; doi: 10.1111/j.1600-0706.2012.20601.x.
22. *People's Daily*. Chinese surname shortage sparks rethink. May 2007; http://en.people.cn/200706/19/eng20070619_385861.html.
23. Ranga, M., Gupta, N., and Etkowitz, H. *Gender Effects in Research Funding*, Mar. 2012.
24. Researchgate. *LinkedIn*. July 2018. url: https://www.researchgate.net.
25. Santamaria, L. and Mihaljević, H. Comparison and benchmark of name-to-gender inference services. In: *PeerJ Computer Science* (2018). doi: 10.7717/peerj-cs.156.
26. Science-Matrix, Inc. *Analytical support for bibliometrics Indicators to measure women's contribution to scientific publications*. Jan. 2018; http://bit.ly/35MNIao.
27. Shokhenmayer, E. and Carsenat, E. *Onomastics to Measure Cultural Bias in Medical Research Sing Scientists' Personal Name*. (Aug. 2014); http://bit.ly/2tGdSbr.
28. Stack, S. Gender, children and research productivity. *Research in Higher Education* 45, 8 (Dec. 2004), 891–920; doi: 10.1007/s11162-004-5953-z.
29. Stephen, C.J. and Williams, W.M. Understanding current causes of women's underrepresentation in science. In *Proceedings of the National Academy of Sciences* 108, 8 (2011), 3157–3162; doi: 10.1073/pnas.1014871108.
30. *Studentinnenanteile in Mathematik, Naturwissenschaften und Informatik sowie Ingenieurwissenschaften im internationalen Vergleich*. Center of Excellence Woman and Science, 2016.
31. Vichnevskaja, T. *Applying Onomastics to Scientometrics*. Jan. 2015; https://inserm.academia.edu/taniavichnevskaja.
32. Wang, M.T. and Degol, J.L. Gender gap in science, technology, engineering, and mathematics (STEM): Current knowledge, implications for practice, policy, and future directions. *Educational Psychology Review* 29, 1 (Mar. 2017), 119–140; doi: 10.1007/s10648-015-9355-x.
33. West, J., Jacquet, J., King, M., Correll, S.J. and Bergstrom, C.T. The role of gender in scholarly authorship. *PLoS ONE* 8 (July 2013), e66212; doi: 10.1371/journal.pone.0066212.
34. *Women, Minorities, and Persons with Disabilities in Science and Engineering*. National Science Foundation, National Center for Science and Engineering Statistics, 2018; http://www.nsf.gov/statistics/wmpdp/.
35. Zhao, H. and Kamareddine, F. Recursion identify algorithm for gender prediction with Chinese names. In *Proceedings of the Intern. Conf. Data Science* (Las Vegas, NV, USA, July 3–Aug. 2, 2018), 137–142.

Sandra Mattauch is a postdoctoral researcher at Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen, Germany.

Katja Lohmann is a postdoctoral researcher at Leibniz Universität Hannover, Germany.

Frank Hannig is a lecturer and senior researcher at Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen, Germany.

Daniel Lohmann is a professor at Leibniz Universität Hannover, Germany.

Jürgen Teich is a professor at Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen, Germany.

Copyright held by ACM.



Watch the authors discuss this work in the exclusive *Communications* video. <https://cacm.acm.org/videos/gender-gap>

Concurrency

The Works of Leslie Lamport

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

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

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

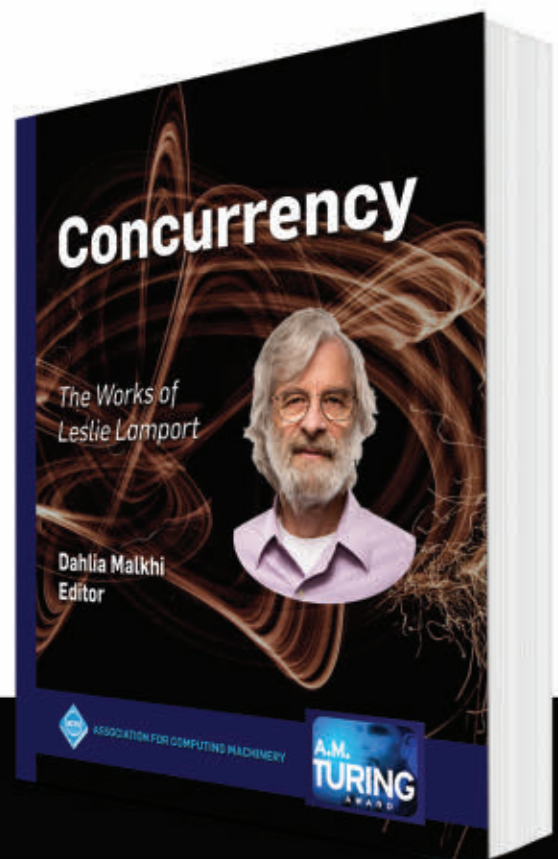
Dahlia Malkhi, Editor

ISBN: 978-1-4503-7271-8

DOI: 10.1145/3335772

<http://books.acm.org>

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



ACM BOOKS
Collection II

A group of industry, academic, and government experts convene in Philadelphia to explore the roots of algorithmic bias.

BY ALEXANDRA CHOULDECHOVA AND AARON ROTH

A Snapshot of the Frontiers of Fairness in Machine Learning

THE LAST DECADE has seen a vast increase both in the diversity of applications to which machine learning is applied, and to the import of those applications. Machine learning is no longer just the engine behind ad placements and spam filters; it is now used to filter loan applicants, deploy police officers, and inform bail and parole decisions, among other things. The result has been a major concern for the potential for data-driven methods to introduce and perpetuate discriminatory practices, and to otherwise be unfair. And this concern has not been without reason: a steady stream of empirical findings has shown that data-driven methods can unintentionally both encode existing human biases and introduce new ones.^{7,9,11,60}

At the same time, the last two years have seen an unprecedented explosion in interest from the academic community in studying fairness and machine learning. “Fairness and transparency” transformed from a niche topic with a trickle of papers produced every year (at least since the work of Pedresh⁵⁶ to a major subfield of machine learning, complete with a dedicated archival conference—ACM FAT*). But despite the volume and velocity of published work, our understanding of the fundamental questions related to fairness and machine learning remain in its infancy. What should fairness mean? What are the causes that introduce unfairness in machine learning? How best should we modify our algorithms to avoid unfairness? And what are the corresponding trade offs with which we must grapple?

In March 2018, we convened a group of about 50 experts in Philadelphia, drawn from academia, industry, and government, to assess the state of our understanding of the fundamentals of the nascent science of fairness in machine learning, and to identify the unanswered questions that seem the most pressing. By necessity, the aim of the workshop was not to comprehensively cover the vast growing field, much of which is empirical. Instead, the focus was on theoretical work aimed at providing a scientific foundation for understanding algo-

» key insights

- **The algorithmic fairness literature is enormous and growing quickly, but our understanding of basic questions remains nascent.**
- **Researchers have yet to find entirely compelling definitions, and current work focuses mostly on supervised learning in static settings.**
- **There are many compelling open questions related to robustly accounting for the effects of interventions in dynamic settings, learning in the presence of data contaminated with human bias, and finding definitions of fairness that guarantee individual-level semantics while remaining actionable.**



rhythmic bias. This document captures several of the key ideas and directions discussed. It is not an exhaustive account of work in the area.

What We Know

Even before we precisely specify what we mean by “fairness,” we can identify common distortions that can lead off-the-shelf machine learning techniques to produce behavior that is intuitively unfair. These include:

1. *Bias encoded in data.* Often, the training data we have on hand already includes human biases. For example, in the problem of recidivism prediction used to inform bail and parole decisions, the goal is to predict whether an inmate, if released, will go on to commit another crime within a fixed period of time. But we do not have data on who commits crimes—we have data on who is arrested. There is reason to believe that arrest data—especially for drug crimes—is skewed toward minority populations that are policed at a higher rate.⁵⁹ Of course, machine learning techniques are designed to fit the data, and so will naturally replicate any bias already present in the data. There is no reason to expect them to remove existing bias.

2. *Minimizing average error fits majority populations.* Different populations of people have different distributions over features, and those features have different relationships to the label that we are trying to predict. As an example, consider the task of predicting college performance based on high school data. Suppose there is a majority population and a minority population. The majority population employs SAT tutors and takes the exam multiple times, reporting only the highest score. The minority population does not. We should naturally expect both that SAT scores are higher among the majority population, and that their relationship to college performance is differently calibrated compared to the minority population. But if we train a group-blind classifier to minimize overall error, if it cannot simultaneously fit both populations optimally, it will fit the majority population. This is because—simply by virtue of their numbers—the fit to the majority population is more important to overall error than the fit to

Given the limitations of extant notions of fairness, is there a way to get some of the “best of both worlds?”

the minority population. This leads to a different (and higher) distribution of errors in the minority population. This effect can be quantified and can be partially alleviated via concerted data gathering effort.¹⁴

3. *The need to explore.* In many important problems, including recidivism prediction and drug trials, the data fed into the prediction algorithm depends on the actions that algorithm has taken in the past. We only observe whether an inmate will recidivate if we release him. We only observe the efficacy of a drug on patients to whom it is assigned. Learning theory tells us that in order to effectively learn in such scenarios, we need to explore—that is, sometimes take actions we believe to be sub-optimal in order to gather more data. This leads to at least two distinct ethical questions. First, when are the individual costs of exploration borne disproportionately by a certain sub-population? Second, if in certain (for example, medical) scenarios, we view it as immoral to take actions we believe to be sub-optimal for any particular patient, how much does this slow learning, and does this lead to other sorts of unfairness?

Definitions of fairness. With a few exceptions, the vast majority of work to date on fairness in machine learning has focused on the task of batch classification. At a high level, this literature has focused on two main families of definitions:^a statistical notions of fairness and individual notions of fairness. We briefly review what is known about these approaches to fairness, their advantages, and their shortcomings.

Statistical definitions of fairness. Most of the literature on fair classification focuses on statistical definitions of fairness. This family of definitions fixes a small number of protected demographic groups G (such as racial groups), and then ask for (approximate) parity of some statistical measure across all of these groups. Popular measures include raw positive classification rate, considered in

^a There is also an emerging line of work that considers causal notions of fairness (for example, see Kilbertus,⁴³ Kusner,⁴⁸ Nabi⁵⁵). We intentionally avoided discussions of this potentially important direction because it will be the subject of its own CCC visioning workshop.

work such as Calders,¹⁰ Dwork,¹⁹ Feldman,²⁵ Kamishima,³⁶ (also sometimes known as statistical parity,¹⁹ false positive and false negative rates^{15,29,46,63} (also sometimes known as equalized odds²⁹), and positive predictive value^{15,46} (closely related to equalized calibration when working with real valued risk scores). There are others—see, for example, Berk⁴ for a more exhaustive enumeration.

This family of fairness definitions is attractive because it is simple, and definitions from this family can be achieved without making any assumptions on the data and can be easily verified. However, statistical definitions of fairness do not on their own give meaningful guarantees to individuals or structured subgroups of the protected demographic groups. Instead they give guarantees to “average” members of the protected groups. (See Dwork¹⁹ for a litany of ways in which statistical parity and similar notions can fail to provide meaningful guarantees, and Kearns⁴⁰ for examples of how some of these weaknesses carry over to definitions that equalize false positive and negative rates.) Different statistical measures of fairness can be at odds with one another. For example, Chouldechova¹⁵ and Kleinberg⁴⁶ prove a fundamental impossibility result: except in trivial settings, it is impossible to simultaneously equalize false positive rates, false negative rates, and positive predictive value across protected groups. Learning subject to statistical fairness constraints can also be computationally hard,⁶¹ although practical algorithms of various sorts are known.^{1,29,63}

Individual definitions of fairness. Individual notions of fairness, on the other hand, ask for constraints that bind on specific pairs of individuals, rather than on a quantity that is averaged over groups. For example, Dwork¹⁹ gives a definition which roughly corresponds to the constraint that “similar individuals should be treated similarly,” where similarity is defined with respect to a task-specific metric that must be determined on a case by case basis. Joseph³⁵ suggests a definition that corresponds approximately to “less qualified individuals should not be favored over more qualified individuals,” where quality is de-

finied with respect to the true underlying label (unknown to the algorithm). However, although the semantics of these kinds of definitions can be more meaningful than statistical approaches to fairness, the major stumbling block is that they seem to require making significant assumptions. For example, the approach of Dwork¹⁹ presupposes the existence of an agreed upon similarity metric, whose definition would itself seemingly require solving a non-trivial problem in fairness, and the approach of Joseph³⁵ seems to require strong assumptions on the functional form of the relationship between features and labels in order to be usefully put into practice. These obstacles are serious enough that it remains unclear whether individual notions of fairness can be made practical—although attempting to bridge this gap is an important and ongoing research agenda.

Questions at the Research Frontier

Given the limitations of extant notions of fairness, is there a way to get some of the “best of both worlds?” In other words, constraints that are practically implementable without the need for making strong assumptions on the data or the knowledge of the algorithm designer, but which nevertheless provide more meaningful guarantees to individuals? Two recent papers, Kearns⁴⁰ and Hèbert-Johnson³⁰ (see also Kearns⁴² and Kim⁴⁴ for empirical evaluations of the algorithms proposed in these papers), attempt to do this by asking for statistical fairness definitions to hold not just on a small number of protected groups, but on an exponential or infinite class of groups defined by some class of functions of bounded complexity. This approach seems promising—because, ultimately, they are asking for statistical notions of fairness—the approaches proposed by these papers enjoy the benefits of statistical fairness: that no assumptions need be made about the data, nor is any external knowledge (like a fairness metric) needed. It also better addresses concerns about “intersectionality,” a term used to describe how different kinds of discrimination can compound and interact for individuals who fall at the intersection of

several protected classes.

At the same time, the approach raises a number of additional questions: What function classes are reasonable, and once one is decided upon (for example, conjunctions of protected attributes), what features should be “protected?” Should these only be attributes that are sensitive on their own, like race and gender, or might attributes that are innocuous on their own correspond to groups we wish to protect once we consider their intersection with protected attributes (for example clothing styles intersected with race or gender)? Finally, this family of approaches significantly mitigates some of the weaknesses of statistical notions of fairness by asking for the constraints to hold on average not just over a small number of coarsely defined groups, but over very finely defined groups as well. Ultimately, however, it inherits the weaknesses of statistical fairness as well, just on a more limited scale.

Another recent line of work aims to weaken the strongest assumption needed for the notion of individual fairness from Dwork:¹⁹ namely the algorithm designer has perfect knowledge of a “fairness metric.” Kim⁴⁵ assumes the algorithm has access to an oracle which can return an unbiased estimator for the distance between two randomly drawn individuals according to an unknown fairness metric, and show how to use this to ensure a statistical notion of fairness related to Hèbert-Johnson³⁰ and Kearns,⁴⁰ which informally state that “on average, individuals in two groups should be treated similarly if on average the individuals in the two groups are similar” and this can be achieved with respect to an exponentially or infinitely large set of groups. Similarly, Gillen²⁸ assumes the existence of an oracle, which can identify fairness violations when they are made in an online setting but cannot quantify the extent of the violation (with respect to the unknown metric). It is shown that when the metric is from a specific learnable family, this kind of feedback is sufficient to obtain an optimal regret bound to the best fair classifier while having only a bounded number of violations of the fairness metric. Rothblum⁵⁸ considers the case in which

the metric is known and show that a PAC-inspired approximate variant of metric fairness generalizes to new data drawn from the same underlying distribution. Ultimately, however, these approaches all assume fairness is perfectly defined with respect to some metric, and that there is some sort of direct access to it. Can these approaches be generalized to a more “agnostic” setting, in which fairness feedback is given by human beings who may not be responding in a way that is consistent with any metric?

Data evolution and dynamics of fairness. The vast majority of work in computer science on algorithmic fairness has focused on one-shot classification tasks. But real algorithmic systems consist of many different components combined together, and operate in complex environments that are dynamically changing, sometimes because of the actions of the learning algorithm itself. For the field to progress, we need to understand the dynamics of fairness in more complex systems.

Perhaps the simplest aspect of dynamics that remains poorly understood is how and when components that may individually satisfy notions of fairness compose into larger constructs that still satisfy fairness guarantees. For example, if the bidders in an advertising auction individually are fair with respect to their bidding decisions, when will the allocation of advertisements be fair, and when will it not? Bower⁸ and Dwork²⁰ have made a preliminary foray in this direction. These papers embark on a systematic study of fairness under composition and find that often the composition of multiple fair components will not satisfy any fairness constraint at all. Similarly, the individual components of a fair system may appear to be unfair in isolation. There are certain special settings, for example, the “filtering pipeline” scenario of Bower⁸—modeling a scenario in which a job applicant is selected only if she is selected at every stage of the pipeline—in which (multiplicative approximations of) statistical fairness notions compose in a well behaved way. But the high-level message from these works is that our current notions of fairness compose poorly. Experience

from differential privacy^{21,22} suggests that graceful degradation under composition is key to designing complicated algorithms satisfying desirable statistical properties, because it allows algorithm design and analysis to be modular. Thus, it seems important to find satisfying fairness definitions and richer frameworks that behave well under composition.

In dealing with socio-technical systems, it is also important to understand how algorithms dynamically effect their environment, and the incentives of human actors. For example, if the bar (for example, college admission) is lowered for a group of individuals, this might increase the average qualifications for this group over time because of at least two effects: a larger proportion of children in the next generation grow up in households with college educated parents (and the opportunities this provides), and the fact that a college education is achievable can incentivize effort to prepare academically. These kinds of effects are not considered when considering either statistical or individual notions of fairness in one-shot learning settings.

