

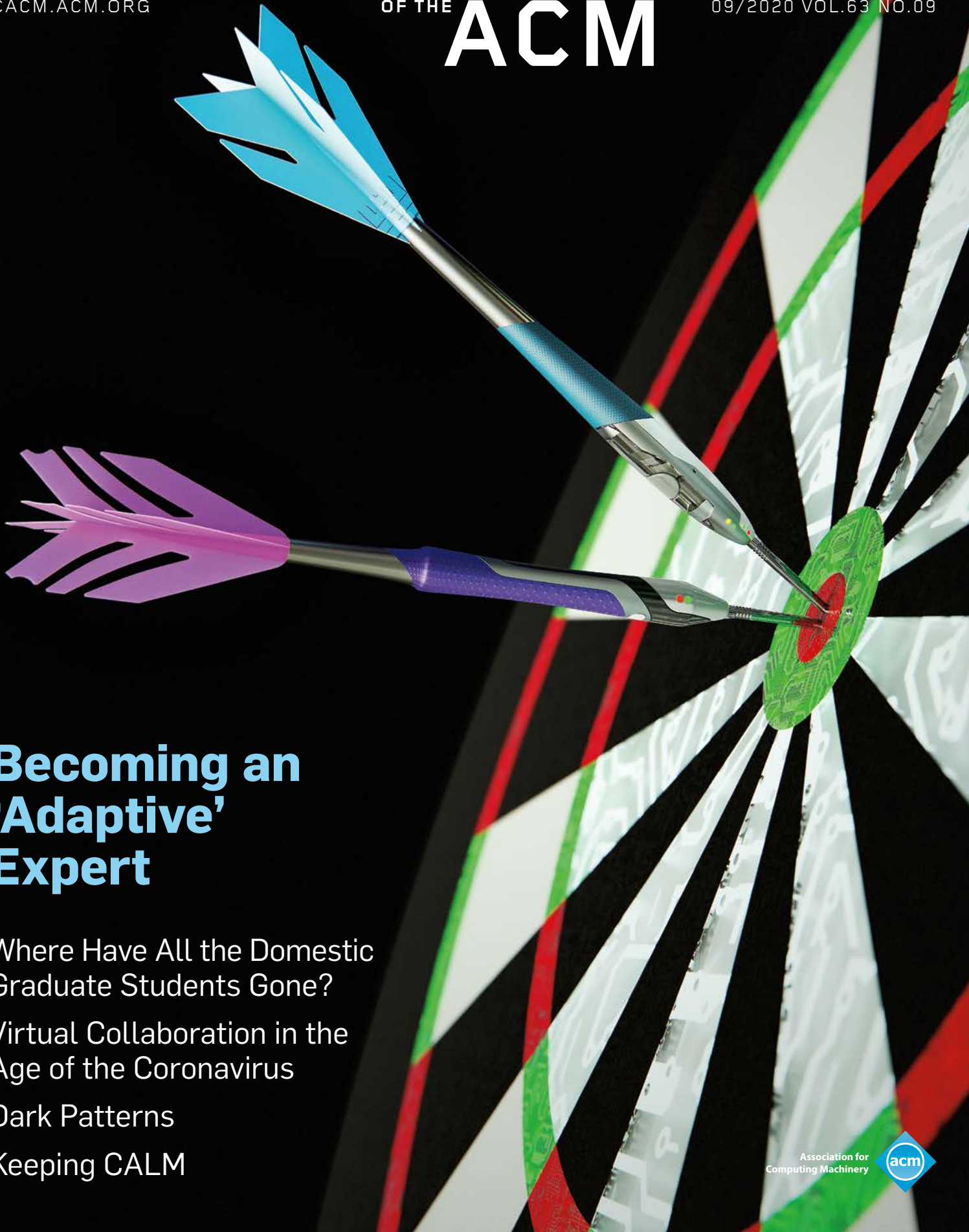
COMMUNICATIONS

OF THE

ACM

CACM.ACM.ORG

09/2020 VOL.63 NO.09



Becoming an 'Adaptive' Expert

Where Have All the Domestic
Graduate Students Gone?

Virtual Collaboration in the
Age of the Coronavirus

Dark Patterns

Keeping CALM

Association for
Computing Machinery

acm



The 11th International Learning Analytics and Knowledge Conference

April 11 - 15, 2021
Newport Beach, CA, USA

Call for papers is now available!
Visit lak21.solaresearch.org for more details

Important Dates:

Submission deadlines:

1 Oct 2020: Deadline for full and short research papers, practitioner reports, and workshop/tutorial proposal submissions

14 Oct 2020: Deadline for doctoral consortium submissions

1 Nov 2020: Deadline for posters and interactive demo submissions

14 Nov 2020: Deadline for full and short research paper rebuttal (opens 8 Nov 2020) submissions

20 Dec 2020: Deadline for camera-ready versions of all accepted submissions

Registration will open in November 2020

Early Bird Deadline is January 28, 2021 at 11:59pm PST

Sponsored By:

SOLAR
SOCIETY for LEARNING
ANALYTICS RESEARCH

 **SIGCHI**

UCI University of
California, Irvine

 **sigweb**

Program Chairs

Nia Dowell, *University of California, Irvine*

Srecko Joksimovic, *University of S. Australia*

Maren Scheffel, *Open Universiteit*

George Siemens, *University of Texas, Arlington & University of S. Australia*

Organizing Chairs

Grace Lynch, *SoLAR*

Mark Warschauer, *University of California, Irvine*

Nicole Hoover, *SoLAR*

Practitioner Chairs

Liz Gehr, *Boeing*

Mike Sharkey, *Arizona State University*

Workshop Chairs

Caitlin Mills, *University of New Hampshire*

Paul Prinsloo, *University of South Africa*

Angela Stewart, *University of Colorado Boulder*

Poster & Demo Chairs

Justin Dellinger, *University of Texas, Arlington*

Yi-Shan Tsai, *University of Edinburgh*

Doctoral Consortium Chairs

Michael Brown, *Iowa State University*

Oleksandra Poquet, *University of S. Australia*

Stephanie Teasley, *University of Michigan*

Simon Buckingham Shum, *U. of Technology, Sydney*



ACM BOOKS

Collection II

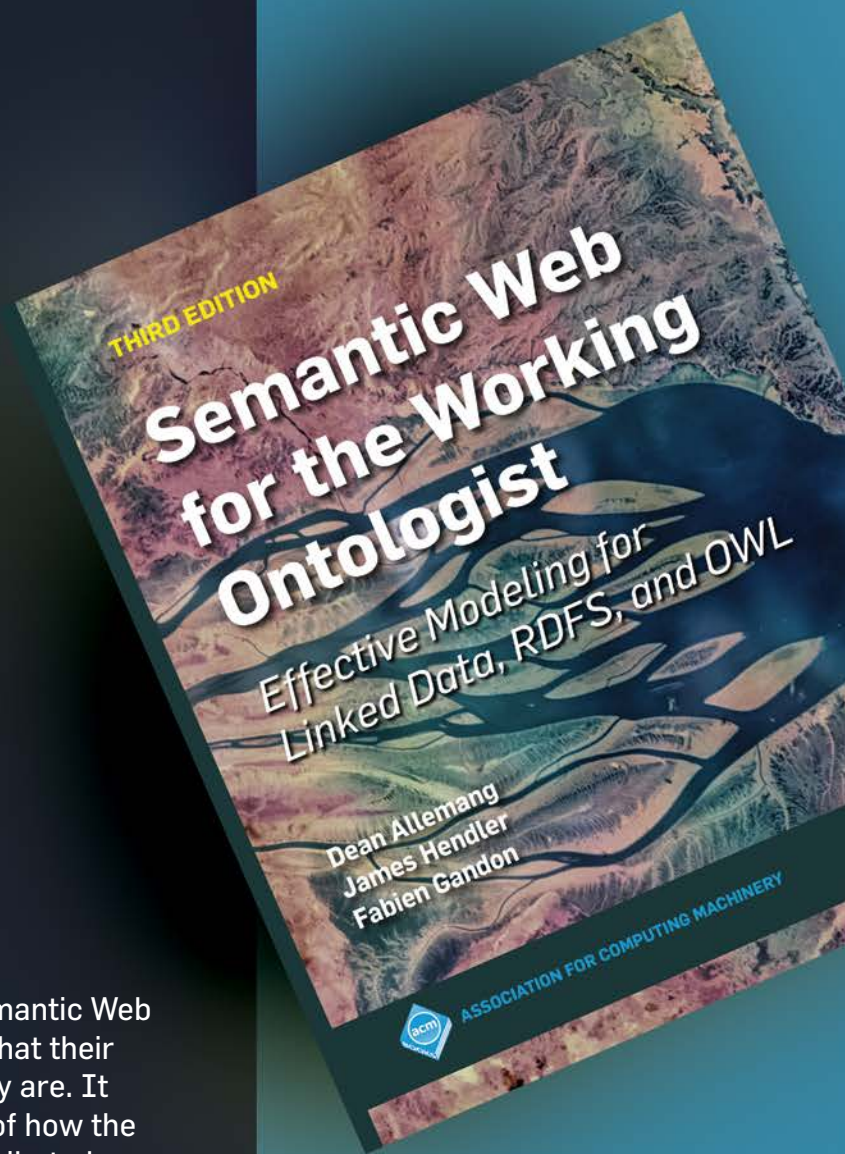
Enterprises have made amazing advances by taking advantage of data about their business to provide predictions and understanding of their customers, markets, and products. But as the world of business becomes more interconnected and global, enterprise data is no long a monolith; it is just a part of a vast web of data. Managing data on a world-wide scale is a key capability for any business today.

The Semantic Web treats data as a distributed resource on the scale of the World Wide Web, and incorporates features to address the challenges of massive data distribution as part of its basic design. The aim of the first two editions was to motivate the Semantic Web technology stack from end-to-end; to describe not only what the Semantic Web standards are and how they work, but also what their goals are and why they were designed as they are. It tells a coherent story from beginning to end of how the standards work to manage a world-wide distributed web of knowledge in a meaningful way.

The third edition builds on this foundation to bring Semantic Web practice to enterprise. Fabien Gandon joins Dean Allemang and Jim Hendler, bringing with him years of experience in global linked data, to open up the story to a modern view of global linked data. While the overall story is the same, the examples have been brought up to date and applied in a modern setting, where enterprise and global data come together as a living, linked network of data. Also included with the third edition, all of the data sets and queries are available online for study and experimentation at: data.world/swwo.

<http://books.acm.org>

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



**Semantic Web for the
Working Ontologist**
*Effective Modeling
for Linked Data, RDFS,
and OWL*

THIRD EDITION

**Dean Allemang
James Hendler
Fabien Gandon**

ISBN: 978-1-4503-7617-4

DOI: 10.1145/3382097

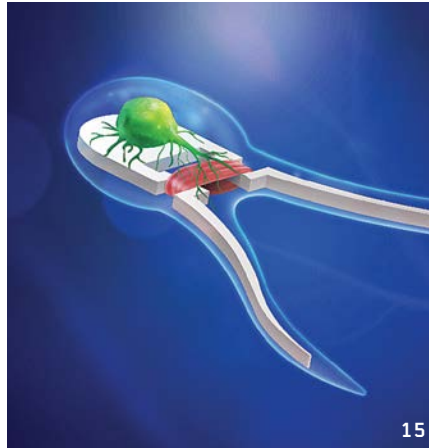
Departments

- 5 **Vardi's Insights**
Where Have All the Domestic Graduate Students Gone?
By Moshe Y. Vardi
-
- 9 **Letters to the Editor**
Lost in Translation
-
- 12 **BLOG@CACM**
Teaching CS Undergrads Online to Work With Others Effectively
Orit Hazzan on the challenges of taking a CS soft skills class online after teaching it in a classroom for a decade.

Last Byte

- 104 **Future Tense**
Little Green Message
A different kind of first-contact scenario.
By Brian Clegg

News



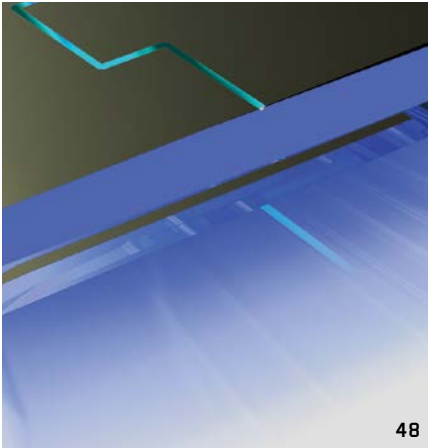
- 15 **It's Alive!**
Scientists and engineers cross the reality gap, transferring simulated evolution into real machines.
By Gregory Mone
-
- 18 **AI on Edge**
Shifting artificial intelligence to the "edge" of the network could transform computing ... and everyday life.
By Samuel Greengard
-
- 21 **Virtual Collaboration in the Age of the Coronavirus**
Videoconferencing apps took off during the COVID-19 lockdowns, but more efficient ways to collaborate virtually are waiting in the wings.
By Paul Marks

Viewpoints

- 24 **Law and Technology**
A Recent Renaissance in Privacy Law
Considering the recent increased attention to privacy law issues amid the typically slow pace of legal change.
By Margot Kaminski
-
- 28 **Security**
Autonomous Vehicle Safety: Lessons from Aviation
How more than 25 years of experience with aviation safety-critical systems can be applied to autonomous vehicle systems.
By Jaynarayan H. Lala, Carl E. Landwehr, and John F. Meyer
-
- 32 **The Profession of IT**
Avalanches Make Us All Innovators
Avalanches generate enormous breakdowns. The practices of innovation adoption may be just what you need to resolve them.
By Peter J. Denning
-
- 35 **Viewpoint**
Integrating Management Science into the HPC Research Ecosystem
How management science benefits from High Performance Computing.
By Guido Schryen
-
- 38 **Viewpoint**
'Have You Thought About ...' Talking About Ethical Implications of Research
Considering the good and the bad effects of technology.
By Amy Bruckman



Practice



48

42 **Dark Patterns: Past, Present, and Future**
The evolution of tricky user interfaces.

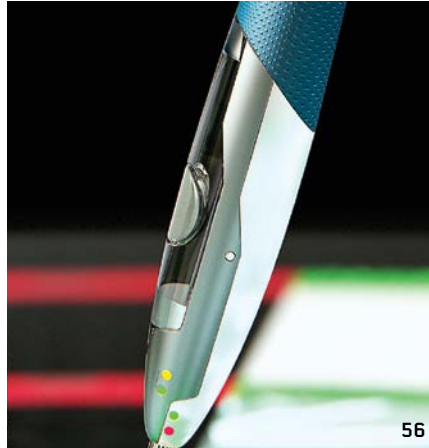
By Arvind Narayanan, Arunesh Mathur, Marshini Chetty, and Mihir Kshirsagar

48 **Is Persistent Memory Persistent?**
A simple and inexpensive test of failure-atomic update mechanisms.

By Terence Kelly

Q Articles' development led by acmqueue.queue.acm.org

Contributed Articles



56

56 **Becoming an 'Adaptive' Expert**
Investigating student knowledge transfer and metacognitive activities at college CS departments and at coding bootcamps.

By Quinn Burke and Cinamon Sunrise Bailey



Watch the authors discuss this work in the exclusive *Communications* video. <https://cacm.acm.org/videos/becoming-an-adaptive-expert>

65 **Improving Social Alignment During Digital Transformation**
Exploring what leaders can do to improve and sustain social alignment over time.

By Andrew Burton-Jones, Alicia Gilchrist, Peter Green, and Michael Draheim

Review Articles



72

72 **Keeping CALM: When Distributed Consistency Is Easy**
In distributed systems theory, CALM presents a result that delineates the frontier of the possible.

By Joseph M. Hellerstein and Peter Alvaro



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

Research Highlights

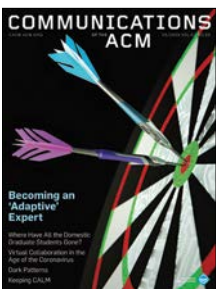
84 **Technical Perspective**
Computing the Value of Location Data
By Cyrus Shahabi

85 **Computing Value of Spatiotemporal Information**
By Heba Aly, John Krumm, Gireeja Ranade, and Eric Horvitz

93 **Technical Perspective**
Progress in Spatial Computing for Flood Prediction
By Shashi Shekhar

94 **Flood-Risk Analysis on Terrains**
By Aaron Lowe, Pankaj K. Agarwal, and Mathias Rav

IMAGES: (L) FROM SHUTTERSTOCK.COM; (C) BY PETER CROWTHER ASSOCIATES; (R) BY DABOOST



About the Cover:
Today's pool of talented programmers come armed with either college degrees or coding bootcamp diplomas. In fact, some have *both*. This month's cover story explores how students perceive their skillsets and preparedness for targeted future employment options. Cover illustration by Peter Crowther Associates.



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

BOARD CHAIRS
Education Board
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
Jonathan Aldrich; Phoebe Ayers;
Chris Hankin; Mike Heroux; James Larus;
Tulika Mitra; Marc Najork;
Michael L. Nelson; Eugene H. Spafford;
Divesh Srivastava; Bhavani Thuraisin;
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; Anupam Joshi;
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 Baha; 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;
Premkumar T. Devanbu; Jane Cleland-Huang;
Yannis Ioannidis; Trent Jaeger; Somesh Jha;
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
Shriram Krishnamurthi,
and Orna Kupferman
Board Members
Martin Abadi; Amr El Abbadi;
Animashree Anandkumar; Sanjeev Arora;
Michael Backes; Maria-Florina Balcan;
Azer Bestavros; David Brooks; Stuart K. Card;
Jon Crowcroft; 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
Sirram Rajamani, Jakob Rehof, and Haibo Chen
Board Members
Sue Moon; P.J. Narayana; 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





Moshe Y. Vardi

DOI:10.1145/3410470

Where Have All the Domestic Graduate Students Gone?

THE U.S. HAS been a magnet for technical talent from all over the world since World War II. For example, immigrants have been awarded nearly 40% of the Nobel Prizes won by Americans in chemistry, medicine, and physics since 2000. Tech industry giants Apple, Amazon, Facebook, and Google were all founded by first- or second-generation immigrants. As Sudip Parikh, the CEO of the American Association for the Advancement of Science, wrote in a recent *Science* editorial, “Immigrants make America great.”

Yet in the past couple of months, the U.S. Government has taken actions to restrict immigration of technical workforce into the country. In June 2020, President Trump temporarily suspended new work visas and barred hundreds of thousands of foreigners from seeking employment in the U.S. In early July 2020, the Trump Administration announced that international students at U.S. universities operating entirely online may not take a full online course load and remain in the U.S.

These actions by the U.S. Government, which the science and engineering community strongly objects to, have the potential of resulting in a dramatic reduction in the number of international graduate students in U.S. universities. This will be compounded by barriers to international mobility due to COVID-19. This reduction will have a devastating impact on U.S. graduate programs in computing (as well as other science and engineering programs). According to the U.S. National Science Foundation survey of graduate and postdoctoral students, about 80% of graduate students in U.S. computer science and engineering programs are international students, and about 90%

of U.S. graduate programs in computer science and engineering have a majority of international students.

The loss of international professional master’s students will result in a serious loss of income to U.S. universities, at a time when the economic crisis inflicted by COVID-19 already unleashed a heavy price on U.S. institutions. But the loss of international doctoral students would significantly diminish the research capability of graduate programs in science and engineering. After all, doctoral students, supervised by principal investigators, carry out the bulk of research in science and engineering in academic departments.

There is no question that making the U.S. less attractive to international graduate students in science and engineering would have long-term adverse consequences on U.S. technology leadership and competitiveness. An obvious question is why the U.S. Government would choose to take actions that are so detrimental to the U.S. economy. But a deeper question is how the U.S. has become so dependent on international students as the major workforce of its academic science and engineering research enterprise.

The common experience of a doctoral-admission committee in computing is that there are not enough qualified domestic doctoral applicants to fill the “needs” of their doctoral programs, where these needs are defined by the number of teaching and research assistantships offered by these programs. Graduate programs admit so many international students not only because they have strong international applicants, but mainly because they do not have enough qualified domestic applicants. We must conclude the doctoral career track is

simply not attractive enough to U.S. undergrad CS students.

Attaining a doctoral degree is a formidable undertaking, as any follower of *Ph.D. Comics* would undoubtedly know. But when a country fails systemically to create an adequate pipeline for its technical workforce, it suggests the existence of a systemic problem. Doctoral programs have a crucial dual role. On one hand, they prepare future faculty members, who will educate the next generation of computing professionals. On the other hand, they educate an advanced workforce for the computing-technology industry. Doctorate holders in computing are in high demand. The economy needs them. The lack of an adequate pipeline is a bug, not a feature!

But instead of acknowledging the existence of this problem and trying to address it, we have found a way to meet our departmental needs by recruiting and admitting international students. That is, the supply of a steady stream of highly qualified international applicants allowed us to ignore the inadequacy of the domestic doctoral pipeline.

We must object to the harmful policies of the U.S. Government. Even though the July policy has been rescinded at press time, and independent of the final outcome, the current crisis provides us with an opportunity for introspection and self-study. We need to understand the roots of the problem and propose remedies. The U.S. should welcome international doctoral students because they *enrich* our doctoral programs, not because they *sustain* our doctoral programs.

Follow me on Facebook and Twitter.

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



SIGGRAPH ASIA 2020 DAEGU

The 13th ACM SIGGRAPH Conference and
Exhibition on Computer Graphics and Interactive
Techniques in Asia

Conference 17 – 20 November 2020

Exhibition 18 – 20 November 2020

EXCO, Daegu, South Korea

Driving Diversity

SA2020.SIGGRAPH.ORG

[#SIGGRAPHAsia](https://twitter.com/SIGGRAPHAsia) | [#SIGGRAPHAsia2020](https://twitter.com/SIGGRAPHAsia2020)



Sponsored by



Organized by



ACM Gordon Bell Special Prize for HPC-Based COVID-19 Research

Call for Nominations

The Gordon Bell Special Prize for High Performance Computing-Based COVID-19 Research will be awarded in 2020 and 2021 to recognize outstanding research achievement towards the understanding of the COVID-19 pandemic through the use of high performance computing.

The purpose of the award is to recognize the innovative parallel computing contributions towards the solution of the global crisis. Nominations will be selected based on performance and innovation in their computational methods, in addition to their contributions towards understanding the nature, spread and/or treatment of the disease.

Teams may apply for the award. Nominations will be evaluated on the basis of the following considerations:

- Evidence of important algorithmic and/or implementation innovations
- Clear improvement over the previous state of the art
- Performance is not dependent on an architecture that is specialized or cannot be replicated
- Detailed performance measurements demonstrate the submission's claims in terms of scalability (strong as well as weak scaling), time to solution, and efficiency in using bottleneck resources (such as memory size or bandwidth, communications bandwidth, I/O), as well as peak performance.
- Achievement is generalizable, in the sense that other scientists can learn and benefit from the innovations
- Although solving an important scientific or engineering challenge is important to demonstrate/justify the work, scientific outcomes alone are not sufficient for this prize.

Financial support of this \$10,000 award is provided by Gordon Bell, a pioneer in high performance and parallel computing.

Nominations for the 2020 award are due on October 8, 2020.

**For more information and to
submit nominations, please visit:**

<https://awards.acm.org/bell/covid-19-nominations>



Systor2020

The 13th ACM International Systems and Storage Conference *October 13-15*

Virtual

Registration: www.systor.org/register/

Keynote Speakers:

Mahadev Satyanarayanan (Satya), Carnegie Mellon University

Ricardo Bianchini, Microsoft Research

Onur Mutlu, ETH Zurich and Carnegie Mellon University

Program Chairs:

Danny Harnik, IBM Research - Haifa, Israel

Bianca Schroeder, University of Toronto, Canada

Presentation on topics including:

Distributed, parallel, and cloud systems;

Security, privacy, and trust;

File and storage systems;

Fault tolerance, reliability, and availability

Full Program: www.systor.org/program

Sponsored by



In cooperation with



Platinum supporters



NetApp



HUAWEI



Gold supporters



NVIDIA

vmware

Silver supporters



YAHOO!
RESEARCH



TWO SIGMA



scan me



DOI:10.1145/3411279

Lost in Translation

ARON HERTZMAN'S VIEW-POINT "Computers Do Not Make Art, People Do," (May 2020, p. 45) makes excellent points as to why it is very unlikely that computers will ever replace artists. While I don't think he quite stated such, it appears to me that he may be of the opinion that replacement of (natural) intelligence (of human beings) with artificial intelligence is very unlikely.

The world is analog. So is nature. So are human beings. Most, if not all, of the endeavors we are addressing are based on digital technology, and possibly cannot replace analog entities. It is unfortunate, however, that with the hype these days, people are either unaware of reality, or simply ignoring reality, with undesirable consequences.

I like to cite a voicemail transcription I received recently. If not for one key word in the transcribed message, I would never have recognized the person claimed in the voicemail.

The transcribed message: "Hey, bro, this is Michael. I'm just wanted to know if you're at home. I need to buy roll the the chainsaw. I'm having real victory over their route here in my backyard, and I just can't get rid of it. Maybe with this all I can do that. Thank you Ral, bye-bye."

The original message: "Hey Rao, this is Marco. (umm) I just wanted to know if you are home. (umm) I need to borrow (umm) the chainsaw, I am having (rrr)real big trouble with the root here in my backyard, and I just can't get rid of it. Maybe with the saw I do that. Thank you, Rao, bye-bye."

I had to play the message a couple of times before I could jot down the details. I would like to think voicemail transcription still has a long way to go.

Raghavendra Rao Loka,
Palo Alto, CA, USA

The Question of DDT and Computing

Andrew A. Chien's editorial in the June 2020 issue (p. 5) recognized both the

benefits and negatives of DDT. However, when discussing the impacts of computing on the environment, he focused only on the negative impacts. A balanced trade-off analysis must consider the benefits, such as the greening of the world and the increased food production brought about by the carbon emissions. The reference to climate change as an existential crisis and citation of Greta Thunberg as an authority lacks credibility. While I agree with his comments on what hardware professionals can and should do, the motivation could be better balanced.

Paul E. Peters, Easton, MD, USA

Editor-in-Chief's response

Thanks for writing, Paul. I believe that computing's numerous positive impacts on the environment is widely appreciated and covered in Communications (see Gomes et al.¹). I find that all too often, computing professionals think the good is enough to justify the negatives. But it's a bit too easy to pass it off as a "trade-off." Why must it be so? As the negatives accumulate in scale and damage, let's rise to the challenge and invent ways to reduce the negative impacts of computing; not to reduce its use, but to enable its benefits to be multiplied. I, for one, am working to enable more computing good by reducing its negative impacts.

P.S. I didn't cite Greta as an authority on climate change; I was only quoting her admonition for how to solve the problem. Those calling it an existential crisis include many world leaders and of course a cadre of environmental scientists, a list far too long to cite.

Reference

1. Gomes, C. et al. Computational sustainability: Computing for a better world and a sustainable future. *Commun. ACM* 62, 9 (Sept. 2019), 56–65

Andrew A. Chien, Chicago, IL, USA

Name Change

During the last 50 years there have been many Letters to the Editor about the "Computing Machinery" part of ACM's name.

Perhaps Association of Computing Members (or Memberships) would be better and would leave the abbreviation unchanged.

Richard Rosenbaum, Bloomfield Hills, MI, USA

Editor-in-Chief's response

Ah, I knew it would not take long to get a recursive acronym! Association of Computing Members, members of what? Members of the Association of Computing Members, of course.

More suggestions? ☺

Andrew A. Chien, Chicago, IL, USA

© 2020 ACM 0001-0782/20/9 \$15.00

Coming Next Month in COMMUNICATIONS

Responsible Vulnerability Disclosure in Cryptocurrencies

A Decade of Social Bot Detection

Real Time Spent on Real Time

What Do Agile, Lean, and ITIL Mean in DevOps?

Mad Max: Surviving Out-of-Gas Conditions in Smart Contracts

Using Computer Programs and Search to Teach Theory

The History, Status, and Future of FPGAs

Plus the latest news about thwarting side-channel attacks, the intersection of block collisions and quantum search, and who can access your phone data.

ACM ON A MISSION TO SOLVE TOMORROW.



Dear Colleague,

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

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

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

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

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

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

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

Sincerely,

A handwritten signature in black ink, appearing to read 'G. Kotsis'.

Gabriele Kotsis
President
Association for Computing Machinery



Association for
Computing Machinery

Advancing Computing as a Science & Profession

SHAPE THE FUTURE OF COMPUTING. JOIN ACM TODAY.

www.acm.org/join/CAPP

SELECT ONE MEMBERSHIP OPTION

ACM PROFESSIONAL MEMBERSHIP:

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

ACM STUDENT MEMBERSHIP:

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

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

PAYMENT INFORMATION

Name

Mailing Address

City/State/Province

ZIP/Postal Code/Country

- Please do not release my postal address to third parties

Email Address

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

- AMEX VISA/MasterCard Check/money order

Credit Card #

Exp. Date

Signature

Purposes of ACM

ACM is dedicated to:

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

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

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

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

BE CREATIVE. STAY CONNECTED. KEEP INVENTING.



ACM 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

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



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

DOI:10.1145/3409780

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

Teaching CS Undergrads Online to Work With Others Effectively

Orit Hazzan on the challenges of taking a CS soft skills class online after teaching it in a classroom for a decade.



Orit Hazzan
The Advantages of Teaching Soft Skills to CS Undergrads Online

<https://bit.ly/3eKThVT>
June 8, 2020

Introduction

The Spring semester of 2020 will be remembered as the Corona semester. After a decade of teaching a classroom course on soft skills at the Technion (described in Hazzan and Har-Shai^{1,2}), I faced the challenge of teaching it online during the Corona semester (while on sabbatical at the Hebrew University of Jerusalem). At first, I wondered whether teaching soft skills online is even possible since, unlike theoretical courses, I assumed that close face-to-face (F2F) interaction is required in order to practice such skills. Eventually, I realized that teaching this course online has, in fact, some advantages, that this teaching format opens up new opportunities, and that this medium can even foster several soft skills I had

not previously considered teaching in the F2F format. This blog demonstrates these advantages by focusing on the use of the breakout rooms option available in Zoom, which I used extensively in the course.

The Breakout Rooms Tool

What are breakout rooms? The built-in breakout rooms option available in Zoom enables the creation of teams by dividing the meeting attendees (in this case, the class) into any number of rooms, either manually or automatically, according to the host's (or in our case, the instructor's) choice. After the room allocation is executed, each team finds itself in a virtual room in which the team members can communicate the same as in a regular Zoom meeting. The instructor can visit the rooms one at a time, listening and observing the teams at work and, if needed, answering questions. He or she can also send messages to all rooms. When the host wishes to gath-

er all the participants back in the main session, he or she closes the rooms and after 1 minute, the rooms close automatically and all of the participants return to the main session.

How did I use breakout rooms? The course was attended by 35 third- and fourth-year students. Each time the course content required it, I divided them automatically into nine rooms of 3-4 students each. The automated room allocation often creates teams whose members have never previously met. Prior to room distribution, I gave the class a task, including the time frame allocated for its completion, which was usually short: between 5 to 15 minutes. The tasks focused on the topics we had discussed in class and required each team to create a Power-Point presentation and to upload it to a dedicated forum that I opened each time for that specific task. The Power-Point presentation had to fulfill with very clear requirements (for example, three slides: the first presents the names of the team members, and the second and third slides each present an answer to a sub-task). The required product can be also a short video clip, a position paper, and so on.

Soft Skills Students Practice While Working in the Breakout Room

Beside the learning process of the soft skills-related topic on which the task focused, it turns out that the students practiced additional soft skills. Furthermore, the students felt comfortable working in the breakout rooms and were actually aware of the

soft skills they practiced in this working mode.

On the mid-semester survey, the students were asked the following question: “On a scale of 1 (not suitable) to 4 (very suitable), indicate the degree to which you feel working in breakout rooms suits you,” Twenty-eight students responded to the survey. None of them indicated 1, six (21.4%) indicated either 2 or 4, and sixteen (57.1%) indicated 3. The average was 3.

The students were also asked to list three soft skills that they implemented while working in the breakout rooms, and to describe how each of those skills was applied.

The following table presents the skills most frequently mentioned by the students:

Skill	Frequency
Teamwork	19
Time management	16
Preparation of presentations	9
Persuasion ability	7
Communication	4
Team leadership	4

In addition to the above six most frequently mentioned skills, the following skills were mentioned as well: communication (3 times), listening (3), working under pressure (2), adaption to change (2), work distribution (2), effective presentation of information (1), written expression (1), message delivery (1), talking before an audience (1), working with new people (1), cooperation (1), creativity (1), using a variety of technological tools (1), decision making (1), critical thinking (1), and multiculturalism (1).

From this list of soft skills which the students indicated they practiced in the breakout rooms, I elaborate here on *teamwork*, the soft skill most frequently mentioned by the students as being implemented in the breakout rooms, and illustrate how students described its implementation. In addition to regular implementation of teamwork (for example, cooperation and work distribution), the students described the following two ways of practicing teamwork: the need to adapt to working with new team members and the need to assume different roles on different occasions. Students had to practice these two skills repeatedly, ev-

ery time they worked in the breakout rooms, and their increased attention to these two expressions of teamwork is illustrated below.

Working with new team members:

► Teamwork—Each time, we worked with a different team and, each time anew, we had to adapt our dynamics and ourselves to the new team.

► Working with new people—Each time, we worked with new people and had to deal with working with different types of people with different approaches to work.

Assuming different roles:

► Working in a group—I learned how to work in a group, in a complex, remote learning situation. I succeeded in applying this skill in different forms: sometimes I was the one leading and, other times, others were at the lead and I brought myself and my opinions. I think this is exactly the meaning of working in a group; finding your place and knowing how to contribute to the entire group according to the situation.

► Teamwork—Many times when we were divided into rooms, I had to work with people whose ways of action differ from my own, to be flexible and to meet them halfway so that we could eventually submit a successful presentation with which we were all happy and satisfied. This is a skill I have had very few opportunities to practice in my studies up until today.

The following quote, which addresses the application of the skill “adapting to change,” encompasses the two facets of teamwork implementation.

► Adapting to change—We are constantly working with new teams, under new dynamics, and with people we do not necessarily know. I think that I try to give others the stage for a few seconds, see whether someone takes the reins and is more leader-like or whether there are those who are less leader-like, and then I seek my place and try to help the group work with my strengths.

Though these two expressions of teamwork may be applied also in a F2F format, the students’ increased attention to them in the online format may be explained by the fact that, due to the social distancing regulations, they worked in *distributed teams*—each in his or her home/dormitory. The online

distributed team format intensifies these expressions of teamwork and students’ attention to their existence and implementation was increased. Furthermore, since most of the students had not met F2F before and had not established any relationships, they had to be more sensitive to their team members. Consequently, these expressions were even more salient and meaningful in the remote distributed online format and were applied more extensively.

I believe this attention to the expressions of teamwork will contribute to the students’ work habits in the future, in F2F formats and not only when working in distributed teams, which are common in the computer science world regardless of social distancing.

Summary

As can be seen, teaching an online course on soft skills is possible, and may even have several advantages over a F2F format. In this blog, I focused on the use of breakout rooms, though additional advantages of online teaching exist as well. These include, for instance, the ease of inviting guest lecturers, who are spared the need to travel to the campus, and the easy use of the chat option which, among its many advantages, enables *all participants* to “talk” and express themselves in parallel and, consequently, to “listen” and be exposed to many opinions, rather than to a limited number of selected opinions voiced in the F2F format. In conclusion, despite my initial concerns, I realized that anything can be taught and learned online with relevant adjustments, which may even offer advantages over the F2F format.

References

- Hazzan, O. and Har-Shai, G. Teaching computer science soft skills as soft concepts, *SIGCSE 2013—The 44th ACM Technical Symposium on Computer Science Education*, Denver, CO, USA, 2013, 59–64 (published but was not presented due to a snowstorm).
- Hazzan, O. and Har-Shai, G. Teaching and learning computer science soft skills using soft skills: The students’ perspective, *SIGCSE 2014 - The 45th ACM Technical Symposium on Computer Science Education*, Atlanta, GA, USA, 2014, 567–572.

Orit Hazzan is a professor at the Technion’s Department of Education in Science and Technology. Her research focuses on computer science, software engineering, and data science education. For additional details, see <https://orithazzan.net.technion.ac.il/>.

ACM Transactions on Computing for Healthcare (HEALTH)

Open for
Submissions

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



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

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

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

health.acm.org



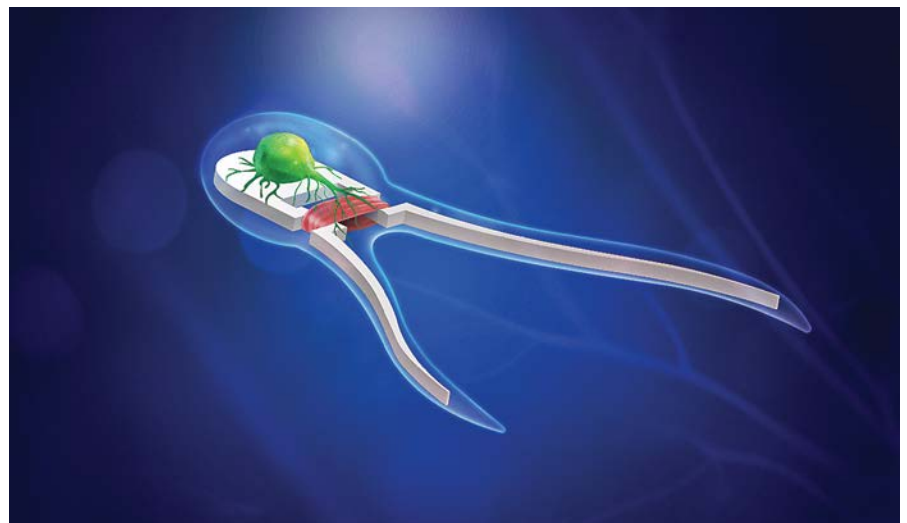
Association for
Computing Machinery

It's Alive!

Scientists and engineers cross the reality gap, transferring simulated evolution into real machines.

ALTHOUGH ROBOTIC HUMANOIDS now perform backflips and autonomous drones fly in formation, even the most advanced robots are relatively primitive when compared with living machines. The running, jumping, swimming, and flying creatures that cover our planet's surface have long inspired engineers. Yet a subset of researchers are not just taking tips from living creatures. These roboticists, computer scientists, and bioengineers are combining artificial materials with living tissue, or making machines entirely from living cells. Some projects are even borrowing the tricks of life's greatest designer, evolution, to create robots that reimagine what the very term implies.

The field of evolutionary robotics, or allowing machines to develop their own forms without human intervention, is several decades old. Similarly, the idea of integrating living and artificial tissue is not a new one. The famed Terminator robot of the Arnold Schwarzenegger movies defined itself as a cybernetic organism, a mix of metal and flesh. In the real world, roboticist Francisco Valero-Cuevas of the University of Southern California has been testing his artificial nervous systems and spinal cords by hooking them up to biological tissue for years. In 2015, Harvard University bio-



Artist's rendering of the two-tailed generation of biobots powered by skeletal muscle tissue stimulated by on-board motor neurons.

physicist Kevin Kit Parker and his colleagues designed an artificial sting ray powered by muscle cells harvested from rat hearts.

Now, however, researchers are also building living biological machines at a smaller scale. At the University of Illinois, mechanical engineer Taher Saif created a miniature swimmer by combining muscle tissue, motor neurons, and an engineered scaffold. The neuronal cells are optogenetic, which means they respond to light. Saif and his team activate their robots, called biobots, by shining light on these cells, which then

switch on the muscle tissue and cause the miniature machines to swim.

Meanwhile, his colleague at the University of Illinois, bioengineer Rashid Bashir, and his team made miniature walkers using cells in combination with a three-dimensionally (3D) printed polymer skeleton approximately the length of a staple. To get the artificial legs to move, the researchers layered them with muscle cells that respond to electricity or light. "If you give the biobot a current or shine a blue light on it, the muscle actuates and the biobot walks," Bashir explains.

Programming these tiny machines is another sort of challenge. The cells in the biobots assembled around the scaffold on their own. The neuronal cells synchronized their activities and activated the muscle cells without intervention or instructions from the researchers. Even though the biobots do not have artificial brains, they still follow some kind of program. How this happens—how cells communicate and carry out a plan—remains a mystery.

“When the Wright Brothers were making planes, we already knew Newton’s laws, so it was more of an engineering challenge. With biohybrid robots, we don’t know how two cells talk to each other, the language by which they communicate with each other or with the engineered scaffold,” says Saif. “So it’s more like the Wright Brothers trying to make a plane in the 13th century, before we knew Newton’s laws of gravity.”

Weird, Wonderful Designs

At the University of Vermont, computer scientists Josh Bongard and Sam Kriegman are working in tandem with developmental biologists Mike Levin and Douglas Blackiston at Tufts University to better understand some of these rules. The group is evolving living robots the size of a grain of sand, made from the skin cells of *Xenopus laevis* frog embryos, and dubbed xenobots. One of the main differences with their work is that the researchers play a smaller role in the design of the robot itself.

Bongard uses an evolutionary algorithm to generate candidate robot de-

“It’s...like the Wright brothers trying to make a plane in the 13th century, before we knew Newton’s laws of gravity,” said University of Illinois mechanical engineer Taher Saif.

signs. “We tell the computer what we would like the xenobot to do—in our case, move across a surface as quickly as possible—and the computer evolves solutions,” he says. The algorithm simulates tens of thousands of generations of different xenobots theoretically capable of carrying out this task on a 3,000-CPU-core supercomputer. From a day to a week later, Bongard and his team end up with approximately 100 designs for miniature moving robots. “You get all sorts of weird and wonderful shapes and designs,” he says, “just as evolution tends to produce in nature.”

Next, Bongard and his team send their top candidates to their colleagues at Tufts, who decide which they might actually be able to make. This last part, for now, is a painstaking micro-

surgery process carried out by hand. At Tufts, Levin and Blackiston select a few promising candidates, and Blackiston sculpts each xenobot out of living cells using a micro-cauterization tool.

The end-result, a physical xenobot about the size of a grain of sand, has no digestive or reproductive capacity. It is made from skin cells, but in a totally different configuration. Each xenobot uses energy stored in its cells, then degrades after a week. In this way, Bongard compares them to highly sophisticated wind-up toys. “They’d put them in the bottom of a petri dish, and in many cases the physical xenobots moved in the way the supercomputer predicted they would,” Bongard says. “This was evidence we could combine automated design tools and simulation with manufacturing.”

The work also represented proof they had crossed the reality gap.

Crossing the Reality Gap

Most of the work in evolutionary robotics has been carried out in simulation. Transferring these designs into functional real-world robots has proven difficult. “The fact that you can evolve something in simulation doesn’t necessarily mean it will work in the real world,” says roboticist Alan Winfield of the University of the West of England, Bristol.

The xenobots are physical manifestations of life that evolved in simulation. As Levin explains, if you were to look at any life form on Earth and ask why it walks, jumps, flies, or does any-

Milestones

Simons Honored With ACM Policy Award

ACM recently named Barbara Simons to receive the 2019 ACM Policy Award for her high-impact leadership as ACM President and founding chair of ACM’s U.S. Public Policy Committee (USACM), while making influential contributions to improve the reliability of and public confidence in election technology.

Simons founded USACM 26 years ago to address emerging public policy issues around technology, and led the committee for nine years. She worked to build ACM’s policy activities and

pioneered bridging the technical expertise of computer scientists with the policymaking of the U.S. government.

An expert on voting technology, Simons has been an advocate for auditable paper-based voting systems, and the author of numerous papers on secure election technology. Through her publications, reports, testimony before the U.S. Congress, and advocacy, Simons has been a key player in persuading election officials to shift to paper-based voting

systems, and has contributed to proposals for reforms in election technologies, including post-election ballot audits.

Simons served as ACM President from 1998 to 2000. In 2001, she served on President Clinton’s Export Council’s Subcommittee on Encryption and the National Workshop on Internet Voting, which conducted one of the first studies of Internet voting.

A Fellow of ACM and the American Association for the Advancement of Science,

Simons has received the Computing Research Association Distinguished Service Award, the Electronic Frontier Foundation Pioneer Award, the ACM Outstanding Contribution Award, and the Computer Professionals for Social Responsibility Norbert Wiener Award.

The ACM Policy Award recognizes an individual or small group that had a significant positive impact on the formation or execution of public policy affecting computing or the computing community.

thing else especially well, the answer would be that its ancestors were selected to do so. “This is the only life form where that whole evolutionary history took place inside a computer,” Levin says. “These cells were not evolved for the ability to run around and do the things we see them doing. The evolutionary history happened on a computer in the Bongard lab.”

The Vermont/Tufts team is not the only group looking to cross the reality gap. An international group of roboticists is working on this problem at a larger scale as part of the Autonomous Robot Evolution (ARE) project. ARE robots evolve both in simulation and in real space. Not all proposed forms are viable; some might rely on wheels that don’t touch the ground, for example. The ARE program filters these candidates out, and viable robots are manufactured autonomously from a combination of 3D-printed and pre-fabricated parts. The latter, which include batteries, microprocessors, and wheels, can be thought of as organs, says roboticist Alan Winfield. “This is still biologically plausible, in an evolutionary sense, because the organs in our bodies evolved long before *Homo sapiens*,” he says. “Biological evolution is highly modular.”


A robotic fabricator prints the skeletal or structural parts, clips everything into place, and completes the wiring, with minimal human intervention. “It is literally fabricating a complete, evolved robot,” Winfield says.

Useful Work

The eventual goal of all these projects is to make machines for real-world applications. Bashir and Saif at the University of Illinois envision robots with medical applications, such as miniature machines that actively pump blood through vessels. Winfield and his colleagues dream of robots that could evolve in situ on distant planets or moons; instead of sending a finished robot to the surface of an unknown world with unknown conditions, a space agency could send the machinery for evolving a robot. “The fitness function—locomotion, exploration, discovery of certain materials—could still be designed by humans,” Winfield says, but the robot itself would be optimized for and manufactured on the

surface of that planet, once the local physics is understood.

Xenobots are not purely intellectual constructs, either. Levin envisions them being used inside the human body to hunt down cancer cells in lymph nodes, reshape cartilage in arthritic knees, or scrape plaque off artery walls. Bongard imagines they could be used for environmental remediation in oceans, too; a swarm of such robots could be designed to clean up an oil spill over a wide area. Any such plan would have to be designed to minimize the negative impact on the environment, Bongard warns, but given that the xenobots degrade after seven days and have no reproductive capacity, the potential is real.

As for whether these cellular machines should be considered robots, experts like Bongard, Saif, and Bashir say there’s no question. “Most people associate robots with materials, something made out of metal and electronics,” Bongard says. “But the original meaning was something that does useful work for humans. That’s the hope: that we can make these biological constructs do useful things for people.” 

Further Reading

Lipson, H. and Pollack, J.B. Automatic design and manufacture of robotic lifeforms, *Nature*, Volume 406, Issue 6799, 2000, <https://www.nature.com/articles/35023115>

Kriegman, S., Blackiston, D., Levin, M., and Bongard, J. A scalable pipeline for designing reconfigurable organisms, *PNAS*, 117 (4), 2020, <https://www.pnas.org/content/117/4/1853>

Eiben, A.E. and Smith, J. From evolutionary computation to the evolution of things, *Nature*, Volume 521, Issue 7553, 2015, <https://www.nature.com/articles/nature14544>

Pagan-Diaz, G.J., Zhang, X., Bashir, R., et. al. Simulation and fabrication of stronger, larger, and faster walking biohybrid machines, *Advanced Functional Materials*, Volume 28, Issue 23, 2018, <https://onlinelibrary.wiley.com/doi/abs/10.1002/adfm.201801145>

Video: The Autonomous Robot Evolution project's proof-of-concept Robot Fabricator: <https://youtu.be/ASEWVsSdKzE>

Gregory Mone is the author or co-author of 10 books for children and adults, including the forthcoming novel *The Accidental Invasion*.

© 2020 ACM 0001-0782/20/9 \$15.00

ACM Member News

USING GAME THEORY AND SOCIAL CHOICE IN MULTI-AGENT SYSTEMS



“I built a tic-tac-toe computer for an eighth-grade science fair project, which got me into

computers,” says Jeffrey S. Rosenschein, a professor in the Rachel and Selim Benin School of Computer Science and Engineering at The Hebrew University of Jerusalem in Israel.

Rosenschein earned his undergraduate degree in Applied Mathematics from Harvard University, and his master’s degree and doctorate in computer science from Stanford University.

On completing his Ph.D., Rosenschein joined the faculty of The Hebrew University, where he is director of the Multi-agent Systems Research Group.

Rosenschein explores the use of game theory and social choice to establish foundations for multi-agent systems. Much of his research has focused on examining the role of incentives in intelligent agents.

“Classic artificial intelligence looked at the ‘what’ and the ‘how’ of intelligent systems, means-ends analysis; finding actions to accomplish a goal,” Rosenschein said. AI does not consider the ‘why’ of intelligent action and the role of incentives, which became central to his work.

“What’ and ‘how’ are often sufficient when relating to the computer as a directed machine, but not when you relate to the computer as a representative or assistant. A directed machine’s incentives can remain external and unformalized, but an autonomous machine should internalize formalized incentives,” says Rosenschein.

Regarding his current research, Rosenschein says, “Integrating machine learning techniques and connectionist approaches with symbolic approaches is a very promising direction for many subfields of artificial intelligence, and it also has potential for multi-agent systems.”

—John Delaney

AI on Edge

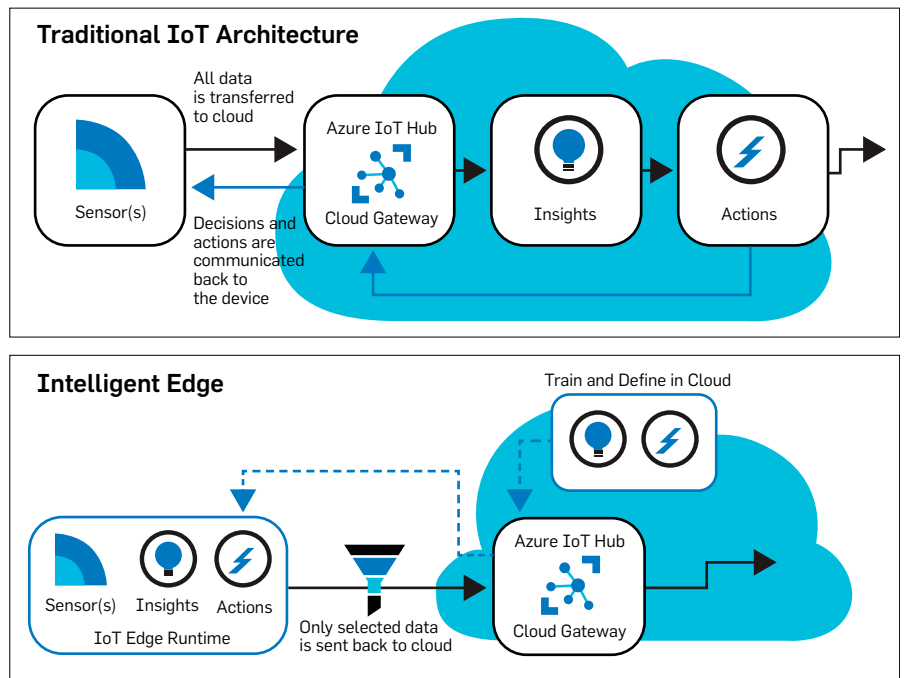
Shifting artificial intelligence to the “edge” of the network could transform computing ... and everyday life.

A REMARKABLE THING ABOUT artificial intelligence (AI) is how rapidly and dramatically it has crept into the mainstream of society. Automobiles, robots, smartphones, televisions, smart speakers, wearables, buildings, and industrial systems have all gained features and capabilities that would have once seemed futuristic. Today, they can see, they can listen, and they can sense. They can make decisions that approximate—and sometimes exceed—human thought, behavior, and actions.

Yet, for all the remarkable advancements, there’s a pesky reality: smart devices could still be a whole lot more intelligent—and tackle far more difficult tasks. What’s more, as the Internet of Things (IoT) takes shape, the need for low latency and ultra-low energy sensors with on-board processing is vital. Without this framework, “Systems must depend on distant clouds and data centers to process data. The full value of AI cannot be realized,” says Mahadev Satyanarayanan, Carnegie Group Professor of Computer Science at Carnegie Mellon University.

Edge AI takes direct aim at these issues. “To truly and pervasively engage AI in the processes within our lives, there’s a need to push AI computation away from the data center and toward the edge,” says Naveen Verma, a professor of electrical engineering at Princeton University. This approach reduces latency by minimizing—and sometimes complexly bypassing—the need for a distant datacenter. In many cases, computation takes place on the device itself. “Edge AI will enable new types of systems that can operate all around us at the beat of life and with data that is intimate and important to us,” Verma explains.

The power of this framework lies in processing data exactly when and where it is needed. “Edge AI introduces new computational layers between the cloud and the user devices. It distributes application computations be-



tween these layers,” says Lauri Lovén, a doctoral researcher and data scientist at the University of Oulu in Finland.

Pushing intelligence to the edge could also fundamentally alter data privacy. Specialized chips and cloudlets—essentially micro-clouds or ad hoc clouds that would function in a home, business, or vehicle—could control what information is sent from smart devices, such as TVs and digital speakers.

Beyond the Data Center

At the heart of edge AI is a simple but profound challenge: getting computing systems to make decisions at the pace of the human mind and real-time events. For artificial intelligence to fully blossom, any system incorporating AI must operate without any significant drop-off in speed and accuracy. This typically requires latency below 10 milliseconds. However, today’s clouds respond in the neighborhood of 70-plus milliseconds; connections that incorporate wireless connectivity are even slower, Satyanarayanan points out.

The current approach of forcing data

streams through a few large datacenters inhibits the capabilities of increasingly sophisticated digital technologies. Edge AI takes a different tack; it runs algorithms locally on chips and specialized hardware, rather than in distant clouds and remote datacenters. This means a device can operate without a persistent connection to a dedicated network, or the Internet, and it can access remote connections and transfer data on an “as needed” basis. Current frameworks, including “edge computing” and “fog” networks, offer only incremental benefits because chips are not optimized for AI and networks were not specifically designed for edge AI.

By creating “smarter” devices and curbing reliance on conventional datacenters, it is also possible to dramatically lower energy consumption. Chip maker Qualcomm claims its edge AI-optimized chips produce energy savings as great as 25x compared to conventional chips and standard computing approaches. Low-power wireless connectivity also reduces reliance on batteries that must be swapped

out or recharged constantly. Yet another benefit is that edge AI introduces stronger controls for sensitive and private information because data stays on the device or chip, or in a local cloud the user controls.

A low-latency framework requires new chips, storage devices, and algorithms, however. It significantly alters conventional computing models. “Modern mainstream data-driven AI methods, particularly in decision-making and machine learning (ML), are designed to be run in a cloud environment, with all data items always available for learning or inference on abundant and homogeneous computing resources,” Lovén observes. “The cloud-native paradigm is a poor fit for the opportunistic, distributed, and heterogeneous edge computing environment, where devices appear and disappear, connections fail, and device batteries run out—and where user and edge devices have widely varying computational resources.”

Smarter Devices

Pushing decision-making and other functions to the edge of the network produces dramatic changes, Verma says. For example, an autonomous vehicle could use onboard machine learning to adapt to different conditions and drivers dynamically. A collection of sensors in a home or hospital could better track patients, including the elderly, and detect potential problems, such as a patient’s inability to get out of bed or failing to take medications. Edge AI also could monitor the condition of underground pipes without any need to change a hard-to-reach sensor battery for decades. “Right now, what we do at the edge is fairly basic, but within a few years we will likely see robust functionality,” says Kurt Busch, CEO and co-founder of Syntiant Corp., a company developing Edge AI chips.

While many of these things already take place today without edge AI, eliminating the round trip to the cloud would significantly alter functionality. For example, it’s a safe bet that a language translation app today will function reasonably well in Barcelona or Beijing, but things get trickier in, say, the Gobi Desert of Mongolia, where there is no cellular connection. Yet, even when a strong signal exists, the process of bouncing phrases to the

cloud and back takes time, and it creates awkward, and often unacceptable, lags. Edge AI could solve the problem by storing all the needed data on the device and hitting the Internet only when it is necessary and desirable.

Another particularly appealing feature of edge AI is wake-on-command functions. These systems can dial down power consumption to near zero when a device isn’t in use. This allows some devices to operate for years or decades without a recharge or a new battery. Remote video cameras, medical implants, and embedded sensors would benefit from this feature. What’s more, many appliances—microwave ovens or coffee makers, for example—don’t require vast processing capabilities, or a Siri or Alexa, to operate; a couple of hundred hard-wired words will do. “The device becomes more responsive and delivers better privacy because you don’t have to deal with the roundtrip of the cloud,” Busch explains.

Edge AI could add new, more advanced features to smartphones, watches, smart glasses, smart TVs, Bluetooth ear buds, hearing aids, remote control devices, smart speakers, medical devices, and various IoT devices. However, Amit Lal, professor of electrical engineering at Cornell University, believes edge AI could have an impact far beyond microwave ovens that let people bark out cooking instructions, or a hearing aid that automatically adjusts to the user and the surrounding environment. As part of a team that oversaw the NZERO program for the U.S. Defense Advanced Research Projects Agency (DARPA) between 2017 and 2019, Lal and others explored ultra-low-power or zero-power nanomechanical learning chips that could harness acoustical signals or other forms of ambient energy and wake as needed. At some point, this research could lead to vehicles and other machines that can be detected by a unique acoustical signature. “You would verify the identity of the vehicle or other device before it gets close and poses a threat,” he says.

Rethinking and Rewiring AI

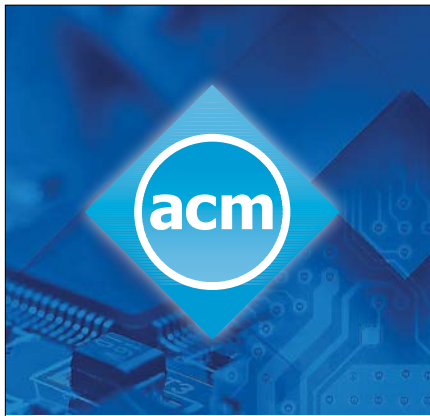
Realizing the full potential of edge AI requires a focus on things both practical and technical. There is a need for new devices and network models that bypass virtual assistants, smart speakers, and the cloud. A starting point for

addressing this task is engineering microprocessors designed specifically for deep learning and on-chip AI functions, including speech processing and wake-on-demand features. “Edge AI requires an entirely different framework for data collection, modeling, validation, and the production of a deep learning model,” Syntiant’s Busch says.

Syntiant is one of several companies developing chips specifically engineered for edge AI. Others include Ambient, BrainChip, Coral, GreenWaves, Flex Logix, and Mythic. Such chips typically run machine learning algorithms as 8-bit or 16-bit computations, which optimizes local performance but also reduces energy consumption, in some cases by orders of magnitude. Unlike traditional Von Neumann or stored-program chips such as central processing units (CPUs) and digital signal processors (DSPs), edge AI chips don’t need to swap data between the memory and the processor; instead, they typically rely on in-memory or near-memory data flow designs that place the logic and the memory data closer together. Busch says Syntiant’s Neural Decision Processor produces a 100x efficiency improvement over stored program architectures such as CPUs and DSPs.

Yet, the current class of edge AI chips is only a starting point. Busch says future edge chips likely will take on different designs and features, depending on the use case. Emerging memory technologies like Magnetoresistive Random-access Memory (MRAM) and Resistive Random-Access memory (ReRAM) could further optimize performance and power for specific uses cases, including ultra-low-power applications running independent of a data center. Other chipmakers are studying nonvolatile flash memory (NOR) as a way to store code on devices for more advanced machine learning functionality.

It will take more than new and better chips to push edge AI into the mainstream, however. Satyanarayanan says there’s a need to deploy cloud computing in entirely new ways. A decade ago, he introduced the idea of cloudlets—essentially a datacenter in a box—that could operate in planes, trains, automobiles, houses, and offices. “The same Xeon hardware that occupies a football-sized building would be adapted to a small box or rack to fit the envi-



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



“Widely deployed cloudlets would fundamentally change the way data flows, processes take place, and machines handle decisions.”

ronment. These hyperconverged clouds bring compute closer to the user. You wind up with high bandwidth and low latency,” he says. Such systems, and edge AI, could be further enhanced with the introduction of 5G, which better supports IoT frameworks.

Over the last couple of years, the idea of cloudlets and edge AI has begun to take shape. Amazon Web Services has introduced Wavelength, and Google has introduced Edge TPU, hardware and software solutions that accommodate edge functionality. Although edge AI technology poses questions, including how to approach physical protection and cybersecurity optimally, the model is garnering attention and gaining momentum. “Widely deployed cloudlets would fundamentally change the way data flows, processes take place, and machines handle decisions,” Satyanarayanan says.

On the Leading Edge

Moving edge AI off the drawing board and into everyday life will require a few other things. One particularly important requirement is distributed learning and inference algorithms that function in a dispersed, opportunistic, and heterogeneous edge environment with non-IID data (data that is dependent or unidentically distributed), Lovén says. How well these systems accomplish the task will determine how effectively they work and how much value they provide—particularly in highly connected IoT ecosystems.

In addition, there’s a need for libraries and frameworks that implement new and more efficient algorithms. Edge AI application developers and on-chip or on-device machine learning

tasks will require ready-made tools and resources. Moreover, these libraries must operate in different edge environments, including ad hoc clouds or cloudlets from different manufacturers. Lacking this framework, compatibility and data quality issues will emerge, and edge AI could stumble. “Existing frameworks such as Spark, Tensorflow, or Ray are essentially cloud-native, and their computational models are a poor fit to the edge environment,” Lovén says.

Despite technical challenges and new security concerns, edge AI will almost certainly gain momentum over the next few years. Not only will edge chips and other components appear in appliances, devices, and sensors, they will introduce entirely new ways to tap AI, neural nets, and machine learning—while perhaps recapturing a sense of privacy that has been largely lost in the digital era. Says Lal, “There are an incredible number of applications and possibilities for edge AI. If you make machines and sensors smarter and lower their power requirements, you open up a world of possibilities.”

Further Reading

Satyanarayanan, M. and Davies, N. **Augmenting Cognition Through Edge Computing**, IEEE Computer Society, Volume: 52, Issue: 7, July 2019, Pages 37-49. <https://ieeexplore.ieee.org/document/8747287>

Lovén, L., Leppänen, T., Peltonen, E., Partala, J., Harjula, E., Porombage, P., Ylianttila, M., and Riekkki, J. **Edge AI: A Vision for Distributed, Edge-native Artificial Intelligence in Future 6G Networks**, 6G Wireless Summit, March 24-26, Levi, Finland. <http://jultika.oulu.fi/files/nbnfi-fe2019050314180.pdf>

Rausch, T. and Dustdar, S. **Edge Intelligence: The Convergence of Humans, Things, and AI**, 2019 IEEE International Conference on Cloud Engineering (IC2E), 24-27 June 2019. <https://ieeexplore.ieee.org/abstract/document/8789967>

Murshed, M.G.S., Murphy, C., Hou, D., Khan, N., Ananthanarayanan, G., and Hussain, F. **Machine Learning at the Network Edge: A Survey**, <https://deeplearn.org/arxiv/113246/machine-learning-at-the-network-edge:-a-survey>

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

© 2020 ACM 0001-0782/20/9 \$15.00

Virtual Collaboration in the Age of the Coronavirus

Videoconferencing apps took off during the COVID-19 lockdowns, but more efficient ways to collaborate virtually are waiting in the wings.

WHEN THE COVID-19 pandemic began sweeping the globe early in the year, and governments began enforcing lockdowns that forced people to stay at home to depress infection rates, videoconferencing technologies rocketed into public consciousness as never before.

Professional apps including Zoom, Skype, Webex, and Microsoft Teams were suddenly thrown into the hands of people who had never used them, alongside more social-media-oriented ones like Houseparty and Whereby, as people sought virtual connection and collaboration tools to cope with the stay-at-home and work-from-home orders.

The effect of this rapid adoption of video chat systems was dramatic. Suddenly, debate in the media and on social networks centered on which was the best app or desktop package, with users treating it almost like an exercise in comparative religion.

Uses for the technologies flourished along with those ballooning user numbers, with video livestreams suddenly dominating locked-down domestic and work agendas. From live exercise workouts to yoga and meditation sessions before breakfast, to gaming at a distance, to attending virtual church services and craft lessons, to online school classes and workplace meetings, as well as convivial drinking and socializing sessions with friends of an evening, Internet-based videoconferencing finally came into its own.

Enduring memes were born, too: perhaps one of the most memorable being Sting's online, at-home jam with The Roots, aired on NBC's *Tonight Show*. The combo played a "quarantine remix" of "Don't Stand So Close To Me"



The ACM Publications Board held one of their annual face-to-face meetings via a series of Zoom meetings in July.

(surely one of the social distancing anthems of the lockdown) with improvised musical instruments.

Yet despite the blizzard of media and social media coverage, videoconferencing technologies are just one way of collaborating remotely and eliminating the need for personal, commuter, and business travel. Developments in computer science and robotics ensure other options are emerging, involving the use of telepresence robots, drones, augmented and virtual reality systems, and even holographic teleportation systems, which can put a three-dimensional, life-size version of you in a target room anywhere on Earth with a broadband connection.

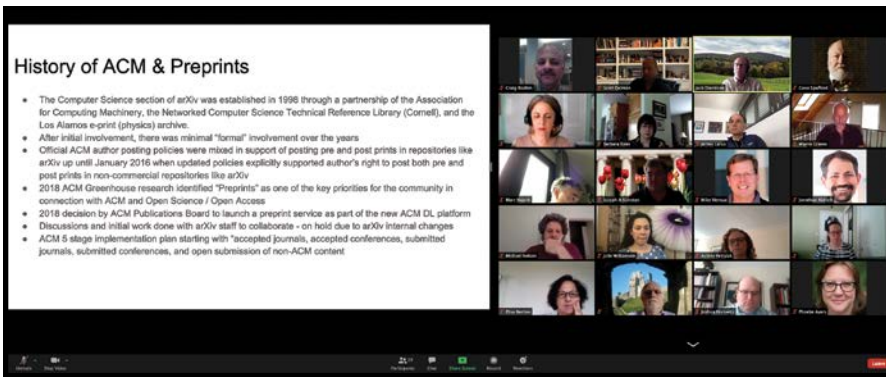
It is possible the still-unfolding societal changes wrought by the COVID-19 pandemic could see some of these alternate collaboration methods come to the fore in future lockdowns, perhaps due to new waves of infection from the SARS-CoV-2 virus, or mutated versions of it, or completely new pathogens.

Signs from a Webcam Sellout

It is fair to say, however, that emerging collaboration technologies will have a difficult time reproducing the surprising, runaway success videoconferencing and group video chat apps experienced as the early 2020 pandemic lockdowns took hold. Carman Neustaedter, who researches interactive and collaborative technologies at Simon Fraser University (SFU) in Vancouver, British Columbia, says the first sign he had that video chat was on an unexpected ascendant was when he found the shelves cleared in online tech stores.

"I suddenly found I couldn't buy a new webcam; all the decent ones were sold out, everywhere. It was a sign of how quickly people were adopting videoconferencing tools. But that immediate shortage of the hardware to support it really surprised me," says Neustaedter.

That sales rush was indeed a sign of things to come, and in the following weeks, videoconferencing growth was



stellar. For instance, Zoom alone grew from 10 million users worldwide in December to 200 million at the end of March, as the system moved from being an enterprise IT conferencing system to, frankly, something of an unlikely consumer product. Skype use was up 70% to 40 million daily users in March versus February, said a Microsoft spokesperson in Redmond, WA, who added the growth was accompanied by a 220% increase in Skype calling minutes. The professional collaboration app Microsoft Teams saw usage boosted 200%, from 900 million meeting minutes on March 16 to 2.7 billion on March 31, then a record for the platform, the spokesperson says.

Defending the Castle

However, that runaway uptake rate came at a price: security, at least for Zoom, was found wanting as users found trolls could join any password-free, open virtual meetings and shock people with violent pornographic images or other content designed to generally disgust and disrupt. These intrusions quickly got a name: Zoombombs. The U.S. National Aeronautics and Space Administration (NASA), Google, and SpaceX were among firms barring Zoom use by staff on security grounds. Some Zoom user data was spilled onto the dark web, too.

All this resulted in Zoom Video Communications of San Jose, CA—which says it had been too sidetracked laying on extra AWS and Oracle server capacity to cope with all its newfound global users—issuing a hasty security update to the app. This placed user security and privacy settings front and central, allowing consumers, for instance, to easily and clearly password-protect their meeting URLs. “We’ve always had all kinds of security features built

in, but normally enterprise IT teams would decide which features to enable and which to disable,” said Zoom CEO Eric Yuan in a Bloomberg interview. He accepted the firm “made a mistake” in not having made those features easily accessible and understandable by people at all levels, from consumers to professionals, from the start.

“With the shift to online video meetings, you do have to have this kind of barrier,” says Neustaedter of the locking-down of video chat systems to known, passworded invitees, and controlling who can be trusted to show content on screens. He says there is an interaction downside to that buttoned-down rigor: a lack of spontaneity, which reduces the chances that people experience idea-generating casual interactions with unexpected people—like an electronic equivalent of bumping into people “in the hallway, coffee room, or at the water cooler.”

Restoration By Robot

All is not lost, however; those valuable casual interactions can be recovered, to a degree at least, by switching from video chat apps to mobile telepresence robots. These wireless, remote-controlled, wheeled, flatscreen-display platforms go out into the world and act as a user’s eyes and ears and, controlled from a laptop or phone, allow users to explore a remote workplace, conference, mall, street, or pretty much anywhere with flat surfaces where there’s strong Wi-Fi. Because the robot can still randomly bump into people, they can experience the kind of informal verbal exchanges that video chat app users cannot, says Neustaedter.

Indeed, this London-based reporter can attest to the validity of this: at ACM SIGCHI’s Conference on Human Factors in Computing Systems in Denver,

CO, in 2017, I spent a day using a Beam telepresence robot from Suitable Technologies of Palo Alto, CA, and the experience was both liberating and unforgettable. On top of attending sessions, my droid could trundle up to, and let me interview, HCI researchers during the breaks, in corridors, outside session rooms, and at the coffee stall, and even record interviews, and all from my desk in London. “It’s those exchanges I think that telepresence robots are actually really good for,” says Neustaedter.

Telepresence robots are already fielded commercially and have found some strong healthcare roles in the COVID-19 pandemic. “Venues are using our mobile telepresence devices to monitor inside areas that have been converted to hospitals. They are also being used to maintain the security of those venues, and doctors are using them to visit and evaluate patients quickly,” said Steve Ernst, CEO of Event Presence, a company that provides Suitable’s Beam and BeamPro robots to event organizers.

If further pandemic waves strike, telepresence is ready to step up again as a virtual collaboration technology by allowing people prevented by lockdowns from traveling to conferences and trade-shows to attend them instead via robot. “We have developed a way to onboard up to 1,000 people on the Beam. We can now bring remote attendees not only to an event, but also directly to the participating event company’s showroom, corporate headquarters, or manufacturing plant anywhere in the world,” says Ernst.

Those 1,000 people, seeing and hearing through just one Beam robot, will tour 15 booths in a 1.5- to 2-hour session, which Ernst dubs a “Smart Tour,” taking in the booths of all the major vendors. For instance, at a flash memory event, remote users got to hear from the likes of Samsung, Toshiba, Sandisk, and Micron, Ernst says. “In fact, we don’t really need a traditional conference, with booths, to operate the robotic tour. We simply bring the attendees to where the client wants them,” he adds.

Getting Personal

Pandemic lockdowns affect private lives as well as business lives, and Neustaedter’s research group at SFU has been exploring telepresence’s prospects in many everyday homespun scenarios, too. For instance, one SFU

team gave one member of a couple in a long-distance relationship a robot, allowing their distant partner to be both robotically present and mobile in their apartment. For added “presence” effect, using a voice-activated system, the robot user could turn on the lights, switch on the vacuum cleaner, or turn on a slow cooker. The seven couples in the study found the robot lent them a stronger sense of sharing a home and a feeling of companionship versus communicating via video chat apps.

Still another SFU couples study found that sending a robot shopping with their partner was better than discussing what to buy on the phone: the remote user, embodied in the robot, mirrored their normal shopping behaviors (like which side of their partner they walk on, and whether they took off to look at goods alone) and were much better able to take part in joint purchasing decisions and accept responsibility for them. However, Neustaedter warns, “You just have to be really careful driving the robot, especially if you’re in a shop that has very breakable items.”

His group also cajoled some couples into having one partner trundle around as a telepresence robot around parks and urban areas to see how it fared as a walking partner on strolls taking in Vancouver’s beauty spots, sometimes undertaking undemanding outdoor pursuits like geocaching—that is, hiding curious objects at certain mapped GPS locations for other walkers to find. While that provided a great way for the remote user to experience the scenery, the robots have little awareness of what’s around them, from inclines they might topple down, to bicycles and people. Safety issues from potential collisions with members of the public, or damage to the expensive robot, were concerns; “We had to strap foam around the Beam robot just in case it fell, because they’re so expensive,” says Neustaedter.

Send in the Drone

Robots, however, are just one kind of telepresence proxy that a remote user in some future pandemic lockdown might send off to explore some facet of the world. Further into the future, multi-rotor drones are another potential telepresence proxy, with ever-smarter, geofenced autopilots in control of them,

and augmented and virtual reality tools providing user visualization.

A leading light in this field of drone-based telepresence is Xtend, based in Tel Aviv, Israel, which is honing its technology in the military space but has designs on the tourism, entertainment, mixed-reality gaming, and cinematography markets. Xtend’s overarching aim is to allow people to don an augmented reality headset and, with very little training, send a local drone into an area they want to explore (think of the Grand Canyon, the sights of Paris, or the verdant waterways of the Amazon jungle), much like its military customers currently send drones for short-range reconnaissance on the Gaza border.

It is part of a move to marry AR and VR with drones that market analyst Accenture is calling Extended Reality (XR), in which user location ceases to matter. Instead of displaying recorded or generated AR/VR content, XR will display live 360-degree content. It is still very much a future prospect for at-a-distance virtual collaboration, not least because the amount of data needing reliable backhaul between a drone and a highly distanced user, with low latency, will be high.

On top of that, telepresence robots and drones alike are not yet trusted devices: people coming across them do not know who is controlling them or to where they are broadcasting video footage. “We found certain stores would not allow our robots in,” says Neustaedter, during their shopping trials. In addition, VR-based solutions, he says, still need to solve one of its fundamental issues: the motion sickness experienced by many users.

That said, there is one emerging virtual collaboration technology that looks much more likely to be a near-term hit, at least for the deep of pocket: it’s called the Holoportl, a “single passenger,” life-size, human hologram transmission machine that’s a little bit bigger than a phone booth. Built and sold by PORTL of Los Angeles, the machine has a 4K transparent flatscreen display on the front of a 3D lightbox that, in a patent-pending projection process, displays a 3D video image of somebody elsewhere in the world, in very-near-real time. The projected person gets audiovisual feedback from the room they have

beamed into, and so can interact with the people there.

Holoportl proved its worth in the COVID-19 pandemic, says PORTL CEO and founder David Nussbaum. “At the moment, nobody can leave their homes, so we are being hired by doctors, speakers, educators, and politicians to beam them from the safety and security of their own homes or offices into classrooms and other places they need to be,” he says.