The economics literature on affirmative action has long considered such effects—although not with the specifics of machine learning in mind: see, for example, Becker,³ Coate,¹⁶ Foster.²⁶ More recently, there have been some preliminary attempts to model these kinds of effects in machine learning settings—for example, by modeling the environment as a Markov decision process,³² considering the equilibrium effects of imposing statistical definitions of fairness in a model of a labor market,³¹ specifying the functional relationship between classification outcomes and quality,⁴⁹ or by considering the effect of a classifier on a downstream Bayesian decision maker.³⁹ However, the specific predictions of most of the models of this sort are brittle to the specific modeling assumptions made—they point to the need to consider long term dynamics, but do not provide robust guidance for how to navigate them. More work is needed here.

Finally, decision making is often distributed between a large number of actors who share different goals

and do not necessarily coordinate. In settings like this, in which we do not have direct control over the decision-making process, it is important to think about how to incentivize rational agents to behave in a way that we view as fair. Kannan³⁷ takes a preliminary stab at this task, showing how to incentivize a particular notion of individual fairness in a simple, stylized setting, using small monetary payments. But how should this work for other notions of fairness, and in more complex settings? Can this be done by controlling the flow of information, rather than by making monetary payments (monetary payments might be distasteful in various fairness-relevant settings)? More work is needed here as well. Finally, Corbett-Davies¹⁷ take a welfare maximization view of fairness in classification and characterize the cost of imposing additional statistical fairness constraints as well. But this is done in a static environment. How would the conclusions change under a dynamic model?


Modeling and correcting bias in the data. Fairness concerns typically surface precisely in settings where the available training data is already contaminated by bias. The data itself is often a product of social and historical process that operated to the disadvantage of certain groups. When trained in such data, off-the-shelf machine learning techniques may reproduce, reinforce, and potentially exacerbate existing biases. Understanding how bias arises in the data, and how to correct for it, are fundamental challenges in the study of fairness in machine learning.

Bolukbasi⁷ demonstrate how machine learning can reproduce biases in their analysis of the popular word-2vec embedding trained on a corpus of Google News texts (parallel effects were independently discovered by Caliskan¹¹). The authors show that the trained embedding exhibit female/male gender stereotypes, learning that “doctor” is more similar to man than to woman, along with analogies such as “man is to computer programmer as woman is to homemaker.” Even if such learned associations accurately reflect patterns in the source text corpus, their use in automated systems may exacerbate existing bi-


ases. For instance, it might result in male applicants being ranked more highly than equally qualified female applicants in queries related to jobs that the embedding identifies as male-associated.

Similar risks arise whenever there is potential for feedback loops. These are situations where the trained machine learning model informs decisions that then affect the data collected for future iterations of the training process. Lum⁵¹ demonstrate how feedback loops might arise in predictive policing if arrest data were used to train the model.^b In a nutshell, since police are likely to make more arrests in more heavily policed areas, using arrest data to predict crime hotspots will disproportionately concentrate policing efforts on already over-policed communities. Expanding on this analysis, Ensign²⁴ finds that incorporating community-driven data, such as crime reporting, helps to attenuate the biasing feedback effects. The authors also propose a strategy for accounting for feedback by adjusting arrest counts for policing intensity. The success of the mitigation strategy, of course, depends on how well the simple theoretical model reflects the true relationships between crime intensity, policing, and arrests. Problematically, such relationships are often unknown, and are very difficult to infer from data. This situation is by no means specific to predictive policing.

Correcting for data bias generally seems to require knowledge of how the measurement process is biased, or judgments about properties the data would satisfy in an “unbiased” world. Friedler²⁷ formalize this as a disconnect between the *observed space*—features that are observed in the data, such as SAT scores—and the unobservable *construct space*—features that form the desired basis for decision making, such as intelligence. Within this framework, data correction efforts attempt to undo the effects of biasing mechanisms that drive discrepancies between these spaces. To the extent that the biasing



Fairness concerns typically surface precisely in settings where the available training data is already contaminated by bias.



mechanism cannot be inferred empirically, any correction effort must make explicit its underlying assumptions about this mechanism. What precisely is being assumed about the construct space? When can the mapping between the construct space and the observed space be learned and inverted? What form of fairness does the correction promote, and at what cost? The costs are often immediately realized, whereas the benefits are less tangible. We will directly observe reductions in prediction accuracy, but any gains hinge on a belief that the observed world is not one we should seek to replicate accurately in the first place. This is an area where tools from causality may offer a principled approach for drawing valid inference with respect to unobserved counterfactually ‘fair’ worlds.

Fair representations. Fair representation learning is a data debiasing process that produces transformations (intermediate representations) of the original data that retain as much of the task-relevant information as possible while removing information about sensitive or protected attributes. This is one approach to transforming biased observational data in which group membership may be inferred from other features, to a construct space where protected attributes are statistically independent of other features.

First introduced in the work of Zemel⁶⁴ fair representation learning produces a debiased data set that may in principle be used by other parties without any risk of disparate outcomes. Feldman²⁵ and McNamara⁵⁴ formalize this idea by showing how the disparate impact of a decision rule is bounded in terms of its balanced error rate as a predictor of the sensitive attribute.

Several recent papers have introduced new approaches for constructing fair representations. Feldman²⁵ propose rank-preserving procedures for repairing features to reduce or remove pairwise dependence with the protected attribute. Johndrow³³ build upon this work, introducing a likelihood-based approach that can additionally handle continuous protected attributes, discrete features, and which promotes joint independence


^b Predictive policing models are generally proprietary, and so it is not clear whether arrest data is used to train the model in any deployed system.

between the transformed features and the protected attributes. There is also a growing literature on using adversarial learning to achieve group fairness in the form of statistical parity or false positive/false negative rate balance.^{5,23,52,65}


Existing theory shows the fairness-promoting benefits of fair-representation learning rely critically on the extent to which existing associations between the transformed features and the protected characteristics are removed. Adversarial downstream users may be able to recover protected attribute information if their models are more powerful than those used initially to obfuscate the data. This presents a challenge both to the generators of fair representations as well as to auditors and regulators tasked with certifying that the resulting data is fair for use. More work is needed to understand the implications of fair representation learning for promoting fairness in the real world.

Beyond classification. Although the majority of the work on fairness in machine learning focuses on batch classification, it is but one aspect of how machine learning is used. Much of machine learning—for example, online learning, bandit learning, and reinforcement learning—focuses on dynamic settings in which the actions of the algorithm feed back into the data it observes. These dynamic settings capture many problems for which fairness is a concern. For example, lending, criminal recidivism prediction, and sequential drug trials are so-called bandit learning problems, in which the algorithm cannot observe data corresponding to counterfactuals. We cannot see whether someone not granted a loan would have paid it back. We cannot see whether an inmate not released on parole would have gone on to commit another crime. We cannot see how a patient would have responded to a different drug.

The theory of learning in bandit settings is well understood, and it is characterized by a need to trade-off exploration with exploitation. Rather than always making a myopically optimal decision, when counterfactuals cannot be observed, it is necessary for algorithms to sometimes take ac-



Much of machine learning focuses on dynamic settings in which the actions of the algorithm feed back into the data it observes. These dynamic settings capture many problems for which fairness is a concern.




tions that appear to be sub-optimal so as to gather more data. But in settings in which decisions correspond to individuals, this means sacrificing the well-being of a particular person for the potential benefit of future individuals. This can sometimes be unethical, and a source of unfairness.⁶ Several recent papers explore this issue. For example, Bastani² and Kannan³⁸ give conditions under which linear learners need not explore at all in bandit settings, thereby allowing for best-effort service to each arriving individual, obviating the tension between ethical treatment of individuals and learning. Raghavan⁵⁷ show the costs associated with exploration can be unfairly borne by a structured sub-population, and that counter-intuitively, those costs can actually increase when they are included with a majority population, even though more data increases the rate of learning overall. However, these results are all preliminary: they are restricted to settings in which the learner is learning a linear policy, and the data really is governed by a linear model. While illustrative, more work is needed to understand real-world learning in online settings, and the ethics of exploration.

There is also some work on fairness in machine learning in other settings—for example, ranking,¹² selection,^{42,47} personalization,¹³ bandit learning,^{34,50} human-classifier hybrid decision systems,⁵³ and reinforcement learning.^{18,32} But outside of classification, the literature is relatively sparse. This should be rectified, because there are interesting and important fairness issues that arise in other settings—especially when there are combinatorial constraints on the set of individuals that can be selected for a task, or when there is a temporal aspect to learning.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. 1136993. Any opinions, findings, conclusions, or recommendations expressed here are those of the authors and do not necessarily reflect the views of the National Science Foundation.

We are indebted to all of the participants of the CCC visioning work-

shop; discussions from that meeting shaped every aspect of this document. Also, our thanks to Helen Wright, Ann Drobnis, Cynthia Dwork, Sampath Kannan, Michael Kearns, Toni Pitassi, and Suresh Venkatasubramanian. 

References

- Agarwal, A., Beygelzimer, A., Dudík, M., Langford, J. and Wallach, H. A reductions approach to fair classification. In *Proceedings of the 35th Intern. Conf. Machine Learning*. ICML, JMLR Workshop and Conference Proceedings, 2018, 2569–2577.
- Bastani, H., Bayati, M. and Khorasani, K. Exploiting the natural exploration in contextual bandits. arXiv preprint, 2017, arXiv:1704.09011.
- Becker, G.S. *The Economics of Discrimination*. University of Chicago Press, 2010.
- Berk, R., Heidari, H., Jabbari, S., Kearns, M. and Roth, A. Fairness in criminal justice risk assessments: The state of the art. *Sociological Methods & Research* (2018), 0(0):0049124118782533.
- Beutel, A., Chen, J., Zhao, Z. and Chi, E.H. Data decisions and theoretical implications when adversarially learning fair representations. arXiv preprint, 2017, arXiv:1707.00075.
- Bird, S., Barocas, S., Crawford, K., Diaz, F. and Wallach, H. Exploring or exploiting? Social and ethical implications of autonomous experimentation in AI. In *Proceedings of Workshop on Fairness, Accountability, and Transparency in Machine Learning*. ACM, 2016.
- Bolukbasi, T., Chang, K-W., Zou, J.Y., Saligrama, V. and Kalai, A.T. Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. *Advances in Neural Information Processing Systems*, 2016, 4349–4357.
- Bower, A. et al. Fair pipelines. arXiv preprint, 2017, arXiv:1707.00391.
- Buolamwini, J. and Gebru, T. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Proceedings of the Conference on Fairness, Accountability and Transparency*. ACM, 2018, 77–91.
- Calders, T. and Verwer, S. Three naive Bayes approaches for discrimination-free classification. *Data Mining and Knowledge Discovery* 21, 2 (2010), 277–292.
- Caliskan, A., Bryson, J.J. and Narayanan, A. Semantics derived automatically from language corpora contain human-like biases. *Science* 356, 6334 (2017), 183–186.
- Celis, L.E., Straszak, D. and Vishnoi, N.K. Ranking with fairness constraints. In *Proceedings of the 45th Intern. Colloquium on Automata, Languages, and Programming*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.
- Celis, L.E. and Vishnoi, N.K. Fair personalization. arXiv preprint, 2017, arXiv:1707.02260.
- Chen, I., Johansson, F.D. and Sontag, D. Why is my classifier discriminatory? *Advances in Neural Information Processing Systems*, 2018, 3539–3550.
- Chouldechova, A. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big Data* 5, 2 (2017), 153–163.
- Coat, S. and Loury, G.C. Will affirmative-action policies eliminate negative stereotypes? *The American Economic Review*, 1993, 1220–1240.
- Corbett-Davies, S., Pierson, E., Feller, A., Goel, S. and Huq, A. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd ACM SIGKDD Intern. Conf. Knowledge Discovery and Data Mining*. ACM, 2017, 797–806.
- Doroudi, S., Thomas, P.S. and Brunskill, E. Importance sampling for fair policy selection. In *Proceedings of the 33rd Conference on Uncertainty in Artificial Intelligence*. AUAI Press, 2017.
- Dwork, C., Hardt, M., Pitassi, T., Reingold, O. and Zemel, R. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conf*. ACM, 2012, 214–226.
- Dwork, C. and Ilvento, C. Fairness under composition. Manuscript, 2018.
- Dwork, C., McSherry, F., Nissim, K. and Smith, A. Calibrating noise to sensitivity in private data analysis. In *Proceedings of Theory of Cryptography Conference*. Springer, 2006, 265–284.
- Dwork, C. and Roth, A. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science* 9, 3–4 (2014), 211–407.
- Edwards, H. and Storkey, A. Censoring representations with an adversary. arXiv preprint, 2015, arXiv:1511.05897.
- Ensign, D., Friedler, S.A., Neville, S., Scheidegger, C. and Venkatasubramanian, S. Runaway feedback loops in predictive policing. In *Proceedings of 1st Conf. Fairness, Accountability and Transparency in Computer Science*. ACM, 2018.
- Feldman, M., Friedler, S.A., Moeller, J., Scheidegger, C. and Venkatasubramanian, S. Certifying and removing disparate impact. *Proceedings of KDD*, 2015.
- Foster, D.P. and Vohra, R.A. An economic argument for affirmative action. *Rationality and Society* 4, 2 (1992), 176–188.
- Friedler, S.A., Scheidegger, C. and Venkatasubramanian, S. On the (im) possibility of fairness. arXiv preprint, 2016, arXiv:1609.07236.
- Gillen, S., Jung, C., Kearns, M. and Roth, A. Online learning with an unknown fairness metric. *Advances in Neural Information Processing Systems*, 2018.
- Hardt, M., Price, E. and Srebro, N. Equality of opportunity in supervised learning. *Advances in Neural Information Processing Systems*, 2016, 3315–3323.
- Hébert-Johnson, U., Kim, M.P., Reingold, O. and Rothblum, G.N. Calibration for the (computationally identifiable) masses. In *Proceedings of the 35th Intern. Conf. Machine Learning*. ICML, JMLR Workshop and Conference Proceedings, 2018, 2569–2577.
- Hu, L. and Chen, Y. A short-term intervention for long-term fairness in the labor market. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*. P.A. Champin, F.L. Gandon, M. Lalmas, and P.G. Ipeirotis, eds. ACM, 2018, 1389–1398.
- Ja bbari, S., Joseph, M., Kearns, M., Morgenstern, J.H. and Roth, A. Fairness in reinforcement learning. In *Proceedings of the Intern. Conf. Machine Learning*, 2017, 1617–1626.
- Johndrow, J.E., Lum, K. et al. An algorithm for removing sensitive information: application to race-independent recidivism prediction. *The Annals of Applied Statistics* 13, 1 (2019), 189–220.
- Joseph, M., Kearns, M., Morgenstern, J.H., Neel, S. and Roth, A. Fair algorithms for infinite and contextual bandits. In *Proceedings of AAAI/ACM Conference on AI, Ethics, and Society*, 2018.
- Joseph, M., Kearns, M., Morgenstern, J.H. and Roth, A. Fairness in learning: Classic and contextual bandits. *Advances in Neural Information Processing Systems*, 2016, 325–333.
- Kamishima, T., Akaho, S. and Sakuma, J. Fairness-aware learning through regularization approach. In *Proceedings of the IEEE 11th Intern. Conf. Data Mining Workshops*. IEEE, 2011, 643–650.
- Kannan, S. et al. Fairness incentives for myopic agents. In *Proceedings of the 2017 ACM Conference on Economics and Computation*. ACM, 2017, 369–386.
- Kannan, S., Morgenstern, J., Roth, A., Waggoner, B. and Wu, Z.S. A smoothed analysis of the greedy algorithm for the linear contextual bandit problem. *Advances in Neural Information Processing Systems*, 2018.
- Kannan, S., Roth, A. and Ziani, J. Downstream effects of affirmative action. In *Proceedings of the Conf. Fairness, Accountability, and Transparency*. ACM, 2019, 240–248.
- Kearns, M.J., Neel, S., Roth, A. and Wu, Z.S. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *Proceedings of the 35th International Conference on Machine Learning*. J.G. Dy and A. Krause, eds. JMLR Workshop and Conference Proceedings, ICML, 2018, 2569–2577.
- Kearns, M., Neel, S., Roth, A. and Wu, Z.S. An empirical study of rich subgroup fairness for machine learning. In *Proceedings of the Conf. Fairness, Accountability, and Transparency*. ACM, 2019, 100–109.
- Kearns, M., Roth, A. and Wu, Z.S. Meritocratic fairness for cross-population selection. In *Proceedings of International Conference on Machine Learning*, 2017, 1828–1836.
- Kilbertus, N. et al. Avoiding discrimination through causal reasoning. *Advances in Neural Information Processing Systems*, 2017, 656–666.
- Kim, M.P., Ghorbani, A. and Zou, J. Multiaccuracy: Blackbox postprocessing for fairness in classification. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*. ACM, 2019, 247–254.
- Kim, M.P., Reingold, O. and Rothblum, G.N. Fairness through computationally bounded awareness. *Advances in Neural Information Processing Systems*, 2018.
- Kleinberg, J.M., Mullainathan, S. and Raghavan, M. Inherent trade-offs in the fair determination of risk scores. In *Proceedings of the 8th Innovations in Theoretical Computer Science Conference*, 2017.
- Kleinberg, J. and Raghavan, M. Selection problems in the presence of implicit bias. In *Proceedings of the 9th Innovations in Theoretical Computer Science Conference* 94, 2018, 33. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.
- Kusner, M.J., Loftus, J., Russell, C. and Silva, R. Counterfactual fairness. *Advances in Neural Information Processing Systems*, 2017, 4069–4079.
- Liu, L.T., Dean, S., Rolf, E., Simchowitz, M. and Hardt, M. Delayed impact of fair machine learning. In *Proceedings of the 35th Intern. Conf. Machine Learning*. ICML, 2018.
- Liu, Y., Radanovic, G., Dimitrakakis, C., Mandal, D. and Parkes, D.C. Calibrated fairness in bandits. arXiv preprint, 2017, arXiv:1707.01875.
- Lum, K. and Isaac, W. To predict and serve? *Significance* 13, 5 (2016), 14–19.
- Madras, D., Creager, E., Pitassi, T. and Zemel, R. Learning adversarially fair and transferable representations. In *Proceedings of Intern. Conf. Machine Learning*, 2018, 3381–3390.
- Madras, D., Pitassi, T. and Zemel, R.S. Predict responsibly: Increasing fairness by learning to defer. *CoRR*, 2017, abs/1711.06664.
- McNamara, D., Ong, C.S. and Williamson, R.C. Provably fair representations. arXiv preprint, 2017, arXiv:1710.04394.
- Nabi, R. and Shpitser, I. Fair inference on outcomes. In *Proceedings of the AAAI Conference on Artificial Intelligence* 2018 (2018), 1931. NIH Public Access.
- Pedreshi, D., Ruggieri, S. and Turini, F. Discrimination-aware data mining. In *Proceedings of the 14th ACM SIGKDD Intern. Conf. Knowledge Discovery and Data Mining*. ACM, 2008, 560–568.
- Raghavan, M., Slivkins, A., Wortman Vaughan, J. and Wu, Z.S. The unfair externalities of exploration. *Conference on Learning Theory*, 2018.
- Rothblum, G.N. and Yona, G. Probably approximately metric-fair learning. In *Proceedings of the 35th Intern. Conf. Machine Learning*. JMLR Workshop and Conference Proceedings, ICML 80 (2018), 2569–2577.
- Rothwell, J. How the war on drugs damages black social mobility. The Brookings Institution, Sept. 30, 2014.
- Sweeney, L. Discrimination in online ad delivery. *Queue* 11, 3 (2013), 10.
- Woodworth, B., Gunasekar, S., Ohannessian, M.I. and Srebro, N. Learning non-discriminatory predictors. In *Proceedings of Conf. Learning Theory*, 2017, 1920–1953.
- Yang, K. and Stoyanovich, J. Measuring fairness in ranked outputs. In *Proceedings of the 29th Intern. Conf. Scientific and Statistical Database Management*. ACM, 2017, 22.
- Zafar, M.B., Valera, I., Gomez-Rodriguez, M. and Gummadi, K.P. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th Intern. Conf. World Wide Web*. ACM, 2017, 1171–1180.
- Zemel, R., Wu, Y., Swersky, K., Pitassi, T. and Dwork, C. Learning fair representations. In *Proceedings of ICML*, 2013.
- Zhang, B.H., Lemoine, B. and Mitchell, M. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conf. AI, Ethics, and Society*. ACM, 2018, 335–340.