The Zoom phenomenon in the pandemic lockdowns did not go unnoticed: Nussbaum says PORTL’s roadmap includes Zoom-style holographic software, which will allow multiple people to appear in the booth in 3D. He also promises “a few new things that’ll blow people’s minds”.

“PORTL looks pretty cool,” says Peter Ladkin, a researcher in safety-critical systems at Bielefeld University in Germany, and a frequent user of videoconferencing systems on global standards making committees. “But it will make the cat crazy.”

Further Reading

Zoom reports massive user growth after lockdowns, *The Hill*, April 2, 2020, <https://thehill.com/policy/cybersecurity/490794-zoom-ceo-says-company-reached-200-million-daily-users-in-march>

Blog Post, How to tighten security on Zoom videoconference meetings, Graham Cluley blog, April 15, 2020, <https://www.grahamcluley.com/how-to-host-safer-zoom-meetings/>

Heshmat, Y., Jones, B., Xiong, X., Neustaedter, C., Tang T., Riecke, B., and Yang, L.

Geocaching with a Beam: Shared Outdoor Activities through a Telepresence Robot with 360 Degree Viewing, *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, April 2018, <https://dl.acm.org/doi/abs/10.1145/3173574.3173933>

Yang, L., and Neustaedter, C.

Our House: Living in a Long Distance Relationships through a Telepresence Robot, *Proceedings of the ACM on Human-Computer Interaction*, November 2018, <https://dl.acm.org/doi/abs/10.1145/3274459>

Holoportl: a single-passenger hologram system <https://portlhologram.com/products-services>

Paul Marks is a technology journalist, writer, and editor based in London, U.K.

© 2020 ACM 0001-0782/20/9 \$15.00

► James Grimmelman, Column Editor

Law and Technology

A Recent Renaissance in Privacy Law

Considering the recent increased attention to privacy law issues amid the typically slow pace of legal change.

UNTIL VERY RECENTLY, it was difficult to be an optimist about privacy in the U.S. Privacy laws in the U.S. have been notoriously ineffective. U.S. companies engage in rampant data profiling, from established giants like Google, to shadowy data brokers like Axciom, to headline-grabbing startups like Clearview AI. Edward Snowden's 2013 revelations about the scope of U.S. national security surveillance showed the extensive cooperation, and sometimes even active involvement, of private companies. In 2015, and again in 2020, the top European Union court invalidated the framework that allowed U.S. companies to export E.U. persons' data to the U.S., reasoning that U.S. privacy protections are too weak.

But both privacy talk and privacy law in the U.S. have shifted sharply toward increased protection. U.S. companies now often must comply with both European and California regulations. State after state has enacted new privacy laws, and Congress has been making the most serious attempts at

enacting a national privacy law in decades. Former U.S. Presidential candidate Andrew Yang even made data privacy a centerpiece of his campaign.

Privacy isn't dead, it turns out. It is very much alive. We are just learning, finally, how to talk about it.

The Data Privacy Dark(er) Ages

The U.S. has historically had a messy but extensive patchwork of privacy laws. The state privacy tort of "intrusion upon seclusion" prohibits obnoxious snooping like taking surreptitious photos in someone's house,

Privacy isn't dead, it turns out. It is very much alive. We are just learning, finally, how to talk about it.

and "public disclosure of private fact" prohibits publishing embarrassing secrets. There are some sector-specific privacy laws, such as the Health Insurance Portability and Accountability Act (HIPAA), which protects health data. State-specific laws, like California's anti-paparazzi law, have been adapted to address newer technologies such as drones. There are wire-tapping laws, some Fourth Amendment protections against surveillance by law enforcement, and general-purpose consumer protection laws that have recently been interpreted to hold companies to their published privacy policies.^{1,9}

What the U.S. does not have, however, is a comprehensive (or "omnibus") national data privacy law. This puts the U.S. out of step with much of the world, most strikingly the E.U., which now famously has the General Data Protection Regulation (GDPR). Unlike the U.S. patchwork, the GDPR applies to all personal data regardless of sector, and does not contain the kind of easy workarounds companies



have found in U.S. privacy laws. For example, U.S. companies that process personal health information point out HIPAA does not apply to them, because they do not technically provide health services or insurance. Others have argued they can ignore privacy laws as long as they work with “anonymized” data, even when it is easily reidentifiable.⁴

U.S. privacy law has mostly been built around the concept of “notice and choice,” which relies on giving individuals information (notice) about company practices and letting them make a choice (choice) about whether to hand over their data. All of us who regularly ignore privacy notices and click “I agree” to access websites know this does not work. Even broader versions of notice, such as requiring companies to notify consumers of data security breaches, often fail to incentivize good company behavior, since in reality consumers have few choices about which companies to use.

E.U.-style data protection, by contrast, puts in place substantive re-

quirements that “follow the data.”⁶ That is: under a true *data protection* regime, you can still get access to your information, request a correction or deletion, or require that a company stop processing your information, even if you initially voluntarily handed your information over to the company.

Perhaps the biggest structural weakness in U.S. privacy laws has been the maxim that once you hand your personal data over to somebody else, you assume the risk they will share it further. This rule does not fit everyday expectations about privacy: when you share your personal health information with your doctor, you do not expect that they will go tell your employer.⁷ But this reasoning runs throughout U.S. privacy law. It has gutted the privacy torts discussed here—courts have found that people do not have an expectation of privacy in information they have handed over to online platforms.³ It is only very recently (in a Fourth Amendment case about cellphone location tracking, *Carpenter v. United States*)

that courts have started to question this reasoning.

The irony is that we now think of as a “European” approach to privacy is actually very similar to some U.S. data privacy laws from the 1970s, like the Privacy Act of 1974, which regulates government databases. These early laws required transparency about how data is collected and used, restricted some kinds of sharing and use, and gave individuals rights to correct incorrect data and sometimes even have it deleted. In fact, these Fair Information Practice Principles (FIPPs), which now form the backbone of data protection laws around the world, arguably originated in the U.S. These principles were built upon the understanding that data privacy is largely about power, and that without transparency and accountability, the accumulation of data dossiers about individuals by governments and companies leads to huge power imbalances. These imbalances have consequences not just for individuals, but for democratic values and society at large.

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

So the U.S. does have privacy laws. But there are gaping holes between existing privacy laws; outdated understandings of reasonable expectations of privacy; and plenty of ways for companies to evade, avoid, or challenge the application of what privacy laws do exist.

But recently, things have started changing.

The Beginning of a Renaissance?

A line of Supreme Court cases addressing government surveillance heralds the recent shift in U.S. thinking about privacy: these cases recognize expectations of privacy in public, that we expect privacy even when we hand information over to technology providers, that data analysis can reveal sensitive information from individually innocuous data points.⁵ Over the past two years, a majority of U.S. states have either enacted or seriously proposed something more like European data privacy law. Federal lawmakers, too, have gotten in on the debate. What sparked this recent renaissance in U.S. privacy law?

The GDPR went into effect in May 2018. In part the GDPR was adopted to update existing European data protection law. In part, it was a reaction to deepening skepticism about U.S.-based companies and their practices. The GDPR made European data protection law *broader, stronger, and deeper*: it applies to a wider range of activity (broader), establishes stronger enforcement mechanisms (stronger), and includes additional substantive protections (deeper), compared to previous law.

The GDPR, unlike U.S. laws, covers nearly all processing of all kinds of personal data. It is quintessentially omnibus; it attempts to be both technology neutral and comprehensive. It “follows the data” in the sense that personal data receives numerous protections not just at the point when a consumer transacts with a business. That is, you do not waive the GDPR’s protections just by agreeing to let a company collect your data. Approximately half of the GDPR affords individuals a series of rights: of access, notification, correction, deletion, and more. The other half tells companies and government agencies what to do.

The GDPR, in short, establishes a data privacy compliance program, like the kind of thing one sees in highly regulated sectors such as banking. For example, many companies have to appoint a Data Protection Officer (DPO), who is responsible for ensuring compliance with the GDPR. Companies conducting “high risk” projects, such as extensive monitoring of public places, must conduct impact assessments and under some circumstances get government approval before proceeding. Companies must keep records about data processing, and build new technologies with data privacy in mind. These and other requirements establish a compliance system that aims to change both companies’ infrastructure and the substance of their decisions around data processing.

The GDPR has clearly had a global effect. It intentionally reaches data processing around the world, including companies that target European users on the Internet, or monitor the behavior of Europeans in Europe. The intentionally global reach of the GDPR, coupled with its threat of huge fines, has led companies around the world to adjust their privacy practices—and countries around the world to update their privacy laws.⁸

One theory of what has recently been happening in the U.S., with the startling uptick in proposed state and federal data privacy laws, is that the GDPR has spawned a host of imitators. When California enacted the California Consumer Privacy Act (CCPA) in June 2018, many journalists referred to it as “GDPR-lite.” To some extent this is true. Both the CCPA and recent state and federal proposals are fundamentally different from U.S. privacy laws that came before. Like the GDPR, they aim at *all* data processing, not just processing in particular sectors.

Also like the GDPR, many of the U.S. proposals follow the data. The CCPA, for example, famously allows California residents to opt out of the sale of their personal data, even when they have voluntarily given it over to a company. It also allows individuals to make access requests for personal data, providing an unprecedented degree of transparency over private sector data processing in the U.S.

But claiming the CCPA and follow-on state and federal proposals are the consequence of the GDPR is largely inaccurate.² The E.U. has long had data protection laws, and the U.S. has long decided to ignore them.

The CCPA was not enacted in response to the GDPR; it was enacted when a real estate billionaire, Alastair Mactaggart, coordinated with other privacy activists to put forward a data privacy law as a California ballot initiative. At the last minute, California's lawmakers begged for a compromise (it is very, very difficult to amend a law passed by ballot initiative), and passed the CCPA in order to get Mactaggart to withdraw his proposal.

The CCPA is also substantively different from the GDPR. First, and importantly, it exists against the backdrop of U.S. law, which prioritizes free speech and does not have constitutional protections for data privacy, unlike Europe, where data protection is enshrined as a human right. The CCPA is still largely an American-style transparency law, one that amplifies the "notice" in "notice and choice." The hope is that true transparency about data practices might lead consumers to behave differently, or lead to public outrage and new laws.


While it echoes a number of individual rights from the GDPR, the CCPA does not create structural requirements for companies. It does not require a data privacy officer, or records of data processing activity, or that companies minimize privacy violations and bake data privacy into the design of their technologies. The CCPA might obliquely trigger some changes in corporate practices, but mostly it relies on individuals to invoke their rights, rather than requiring companies to behave in particular ways.

Other states' proposals largely mimic the CCPA, not the GDPR. Some states just copy and paste it; others have established legislative committees specifically to study the CCPA in action. Other states are pushing forward with yet more sectoral privacy laws, rather than omnibus protections. These new laws address cybersecurity, biometric surveillance, and ISP privacy.

The flurry of state activity (with its risk of a high degree of variation) has

The story of U.S. privacy law is not yet at happily ever after. It is, however, meaningfully improving.

driven numerous privacy law proposals in Congress. There seems to be bipartisan agreement that there should be new federal privacy law. There is substantial disagreement, however, about whether that law should preempt (override) state laws, whether it should allow people to sue on their own behalf versus rely on government enforcement, and of course what should actually be in it.

The story of U.S. privacy law is not yet at happily ever after. It is, however, meaningfully improving. Major hurdles still remain, including significant First Amendment challenges (do privacy laws violate rights to free speech?). But in a very short time period, compared with the usually glacial pace of legal change, the paradigm has shifted. Data privacy law is no longer a matter of whether, but what and when. 

References

1. Bamberger, K.A. and Mulligan, D. Privacy on the books and on the ground. *63 Stan. L. Rev.* 247 (2010).
2. Chander, A., Kaminski, M.E., and McGeeveran, W. Catalyzing privacy law. *105 Minn. L. Rev.* (forthcoming 2020).
3. Citron, D. Mainstreaming privacy torts. *98 California Law Review* 1805 (2010).
4. Hartzog, W. and Rubinstein, I. The anonymization debate should be about risk, not perfection. *Commun. ACM* 60, 5 (May 2017), 22–24; DOI: 10.1145/3068787
5. Joh, E. Increasing automation in policing. *Commun. ACM* 63, 1 (Jan. 2020), 20–22; 10.1145/3372912
6. McGeeveran, W. Friending the privacy regulators. *58 Ariz. L. Rev.* 960 (2016).
7. Nissenbaum, H. *Privacy in Context: Technology, Policy, and the Integrity of Social Life*. Stanford Law Books, First edition, 2009.
8. Schwartz, P.M. Global data privacy: The EU way. *NYU L. Rev.* 771 (2019), 94.
9. Solove, D.J. and Hartzog, W. The FTC and the new common law of privacy. *Colum. L. Rev.* 583 (2011), 114.

Margot Kaminski (margot.kaminski@colorado.edu) is Associate Professor at the University of Colorado Law and the Director of the Privacy Initiative at Silicon Flatirons, Boulder, CO, USA.

Copyright held by author.



Digital Threats: Research and Practice

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



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

► Terry Benzel, Column Editor

Security

Autonomous Vehicle Safety: Lessons from Aviation

How more than 25 years of experience with aviation safety-critical systems can be applied to autonomous vehicle systems.

AUTONOMOUS VEHICLES SEEM to hold great promise for relieving humans of the boring task of guiding a car through congested traffic or along a monotonous turnpike, while at the same time reducing the annual highway death toll. However, the headlong rush to be the first to market, without adequate considerations of life-critical control system design, could cause irreparable public harm and ultimately set back the promise of autonomous driving. With the current goal of being at least as safe as human driving, espoused by business leaders as well as some regulatory agencies, the annual death toll attributed to automation killing innocent people, just in the U.S., would be approximately 36,500 per year or 100 per day. Think about that for a minute!

This column highlights this important dependability need, the dire consequences of falling short, and how leveraging the knowledge gained by the aviation industry in operating safety-critical flight control systems without fatalities for over a quarter century can help avoid this outcome.

SAE International has defined six levels of autonomy for on-road motor vehicles in SAE J3016, where an updated graphic summary of the levels was released last year.¹⁰ An excerpt of this visual describing “What does a human in the driver’s seat have to do?” is displayed in the figure on p. 29.



Numerous automobile companies² are racing to be the first to market. Ford says it will have an L5 vehicle in operation by 2021. More ambitiously, Toyota announced in February 2019 that it plans to have a self-driving vehicle (“the most intelligent supercomputer on wheels”) available for purchase within a year. To achieve this objective, Toyota’s vice president in charge of software says “Our goal is to teach a Silicon Valley mindset here.”

Waymo, a spinoff of Google’s self-driving car project, is already rolling out a fully autonomous ride-hailing service in Phoenix, AZ, and has run road tests of autonomous “big rig” trucks in Atlanta, GA. Volkswagen

announced it would field a fleet of self-driving vehicles by 2021, and has recently teamed with Ford. As early as 2016, Tesla announced all cars it produces have the hardware needed for L5 driving capability; evidently it is just a small matter of programming.

Every manufacturer has a safety and security argument to go along with its autonomous vehicle control system (AVCS) plans. For cybersecurity, they may point to guidelines published by the U.S. Department of Transportation¹² or best practices published by the Auto-ISAC.¹

Twelve leading companies have collaborated on a 150-page white paper “Safety First for Automated Driving,”

working toward “industry-wide standardization of automated driving,”¹¹ which builds on guidelines and standards worldwide, including some still under development.⁴ It describes both development processes aimed at achieving “safety by design” and verification and validation of elements and systems at L3 and L4 autonomy. Use of Deep Neural Nets to implement safety-related elements is addressed in an appendix in the white paper.

This effort appears comprehensive and a strong step in the right direction, but, as its authors recognize, it is a work in progress. Further, as observers have noted,³ it is strictly a voluntary initiative among a set of companies, outside of the usual channels for standards development. Moreover, the effort is generally focused on standards companies apply to their internal design and development processes without external review or certification. They do not provide a quantifiable level of safety or security performance in terms of, for example, expected failure rate of control systems per hour or mile of operation.

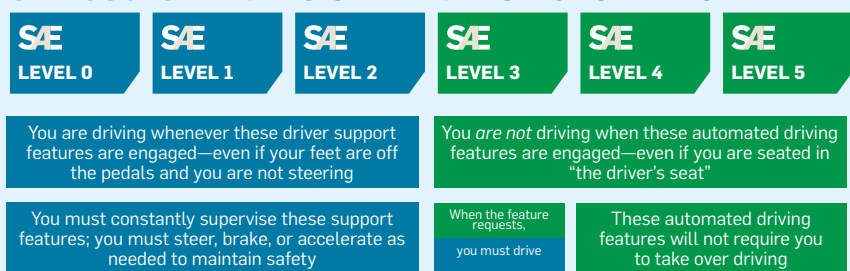
While the quest for autonomous vehicles may seem novel, we have been here before. Aviation provides a successful model with many parallel challenges, including hard real-time control, dependability commensurate with the adverse economic and life-threatening consequences of failures, scalability, affordability, and non-technical factors such as certification and governance. A case study of the most relevant avionics application—full authority, full time, fly-by-wire (FBW) aircraft flight control systems (FCS) is instructive.

The dependability requirements, including safety and reliability, were defined only after several years of discussions with NASA and the FAA in the mid-1970s. NASA was sponsoring research in FBW FCS to enable commercial aircraft to be more fuel efficient by taking advantage of statically unstable aircraft designs. An aircraft of this design would require computer control to maintain its stability.

The FAA’s initial proposal was to mandate FCS to be as reliable as the wings of the airplane, that is, the FCS should never fail. The rationale was that wings never fall off and the FCS is

SAE levels L0–L5.

SAE J3016™ LEVELS OF DRIVING AUTOMATION



just as integral to an aircraft’s safety as its structure. However, from an engineering viewpoint, that was not a requirement that one could design to.

After further discussions, the FAA defined a quantitative requirement in a one-page memo to NASA, in May 1974. Key excerpts of the memo are quoted below.

“Section 25.1309 of the Federal Aviation Regulations requires that airplane systems be designed so that occurrence of any failure conditions (combinations of failures in addition to single failure considerations) which would prevent the continued safe flight and landing of the airplane is extremely improbable. The FAA has accepted substantiating data for compliance with that requirement which shows by analysis that the predicted probability of occurrence of each such failure condition is 10^{-9} per hour of flight.”

“We further believe that failure of all channels on the same flight in a “fly-by-wire” flight control system should be extremely improbable.”

In the late 1970s, Draper Lab and SRI International produced two competitive designs, Fault Tolerant Multi-Processor (FTMP) and Software-

Implemented Fault Tolerance (SIFT), respectively, under contract with NASA Langley Research Center. These designs were realized in flight-worthy computers by Collins Avionics and Bendix, respectively, and subjected to many theoretical and experimental tests. The architectures relied on hardware and process redundancy; real-time fault detection, identification and reconfiguration; and software fault-tolerance, among many other dependability-enhancing techniques.

Verification and validation techniques included proofs of correctness, hardware and software fault injections, measurement of computer response to such events, and Markov reliability models with some of the model parameter values determined via experimentation.

These pioneering research and development efforts resulted in fundamental architectures, designs, theories, and certification methods that continue to shape today’s FBW systems.⁵

What Can We Learn from the Aviation Example With Respect to Autonomous Vehicle Dependability Requirements?

First, vehicle control imposes hard real-time requirements, and stringent low latency, just as FCSs do. A control or communication failure during critical vehicle maneuvers can lead to a cascading series of life-threatening accidents. The autonomous control system must detect any consequential fault and take corrective action within fractions of a second to keep the vehicle under control.

Although aircraft FBW systems must function for the duration of the flight, ground vehicles have the luxury of pulling off the road in case of a malfunction.

While the quest for autonomous vehicles may seem novel, we have been here before.

Still, the control system must, at a minimum, continue to function correctly for the time it takes to maneuver the vehicle to a safe place while also configuring itself into a safe state.

The control system must continue to function correctly after a fault, that is, it must be designed so there are no single points of failure. In case of faults or errors, the system must compensate, and still produce correct results in a timely manner long enough to reach a safe place and configure the vehicle into a safe state. A graceful degradation to a limited functionality Fail-Operational requirement would seem to be adequate.

Aircraft FBW systems use masking, redundancy, and sophisticated reconfigurations to continue to provide full functionality after a fault. Autonomous vehicle control systems can be simpler and less expensive by providing a limited Fail-Operational architecture. Some inspiration can be gained from early aviation experience.

First-generation jumbo-jets, in the early 1970s, used computers to provide “all-weather” auto-land capabilities for Cat IIIB conditions: zero visibility, zero ceiling. They had safety and reliability requirements and mission times very similar to those for au-

tonomous vehicles. There could be no single point of failure and the system had to be operational for only several minutes: the duration of approach and landing. The architectures ranged from dual redundant self-checking pair of computers (Lockheed TriStar L-1011), to duplex channels, each with dual fail-disconnect computers for pitch, roll, and yaw axes (Douglas DC-10) to triple redundant analog computers (Boeing 747).

How Does AVCS Reliability Correlate With the FAA's Mandated Failure Rate of 10^{-9} per Hour for FCS?

Table 1 summarizes recent U.S. motor vehicle death rates, which translate to a fatality rate of 5×10^{-7} /hour per vehicle, assuming an overall average speed of 40 MPH. What would be an acceptable failure rate of an L5 control system? Table 2 illustrates the effects of some alternative rates.

Many people in the industry say just slightly better than status quo would save lives. But that would mean nearly 100 people being killed by road vehicles every single day. Would society accept such mayhem attributed to machines? Recall the public reaction when a single pedestrian was

killed by an autonomous Uber in Tempe, AZ.¹³

The FAA set the failure rate for FBW systems at two to three orders of magnitude *smaller* than that of human pilots who are highly trained for safety. Shouldn't society set similar goals for autonomous vehicle safety? Even a failure rate of 100 times better than an average driver, whose safety behavior is unlikely to approach that of a pilot, would still result in 365 U.S. deaths per year attributable to AVCSs. By comparison, no single death has been caused by FBW systems in over a quarter century of operations worldwide.

This is a once-in-a-century opportunity to ride on the revolutionary re-making of ground transportation to make it as safe as aviation, a gift to humanity worldwide.

Regarding safety standards for road vehicles, ISO 26262 concerns the functional safety of on-vehicle electrical and electronic (E/E) systems, where components are ranked according to an ASIL (Automotive Safety Integrity Level), the most critical being level D. However, it should be noted that ISO 26262 (and more generally the auto industry) shies away from quantitative safety requirements, particularly with regard to fatalities.

ICCQ

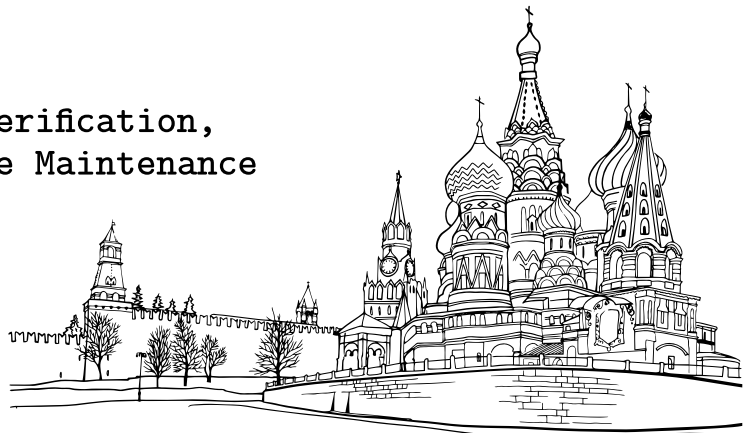
www.iccq.ru

The First International Conference on Code Quality
in cooperation with IEEE Computer Society

Moscow, Russia
27 Mar 2021

Static Analysis, Program Verification,
Bug Detection, and Software Maintenance

CfP closes:
4 Dec 2020



While drafting ISO 26262, the automotive industry recognized that, in addition to faults in E/E equipment, unsafe behavior could likewise be caused by faults in the specified functionality, spinning off a complementary functional safety standard ISO/PAS 21448 or SOTIF (Safety of The Intended Functionality). The latter presumes the realized system is fault-free and focuses on reducing safety risks due to insufficiencies in the intended behavior. However, these are process-related requirements and not quantitative. Furthermore, the automotive industry “self-certifies” compliance to these standards. Self-certification of the Boeing 737 MAX led to the MCAS system, at the center of the two crashes, being declared non-safety-critical.⁷

Additionally, a safety-assurance case must go beyond the simplistic, non-scientific miles driven and fatal accidents recorded for autonomous vehicles. The data must include the envelope of corner/edge cases explored, realism of testing conditions, unplanned disengagements, and incorrect decision making. Further, this effort should be complemented with analytical models, fault injections, and proofs-of-correctness, where appropriate.

What Can We Learn from Aviation Human Factors?

Much of the automotive industry is moving successively from L0 (fully manual) to L5 (fully autonomous), believing this step-wise increase in autonomy is the safest way to proceed. Counterintuitively, this approach poses a real human-factors challenge. L3 will be a semi-autonomous mode where routine driving is performed by the control system and the human driver’s role will be to intervene in emergencies/malfunctions.

In the cockpit, highly trained pilots are primed to recognize unusual situations quickly and take corrective action. Recurring simulator training focuses on dealing with emergencies. Cockpits and flight control systems are designed to optimize human-machine interaction in the safest way possible.

Ordinary driver’s license certification requires no such training, nor is it reasonable to expect the public at large to perform at this level. The rarity of

Table 1. U.S. motor vehicle accidents for 2018.⁹

Year	Deaths	Vehicle Miles Traveled (VMT) (Billions)	Fatalities/100 Million VMT	Population	Fatalities/100K People
2018	36,560	3,174	1.13	327,200,000	11.17

Table 2. Projected death rates for autonomous vehicles.

Case	Safety relative to current manual benchmark	Failure Rate (per hour)	Annual Deaths caused by Control System (US)	Deaths/Day (US)
1	Same as	5×10^{-7}	36,560	100
2	10X better	5×10^{-8}	3,656	10
3	100X better	5×10^{-9}	365	1

emergencies requiring human intervention makes it nearly impossible to keep the driver, who is busy doing other things, sufficiently engaged to take over within a fraction of a second of an alert.

The proper balance between fully automated functions and relying on the pilot to deal with emergencies continues to draw attention in aviation, with Airbus favoring the former while Boeing has favored the latter, up until MCAS.⁶

The automotive industry needs to consider the path to L5 very carefully. It might be worthwhile changing course, skipping L3 altogether (Ford, Waymo, and Audi had announced they would skip L3, but Ford has since reversed its position⁸), and designing a fully autonomous vehicle, bypassing the fraught nature of semi-autonomous controls.

Conclusion

The interest from both the industry and the driving public in autonomous vehicles is considerable and justified. But fielding technology that promises huge societal impact without a serious consideration of its dependability requirements is unsound. “Better than the average driver” is a particularly weak requirement. Engineers have proven they can do much better than that in other fields. Society needs to provide the incentive for them to do what needs to be done in the automotive domain. □

References

1. Automotive Information Sharing and Analysis Center (Auto-ISAC). *Best Practices Executive Summary*. (July 2016); <https://www.automotiveisac.com/best-practices/>
2. DeNisco-Rayome, A. Dossier: The leaders in self-driving cars. *ZDNet* (Feb. 1, 2018); <https://www.zdnet.com/article/dossier-the-leaders-in-self-driving-cars/>

3. Eliot, L. Discussing safety first for automated driving with Aptiv’s Karl Iagnemma. *Forbes*. (July 19, 2019); <https://www.forbes.com/sites/lanceeliot/2019/07/19/discussing-safety-first-for-automated-driving-with-aptiv-karl-iagnemma/>
4. ISO/SAE 21434. Road vehicles—Cybersecurity Engineering; <https://www.iso.org/standard/70918.html> and <https://www.sans.org/cyber-security-summit/archives/file/summit-archive-1525889601.pdf>
5. Lala, J. History and future directions of mission- and safety-critical digital avionics. In *Proceedings of the AIAA Guidance, Navigation & Control Conference* (Aug. 2013), Boston, MA; <https://doi.org/10.2514/6.2013-5206>
6. Langewiesche, W. What really brought down the Boeing 737 Max? *New York Times Magazine*. (Sept. 18, 2019); <https://www.nytimes.com/2019/09/18/magazine/boeing-737-max-crashes.html>
7. Levin, A. and Beene, R. Max disasters fuel outcry over how FAA let Boeing self-certify. *Bloomberg Markets* (Dec. 3, 2019); <https://www.bloomberg.com/news/articles/2019-12-03/max-disasters-fuel-outcry-over-how-faa-let-boeing-self-certify>
8. Martinez, M. Ford rethinks Level 3 autonomy. *Automotive News Europe*. (Jan. 20, 2019); <https://europe.autonews.com/automakers/ford-rethinks-level-3-autonomy>
9. National Highway Traffic Safety Administration (NHTSA) Traffic Safety Facts Research Note. (Oct. 2019); <https://www.nhtsa.gov/traffic-deaths-2018>
10. SAE Standards News: J3016 automated-driving update. (Jan. 2019); <https://www.sae.org/news/2019/01/sae-updates-j3016-automated-driving-graphic>
11. *Safety First for Automated Driving*. Technical Report. (July 2019); <https://www.aptiv.com/docs/default-source/white-papers/safety-first-for-automated-driving-aptiv-white-paper.pdf>
12. USDOT Automated Vehicle Activities. (Apr. 2020); <https://www.transportation.gov/AV>
13. Wakabayashi, D. Self-driving Uber car kills pedestrian in Arizona, where robots roam. *New York Times* (Mar. 19, 2018); <https://www.nytimes.com/2018/03/19/technology/uber-driverless-fatality.html>

Jaynarayan H. Lala (jay.lala@rtx.com) is a Senior Principal Engineering Fellow at Raytheon Technologies, San Diego, CA, USA.

Carl E. Landwehr (carl.landwehr@gmail.com) is a Research Scientist at George Washington University and a Visiting Professor at University of Michigan, Ann Arbor, MI, USA.

John F. Meyer (jfm@umich.edu) is a Professor Emeritus of Computer Science and Engineering at University of Michigan, Ann Arbor, MI, USA.

This Viewpoint is derived from material produced as part of the Intelligent Vehicle Dependability and Security (IVDS) project of IFIP Working Group 10.4.

Copyright held by authors.



Peter J. Denning

DOI:10.1145/3411055

The Profession of IT Avalanches Make Us All Innovators

Avalanches generate enormous breakdowns. The practices of innovation adoption may be just what you need to resolve them.

THE WORD “AVALANCHE” brings up vivid images of destructive masses of snow suddenly sweeping down wintry mountain slopes, deadly cascades of ice and rock crushing everything beneath them. Avalanches are as unpredictable as they are destructive. No one can say with any certainty when or if an avalanche will occur or how severe it might be. The best mountain authorities can do is shut down a large area when they judge avalanche danger to be high.

Economists have adopted avalanche as a metaphor for disruptive changes that sweep through an economy. Economic avalanches can cause massive losses, wiping out professions, jobs, and wealth. Severe stock market crashes are avalanches. The housing bubble of 2007–2008 precipitated an avalanche that hurt many people badly. In the aftermath, a natural reaction is to point many fingers of blame for the calamity. This gives way to the more constructive reaction of trying to learn from what happened so as to be prepared if it happens again.

There are two kinds of avalanche: the slow-moving selective kind, and the fast-moving all-encompassing kind. The first kind is much more familiar. Take the Internet. The world of 1990, before the World Wide Web exploded the size and reach of the Internet, was quite different from today’s world. Today’s commonplace things—such as smartphones, instant worldwide com-



munications, video and music streaming, online commerce, digital currency, Facebook, Amazon, Google, Microsoft—either did not exist or were too small to notice. Many ways of doing business and many professions have disappeared since those days, and many new businesses and professions have sprung up in their place. The Internet was a slow-moving avalanche. At the start of the Internet revolution, a few visionaries saw the possible changes, but few had any idea of the extent of change that was going to happen.

The Cloud has been another slow-motion avalanche that facilitated the

birth of many new companies. A decade ago, most startup businesses had to include an IT department in their business plans. Big computing power was available only on supercomputers and was way too expensive for many founders to afford. Now Amazon, Microsoft, Google and others have established vast networks of processing and storage servers around the world, delivering computing power and memory cheaply like a utility. Anyone can rent the computing power they need at a price they can afford. Getting the IT needed for a startup has never been easier.

IMAGE BY OLGA GAVRILOVA

No sector is immune to possible avalanches. In 2013, Michael Barber and colleagues warned an avalanche loomed for higher education.¹ University leaders downplayed that assessment because they believed everyone wants and needs education. In 2017, Tony Seba bet on an avalanche in the energy sector from the transition from carbon-powered industry and transportation to solar and electric powered.⁴ No one was prepared when the pandemic triggered an economic avalanche as bad as the Great Depression of the 1930s, which spread rapidly into education and energy. The secondary avalanches forced many universities into bankruptcy and collapsed world oil prices.

From all these examples, we can characterize an avalanche as a disruptive change of conditions and practice that sweeps through a social community, taking with it many professions, identities, jobs, and wealth. This is not the same as the Black Swan phenomenon.⁵ A Black Swan is a surprise event that no one predicted. An avalanche is a cascading, chaotic process that may be triggered by a Black Swan event. The few visionaries who see the warning signs cannot predict if or when or if an avalanche will occur. Few people are prepared when it strikes. When it does, most people get disoriented and unsettled not only because of their losses but also impenetrable uncertainty about what comes next.

Many startup companies speak of their ideas as “disruptive innovations” that they hope will trigger avalanches—often called “viral adoption.” They hope their inventions will sweep them to riches and wipe out competitors. Unwilling to be disrupted, competitors take preventive countermeasures. Often the initial proposal or the countermeasure fails. Brave talk on both sides hides the underlying fact that no one can predict whether an avalanche will occur. Wishful thinking and heroic hype do not improve predictive accuracy.

Since the early 1980s with the founding of complexity theory and the Santa Fe Institute, the sharpest minds of mathematics and physics have sought to build a mathematical theory that would among other things predict the onset and extent of ava-

Avalanches force innovation. When old practices have been swept away, we have no choice but to adopt new ones.

lanes in chaotic processes. They developed beautiful and elegant theories. One of their surprising conclusions is that complexity theory explains past events in chaotic processes but cannot predict the timing or severity of future events. Without mathematics or science to predict avalanches, how can we be prepared?

A Virus that Defied Modeling

The COVID-19 pandemic came as the greatest medical, economic, social, and political shock since 1940. It has been an all-encompassing avalanche that has left no part of the world untouched. It transformed the world in just a few months. Governments are reeling as they search for policies that will contain the virus and stanch the debt they have taken on to survive. Governments say their policies are “science based,” meaning mostly computational modeling. Yet there are enough disagreements between the models and missed predictions that policymakers do not know whether they can trust the models, the data that powers them, and even the modelers themselves.

Complexity science tells us it is no surprise the models do not do well. It can explain the past of a chaotic process but not predict its future. The best models can do is play out “what-if” scenarios that compute a probable future state based on assumptions of how the model parameters will unfold. Policymakers are in the uncomfortable position of not knowing what forecast to trust and yet having to choose one to justify their responses.

Governments have been criticized for being unprepared, despite warnings based on previous coronavirus

outbreaks that a new one could appear at any moment given the encroachments of humans into wild-animal habitats and the increasing number of genetic experiments with viruses. The preparations would include having test kit, protective gear, and confinement plans at the ready. But governments are constitutionally inclined to give priority to urgent issues and downplay warnings of unlikely events. This pandemic is likely to move many governments to prepare for future pandemics and know there is popular support for doing so.

The COVID-19 avalanche precipitated other avalanches about which we have been warned but took no preparatory action. Educators thought that education was resistant to collapse because everyone wanted it. Yet with campuses closed and foreign students locked out, university revenues have plummeted, faculty are furloughed, and programs cut, and it now seems likely that hundreds of smaller universities may disappear by the end of 2021 and many others will be severely impaired. Carbon-based energy producers were confident that coal, oil, and gas would be staples indefinitely. But the coronavirus pandemic stopped most travel and precipitated the collapse of oil prices and a surge of interest in solar, wind, and other non-carbon energy technologies. Many oil producers face bankruptcy. The suffering has exposed social inequities many people no longer wished to tolerate, precipitating further avalanches of social unrest over oppression around the world. Everyone began to realize there would be no going back—what they considered normal will never return.

What You Can Do

Avalanches force innovation. When old practices have been swept away, we have no choice but to adopt new ones. Here are a few examples of innovations people were forced to adopt after the pandemic struck. Managers of closed businesses avoided furloughs by authorizing employees to work from home. Suddenly online meeting platforms such as Zoom, Connect, WebEx, Teams and became hot for telework. Schools started using these plat-



Association for
Computing Machinery

2018 JOURNAL IMPACT
FACTOR: 6.131

ACM Computing Surveys (CSUR)

ACM Computing Surveys (CSUR) publishes comprehensive, readable tutorials and survey papers that give guided tours through the literature and explain topics to those who seek to learn the basics of areas outside their specialties. These carefully planned and presented introductions are also an excellent way for professionals to develop perspectives on, and identify trends in, complex technologies.



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

The avalanche has also generated some wicked problems where finding a workable innovation is not easy.

forms to continue classes while students were confined to home. Educators learned in a very short time how to use them as a wholly new teaching medium. Shops and restaurants learned to operate curbside service with orders taken online. Unable to visit in person, families started meeting on these platforms. Many of these new practices will persist after the pandemic is over. More businesses will telework, schools will use distance learning, shops will continue social distancing, and distributed families will continue meeting.

The avalanche has also generated some wicked problems where finding a workable innovation is not easy. One of the toughest has been the need for homeschooling when both parents are home with the children. There is no school or day-care center to tend the children while the parents work. No good solution has yet been found. Another tough problem is finding new customers. How do professionals get new customers when professional gatherings are severely restricted? What new offers would attract customers?

Over a decade ago, my colleagues and I designed an approach for intentionally generating innovations, defined as adoption of new practices in a community.² We found a set of eight essential practices by which innovators generate the commitments needed to adopt an innovation. They are:

- ▶ *Sensing*—giving voice to a concern over a breakdown in the community;
- ▶ *Envisioning*—design a compelling story about a future without the breakdown;
- ▶ *Offering*—committing to do the work to produce that future;
- ▶ *Adopting*—gaining commitments

from early adopters to join the innovation for a trial period;

- ▶ *Sustaining*—gaining commitments from majority adopters to join the innovation for an indefinite period;
- ▶ *Embodying*—working with the community until the new practice is fully embodied, ordinary, and transparent;
- ▶ *Navigating*—moving ever closer to the goal despite surprises, contingencies, and obstacles; and
- ▶ *Mobilizing*—building a network of supporters of the innovation from within dispersed communities

We offered these as optional skills for those who wanted to increase their personal, team, or business success rates considerably higher than the prevailing industry average of 4%. But the same skills are no longer optional when you are forced into finding innovations to live in the new environment generated by an avalanche. The new environment fosters a plethora of new concerns that can be addressed with new offers. These leadership practices will open pathways to move forward, navigate among many options and obstacles, and mobilize a network to join with you in making it happen. You do not have to be stymied by massive uncertainty and overwhelm. We have taught these skills to hundreds of students and clients. Over two-thirds of our students have produced significant innovations in their communities. Compare that with the meager 4% success rate we were used to before the pandemic. The practices work!

When confronted with the need to devise new practices to live post avalanche, these skills may be exactly what you need. ■

References

1. Barber, M., Donnelly, K., and Rizvi, S. An avalanche is coming: Higher education and the revolution ahead. 2013; <https://bit.ly/3hfHPSP>
2. Denning, P. and Dunham, R. *The Innovator's Way*. MIT Press, 2010.
3. Denning, P. and Lewis, T. Uncertainty. *Commun. ACM* 62, 12 (Dec. 2019), 26–28.
4. Seba, T. Clean Disruption—Why Conventional Energy and Transport will be Obsolete by 2030. 2017. Video recording available from <https://www.youtube.com/watch?v=4hoB7HN4B0k>
5. Taleb, N.N. *The Black Swan* (2nd Ed). Random House, 2010.

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

Copyright held by author.

Viewpoint

Integrating Management Science into the HPC Research Ecosystem

How management science benefits from High Performance Computing.

HIGH PERFORMANCE COMPUTING (HPC) refers to the practice of aggregating computing power in a way that delivers much higher performance than one could get out of a typical desktop computer or workstation in order to solve problems in science, engineering, or business. HPC is usually realized by means of computer clusters or supercomputers. Interestingly, the June 2019 edition of the list of TOP500 supercomputer sites marks a milestone in its history because, for the first time, all 500 systems deliver a petaflop or more. But looking at the development of single core performance reveals that it has stopped growing due to heat dissipation and energy consumption issues. As a result, substantial performance growth has started to come only from parallelism, which, in turn, means that sequential programs will not run faster on successive generations of hardware.

Many academic disciplines have been using HPC for research. For example, HPC has become important in systems medicine for Alzheimer's research, in biophysics for HIV-1 antiviral drug development, in earth system sciences for weather simulations, in material science for discovering new materials for solar cells and LEDs, and in astronomy for exploring the universe. Usage statistics of academic su-



percomputing centers, which have started offering HPC as a commodity good for research, show a high diversity of scientific disciplines that make use of HPC in order to address unresolved scientific problems with computational resources that have been unavailable in the past. These statistics are also an indicator for other scientific disciplines that have hardly used HPC to solve their research problems. Undoubtedly, several of these other research areas will hardly benefit from HPC, for example, because

their research is not computationally intensive. But there are also areas where using massive computational resources can help solving scientific problems. One of these areas is Management science (MS), including its strong link to economics.

Management Science Benefits from HPC

Management science refers to any application of science to the study of management problems. Originally synonymous with operations research, MS

has become much broader in terms of problems studied and methods applied, including those related to econometrics, mathematical modeling, optimization, data mining and data analytics, engineering, and economics. Its multidisciplinary and often quantitative nature makes it a promising scientific field for HPC. The potential of applying HPC to MS includes the improvement of efficiency (“solving a problem faster”), effectiveness (“solving a problem of larger size and/or with enhanced quality”) and robustness (“solving a problem in a way that makes the solution robust against changes”).

Although MS very rarely occurs in HPC usage statistics, the potential of HPC for solving problems in MS is being tapped in several of its subfields. A first example are fraud detection services in finance provided by Bertelsmann Arvato Financial Solutions.⁸ Machine learning approaches are integrated into the real-time analysis of transaction data. These approaches are based on the development of self-learning analytical models from past fraud cases for early recognition of new fraudulent cases. From a technological perspective, the Hadoop framework and the Microsoft Azure cloud infrastructure are used.

A second example is the provision of the cloud-enabled software PathWise (provided by Aon) that uses HPC to manage financial guarantee risk embedded in life insurance retirement products.² Applying HPC capabilities allows reducing time required to evaluate policies from hours and days to minutes through a variety of approaches, including the parallelization of Monte-Carlo simulations.

A third example of benefiting from HPC when solving problems in MS is the parallelization of algorithms for solving discrete optimization problems in operations research. In Raucher and Schryen,⁹ the authors parallelize a branch-and-price algorithm for solving the “unrelated parallel machine scheduling problem with sequence-dependent setup times.” Their computational experiments conducted on a large university cluster used MPI (Message Passing Interface) to connect 960 computing nodes. Results on efficiency show their paral-

lization approach can even lead to superlinear speedup.

Further examples can be found in economics, where dimensional decomposition for dynamic stochastic economic models has been implemented on a supercomputer³ and equilibria in heterogeneous agent macro models have been solved on HPC platforms.⁷ In data analytics, HPC has been applied not only to detect financial fraud but also to solve problems in social network analysis (for example, Zhang et al.⁷) and in global health management (for example, Juric et al.⁶). Overall, the applicability of HPC to problems occurring in MS shows a large methodological diversity, with methods from machine learning and artificial intelligence, simulation, and optimization being included, and the identification of many computational problems in MS that may be solved through HPC is not difficult. However, what turned out to be difficult is bringing HPC to MS and fostering the position of MS in the HPC research ecosystem. This deployment essentially targets exploiting technical HPC capabilities for solving managerial problems through raising awareness of this potential in the MS community; implementing HPC-related education for MS researchers; and providing software development support with frameworks, libraries, programming languages, and so forth, which focus on solving specific types of problems occurring in MS.

Despite promising applications of HPC in different areas of MS, its deployment in MS is far away from being an established and well-known research approach.

Raising Awareness of HPC Benefits for Management Science

Despite promising applications of HPC in different areas of MS, its deployment in MS is far away from being an established and well-known research approach. This subordinate role of HPC in MS is reflected in different phenomena, which, at the same time, point to opportunities for raising awareness of HPC benefits in MS and for informing researchers that HPC is not identical to “High Performance Technical Computing.”

In the MS community, only very rarely are (special issues of) journals, workshops, or conference tracks dedicated to HPC-based research. A few examples exist (for example, Schryen et al.¹⁰) but much more of these efforts to identify and communicate the potential that HPC brings for MS and to foster corresponding research is needed. In addition, MS departments may profit from introducing HPC to master’s and Ph.D. students and young scientists by offering HPC courses and HPC summer/winter schools, in close cooperation with HPC sites of universities. Such courses and schools are of particular benefit when MS students and researchers can bring their own problems, algorithmic blueprints, or codes, and learn how to think, design, and implement parallel. In short, we need a more thorough computational and HPC-oriented education of MS students, who are the MS scientists of tomorrow.

HPC sites at universities and research institutions today generally focus on their current “power users,” who are scholars from the natural sciences, engineering disciplines, medical sciences, among others. Often, the expectation of these sites on users’ expertise includes a clear understanding of how HPC works technically (for example, shared vs. distributed memory), which parallel programming paradigms exist (threads, processes, and so forth), which libraries and APIs are state-of-the-art (OpenMP, MPI, CUDA, and others), and how parallel programming should be done (take care of data races, deadlocks, and so forth). Unfortunately, MS researchers often do not (need to) have this deep knowledge for understanding their research

field. This gap between expected and existing knowledge of HPC finally prevents MS researchers from tapping the potential that HPC might bring to their research. HPC sites should contribute to closing this gap by providing high-educational courses dedicated to MS. It might be helpful to bridge the gap by establishing and financing (jointly with MS departments) positions of HPC-MS engineers.

Finally, funding programs dedicated to computational and HPC research in MS are likely to foster the awareness of HPC benefits for MS and the attractiveness of HPC for MS researchers.

Scalability of Parallel Applications in Management Science Can Differ Fundamentally

Depending on the specific type of MS problem to be solved, the parallelization of algorithms may scale substantially differently over the number of parallel processing units. Therefore, it is important to thoroughly inform MS researchers on issues of efficiency and scalability so they can assess what to expect when solving their particular problem types with HPC.

Some research problems in MS involve solving specific instances of an optimization problem. In such cases, often a fixed-size model (constant total workload, variable execution time) occurs and strong scaling applies: according to Amdahl's Law,¹ the speedup factor that can be achieved from parallelization is upper-bounded by 1 divided by the serial fraction of code, which is always larger than zero in practice. Due to execution time required for coordination, this upper bound is usually not achieved, and speedup values even start dropping when a particular number of parallel processing units (referred to as "processors") is exceeded. Even when ignoring all coordination efforts, a serial fraction of code amounting to 20%, for example, would limit the maximum speedup by the factor of 5 regardless of the number of processors. Consequently, MS researchers must be informed on speedup, efficiency, and scalability that can be expected and their determining factors.

Other problems in MS, often occurring in data analytics and in a real-time decision making context, follow

It is important to support MS researchers in designing parallel algorithms and to release them from parallel implementation and technical issues.

a scale-size model (variable total workload, constant workload per processor, constant execution time). Then, according to Gustafson's Law,⁵ weak scaling applies, which means speedup can increase (almost) linearly with the number of processors (even when coordination efforts are considered).

Applications Need Not Primarily to be Rewritten but Rethought by (Re-)design

While some sequential applications can be parallelized straightforward, limiting parallelization efforts to the implementation phase is myopic. As noted in Fuller and Millett,⁴ "attempts to extract parallelism from the serial implementations are unproductive exercises and likely to be misleading if they cause one to conclude that the original problem has an inherently sequential nature." Thus, it is important to support MS researchers in designing parallel algorithms and to release them from parallel implementation and technical issues. While this approach is useful for scholars of all scientific disciplines, it is of particular importance in fields where researchers are not used to programming-intensive tasks, as this is often the case in the MS community.

Several frameworks applicable to MS have already been suggested (for example, Apache Hadoop, Ubiquity Generator by ZIB, Branch-Cut-Price framework in COIN-OR) or are under development (for example, the PASC project "Framework for computing

equilibria in heterogeneous agent macro models"); however, we should strengthen our efforts to develop IT artifacts that support MS researchers in parallel design and parallel implementation. In particular, high-level languages at the application level rather than multipurpose parallel languages at the programming level would need to be provided. The availability and usability of such application languages would allow MS researchers to focus on parallel design issues and release them from writing parallel code at the programming level, which would be generated automatically by application language compilers. Such approaches are appropriate for deploying HPC in MS at a large scale. Concluding, it is currently an auspicious time for integrating MS into the HPC research ecosystem, and the MS community can look forward to the promising developments to come. ■

References

1. Amdahl, G.M. Validity of the single-processor approach to achieving large scale computing capabilities. In *AFIPS Conference Proceedings*, Vol. 30, AFIPS Press, Reston, VA, (1967), 483–485.
2. Aon. PathWise. 2019; [https://www.aon.com/reinsurance/PathWise-\(1\)/default.jsp](https://www.aon.com/reinsurance/PathWise-(1)/default.jsp)
3. Eftekhari, A., Scheidegger, S., and Schenk, O. Parallelized dimensional decomposition for large-scale dynamic stochastic economic models. In *Proceedings of the Platform for Advanced Scientific Computing Conference*. (2017).
4. Fuller, S.H. and Millett, L.I. *The Future of Computing Performance: Game Over or Next Level?* National Academy Press. (2011).
5. Gustafson, J.L. Reevaluating Amdahl's Law. *Commun. ACM* 31, 5 (May 1988), 532–533.
6. Juric, R., Kim, I., Panneerselvam, H., and Tesanovic, I. Analysis of ZIKA virus tweets: Could Hadoop platform help in global health management? In *Proceedings of the 50th Hawaii International Conference on System Sciences*. (2017).
7. Kübler, F., Scheidegger, S., and Schenk, O. Computing Equilibria in Heterogeneous Agent Macro Models on Contemporary HPC Platforms (2017); www.pasc-ch.org/projects/2017-2020/computing-equilibria-in-heterogeneous-agent-macro-models/ <<http://www.pasc-ch.org/projects/2017-2020/computing-equilibria-in-heterogeneous-agent-macro-models/>>
8. Microsoft. Big data: Improving fraud recognition with Microsoft Azure. (2017); <https://customers.microsoft.com/en-us/story/arvato-azure-powerbi-germany-media-inovex>
9. Rauchecker G. and Schryen G. Using High Performance Computing for unrelated parallel machine scheduling with sequence-dependent setup times: Development and computational evaluation of a parallel branch-and-price algorithm. *Computers and Operations Research* 104 (2019), 338–357.
10. Schryen, G., Kliewer, N., and Fink, A. Call for Papers Issue 1/2020—High Performance Business Computing. *Business and Information Systems Engineering* 60, 5 (2018), 439–440.
11. Zhang, K., Bhattacharyya, S., and Ram, S. Large-scale network analysis for online social brand advertising. *MIS Quarterly* 40, 4 (2016), 849–868.

Guido Schryen (guido.schryen@upb.de) is a full professor of Management Information Systems and Operations Research at Paderborn University, Germany.

Copyright held by author.

Viewpoint

‘Have You Thought About ...’ Talking About Ethical Implications of Research

Considering the good and the bad effects of technology.

HOW DO RESEARCHERS talk to one another about the ethics of our research? How do you tell someone you are concerned their work may do more harm than good for the world? If someone tells you your work may cause harm, how do you receive that feedback with an open mind, and really listen? I find myself lately on both sides of this dilemma—needing both to speak to others and listen myself more. It is not easy on either side. How can we make those conversation more productive?

We are at an unprecedented moment of societal change brought on by new technologies. We create things with both good and bad possible uses, and with implications we may or may not be able to meaningfully foresee. Technologies have both intended and unintended consequences.⁵ To complicate things, we inevitably lose control of the technologies we create. The ethical responsibility of us as creators is difficult to understand. But one thing we can and should do is to talk about those implications—relentlessly. Even when the conversations are difficult.

This semester, I assigned my students in my class “Computers, Society, and Professionalism” to listen to a podcast from Planet Money, “Stuck in China’s Panopticon” (see <https://www.npr.org/2019/07/05/738949320/episode-924-stuck-in-chinas-panopticon>). The podcast documents unprecedented



levels of surveillance being used to oppress China’s Uighur minority. The Chinese police are creating comprehensive profiles of each Uighur person, including their DNA, face, voice, and even their gait. Technology that can determine your ethnic background from your DNA was developed by Yale geneticist Kenneth Kidd. Years ago, Kidd allowed a researcher from the Chinese Ministry of Public Security to spend time in his lab learning how his techniques worked, and he shared DNA data with them. These were later used to oppress the Uighur.

But at the time, it was pure research. Planet Money asks Kidd if he regrets collaborating with the Chinese secret police, and he says he “[couldn’t] know everything that’s going to happen in the future.” On the other hand, the Uighur man profiled in the story is outraged at what Kidd did, saying he should have known. How do we resolve this stand-off?

The Dilemma of Good and Bad Uses
Another example of a technology with good and bad uses is face recognition. For many years, I have been disturbed

by the social implications of face recognition technology. It is possible to use face recognition responsibly, of course. But any technology developed will eventually become widely available, including to less-responsible individuals. Yes, we can use face recognition technology for simple conveniences like unlocking our phones, and also for important security applications. But this technology also inevitably will fall into the hands of state and non-state actors who will use it in oppressive ways (like the Chinese Ministry of Public Security). When you think about it, it is frightening.

I was against this technology for many years. That is until we had a blind student in our department, and he told me that working face recognition software would change his life. If he could only easily know who is in the room with him, it would be transformative.

Face recognition has good and bad uses. The number of people who will be harmed by it in less-free societies (and maybe in our own) greatly outnumber the population of blind people who will be helped by it. So all in all, I personally would not do research on it. But I realize I may not fully understand its implications, just like I initially did not understand its importance for the blind. How can any of us fully understand the implications of something so transformative?

It is tempting to just say, “face recognition is going to be developed no matter what I do,” and shrug and go about our own business. But I think that’s a cop out. I am not an expert on face recognition, but I have colleagues who are. How can I talk with them about it? Ethics are discursive—ethical understandings emerge from conversation. *But we’re not talking about ethical issues of new technology enough.*

Here, I am using face recognition as a stand-in for all technologies that have possible strong negative consequences (as well as good ones). And there are a lot of them. We approach an era of rapid change in norms of privacy, jobs supported by the economy, the degree of education needed for those jobs, and the economic inequality that the structure of the new work force will generate if we do not have the political will to balance it better. While much of the power to shape what happens next

Ethics are discursive—ethical understandings emerge from conversation.

is in the hands of policymakers, some of it is in the hands of the people inventing these technologies.

I teach professional ethics to our undergraduates, and one course topic is how best to raise ethical issues that come up in an organization. The first rule is always go through internal channels before you contact people outside your organization. For conversations about ethical research, the analogous principle is: always talk to the researcher first, in private, before you make public pronouncements. If you are attending a public presentation about the work and there is a Q&A session, ask a polite question. You might, at the start, use hedging language: “Have you thought about ...” Or “I’m concerned about ...” You could also offer to follow up with the author, and try to catch them privately. A key principle for delicate ethical discussions is to give people opportunities to *save face*. Someone is more likely to listen if they can plausibly think that doing the right thing was their idea all along. If you antagonize them, they will just dig their heels in.

You always need to think about who you are trying to influence. If you are trying to influence researchers, you need to talk to them in ways they can hear. Try to validate their goals and present the change you are suggesting as a modest deviation from their current plan (even if a bigger change is more what you are hoping for.)

Criticizing the ethics of someone else’s research requires humility. It is their research. Unless you are in exactly the same field, they know more about it than you do. They may have considered the issues you are worried about and thought them through. They may be several steps ahead of

you. They also may share some of your ethical concerns, and not express them explicitly because to them they are just so obvious.

Consider the possibility that your critique is misguided. When you use the phrasing, “Have you thought about ...,” it should not just be an attempt at being polite but also a sincere acknowledgment they may well have thought about it—a lot. They may be way ahead of you.

People do not adjust the way they think about things right away. Helping someone to rethink ethical implications of their work takes time—often years. And it is not fun. It is so much easier to “stay in your lane” and shrug off problematic work. But maybe we can help one another to see important issues if we focus on politeness and humility.

If you see something that you sincerely (checking your knowledge and assumptions more than once) believe is going to cause immediate harm, it may be necessary to forget about being polite or humble about it. But that is a rare occurrence. Most of the time, things are subtle, and a more delicate approach is strategic.

Changing the System

ACM had the great idea a few years ago to create a “Future of Computing Academy” and gathered together some of the brightest young computing researchers. The group came up with an ambitious plan for helping draw more attention to the social implications of technology. They proposed that, “Peer reviewers should require that papers and proposals rigorously consider all reasonable broader impacts, both positive and negative.”³ The plan is forward-thinking and insightful, though there are practical details that need working out. Presumably under this plan, work will be evaluated on how honest you are about possible implications, not how bad the implications are. But what if the consequences are difficult to foresee? What if the consequences are potentially really scary? The details make my head hurt. That said, it is a first step toward taking these implications more seriously. The challenge is how to convince ACM and other professional societies and publishers that this is important and necessary.

A Portfolio of Approaches

There are many things we need to do to better shape the future of technology. At the individual level, we can encourage technologists (especially students and young professionals) to choose to work on things they believe will make a positive difference.


While individual choices matter, the big and pressing problems require a more coordinated approach. I am encouraged to see the beginnings of collective action in the technology industry. For example, in 2018 over 3,000 Google employees signed a letter protesting the company's participation in a military AI initiative called Project Maven,² and as a result the company chose not to renew its contract for this work.⁴

In addition to individual choice and collective action, the third key strategy is policy. In May 2019, San Francisco proactively outlawed the use of face recognition technology by police.¹ SF police were not actually using face recognition and it does not reliably work yet, but policymakers are anticipating consequences and doing something about them in ad-

There are many things we need to do to better shape the future of technology.

vance. We need more politicians who really understand technology, and to hold those politicians accountable for forward-thinking policy change rather than simply reacting to disastrous situations after the fact.

Geneticist Kenneth Kidd says he could not have predicted how his technology would be used by the Chinese secret police. Maybe he would have realized the implications if colleagues had talked with him about it. Talking to one another is just one among a host of strategies that are needed

to take the implications of new technologies seriously and try to help to shape them. We all need to make the effort (even when it is uncomfortable) to say, "Have you thought about ... " And to listen with an open mind when someone says that to us. 

References

1. Conger, K., Fausset, R., and Kavaleski, S.F. San Francisco bans facial recognition technology. *The New York Times*, (May 14, 2019); <https://www.nytimes.com/2019/05/14/us/facial-recognition-ban-san-francisco.html>
2. Google employees. Letter to Sundar Pichai about Project Maven. (2018); <https://static01.nytimes.com/files/2018/technology/googleletter.pdf>
3. Hecht, B., et al. It's time to do something: Mitigating the negative impacts of computing through a change to the peer review process. *ACM Future of Computing Blog*. 2018; <https://acm-fca.org/2018/03/29/negativeimpacts/>.
4. Statt, N. Google reportedly leaving Project Maven military AI program after 2019. (*The Verge*, June 1, 2018); <https://www.theverge.com/2018/6/1/17418406/google-maven-drone-imagery-ai-contract-expire>
5. Winner, L. Do artifacts have politics?. *Daedalus*, (1980), 121–136.

Amy Bruckman (asb@cc.gatech.edu) is Professor and Senior Associate Chair in the School of Interactive Computing at the Georgia Institute of Technology, Atlanta, GA, USA.

Eric Gilbert and Lana Yarosh provided helpful comments on a draft of this Viewpoint.

Copyright held by author.



上海科技大学
ShanghaiTech University

TENURE-TRACK AND TENURED POSITIONS School of Information Science and Technology (SIST)

ShanghaiTech University invites highly qualified candidates to fill multiple tenure-track/tenured faculty positions as its core founding team in the School of Information Science and Technology (SIST). We seek candidates with exceptional academic records or demonstrated strong potentials in all cutting-edge research areas of information science and technology. They must be fluent in English. English-based overseas academic training or background is highly desired.

ShanghaiTech is founded as a world-class research university for training future generations of scientists, entrepreneurs, and technical leaders. Boasting a new modern campus in Zhangjiang Hightech Park of cosmopolitan Shanghai, ShanghaiTech shall trail-blaze a new education system in China. Besides establishing and maintaining a world-class research profile, faculty candidates are also expected to contribute substantially to both graduate and undergraduate educations.

Academic Disciplines: Candidates in all areas of information science and technology shall be considered. Our recruitment focus includes, but is not limited to: computer science and technology, electronic science and technology, information and communication engineering, applied mathematics and statistics, data science, robotics, bioinformatics, biomedical engineering, internet of things, smart energy, computer systems and security, operation research, mathematical optimization and other interdisciplinary fields involving information science and technology, especially areas related to AI.

Compensation and Benefits: Salary and startup funds are highly competitive, commensurate with experience and academic accomplishment. We also offer a comprehensive benefit package to employees and eligible dependents, including on-campus housing. All regular ShanghaiTech faculty members will join its new tenure-track system in accordance with international practice for progress evaluation and promotion.

Qualifications:

- Strong research productivity and demonstrated potentials;
- Ph.D. (Electrical Engineering, Computer Engineering, Computer Science, Statistics, Applied Math, or related field);
- A minimum relevant (including PhD) research experience of 4 years.

Applications: Submit (in English, PDF version) a cover letter, a 2-page research plan, a CV plus copies of 3 most significant publications, and names of three referees to: sist@shanghaitech.edu.cn

For more information, please visit: <http://sist.shanghaitech.edu.cn/>

Deadline: December 31, 2020



ADVERTISING IN CAREER OPPORTUNITIES

How to Submit a Classified Line Ad: Send an e-mail to acmm mediasales@acm.org. Please include text, and indicate the issue/or issues where the ad will appear, and a contact name and number.

Estimates: An insertion order will then be e-mailed back to you. The ad will be typeset according to CACM guidelines. NO PROOFS can be sent. Classified line ads are NOT commissionable.

Deadlines: 20th of the month/2 months prior to issue date. For latest deadline info, please contact: acmm mediasales@acm.org

Career Opportunities Online: Classified and recruitment display ads receive a free duplicate listing on our website at: <http://jobs.acm.org>

Ads are listed for a period of 30 days.

For More Information Contact:

**ACM Media Sales
at 212-626-0686 or
acmm mediasales@acm.org**

ACM Welcomes the Colleges and Universities Participating in ACM's Academic Department Membership Program

ACM offers an Academic Department Membership option, which allows universities and colleges to provide ACM Professional Membership to their faculty at a greatly reduced collective cost.

The following institutions currently participate in ACM's Academic Department Membership program:

- Abilene Christian University
- Afrisol Technical College, Zimbabwe
- Alfred State College
- Amherst College
- Appalachian State University
- Augusta University, School of Computer and Cyber Sciences
- Ball State University
- Bellevue College
- Berea College
- Binghamton University
- Boise State University
- Bridgewater State University
- Bryant University
- California Baptist University
- Calvin College
- Clark University
- Colgate University
- Colorado School of Mines
- Columbus State University
- Cornell University
- Creighton University
- Cuyahoga Community College
- Denison University
- European University (Tbilisi, Georgia)
- Franklin University
- Gallaudet University
- Georgia Institute of Technology
- Georgia State University Perimeter College
- Governors State University
- Harding University
- Harvard University
- Harvey Mudd College
- Hochschule für Technik Stuttgart - University of Applied Sciences
- Hofstra University
- Hope College
- Howard Payne University
- Indiana University Bloomington
- Kent State University
- Klagenfurt University, Austria
- Madinah College of Technology, Saudi Arabia
- Massasoit Community College
- Messiah College
- Metropolitan State University
- Missouri State University
- Modesto Junior College
- Monash University, Australia
- Montclair State University
- Mount Holyoke College
- New Jersey Institute of Technology
- New Mexico State University
- Northeastern University
- Ohio State University
- Old Dominion University
- Pacific Lutheran University
- Pennsylvania State University
- Potomac State College of West Virginia University
- Purdue University Northwest
- Regis University
- Rhodes College
- Rochester Institute of Technology
- Rutgers University
- Saint Louis University
- San José State University
- Shippensburg University
- Simmons University
- Spelman College
- St. John's University
- Stanford University
- State University of New York at Fredonia
- State University of New York at Oswego
- Stetson University
- Trine University
- Trinity University
- Union College
- Union University
- Univ. do Porto, Faculdade de Eng. (FEUP)
- University at Albany, State University of New York
- University of Alabama
- University of Arizona
- University of California, Riverside
- University of California, San Diego
- University of Colorado Boulder
- University of Colorado Denver
- University of Connecticut
- University of Houston
- University of Illinois at Chicago
- University of Jamestown
- University of Liechtenstein
- University of Lynchburg
- University of Maribor, Slovenia
- University of Maryland, Baltimore County
- University of Memphis
- University of Namibia
- University of Nebraska at Kearney
- University of Nebraska Omaha
- University of New Mexico
- University of North Dakota
- University of Pittsburgh
- University of Puget Sound
- University of Southern California
- University of St. Thomas
- University of the Fraser Valley
- University of Victoria, BC Canada
- University of Wisconsin–Parkside
- University of Wyoming
- Virginia Commonwealth University
- Wake Forest University
- Wayne State University
- Wellesley College
- Western New England University
- William Jessup University

Through this program, each faculty member receives all the benefits of individual professional membership, including *Communications of the ACM*, member rates to attend ACM Special Interest Group conferences, member subscription rates to ACM journals, and much more.

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

The evolution of tricky user interfaces.

BY ARVIND NARAYANAN, ARUNESH MATHUR,
MARSHINI CHETTY, AND MIHIR KSHIRSAGAR

Dark Patterns: Past, Present, and Future

DARK PATTERNS ARE user interfaces that benefit an online service by leading users into making decisions they might not otherwise make. Some dark patterns deceive users while others covertly manipulate or coerce them into choices that are not in their best interests. A few egregious examples have led to public backlash recently: TurboTax hid its U.S. government-mandated free tax-file program for low-income users on its website to get them to use its paid program;⁹ Facebook asked users to enter phone numbers for two-factor authentication but then used those numbers to serve targeted ads;³¹ Match.com knowingly

let scammers generate fake messages of interest in its online dating app to get users to sign up for its paid service.¹³ Many dark patterns have been adopted on a large scale across the Web. Figure 1 shows a deceptive countdown timer dark pattern on JustFab. The advertised offer remains valid even after the timer expires. This pattern is a common tactic—a recent study found such deceptive countdown timers on 140 shopping websites.²⁰

The research community has taken note. Recent efforts have catalogued dozens of problematic patterns such as nagging the user, obstructing the flow of a task, and setting privacy-intrusive defaults,^{1,18} building on an early effort by Harry Brignull (darkpatterns.org). Researchers have also explained how dark patterns operate by exploiting cognitive biases^{4,20,33} uncovered dark patterns on more than 1,200 shopping websites,²⁰ shown that more than 95% of the popular Android apps contain dark patterns,⁸ and provided preliminary evidence that dark patterns are indeed effective at manipulating user behavior.^{19,30}

Although they have recently burst into mainstream awareness, dark patterns are the result of three decades-long trends: one from the world of retail (deceptive practices), one from research and public policy (nudging), and the third from the design community (growth hacking).

Figure 2 illustrates how dark patterns stand at the confluence of these three trends. Understanding these trends—and how they have collided into each other—is essential to help us appreciate what is actually new about dark patterns, demystifies their surprising effectiveness, and shows us why it will be difficult to combat them. We end this article with recommendations for ethically minded designers.

Deception and Manipulation in Retail

The retail industry has a long history of deceptive and manipulative practices that range on a spectrum from



Don't uncheck this box if you do want to avoid making a choice which isn't the opposite of how you would prefer to select.

normalized to unlawful (Figure 3). Some of these techniques, such as psychological pricing (that is, making the price slightly less than a round number), have become normalized. This is perfectly legal, and consumers have begrudgingly accepted it. Nonetheless, it remains effective: consumers underestimate prices when relying on memory if psychological pricing is employed.³

More problematic are practices such as false claims of store closings, which are unlawful but rarely the target of enforcement actions. At the other extreme are bait-and-switch car ads such as the one by a Ford dealership in Cleveland that was the target of an FTC action.¹⁴

The Origins of Nudging

In the 1970s, the heuristics and biases literature in behavioral economics sought to understand irrational decisions and behaviors—for example, people who decide to drive because they perceive air travel as dangerous, even though driving is, in fact, orders of magnitude more dangerous per mile.²⁹ Researchers uncovered a set of cognitive shortcuts used by people that make these irrational behaviors not just explainable but even predictable.

For example, in one experiment, researchers asked participants to write down an essentially random

two-digit number (the last two digits of each participant's social security number), then asked if they would pay that number of dollars for a bottle of wine, and finally asked the participants to state the maximum amount they would pay for the bottle.² They found the willingness to pay varied by approximately threefold based on the arbitrary number. This is the *anchoring effect*: lacking knowledge of the market value of the bottle of wine, participants' estimates become anchored to the arbitrary reference point. This study makes it easy to see how businesses might be able to nudge customers to pay higher prices by anchoring their expectations to a high number. In general, however, research on psychological biases has not been driven by applications in retail or marketing. That would come later.

Nudging: The Turn to Paternalism

The early behavioral research on this topic focused on understanding rather than intervention. Some scholars, such as Cass Sunstein and Richard Thaler, authors of the book *Nudge*,²⁸ went further to make a policy argument: Governments, employers, and other benevolent institutions should engineer “choice architectures” in a way that uses behavioral science for

the benefit of those whom they serve or employ.

A famous example (Figure 4) is the striking difference in organ-donation consent rates between countries where people have to explicitly provide consent (red bars) versus those where consent is presumed (orange bars). Because most people tend not to change the default option, the latter leads to significantly higher consent rates.¹⁷

Today, nudging has been enthusiastically adopted by not only governments and employers, but also businesses in the way they interact with their customers. The towel reuse message you may have seen in hotel rooms (“75% of guests in this hotel usually use their towels more than once”) is effective because it employs descriptive social norms as a prescriptive rule to get people to change their behavior.¹⁶

With the benefit of hindsight, neither the proponents nor the critics of nudging anticipated how readily and vigorously businesses would adopt these techniques in adversarial rather than paternalistic ways. In *Nudge*, Sunstein and Thaler briefly address the question of how to tell if a nudge is ethical, but the discussion is perfunctory. The authors seem genuinely surprised by recent developments and have distanced themselves from dark patterns, which they label “sludges.”²⁷

Figure 1. A deceptive countdown timer on JustFab.

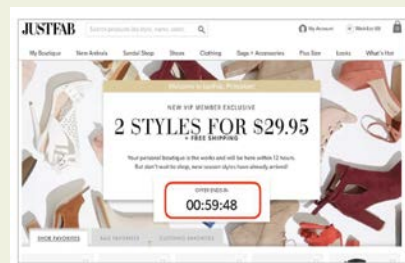
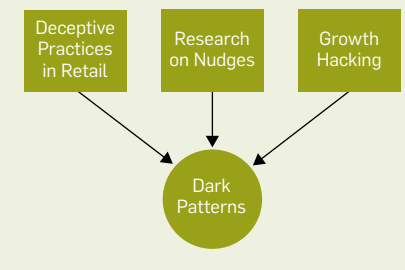


Figure 2. The origins of dark patterns.



Growth Hacking

The third trend—and the one that most directly evolved into dark patterns—is growth hacking. The best-known and arguably the earliest growth hack was implemented by Hotmail. When it launched in 1996, the founders first considered traditional marketing methods such as billboard advertising. Instead, they hit upon a viral marketing strategy: The service automatically added the signature, “Get your free email with Hotmail,” to every outgoing email, essentially getting users to advertise on its behalf, resulting in viral growth.²¹

Successes like these led to the emergence of growth hacking as a distinct community. Growth hackers are trained in design, programming, and marketing and use these skills to drive product adoption.

Growth hacking is not inherently deceptive or manipulative but often is in practice. For example, in two-sided markets such as vacation rentals, upstarts inevitably face a chicken-and-egg problem: no travelers without hosts and no hosts without travelers. So it became a common practice to “seed” such services with listings that were either fake or scraped from a competitor.^{22,23}

Unsurprisingly, growth hacking has sometimes led to legal trouble. A hugely popular growth hack involved obtaining access to users’

contact books—often using deception—and then spamming those contacts with invitations to try a service. The invitations might themselves be deceptive by appearing to originate from the user, when in fact users were unaware of the emails being sent. LinkedIn settled a class action for exactly this practice, which it used from 2011 to 2014.²⁵

From Growth Hacking to Dark Patterns

But why growth rather than revenue or some other goal? It is a reflection of Silicon Valley’s growth-first mantra in which revenue-generating activities are put aside until after-market dominance has been achieved. Of course, eventually every service runs into limits on growth, because of either saturation or competition, so growth hackers began to adapt their often-manipulative techniques to extracting and maximizing revenue from existing users.

In developing their battery of psychological tricks, growth hackers had two weapons that were not traditionally available in offline retail. The first was that the nudge movement had helped uncover the principles of behavior change. In contrast, the marketing literature that directly studied the impact of psychological tricks on sales was relatively limited because it didn’t get at the foundational principles and was limited to the domain of retail.

The second weapon was A/B testing (Figure 5). By serving variants of Web pages to two or more randomly selected subsets of users, designers began to discover that even seemingly trivial changes to design elements can result in substantial differences in behavior. The idea of data-driven optimization of user interfaces has become deeply ingrained in the design process of many companies. For large online services with millions of users, it is typical to have dozens of A/B tests running in parallel, as noted in 2009 by Douglas Bowman, once a top visual designer at Google:

Yes, it’s true that a team at Google couldn’t decide between two blues, so they’re testing 41 shades between each blue to see which one performs better. I had a recent debate over whether a border should be 3, 4, or 5 pixels wide, and was asked to prove my case. I can’t operate in an environment

like that. I’ve grown tired of debating such minuscule design decisions. There are more exciting design problems in this world to tackle. —Douglas Bowman

A/B testing proved key to the development of dark patterns because it is far from obvious how to translate an abstract principle like social proof into a concrete nudge (“7 people are looking at this hotel right now!”). Another example: For how long should a fake countdown timer be set (“This deal expires in 15 minutes!” ... “14:59” ... “14:58” ...), so the user acts with urgency but not panic? Online experiments allow designers to find the answers with just a few lines of code.

Figure 3. Examples of deceptive and manipulative retail practices.



Source: <https://www.crazyspeedtech.com/5-major-stages-psychological-pricing/>



Source: <https://www.dealnews.com/features/What-Happens-When-a-Store-Closes/2203265.html>



Source: <https://www.ftc.gov/enforcement/cases-proceedings/1223269/ganley-ford-west-inc-matter>

Money, Data, Attention

Let's recap. As the online economy matured, services turned their attention from growth to revenue. They used the principles of behavioral influence but subverted the intent of the researchers who discovered those principles by using them in ways that undermined consumers' autonomy and informed choice. They used A/B testing to turn behavioral insights into strikingly effective user interfaces. In some cases these were optimized versions of tricks that have long been used in retail, but in other cases they were entirely new.