Alexandra Chouldechova (achould@cmu.edu) is Estella Loomis Assistant Professor of Statistics and Public Policy in the Heinz College at Carnegie Mellon University, Pittsburgh, PA, USA.

Aaron Roth (aaroth@cis.upenn.edu) is Class of 1940 Associate Professor in the Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA, USA. Together with Michael Kearns, he is the author of *The Ethical Algorithm*.

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/frontiers-of-fairness>

Diverse examples depict how indistinguishability plays a central role in computer science.

BY HAGIT ATTIYA AND SERGIO RAJSBAUM

Indistinguishability

The properties commonly ascribed to any object are, in last analysis, names for its behavior.

—Judson Herrick, *An Introduction to Neurology*, 1918

Dost thou love me? I know thou wilt say “ay,”
And I will take thy word. Yet if thou swear’st
Thou mayst prove false. At lovers’ perjuries,
They say, Jove laughs.

—Shakespeare’s *Romeo and Juliet*, Act 2

ABSTRACTION—ALLOWING THE details of lower-level components to be ignored—and *interaction*—allowing individual computing entities to cooperate—are key concepts in computer science. Many would argue that they play a crucial role in the success of computing: abstraction allows separate layers of the computing stack to be improved orthogonally, whereas interaction allows the abundance of computing power to be harnessed. This comes at a significant cost: each component of a computer system has limited knowledge about the state of other components. This happens either by choice, in the case of abstraction, or out of necessity, in the case of interaction.

From the perspective of an individual component, all other components, either other layers within the same computing entity or other computing entities,

can be considered as an *environment*. Seen in this way, lack of knowledge about other components can formally be captured through the concept of *indistinguishability*, namely inability to tell apart different behaviors or states of the environment. Indistinguishability is therefore a consequence of the fact that computer systems are built of individual components, each with its own perspective of the system.

This article argues that because of its intimate relation with key issues in computing, indistinguishability, in its various flavors, plays a critical role in many computing areas. We explain this core concept and demonstrate some of its variants and applications, through four examples, trying to illustrate different, fundamental aspects of indistinguishable situations of abstraction and interaction.

Indistinguishability is at the core of the difficulty of constructing theoretical models for the behavior of a physical system. In our first example, we overview the role of indistinguishability in some of the most basic notions in computer science: state, automata, and learning. We will encounter both interaction (as means to reduce indistinguishability) and abstraction (captured by behavioral equivalence). Here, the environment is seen as a blackbox, implemented

» key insights

- **Lack of knowledge by a computer system component about other components can formally be captured through the concept of indistinguishability. Whenever abstraction or interaction take place in a computer system, indistinguishability plays a critical role.**
- **Indistinguishability is the source of many lower bounds and impossibility results in CS. It is also the essence behind abstraction techniques so important in computing theory and in the design of large complex systems.**
- **Indistinguishability has a topological nature: local states of components that do not distinguish between two system states induce a higher-dimensional simplicial complex, a combinatorial structure with topological properties preserved as the system execution evolves.**



IMAGE BY GLUIJKI

by an unknown automaton. What can an experimenter interacting with its environment through input/output symbols infer about the blackbox internals? The experimenter has an evolving mental model of the blackbox as an hypothesis automaton, which is indistinguishable from the actual automaton, given the current state of the interaction. The very notion of “state” is in terms of indistinguishability. In this example, indistinguishability has a “semantic” nature, related to computational complexity, namely the number of states in the automaton and the complexity of the learning algorithm.

Our second example demonstrates that indistinguishability is a powerful tool for deriving positive results. Examples abound, such as in artificial intelligence (for example, Turing’s test), cryptography (for example, pseudo-randomness), logic, and others. We consider the example of serializability in concurrent programming, where interaction is through shared variables, and locks permit the set of indistinguishable executions to be reduced. The correctness specification of a program is in terms of requiring that concurrent executions are indistinguishable from appropriate sequential executions. Abstraction is key, and indistinguishability becomes a powerful tool to

design concurrent programs and prove their correctness, and in particular, to enable sequential reasoning.

We move in our third example to another very basic form of indistinguishability, related to time, and to the impossibility of observing real-time. An interaction among a set of computing entities can be seen as a partial order, representing causality relations between events happening in the system. Lamport’s seminal paper²⁶ can be seen as using indistinguishability in two senses. First, it observed the relation to relativity theory, motivating the idea of reducing concurrent systems by indistin-

guishability to sequential thinking (by implementing a fault-tolerant distributed system as a replicated state machine). And second, it provided the framework for analyzing time-based algorithms, which depend on quantifying real-time indistinguishability. We illustrate this with a simple example showing how inherent limitations on clock synchronization can be derived through the impossibility of distinguishing the real-time occurrence of events in an execution up to given bounds on message transmission delays and clock drifts.

Prior examples consider a *single* execution, and analyze a set of executions that are indistinguishable from it, from the perspective of all the participating processes. Our final example considers how distributed computation is limited by the global indistinguishability structure of *all* possible executions. This structure is defined by a Kripke graph, where edges are labeled by processes that do not distinguish between the global states of the system represented by the two endpoints of the edge. It turns out that higher dimensional topological properties of this graph (more precisely, its dual, a simplicial complex) determine computability and the amount of interaction needed to distributively solve a problem.

Automata and Learning

We start with a simple scenario where a *learner* is trying to infer the internal construction of a *blackbox*. The learner knows that the blackbox is a *deterministic finite automaton* (DFA) accepting a language over an alphabet Σ , but does not know which specific automaton it is. Through a conversation, the learner and the blackbox exchange symbols, and there is a set of automata all indistinguishable with respect to the current conversation. As the interaction evolves, this set of indistinguishable automata shrinks. Eventually, the learner would like it to shrink until it captures the language accepted by the blackbox.

Moore's theorem. Indistinguishability is at the core of the difficulty of constructing theoretical models for the behavior of a physical system. Ashby's Cybernetics book³ from 1956 already includes a chapter called "The blackbox." At the same time, Moore¹² pro-

Indistinguishability is at the core of the difficulty of constructing theoretical models for the behavior of a physical system.

posed the problem of learning finite automata, and studied indistinguishability of deterministic finite state machines, stating (Theorem 2):

"Given any machine S and any multiple experiments performed on S , there exist other machines experimentally distinguishable from S for which the original experiment would have had the same outcome."

Moore's theorem shows an impossibility in the characterization of any physical system as a deterministic state machine on the basis of a finite number of observational outcomes. This is because after a finite interaction with the blackbox, approximately, if all words are at most of length k , the learner has explored only paths of length k in the automaton A of the blackbox.

This does not prevent the construction of theoretical models of the behavior of a system, but it does challenge the assumption that a system has *only* the behaviors that have been characterized by experimental observations, namely the assumption that any theoretical model is complete. Further discussion of the relation between Moore's theorem and physics appears in Fields.¹⁶

The Myhill–Nerode theorem. If the interaction with the blackbox is only through input/output symbols, how can the learner know anything at all about its internal construction, even if it has any states at all? States are not directly observable, so what *is* a state, from the perspective of the learner? The Myhill–Nerode theorem, "one of the conceptual gems of theoretical computer science" according to Rosenberg,³³ offers a complete mathematical characterization of the notion of state, via basic algebraic properties defined only on input/output behavior.

A string $t \in \Sigma^*$ *distinguishes* two strings u and v in a language L , if $ut \in L$ and $vt \notin L$. If there is a string t distinguishing u and v , then the state $s = \delta(q_0, u)$ must be different from the state $s' = \delta(q_0, v)$, for any automaton M with transition function δ , recognizing L . Conversely, two strings x and y are *indistinguishable* (by L) if there is no string $t \in \Sigma^*$ that distinguishes them. We have the equivalence *Nerode congruence* on Σ^* , defined by

$$x \equiv_L y \text{ iff } \forall z \in \Sigma^*, xz \in L \Leftrightarrow yz \in L.$$

Let $[s]_L$ be the set of all strings that are indistinguishable from s , and Q be the set of all corresponding equivalence classes. Thus, the essence of the notion of “state” is an indistinguishability equivalence class; define a DFA Z as follows:

- ▶ the states Q are the equivalence classes of \equiv_L ,
- ▶ the initial state q_0 is $[\epsilon]_L$, the equivalence class of the empty word,
- ▶ $\delta([u]_L, a) = [ua]_L$ for all $[u]_L \in Q$ and $a \in \Sigma$, and
- ▶ the accepting states are $F = \{[u]_L : u \in L\}$.

Selecting a representative for each equivalence class of \equiv_L , we get a set of *access strings* $\mathcal{S} \subset \Sigma^*$. Starting in the initial state, if we follow the transitions as indicated by $u \in \mathcal{S}$, it leads us to a state q that is uniquely identified by u . Figure 1 depicts an example of a DFA \mathcal{A} , and then it is explicitly represented by access strings as \mathcal{H}_2 .

The Myhill-Nerode theorem states that L is recognized by Z as defined earlier, and furthermore, Z is minimal: if a DFA M accepts L , then the equivalence relation \equiv_M is a refinement of the equivalence relation \equiv_L , where

$$x \equiv_M y \text{ iff } \delta(q_0, x) = \delta(q_0, y),$$

and we say that x and y are *indistinguishable to M* .

Proofs that a given language cannot be recognized by a finite automaton can be viewed as indistinguishability arguments, based on the Myhill-Nerode theorem. Automata with infinitely many states can be viewed as abstractions of programs that can make infinitely many discriminations regarding the structure of a set of possible input strings.

Let $\lambda_q(v) = 1$ whenever Z accepts $v \in \Sigma^*$ starting at state q , and $\lambda_q(v) = 0$ otherwise. If $q = q_0$, we may omit the sub-index, that is, $L = \{w : \lambda(w) = 1\}$. For learning, we will use the notion of a string t being a *witness* that two states are different. Notice that:

- ▶ For any pair of distinct states q, q' of Z , there is a distinguishing word $t \in \Sigma^*$ such that $\lambda_q(t) \neq \lambda_{q'}(t)$.

Learning automata. Following the classic approach of *learning finite automata*,³⁶ three additional approaches have been studied: *computational learning*,²⁵ *model learning*,³⁷ and *grammatical inference*.³⁴ We next describe automata learning algorithms with a *minimally adequate teacher* (MAT), demonstrating fundamental ideas that are relevant to all four learning branches.

Minimization algorithms related to the Myhill-Nerode theorem work by *merging* indistinguishable states of a DFA. We describe algorithms working in the opposite direction, *splitting* states when discovering a witness string t demonstrating they are distinguishable.

The learner poses membership queries to the blackbox to try to learn the language L it accepts: Does $x \in \Sigma^*$ belong to L ? The learner starts with a hypothesis automaton H , that it updates during the conversation. The experimenter has no way of knowing when to stop asking questions, because there could be machines with more and more states, which return answers consistent with the current experiment. Even if the number of states of M is known to the experimenter, an exponential number of membership queries is required.² To circumvent this, the MAT framework admits *equivalence queries*:

- ▶ Does H correctly recognize L ? If not, give me an example of a string $x \in \Sigma^*$ such that $x \in L(H) - L(M)$ or $x \in L(M) - L(H)$.

Using membership and equivalence queries, the experimenter can learn L with a number of queries that is polynomial in n , the number of states in Z , the Myhill-Nerode automaton for L , and in m the longest counterexample returned

by the blackbox. (There are always counterexamples of length at most $2n$.) The algorithm terminates with a DFA H that is isomorphic to Z . The MAT framework and the efficient algorithm, called L^* , were introduced in a seminal paper of Angluin.¹ We stress that this kind of learning algorithms can be extended to learn other types of blackboxes, for example, logical formulas.

We illustrate the ideas behind the MAT framework through an example (inspired by Isberner et al.²⁴), to show how *distinguishing* is the basis of learning. Learning something new means splitting a state into two states (which are different, as evidenced by a new witness t).

Assume the blackbox is implemented by the DFA A in Figure 1. The learner maintains a set of prefix-closed access strings $\mathcal{S} \subset \Sigma^*$; recall that access strings are representatives of equivalence classes. Distinct access strings u, u' correspond to distinct states of A that the learner has identified, and the learner has a witness of this fact, through a string t , such that $\lambda(u \cdot t) \neq \lambda(u' \cdot t)$. The learner maintains this set of *discriminating suffixes* $D \subset \Sigma^*$, that it has found through membership queries.

The basic data structure is the *observation table*, with two types of rows (in the figure, a horizontal line in a table divides the two types). Each row of the first type is identified by an access string $u \in \mathcal{S}$, and each row of the second type identifies a transition of the hypothesis automaton. Each column is identified by a discriminating string t . The content of a cell in the table is $\lambda(u \cdot t)$ (where λ refers to the current hypothesis automaton). Each time the learner gets a counterexample, it extracts from it a discriminating suffix. Many algorithms have been proposed, differing in how they extract a discrim-

Figure 1. Learning example.

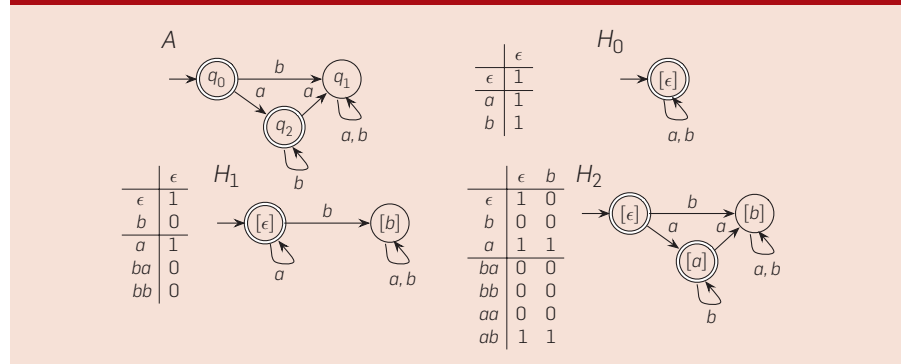
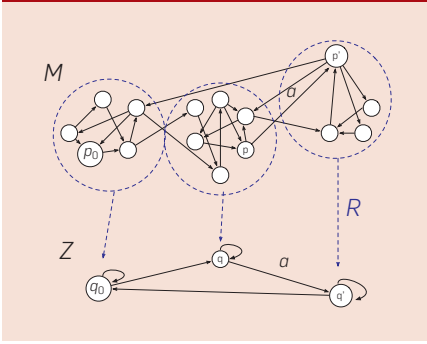
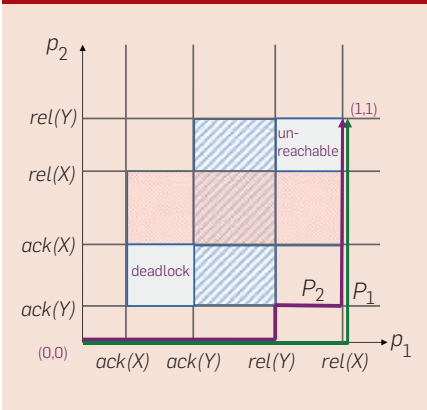


Figure 2. Schematic illustration of bisimulation.

Figure 3. Geometric interpretation of all interleavings of two processes acquiring and releasing shared variables X, Y.


inating suffix from a counterexample. Here we are only concerned with the fact that it is always possible to do so.