How, exactly, do dark patterns help maximize a company's ability to extract revenue from its users? The most obvious way is simply to nudge (or trick) consumers into spending more than they otherwise would.

A less obvious, yet equally pervasive, goal of dark patterns is to invade privacy. For example, cookie consent dialogs almost universally employ manipulative design to increase the likelihood of users consenting to tracking. In fact, a recent paper shows that when asked to opt in, well under 1% of users would provide informed consent.³⁰ Regulations such as the GDPR (General Data Protection Regulation) require companies to get explicit consent for tracking, which poses an existential threat to many companies in the online tracking and advertising industry. In response, they appear to be turning to the wholesale use of dark patterns.³⁰

A third goal of dark patterns is to make services addictive. This goal supports the other two, as users who stay on an app longer will buy more, yield more personal information, and see more ads. Apps like Uber use gamified nudges to keep drivers on the road longer (Figure 6). The needle suggests the driver is extremely close to the goal, but it is an arbitrary goal set by Uber when a driver wants to go offline.²⁴ To summarize, dark patterns enable designers to extract three main resources from users: money, data, and attention.

Dark Patterns Are Here to Stay

Two years ago, few people had heard the term dark patterns. Now it's everywhere. Does this mean dark patterns

are a flash in the pan? Perhaps, as users figure out what's going on, companies will realize that dark patterns are counterproductive and stop using them. The market could correct itself.

The history sketched here suggests that this optimistic view is unlikely. The antecedents of dark patterns are decades old. While public awareness of dark patterns is relatively new, the phenomenon itself has developed gradually. In fact, the darkpatterns.org website was established in 2010.

The history also helps explain what is new about dark patterns. It isn't just tricky design or deceptive retail practices online. Rather, design has been weaponized using behavioral research to serve the aims of the surveillance economy. This broader context is important. It helps explain why the situation is as bad as it is and suggests that things will get worse before they can get better.

One worrying trend is the emergence of companies that offer dark patterns as a service, enabling websites to adopt them with a few lines of JavaScript.²⁰ Another possible turn for the worse is personalized dark patterns that push each user's specific buttons.²⁶ This has long been predicted⁵ but remains rare today (manipulative targeted advertising can arguably be viewed as a dark pattern, but ads are not user interfaces). The absence of personalized UI is presumably because companies are busy picking lower-hanging fruit, but this can change any time.

Recommendations for Designers

Designers should be concerned about the proliferation of dark patterns. They are unethical and reflect badly on the profession. But this article is not a doom-and-gloom story. There are steps you can take, both to hold yourself and your organization to a higher standard,

Figure 4. Organ-donation consent rates by countries.

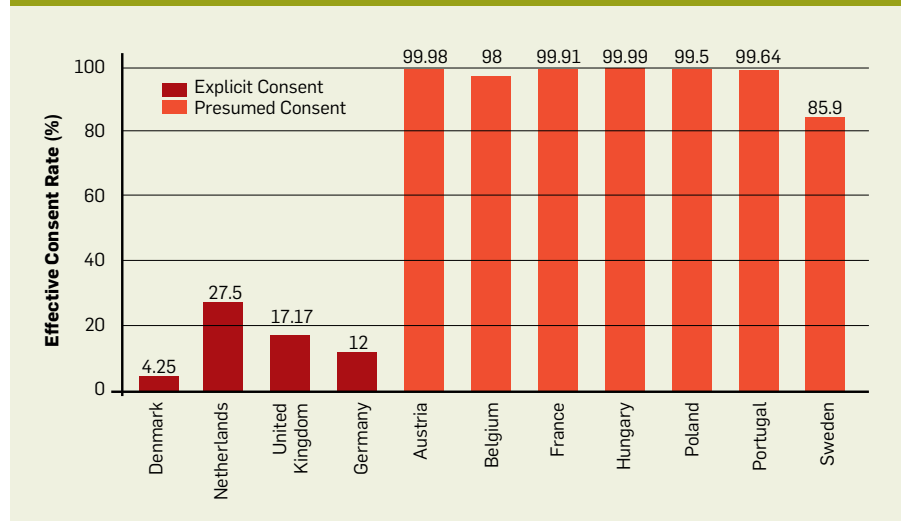


Figure 5. Hypothetical illustration of A/B testing on a website.

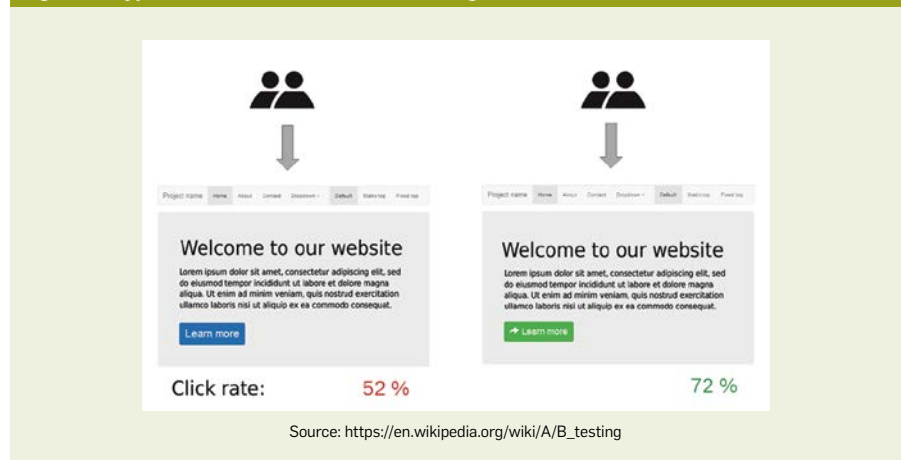
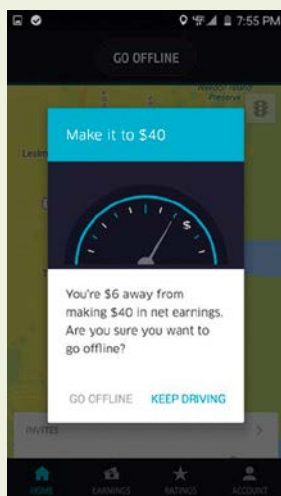


Figure 6. One of Uber's gamified nudges to keep drivers on the road.



Source: <https://www.nytimes.com/interactive/2017/04/02/technology/uber-drivers-psychological-tricks.html>

and to push back against the pressure to deploy dark patterns in the industry.

Go beyond superficial A/B testing metrics. Earlier we discussed how designers use A/B tests to optimize dark patterns. But there's a twist: a design process hyperfocused on A/B testing can result in dark patterns even if that is not the intent. That's because most A/B tests are based on metrics that are relevant to the company's bottom line, even if they result in harm to users. As a trivial example, an A/B test might reveal that reducing the size of a "Sponsored" label that identifies a

search result as an advertisement causes an increase in the CTR (click-through rate). While a metric such as CTR can be measured instantaneously, it reveals nothing about the long-term effects of the design change. It is possible that users lose trust in the system over time when they realize they are being manipulated into clicking on ads.

In a real example similar to this hypothetical one, Google recently changed its ad labels in a way that made it difficult for users to distinguish ads from organic search results, and presumably increased CTR for ads (Figure 7). A backlash ensued, however, and Google rolled back this interface.³²

To avoid falling into this trap, evaluate A/B tests on at least one metric that measures long-term impacts. In addition to measuring the CTR, you could also measure user retention. That will tell you if a different-sized label results in more users abandoning the website.

Still, many attributes that matter in the long term, such as trust, are not straightforward to observe and measure, especially in the online context. Think critically about the designs you choose to test, and when you find that a certain design performs better, try to understand why.

While the overreliance on A/B testing is a critical issue to be addressed, let's next turn to a much broader and longer-term concern.

Incorporate ethics into the design

process. While dark patterns are a highly visible consequence of the ethical crisis in design, resolving the crisis entails far more than avoiding a simple list of patterns. It requires structural changes to the design process.

Start by articulating the values that matter to you and that will guide your design.¹⁵ Not every organization will have an identical set of values, but these values must be broadly aligned with what society considers important.

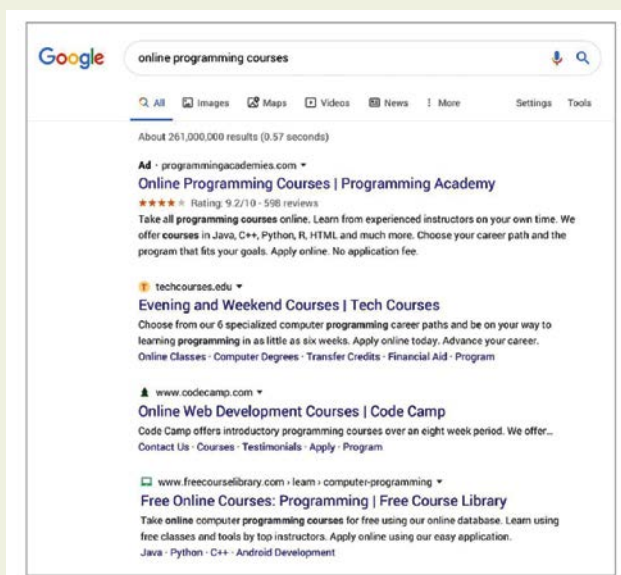
In fact, much of the present crisis can be traced to a misalignment of values between society and companies. Autonomy and privacy are two values where this is particularly stark. Consider frictionless design, a bedrock value in the tech industry. Unfortunately, it robs users of precisely those moments that may give them opportunities for reflection and enable them to reject their baser impulses. Frictionlessness is antithetical to autonomy. Similarly, designing for pleasure and fun is a common design value, but when does fun cross the line into addiction?

Once you have articulated your values, continue to debate them internally. Publicize them externally, seek input from users, and, most importantly, hold yourself accountable to them. Effective accountability is challenging, however. For example, advisory boards established by technology companies have been criticized for not being sufficiently independent.

Everyday design decisions should be guided by referring to established values. In many cases it is intuitively obvious whether a design choice does or does not conform to a design value, but this is not always so. Fortunately, research has revealed a lot about the factors that make a design pattern dark, such as exploiting known cognitive biases and withholding crucial information.^{4,20} Stay abreast of this research, evaluate the impact of design on your users, and engage in critical debate about where to draw the line based on the company's values and your own sense of ethics. Rolling back a change should always be an option if it turns out that it didn't live up to your values.

As you gain experience making these decisions in a particular context, higher-level principles can be codified into design guidelines. There is a long tradition of usability guidelines in the design

Figure 7. Google's recent change to its ad labels.



community. There are also privacy-by-design guidelines, but they are not yet widely adopted.¹⁰ There is relatively little in the way of guidelines for respecting user autonomy.

All of this is beyond the scope of what individual designers can usually accomplish; the responsibility for incorporating ethics into the design process rests with organizations. As an individual, you can start by raising awareness within your organization.


Self-regulate or get regulated. Dark patterns are an abuse of the tremendous power that designers hold in their hands. As public awareness of dark patterns grows, so does the potential fallout. Journalists and academics have been scrutinizing dark patterns, and the backlash from these exposés can destroy brand reputations and bring companies under the lenses of regulators.

Many dark patterns are already unlawful. In the U.S., the Federal Trade Commission (FTC) Act prohibits “unfair or deceptive” commercial practices.¹¹ In a recent example, the FTC reached a settlement with Unroll. Me—a service that unsubscribed users’ email addresses from newsletters and subscriptions—because it was in fact selling information it read from their inboxes to third parties.¹² European Union authorities have tended to be stricter: French regulator CNIL (Commission Nationale de l’Informatique et des Libertés) fined Google 50 million euros for hiding important information about privacy and ad personalization behind five to six screens.⁶

There is also a growing sense that existing regulation is not enough, and new legislative proposals aim to curb dark patterns.⁷ While policymakers should act—whether by introducing new laws or by broadening and strengthening the enforcement of existing ones—relying on regulation is not sufficient and comes with compliance burdens.

Let’s urge the design community to set standards for itself, both to avoid onerous regulation and because it’s the right thing to do. A first step would be to rectify the misalignment of values between the industry and society, and develop guidelines for ethical design. It may also be valuable to partner with neutral third-party consumer advocacy agencies to develop processes to certify apps that are free of known dark patterns. Self-regulation also requires

cultural change. When hiring designers, ask about the ethics of their past work. Similarly, when deciding between jobs, use design ethics as one criterion for evaluating a company and the quality of its work environment.

Design is power. In the past decade, software engineers have had to confront the fact that the power they hold comes with responsibilities to users and to society. In this decade, it is time for designers to learn this lesson as well. 

Related articles on queue.acm.org

User Interface Designers, Slaves of Fashion

Jef Raskin

<https://queue.acm.org/detail.cfm?id=945161>

The Case Against Data Lock-in

Brian W. Fitzpatrick and J.J. Lueck

<https://queue.acm.org/detail.cfm?id=1868432>

Bitcoin’s Academic Pedigree

Arvind Narayanan and Jeremy Clark

<https://queue.acm.org/detail.cfm?id=3136559>

References

- Acquisti, A. et al., Wilson, S. Nudges for privacy and security: understanding and assisting users’ choices online. *ACM Computing Surveys* 50, 3 (2017), 1–41; <https://dl.acm.org/doi/10.1145/3054926>.
- Ariely, D. *Predictably Irrational*. Harper Audio, New York, NY, 2008.
- Bizer, G.Y. and Schindler, R.M. Direct evidence of ending-digit drop-off in price information processing. *Psychology & Marketing* 22, 10 (2005), 771–783.
- Bösch, C., Erb, B., Kargl, F., Kopp, H. and Pfattheicher, S. Tales from the dark side: Privacy dark strategies and privacy dark patterns. In *Proceedings on Privacy Enhancing Technologies* 4, (2016), 237–254.
- Calo, R. Digital market manipulation. *George Washington Law Review* 82, 4 (2014), 995–1051; http://www.gwlr.org/wp-content/uploads/2014/10/Calo_82_41.pdf.
- Commission Nationale de l’Informatique et des Libertés. The CNIL’s restricted committee imposes a financial penalty of 50 million euros against Google LLC, 2019; <https://bit.ly/3dtTcoS>.
- Fischer, D. United States Senator for Nebraska. Senators introduce bipartisan legislation to ban manipulative dark patterns, 2019; <https://bit.ly/3eQQ4LT>.
- Di Geronimo, L., Braz, L., Fregnan, E., Palomba F. and Bachelii, A. UI dark patterns and where to find them: a study on mobile applications and user perception. In *Proceedings of the 2020 ACM Conference on Human Factors in Computing Systems*.
- Elliott, J. and Waldron, L. Here’s how TurboTax just tricked you into paying to file your taxes. *ProPublica* (April 22, 2019); <https://www.propublica.org/article/turbotax-just-tricked-you-into-paying-to-file-your-taxes>.
- European Data Protection Board. Guidelines 4/2019 on Article 25, Data Protection by Design and by Default; <https://bit.ly/3710o9s>.
- Federal Trade Commission. A brief overview of the Federal Trade Commission’s investigative, law enforcement, and rulemaking authority, 2019; <https://www.ftc.gov/about-ftc/what-we-do/enforcement-authority>.
- Federal Trade Commission. FTC finalizes settlement with company that misled consumers about how it accesses and uses their email, 2019; <https://bit.ly/3h0L7PF>.
- Federal Trade Commission. FTC sues owner of online dating service Match.com for using fake love interest ads to trick consumers into paying for a Match.com subscription. Sept. 25, 2019; <https://bit.ly/3dArhUb>.
- Federal Trade Commission. Ganley Ford, 2013; <https://bit.ly/2XxHwft>.
- Friedman, B., Kahn, P. H., Borning, A. and Huldgtren, A. Value-sensitive design and information systems. *Early Engagement and New Technologies: Opening Up the Laboratory*. N. Doorn, D. Schuurbers, I. van de Poel, M.E. Gorman, Eds. Springer, Dordrecht, Germany, 2013, 55–95; <https://link.springer.com/book/10.1007/978-94-007-7844-3>.
- Goldstein, N.J., Cialdini, R.B. and Griskevicius, V. A room with a viewpoint: using social norms to motivate environmental conservation in hotels. *J. Consumer Research* 35, 3 (2008), 472–482.
- Goldstein, D. and Johnson, E.J. Do defaults save lives? *Science* 302, 5649 (2003), 1338–1339; <https://science.sciencemag.org/content/302/5649/1338>.
- Gray, C.M., Kou, Y., Battles, B., Hoggatt, J., Toombs, A.L. The dark (patterns) side of UX design. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (April), 1–14; <https://dl.acm.org/doi/10.1145/3173574.3174108>.
- Luguri, J. and Strahilevitz, L. Shining a light on dark patterns. University of Chicago, Public Law Working Paper. No. 719, 2019.
- Mathur, A., Acar, G., Friedman, M.J., Lucherini, E., Mayer, J., Chetty, M., Narayanan, A. Dark patterns at scale: findings from a crawl of 11K shopping websites. In *Proceedings of the ACM on Human-Computer Interaction* 3 (2019), 1–32; <https://dl.acm.org/doi/10.1145/3359183>.
- McLaughlin, J. 9 iconic growth hacks tech companies used to boost their user bases. *The Next Web*, 2014; <https://bit.ly/2MtXOL1>.
- Mead, D. How Reddit got huge: tons of fake accounts. *Vice*, 2012; https://www.vice.com/en_us/article/z4444w/how-reddit-got-huge-tons-of-fake-accounts-2.
- Rosoff, M. Airbnb farmed Craigslist to grow its listings, says competitor. *Business Insider*, 2011; <https://bit.ly/2Mv23L7>.
- Scheiber, N. How Uber uses psychological tricks to push its drivers’ buttons. *New York Times* (Apr. 22, 2017); <https://nyti.ms/3h3RuNk>.
- Strange, A. LinkedIn pays big after class action lawsuit over user emails. *Mashable*, 2015; <https://mashable.com/2015/10/03/linkedin-class-action>.
- Susser, D., Roessler, B. and Nissenbaum, H. Online manipulation: hidden influences in a digital world. *Georgetown Law Technology Review* 4.1 (2019), 1–45; <https://philarchive.org/archive/SUSOMHv1>.
- Thaler, R.H. Nudge, not sludge. *Science* 361 (2018), 431–431.
- Thaler, R.H., Sunstein, C.R. *Nudge: Improving Decisions About Health, Wealth, and Happiness*. Penguin Books, 2009.
- Tversky, A. and Kahneman, D. Judgment under uncertainty: Heuristics and biases. *Science* 185, 4157 (1974), 1124–1131.
- Utz, C., Degeling, M., Fahl, S., Schaub, F., Holz, T. (Un) informed consent: Studying GDPR consent notices in the field. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2019, 973–990; <https://dl.acm.org/doi/10.1145/3319535.3354212>.
- Venkatadri, G., Lucherini, E., Sapiezynski, P. and Mislove, A. Investigating sources of PII used in Facebook’s targeted advertising. In *Proceedings on Privacy Enhancing Technologies* 1 (2019), 227–244; <https://content.sciendo.com/view/journals/popets/2019/1/article-p227.xml?lang=en>.
- Wakabayashi, D. and Hsu, T. Why Google backtracked on its new search results look. *New York Times* (Jan. 31, 2020); <https://nyti.ms/2XYvg6I>.
- Waldman, A.E. Cognitive biases, dark patterns, and the ‘privacy paradox’. SSRN, 2019.

Arvind Narayanan is an associate professor of computer science at Princeton University, Princeton, NJ, USA, where he leads the Princeton Web Transparency and Accountability Project to uncover how companies collect and use our personal information.

Arunesh Mathur is a graduate student in the department of computer science at Princeton University, Princeton, NJ, USA.

Marshini Chetty is an assistant professor in the department of computer science at the University of Chicago, IL, USA.

Mihir Kshirsagar leads the Tech Policy Clinic at Princeton University’s Center for Information Technology Policy, Princeton, NJ, USA.

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

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

A simple and inexpensive test of failure-atomic update mechanisms.

BY TERENCE KELLY

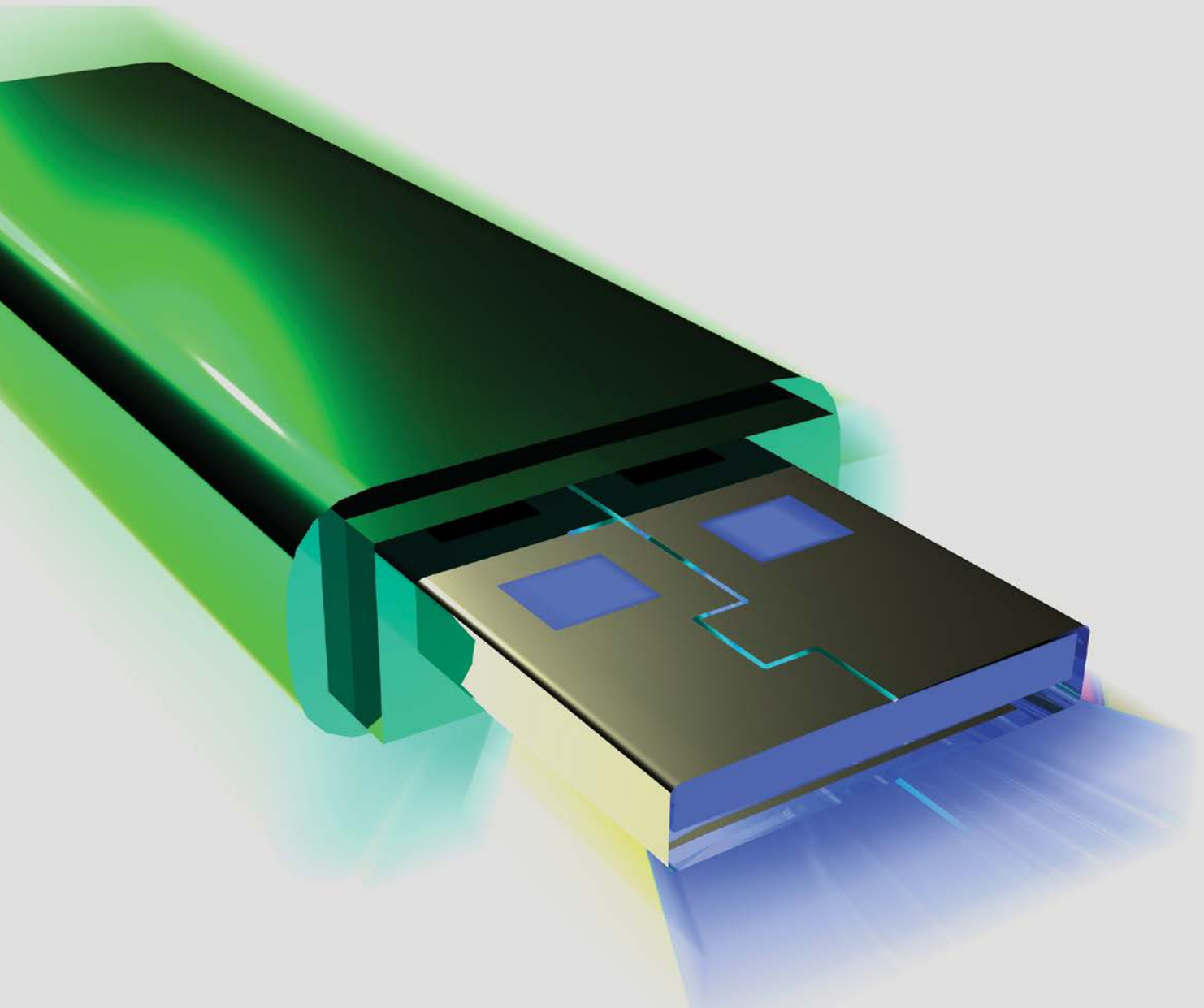
Is Persistent Memory Persistent?

PRESERVING THE INTEGRITY of application data is a paramount duty of computing systems. Failures such as power outages are major perils: A sudden crash during an update may corrupt data or effectively destroy it by corrupting metadata. Applications protect data integrity by using update mechanisms that are atomic with respect to failure. Such mechanisms promise to restore data to an application-defined consistent state following a crash, enabling application recovery.

Unfortunately, the checkered history of failure-atomic update mechanisms precludes blind trust. Widely used relational databases and key-value stores often fail to uphold their transactionality guarantees.²⁴ Lower on the stack, durable storage devices may corrupt or destroy data when power is lost.²⁵ Emerging NVM (non-volatile memory) hardware and corresponding failure-atomic update mechanisms^{7,8} strive to avoid repeating the mistakes of earlier technologies, as do software

abstractions of persistent memory for conventional hardware.^{10,11} Until any new technology matures, however, healthy skepticism demands first-hand evidence that it delivers on its integrity promises.

Prudent developers follow the maxim, “Train as you would fight.” If requirements dictate that an application must tolerate specified failures, then the application should demonstrably survive such failures in pre-production tests and/or in game-day failure-injection testing on production systems.¹ Sudden whole-system power interruptions are the most strenuous challenge for any crash-tolerance mechanism, and there’s no substitute for extensive and realistic power-failure tests. In the past, my colleagues and I tested our crash-tolerance mechanisms against



power failures,^{3,17,20,21} but we did not document the tribal knowledge required to practice this art.

This article describes the design and implementation of a simple and cost-effective testbed for subjecting applications running on a complete hardware/software stack to repeated sudden whole-system power interruptions. A complete testbed costs less than \$100, runs unattended indefinitely, and performs a full test cycle in a minute or less. The testbed is then used to evaluate a recent crash-tolerance mechanism for persistent memory.¹⁰ Software developers can use this type of testbed to evaluate crash-tolerance software before releasing it for production use. Application operators can learn from this article principles and techniques that they can apply to

power-fail testing their production hardware and software.

Of course, power-failure testing alone is but one link in the chain of overall reliability and assurance thereof. Reliability depends on thoughtful design and careful implementation; assurance depends on verification where possible and a diverse and thorough battery of realistic tests.^{2,13} The techniques presented in this article, together with other complementary assurance measures, can help diligent developers and operators keep application data safe.

Persistent memory and corresponding crash-tolerance mechanisms are briefly reviewed, emphasizing the software that will be tested later in the article. This is followed by a description of the power-interruption testbed and test results on the persistent memory

crash-tolerance mechanism. All software described in this article is available at https://queue.acm.org/downloads/2020/Kelly_powerfail.tar.gz.

Persistent Memory

Whereas *non-volatile* memory is a type of hardware, *persistent* memory is a more general hardware-agnostic abstraction, which admits implementation on conventional computers that lack NVM.¹¹ The corresponding persistent memory style of programming involves laying out application data structures in memory-mapped files, allowing application logic to manipulate persistent data directly via CPU instructions (LOAD and STORE).

The main attraction of persistent memory programming is simplicity: It requires neither separate external

persistent stores such as relational databases nor translation between the in-memory data format and a different persistence format. An added benefit of persistent memory on conventional hardware is storage flexibility. Persistent application data ultimately resides in files, and the durable storage layer beneath the file system can be chosen with complete freedom: Even if persistent application data must be geo-replicated across high-availability elastic cloud storage, applications can still access it via `LOAD` and `STORE`. Not surprisingly, persistent memory in all of its forms is attracting increasing attention from industry and software practitioners.¹⁶

The right crash-tolerance mechanism for persistent memory on conventional hardware is FAMS (failure-atomic `msync()`).¹⁷ Whereas the conventional Posix `msync()` system call pushes the modified pages of a file-backed memory mapping down into the backing file *without* any integrity guarantees in the presence of failure, FAMS guarantees that the state of the backing file always reflects the most recent successful `msync()` call. FAMS allows applications to evolve persistent data from one consistent state to the next without fear of corruption by untimely crashes. My colleagues and I have implemented FAMS in the Linux kernel,¹⁷ in a commercial file system,²⁰ and in user-space libraries.^{9,23} At least two additional independent implementations of FAMS exist.^{4,5,22} While this article emphasizes Posix-like environments, analogous features exist on other operating systems. For example, Microsoft Windows has an interface similar to `mmap()` and has implemented a failure-atomic file-update mechanism.¹⁷

This article uses the power-failure testbed described in the next section to evaluate the most recent FAMS implementation: `famus_snap` (failure-atomic `msync()` in user space via snapshots).¹⁰ The `famus_snap` implementation is designed to be audited easily. Whereas previous FAMS implementations involved either arcane kernel/file-system code or hundreds of lines of user-space code, `famus_snap` weighs in at 51 nonblank lines of straightforward code, excluding comments. It achieves brevity by leveraging an efficient per-file snapshotting feature currently available



The techniques presented in this article, together with other complementary assurance measures, can help diligent developers and operators keep application data safe.



in the Btrfs, XFS, and OCFS2 file systems.¹² A side benefit of building FAMS atop file snapshotting is efficiency: Snapshots employ a copy-on-write mechanism and thus avoid the double write of logging.²⁰ While in principle `famus_snap` may seem so clear and succinct that its correctness can be evaluated by inspection, in practice its correctness depends on the file-system implementation of snapshotting, and therefore whole-system power-failure testing is in order.

Power-Failure Testbed

The most important requirement for a power-failure testbed is that software running on the host computer must be able to cut the host's power abruptly at times of its own choosing. Power must then somehow be restored to the host, which must respond by rebooting and starting the next test cycle. The host computer should be rugged, able to withstand many thousands of power interruptions, and it should be able to perform power-off/on cycles rapidly. It should also be affordable to developers with modest budgets, and it should be cheap enough to be expendable, as the stress of repeated power cycling may eventually damage it. Indeed, you should positively *prefer* to use the flimsiest hardware possible: By definition, such hardware increases the likelihood of test failure, therefore successful tests on cheap machines inspire the most confidence. The remainder of this section describes the host computer, auxiliary circuitry, and software that together achieve these goals.

Host computer. Choosing a good host computer isn't easy. Renting from a cloud provider would satisfy the frugality objectives, but unfortunately, cloud hardware is so thoroughly mummified in layers of virtualization that, by design, customer software cannot physically disconnect power from the bare metal. High-end servers expose management interfaces that allow them to be rebooted or powered off remotely, but such shutdowns are much gentler than abrupt power cut-offs; if you could somehow abruptly cut a high-end server's power, you would risk damaging the expensive machine. Laptops are cheap and

throwaways abundant, but laptops lack BIOS features to trigger an automatic reboot upon restoration of external power. Workstations and desktop PCs have such BIOS features, but they are bulky, power-hungry, and they boot slowly.

Single-board computers such as the Raspberry Pi are well suited for our purposes: They are small, rugged, cheap enough to be expendable, and draw very little power. The Pi runs the Linux operating system and nearly all Linux software. It boots quickly and automatically when powered on. Its GPIO (general-purpose input/output) pins enable unprivileged software to control external circuitry conveniently. Most importantly, it's a minimalist no-frills machine. If software and storage devices pass power-failure tests on a Pi, they would likely fare no worse on more expensive feature-rich hardware. The testbed in this article uses the Raspberry Pi 3 Model B+, which will be in production until at least 2026.¹⁸

The main downsides of single-board computers are restrictions on CPU capabilities and peripheral interfaces. For example, the Pi 3B+ CPU doesn't support Linux "soft dirty bits," and storage is limited to the onboard microSD card and USB-attached devices. It's possible, however, to connect a wide range of storage devices via, for example, SATA-to-USB adapters. Overall, the attractions of single-board computers for the present purpose outweigh their limitations.

Power-interruption circuits. In the

past, my colleagues and I tested our crash-consistency mechanisms using AC power strips with networked control interfaces.^{3,17,20,21} These power strips tend to be fussy and poorly documented. Our previous test systems included a separate control computer in addition to the computer that hosted the software and storage devices under test; the control machine used the power strip to cut and restore power to the host. In retrospect, these earlier test frameworks seem unnecessarily complex, rather like "buying a car to listen to the radio." The minimalist power-interruption circuits described in this section are cheaper, more elegant, support more strenuous tests, and enable the host machine to control power cutoffs directly.

The testbed presented here uses electromechanical relays to physically disconnect power from the host computer. Relays faithfully mimic the effects of abrupt power interruptions and completely isolate the host.

Power-supply circuitry can contain a surprising amount of residual energy, enough to enable even server-class computers to shut down somewhat gracefully when utility mains power fails.¹⁵ Our power-interruption circuit therefore interposes between the host computer and its power supply, which eliminates the possibility that residual energy in the power supply might somehow enable an orderly host shutdown rather than a sudden halt.

It turns out that a remarkably simple circuit suffices to disconnect

power momentarily from the host computer, which reliably triggers an immediate reboot. The circuit, shown in Figure 1, is built around a monostable (nonlatching) relay. When sufficient current energizes the relay's coil, movable poles switch from their normally closed position to their normally open position. (Figure 1 follows the convention found on many relay datasheets: The normally *closed* contacts are shown closest to the coil and you are to imagine that current in the coil pushes the relay's poles up toward the normally *open* contacts.) We use a relay¹⁹ whose contacts can carry enough power for the Pi and whose coil can be operated by a Pi's 3.3-volt GPIO pins without exceeding their 16-milliamp current limit. The relay's 180 Ω coil, together with the 31 Ω internal resistance of a GPIO pin,¹⁴ appropriately limits the current.

As shown in Figure 1, the Pi's power supply is routed through the relay's normally closed contacts. When software on the Pi uses a GPIO pin to energize the relay's coil, power to the Pi is cut as the relay's poles jump away from the normally closed contacts. The GPIO pin on the now-powerless Pi then stops pushing current through the coil, so the poles quickly fall back to the normally closed position, restoring power to the Pi and triggering a reboot. When current ceases to flow through the relay coil, the magnetic field in the coil collapses, releasing energy that could harm the delicate circuitry on the Pi

Figure 1. Relay-only circuit.

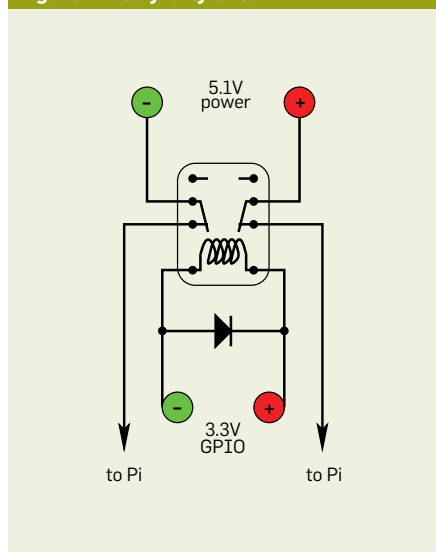
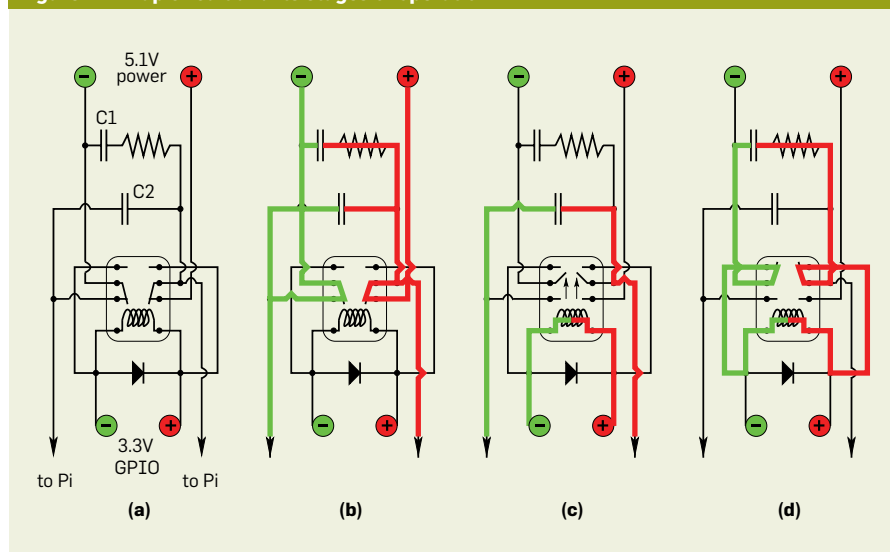


Figure 2. PiNap circuit and its stages of operation.



that controls the GPIO pin. A customary *protection diode* connected in parallel to the relay's coil prevents such damage. (For a description of diode protection for inductive loads, see *The Art of Electronics*.⁶ My circuits use IN4001 diodes.)

The main worry surrounding the circuit of Figure 1 is that it restores power to the host computer very quickly, which might somehow mask failures that a longer power outage would expose.

Figure 2 shows a second circuit, called PiNap, that interrupts power to the Pi for several seconds.

As shown in Figure 2(a), the circuit includes two capacitors: One helps to switch the relay and the other takes over the role of energizing the relay's coil while the Pi is powered off. Figure 2(b) depicts normal operation: The external 5.1V source supplies power to the Pi through the relay's normally closed contacts; it also charges both capacitors. Figure 2(c) shows the transient situation as software on the host computer energizes the relay's coil via a GPIO pin: The external power supply is immediately disconnected from the host, but energy from capacitor C2 enables the GPIO pin to push the relay's poles all the way up to close the nor-

mally open contacts. Then, as shown in Figure 2(d), capacitor C1 discharges through the coil, keeping the Pi powered off for a few seconds. When the energy in C1 is spent, the relay's poles drop back to the normally closed contacts, restoring power to the Pi, which reboots for the next test cycle.

Simple calculations determine the specifications of all components in the PiNap circuit. The relay is the same as in the circuit of Figure 1, for the same reasons.¹⁹ Capacitor C1 is charged to 5.1V so you first choose the resistor to reduce the voltage drop across the relay coil to approximately 3V; by Ohm's law 120Ω is appropriate. Now you choose C1 such that the resistor-capacitor (RC) time constant of C1 and the resistor is on the order of a few seconds; anything in the neighborhood of 10–40 millifarads (10,000–40,000 μF) will do. Electrolytic capacitors in this range are awkwardly large—roughly the size of a salt shaker—so I use a much smaller 22,000 μF supercap. Finally, capacitor C2 must be capable of holding enough energy to keep the coil energized while the relay's poles are in flight (roughly one millisecond, according to the relay datasheet); 220 μF or greater does the trick.

Figure 3 shows a closeup of PiNap on a breadboard. The relay is the white box near the center. Capacitor C2 is the cylinder on the right edge; the black rectangle at the top right is supercap C1. The resistor and protection diode are the small cylinders oriented vertically and horizontally, respectively. All of the components fit comfortably on the U.S. quarter at the bottom of the photo, and the complete circuit occupies the top half of a breadboard the size of a playing card. The host computer's GPIO and ground pins are connected via the red and black vertical jumpers flanking the relay, respectively. External 5.1V power enters the breadboard via the inner rails on either side of the breadboard and exits via the outer rails.

Some relays and capacitors require correct DC polarity; mine do. Sending current the wrong way through a polarity-sensitive relay coil will fail to switch the poles, and the protection diode will provide a short-circuit path. Incorrect polarity can cause a

polarized capacitor to malfunction. Pay close attention to polarity when assembling circuits.

Figure 4 shows the complete testbed on a sheet of U.S. letter-size graph paper: the PiNap breadboard is at bottom center, the Raspberry Pi with USB-attached storage device at right, and the power supply at left. The Pi's USB mouse and keyboard have been removed for clarity. To interpose PiNap between the power supply and the Pi, I cut the power cord and soldered breadboard-friendly 22 AWG solid copper wires to its stranded wires; this was by far the slowest step in assembling the testbed hardware. Everything shown in Figure 4 can be purchased for less than \$100 U.S.

Two final tips for building this testbed:

- First, the relay's pin rows are spaced too closely to span the center furrow of a breadboard, so I fashioned an adapter from a wire-wrap socket (visible between the relay and breadboard in Figure 3). I have also tried the alternative of clamping and/or soldering wires to the relay's pins, but the makeshift adapter shown in Figure 3 is neater and seems less likely to damage the relay.

- Second, the output voltage on some of the Raspberry Pi 3B+'s GPIO pins fluctuates during boot. The power-interruption circuits require a pin that remains at zero volts until software running on the Pi sets it to output logical "HI" (3.3 V). Physical pin 40 works well as a GPIO output pin, and physical pin 39 provides zero-volt ground. These pins are nearest the Pi's USB ports; see the jumpers in Figure 4.

Omitted from this article for both brevity and aesthetics is a third power-interruption circuit that I designed and built before the circuits of Figures 1 and 2. It used *two* relays, an integrated circuit timer chip, several resistors, capacitors, and diodes, and a separate 12 VDC power supply in addition to the Pi's 5.1V power supply. My first circuit worked reliably in thousands of tests, and in some ways it is easier to explain than PiNap, but it is costlier, more complex, and harder to assemble than the circuits presented in this article. The main contribution of my first circuit was to showcase under-simplification, inspiring a search for leaner alternatives.

System configuration and test soft-

Figure 3. PiNap circuits on breadboard.

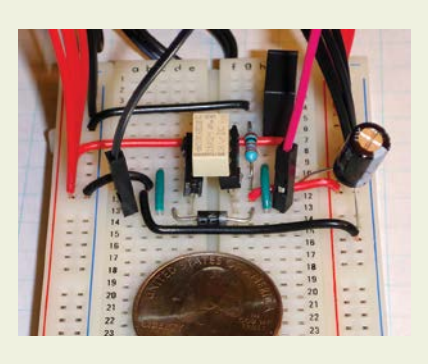
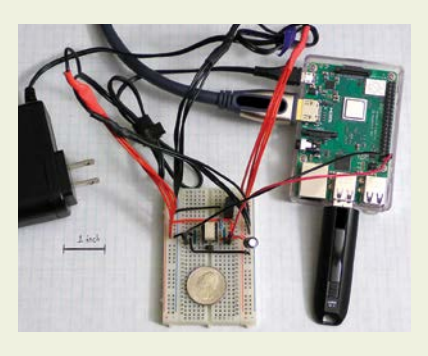


Figure 4. Complete PiNap testbed.




ware. The particulars of configuring the host computer and test software to test the `famus_snap` library against power failures are somewhat tedious. A brief high-level summary of test setup procedures is provided in this section. Detailed instructions are part of the source-code tarball that accompanies this article. (Ambitious readers: Log in to your Pi as default user “pi,” untar in home directory `/home/pi/`, and see the README file.)

The host hardware configuration is relatively straightforward. The host computer must be attached to the external circuit of Figure 1 or Figure 2 via GPIO and ground pins, with the Pi’s power routed through the circuit. A storage device connects via one of the Pi’s USB ports. Note that configuring the host for testing *destroys all data on the storage device*.


Host computer software configuration involves several steps. The host runs the Raspbian variant of Linux; I have used several versions of this operating system from 2018 and 2019. Several nondefault software packages must be installed, notably `xfsplogs`, which is used to create a new XFS file system on the USB-attached storage device. XFS is used because `famus_snap` relies on efficient reflink snapshots, and XFS is one of a handful of file systems that support this feature. It is possible to configure a Pi to boot in a stripped-down fashion (for example, without starting a graphical user interface). A lean boot might be faster, but the difference is small and a default boot is fast enough (less than one minute).

The sample code tarball contains additional software specific to the `famus_snap` tests. The main power-failure test program is called `pft`. It maps a backing file into memory, repeatedly fills the in-memory image with a special pseudo-random pattern, and calls the `famus_snap` analog of `msync()` to commit the in-memory image to a snapshot file. The goal of these tests is to see if power failures corrupt the application-level data of `pft`; thus, the pseudo-random pattern is designed so that corruption is easy to detect.

The `cron` utility is set up to run a script called `rab` every time the Pi boots, which happens when the exter-



Simple calculations determine the specifications of all components in the PiNap circuit.



nal circuit restores power. The main job of `rab` is to invoke a test script, `pft_run`. Script `pft_run` starts test program `pft`, waits for a few seconds, and then uses a GPIO pin to activate the external circuit that cuts and restores power to the Pi.

The `rab` script supports an alternative to power-failure tests: It can be used to *reboot* the Pi suddenly while `pft` is running, which is less stressful than a power failure but easier to set up, as the external circuitry is unnecessary.

After a suitable number of off/on cycles have completed, you can check to see whether `pft`’s data have been corrupted by running the check script. The check script runs `pft` in recovery mode, which inspects the appropriate snapshot files to see if they contain the expected pseudo-random pattern.

Results

The tests of `famus_snap` used all three external power-interruption circuits discussed in this article: the circuits of Figures 1 and 2 and the third complex circuit mentioned briefly before. The tests were run on three different storage devices: a cheap (\$30) 64GB flash thumb drive; an allegedly rugged and rather expensive (\$220) 512GB flash memory stick; and a moderately priced (\$90) 500GB portable SSD (solid-state drive). A total of more than 58,000 power-off/on test cycles ran. Each power-off/on cycle takes roughly one minute, which is considerably faster than the five-minute cycle times of test environments that my colleagues and I built in the past.^{3,17,20,21} All tests passed perfectly; not a single byte of data was corrupted.

Inspecting the detritus left behind by tests sheds light on what the software under test was doing when power was cut. The `famus_snap` library creates a snapshot of the backing file when application software calls its `msync()` replacement; the caller chooses the snapshot file names and decides when to delete them. The `pft` test application alternates between two snapshot files; `famus_snap`’s rules state that post-crash recovery should replace the backing file with the most recent readable snapshot file.¹⁰

During a month-long test run that completed 49,626 power-off/on test cy-

cles using the pricey flash memory stick, power cuts left the pair of snapshot files in all four logically possible situations: only one snapshot file exists, and it is suitable for recovery (0.054% of tests); both files exist and are full (that is, the same size as the backing file), but only one is readable (4.7%); both snapshot files are full and readable, so recovery must compare their last-mod time-stamps (43.8%); and one file is full and readable, but the other is undersized and writable (51.4%).

These results conform to my expectations based on the relative amounts of time the `pft` application and the `famus_snap` library spend in different states. One way to alter the balance of test coverage would be to trigger power failures from *within* either the `pft` application or the `famus_snap` library, analogous to the inline “crash-point” tests used in the `famus` library⁹ (not to be confused with `famus_snap`).

As Dijkstra famously noted, testing can show the presence of bugs but not their absence. My results don’t prove that `famus_snap` will always uphold its data integrity guarantees, nor that the Raspbian operating system, the XFS file system, the Raspberry Pi computer, or the tested storage devices are reliable under power fault. I can merely report that these artifacts did not avail themselves of numerous opportunities to disappoint. Successful results on a minimalist library such as `famus_snap` and modest hardware such as the Pi furthermore raise the bar for full-featured, expensive hardware and/or software. A commercial relational database running on a server-class host and enterprise-grade storage had better survive tens of thousands of sudden whole-system power interruptions flawlessly—or the vendors have some explaining to do!

Conclusion

Power failures pose the most severe threat to application data integrity, and painful experience teaches that the integrity promises of failure-atomic update mechanisms can’t be taken at face value. Diligent developers and operators insist on confirming integrity claims by extensive firsthand tests. This article presents a simple and inexpensive testbed capable of subjecting storage devices, system software, and

application software to 10,000 sudden whole-system power-interruption tests per week.

The recent `famus_snap` implementation of failure-atomic `msync()` passed tens of thousands of power-failure tests with flying colors, suggesting that all components of the hardware/ software stack—test application code, `famus_snap` library code, XFS file system, operating system, storage devices, and host computer—are either functioning as intended or remarkably lucky.

Future work might adapt the techniques of this article to design testbeds around other types of single-board computers (for example, those based on other CPU types). Arguably the most important direction for future work is the deployment and widespread application of thorough torture-test suites for artifacts that purport to preserve data integrity in the presence of failures. It’s ironic that *performance* benchmarks for transaction-processing systems abound, but *crash-consistency* test suites are comparatively rare, as though speed were more important than correctness. The techniques of this article and methods from the research literature²⁴ have identified effective test strategies. It’s time to put this knowledge into practice. **□**

Related articles on queue.acm.org

Persistent Memory Programming on Conventional Hardware

Terence Kelly

<https://queue.acm.org/detail.cfm?id=3358957>

Fault Injection in Production

John Allspaw

<http://queue.acm.org/detail.cfm?id=2353017>

Abstracting the Geniuses Away from Failure Testing

Peter Alvaro and Severine Tymon

<https://queue.acm.org/detail.cfm?id=3155114>

References

- Allspaw, J. Fault injection in production. *acmqueue* 10, 8 (2012); <http://queue.acm.org/detail.cfm?id=2353017>.
- Alvaro, P. and Tymon, S. Abstracting the geniuses away from failure testing. *acmqueue* 15, 5 (2017); <https://queue.acm.org/detail.cfm?id=3155114>.
- Blattner, A., Dagan, R. and Kelly, T. Generic crash-resilient storage for Indigo and beyond. Technical Report HPL-2013-75, Hewlett-Packard Laboratories, 2013; <http://www.hpl.hp.com/techreports/2013/HPL-2013-75.pdf>.
- Hellwig, C. Failure-atomic writes for file systems and block devices, 2017; <https://lwn.net/Articles/715918/>.
- Hellwig, C. Failure-atomic file updates for Linux. Linux Piter 2019; Presentation: <https://linuxpiter.com/en/materials/2307/>; patches: <https://www.spinics.net/lists/linux-xfs/msg04536.html> and

http://git.infradead.org/users/hch/vfs.git/shortlog/refs/heads/O_ATOMIC.

- Horowitz, P. and Hill, W. *The Art of Electronics, 3rd Edition*. Cambridge University Press, 2015, 38–39, 818.
- Intel. Optane technology; <http://www.intel.com/optane/>.
- Intel. Persistent Memory Development Kit; <http://pmem.io/pmdk/>.
- Kelly, T. `famus`: Failure-Atomic `msync()` in User Space; <http://web.eecs.umich.edu/~tpkelly/famus/>.
- Kelly, T. Good old-fashioned persistent memory. *login*: 44, 4 (2019), 29–34; https://www.usenix.org/system/files/login/articles/login_winter19_08_kelly.pdf. (Source code for `famus_snap` library available at https://www.usenix.org/sites/default/files/kelly_code.tgz.)
- Kelly, T. Persistent memory programming on conventional hardware. *acmqueue* 17, 4 (2019); <https://dl.acm.org/citation.cfm?id=3358957>.
- Linux Programmer’s Manual. `ioctl_ficlone()`; http://man7.org/linux/man-pages/man2/ioctl_ficlone.2.html.
- McCaffrey, C. The verification of a distributed system. *acmqueue* 13, 9 (2016); <http://queue.acm.org/detail.cfm?id=2889274>.
- McManus, S., Cook, M. *Raspberry Pi, 2nd edition*. John Wiley & Sons, 2015, p. 281.
- Narayanan, D. and Hodson, O. Whole-system persistence. In *Proceedings of the 17th Architectural Support for Programming Languages and Operating Systems*, 2012; <https://dl.acm.org/doi/proceedings/10.1145/2150976>.
- Swanson, S. (organizer). Persistent programming in real life (conference); <https://pir.lnvs.io/>.
- Park, S., Kelly, T. and Shen, K. Failure-atomic `msync()`: A simple and efficient mechanism for preserving the integrity of durable data. In *Proceedings of the 8th ACM European Conf. Computer Systems*, 2013; <https://dl.acm.org/citation.cfm?id=2465374>.
- Raspberry Pi 3 Model B+; <https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>.
- TE Connectivity. Axicom relay, product code IM21TS, part number 1-1462039-5. Vendor datasheets: <https://www.te.com/usa-en/product-1-1462039-5.html>; <https://www.te.com/usa-en/product-1-1462039-5.datasheet.pdf>; <https://bit.ly/3eYJ75n>.
- Verma, R., Mendez, A.A., Park, S., Mannarswamy, S., Kelly, T. and Morrey, B. Failure-atomic updates of application data in a Linux file system. In *Proceedings of the 13th Usenix Conference on File and Storage Technologies*, 2015; <https://www.usenix.org/system/files/conference/fast15/fast15-paper-verma.pdf>.
- Verma, R., Mendez, A.A., Park, S., Mannarswamy, S., Kelly, T. and Morrey, B. SQLearn: Database acceleration via atomic file update. Technical Report HPL-2015-103, 2015. Hewlett-Packard Laboratories; <http://www.labs.hpe.com/techreports/2015/HPL-2015-103.pdf>.
- Xu, J. and Swanson, S. NOVA: A log-structured file system for hybrid volatile/nonvolatile main memories. In *Proceedings of the 14th Usenix Conf. File and Storage Technologies*, 2016; <https://www.usenix.org/system/files/conference/fast16/fast16-papers-xu.pdf>.
- Yoo, S., Killian, C., Kelly, T., Cho, H. K. and Plite, S. Composable reliability for asynchronous systems. *Proceedings of the Usenix Annual Technical Conf.*, 2012; <https://www.usenix.org/conference/atc12/technical-sessions/presentation/yoo>.
- Zheng, M., Tucek, J., Huang, D., Qin, F., Lillibridge, M., Yang, E.S., Bill W. Zhao, B.W. and Singh, S. Torturing databases for fun and profit. In *Proceedings of the 11th Usenix Symp. Operating Systems Design and Implementation*, 2014; https://www.usenix.org/system/files/conference/osdi14/osdi14-paper-zheng_mai.pdf (Note that an errata sheet is provided separately.)
- Zheng, M., Tucek, J., Qin, F. and Lillibridge, M. Understanding the robustness of SSDs under power fault. In *Proceedings of the 11th Usenix Conf. File and Storage Technologies*, 2013; <https://www.usenix.org/system/files/conference/fast13/fast13-final80.pdf>.

Terence Kelly (tpkelly@acm.org) spent 14 years at Hewlett-Packard Laboratories. During his final five years at HPL, he developed software support for nonvolatile memory. Kelly now teaches the persistent memory style of programming.

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

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

DOI:10.1145/3366171

Investigating student knowledge transfer and metacognitive activities at college CS departments and at coding bootcamps.

BY QUINN BURKE AND CINAMON SUNRISE BAILEY

Becoming an ‘Adaptive’ Expert

IN TODAY’S SOFTWARE development industry, jobs have become more cognitively complex and require workers who are more collaborative and creative in their problem-solving techniques.¹⁴ Employees also must be able to combine diverse specializations rather than just having routine knowledge in one domain.²² While the “hard” technical skills associated with programming remain a prerequisite for new hires, the industry also wants software developers who can readily demonstrate a range of so-called “soft” skills, including the capacity to communicate clearly, facilitate an open and inclusive workplace environment, and demonstrate the resiliency and flexibility to work on a range of tasks.²⁴

Our own past research⁴ interviewing software industry hiring managers indicates that discerning such soft skills among new hires is an overwhelming priority across companies. The industry hiring managers and directors we interviewed over the past two years stated that while the capacity to code is a necessity for employment, these managers actually spend the vast majority of their recruitment time assessing a candidate’s soft skills, as these suggest the presence of adaptive expertise (AE) and the candidate’s potential for persistence and continual learning on the job.⁴ What was also intriguing to us in discussion with a wide range of hiring managers was their expressed willingness to consider graduates from alternative educational settings—in particular, so-called “coding bootcamps”—alongside more traditional hires from undergraduate computer science (CS) programs.⁴ While there is no single representative model of a coding bootcamp, these intense training programs extend, on average,¹⁴ weeks in duration, cost approximately \$12,000, and emphasize teaching the programming skills that employers look for from new software developer hires (particularly front-end programming) while also enabling their graduates to

» key insights

- While coding bootcamps were once touted as an “alternative” post-secondary educational option, they are now increasingly tapping into college graduates unemployed (or underemployed), despite their college degree.
- In surveys, coding bootcamp students and undergrad CS students didn’t perceive their learning preferences as different, each indicating they liked hands-on activities and collaborative environments. Later, however, undergrads expressed greater discontent with group work and indicated collaboration often only came at the end of their programs.
- Given the difference in program duration, undergrads unsurprisingly reported being exposed to a wider range of computing activities and concepts; but much “depends on the professor” in terms of inculcating AE opportunities.



grasp the most essential aspects of coding.⁶ Much of this expressed willingness to hire codecamp graduates stemmed directly back to hiring managers' perceptions that what bootcamp students may lack in rigorous CS knowledge is counterbalanced with greater work experience and the interpersonal and intrapersonal skills to join a wider team while remaining resilient in the face of unexpected challenges.

This, of course, represented only one party's perspective. Moving forward with our research, we were especially interested in exploring student perspectives from both bootcamps and undergraduate CS programs to better understand to what extent students felt prepared by their respective educational programs. We focused on three research questions: Who are these different programs attracting as learners? How do these students perceive themselves as learners? To what extent do students (at both camp and college) self-report having the opportunity to develop components of adaptive expertise? More specifically, through student perspectives, this research reports how each educational environment promotes guided self-learning, knowledge-transfer, and collaboration on a range of tasks.

Background

While the U.S. education system has redoubled its efforts to promote STEM learning across U.S. classrooms, a recent American Enterprise Institute domestic policy report²¹ aptly points out the STEM education gap is not simply a deficit in the hard cognitive skills associated with science and engineering but also in the soft interpersonal and intrapersonal skills linked with effective communication and collaboration and adaptability. This point has been supported by recent studies examining the requisite skills of effective software developers.^{13,18} However, whereas the general public differentiates between hard and soft skill classification, researchers Hatano and Inagaki¹¹ made the distinction between "routine" and "adaptive" expertise as a more meaningful distinction. Individuals who have mastered a designated set of routines have gained routine expertise. They will continue



Managers actually spend the vast majority of their recruitment time assessing a candidate's soft skills, as these suggest the presence of adaptive expertise and the candidate's potential for persistence and continual learning on the job.



to learn throughout their lifetime, Hatano and Inagaki point out, but will often only apply their new knowledge in a manner that makes the existing procedures/routines more efficient. On the other hand, an individual with adaptive expertise (AE) will utilize the knowledge they obtain and apply it to new, innovative procedures and unexpected problems. According to Bransford,³ "adaptive expertise involves habits of mind, attitudes, and ways of thinking and organizing one's knowledge that are different from routine expertise and that take time to develop." While routine experts possess strong procedural knowledge, adaptive experts are likewise endowed with a strong conceptual knowledge base, allowing them to utilize their understanding to adapt previous mental models and frameworks to new situations.¹¹ There have been differing views on whether or not elements of AE are learned *skills* versus personal *attributes* that individuals either have or do not have upon programmatic entry. Some research indicates that AE can be developed and practiced, but in order to do this, learners must be exposed to metacognitive practices alongside cognitive practices.¹² In particular, research indicates that the development of adaptive expertise can be enhanced through guided self-learning, knowledge-transfer, and collaboration on a range of tasks.

Developing students' adaptive expertise through metacognitive activities and collaboration. Research indicates that guided self-learning practices may enhance the development of AE by promoting metacognition through the students' mindful processing and abstraction in order to apply to new problem sets and innovative solutions. These practices include video reviewing of self or others on the job;² peer coaching;¹⁵ engaging with colleagues (in the field) in collaborative activities;¹ meetings with and/or teaching alongside mentors;⁴ engaging in reflective conversation that draws upon shared real world situations;¹⁶ and error based learning/error management training.⁸

Developing students' adaptive expertise through knowledge-transfer. Research also indicates the development of AE can be further fortified by

providing students with opportunities for transfer of knowledge and adaptability. Effective techniques include providing variation of tasks and projects during practice; placing coursework in the field through internships and capstone coursework;¹⁹ scaffolding by starting with lower variability in tasks in the beginning in order to allow the learner to comprehend concept and abstract general rules prior to introducing higher variability in tasks;²⁵ and helping link previous knowledge to new concepts/practice sets.¹⁰

Based on the existing literature on inculcating AE, we decided to organize and analyze our student data according to two primary categories and 12 sub-indicators:

► First, there are **Metacognitive and Collaborative Activities**, which include the following activities: Video reviewing; peer coaching; engaging with colleagues in collaborative and joint planning, teaching and assessment activities; meetings with and/or teaching alongside mentors; self-reflecting or engaging in reflective conversation that draws on shared real world situations; assisting learners in being open about changing their current way of thinking about problem sets; examining learner products/portfolios that follow instruction; and error based learning/error management training.

► Secondly, the **Transfer of Knowledge and Learner Adaptability** entails providing opportunities to vary tasks on the job; placing coursework with field practice; deliberate scaffolding of tasks into composite parts; and helping link previous knowledge to new concepts/practice sets.

Methods

In our study, the goal was to analyze the degree to which current students and recent graduates from both coding bootcamps and undergraduate CS programs articulate their participation in, appreciation of, and learning from these designated activities associated with metacognition, knowledge transfer, learner adaptability.

Participants. In our investigation, we interviewed a total of 49 students from four different four-year college CS programs (27 students, 12 females, 15 males), as well as from a total of three coding boot camps (22 students, 9 females, 13 males). All programs were located in medium-sized (130K–150K population) southeastern U.S. cities. The sample was one of convenience, and students were recruited through direct visits to bootcamp and college classrooms, as well as through flyers delivered to instructors. Some of the bootcamp students were recruited through email messages posted via the LinkedIn website.

Data collection and analysis. From April 2017 through February 2018, as part of a larger collaborative study, the researchers conducted 49 one-on-one interviews with participating students. Given that participants were unlikely to be familiar with the designations of routine and adaptive expertise, interview questions focused on the more immediate and mundane elements of admission processes; the skills and knowledge they believed they had obtained in each training ground; and, the teaching methods/learning environments characteristic of their respective education programs. It is important to note that interviews were semi-structured in

nature; while they adhered to the three categories identified earlier, questions (particularly follow-up questions) built upon specific responses related to participants’ perception of themselves as learners and the skills and knowledge they felt they themselves gained (or lacked) from coursework. Prior to the interviews, students were asked to complete a 6-point Likert survey (6: strongly agree to 1: strongly disagree) focusing on how they perceived themselves as learners based on Fischer and Peterson’s⁷ survey constructs (Were they open to new ideas/perspectives? Did they try multiple solutions when tackling a problem? Did they prefer to stick with a known solution as opposed to exploring other options?). All focus groups and interviews were subsequently transcribed and qualitatively analyzed using Deoose software.

In terms of the subsequent data organization and analysis, we thematically coded the transcribed interviews first individually and then collaboratively.⁵ With this more in-depth analysis, we paid special attention to what degree coded utterances from participants potentially related to our two primary categories (Metacognitive and Collaborative Activities and Transfer of Knowledge and Learner Adaptability) and 12 sub-criteria representing the specific activities that potentially facilitate AE among learners. Multiple examples of utterances were reviewed between the two researchers in order to determine agreement of categorization. Table 1 provides examples of student responses and the criterion under which it was coded.

Table 1. Coding schema with sample participant utterances.

Example utterance from student participant	Primary Category	Sub-criteria
<i>"Near the end of the bootcamp, we would do a lot of group programming where we worked together, which I found was really helpful 'cause when you're doing a group work and then you have to learn how to explain kind of what you're thinking to your partner"</i>	Metacognitive and Collaborative Activities	► Colleague collaboration ► Self-reflection
<i>"So besides the coursework, to be clear, they do the internship, and then they also have this capstone, sort of a senior closing project, that they do before graduation."</i>	Transfer of Knowledge and Learner Adaptability	► Placing coursework in the field
<i>"[Our instructors] were really straightforward about it, 'Expect there to be lots of errors, expect there to be issues, because that's how you're going to learn. Nobody writes perfect code, the first time.' They were pretty encouraging about it."</i>	Metacognitive and Collaborative Activities	► Error-based learning
<i>"You just keep building on your knowledge base, and evolving your knowledge, and just keep expanding it to incorporate new ideas."</i>	Transfer of Knowledge and Learner Adaptability	► Help link previous knowledge to new concepts/practice sets

Results

Profile of students. The profile of the students and educational settings are listed in Table 2.

These demographics correspond to Course Report’s⁶ national profile of bootcamp students, who are significantly older (mean age of 29 years) and more experienced (six years of work experience, on average) than four-year CS undergraduates. However, it is interesting to note that Course Report also indicates the typical attendee of a bootcamp has never formally worked as a programmer. Yet, 20 out of the 22 (91%) bootcamp participants in this study indicated they had some form of prior training in database management, website development, programming, software development, game creation, CS theory, and/or computer programming languages. It is also interesting how many females participated in this study when considering the national average of women in CS majors has decreased from 30% in 1984 to 18% in 2014.⁹ We expect this is partly reflective of the higher percentage of female undergraduates at two of the participating undergraduate CS programs, which reported having approximately 35% and 40% female enrollments.

With regards to national trends for

undergraduate CS students, the results support a recent report from the National Academies of Sciences, Engineering, and Medicine,²⁰ which points to a growing number of non-majors who take computing courses. In fact, 14 of the 27 undergraduate participants from this study entered the program as undecided-without a CS major in mind, and, notably, all of these same students were initially leaning toward degrees in math, science, and/or engineering.

Students’ self-perception of themselves as learners. The survey results, which were measured on a 6-point Likert scale (6: strongly agree to 1: strongly disagree), revealed that, on average, both undergraduate students and bootcamp students viewed themselves as learners who are open to new ways of looking at things and are willing to change their views when presented with new facts and evidence. Figure 1 presents the average scores on select pre-survey questions from both undergraduates and bootcamp students.

Metacognitive activities. As illustrated in Figure 2, the interviews with undergraduate students revealed their undergraduate setting provided opportunities for metacognitive activities that have been considered useful practices in helping develop metacognition.

Out of the 27 undergraduate students in this study, 19 commented on colleague collaboration activities being implemented in their CS classrooms. This was mostly reflected in peer/group projects during which students had to plan, design, and implement projects and/or tasks as a group. However, it is interesting to note that while students reported collaboration a priority in coursework, many of the undergraduate students did not view group collaboration in a wholly positive manner. For example, during follow-up interviews, one male student in his early 20s stated: “I don’t really like group projects—I guess I like having all of the responsibility instead of depending on someone else ... Sometimes not everyone carries their weight, which isn’t fair for everyone.” A female student at a separate university seconded his thoughts: “I do think that my peers would generally prefer less group projects. Group projects are not generally thought of positively just because you so often get bad eggs in your groups, and some professors don’t let you choose your team. Maybe in companies you’re not able to choose your teams, but if you’re working at a startup, you are absolutely able to choose who you start a company with, and I think such an important skill to acquire is

Table 2. Participant/institutional profile.

Program	Participants/ Gender	Ethnicity/Race	Institutional Profile	Student Reasons for Attending	Students Prior Education/Work Experience
Coding Boot Camp	22 (9 females, 3 males) Age (early 20s to mid 50s)	14 Caucasian 3 Black/African American 2 Asian/ South Asian 1 Latinx 2 Unknown	3 Bootcamps (2 mid- sized cities in GA and SC) Programs lasted 12-15 weeks on average	86.5% wanted to advance within company, update skill set; and/or change careers 27.0% wanted hands-on instruction and/or practical real-world experience	82% already had a bachelor’s degree or higher 91% had some form of training in database management, website development, programming, software development, game creation, CS theory, and/or computer programming languages 27% indicated that they had some form of university or community college education in CS or website design 45% stated they were self-taught through free online courses via YouTube, free-Code-Camp, and Codecademy
Undergraduate	27 (12 females, 15 males) Mean age 22 y/o	16 Caucasian 4 Black/African American 3 Asian/South Asian 1 Latinx 1 Kurdish 3 Unknown	4 Universities (2 mid-sized cities in SC) Programs 3-5 years on average	25% chose this training ground due to financial aid or scholarships 14% chose this route because of the importance industry places on having a bachelor’s degree or higher 50% entered as undecided without a CS major in mind	36% had prior work or internship experience 68% were exposed to CS related courses/assignments while in HS, had self-taught themselves via free computer programming online courses, and/or were taught about software and hardware development by tech adjacent family members and friends

your ability to identify who is a good worker and who you would mesh with and be a good business partner with.” Another student, although stating she doesn’t enjoy group projects, was able to reflect on the positives as indicated by stating “I’m not an advocate for group work in college just because if you have bad group partners, they affect your grade. I pay a lot of money to go to school. But they do emphasize teamwork, and I think maybe half of my classes included a large project with a team. And that does definitely increase the ability for people to communicate and work together, especially people who are more introverted ...” In some cases, undergraduate students were not permitted opportunities to collaborate. A 19-year-old male indicated that collaboration is not a programmatic expectation but varies based on the disposition of instructors. “(S)ome professors don’t allow collaboration,” he states, “but some professors do. It just depends on if we’re allowed to talk about the assignments outside of class.”

As mentioned earlier, collaborating with mentors in the field may contribute to the development of AE. Undergraduate participants indicated they encountered these mentors over their junior and/or senior year of their respective programs through class presentations provided by the visiting industry representatives regarding the interview process and the workplace environment (if it occurred at all). Some instructional settings, according to these students, implemented peer coaching as a method by which to increase feedback about instruction and curriculum. One cited example referred to the potential of reviewing coded projects as a class in order to have peers comment on errors and/or suggest alternative solutions. In this study though, only nine undergraduate CS students (one-third of the total undergraduate participants) commented on the presence of peer-coaching within their respective programs. And eight of these nine students noted that peer review was largely informal in nature and was contingent on whether the professor allowed for students to talk to each other about projects during class time.

With regards to receiving feedback from the instructor, 19 of the 27 (70%) undergraduates spoke about their

Figure 1. Average student Likert rating on survey responses.

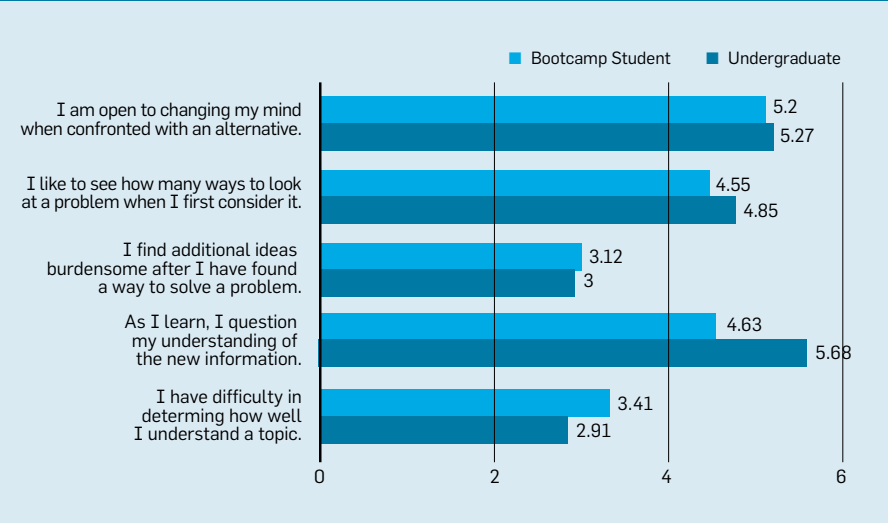
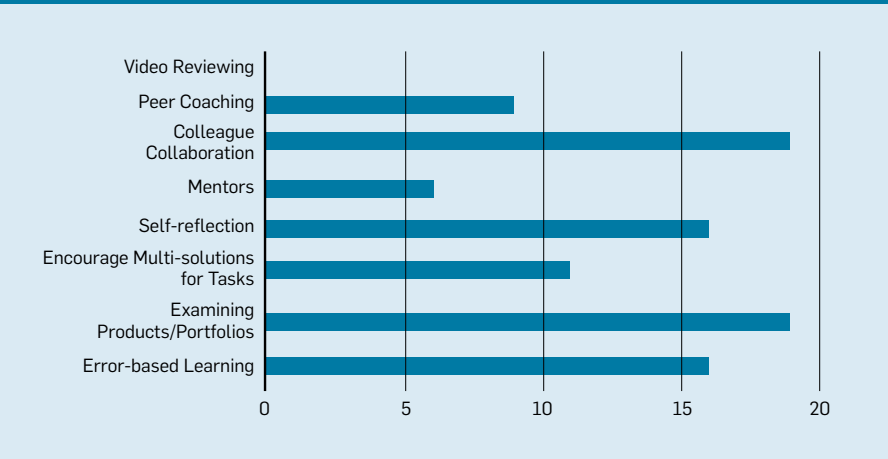


Figure 2. Undergraduate opportunities for metacognitive activities and collaboration.

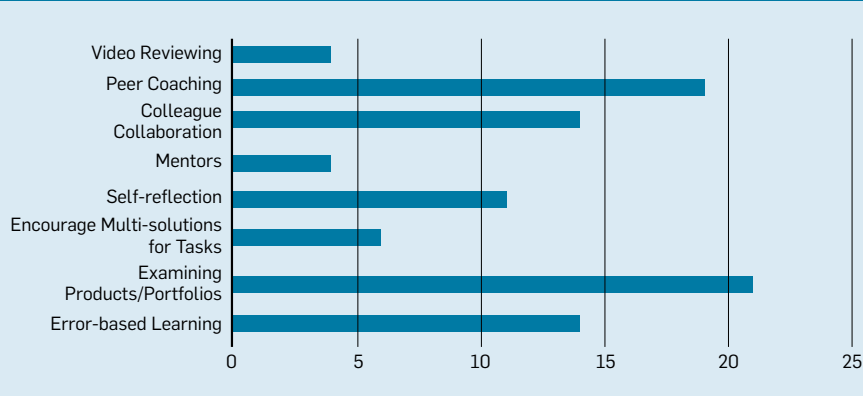


professors examining their products and/or portfolios following instruction. This allowed students to receive feedback regarding their progress in the course, their performance on a task, and/or their conceptual knowledge. Slightly more than half (11 of the 19) of these students indicated that the design of the course and the feedback they received allowed for them to discover their errors and learn from their mistakes. For example, a female student in her early 20s stated that “a couple of weeks ago, we had to make a program that sorted a list of random numbers. And I couldn’t get it to work just right ... But the time was out so I had to send it in. She sent it back explaining to me how I could tweak it and she said I could send it back in. So, I looked at what she had said in her comment and it allowed me to understand what I was doing wrong.” However, not all of the university students viewed the feedback

process in a positive light. According to one student, whether or not one received timely opportunities to correct errors “depends mostly on the professor. Some allow you to redo assignments ... They’ll give you feedback on it and you can try again. Others, you just have to get the grade you get.” Sometimes, undergraduate students do not receive any consistent feedback in certain courses, with two students reporting that their instructors did not formally grade them until the final weeks of the semester.

Finally, in terms of metacognitive activities on the undergraduate level, several students evaluated whether or not their learning environment promotes self-reflection on the learning process. Sixteen indicated the design of the course, the professor’s instruction, and/or design of the curriculum provided them with an opportunity for self-reflection on how they learned, an attribute that has been noted in research as beneficial in

Figure 3. Bootcamp opportunities for metacognitive activities.



the development of metacognition. One female senior in her early 20s stated the program promoted “general abstract thinking ... just learning to remove myself from the implementation only and looking more at how I can think about this instead of just how I can code this.” Some 41% of these undergraduates likewise mentioned their respective programs provided opportunities for them to question their thinking and challenged them to be open about changing their current way of thinking about problem sets.

In terms of metacognitive activities among coding camps, the interviews with the 22 bootcamp research participants also revealed their training ground provided opportunities characteristic of practices used to develop metacognition. This is illustrated in Figure 3.

Nineteen of the bootcamp students commented on colleague collaboration activities being implemented in their bootcamp training grounds. Similar to the undergraduate students, these students indicated this mostly occurred in peer/group projects during which students had to plan, design, and implement projects and/or tasks as a group or a pair. Notably, unlike the undergraduate students, the bootcamp students did not list any negative reasons for peer/group work. With regards to mentors and collaboration, only 18% of these students indicated that industry mentors collaborated with them; and just as with the undergraduate participants, all of the bootcamp students indicated that the mentors spoke with their class about the interview process and what the industry workplace environment was like.