The learner initially has as hypothesis the DFA H_0 . It then learns that ϵ is discriminating ϵ and b , and hence splits state $[\epsilon]$ creating state $[b]$. In the table, the new row for access string b is added, and the transition for b is replaced by the two transitions ba , bb . Thus, the new hypothesis automaton is H_1 , and by following string b in this automaton, one “accesses” state $[b]$, an equivalence class of strings indistinguishable from the representative of the class, b (for example, aab also belongs to $[b]$; it is indistinguishable from b and also accesses $[b]$). In H_1 , we have a (single) column identified by ϵ , witnessing that states $[\epsilon]$ and $[b]$ are different, because ϵ concatenated with ϵ is in L , whereas ϵ concatenated with b is not. Then, H_2 is produced when it learns that b discriminates ϵ and a , $\lambda(\epsilon \cdot b) \neq \lambda(a \cdot b)$, and hence the state $[\epsilon]$ is split creating the state $[a]$. More generally, if w is a counterexample for H , then it has a suffix at , s.t. for two access strings $u, u' \in U$, ua and u'

reach the same state in H , but $\lambda(ua \cdot t) \neq \lambda(u't)$. Thus, $u' \in U$, ua is a transition in the observation table, and both rows are equal, and adding t to table distinguishes ua and u' , with ua being moved to the upper part of table.

Behavioral equivalences. Behavioral equivalences¹⁷ are based on the idea that two systems are equivalent whenever no external observation can distinguish between them. They are used to *abstract* from unwanted details; to formalize the idea that it is not the internal structure of a system which is of interest but its behavior with respect to the outside world.

Bisimulation, the strongest form, is a rich concept independently discovered in computer science, modal logic, and set theory, with applications to many areas,³⁵ and we would have devoted much more space to it if it was not for lack of space. We touched on it, with the Myhill-Nerode theorem example, which is the basis for automata minimization algorithms modulo bisimilarity.²³ Another typical application is to prove the correctness of an algorithm, with a big automaton representation M , by analyzing a smaller bisimilar model Z that captures its essence, as illustrated in Figure 2, where R is the bisimilar relation between states of Z and M . Intuitively, two systems are *bisimilar* if they match each other’s moves. Verifying the algorithm M using a model checking problem $M \models \phi$ is equivalent to solving the much smaller problem $Z \models \phi$. From the indistinguishability perspective, it is interesting to consider iterative abstraction-refinement, see Clarke et al.⁹

Sequential Reductions in Concurrent Programming

A notable example of behavioral equivalence is the notion of *serializability*, utilized in most of the database systems (in various variants) since their early days in the 1970s. The notion is used in concurrency control of databases and in various transactional systems (processing, management, transactional memory, etc.), both centralized and distributed. A key challenge in the design and analysis of concurrent systems is dealing with all possible interleavings of concurrent processes. Indistinguishability is useful for defining the semantics of a concurrent program, in terms of the notion of serializability. It is also impor-

tant in verification, as it can be exploited to verify a concurrent program by checking only its sequential executions.³

Serializability and two-phase locking. Serializability is studied in a setting where processes interact through shared variables. Two executions α_1 and α_2 are *indistinguishable to a specific process*, if the process accesses the same sequence of variables in both executions, and returns the same results. An execution is *serializable*^{8,39} if it is indistinguishable to all processes from a *sequential* execution, in which each process executes its procedure invocation to completion, without interleaving of any other process.

The classic way to ensure serializability is to protect shared variables with locks, using a locking protocol governing how locks are acquired and released. Thus, an execution of the system, α , is a sequence of *events* each taken by a single process; the events either access shared variables, or acquire and release locks on these shared variables. In *two-phase locking (2PL)*,¹³ each process has a *growing* phase of lock acquisition (in some order), followed by a *shrinking* phase of lock release. Namely, once a process released a lock, it can no longer acquire any lock, even on another variable. For example, given shared variables X, Y , and two processes p_1, p_2 :

$$p_1: acq(X); access(X); acq(Y); access(Y); rel(Y); rel(X)$$

$$p_2: acq(Y); access(Y); acq(X); access(X); rel(X); rel(Y)$$

Two-phase locking is a mechanism for enforcing indistinguishability from sequential executions, as demonstrated by the following geometric interpretation. An execution of the processes p_1, p_2 defines a particular interleaving of the order in which the processes acquire and release the locks. It can be represented as a path in a two-dimensional space (see Figure 3). If a lock is acquired or released by p_1 , the path moves one unit on the horizontal axis; similarly, when a lock is acquired or released by p_2 , the path moves one unit on the vertical axis. All paths start in $(0, 0)$, when no operations have occurred, and they all

a Indistinguishability is central also in other consistency conditions like linearizability.

end in (1, 1), where all operations have occurred, by both processes.

Each time two operations of an execution are swapped, in a way that is indistinguishable to both processes, the path is deformed. In Figure 3, two such paths are illustrated: P_1 which is sequential (p_1 then p_2), and P_2 where $acq(Y)$ by p_2 is swapped with $rel(X)$ by p_1 .

There are two forbidden rectangles, where no execution path can go through: in the vertical (blue) one, Y would be acquired simultaneously by both, whereas in the horizontal rectangle (red), the same holds for X . Their union is the forbidden region where no execution enters. Notice that if both processes acquire X and Y (in either order), the protocol enters the deadlock region. The main point is that there are two classes C_1, C_2 , of *homotopic* paths, that is, paths within a class can be deformed to each other. In one class, all paths go above the forbidden region and are indistinguishable from a sequential execution in which p_2 goes first, whereas in the other class, all executions go below the forbidden region and are indistinguishable from a sequential execution where p_1 goes first.

Notice that in a program where both processes acquire the locks in the same order, the forbidden region is a square, and hence no deadlocks can happen. Directed topology and the geometric theory of execution paths homotopy are studied in Fajstrup et al.,¹⁵ showing a direct representation of indistinguishability as continuous deformation of paths in an n -dimensional space (for n processes).

Verifying two-phase locking. Because indistinguishable executions can be substituted for each other, it means that checking whether one execution satisfies a particular property informs us whether all indistinguishable executions satisfy this property. Therefore, indistinguishability facilitates the verification of concurrent programs. When a program is serializable certain properties can be verified by considering only sequential (noninterleaved) executions of the program. This is equivalent to reasoning assuming a sequential setting.

But how can we prove that a program is serializable? Obviously, if we

prove that it follows the two-phase locking protocol, then it is serializable. However, in reality, we are not given an execution example, but a program, possibly including conditional and repeat statements. Thus, we need to consider all its possible executions, to see if each one satisfies the two-phase locking regime. It turns out that we can ensure that the program follows 2PL, by considering only its sequential executions. The next theorem holds provided the program has no nonterminating loops.

THEOREM 3.1. *If any execution satisfies two-phase locking when events of different processes are not interleaved, then any interleaved execution also satisfies two-phase locking.*

Proving the theorem goes through showing that every execution that violates 2PL is indistinguishable from a noninterleaved execution in which the protocol is also violated. This implies that if we check (manually or mechanically) all noninterleaved executions of the protocol without finding a violation of 2PL, then all executions of the protocol do not violate 2PL.

Toward a contradiction, assume the claim does not hold and let $\alpha = \alpha'(i, e)$ be the shortest execution that violates 2PL for which there is no indistinguishable noninterleaved execution; see Figure 4. Note that (i, e) is an event of process p_i that violates 2PL, that is, acquires a lock after releasing a lock, or accesses an unlocked location. As α is the shortest such execution, we know that for prefix α' of α there is an indistinguishable noninterleaved execution $\bar{\alpha}' = \alpha_{i_1} \alpha_{i_2} \dots \alpha_{i_n}$ (where α_{i_j} contains events by p_{i_j} only).

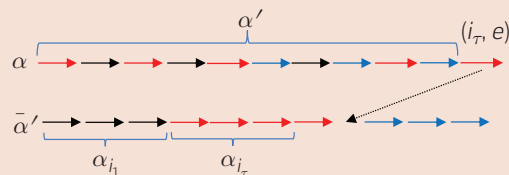
We argue that moving the event (i, e) to after p_i 's events in $\bar{\alpha}'$, will still cause p_i to take the offending event. Intuitively, this happens because the event depends only on information that is local to the process p_i or

locked by it, and p_i does not distinguish between the original execution and the noninterleaved execution. Namely, p_i has the same state at the end of α and at the end of $\alpha_{i_1} \dots \alpha_{i_n}$. Therefore, the event can be moved to appear after the events α_{i_j} of the same process. Hence, p_i will make the same offending event (i, e) , implying that the noninterleaved execution $\alpha_{i_1} \dots \alpha_{i_n}(i, e)$, also violates 2PL.

The reduction holds for any noncentralized locking protocol, such as commonly used ones like *two-phase, hand-over-hand, tree, and dynamic graph* locking. It allows *sequential reasoning*, whether manual or automated, about concurrent programs both in verifying that they adhere to a locking protocol and in development of algorithms for them. The reduction enables simpler and more efficient verification algorithms of a class of properties, called *transaction-local*. It justifies the use of sequential Hoare Logic or sequential type systems or sequential abstract interpretation to verify that the program adheres to a locking protocol. Programmers wishing to add, for example, a new procedure to swap two adjacent elements in a list to a program that uses hand-over-hand locking, do not have to worry about concurrent interleaving with other procedures. More details are in Attiya et al.,⁶ such as the case of nonterminating loops.

Indistinguishability is also used to prove a theorem that shows that if serializability is ensured in a program with two processes and two variables, it is ensured in any program, provided the implementation satisfies certain structural properties, one of them being symmetry.¹⁹ The proof goes by contradiction, taking an execution of the larger system that violates serializability and perturbing it into a bad execution for a system with two processes and two variables; a key step relies on an indistinguishability argument using symmetry.

Figure 4. Moving the event (i, e) to after p_i 's events.



Real-Time Indistinguishability

The previous examples describe asymmetric interactions, where one party interacts with another party, whose semantics (internal details) are hidden or abstracted away. Our next example ignores the semantics of the interactions, concentrating only on their timing.

The fundamental problem is estimating *distant simultaneity*—the time difference between the occurrence of two spatially separated (at different processes) events. This is behind many real-time applications in computer science that depend on clock synchronization, such as synchronizing cellphone communications, positioning systems (for example, GPS), failure detection, efficient use of resources (for example, releasing a connection), timestamping events and timeouts, and so on.

Computer clocks are typically based on inexpensive oscillator circuits and quartz crystals that can easily drift seconds per day. However, atomic clock time, so ubiquitous and integral to modern life, trickles down to the clocks we use daily, distributed through the Network Time Protocol and other means. Atomic clocks are so precise that if such a clock existed when Earth began, about 4.5 billion years ago, it would be off by only 30s today.

How precise the time of an atomic clock can be estimated depends on the transmission delay bounds along communication paths from the atomic

clock to the local computer, and on the drift bounds of the clocks of the computers along such paths. In other words, when a computer gets a message with the time of some atomic clock, the actual moment when the clock reading took place could have occurred at any moment within some range B , and from the computer's perspective, it is indistinguishable which exact moment within B is the actual one. Thus, the computer's best estimate of the atomic clock time is based on $|B|/2$. Indeed, selecting the mid-point is hedging the bets, because anything else leaves open the possibility of a bigger mistake. We now explain in more detail how to compute B .

Consider a process p_1 trying to synchronize its clock with an atomic reference clock, assumed to give real-time exactly, located in p_0 . The basic interaction is when p_1 has a direct link to p_0 , as illustrated in Figure 5. Process p_1 sends a message to p_0 and gets back a response. The send event e_1^1 by p_1 occurs at real-time 1, the event of p_0 receiving it, e_1^0 , occurs at real-time 6 (to simplify the example, we assume p_0 responds immediately, in the same event), and p_1 receives the response in event e_2^1 at real-time 12. Real-time is not directly observable, instead, each event occurs at some local time, which the process can observe. The precise meaning of real-time not being observable is through indistinguishability. Namely, suppose that, although the first

message delay was 5 time units, it is known that it must have taken at least 4 time units; also, assume the return message cannot take more than 9 time units. As for the local clock of p_1 , suppose its drift is bounded, such that between the sending and the receiving events, at most 12 time units could have passed.

What is the *latest* time that e_2^1 could have occurred with respect to e_1^0 ? Answering also what is the *earliest* it could have occurred, would yield the desired *indistinguishability interval* B , where e_2^1 could have occurred, and selecting the midpoint would be used to compute the optimal correction to the local clock time of p_1 . The crucial insight is that to compute how late e_2^1 could have occurred with respect to e_1^0 we have to shift to the right as much as possible the point of occurrence of e_2^1 , subject to two constraints: (1) the maximum delay of the second message (9 units) and (2) the minimum delay of the first message plus the minimum length of the time interval from e_1^1 to e_2^1 (the fastest that p_1 's clock could have been running). In the example, the latest that e_2^1 can happen is at real-time 14 determined by the fastest delay of the first message and the slowest clock drift of p_1 , and not by the largest delay of the second message (which could have been delivered at 15).

More generally, p_1 may be further away from the process p_0 with an atomic reference clock, and an arbitrary execution α is used to synchronize p_1 's clock, where many more message exchanges take place, along different paths between p_1 and p_0 . The goal is to estimate the indistinguishability interval of an event e at process p_1 , with respect to an event e_0 in p_0 . The previous example hints that the task at hand has to do with computing distances, on paths formed by indistinguishability intervals, formalized as follows.

The execution α is represented by a weighted directed graph $G = (V, E, r, \ell)$. Each vertex of V is an event of α , either a *send* or a *receive* event. The i th event happening in process j is denoted $e_i^j \in V$. The directed edges E are causal relationships: there is a directed edge between two consecutive

Figure 5. p_0 sends and p_1 responds.

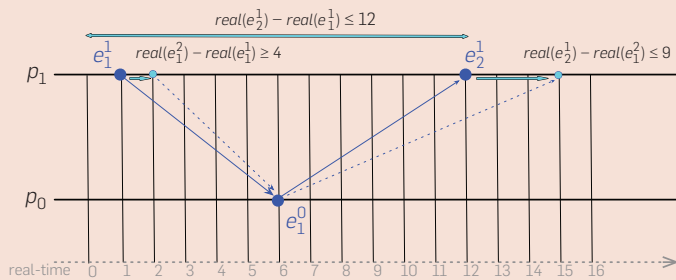
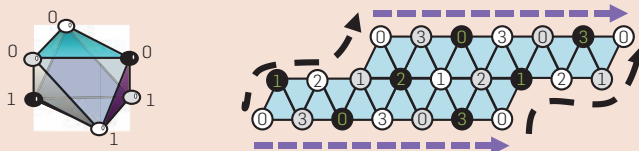


Figure 6. Consensus and renaming input complexes.



events in the same process, $e_i^j \rightarrow e_{i+1}^j$, and there is a directed edge $e_i^j \rightarrow e_k^\ell$, whenever e_i^j is a send event and e_k^ℓ is the corresponding receive event. The weight functions r, ℓ timestamp the events. For each $e \in V$, $real(e)$ is the real-time occurrence of event e , and $local(e)$ is the time according to the clock of the process where e happens. Since the clock of p_0 is perfect, for all events e_i^0 in p_0 , we have $real(e_i^0) = local(e_i^0)$.

For each pair of events e_1, e_2 joined (in either direction) by a directed edge of G , bounds on the relative real-time occurrence of two events can be estimated,

$$-B(e_2, e_1) \leq real(e_1) - real(e_2) \leq B(e_1, e_2),$$

both when the edge represents a message transmission delay, and when it represents the time it takes a process to execute consecutive computational events. Then, define $local(e_1, e_2) = local(e_1) - local(e_2)$, and let $w(e_1, e_2) = B(e_1, e_2) - local(e_1, e_2)$. These weights w can be positive or negative, but summing them along a cycle always gives a nonnegative value (the telescopic sum of $local(e_i, e_{i+1})$ along a cycle is 0). Thus, for a pair of events e_1 and e_2 , the distance $d(e_1, e_2)$ with respect to these weights is well defined. Interestingly, observe that $d(e, e') = 0$, for any two events in p_0 . It is not hard to show³⁰ that the indistinguishability interval of an event e at some process p_1 , with respect to an event e_0 in p_0 is as follows.

THEOREM 4.1. $real(e) \in [-d(e_0, e), d(e, e_0)]$

The meaning of this theorem is that e might have occurred at any time in this interval. Furthermore, for each such time, there is an execution indistinguishable to all processes.

These results are based on Patt-Shamir and Rajsbaum,³⁰ a follow up of,^{5,20} which studied how closely in terms of real-time processes can be guaranteed to perform a particular action, in a failure-free environment. The possibility of failures affects the size of the indistinguishability interval, providing a very interesting topic from the indistinguishability perspective. The standard technique is to consider several clock reference values, and taking the average after disregarding the most extreme values. There

Indistinguishability is useful for defining the semantics of a concurrent program in terms of the notion of serializability.

are many papers on clock synchronization algorithms, see, for example, Attiya and Ellen⁴ for references on the more theoretical perspective, and the book²⁸ from the more practical perspective.

Global Indistinguishability Structure

The previous examples of indistinguishability have a *local* flavor: we look at a single execution α and the executions indistinguishable from α to *all* processes. It turns out that studying executions that are indistinguishable to a *subset* of processes lead to understanding the global indistinguishability structure of *all* executions. This uncovers an intimate relation between indistinguishability and higher-dimensional topological properties. The overview presented here is very informal; for a more precise description, see Herlihy et al.²²

Initial indistinguishability structure. Consider three processes b, g, w (black, gray, white) that communicate with each other to solve some task. When the computation begins, each process receives an input value. In the *binary consensus* task, the set of input values is $\{0, 1\}$. In certain *renaming* tasks, processes start with distinct input values taken from the set $\{0, 1, 2, 3\}$. Initially each process knows only its own input. An *initial state*

$$\{(b, input(b)), (g, input(g)), (w, input(w))\},$$

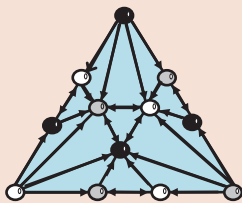
is a set of three *initial local states*, each one consisting of a pair of values. Two initial states, I_1 and I_2 are indistinguishable to a process, if the process has the same input value in both states, that is, if $I_1 \cap I_2$ contains its initial local state. If we draw an initial state as a triangle, whose vertices are the local initial states, I_1 and I_2 share an edge if they are indistinguishable to two processes, and they share only a vertex if only one process does not distinguish between them. Figure 6 shows the *input complex* for consensus looks like a triangulated sphere, and the one for renaming looks like a triangulated torus. Each one is a *simplicial complex* because it consists of a family of sets closed under containment (each edge of a triangle is a set of two local states, and each vertex is a singleton set).

How indistinguishability evolves.

As processes communicate with each other, learning about each other’s input values, the structure of indistinguishability evolves. Suppose that the processes publicly announce their input values, but each process may miss hearing either or both of the other processes’ announcements, as determined by a *communication pattern*, namely a directed graph G on the vertices b, g, w ; an arrow $v \rightarrow v'$ signifies that v' hears the input from v . Thus, v' hears inputs from the set $\mathcal{N}^-(v')$ of processes which have an arrow toward vertex v' . Which input value v hears from v depends on which initial state I is G applied to. Applying G to an initial state I , produces a new state, $\{(b, \text{view}(b)), (g, \text{view}(g)), (w, \text{view}(w))\}$, where the local state of p , $\text{view}(p)$, is the subset of I of processes $\mathcal{N}^-(p)$.

Figure 7 illustrates the *IS-patterns* (immediate snapshot or block executions), a subset of all possible communication patterns. An IS-pattern for a set of processes P is defined by an ordered partition S_1, \dots, S_k of P ($1 \leq k \leq |P|$), specifying that processes in S_i hear the values from all processes in $S_j, j \leq i$. Consider, for instance, the IS-pattern $\{b, g, w\}$ consisting of the trivial partition of $\{b, g, w\}$, which corresponds to the center triangle, where

Figure 7. IS-communication patterns.



all processes hear from each other. The arrows $g \leftrightarrow w$ belong also to the top triangle, corresponding to the partition $\{b\}, \{g, w\}$ where the only difference is that b does not hear from the other two processes.

IS-patterns are important because when applied to an input complex, I , the resulting *protocol complex* \mathcal{P} is a subdivision of I . In Figure 8, IS-patterns are applied to two consensus input simplexes. One can see that b and w with input 0 belong to two input triangles, and this edge is subdivided into three edges in \mathcal{P} , which belong to both the blue and the yellow subdivided triangles, due to IS-patterns where b and w do not hear from g (and hence cannot tell if its input is 0 or 1).

In the same way that we applied each IS-pattern to each initial state to get \mathcal{P} , we can again apply each IS-pattern, but now to each state of \mathcal{P} , obtaining a subdivision of \mathcal{P} , and so forth. Each time the processes communicate once more through an IS-pattern, the input complex is subdivided more and more finely. Indeed, a fundamental discovery is that there are topological invariants, preserved no matter how many times the processes communicate, and no matter what they tell each other each time they communicate. In the case of any unreliable asynchronous communication by either message passing or read/write shared-memory, \mathcal{P} “looks like” (is homotopic to) the input complex I .

Remarkably, topological invariants determine the computational power of the model. In other, more reliable models of computation (for example, at most t out of $n, t < n - 1$ processes can fail, or synchronous models, or

shared-memory primitives stronger than read/write registers), \mathcal{P} preserves weaker topological invariants, and “holes” are created, giving the model its additional computability power.

Specifications as indistinguishability requirements. Suppose that after communicating through IS-patterns, each process produces an output value. Let $(p, \text{view}(p))$ be the local state of a process p in the protocol complex \mathcal{P} , after an IS-pattern. Hence, the output value produced by p is a function of its view, $\delta p, \text{view}(p)$. Namely, if p does not distinguish between two triangles of \mathcal{P} , then it must decide the same value in both.

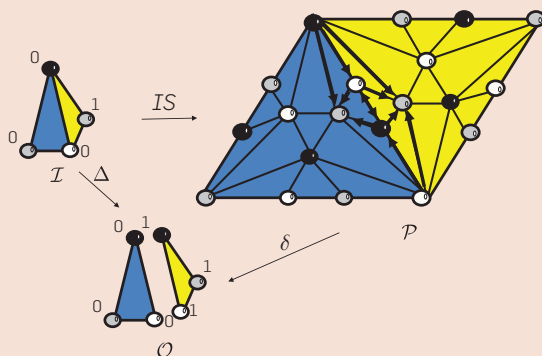
A simplicial complex defined by triangles labeled with output values is used to specify the task that the decision values should satisfy. For binary consensus, the *output complex*, in Figure 8, consists of two disjoint triangles, one labeled with 0 output values in all its three vertices, and another labeled with 1 in all its three vertices. Thus, a *task* $\langle I, O, \Delta \rangle$ consists of an input complex I , an output complex O , and a relation Δ specifying for each input triangle $\sigma \in I$, which output of O , $\Delta(\sigma)$, represent valid outputs for the task.

Finally, Figure 8 is meant to represent that the decision function δ solves the task, if for any triangle σ' in \mathcal{P} , $\delta(\sigma')$ is a triangle $\tau \in O$, such that $\tau \in \Delta(\sigma)$, where σ is the input triangle for σ' .

To summarize, a new indistinguishability global structure (represented by \mathcal{P}) is generated after communication, and a task specifies a target indistinguishability structure (represented by O). The question is whether \mathcal{P} can be (simplicially) mapped to O respecting Δ . This is a topological question with deep implications to distributed task computability in various models (message-passing and shared memory, synchronous and asynchronous, with crash and Byzantine failures).

This formalization can be interpreted as a question of gaining knowledge, as explained in Goubault et al.,¹⁸ where it is described how the simplicial complexes described in this section have an equivalent representation as Kripke models. Roughly speaking, each triangle is a state of the Kripke graph, and if two triangles share a vertex of process

Figure 8. Two input triangles, application of IS-patterns on them, and the requirement to produce consensus outputs.



p , then the two corresponding states are connected by an edge labeled p . Indeed, there is an intimate relation between indistinguishability and the theory of reasoning about knowledge for distributed computing described in Fagin et al.¹⁴

Conclusion

Indistinguishability plays a central role in computer science. Examples from different areas (automata theory, learning, specification, verification, distributed computing and epistemic logic) demonstrate how different levels of abstraction entail distinct notions of indistinguishable observations, and different uses of indistinguishability (to show computability and complexity limitations, and also to design solutions). Some examples should be treated in more depth, and there are many additional application areas.

One application area is *computational learning* and related complexity topics, as recently reviewed in Wigderson.⁴⁰ Many subareas can be viewed through the lenses of *probabilistic indistinguishability*, for example, PAC learning,³⁸ cryptography, communication complexity, indistinguishability despite errors,³² and coding theory.

Indistinguishability plays a role in artificial intelligence, for example, in Turing's test, and more generally, Turing-like tests for other applications, such as Go simulators¹⁰ and writing a program simulating a living organism.²¹

We discussed formal methods, another area where indistinguishability is a key, notably in behavioral equivalences.¹¹ And we discussed logic, where the long-standing connection between modal logic and topology goes back to McKinsey and Tarski,²⁷ and up to today, with a topological semantics for belief.⁷ Another interesting example from logic is Ehrenfeucht–Fraïssé games.³¹

Distributed computing is all about interactions, with abundant instances where indistinguishability is a key. Examples include labeling schemes, synchronizers, mutual exclusion, anonymity and symmetry, and partitioning. Many impossibility results are discussed in Attiya and Ellen.⁴

Finally, indistinguishability cuts across topics. Multi-agent epistemic

logic relies on Kripke models to represent indistinguishability.¹⁴ These in turn, can be considered as the dual of simplicial complexes,¹⁸ and we described how the indistinguishability structure evolves as interaction occurs preserving topological properties. Also, having knowledge means being able to distinguish between situations, so the same action must be taken in indistinguishable setups.²⁹ We discussed the duality between indistinguishability and knowledge also in the context of learning automata.

Acknowledgments. We would like to thank Hans van Ditmarsch, Jérémy Ledent, Arnold Rosenberg, Jennifer Welch, and the reviewers for helpful comments. Supported by grants from UNAM-PAPIIT IN106520 and ISF 380/18. □

References

1. Angluin, D. Learning regular sets from queries and counterexamples. *Inf. Comput.* 2, 75 (1987), 87–106.
2. Angluin, D. Queries and concept learning. *Mach. Learn.* 2, 4 (1988), 319–342.
3. Ross Ashby, W. *An Introduction to Cybernetics*. John Wiley, New York, 1956.
4. Attiya, H., Ellen, F. *Impossibility Results for Distributed Computing*. Morgan & Claypool, 2014, 162.
5. Attiya, H., Herzberg, A., Rajsbaum, S. Optimal clock synchronization under different delay assumptions. *SIAM J. Comput.* 25, 2 (1996), 369–389. DOI: 10.1137/S0097539794266328
6. Attiya, H., Ramalingam, G., Rinetzky, N. Sequential verification of serializability. In *Proceedings of the 37th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages POPL*, ACM, 2010, 31–42.
7. Baltag, A., Bezhanishvili, N., Özgün, A., Smets, S. A topological approach to full belief. *J. Philos. Logic* 48, 2 (2019), 205–244.
8. Bernstein, P.A., Hadzilacos, V., Goodman, N. *Concurrency Control and Recovery in Database Systems*. Addison-Wesley Pub. Co. Inc., Reading, MA, 1987.
9. Clarke, E., Grumberg, O., Jha, S., Lu, Y., Veith, H. Counterexample-guided abstraction refinement. In *LNCS-Computer Aided Verification*. E. Allen Emerson and A.P. Sistla, eds. Volume 1855. Springer Berlin, Heidelberg, Berlin, Heidelberg, 2000, 154–169.
10. Coquidé, C., Georgeot, B., Giraud, O. Distinguishing humans from computers in the game of go: A complex network approach. *Europhys. Lett.* 4, 119 (2017), 48001. <http://stacks.iop.org/0295-5075/119/i=4/a=48001>
11. De Nicola, R. *Behavioral Equivalences*. Springer US, Boston, MA, 2011, 120–127. DOI: 10.1007/978-0-387-09766-4_517
12. Edward, M. Gedanken-experiments on sequential machines. In *Automata Studies*. C.E. Shannon and J. McCarthy, eds. Number 34 in Annals of Mathematics Studies. Princeton University Press, Princeton, 1958, 129–153.
13. Eswaran, K.P., Gray, J.N., Lorie, R.A., Traiger, I.L. The notions of consistency and predicate locks in a database system. *Commun. ACM* 19, 11 (1976), 624–633.
14. Fagin, R., Halpern, J.Y., Moses, Y., Vardi, M.Y. *Reasoning About Knowledge*. MIT Press, Cambridge, MA, USA, 2003.
15. Fajstrup, L., Goubault, E., Haucourt, E., Mimram, S., Raussen, M. *Directed Algebraic Topology and Concurrency*. Springer, Switzerland, 2016.
16. Fields, C. *Bell's Theorem from Moore's Theorem*. Cambridge, MA, 2012. arXiv 1201.3672v6. <https://arxiv.org/pdf/1201.3672.pdf>
17. van Glabbeek, R.J. *The Linear Time—Branching Time Spectrum I*. Elsevier, 2001, 3–99.
18. Goubault, E., Ledent, J., Rajsbaum, S. A simplicial complex model for dynamic epistemic logic to study distributed task computability. In *Proceedings 9th International Symposium on Games, Automata, Logics, and Formal Verification (GandALF) (EPTCS)*. A. Orlandini and M. Zimmermann, eds. Volume 277, Electronic Proceedings in Theoretical Computer Science, 2018, 73–87.
19. Guerraoui, R., Henzinger, T.A., Vasu, S. Model checking transactional memories. *Distrib. Comput.* 22, 3 (2010), 129–145.
20. Halpern, J.Y., Megiddo, N., Munshi, A.A. Optimal precision in the presence of uncertainty. *J. Complex.* 1, 2 (1985), 170–196. DOI: 10.1016/0885-064X(85)90010-X
21. Harel, D. A grand challenge for computing: Towards full reactive modeling of a multi-cellular animal. *Bull. EATCS*, 81 (2003), 226–235.
22. Herlihy, M., Kozlov, D., Rajsbaum, S. *Distributed Computing Through Combinatorial Topology*, 1st edn. Elsevier-Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2013.
23. Hopcroft, J.E., Motwani, R., Ullman, J.D. *Introduction to Automata Theory, Languages, and Computation*, 3rd edn. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2006.
24. Isberner, M., Howar, F., Steffen, B. The TTT algorithm: A redundancy-free approach to active automata learning. In *Runtime Verification (LNCS)*. B. Bonakdarpour and S.A. Smolka, eds. Volume 8734. Springer International Publishing, Cham, 2014, 307–322.
25. Kearns, M.J., Vazirani, U. *An Introduction to Computational Learning Theory*. MIT Press, Boston, MA, USA, 1994. <http://ieeexplore.ieee.org/servlet/opac?bknumber=6267405>
26. Lamport, L. Time, clocks, and the ordering of events in a distributed system. *Commun. ACM* 21, 7 (1978), 558–565.
27. McKinsey, J.C.C., Tarski, A. The algebra of topology. *Ann. Math.* 45, 1 (1944), 141–191. <http://www.jstor.org/stable/1969080>
28. Mills, D.L. *Computer Network Time Synchronization: The Network Time Protocol on Earth and in Space*, 2nd edn. CRC Press, Florida, USA, 2016.
29. Moses, Y. Relating knowledge and coordinated action: The knowledge of preconditions principle. In *Proceedings of the Fifteenth Conference on Theoretical Aspects of Rationality and Knowledge, TARK 2015*, Carnegie Mellon University, Pittsburgh, USA, June 4–6, 2015, 231–245.
30. Patt-Shamir, B., Rajsbaum, S. A theory of clock synchronization (extended abstract). In *Proceedings of the Twenty-Sixth Annual ACM Symposium on Theory of Computing (STOC '94)*. 1994, 810–819.
31. Poizat, B. *A Course in Model Theory—An Introduction to Contemporary Mathematical Logic*. Springer, New York, NY, USA, 2000.
32. Ron, D., Rubinfeld, R. Learning fallible deterministic finite automata. *Mach. Learn.* 2-3, 18 (1995), 149–185. DOI: 10.1007/BF00993409
33. Rosenberg, A.L. *The Pillars of Computation Theory: State, Encoding, Nondeterminism*, 1st edn. Springer Publishing Company, Incorporated, 2009.
34. Sakakibara, Y. Recent advances of grammatical inference. *Theor. Comput. Sci.* 1, 185 (1997), 15–45.
35. Sangiorgi, D. *Introduction to Bisimulation and Conduction*. Cambridge University Press, New York, NY, USA, 2011.
36. Trakhtenbrot, B.A., Barzdin, Y.M. *Finite Automata, Behavior and Synthesis*. North Holland, Amsterdam, 1973.
37. Vaandrager, F. Model learning. *Commun. ACM* 2, 60 (2017), 86–95.
38. Valiant, L. *Probably Approximately Correct: Nature's Algorithms for Learning and Prospering in a Complex World*. Basic Books, Inc., New York, 2013.
39. Weikum, G., Vossen, G. *Transactional Information Systems: Theory, Algorithms, and The Practice of Concurrency Control and Recovery*. Elsevier, 2001.
40. Wigderson, A. *Mathematics and Computation*. Princeton University Press, Princeton, USA, 2018. To appear. <https://www.math.ias.edu/avi/book>

Hagit Attiya (hagit@cs.technion.ac.il) is a professor in the Department of Computer Science, Technion, Haifa, Israel.

Sergio Rajsbaum (rajsbaum@im.unam.mx) is a professor at the Instituto de Matemáticas, Universidad Nacional Autónoma de México, México City, México.

Providing Sound Foundations for Cryptography

On the work of Shafi Goldwasser and Silvio Micali

Cryptography is concerned with the construction of schemes that withstand any abuse. A cryptographic scheme is constructed so as to maintain a desired functionality, even under malicious attempts aimed at making it deviate from its prescribed behavior. The design of cryptographic systems must be based on firm foundations, whereas ad hoc approaches and heuristics are a very dangerous way to go. These foundations were developed mostly in the 1980s, in works that are all co-authored by Shafi Goldwasser and/or Silvio Micali. These works have transformed cryptography from an engineering discipline, lacking sound theoretical foundations, into a scientific field possessing a well-founded theory, which influences practice as well as contributes to other areas of theoretical computer science.

This book celebrates these works, which were the basis for bestowing the 2012 A.M. Turing Award upon Shafi Goldwasser and Silvio Micali. A significant portion of this book reproduces some of these works, and another portion consists of scientific perspectives by some of their former students. The highlight of the book is provided by a few chapters that allow the readers to meet Shafi and Silvio in person. These include interviews with them, their biographies and their Turing Award lectures.