In the bootcamp environment, 19 students indicated their training included deliberate peer coaching as a means of receiving feedback on projects and

clarification of concepts. This is interesting in that a much smaller number of undergraduates in this study either did not comment on peer coaching or indicated that it did not occur. One such example came from a bootcamp graduate, who is now employed in healthcare software development: “My instructor would give us an assignment on Monday. And then Tuesday morning from 9 AM to 9:30 AM, he would call on a few different students, and they would come to the front of the class and plug their computer up to the projector and we would look at each other’s code. You’re looking at code that you wrote the night before in front of 10 to 15 other people and he’s telling you what you did wrong and what you did right and what you could improve on, and so would the other students.” In addition, four students stated the instructor provided in-person or video demos prior to starting a project so that the students were able to visualize their task at hand. For example, a male student in his mid-30s stated: “most of the assignments that we did were demo-ed first. So, we’d be like, ‘Okay we’re gonna do this.’ This is how you do it and we would just watch, and then we would go back and do it all again. But we would, like, do it along with him, with the instructor. Then, the third time would be no instructor just try to do it yourself.”

With regards to receiving feedback from the instructor, nearly all participating bootcamp students (approximately 95%) spoke about their instructors reviewing their products and/or portfolios following instruction. In some of these cases, it was in the form of active learning using the flipped classroom model: “We did the flipped classroom model where we would do some sort of reading ahead of time and

then the next day we would have activities so that we could show that we’ve grasped the concept and kinda fill in any knowledge gaps that we may have.”

A commensurate percentage of bootcamp students (63.6%) mentioned that feedback encouraged learning through errors. “It’s one of the things they tell you. Go in, break the code, once you break it you know what not to do, and so now we know that doesn’t work, so let’s try something else.” In addition, 27.3% of bootcamp students in this study explicitly mentioned their instructors and/or program assisted them in being open about changing their current way of thinking about a problem set. For example, one student stated, “they would never be like, ‘this is an error and this is a problem.’ They would just be like ‘I see that you did it this way, did you consider doing it this way instead.’ Things of that nature.”

Transfer of knowledge and learner adaptability. The responses of undergraduate students revealed the college setting provided opportunities for transfer of knowledge and adaptability. This is illustrated in Figure 4.

Nineteen undergraduate students (70%) in this study indicated they were exposed to a variation of tasks and projects during practice. This does not necessarily take place during the same course. Instead, it is the exposure to multiple coursework practices and instruction over their college career.

Meanwhile, 25 of these students (93%) mentioned that their programs have included placing coursework in the field, such as internships, class/individual projects with real world projects, or capstone projects. The students were very positive regarding this aspect of their program. For example, one student stated “I would say the college does a very good job with their final two, kind of like capstone-ish courses with the software engineering and then the software engineering practicum. I feel like that course in its own gives you the most experience to what it’s like to work for, to work on a project in a group. That’s your first real exposure to a full-fledged product top to bottom. So, I feel like that course is probably the most important course I’ve taken here.”

Undergraduate students also spoke about how their professors and program helped them link previous knowledge to new concepts/practice sets and that

this has also helped in the area of self-reflection. For example, a senior stated “going to a different programming language is really just a different wording and syntax of the same concept. As you learn one and get good at one or get good at two and you go to different ones and you start to see the same things reappearing that are very similar to the things you’ve done before. Then once you start making those connections between the subsequent ones and the first ones that you learned, that’s when it really starts to click for me.” Several of these students reflected their classes or activities, outside of the CS program, such as music, sports, and foreign languages, provided them with skills and knowledge, which they need in their CS task work.

Finally, seven undergraduate students mentioned that scaffolding has been used in order to help them transfer knowledge by starting with smaller tasks in the beginning in order to allow for them to comprehend the concepts and/or abstract general rules, followed by higher variability in tasks later. For example, a female senior student stated: “in Operating Systems, I had to design a shell, and we sort of did that piece by piece, adding in functionality as we went. At first, we could only run one command. Then we could chain commands.”

The interviews with bootcamp students revealed their setting likewise provided opportunities for transfer of knowledge and adaptability. This is illustrated in Figure 5.

Eleven bootcamp students in this study indicated they have been exposed to a variation of tasks and projects during practice. The other 50% indicated the training group focused on a single language or technology with more routine practices (that is, practice the same types of tasks throughout the program). Thus, this seems to be a matter of differences between different bootcamps.

Nine students of these bootcamp students mentioned their programs have included placing coursework in the field, such as class/individual tasks reflecting real world projects. As an example, one bootcamp student noted “[T]here’s 10 weeks of class time where you’re learning new subjects and, then, the final two weeks is when you choose a project and you write a full stack Web application. That’s the first time you really get the

chance to put everything together. That was really cool because we used absolutely everything we learned without exception during that final two weeks.”

Bootcamp students (36%) also spoke about how their instructors and program helped them link previous knowledge to new concepts/practice sets and that this has also helped in the area of self-reflection. Many times, it was also linked to employability. For example, one learner stated that “[t]hey kind of show you how what you already know can help you with what you need to learn. So how already knowing Java will help me in the future for C+ or C++ or whatever else I may need to learn in the future—showing how to kind of mesh the two so that you can quickly pick up new languages if you have to learn a new language when you get to your first job.”

Finally, five of these bootcamp students mentioned that scaffolding has been used in order to help them transfer knowledge to other tasks and/or concepts. In some cases, these scaffolding exercises also helped students build a portfolio. For example, one student stated that “as far as the assignments go, we were given assignments ... early on in the course and as the course progressed we were given kind of more complex assignments that maybe built

off of itself and we’d start a project on Monday and it would be get the bare bones done, and then on Tuesday, we’d implement a feature and so on and so forth and kind of build-up different individual portfolio pieces.”

Discussion

Returning to our initial research questions, the first research question about who such programs attract proved unsurprising. In terms of entering students, age and work experience were the leading differentiators between these southeastern undergraduate and bootcamp students in this study, and the student profiles closely correspond to Course Report’s national annual statistics (29 years of age, six-year work experience).

In terms of the second research question about how undergraduates and bootcamp students perceive themselves as learners, we were surprised to find little difference between the two groups. In surveys, college students and the more mature code camp students both reported themselves as hands-on learners who enjoyed working in teams on a range of tasks. Of course, some of this self-reported survey data was undercut by individual interview responses with these same students where a number of undergraduates (but interestingly no

Figure 4. Opportunities for transfer of knowledge and adaptability in the undergraduate classroom.

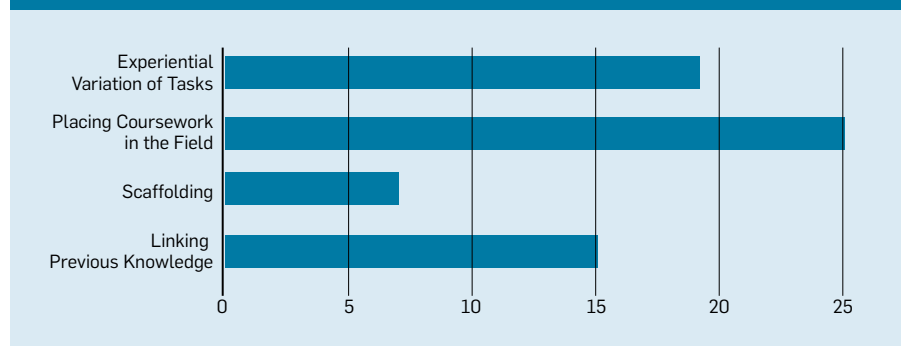
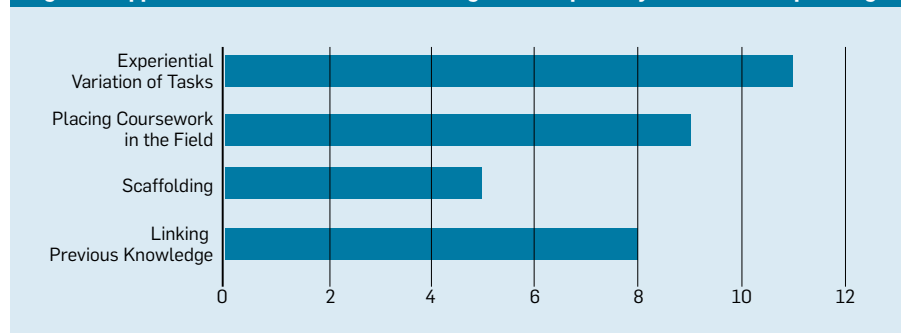


Figure 5. Opportunities for transfer of knowledge and adaptability in the bootcamp setting.




code camp students) expressed reservations about the actual productivity of working in groups.

The third research question investigating to what extent each learning environment inculcates adaptive expertise proved most telling. Clearly the short and intensive timeline associated with bootcamps meant their students had to be ready to “drop in” and work closely with each other and their instructor from day one. This is evident with the fact that virtually no bootcamp participant pushed back on the value of group work and 86% indicated that peer coaching was built into their coursework. A smaller number of undergraduates spoke on peer coaching within their program. Bootcamp students were also more likely to report receiving timely feedback from their instructor (96% compared to 70% of undergraduate students). Of course, this finding must be tempered by the fact that the majority of code camp students reported having only one to two instructors over the course of their respective programs, while CS undergraduates had eight plus instructors and consistently indicated that the nature of classroom collaboration and feedback was highly contingent on the individual instructor (that is, a regular response from undergraduates during interviews was “it depends on the professor”). Yet this range of instructors corresponds to a wider range of activities on the undergraduate level with 74% of college students reporting exposure to a variation of tasks and projects; whereas, it depended on the bootcamp chosen with regards to whether the student was exposed to a variety of tasks versus routine practices of the same task. Here, bootcamp students also reported significantly less opportunities for knowledge transfer, likely due to the relative brevity of their programs.

While it is not surprising that program duration and student age play no small role in distinguishing between college and camp, it is important to note that based on the survey data, the students themselves did not perceive themselves any differently as hands-on, collaborative learners. The peer-to-peer collaboration and immediacy of feedback more widely characteristic of coding camps (according to student interviews) is a learning environment that students from both groups prefer. Although the number of recruited participants was

small, this could have implications for CS departments nationally, where CS1 classes have swollen to capacity and instructors are increasingly finding themselves using the initial coursework as a means to sort and “weed out” students from the major. Meanwhile, the “hands on” collaborative coursework characteristic of CS capstones only comes at the end of four-year programs. This pedagogical approach has wider implications around inclusivity and diversity in the CS field, which colleagues are currently investigating¹⁷ as CS persists as being inordinately populated by Caucasian and Asian males.

The alternative environment of coding bootcamps offers a new stream into the CS field, and, as a recent paper entitled “Betting on Bootcamps”²³ indicated, such camps could very well be a potential disruptor in higher education. Yet, as evident with our study, with some bootcamps recruiting and admitting students *already with a college degree*, to what degree these camps will truly become an alternative to college seems unlikely. As an alternative to graduate education, particularly master’s degrees, their future appears much more secure. Future studies involving a larger sample size in various geographic regions could help increase knowledge regarding these two training settings.

Acknowledgment. This article was supported through a National Science Foundation (NSF) Core Research and Development award (#1561705) to the first author. The views expressed are solely those of the authors. 

References

1. Anthony, G., Haigh, M. and Kane, R. The power of the ‘object’ to influence teacher induction outcomes. *Teaching and Teacher Education* 27 (2011), 861–840.
2. Blomberg, G., Stürmer, K. and Seidel, T. How pre-service teachers observe teaching on video: Effects of viewers’ teaching subjects and the subject of the video. *Teaching and Teacher Education* 27 (2011), 1131–1140.
3. Bransford, J. 2001. Thoughts on adaptive expertise (unpublished manuscript), 2001; <http://www.vanthonline.org/docs/AdaptiveExpertise.pdf>.
4. Burke, Q., Bailey, C.S., Lyon, L. and Green, E. Assessing industry’s perspective of coding boot camps through the lens of routine and adaptive expertise. In *Proceedings of 50th ACM Technical Symposium on Computer Science Education*, 2018, 345–350.
5. Campbell, J.L., Quincy, C. and Osserman, J. Coding in-depth semi structured interviews: Problems of unitization and intercoder reliability and agreement. *Sociological Methods & Research* 42, 3 (2013), 294–320.
6. Course Report. 2018 coding bootcamp market size; study; <https://www.coursereport.com/reports/2018-coding-bootcamp-market-size-research>.
7. Fischer, F.T. and Peterson, P.L. A tool to measure adaptive expertise biomedical engineering students. In *Proceedings from the American Society for Engineering Education Annual Conf. and Exposition*. Northwestern University, Evanston, IL, 2001.
8. Frese, M. and Altmann, A. The treatment of errors in learning and transfer. L. Bainbridge and S.R. Quintanilla,

- Eds. *Developing Skills with New Technology*. John Wiley, Chichester, England, U.K., 1989.
9. Galvin, G. Study: Middle school is key to girls coding interest. 2016; <https://bit.ly/2KmpLYW>
10. Gick, M.L. and Holyoak, K.J. The cognitive basis of knowledge transfer. S.M. Cormier and J.D. Haginan, Eds., *Transfer of Training: Contemporary Research and Applications*. Academic Press, New York, NY, 1987, 9–46.
11. Hatano, G. and Inagaki, K. Two courses of expertise. H. Stevenson, H. Azuma and K. Hakuta, Eds. *Child Development and Education in Japan*. W.H. Freeman, NY, 1986, 262–272.
12. Hicks, N., Bumbaco, A.E. and Douglas, E.P. Critical thinking, reflective practice, and adaptive expertise in engineering. In *Proceedings of the ASEE Annual Conf. and Exposition* (Indianapolis, IN, USA, 2014).
13. Holtkamp P., Jokinen, J.P. and Pawlowski, J.M. Soft competency requirements in requirements engineering, software design, implementation, and testing. *J. Systems and Software* 101 (2015), 136–146.
14. Li, P.L., Ko, A.J. and Begel, A. Cross-disciplinary perspectives on collaborations with software engineers. In *Proceedings of the 10th International Workshop on Cooperative and Human Aspects of Software Engineering*. 2017, 2–8.
15. Lu, H. Research on peer coaching in preservice teacher education—A review of literature. *Teaching and Teacher Education* 26 (2010), 748–753.
16. Lunenberg, M. and Samaras, A. Developing a pedagogy for teaching self-study research: Lessons learned across the Atlantic. *Teaching and Teacher Education* 27 (2011), 841–850.
17. Lyon, L.A. and Green, E. (In review). Women in coding boot camps: How colleges miss opportunities to broaden participation in computing.
18. Maturro, Raschetti and Fontán, A systematic mapping study on soft skills in software engineering. *J. Universal Computer Science* 25, 1 (2019) 16–41.
19. Montecinos, C., Walker, H., Rittershaussen, S., Nuñez, C. Contreras, I. and Solís, M. Defining content or field-based coursework: Contrasting the perspectives of secondary preservice teachers and their teacher preparation curricula. *Teaching and Teacher Education* 27 (2011), 278–288.
20. National Academies of Sciences, Engineering, and Medicine. Assessing and responding to the growth of computer science undergraduate enrollments. The National Academies Press, Washington, D.C., 2018; <https://doi.org/10.17226/24926>.
21. Orell, B. STEM without fruit: How noncognitive skills improve workforce. *Outcomes*, 2018; <http://www.aei.org/wp-content/uploads/2018/11/STEM-Without-Fruit.pdf>.
22. Pellegrino, J.W. and Hilton, M.L., eds Education for life and work: Developing transferable knowledge and skills in the 21st century. National Academies Press, Washington, D.C., 2012.
23. Price, R. and Dunagan, A. Betting on bootcamps: How short-course training programs could change the landscape of higher ed. 2019; <https://www.christenseninstitute.org/publications/bootcamps/>
24. Robles, M.M. Executive perceptions of the top 10 soft skills needed in today’s workplace. *Sage Journals* 75, 4 (2012), 453–465; <https://doi.org/10.1177/1080569912460400>.
25. Schmidt, R.A. and Bjork, R.A. New conceptualizations of practice: Common principles in three paradigms suggest new concepts for training. *Psychological Science* 3 (1992), 207–217.

Quinn Burke (qburke@digitalpromise.org) is Senior Research Scientist at Digital Promise, San Mateo, CA, USA.

Cinamon Sunrise Bailey (cinamob@g.clemson.edu) is a doctoral student at Clemson University, Clemon, SC, USA.

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



Watch the authors discuss this work in the exclusive *Communications* video. <https://caem.acm.org/videos/becoming-an-adaptive-expert>

Exploring what leaders can do to improve and sustain social alignment over time.

BY ANDREW BURTON-JONES, ALICIA GILCHRIST,
PETER GREEN, AND MICHAEL DRAHEIM

Improving Social Alignment During Digital Transformation

SOCIAL ALIGNMENT AMONG groups of stakeholders occurs when different stakeholder groups share understanding of a business outcome and commit to the outcome and the means to achieve it.¹² Project management is more effective and efficient when

stakeholders are socially aligned because it reduces friction each time a project decision is made. Without agreement among stakeholders as to what needs to be accomplished and how to do so, success becomes harder to achieve. While the benefits of stakeholder social alignment are clear, how project stakeholders can move toward and sustain social alignment remains unknown.¹³

To address this issue, we sought to determine how social alignment or misalignment develops, and how leaders can improve social alignment over time. We had a unique chance to learn answers to these questions through our involvement in a longitudinal case study of a digital transformation. The case involved the launch of one of Australia's first large-scale digital hospitals, one of the most significant organizational changes ever undertaken by

an Australian health service.³ Given the size and consequences of transformational projects, this is a context where social alignment is likely to be critical.

We found in our research that the process of achieving *social misalignment* involved four phases as did the process of achieving *social alignment*. These processes were linked, such that stakeholders moved through misalignment and alignment in a non-linear fashion. When we studied the trajectory closely and mapped it out, we noticed improvements in the trajectory

a It was transformational due to its size (a large, multi-site implementation) and its effect (dramatically changing how the service functioned). For a similar definition, see Burton-Jones et al.³ After implementation, the site achieved Stage 6 on the HIMSS Electronic Medical Record Adoption Model (EMRAM), one of only three Australian hospitals at that time, and the largest of them, to be so designated.

due to practices that managers adopted. We also noticed poor trajectories when managers enacted these practices ineffectively or not at all. Through analysis, we identified three practices that help improve social alignment—Ramping, Holding, and Peaking.

In this article, we identify and describe the characteristics of these practices for improving social alignment in large complex projects. We begin by describing the project's context, because every project is different and the context of our site may have affected what we observed. With this context laid out, we then describe how alignment and misalignment developed at the sites,^b followed by the practices that helped improve alignment.

The Digital Hospital Transformation

We studied the rollout of a Digital Hospital implementation across public hospitals in a state in Australia. Two prior attempts had failed and it was widely agreed this was the final chance. Many factors involved were similar to those in comparable cases internationally, for example, major work practice changes, significant training, the required buy-in of clinical groups.¹¹ However, some factors were also very influential in this local setting. We highlight two:

Institutional complexity and risk: The rollout was supported by the state government but each hospital service across the state was quasi-independent. Thus, each hospital's implementation involved multiple layers of governance—state, health-service, and hospital. The prior failed implementations had been led by the state, but this time the hospital services were taking greater control. Risks were significant because the health sector had experienced numerous recent IT scandals, including one of the largest IT failures in Australia's history.⁴ Digital hospital implementations globally were also being criticized.⁸

Strong funding and leadership: The project was relatively well-funded, with over AUD\$0.5B invested by the state and other funds co-invested by each health service to go-live. The project also benefited from strong leadership.

^b Earlier versions of this first part of our analysis are available in Gilchrist et al.^{5,6}



We found the process of social misalignment involved four phases as did the process of achieving social alignment.

The CEO of one health service was a strong advocate and took it upon him/herself to wrest control of the project from the State, volunteer for his/her service to go live first, and prepare the health service for change.

Two hospitals were selected to be the lead hospitals in the rollout, with others to follow. Each hospital was to implement a U.S.-developed Digital Hospital solution configured to the Australian context, with each site joining the one system instance (that is, one system for the state).^c The two lead sites were called *configuration sites* because the system was configured to their needs as representatives of the other hospitals. To reduce risk, the implementation at the configuration sites and first few rollout sites was split into two waves. The first involved implementing modules for scheduling, documenting, order-entry and results-reporting, and wireless device integration. The second involved modules for research trials, anaesthetics, and medications. Reports and dashboards were also developed across both waves.

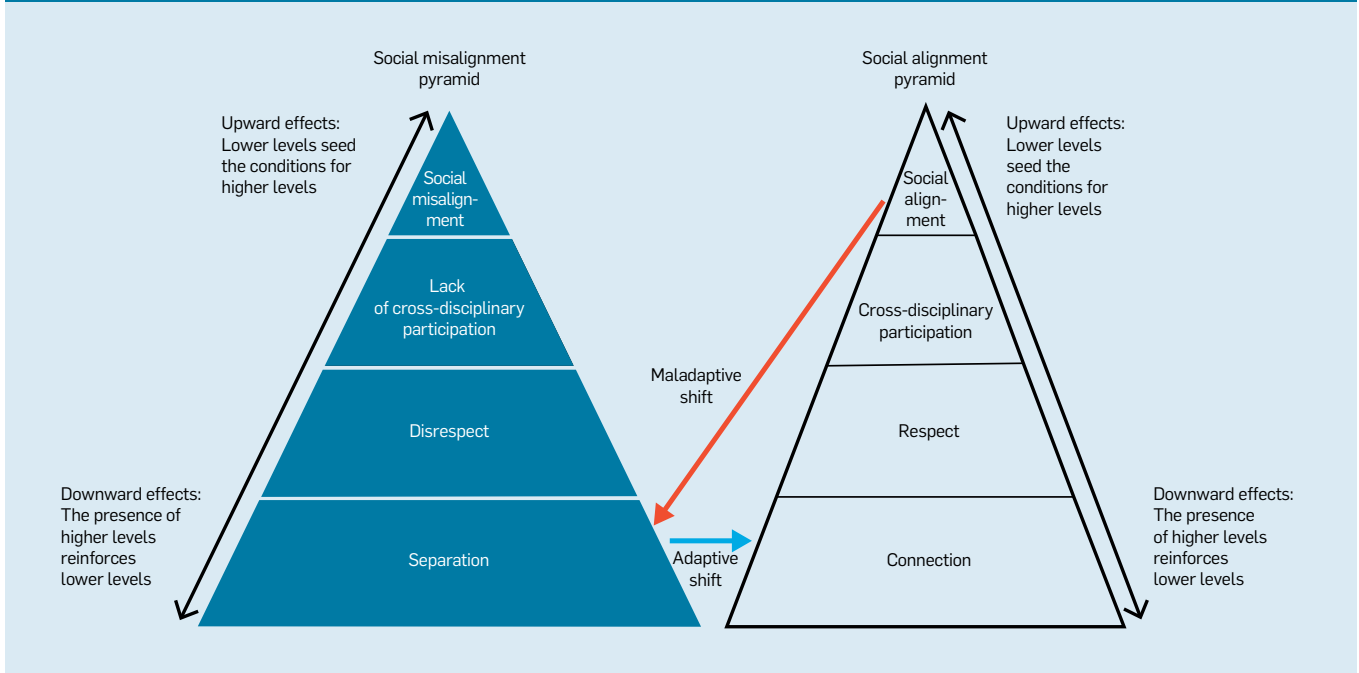
The findings in this article stem mostly from our findings at the larger of the two co-lead hospitals, which was the first hospital to go live in the state. The board meetings for the statewide rollout, which we attended, were held at this site. Through our attendance in these meetings, we observed how social alignment evolved during the implementation, not just at this site, but at other sites too. We also conducted 30 interviews with members from all major groups during the project to learn their perceptions and experiences over time, following an inductive research approach.⁷

How Social Alignment and Misalignment Developed

We found that social alignment evolved non-linearly during the digital hospital

^c Such systems are often known by their component functionality, such as an electronic medical record (EMR), electronic prescribing (e-prescribing), computerized decision support system (CDSS), and computerized physician order entry system (CPOE). The implementation we studied comprised all these components together with wireless device integration and a full (and growing) set of reporting and data analytic components, with the aim of transforming how the hospitals provided their services.

Figure 1. Social alignment and misalignment observed in the case.



project, shifting between misalignment and alignment. As Figure 1 shows, we identified four phases of social misalignment and four phases of social alignment.⁵ The accompanying table provides illustrative quotes from our interviews describing each phase.

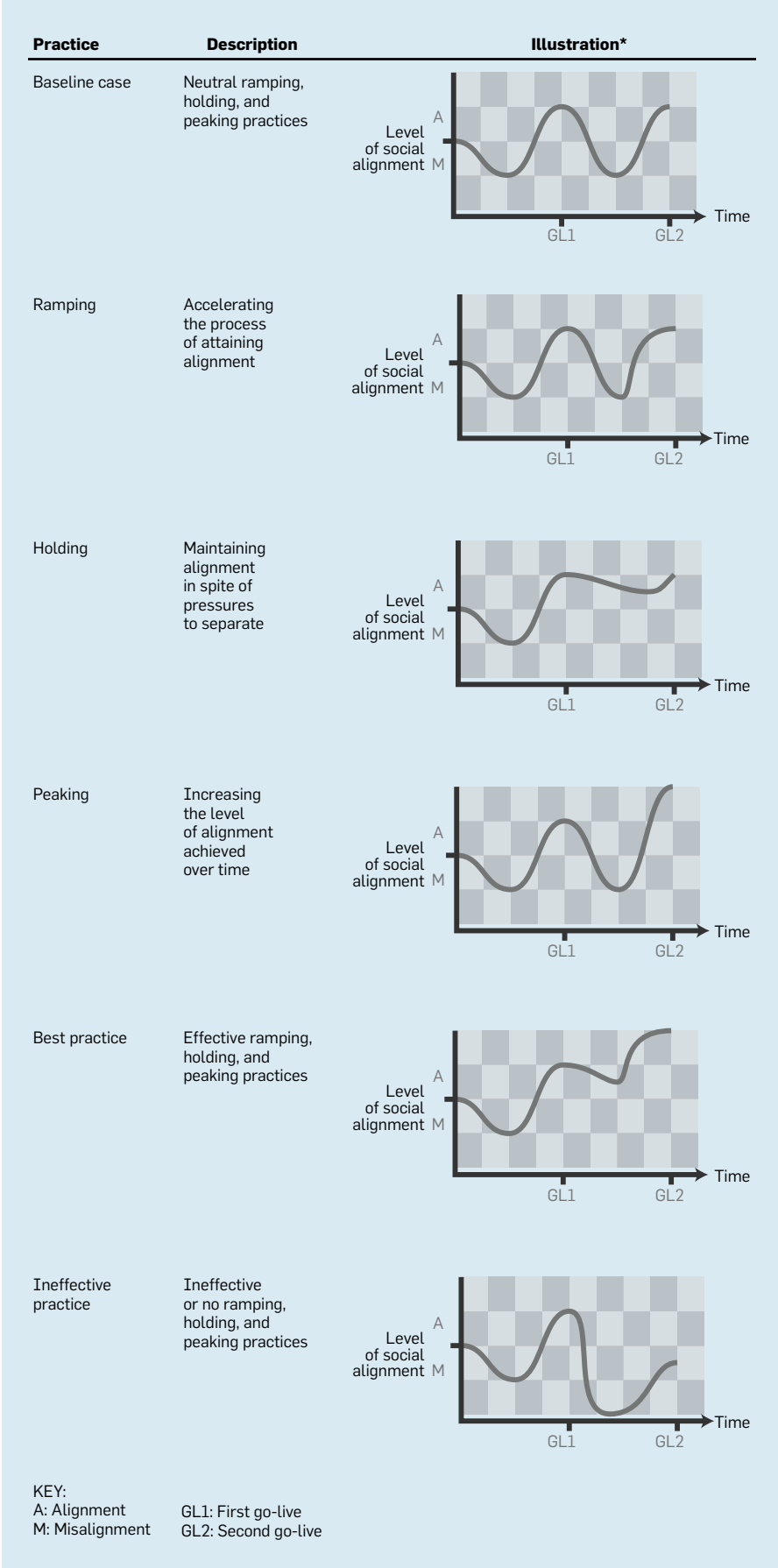
Our interpretation of our data was that periods of misalignment and alignment operated like two opposing pyramids that stakeholders scaled, with alignment and misalignment as the two peaks. We refer to them as pyramids because the lower levels appeared to provide the conditions for higher levels. For instance, the first phase of misalignment (separation) seeded the conditions for disrespect to emerge, which in turn seeded the conditions for lack of cross-disciplinary participation, which then led to social misalignment. Each higher level then fed into and reinforced conditions at lower levels, in turn deepening the issues at higher levels.

Fortunately, we found periods of misalignment could be broken if the stakeholders self-corrected and shifted adaptively toward alignment. The trigger was to identify feelings of separation as signaling the need to reconnect. For instance, project managers and clinicians typically come from very separate professional groups, but they could learn much from each other. Once they reconnected, stakeholders began to learn from each other, which

Example quotes to illustrate the phases of alignment and misalignment.

Phases	Example quotes
Misalignment phases	
Social misalignment	"There was a disconnect between the project [and the business]. The project's just going ahead and the business has gone 'Well, where the hell are we going?'"
Lack of cross-disciplinary participation	"So what we worry about is that there's been a number of meetings around this and no one's involved us."
Disrespect	"The people ... in those meetings didn't have the power to change it anyway. They were just project people with a clipboard of tasks that they had to get done. ...They would just nod politely and move on [to] the next thing on their little Gantt chart."
Separation	"So, we've been set up as silos, essentially, but we're [supposed to be] implementing a system that is fully integrated."
Alignment phases	
Social alignment	"The governance over [the implementation] was awesome. ...All those competing priorities—those competing people and groups and organizations ... just coalesced around 'this is what we're doing' and it worked really well, and you would've sat in on some of the meetings to see everybody was in the room and they were able to say, 'This is what I can do to make that happen.'"
Cross-disciplinary participation	"One of the things I've said to [my colleague] is 'can you go and take the group who's going to be full medicine or surgery, take them to the executive meeting, introduce them, get people to see them, names, faces, whatever else, do that. Say how would you like then for these people to be part of your unit to help support them'"
Respect	"You need to be listening. You've got to be listening and taking heed. You can have a command and control structure but you've also got to be turned on to what's actually happening and what people are saying, and respecting the information that's available and making sense of that."
Connection	"And there's a couple of things that come out of [all three groups coming together for scenario building], one is ... they start to understand what the other one is doing. But they also start to have a bit of knowledge building going on in both areas."

Figure 2. Improved social alignment.



led to respect, which then provided the foundation for cross-disciplinary participation and onto social alignment. Unfortunately, we also found social alignment was not necessarily self-sustaining because stakeholders could shift maladaptively back to periods of misalignment.

When we began our data collection at the project's inception phase, the first stage we observed was separation. This was natural because the different groups (clinicians, project managers, and executives) had such different backgrounds and historically had not been part of the same team. During the early period of our data collection, well before the first *go-live*, we found separation led to disrespect and growing social misalignment. However, as *go-live* approached, we found the negativity of misalignment and the presence of separation became salient, leading stakeholders to switch mindsets. They began to reconnect which helped them learn and respect each other and then work together. This led to a peak of social alignment around the *go-live* date. We then observed that the team reverted back to previous silos after *go-live*, leading to a phase of misalignment, only to rise again for the second *go-live*. In short, we observed an inverted *S*-shaped curve, falling, rising, falling, and rising again.

The Baseline case in Figure 2 shows our first impression of this curve. However, as we analyzed our interview data, we found it did not paint a correct picture because the second period of misalignment did not appear as strong as the first, while the second period of alignment appeared stronger. This led us to collect more data to learn what was causing the improved trajectory. We discovered several practices that were improving alignment over time. That is, while some aspects of the process we were observing were quite natural and perhaps to be expected, it did not have to be that way. We felt these insights could help other researchers and practitioners too given that prior studies have observed performance dips and learning curves in past work and have explicitly called for further study of them.⁹

How Social Alignment Can Be Improved

Figure 2 illustrates the practices we discovered in the case and how they

helped improve the alignment trajectory. We identified the practices from our data and confirmed them with stakeholders at the study site. The graphs are stylized rather than derived from quantitative data, but they reflect our interpretation of our data.^d Next, we discuss each practice and then discuss their combination in terms of effective/ineffective practice.

Improving Social Alignment through Ramping

Ramping practices accelerate alignment by getting stakeholders to commit at a faster rate than they otherwise would. This is shown in Figure 2, where the second rise in the curve for Ramping rises faster than the second rise in the Baseline curve. Three practices appeared to improve Ramping. The common idea underlying all three is that to commit early and strongly, stakeholders had to believe the project was *credible*—that it would go ahead and succeed. There was a long history of failed IT projects in that region, and in EMR projects globally. No one wanted to be part of a failure.

Creating multifunctional teams with recognized leaders. The first important Ramping practice was to create multifunctional teams with recognized leaders at all levels. Project teams included clinicians from all major professional groups, directly enabling the cross-disciplinary participation shown earlier in Figure 1. Individuals were sought who had strong leadership qualities and peer respect. Getting doctors involved was especially critical and the hospital involved key doctors who had respect across all subspecialties and across all levels of seniority. In addition to helping internally, this helped motivate external stakeholders to get involved.

For instance, the Deputy Chair of one medical division began leading one project subgroup. Until that time, junior vendor representatives attended these meetings but at the end of an early meeting, the doctor stressed that he/she expected someone of his/her equivalent level to attend. From then on, a leader from the vendor attended all

those meetings. Likewise, clinicians from outside the hospital who needed to provide input at key stages gave input early, and caused less obstruction when they disagreed, because they knew the reputation of those involved.

Motivating attention to operational matters as the foundation for innovation. The second important Ramping practice was to attend to operational matters as the foundation for innovation. Week after week, the researchers observed project meetings in which senior leaders went into great operational detail regarding the implementation. Several of them told us they wished they did not have to engage in such detail but they knew the need to get these details right if the system was to provide the platform for innovation they desired. Although this was recognized in both waves, a senior clinician with a very strong reputation for attention to detail was brought in during the second wave to bolster this effort. By getting their hands dirty in the detail, the leaders could then talk credibly with colleagues from across the hospital who were nervous about the system and who did not trust outsiders to honestly tell them about its fitness. By getting into operational detail, the leaders were building the respect they needed to further improve participation across the hospital.

Managing not avoiding risk. The third important Ramping practice we observed was to manage rather than avoid risk. The project was being implemented at a time when past IT failures were in everyone's mind. There were great pressures to reduce risk by delaying and descope work, but project leaders also knew this would reduce the project's chance of ever being implemented because the public sector was so risk averse. Therefore, project leaders kept deadlines firm and managed risk rather than avoiding it (for example, giving clinicians the autonomy needed to act quickly and ensuring adequate support was on hand for them). This increased the project's credibility in everyone's eyes. As one of the most senior doctors stressed to all other doctors in a medical 'Grand Round' in the months before go-live, the project was simply going ahead and they all had to get on board to ensure its success. Moreover,

their success in managing risk during the first wave's go-live gave them confidence in their ability to do so in the second wave. They were effectively honing their risk management capabilities over each wave, which in turn improved alignment.

Improving Social Alignment through Holding

Holding practices maintain alignment in the face of pressures to separate. This is shown in Figure 2, where the second fall in the curve for Holding falls slower and shallower than the second fall in the Baseline curve. We observed three practices that facilitated Holding at the hospital. The common idea underlying all three is the transformation was now *owned* by the business and the business needed to partner with IT to "bed down" and improve the system. In short, these practices gave the motivation and the structures to stay together.

Adjusting governance structures. First, the organization adjusted its governance structures to transition from a project mentality to a business-as-usual mentality. This involved giving user groups and the hospital's executive more decision-rights over the system's assimilation, with IT having a strong partnering role. For instance, while the project team made decisions regarding training in the pre-go-live state, the hospital and its divisions took greater responsibility over training post go-live and the IT group shifted to a partnership role by working with user groups to have 'adoption coaches' available to help clinicians as needed across the hospital.

Reallocating funding and resources. Second, the organization reallocated funding and resources in a timely manner to maintain momentum. Because projects are temporary and often underfunded, funding often dries up after projects go live, even though much remains to be done. Such shortfalls can derail projects, leading key staff to leave and misalignment to grow. To avoid such problems, leaders reallocated funding shortly after the project go-live, for example, by identifying how the new system could allow the organization to remove a number of staff positions and how this financial saving could be used to create a new set of roles focused on optimizing the system.


^d The method used to plot our findings was inspired by studies of momentum, see Nelson and Jansen.¹⁰

Managing expectations and knowledge-sharing. Third, the organization managed users' expectations and knowledge-sharing. This was critical because the rush to get the system ready for go-live meant that changes to the system still needed to be made and more training and practice changes were required after go-live. Predictably, many change requests were raised. Some units wanted the system to revert to old practices while others wanted even-more advanced features. The IT group needed to manage expectations, but they had to do so in a way that allowed the business to feel it owned the transformation. They did so by mining usage logs to identify business units that were using the system in advanced ways as well as business units that required more support. They then encouraged leaders from these units to interact to share knowledge about strengths and weaknesses of the system. By doing this at regular intervals, the IT group was able to facilitate knowledge sharing while still ensuring the business groups owned the project.


Improving Social Alignment through Peaking

Peaking practices increase alignment to even higher levels. This is shown in Figure 2, where the second rise in the curve for Peaking rises higher than the corresponding rise in the Baseline curve. To achieve peaking required team members to be more able and motivated than previously. When we observed practices that facilitated Peaking, the common factor was that they *related to the base of the pyramid* in Figure 1—connection and respect. Connection gave teams a common identity and motivation and facilitated mutual learning that enabled members to perform to even higher levels, while respect enabled staff to put differences aside, trust each other, and focus on achieving the best solution possible.