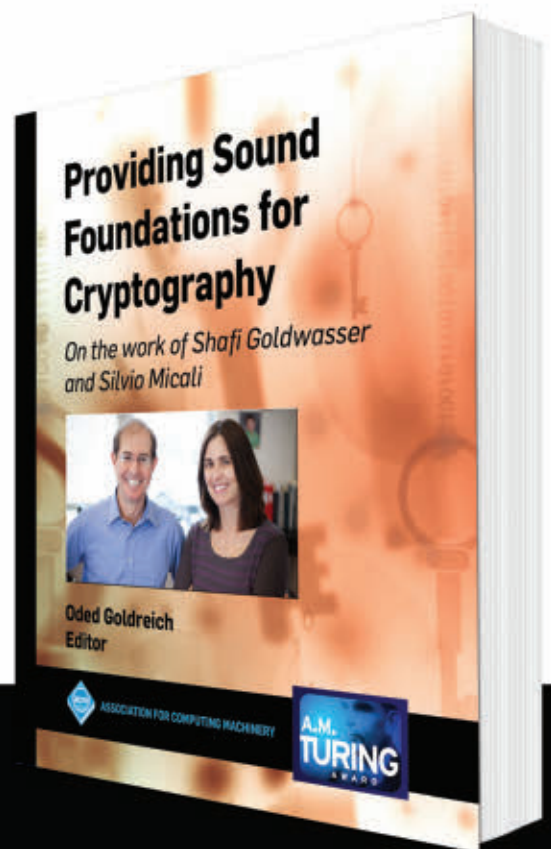
Oded Goldreich, Editor

ISBN: 978-1-4503-7267-1

DOI: 10.1145/3335741

<http://books.acm.org>

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



ACM BOOKS
Collection II

research highlights

P. 102

Technical Perspective Fake 'Likes' and Targeting Collusion Networks

By Geoffrey M. Voelker

P. 103

Measuring and Mitigating OAuth Access Token Abuse by Collusion Networks

By Shehroze Farooqi, Fareed Zaffar,
Nektarios Leontiadis, and Zubair Shafiq

Technical Perspective

Fake ‘Likes’ and Targeting Collusion Networks

By Geoffrey M. Voelker

THE FOLLOWING SCENARIO might sound like fiction. You and a million of your closest Facebook friends are going to band together to artificially improve your social networking reputation. You will willingly give a reputation manipulation service such as “official-liker.net” authorized access to your Facebook account. The manipulation service will cleverly exploit an authentication vulnerability in third-party Facebook apps to automate actions with your account. To use the service, you will view ads or pay explicit fees. The service will then use your account to “like” another Facebook account under their control—and that account will “like” yours back. You and others gain fake “likes,” presumably improving your perceived online social standing, and the reputation service makes a profit.

But this scenario, and the problem it presents to Facebook and other successful online social networks, is both a very real and challenging problem: How to completely undermine this abusive activity without negatively impacting your users (who are knowingly and entirely complicit in the abuse) or changing how apps authenticate (because that would add friction to the app ecosystem).


The following paper presents a rigorous study that explores this reputation manipulation ecosystem, ultimately working with Facebook to examine ways to stop this kind of large-scale online social networking abuse. The manipulation services are called collusion networks since the users who knowingly participate collude with each other to generate fake actions. In their work, the authors describe how to use honeypot accounts to infiltrate the collusion networks and reveal how they operate. The authors detail how the collusion networks take advantage of an authentication vulnerability using leaked access tokens to perform their ac-

tions, and comprehensively measure the extent and activity of the collusion networks they find. Who would do this? Over a million Facebook users. How many apps are vulnerable? More than half of the top 100 third-party Facebook apps. How many services are exploring this unexpected business opportunity? More than 20 such services. Finally, can these collusion networks be safely and effectively shut down? Yes.

As a final effort, the authors performed a series of careful interventions with Facebook against these services. Consider the defensive perspective of the online social network. Companies know which accounts are using collusion networks, which apps are being exploited to perform collusion, and who the collusion networks are. But services cannot shutdown the user accounts: the users are legiti-

mate, and services want them to continue to use the platform. They also cannot shutdown the apps, or how apps perform authentication: the apps have millions of legitimate users, and ease of app development relies upon the client-side token-based authentication.

The authors’ most important contribution is showing how companies can target collusion network activity and, crucially, how collusion networks respond to such interventions. The authors first experimented with a range of rate limit strategies. For access tokens used by the collusion networks, Facebook limited the rate of actions generated by accounts using such access tokens in a variety of ways, from throttling actions per day to invalidating all new tokens identified each day. Impressively, the collusion networks were able to successfully react to all token rate limit strategies, finding ways to adapt to the interventions and maintain their abusive activity. The authors then used network-based identifies, such as the IP addresses of the machines generating Facebook likes or, more broadly, the autonomous systems from which collusion activity originated. Using network identifies was much more effective, undermining nearly all collusion network activity. One of the key lasting contributions of this work is the careful, detailed methodology of experimenting with interventions and evaluating how the collusion networks respond and adapt.

Reputation has value and manipulating reputation can be a profitable enterprise. Read on for a fascinating study exploring this phenomenon in Facebook’s online social network. 

The following paper presents a rigorous study that explores the reputation manipulation ecosystem, ultimately working with Facebook to examine ways to stop this kind of large-scale online social networking abuse.

Geoffrey M. Voelker (voelker@cs.ucsd.edu) is a professor in the Department of Computer Science and Engineering at the University of California San Diego, CA, USA.

Copyright held by author.

Measuring and Mitigating OAuth Access Token Abuse by Collusion Networks

By Shehroze Farooqi, Fareed Zaffar, Nektarios Leontiadis, and Zubair Shafiq

Abstract

We uncovered a thriving ecosystem of large-scale reputation manipulation services on Facebook that leverage the principle of collusion. *Collusion networks* collect OAuth access tokens from colluding members and abuse them to provide fake likes or comments to their members. We carried out a comprehensive measurement study to understand how these collusion networks exploited popular third-party Facebook applications with weak security settings to retrieve OAuth access tokens. We infiltrated popular collusion networks using honeypots and identified more than one million colluding Facebook accounts by “milking” these collusion networks. We disclosed our findings to Facebook and collaborated with them to implement a series of countermeasures that mitigated OAuth access token abuse without sacrificing application platform usability for third-party developers.

1. INTRODUCTION

Reputation is a fundamental tenet of online social networks. People trust the information that is posted by a reputable social media account or is endorsed (e.g., liked) by a large number of accounts. Unfortunately, reputation fraud is prevalent in online social networks. A number of black-hat reputation manipulation services target popular online social networks.^{13,19} To conduct reputation manipulation, fraudsters purchase fake accounts in bulk from underground marketplaces,²¹ use infected accounts compromised by malware,¹⁸ or recruit users to join collusion networks.²²

Online social networks try to counter reputation manipulation activities on their platforms by suspending suspicious accounts. Prior research on detecting reputation manipulation activities in online social networks can be broadly divided into two categories: (a) identifying temporally synchronized manipulative activity patterns^{12,16}; (b) identifying individual accounts suspected to be involved in manipulative activity based on their social graph characteristics.^{11, 25} Recent studies have shown that fraudsters can circumvent these detection methods by incorporating “normal” behavior in their activity patterns.^{13,23} Defending against fraudulent reputation manipulation is an ongoing arms race between fraudsters and social network operators.^{3,8}

In this paper, we uncovered a thriving ecosystem of reputation manipulation services on Facebook that leverage the principle of collusion. In these collusion networks, members like other members’ posts and in return receive likes on their own posts. Such collusion networks of significant size

enable members to receive a large number of likes from other members, making them appear much more popular than they actually are. As expected, colluding accounts are hard to detect because they mix real and fake activity. Our goal in this paper is to understand their methods of coordination and execution to develop effective and long-lasting countermeasures.

OAuth Access Token Leakage. To understand the extent of the problem collusion networks pose, we analyzed popular Facebook collusion networks. We found that collusion networks conduct reputation manipulation activities by exploiting popular third-party Facebook applications with weak security settings. Third-party Facebook applications gain restricted access to users’ accounts using OAuth 2.0,¹⁴ which is an authorization framework. When a user authenticates an application using OAuth 2.0, an *access token* is generated. Collusion networks collect these OAuth access tokens for applications, which utilize the *implicit* mode in OAuth 2.0, with help from colluding members. These access tokens are then used to conduct activities on behalf of these applications and colluding accounts. Using a large pool of access tokens, collusion networks provide likes and comments to their members on an on-demand basis. We found that popular collusion networks exploited a few popular Facebook applications. However, our analysis of top 100 Facebook applications revealed that more than half of them are susceptible to access token leakage and abuse by collusion networks. Although prior research has reported several security weaknesses in OAuth and its implementations,^{6, 15, 20} we are the first to report large-scale OAuth access token leakage and abuse. As OAuth 2.0 is also used by many other large service providers, their implementation may also be susceptible to similar access token leakage and abuse.

Milking Collusion Networks Using Honeypots. We deployed honeypots to conduct a large-scale measurement study of popular Facebook collusion networks. Specifically, we created honeypot Facebook accounts, joined collusion networks, and “milked” them by requesting likes and comments on posts of our honeypot accounts. We then monitored and analyzed our honeypots to understand the strategies used by collusion networks to manipulate reputation. We identified more

The original version of this paper was published in *Proceedings of the Internet Measurement Conference, ACM, 2017*; <https://conferences.sigcomm.org/imc/2017/papers/imc17-final235.pdf>

than one million unique colluding accounts by milking collusion networks. As part of the milking process, we submitted more than 11K posts to collusion networks and received a total of more than 2.7 million likes. We identified the membership size of collusion networks by tracking the number of unique accounts that liked the posts of our honeypot accounts. Our membership estimate of these collusion networks is up to 295K for hublaa.me followed by 233K for official-liker.net in the second place. The short URLs used by collusion networks to retrieve access tokens have more than 289 million clicks to date. Our analysis of short URLs shows that popular collusion networks are used daily by hundreds of thousands of members. Collusion networks monetize their services by displaying advertisements on their heavily visited websites and offering premium reputation manipulation plans.

Countermeasures. We disclosed our findings to Facebook and worked with them to mitigate these collusion-based reputation manipulation services. Although we identified a wide range of possible countermeasures, we decided to implement the countermeasures that provide a suitable tradeoff between detection of access token abuse and application platform usability for third-party developers. For instance, we do not block the third-party applications exploited by collusion networks because it will negatively impact their millions of legitimate users. We do not disallow OAuth implicit mode, which is optimized for browser-based applications, because it will burden third-party developers with prohibitive costs associated with server-side application management. As part of countermeasures, we first introduced rate limits to mitigate access token abuse but collusion networks quickly adapted their activities to avoid these rate limits. We then started invalidating access tokens that are milked as part of our honeypot experiments to mitigate access token abuse by collusion networks. We further rate limited and blacklisted the IP addresses and autonomous systems (ASes) used by collusion networks to completely cease their operations.

2. OAUTH ACCESS TOKEN ABUSE

In this section, we first provide a background of Facebook's third-party application ecosystem and then discuss how attackers can exploit these applications to abuse their OAuth access tokens.

2.1. Background

All major online social networks provide social integration APIs. These APIs are used for third-party application development such as games, entertainment, education, utilities, etc. These applications acquire read/write permissions from the social network to implement their functionalities. Popular social network applications have tens of millions of active users and routinely conduct read/write operations on behalf of their users.

Facebook also provides a development platform for third-party applications. Facebook implements OAuth 2.0 authorization framework¹⁴ which allows third-party applications to gain restricted access to users' accounts without sharing authentication credentials (i.e., username and password).

When a user authenticates an application using OAuth 2.0, an *access token* is generated. This access token is an opaque string that uniquely identifies a user and represents a specific *permission scope* granted to the application to perform read/write actions on behalf of the user. A permission scope is a set of permissions requested by the application to perform actions on behalf of the user.

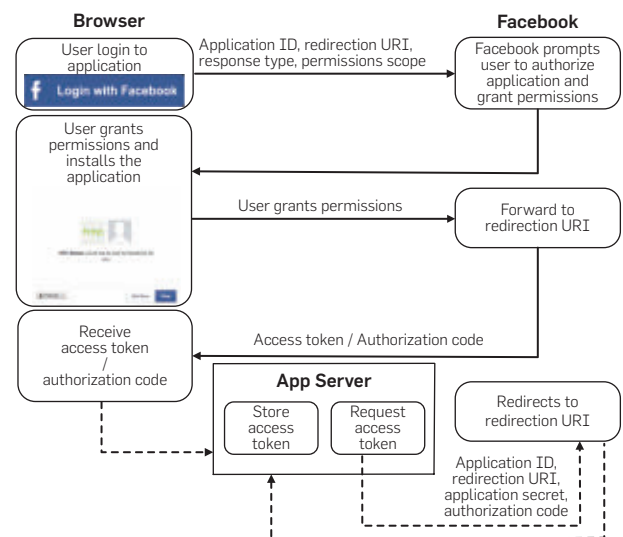
There are two types of permissions that an application may request. The first type of basic permissions does not require Facebook's approval. They include access to profile information, email addresses, and friend lists. The second type of sensitive permissions (e.g., *publish_actions*) requires Facebook's approval.⁴ These permissions allow third-party applications to conduct certain actions on behalf of a user, e.g., posting status updates, generating likes and comments.

Access tokens are invalidated after a fixed *expiration duration*. They can be categorized as short-term or long-term based on their expiration duration. Facebook issues short-term access tokens with 1–2 hours expiration duration and long-term access tokens with approximately 2 months expiration duration.

OAuth 2.0¹⁴ provides two workflows to generate an access token: client-side flow (also referred to as *implicit mode*) and server-side flow (also referred to as *authorization code mode*).^a Both workflows are similar with few changes in request parameters and some additional steps in the server-side flow. Figure 1 illustrates the OAuth 2.0 workflow of a Facebook application to generate an access token for client-side and server-side authorizations.

^a In addition to implicit and authorization code modes, OAuth 2.0 supports *resource owner password credentials mode* and *client credentials mode*. The former mode is used by clients to give their credentials (username and password) directly to the applications. The latter mode does not involve any client interaction and is used by applications to access their resources. We do not discuss these modes because they are not used to generate user access tokens.

Figure 1. Workflow of Facebook applications.



- The flow is initiated by directing a user to Facebook’s authorization server by clicking on a login button. The request to the authorization server includes application ID, redirection URI, response type, and a permission scope. The application ID is a unique identifier assigned to every Facebook application. The redirection URI is configured in the application settings. The response type is set as “token” to return access token in a client-side flow and is set as “code” to return an authorization code in a server-side flow.
- Facebook’s authorization server validates the request and prompts the user to authorize the application and grant permissions in the browser. User authorizes the application and grants the requested permissions to the application.
- Facebook redirects the user to the redirection URI along with an access token or an authorization code in the URL fragment. For the client-side flow, an access token is returned in response which is retrieved and stored by the application terminating the client-side flow. For the server-side flow, an authorization code is returned in response and the following additional step is required.
- The authorization code is exchanged for an access token by requesting Facebook’s authorization server through the application’s server.⁵ The request includes application ID, redirection URI, authorization code, and application secret. The request to exchange an authorization code for an access token is authenticated using the application secret.

The access tokens are then used by applications to perform the Facebook Graph API requests on behalf of users. For each request, an application is generally required to pass on application ID, application secret, and the corresponding access token. As we discuss next, the application secret may not be mandatory to make these requests.

2.2. Identifying susceptible applications

Applications select a suitable OAuth flow based on their access token usage scenarios. Server-side flows are by design more secure than client-side flows because they do not expose access tokens at the browser. Facebook provides an option to disable client-side flow from application settings. Facebook recommends third-party applications to disable client-side flow if it is not used.⁴ The client-side flow is typically allowed by applications that make Facebook Graph API calls only from the client side. For example, the client-side flow is used by browser-based applications which cannot include application secret in client-side code. In fact, some client-side applications may not have an application server at all and perform Graph API requests only from the browser using JavaScript. If the application secret is required, applications will have to expose their application secret in the client-side flow. It is noteworthy that the application secret is treated like a password and hence it should not be embedded in the client-side code.

Prior work has shown that attackers can retrieve access tokens by exploiting security weaknesses in the OAuth protocol and its implementations.^{6, 15, 20} Facebook applications

that use client-side flow and do not require application secret are susceptible to access token leakage and abuse. For example, attackers can retrieve access tokens in client-side flows by eavesdropping,²⁰ cross-site scripting,^{17, 20} or social engineering techniques.¹⁰ A leaked access token has serious security and privacy repercussions depending on its authorized resources. Attackers can abuse leaked access tokens to retrieve users’ personal information. Attackers can also abuse leaked access tokens to conduct malicious activities such as spreading spam/malware.

We implemented a Facebook application scanning tool to identify applications that are susceptible to access token leakage and abuse. Our tool uses Selenium and Facebook SDK for Python to launch the application’s login URL and install the application on a test Facebook account with the full set of permissions. We first infer the OAuth redirection URI used by the application by monitoring redirections during the Facebook login flow. Using the OAuth redirection URI, we install the application on the test Facebook account with the permissions that were originally acquired by the application. If the application is successfully installed, we retrieve the access token at the client-side from the application’s login URL. Using the access token, we make an API call to retrieve the public profile information of the test Facebook account and like a test Facebook post. If we are able to successfully conduct these operations, we conclude that the application can be exploited for reputation manipulation using leaked access tokens.

We analyzed top 100 third-party Facebook applications using our scanning tool. Our tool identified 55 susceptible applications, out of which 46 applications were issued short-term access tokens and 9 applications were issued long-term access tokens. Short-term access tokens pose a limited threat because they are required to be refreshed after every 1–2 hours. On the other hand, long-term access tokens provide a 2-month-long time window for an attacker. The highest ranked susceptible application which was issued long-term access tokens had about 50 million monthly active users. In fact, many of these susceptible applications had millions of monthly active users, which can cloak access token abuse by attackers.

3. COLLUSION NETWORKS

A number of reputation manipulation services provide likes and comments to Facebook users based on the principle of collusion: members like other members’ posts, and in return receive likes from other members. As discussed earlier, these collusion networks exploit Facebook applications with weak security settings. Collusion networks of significant size can enable members to escalate their reputation, making them appear much more popular than they actually are.

We first surveyed the landscape of Facebook collusion networks by querying search engines for the relevant keywords, such as “Facebook AutoLiker,” “Status Liker,” and “Page Liker,” that were found on a few well-known collusion network websites. We compiled a list of 50 such

^b <https://blog.alexandria.com/marketing-research/alexandria-rank/>

websites and used Alexa Rank,^b which is a measure of website popularity, to shortlist popular collusion networks. It is noteworthy that the top 8 collusion networks were ranked within the top 100K. For example, hublaa.me was ranked around 8K and 18% of their visitors were from India, where it was ranked within the top 3K sites. It is interesting to note that other collusion networks also got most of their traffic from countries such as India, Egypt, Turkey, and Vietnam.

We investigated popular collusion networks to understand the features they offer and to identify Facebook applications that they exploited. Collusion networks ask users to install a Facebook application and submit the generated access token in a textbox on their website. The installation link redirects users to a Facebook dialog mentioning the application’s name. Table 1 lists the applications used by popular collusion networks along with their statistics retrieved from the Facebook Graph API. Using our tool, we verified that these Facebook applications used client-side flow and did not require application secret for making the Graph API calls. We observed that *HTC Sense*, which is used by several popular collusion networks, was ranked at 40 and had on the order of a million daily active users (DAU). *Nokia Account* was ranked at 249 and had approximately a hundred thousand daily active users. Similarly, *Sony Xperia smartphone* was ranked at 886 and had on the order of 10,000 daily active users. It is noteworthy that collusion networks cannot create and use their own applications because they would not pass Facebook’s strict manual review process for applications that require write permissions.¹ However, collusion networks can (and do sometimes) switch between existing legitimate applications that are susceptible to access token leakage and abuse.

Most collusion networks have a similar web interface and they all provide a fairly similar user experience. Figure 2 illustrates the workflow of Facebook collusion networks.

- A user visits the collusion network’s website and clicks on the button to install the application. The website redirects the user to the application authorization dialog URL. The user is asked to grant the requested permissions and install the application.
- The user returns to the collusion network website after installing the application and clicks on the button to retrieve the access token. The website again redirects the user to the Facebook authorization dialog URL with *view-source* appended. The authorization dialog redirects the user to a page that contains the access token as a query string in the URL. The use of *view-source*

stops the authorization dialog from further redirections. The user manually copies the access token from the address bar and submits it at a textbox on the collusion network website.

- The collusion network saves the access token and redirects the user to an admin panel, where the user can request likes and comments. Some collusion networks require users to solve CAPTCHAs and/or make users watch ads before allowing them to request likes and comments.

4. MEASURING COLLUSION NETWORKS

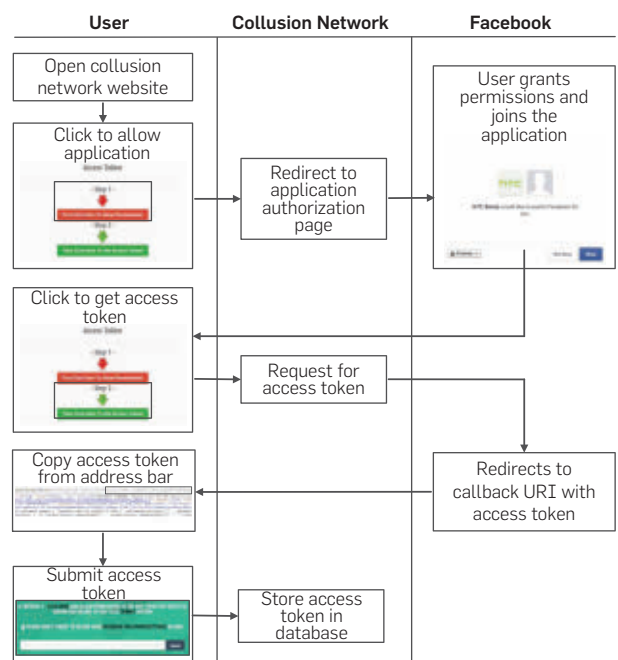
Honeypots have proven to be an effective tool to study reputation manipulation in online social networks.^{13,24} The basic principle of honeypots is to bait and deceive fraudsters for surveilling their activities. In order to investigate the operation and scale of Facebook collusion networks, we deployed honeypots to “milk” them.

We created new Facebook honeypot accounts and joined different collusion networks using the workflow described in Section 3. Our honeypot accounts regularly posted status updates and requested collusion networks to provide likes/comments on these posts. Soon after the submission of our requests to collusion networks, we noticed sudden bursts of likes and comments by a large number of Facebook accounts which were part of the collusion network. As repeated requests result in likes/comments from many unique Facebook accounts, we can uncover the memberships of collusion networks by making a large number of reputation manipulation requests. Our goal is to estimate the scale of collusion networks by tracking their member Facebook accounts. We also want to understand the tactics used by collusion networks to stay under the radar and avoid detection.

Table 1. Facebook applications used by popular collusion networks.

Application identifier	Application name	DAU	DAU rank	MAU	MAU rank
41158896424	HTC Sense	1M	40	1M	85
200758583311692	Nokia Account	100K	249	1M	213
104018109673165	Sony Xperia	10K	866	100K	1563

Figure 2. Workflow of Facebook collusion networks.



4.1. Experimental design

We registered 22 new Facebook accounts intended to be used as active honeypots for studying popular collusion networks. Each honeypot account joined a different collusion network. In an effort to actively engage collusion networks, our honeypot accounts regularly posted status updates on their timelines and requested the collusion networks to provide likes/comments on them. It was challenging to fully automate this process because collusion networks employ several tactics to avoid automation. For example, some collusion networks impose fixed or random delays between two successive requests. Many collusion networks redirect users through various redirection services before allowing request submission. Several collusion networks require users to solve a CAPTCHA in order to login and before making each request. To fully automate our honeypots, we used a CAPTCHA solving service² for automatically solving CAPTCHAs and Selenium for submitting requests to collusion networks. We continuously posted status updates and requested collusion networks to provide likes/comments over the duration of approximately 3 months from November 2015 to February 2016.

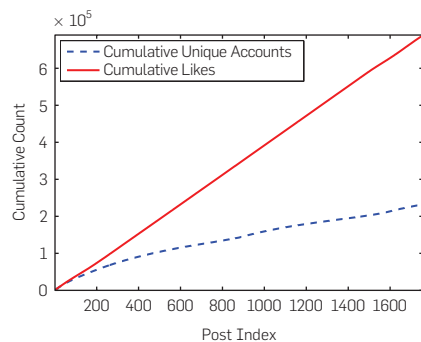
4.2. Data collection

We regularly crawled the timelines of our honeypot Facebook accounts to log incoming likes and comments provided by collusion networks. The number of unique Facebook accounts who liked or commented on a honeypot account is an estimate of the collusion network's size. Note that our membership estimate is strictly a lower bound because we may not have observed all collusion network accounts, which are randomly picked from a large pool of access tokens. We also crawled the activity logs of our honeypot accounts to collect outgoing likes and comments.

4.3. Size of collusion networks

Milking collusion networks. We posted status updates from our honeypot accounts and requested collusion networks to provide likes on the posts. Figure 3 plots the cumulative distribution of likes and unique accounts milked by our honeypots for a collusion network that represents the behavior of most of the collusion networks. We observed that the count of new unique accounts steadily declined even though the new like count remains constant. The decline represents

Figure 3. Cumulative distribution of likes and unique accounts.



diminishing returns due to the increased repetition in users who liked the posts of our honeypot accounts. Specifically, due to the random sampling of users from the database of access tokens, the likelihood of user repetition increases as we post more status updates for a honeypot account. It is important that we milk collusion networks as much as possible to accurately estimate their membership size. Although we were able to max out many collusion networks, we faced some issues for a few collusion networks. For example, djliker.com and monkeyliker.com imposed a daily limit of 10 requests, thus we were not able to fully max out these collusion networks. Moreover, arabfblike.com and a few other collusion networks suffered from intermittent outages when they failed to respond to our requests for likes. The set of unique accounts who liked posts of our honeypot accounts were collusion network members. Table 2 shows that the membership size of collusion networks varied between 295K for hublaa.me to 834 for fast-liker.com. We note that hublaa.me had the largest membership size at 295K accounts, followed by official-liker.net at 233K and mg-likers.com at 178K. The membership size of all collusion networks in our study summed up to 1,150,782. As we discuss later, some accounts were part of multiple collusion networks. After eliminating these duplicates, the total

Table 2. Statistics of the collected data for all collusion networks.

Collusion network	Incoming activities		Outgoing activities		Membership size
	Total number of posts	Total number of likes	Number of activities	Number of target accounts	
hublaa.me	1421	496,714	145	46	294,949
official-liker.net	1757	685,888	1955	846	233,161
mg-likers.com	1537	379,475	1524	911	177,665
monkey-liker.com	710	165,479	956	356	137,048
f8-autoliker.com	1311	331,923	2542	1254	72,157
djliker.com	471	70,046	360	316	61,450
autolikes-groups.com	774	202,373	1857	885	41,015
4liker.com	269	71,059	2254	1211	23,110
myliker.com	320	32,821	1727	983	18,514
kdliker.com	599	82,736	1444	626	18,421
oneliker.com	334	24,374	956	483	18,013
fb-auto-likers.com	244	19,552	621	397	16,234
autolike.vn	139	35,425	2822	1382	14,892
monsterlikes.com	495	72,755	2107	671	5168
postlikers.com	96	8613	2590	1543	4656
facebook-autoliker.com	132	4461	2403	1757	3108
realliker.com	105	19,673	2362	846	2860
autolikesub.com	286	25422	1531	717	2379
kingliker.com	107	5072	1245	587	2243
rockliker.net	99	4376	82	39	1480
arabfblike.com	311	4548	68	31	1328
fast-liker.com	232	10,270	1472	572	834
All	11,751	2,753,153	33,023	16,459	1,150,782

number of unique accounts across all collusion networks was 1,008,021.

4.4. Collusion network activities

Incoming activities. Table 2 summarizes the statistics of the data collected for different collusion networks using our honeypot accounts. In total, we submitted more than 11K posts to collusion networks and garnered more than 2.7 million likes. As shown in Figure 3, we observe that status updates typically received a fixed number of likes per request, ranging between 14 and 390 across different collusion networks. For example, official-liker.net, f8-autoliker.com, and myliker.com provided approximately 400, 250, and 100 likes per request, respectively.

Outgoing activities. Collusion networks also used our honeypot accounts to conduct reputation manipulation activities on other Facebook accounts and pages. In total, our honeypot accounts were used by collusion networks to like more than 33K posts of 16K accounts. We observed that some collusion networks used our honeypots more frequently than others. For example, autolike.vn used our honeypot accounts to provide a maximum of 2.8K likes on posts of 1.3K accounts.

5. COUNTERMEASURES

Ethical considerations. Before conducting any experiments, we received a formal review from our local Institutional Review Board (IRB) because we collected some publicly available account information such as posts and likes. We enforced several mechanisms to protect user privacy. For example, we did not store any personally identifiable information. We were aware that our honeypot accounts were used by collusion networks to conduct some reputation manipulation activities. We argue that these activities represented a small fraction of the overall reputation manipulation activities of collusion networks. Thus, we do not expect normal user activity to be significantly impacted by our honeypot experiments. The benefit of our honeypot approach in detecting collusion network accounts far outweighs the potential harm to regular Facebook users. To further minimize harm, as discussed next, we disclosed our findings to Facebook to remove all artifacts of reputation manipulation during our measurements as well as investigate countermeasures to mitigate collusion network activities.

Before implementing any countermeasures in collaboration with Facebook, we performed honeypot experiments for approximately 10 days to establish a baseline of collusion network activities. We repeated the honeypot milking experiments for popular collusion networks starting August 2016 (and continued until mid-October 2016). Figure 4 shows the average number of likes received by our honeypots for two popular collusion networks. We do not show results for other collusion networks due to space constraints. As we discuss next, while we considered a wide range of countermeasures, we decided to implement countermeasures that provide a suitable tradeoff between detection of access token abuse and application platform usability for third-party developers.

We observed that collusion networks exploited a few applications (listed in Table 1) to conduct reputation manipulation

activities. Therefore, we can immediately disrupt all collusion networks by suspending these applications. As collusion networks can switch between several other susceptible applications, we would need to suspend them as well. Suspending these applications is a relatively simple countermeasure to implement; however, it will negatively impact their millions of legitimate users. We can also make changes in Facebook's application workflow to stop access token abuse by collusion networks. For example, we can mandate application secret (thereby forcing server-side operations) for liking/commenting activities that require `publish_actions` permissions.^{4,7} As a result of this restriction, collusion networks will not be able to conduct reputation manipulation activities even if they retrieve access tokens from colluding users. However, many Facebook applications solely rely on client-side operations for cross-platform interoperability and to reduce third-party developer costs of server-side application management.^{4,14} Therefore, mandating application secret would adversely impact legitimate use cases for these Facebook applications.

5.1. Access token rate limits

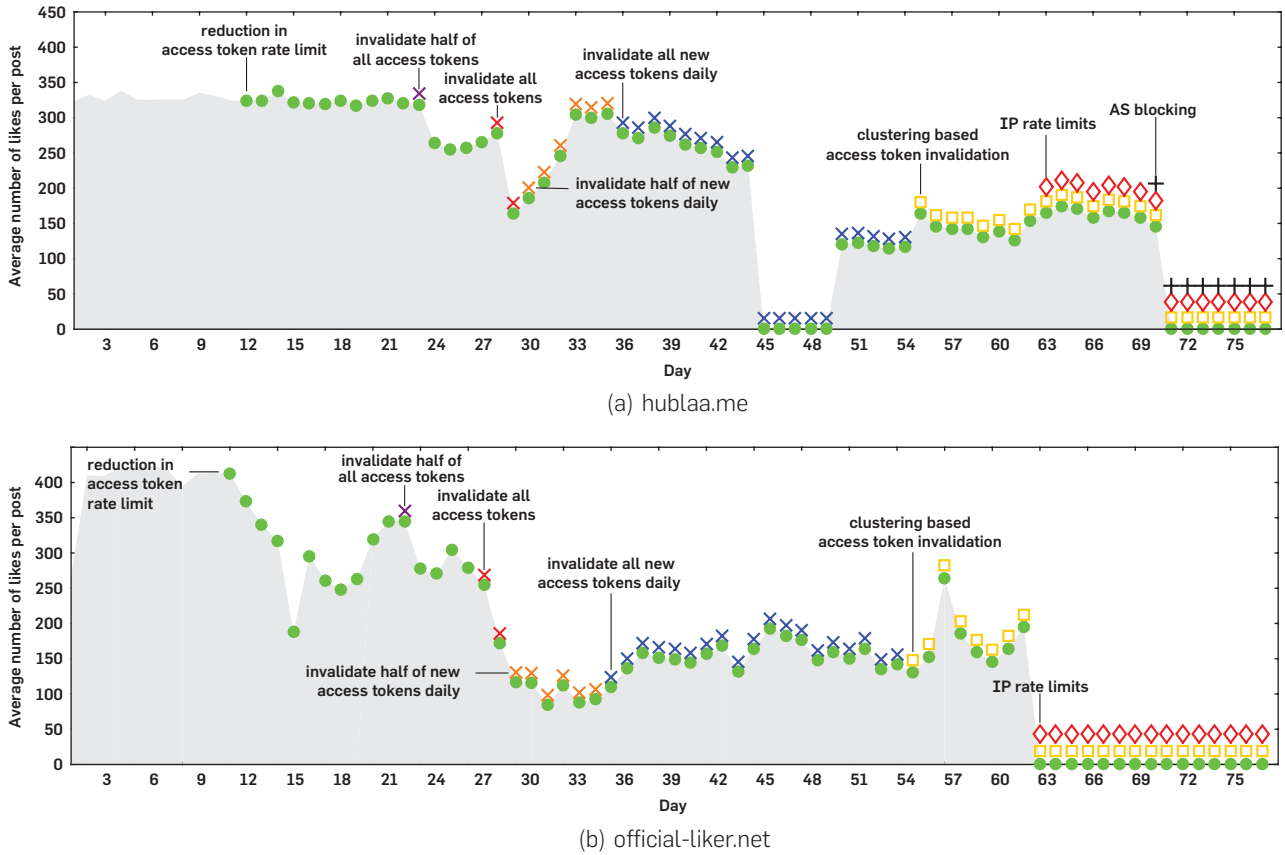
As the first countermeasure, we imposed restrictions on access tokens to mitigate abuse by collusion networks. Facebook employs rate limits to restrict excessive activities performed by an access token. As collusion network activities slip under the current rate limit, we reduced the rate limit by more than an order of magnitude on day 12 as marked by green circles in Figure 4. We observed a sharp initial decrease in activities for official-liker.net. Specifically, the average number of likes provided by official-liker.net decreased from more than 400 to less than 200 on day 16. However, official-liker.net started to bounce back after approximately 1 week. Moreover, this countermeasure did not impact hublaa.me. We surmise that both of these collusion networks had a large pool of access tokens which limited the need to repeatedly use them. Therefore, these collusion networks were able to stay under the reduced access token rate limit while maintaining their high activity levels. We did not reduce the rate limit further to avoid potential false positives.

5.2. Honeypot-based access token invalidation

We next invalidated access tokens of colluding accounts which were identified as part of our honeypot experiments. In the first 22 days, we milked access tokens of 283K and 41K users for hublaa.me and official-liker.net, respectively. We expect that invalidation of these access tokens will curb collusion network activities. To this end, we invalidated randomly sampled 50% of the milked access tokens on day 23 as marked by a black cross in Figure 4. We observed a sharp decrease in collusion network activities. Specifically, the average number of likes provided by hublaa.me decreased from 320 to 250 and for official-liker.net decreased from 350 to 275. Unfortunately, this decline was not permanent and the average number of likes gradually increased again over the next few days. We surmise that collusion networks gradually replenish their access token pool with fresh access tokens from new and returning users.

To mitigate this, we next invalidated all access tokens that were observed till day 28 (marked by red cross) and also

Figure 4. The impact of our countermeasures on two popular collusion networks. We observed that collusion network activities were not impacted by the reduction in access token rate limit. Although access token invalidation significantly reduced collusion network activities, it could not completely stop them. Clustering based access token invalidation also did not help. Our IP rate limits effectively countered most collusion networks that used a few IP addresses. We targeted autonomous systems (ASes) of collusion networks that used a large pool of IP addresses.



began invalidating 50% of newly observed access tokens on a daily basis (marked by orange cross). We observed a sharp decline for both hublaa.me and official-liker.net on day 28 when we invalidated all access tokens. However, average likes by hublaa.me started to bounce back and those by official-liker.net stabilized at 100 over the next few days. We suspect that the rate of fresh access tokens from new and returning users exceeded our rate of daily access token invalidation. This is due to the rather small number of distinct new colluding accounts milked daily by our honeypots.

To increase our access token invalidation rate, starting day 36, we began invalidating all newly observed access tokens on a daily basis as marked by blue crosses in Figure 4. We observed a steady decrease in average likes by hublaa.me from day 36 to day 44. The hublaa.me’s site was temporarily shut down on day 45. The site resumed operations on day 51 and their average number of likes decreased to 120. The official-liker.net sustained their likes between 110 and 192 despite our daily access token invalidation. Although regular access token invalidation curbed collusion network activities, we conclude that it cannot completely stop them because honeypot milking can only identify a subset of all newly joining users. Therefore, we decided not to pursue regular access token invalidation further.

5.3. Temporal clustering

Collusion networks provide likes on submitted posts in less than 1 minute. Such bursts of liking activity can be detected by temporal clustering algorithms^{12, 16} which are designed to detect accounts that act similarly at around the same time for a sustained period of time. Starting day 55, as marked by cyan squares in Figure 4, we used SynchoTrap¹² to cluster synchronized access token abuse by collusion network accounts. Surprisingly, we did not observe any major impact on collusion network activities. Our drill-down analysis shows that collusion networks avoided detection by (1) using a different set of accounts to like target posts and (2) spreading out liking activities performed by each access token over time. Figure 5 shows that different sets of accounts liked posts of our honeypot accounts. We note that 76 and 30% accounts liked at most one post of our honeypots for hublaa.me and official-liker.net, respectively. Figure 6 shows that collusion networks did not binge use our honeypot accounts within a short timeframe. We note that the hourly average of likes performed by our honeypot accounts ranges between 5 and 10. Therefore, collusion network accounts did not behave similarly at around the same time for a sustained period of time.

Figure 5. Number of our honeypot posts liked by collusion network accounts. We observed that a small fraction of collusion network accounts like multiple honeypot posts.

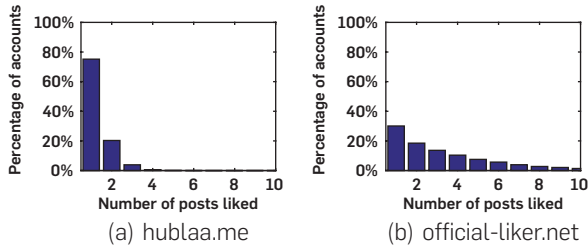


Figure 6. Hourly time series of number of likes performed by our honeypot accounts. We observed that collusion networks spread out liking activities performed by our honeypot accounts over time.

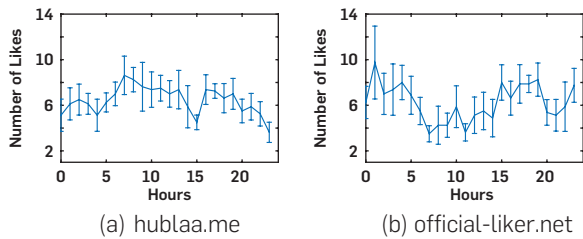
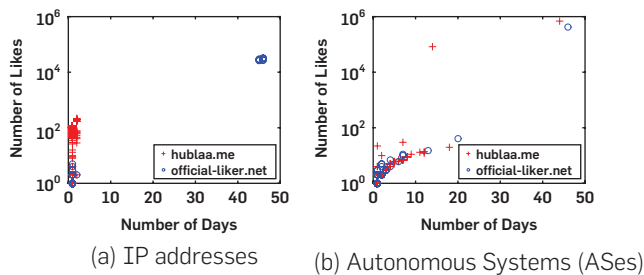


Figure 7. Source IP addresses and ASes of Facebook Graph API requests by hublaa.me and official-liker.net to like posts of our honeypot accounts.



5.4. IP- and AS-based limits

We next targeted the origin of collusion networks to further mitigate their activities. To this end, we tracked the source IP addresses of the Facebook Graph API requests for liking posts of our honeypot accounts. Figure 7(a) shows the scatter plot of these IP addresses where x-axis represents the number of days an IP address was observed during our countermeasures and y-axis represents the total number of likes generated by each IP address. It is noteworthy that a few IP addresses accounted for a vast majority of likes for official-liker.net. Therefore, we imposed a daily and weekly IP rate limits on the like requests beginning day 46. Note that this rate limit will not impact activities of normal users (e.g., regularly accessing Facebook via commodity web browsers) because this IP rate limit is only applicable to like requests by the Facebook Graph API using access tokens. Figure 4 shows that official-liker.net stopped working immediately after we imposed IP rate limits. Although not shown in Figure 4 due to space constraints,

other popular collusion networks in Table 2 also stopped working on day 63. The only exception is hublaa.me, which used a large pool of more than 6000 IP addresses and circumvented the IP rate limits. Further analysis in Figure 7(b) reveals that all of hublaa.me’s IP addresses belonged to two distinct autonomous systems (ASes) of bulletproof hosting providers.⁹ On day 70, we started to block like requests from these ASes for susceptible applications which helped in ceasing all likes from hublaa.me. Note that we targeted a small set of susceptible applications for AS blocking to mitigate the risk of collateral damage to other applications.


5.5. Limitations

First, our countermeasures should not result in collateral damage while being robust to evasion attempts by collusion networks. To date, we have not received any collateral damage complaints from popular third-party developers. Therefore, we conclude that our countermeasures do not result in significant false positives. Second, our countermeasures need to be robust against potential evasion attempts by collusion networks. Our countermeasures have proven to be long-lasting for several months now. In future, collusion networks can try to evade our countermeasures in several ways. For example, collusion networks can use many different IP addresses and ASes (e.g., using botnets and proxies) to circumvent our IP- and AS-based countermeasures. If and when that happens, we can again use honeypots to swiftly identify IP addresses and ASes used by collusion networks. Third, collusion networks may try to identify the honeypot accounts that we use to infiltrate them. For example, collusion networks can try to detect our honeypot accounts which currently make very frequent like/comment requests. To circumvent such detection, we can create multiple honeypot accounts to decrease the frequency of per-account like/comment requests.

6. CONCLUSION

We presented a comprehensive measurement study of collusion-based reputation manipulation services on Facebook. Our results raise a number of questions that motivate future research. First, we would like to investigate potential access token leakage and abuse on other popular online services that implement OAuth 2.0. For instance, YouTube, Instagram, and SoundCloud implement OAuth 2.0 to support third-party applications. Second, in addition to reputation manipulation, attackers can launch other serious attacks using leaked access tokens. For example, attackers can steal personal information of collusion network members as well as exploit their social graph to propagate malware. We plan to investigate other possible attacks as well. Third, although our simple countermeasures have been effective now for more than 6 months, collusion networks may start using more sophisticated approaches to evade them in future. We plan to investigate more sophisticated machine learning-based approaches to robustly detect access token abuse. We are also interested in developing methods to detect and remove reputation manipulation activities of collusion network members. Finally, a deeper investigation into the economic aspects of collusion networks may reveal operational insights that can be leveraged to limit their financial incentives.

Acknowledgments

This work is supported in part by the National Science Foundation under grant number CNS-1715152 and by an unrestricted gift from Facebook. 

References

1. App Review. <https://developers.facebook.com/docs/apps/review>.
2. Death By Captcha | Best and cheapest captcha service. <http://www.deathbycaptcha.com/>.
3. Facebook—Annual Report. <http://investor.fb.com/secfiling.cfm?filingID=1326801-16-43&CIK=1326801>.
4. Facebook Login for Apps—Developer Documentation. <https://developers.facebook.com/docs/facebook-login>.
5. Manually Build a Login Flow—Facebook Login. <https://developers.facebook.com/docs/facebook-login/manually-build-a-login-flow>.
6. OAuth 2.0 Threat Model and Security Considerations. <https://tools.ietf.org/html/rfc6819>.
7. Securing Graph API Requests. <https://developers.facebook.com/docs/graph-api/securing-requests>.
8. Twitter Inc.—Quarterly Report. <http://files.shareholder.com/downloads/AMDA-2F526X/3022492942x0xS1564590-16-21918/1418091/filing.pdf>, August 2016.
9. Alrwais, S., Liao, X., Mi, X., Wang, P., Wang, X., Qian, F., Beyah, R., McCoy, D. Under the shadow of sunshine: Understanding and detecting bulletproof hosting on legitimate service provider networks. In *IEEE Symposium on Security and Privacy* (2017).
10. Bilge, L., Strufe, T., Balzarotti, D., Kirda, E. All your contacts are belong to us: Automated identity theft attacks on social networks. In *WWW* (2009).
11. Boshmaf, Y., Logothetis, D., Siganos, G., Leria, J., Lorenzo, J., Ripeanu, M., Beznosov, K. Integro: Leveraging victim prediction for robust fake account detection in OSNs. In *Network and Distributed System Security Symposium* (2015).
12. Cao, Q., Yang, X., Yu, J., Palow, C. Uncovering large groups of active malicious accounts in online social networks. In *ACM Conference on Computer and Communications Security* (2014).
13. Cristofaro, E.D., Friedman, A., Jourjon, G., Kaafar, M.A., Shafiq, M.Z. Paying for likes? Understanding Facebook like fraud using honeypots. In *ACM Internet Measurement Conference (IMC)* (2014).
14. Hardt, E.D. The OAuth 2.0 authorization framework. In *IETF RFC 6749* (October 2012).
15. Fett, D., Kusters, R., Schmitz, G. A comprehensive formal security analysis of OAuth 2.0. In *ACM Conference on Computer and Communications Security* (2016).
16. Jian, M., Cui, P., Beutel, A., Faloutsos, C., Yang, S. CatchSync: Catching synchronized behavior in large directed graphs. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2014).
17. Shernan, E., Carter, H., Tian, D., Traynor, P., Butler, K. More guidelines than rules: CSRF vulnerabilities from noncompliant OAuth 2.0 implementations. In *Proceedings of the International Conference on Detection of Intrusions & Malware, and Vulnerability Assessment (DIMVA)* (2015).
18. Stein, T., Chen, E., Mangla, K. Facebook immune system. In *Workshop on Social Network Systems (SNS)* (2011).
19. Stringhini, G., Wang, G., Egele, M., Kruegel, C., Vigna, G., Zheng, H., Zhao, B.Y. Follow the green: Growth and dynamics in Twitter follower markets. In *ACM Internet Measurement Conference (IMC)* (2013).
20. Sun, S.-T., Beznosov, K. The devil is in the (implementation) details: An empirical analysis of OAuth SSO systems. In *ACM Conference on Computer and Communications Security* (2012).
21. Thomas, K., McCoy, D., Grier, C., Kolcz, A., Paxson, V. Trafficking fraudulent accounts: The role of the underground market in Twitter spam and abuse. In *USENIX Security Symposium* (2013).
22. Viswanath, B., Bashir, M.A., Crovella, M., Guha, S., Gummadi, K.P., Krishnamurthy, B., Mislove, A. Towards detecting anomalous user behavior in online social networks. In *USENIX Security Symposium* (2014).
23. Wang, G., Wang, T., Zheng, H., Zhao, B.Y. Man vs. machine: Practical adversarial detection of malicious crowdsourcing workers. In *USENIX Security Symposium* (2014).
24. Webb, S., Caverlee, J., Pu, C. Social honeypots: Making friends with a spammer near you. In *Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference (CEAS)* (2008).
25. Yu, H., Gibbons, P.B., Kaminsky, M., Xiao, F. SybilLimit: A near-optimal social network defense against Sybil attacks. In *IEEE Symposium on Security and Privacy* (2008).

Shehroze Farooqi and Zubair Shafiq ([shehroze-farooqi, zubair-shafiq]@uiowa.edu), The University of Iowa, Iowa City, IA, USA.

Nektarios Leontiadis (leontiadis@fb.com), Facebook, Washington, D.C., USA.

Fareed Zaffar (fareed.zaffar@lums.edu.pk), Lahore University of Management Sciences, Lahore, Pakistan.

© 2020 ACM 0001-0782/20/5 \$15.00

ACM Computing Surveys (CSUR)

2018 JOURNAL
IMPACT FACTOR:
6.131

Integration of computer science and engineering knowledge



ACM Computing Surveys (CSUR) publishes comprehensive, readable tutorials and survey papers that give guided tours through the literature and explain topics to those who seek to learn the basics of areas outside their specialties. These carefully planned and presented introductions are also an excellent way for professionals to develop perspectives on, and identify trends in, complex technologies.

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





Dennis Shasha

DOI:10.1145/3386912

Upstart Puzzles

Optimal Chimes

The importance of the space between the notes.

IF YOU HAVE ever spent any time near a cathedral, you will recognize the rich sound of bells announcing each hour and often even the quarter hours. This puzzle asks you to imagine living when the church bells were the only source of determining time. Specifically, imagine a country named Medievaliana whose King Meddy has decreed that at most once every 15 minutes all the churches should ring chimes in such a way that everyone within earshot would know the exact time of day to within one minute. The King Meddy, ever a stickler for precision, wants the time to be known using a 24-hour clock. So, for example, 6 P.M. would be different from 6 A.M.

It turns out the people of Medievaliana are great at counting chimes. But still, if all the chimes come in one long sequence, there seems to be no way to avoid a unary encoding. For example, 0:00 would be one chime. 0:15 would be two chimes, ..., 1:00 would be 5 chimes, ..., 23:45 would be 96 chimes. That is a lot of counting.

So, King Meddy calls in a proto-computer scientist—Ms. Bin—and asks what to do. Ms. Bin asks whether the “chimers” who ring the bells can create a period of silence. The head of the chimers’ guild responds that chimers can stop a ringing bell to create a gap, but if they are to start again afterward it might take a few seconds to do that and the time may vary from one chimer to another. So, basically, a chimer can guarantee the interval between successive chimes without a gap will be significantly less than the interval if there is a gap in between, but the exact length of the gap is unknown.

Ms. Bin thinks about this for a while and decides a listener can distinguish

between two chimes without any gap and two chimes with a gap, but the listener should not infer anything about the length of a gap, which can range from two seconds to several. In a burst of inspiration, Ms. Bin decides to encode a chime by a 1 and a gap of any length by a 0. So, for example, a listener should be able to distinguish 11 from 101, but not 101 from 1001 (because chimers are not that precise). That is, she concludes that:

- ▶ All chiming sequences should begin and end with a 1; and
- ▶ A listener will not be able to distinguish between one and several successive 0s.

Ms. Bin is most concerned about unique decodability. It must never be the case that a given chime sequence could cause the listener to think the time is different from the actual time. So every sequence of 1s and 0s must obey the rules described here and should be unique.

Challenge: Using chimes (1) and gaps (0), can you find an encoding that conforms to the aforementioned rules and has an average duration less than 9 seconds assuming the interval between successive chimes without an intervening gap is one second and the interval between chimes having an intervening gap is (on average) two seconds?

Solution: We start with the encoding of the 15 minutes within the hour. They will all be preceded by a gap and then 00 minutes will be one chime (01), 15 minutes will be two chimes (011), 30 minutes will be three chimes (0111), and 45 minutes will be four chimes (01111). The encoding for hours is shown in the figure here. So, for example, 4:30 A.M.

The encoding for hours.

1	0:00
11	1:00
101	2:00
111	3:00
1101	4:00
1011	5:00
1111	6:00
11101	7:00
11011	8:00
10111	9:00
10101	10:00
11111	11:00
111101	12:00
111011	13:00
110111	14:00
101111	15:00
110101	16:00
101101	17:00
101011	18:00
111111	19:00
1111101	20:00
1111011	21:00
1110111	22:00
1101111	23:00

would be 11010111. This encoding gives an average duration of slightly less than 8.6 seconds.

Chime Upstart. Can you find a minimum average duration uniquely decodable code under these conditions?

Chime Upstart 2. Can you find a minimum average duration uniquely decodable code assuming there can be two tones of bells in every cathedral? How about k different tones? It is still the case, however, that silences can be of varying length.

All are invited to submit their solutions to upstartpuzzles@cacm.acm.org; solutions to upstarts and discussion will be posted at <http://cs.nyu.edu/cs/faculty/shasha/papers/cacmpuzzles.html>

Dennis Shasha (dennisshasha@yahoo.com) is a professor of computer science in the Computer Science Department of the Courant Institute at New York University, New York, USA, as well as the chronicler of his good friend the omniheurist Dr. Ecco.

Copyright held by author.

volume
01

number
01

FIRST
ISSUE
PUBLISHED

ACM Transactions on Internet of Things
is now available in
the ACM Digital Library



ACM Transactions on Internet of Things (TIOT) publishes novel research contributions and experience reports in several research domains whose synergy and interrelations enable the IoT vision. TIOT focuses on system designs, end-to-end architectures, and enabling technologies, and on publishing results and insights corroborated by a strong experimental component.



SIGGRAPH THINK
BEYOND
2020 19-23 JULY WASHINGTON DC

THINK BEYOND

[S2020.SIGGRAPH.ORG](https://s2020.siggraph.org)

SIGGRAPH 2020 offers inspiration, putting the latest in art and tech at your fingertips. Discover inspiring content that demonstrates the latest advancements in computer graphics and interactive techniques research, education, applications and entertainment.

The 47th International Conference & Exhibition
on Computer Graphics and Interactive Techniques



Sponsored by ACM **SIGGRAPH**