Co-locating teams. One practice that facilitated Peaking was co-locating teams. Doing so gave team members the time and place to learn from one another and understand their differences. Keeping teams separate proved costly. For instance, at one stage, one team at an external site did not communicate the fact they were not meet-



Because projects are temporary and often underfunded, funding often dries up after projects go live, even though much remains to be done.



ing deadlines until their deadlines were well exceeded. In contrast, relationships with the vendor improved on the second go-live because they were co-located with the project team in the command center while in the first go-live they were given a separate room. We were told this improved relationships so much the vendor and client staff saw themselves as one while working together.

Getting the right people at the right time in the right roles. A second practice that facilitated Peaking was getting the right people at the right time in the right roles. On one hand, some people were kept on during the project. Their knowledge and networks naturally grew over time, making them increasingly able to facilitate strong alignment. Meanwhile, other key people were replaced with new team members and new roles evolved. For instance, as the project shifted from implementation to assimilation, a new team of Adoption Coaches was created and influential clinicians from across the hospital were recruited into the lead roles. This increased respect for the project across the organization because the team members had the right skills at the right times.

Understanding the 'why?' A third practice that facilitated Peaking was clarifying the answer to "why are we doing this?" Because so much attention to operational detail was required to get the system implemented, it was easy to lose sight of the end goal and become demotivated. This was especially in the down periods in the months immediately after go-live when staff were experiencing what one executive likened to post-natal depression. To overcome this, the organization worked hard to clarify the 'why', and do so in the language of clinicians, that is, focused on supporting improvements in patient care. An important positive that came out of this was that it allowed the team to reflect on the clinical benefits that emerge from a successful project. This success of the first wave buoyed them and motivated them to see additional benefits that could stem from the next wave. Greater motivation could then be reached on the second wave because the end goal was clearer and more compelling.

Integration: Best Practice and Ineffective Practice

Once the practices of Ramping, Holding, and Peaking, are understood, we can see how their combination, or lack thereof, can shape best practice and poor practice. As Figure 2 shows, best practice involves effective Ramping, Holding, and Peaking practices. Each dip becomes shorter, each rise becomes steeper, and each rise hits a new peak. Ineffective practice involves the opposite. Each dip is sharper, each rise is slower, and each new peak is lower. While the graphs in Figure 2 are stylized rather than based on quantitative data, we found evidence of both trajectories in the project we studied. At the lead hospital we studied, where all the practices described earlier were implemented, the trajectory was positive. The drop to misalignment was slower, the rise to social alignment was faster, and the level of social alignment was greater on the second go-live compared to the first.

We observed the opposite trajectory when these practices were enacted poorly or not at all. Our evidence for the opposite trajectory comes from a composite case we gathered data on during the study.^e While the team came together for the first go-live, it soon became clear they had not laid the foundation for that alignment (that is, the base of the pyramid)—their alignment was more appearance than substance. They had, in fact, struggled with each practice we discussed. Ramping suffered because senior and powerful clinicians were not included sufficiently in the project. Disgruntled clinicians then resisted the system soon after go-live and voiced their concerns to external media. Holding suffered because the organization failed to reallocate funding and manage user expectations. This led users to lose faith in the system fueling disrespect on all sides. Peaking suffered because the site had used a large number of external consultants rather than internal staff in key roles. While the consultants could

and did play an important role, the hospital simply did not have the right internal people in the right roles at the right times. Peaking became infeasible; the aim became salvaging any semblance of alignment at all. Of course, our conclusions here are bounded by the time of our study. In the period since we completed the study and wrote this article, our informal observations of the composite case have been that social alignment has recovered somewhat, with greater clinical involvement and realignment of key positions. While the process of recovery lay outside the scope of our study, it appears to conform to the framework in Figure 1, in that misalignment, even if severe, can return to alignment.^f

Conclusion


Social alignment can be viewed as the glue that holds projects together. It helps people work together, stick together, and strive for shared goals. This article provides practical insights for how to achieve social alignment in complex projects when there are many factors working against it, such as when there are strong and diverse professional groups, multiple organizations, high stakes, and long periods. By collecting detailed evidence as a transformation project unfolded, and by comparing and contrasting different cases, we identified three practices that improved social alignment—Ramping, Holding, and Peaking. These practices motivated stakeholders to align, stay aligned, and strengthen alignment over time. While some of the practices can be found in prior work, this article contributes by revealing how they can come together in a coordinated way to improve social alignment over time and the consequences of not doing so.^g While some of these practices might have been expected in advance, not all of them were. In particular, one

^f While we do not have formal data on how alignment improved at the composite case, we hypothesize that it is due to the implementation of some of the best practices noted earlier. We hope to verify this in future.

^g For instance, the case study of Cisco's ERP implementation provides vivid examples of what we would describe as peaking practices and holding practices, see Austin et al.¹

of the Holding practices (adjusting governance structures) proved particularly challenging and represents an area where more attention could be placed in future projects. We hope the results of this study will motivate future research to test and extend the insights we have offered.

Acknowledgments

This research was supported by the Australian Research Council (FT130100942, DP140101815, LP170101154), University of Queensland (UQFEL1719117), and Metro South Health. 

References

1. Austin, R.D., Nolan, R.L. and Cotteleer, M.J. Cisco Systems, Inc.: Implementing ERP. *Harvard Business School Case*, 2002, 1-19.
2. Burke, T.K. Providing ethics a space on the page: Social work and ethnography as a case in point. *Qualitative Social Work* 6, 2 (2007), 177-195.
3. Burton-Jones, A., Akhlaghpour, S., Ayre, S., Barde, P., Staib, A. and Sullivan, C. Changing the conversation on evaluating digital transformation in healthcare: Insights from an institutional analysis. *Information and Organization* 30, 1 (2020), 1-16.
4. Chesterman, R.N. Queensland Health Payroll System Commission of Inquiry. State of Queensland, 2013.
5. Gilchrist, A., Burton-Jones, A. and Green, P. The process of social alignment and misalignment within a complex IT project. *Intern. J. Project Management* 36, 6 (2018), 845-860.
6. Gilchrist, A., Burton-Jones, A., Green, P. and Smidt, M. The process of social alignment and misalignment within a complex IT project. In *Proceedings of the 38th Intern. Conf. Information Systems*. (Seoul, Korea, 2017).
7. Glaser, B.G. and Strauss, A.L. *The Discovery of Grounded Theory*. Aldine, Chicago, 1967.
8. Martin, S.A. and Sinsky, C.A. The map is not the territory: Medical records and 21st century practice. *Lancet* 388 (2016), 2053-2056.
9. McAfee, A. The impact of enterprise information technology adoption on operational performance: An empirical investigation. *Production and Operations Management* 11, (2002), 33-53.
10. Nelson, R.R., and Jansen, K.J. Mapping and managing momentum in IT projects. *MIS Q. Executive* 8, 3 (2009), 141-148.
11. Ranganathan, C., Watson-Manheim, M.B., and Keeler, J. Bringing professionals on board: Lessons on executing IT-enabled organizational transformation. *MIS Q. Executive* 3, 3 (2004), 151-160.
12. Reich, B.H., and Benbasat, I. Factors that influence the social dimension of alignment between business and information technology objectives. *MIS Q.* 21, 1 (2000), 81-113.
13. Van der Hoorn, B. and Whitty, S.J. The praxis of 'alignment seeking' in project work. *Intern. J. Project Management* 35, 6 (2017), 978-993.

Andrew Burton-Jones (a.burtonjones@business.uq.edu.au) is a professor of business information systems in the UQ Business School, The University of Queensland, AU.

Alicia Gilchrist (alicia.gilchrist@anu.edu.au) is a Post-doctoral Research Fellow at Australian National University, Canberra, AU.

Peter Green (p.green@qut.edu.au) is a professor of accounting at Queensland University of Technology, Queensland, AU.

Michael Draheim (Michael.Draheim@cerner.com) is an adjunct professor, University of Queensland, AU, and Chief Nursing and Midwifery Officer at Cerner Corporation, Australia and AsiaPac.

^e A composite case is a case that combines characteristics of multiple cases rather than one case. We use a composite case because the implementation program is ongoing and it is more constructive to learn general lessons than to single out individual cases; see Burke.²

In distributed systems theory, CALM presents a result that delineates the frontier of the possible.

BY JOSEPH M. HELLERSTEIN AND PETER ALVARO

Keeping CALM: When Distributed Consistency Is Easy

DISTRIBUTED SYSTEMS ARE tricky. Multiple unreliable machines are running in parallel, sending messages to each other across network links with arbitrary delays. How can we be confident these systems do what we want despite this chaos?

This issue should concern us because nearly all of the software we use today is part of a distributed system. Apps on your phone participate with hosted services in the cloud; together they form a distributed system. Hosted services themselves are massively distributed systems, often running on machines spread across the globe. Big data systems and large-scale databases are distributed

across many machines. Most scientific computing and machine learning systems work in parallel across multiple processors. Even legacy desktop operating systems and applications like spreadsheets and word processors are tightly integrated with distributed backend services.

The challenge of building correct distributed systems is increasingly urgent, but it is not new. One traditional answer has been to reduce this complexity with *memory consistency* guarantees—assurances that accesses to memory (heap variables, database keys, and so on) occur in a controlled fashion. However, the mechanisms used to enforce these guarantees—*coordination protocols*—are often criticized as barriers to high performance, scale, and availability of distributed systems.

The high cost of coordination. Coordination protocols enable autonomous, loosely coupled machines to jointly decide how to control basic behaviors, including the order of access to shared memory. These protocols are among the most clever and widely cited ideas in distributed computing. Some well-known techniques include the Paxos³³ and Two-Phase Commit (2PC)^{25,34} protocols, and global barriers underly-

» key insights

- **Coordination is often a limiting factor in system performance. While sometimes necessary for consistent outcomes, coordination often needlessly stands in the way of interactivity, scalability, and availability.**
- **Distributed systems deserve a computability theory: When is coordination required for consistency, and when can it be avoided?**
- **The CALM Theorem shows that monotonicity is the answer to this question. Monotonic problems have consistent, coordination-free implementations; non-monotonic problems require coordination for consistency.**
- **The CALM Theorem emerges by shifting the definition of consistency to one of deterministic program outcomes rather than ordered histories of events. CALM thinking is also constructive: it informs the design of new distributed programming languages, program analysis tools, and application design patterns.**



ing computational models like Bulk Synchronous Parallel computing.⁴⁰

Unfortunately, the expense of coordination protocols can make them “forbidden fruit” for programmers. James Hamilton from Amazon Web Services made this point forcefully, using the phrase “consistency mechanisms” where we use coordination:

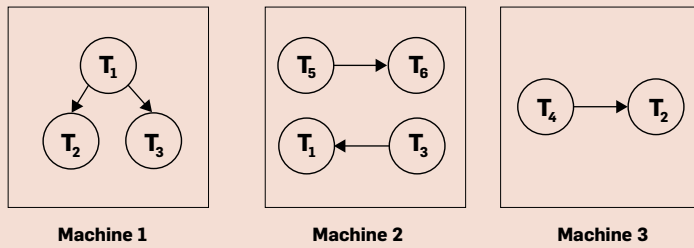
“The first principle of successful scalability is to batter the consistency

mechanisms down to a minimum, move them off the critical path, hide them in a rarely visited corner of the system, and then make it as hard as possible for application developers to get permission to use them.”²⁶

The issue is not that coordination is tricky to implement, though that is true. The main problem is that coordination can dramatically slow down computation or stop it altogether. Some modern

global-scale systems utilize coordination protocols; the Google Spanner transactional database¹⁸ is a notable example that uses both Paxos and 2PC. However, these protocols suffer from high latencies, on the order of 10ms–100ms. Global-scale systems that rely on these protocols are not meant to be used in the fast path of an application. Coordination latency problems translate to the micro scale as well. Recent

Figure 1. A distributed waits-for graph with replicated nodes and partitioned edges. There is a cycle that spans Machines 1 and 2 (T_1, T_3).



work showed that state-of-the-art multiprocessor key-value stores can spend 90% of their time waiting for coordination; a coordination-free implementation called Anna achieves over two orders of magnitude speedup by eliminating that coordination.⁴³ Can we avoid coordination more generally, as Hamilton recommends? When?

The bigger picture: Program consistency. The general question of when coordination is necessary to achieve consistency was not addressed until relatively recently. Traditional work on consistency focused on properties like linearizability³⁰ and conflict serializability,²⁰ which ensure memory consistency by constraining the order of conflicting memory accesses. This tradition obscured the underlying question of whether coordination is required for the consistency of a particular program’s outcomes. To attack the problem holistically we need to move up the stack, setting aside low-level details in favor of program semantics.

Traffic intersections provide a useful analogy from the real world. To avoid accidents at busy intersections, we often install stop lights to coordinate traffic across two intersecting roads. However, coordination is not a necessary evil in this scenario: we can also prevent accidents by building an overpass or tunnel for one of the roads. The “traffic intersection problem” is an example with a coordination-free solution. Importantly, the solution is not found by cleverly controlling the order of access to the critical section where the roads overlap on a map. The solution involves engineering the roads to avoid the need for coordination entirely.

For the traffic intersection problem, it turns out there is a solution that

avoided coordination altogether. Not all problems have such a solution. For any given computational problem, how do we know if it has a coordination free solution, or if it requires coordination for consistency? To sharpen our intuition, we consider two nearly identical problems from the distributed systems canon. Both involve graph reachability, but one is coordination free and the other is not.

Distributed deadlock detection. Distributed databases identify cycles in a distributed graph in order to detect and remediate deadlocks. In a traditional database system, a transaction T_i may be waiting for a lock held by another transaction T_j , which may in turn be waiting for a second lock held by T_i . The deadlock detector identifies such “waits-for” cycles by analyzing a directed graph in which nodes represent transactions, and edges represent one transaction waiting for another on a lock queue. Deadlock is a stable property: the transactions on a waits-for cycle cannot make progress, so all edges on the cycle persist indefinitely.

In a distributed database, a “local” (single-machine) view of the waits-for graph contains only a subset of the edges in the global waits-for graph. In this scenario, how do local deadlock detectors work together to identify global deadlocks?

Figure 1 shows a waits-for cycle that spans multiple machines. To identify such distributed deadlocks, each machine exchanges copies of its edges with other machines to accumulate more information about the global graph. Any time a machine observes a cycle in the information it has received so far, it can declare a deadlock among the transactions on that cycle.

We might be concerned about tran-

sient errors due to delayed or reordered messages in this distributed computation. Do local detectors have to coordinate with other machines to be sure of a deadlock they have observed? In this case, no coordination is required. To see this, note that once we know a cycle exists in a graph, learning about a new edge can never make the cycle go away. For example, once Machine 1 and Machine 2 jointly identify a deadlock between T_1 and T_3 , new information from Machine 3 will not change that fact. Additional facts can only result in additional cycles being detected: the output at each machine grows monotonically with the input. Finally, if all the edges are eventually shared across all machines, the machines will agree upon the outcome, which is based on the full graph.

Distributed garbage collection. Garbage collectors in distributed systems must identify unreachable objects in a distributed graph of memory references. Garbage collection works by identifying graph components that are disconnected from the “root” of a system runtime. The property of being “garbage” is also stable: once a graph component’s connection to the root is removed, the objects in that component will not be re-referenced.

In a distributed system, references to objects can span machines. A local view of the reference graph contains only a subset of the edges in the global graph. How can multiple local garbage collectors work together to identify objects that are truly unreachable?

Note that a machine may have a local object and no knowledge whether the object is connected to the root; Machine 3 and object O_4 in Figure 2 form an example. Yet there still may be a path to that object from the root that consists of edges distributed across other machines. Hence, each machine exchanges copies of edges with other machines to accumulate more information about the graph.

As before, we might be concerned about errors due to message delays or reordering. Can local collectors autonomously declare and deallocate garbage? Here, the answer is different: coordination is indeed required! To see this, note that a decision based on incomplete information—for example, Machine 3 deciding that object O_4 is unreachable in Figure

2—can be invalidated by the subsequent arrival of new information that demonstrates reachability (for example, the edges $\text{Root} \rightarrow O_1, O_1 \rightarrow O_3, O_3 \rightarrow O_4$). The output does not grow monotonically with the input: provisional “answers” may need to be retracted. To avoid this, a machine must ensure it has heard *everything there is to hear* before it declares an object unreachable. The only way to know it has heard everything is to coordinate with all the other machines—even machines that have no reference edges to report—to establish that fact. As we will discuss, a hallmark of coordination is this requirement to communicate even in the absence of data dependencies.

The crux of consistency: Monotonicity. These examples bring us back to our fundamental question, which applies to any concurrent computing framework.

QUESTION: *We say that a computational problem is coordination-free if there exists a distributed implementation (that is, a program solving the problem) that computes a consistent output without using coordination. What is the family of coordination-free problems, and what problems lie outside that family?*

There is a difference between an incidental use of coordination and an intrinsic need for coordination: the former is the result of an implementation choice; the latter is a property of a computational problem. Hence our Question is one of computability, like P vs. NP or Decidability. It asks what is (im)possible for a clever programmer to achieve.

Note that the question assumes some definition of “consistency.” Where traditional work focused narrowly on memory consistency (that is, reads and writes produce agreed-upon values), we want to focus on program consistency: does the implementation produce the *outcome* we expect (for example, deadlocks detected, garbage collected), despite any race conditions across messages and computation that might arise?

Our examples provide clues for answering our question. Both examples accumulate a *set* of directed edges E , and depend on reachability predicates—that is, tests for pairs of nodes in

the transitive closure E^* . But they differ in one key aspect. A node participates in a deadlock if there exists a path to itself in E^* : $\{n \mid \exists(n,n) \in E^*\}$. A node n is garbage if there does *not* exist a path from root to n : $\{n \mid \neg \exists(\text{root},n) \in E^*\}$.

Logical predicates clarify the distinction between the examples. For deadlock detection’s existential predicate, the set of satisfying paths that exist is *monotonic* in the information received:

DEFINITION 1. *A problem P is monotonic if for any input sets S, T where $S \subseteq T, P(S) \subseteq P(T)$.*

By contrast, the set of satisfying paths that do *not* exist in the garbage collection example is non-monotonic: conclusions made on partial information about E may not hold in eventuality as counterexamples appear to revoke prior beliefs about what “did not exist” previously.

Monotonicity is the key property underlying the need for coordination to establish consistency, as captured in the CALM Theorem:

THEOREM 1. *Consistency As Logical Monotonicity (CALM). A problem has a consistent, coordination-free distributed implementation if and only if it is monotonic.*

Intuitively, monotonic problems are “safe” in the face of missing information and can proceed without coordination. Non-monotonic problems, by contrast, must be concerned that truth of a property *could change in the face of new information*. Therefore, they cannot proceed until they know all information has arrived, requiring them to coordinate.

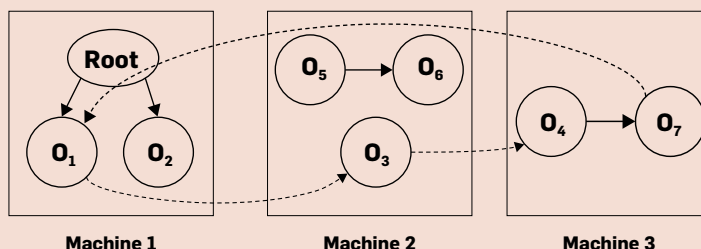
Additionally, because they “change their mind,” non-monotonic problems are order-sensitive: the order in which they receive information determines how they toggle state back and forth, which can in turn determine their final state (as we will see in the example of shopping carts). By contrast, monotonic problems simply accumulate beliefs; their output depends only on the content of their input, not the order in which it arrives.

Our discussion so far has remained at the level of intuition. The next section provides a sketch of a proof of the CALM Theorem, including further discussion of definitions for consistency and coordination. The proof uses logic formalisms from database theory and demonstrates the benefits of bringing the theory of databases (ACM PODS) and distributed systems (ACM PODC) closer together. Problems can be defined as families of declarative queries over relations (sets of records) running across multiple machines. As in our examples, the monotonicity of these queries can often be checked statically via their syntax: for example, $\exists(n,n) \in E^*$ is monotonic, but $\neg \exists(\text{root},n) \in E^*$ is non-monotonic, as evidenced by the use of the negated existential quantifier $\neg \exists$ (“not exists”). Readers seeking a complete proof are directed to the papers by Ameloot, et al.^{8,9}

CALM: A Proof Sketch

Our first challenge in formalizing the CALM Theorem is to define program consistency in a manner that allows us to reason about program outcomes, rather than mutations to storage. Having done that, we can move on

Figure 2. A distributed object reference graph with remote references (dotted arrows). The fact that object O_3 is reachable from Root can be established without any information from Machine 3. Objects O_5 and O_6 are garbage, which can only be established by knowing the entire graph.



to a discussion of consistent computability with and without coordination.

Program consistency: Confluence. Distributed systems introduce significant non-determinism to our programs. Sources of non-determinism include unsynchronized parallelism, unreliable components, and networks with unpredictable delays. As a result, a distributed program can exhibit a large space of possible behaviors on a given input.

While we may not control all the behavior of a distributed program, our true concern is with its *observable* behavior: the program outcomes. To this end, we want to assess how distributed nondeterminism affects program outcomes. A practical consistency question is this: “Does my program produce deterministic outcomes despite non-determinism in the runtime system?”

This is a question of program *confluence*. In the context of nondeterministic message delivery, an operation on a single machine is confluent if it produces the same set of output responses for any non-deterministic ordering and batching of a set of input requests. In this vein, a confluent single-machine operation can be viewed as a *deterministic function from sets to sets*, abstracting away the nondeterministic order in which its inputs happen to appear in a particular run of a distributed system. Confluent operations compose: if the output set of one confluent operation is consumed by another, the resulting composite operation is confluent. Hence, confluence can be applied to individual operations, components in a dataflow, or even entire distributed programs.² If we restrict ourselves to building programs by composing confluent operations, our programs are confluent by construction, despite orderings of messages or execution races within and across components.

Unlike traditional memory consistency properties such as linearizability,³⁰ confluence makes no requirements or promises regarding notions of recency (for example, a read is not guaranteed to return the result of the latest write request issued) or ordering of operations (for example, writes are not guaranteed to be applied in the same order at all replicas). Nevertheless, if an application is confluent, we know that any such anomalies at the

memory or storage level *do not affect the application outcomes*.

Confluence is a powerful yet permissive correctness criterion for distributed applications. It rules out application-level inconsistency due to races and non-deterministic delivery, while permitting nondeterministic ordering and timings of lower-level operations that may be costly (or sometimes impossible) to prevent in practice.

Confluent shopping carts. To illustrate the utility of reasoning about confluence, we consider an example of a higher-level application. In their paper on the Dynamo key-value store,¹⁹ researchers from Amazon describe a shopping cart application that achieves confluence without coordination. In their scenario, a client Web browser requests items to add and delete from an online shopping cart. For availability and performance, the state of the cart is tracked by a distributed set of server replicas, which may receive requests in different orders. In the Amazon implementation, no coordination is needed while shopping, yet all server replicas eventually agree on the same final state of the shopping cart. This is a prime example of the class of program that interests us: eventually consistent, even when implemented atop a non-deterministic distributed substrate that does no coordination.

Program consistency is possible in this case because the fundamental operations performed on the cart (for example, add) commute, so long as the contents of the cart are represented as a set and the internal ordering of its elements is ignored. If two replicas learn along the way they disagree about the contents of the cart, their differing views can be merged simply by issuing a logical “query” that returns the union of their respective sets.

Unfortunately, if we allow a delete operation in addition to add, the set neither monotonically grows nor shrinks, which causes consistency trouble. If instructions to add item *I* and delete item *I* arrive in different orders at different machines, the machines may disagree on whether *I* should be in the cart. As mentioned earlier, this is reflected in the way the existence of *I* toggles on the nodes. On one machine the presence of *I* might start in the state *not-exists*, but a se-

ries of messages `<add(I); delete(I)>` will cause the state to toggle to *exists* and then to *not-exists*; on another machine the messages might arrive in the order `<delete(I); add(I)>`, causing *I*'s state to transition from *not-exists* to *not-exists* to *exists*. Even after the two machines have each received all the messages, they disagree on the final outcome. As a traditional approach to avoid such “race conditions,” we might bracket every non-monotonic delete operation with a global coordination to agree on which add requests come before it. Can we do better?

As a creative application-level use of monotonicity, a common technique is for deletes to be handled separately from adds via two separate monotonically growing sets: Added items and Deleted items.^{19,39} The Added and Deleted sets are both insert-only, and insertions across the two commute. The final cart contents can be determined by unioning up the Added sets across nodes, as well as unioning up the Deleted sets across nodes, and computing the set-difference of the results. This would seem to solve our problem: it removes the need to coordinate while shopping—that is, while issuing add and delete requests to the cart.

Unfortunately, neither the add nor delete operation commutes with checkout—if a checkout message arrives before some insertions into either the Added or Deleted sets, those insertions will be lost. In a replicated setting like Dynamo's, the order of checkout with respect to other messages needs to be globally controlled, or it could lead to different decisions about what was actually in the cart when the checkout request was processed.

Even if we stop here, our lens provided a win: monotonicity allows *shopping* to be coordination free, even though *checkout* still requires coordination.

This design evolution illustrates the technical focus we seek to clarify. Rather than micro-optimize protocols like Paxos or 2PC to protect race conditions in procedural code, modern distributed systems creativity often involves minimizing the use of such protocols.

A sketch of the proof. The CALM conjecture was presented in a keynote

talk at PODS 2010 and written up shortly thereafter alongside a number of corollaries.²⁸ In a subsequent series of papers,^{8,9,44} Ameloot and colleagues presented a formalization and proof of the CALM Theorem, which remains the reference formalism at this time. Here, we briefly review the structure of the argument from Ameloot et al.

Proofs of distributed computability require some formal model of distributed computation: a notion of disparate machines each supporting some local model of computation, data partitioned across the machines, and an ability for the machines to communicate over time. To capture the notion of a distributed system composed out of monotonic (or non-monotonic) logic, Ameloot uses the formalism of a *relational transducer*¹ running on each machine in a network. This formalism matches our use of logical expressions in our graph examples; it also matches the design pattern of sets of items with additions, deletions and queries in Dynamo.

Simply put, a relational transducer is an event-driven server with a relational database as its memory and programs written declaratively as queries. Each transducer runs a sequential event loop as follows:

1. **Ingest and apply** an unordered batch of requests to insert and delete records in local relations. Requests may come from other machines or a distinguished input relation.

2. **Query** the (now-updated) local relations to compute batches of records that should be sent somewhere (possibly locally) for handling in future.

3. **Send** the results of the query phase to relevant machines in the network as requests to be handled. Results sent locally are ingested in the very next iteration of the event loop. Results can also be “sent” to a distinguished output.

In this computational model, the state at each machine is represented via sets of records (that is, relations), and messages are represented via records that are inserted into or deleted from the relations at the receiving machine. Computation at each machine is specified via declarative (logic) queries over the current local relations at each iteration of the event loop.

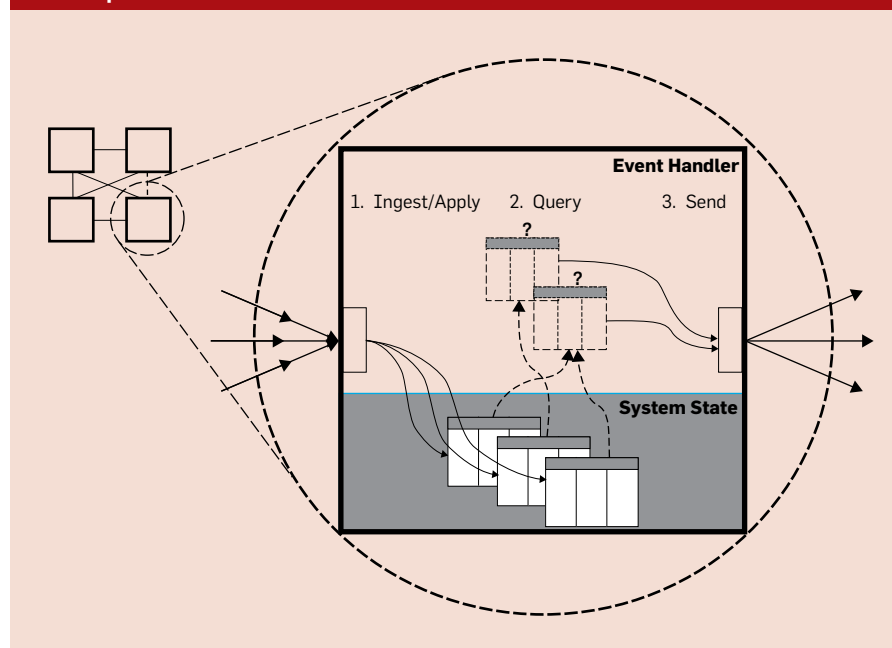
The next challenge is to define monotonicity carefully. The query

languages used by Ameloot are variants of Datalog, but we remind the reader that classical database query languages—relational calculus and algebra, SQL, Datalog—are all based on first-order logic. In all of these languages, including first-order logic, most common expressions are monotonic; the syntax reveals the potentially nonmonotonic expressions. Hence “programs expressed in monotonic logic” are easy to define and identify: they are the transducer networks in which every machine’s queries use only monotonic syntax. For instance, in the relational algebra, we can allow each machine to employ selection, projection, intersection, join and transitive closure (the monotonic operators of relational algebra), but not set difference (the sole non-monotonic operator). If we use relational logic, we disallow the use of universal quantifiers (\forall) and their negation-centric equivalent ($\neg \exists$)—precisely the construct that tripped us up in the garbage collection example noted earlier. If we model our programs with mutable relations, insertions are allowable, but in general updates and deletions are not.^{5,35} These informal descriptions elide a number of clever exceptions to these rules that still achieve semantic monotonicity despite syntactic non-monotonicity,^{8,17} but they give a sense of how the formalism is defined.

Now that we have a formal execution model (relational transducers), a definition of consistency (confluence), and a definition of monotonic programs, we are prepared to prove a version of the CALM Theorem. The forward “if” direction of the CALM Theorem is quite straightforward and similar to our previous discussion: in a monotonic relational transducer network, it is easy to show that any machine will eventually Ingest and Send a deterministic set of messages and generate a deterministic output. As a side benefit, at any time during execution, the messages output by any machine form a valid subset of the final output.

The reverse “only if” direction is quite a bit trickier, as it requires ruling out any possible scheme for avoiding coordination. The first challenge is to formally separate the communication needed to construct outputs (essentially, dataflow messages) from other communication (coordination messages). Intuitively, dataflow messages are those that arise to assemble data whose components are not co-located. To isolate the coordination messages, Ameloot et al. consider all possible ways to partition data across machines in the network at program start. From each of these starting points, a messaging pattern is produced during execution of the program. We say that a program contains coordination if it requires messages to be sent

Figure 3. A simple four-machine relational transducer network with one machine’s state and event loop shown in detail.



under *all possible partitionings*—including partitionings that co-locate all data at a single machine. A message that is sent in every partitioning is not related to dataflow; it is a coordination message. As an example, consider how a distributed garbage collector decides if a locally disconnected object O_g is garbage. Even if all the data is placed at a single machine, that machine needs to exchange messages with the other machines to check that they have no more additional edges—it needs to “coordinate,” not just communicate data dependencies. The proof then proceeds to show that non-monotonic operations require this kind of coordination.

This brief description elides many interesting aspects of the original article. In addition to the connections established between monotonicity and coordination-freeness, connections are also made to other key distributed systems properties. One classic challenge is to achieve distributed agreement on network membership (represented by Ameloot et al. as the *All* relation). It turns out that not only are the monotonic problems precisely the coordination-free problems, they are also precisely those that do not require knowledge of network membership—they need not query *All*. A similar connection is shown with the property of a machine being aware of its own identity/address (querying the *Id* relation).

CALM Perspective on the State of the Art

The CALM Theorem describes what is and is not possible. But can we use it practically? In this section, we address the implications of CALM with respect to the state of the art in distributed systems practice. It turns out that many patterns for maintaining consistency follow directly from the theorem.

CAP and CALM: Going positive. Brewer’s CAP Theorem¹⁴ informally states that a system can exhibit only two out of the three following properties: Consistency, Availability, and Partition-tolerance. CAP is a negative result: it captures properties that cannot be achieved in general. But CAP only holds if we assume the system in question is required to execute arbitrary programs. It does not ask whether there are specific subclasses of programs that can enjoy all three properties! In a retro-

spective, Brewer reframes his discussion of CAP along these very lines:

[The original] “expression of CAP served its purpose, which was to open the minds of designers to a wider range of systems and trade-offs ... The modern CAP goal should be to maximize combinations of consistency and availability that make sense for the specific application.”¹⁴

CALM is a positive result in this arena: it circumscribes the class of problems for which all three of the CAP properties can indeed be achieved simultaneously. To see this, note the following:

OBSERVATION 1. *Coordination-freeness is equivalent to availability under partition.*

In the forward direction, a coordination-free program is by definition available under partition: all machines can proceed independently. When and if the partition heals, state merger is monotonic and consistent. In the reverse direction, a program that employs coordination will stall (become unavailable) during coordination protocols if the machines involved in the coordination span the partition.

In that frame, CALM asks and answers the underlying question of CAP: “Which problems can be consistently computed while remaining available under partition?” CALM does not contradict CAP. Instead, CALM approaches distributed consistency from a wider frame of reference:

1. First, CAP is a negative result over the space of *all problems*: CALM confirms this coarse result, but delineates at a finer grain the negative and positive cases. Using confluence as the definition of consistency, CALM shows that monotone problems can in fact satisfy all three of the CAP properties at once; non-monotone problems are the ones that cannot.

2. The key insight in CALM is to focus on consistency from the viewpoint of *program outcomes* rather than the traditional *ordered histories* of conflicting actions—typically storage mutation. The emphasis on the problem being computed shifts focus from imperative implementation to declarative specification of outputs; that allows us to ask questions about what computations are possible.

The latter point is what motivated our outcome-oriented definition of program consistency. Note that Gilbert and Lynch²³ choose to prove the CAP Theorem using a rubric of linearizability (that is, agreement on a total order of conflicting actions), while Ameloot’s CALM Theorem proofs choose confluence (agreement on program outcomes.) We note that confluence is both more permissive and closer to user-observable properties. CALM provides the formal framework for the widespread intuition that we can indeed “work around CAP”—for monotone problems—even if we violate traditional systems-level notions of storage consistency.

Distributed design patterns. Our shift of focus from mutable storage to program semantics has implications beyond proofs. It also informs the design of better programming paradigms for distributed computing.

Traditional programming languages model the world as a collection of named variables whose values change over time. Bare assignment¹⁰ is a non-monotonic programming construct: outputs based on a prefix of assignments may have to be retracted when new assignments come in. Similarly, assignments make final program states dependent upon the arrival order of inputs. This makes it extremely hard to take advantage of the CALM Theorem to analyze systems written in traditional imperative languages!

Functional programming has long promoted the use of *immutable* variables, which are constrained to take on only a single value during a computation. Viewed through the lens of CALM, an immutable variable is a simple monotonic pattern: it transitions from being undefined to holding its final value, and never goes back. Immutable variables generalize to immutable data structures; techniques such as deforestation⁴¹ make programming with immutable trees, lists and graphs more practical.

Monotonic programming patterns are common in the design of distributed storage systems. We already discussed the Amazon shopping cart for Dynamo, which models cart state as two growing sets. A related pattern in storage systems is the use of *tombstones*: special data values that mark a data item as deleted. Instead of explicitly allowing deletion (a non-monoton-

ic construct), tombstones mask immutable values with corresponding immutable tombstone values. Taken together, a data item with tombstone monotonically transitions from undefined, to a defined value, and ultimately to tombstoned.


Conflict-free replicated data types (CRDTs)³⁹ provide an object-oriented framework for monotonic programming patterns like tombstones, typically for use in the context of replicated state. A CRDT is an abstract data type whose possible internal states form a lattice and evolve monotonically according to the lattice's associated partial order, such as the partial order of set containment under \subseteq or of integers under \leq . Two instances of a CRDT can be merged using the commutative, associative, idempotent join function from the associated internal lattice. Eventually, the states of two CRDT replicas that may have seen different inputs and orders can always be deterministically merged into a new final state that incorporates all the inputs seen by both.

CRDTs are an object-oriented lens on a long tradition of prior work that exploits commutativity to achieve determinism under concurrency. This goes back at least to long-running transactions,^{15,22} continuing through recent work on the Linux kernel.¹⁶ A problem with CRDTs is that their guarantees apply only to individual objects. The benefits of commutativity have been extended to composable libraries and languages, enabling programmers to reason about correctness of whole programs in languages like Bloom,³ the LVish library for Haskell,³² Lasp,³⁷ and Gallifrey.³⁸ We turn to an example of that idea next.


The Bloom programming language.

One way to encourage good distributed design patterns is to use a language specifically centered around those patterns. Bloom is a programming language we designed in that vein; indeed, the CALM conjecture and Bloom language were developed together.³

The main goal of Bloom is to make distributed systems easier to reason about and program. We felt that a good language for a domain is one that obscures irrelevant details and brings into sharp focus those that matter. Given that data consistency is a core chal-



The issue is not that coordination is tricky to implement, though that is true. The main problem is that coordination can dramatically slow down computation or stop it altogether.



lenge in distributed computing, we designed Bloom to be *data-centric*: both system state and events are represented as named data, and computation is expressed as queries over that data. The programming model of Bloom closely resembles that of the relational transducers described previously. This is no coincidence: both Bloom and Ameloot's transducer work are based on a logic language for distributed systems we designed called Dedalus.⁵ From the programmer's perspective, Bloom resembles event-driven or actor-oriented programming—Bloom programs use reorderable query-like handler statements to describe how an agent responds to messages (represented as data) by reading and modifying local state and by sending messages.

Because Bloom programs are written in a relational-style query language, monotonicity is easy to spot just as it was in relational transducers. The relatively uncommon non-monotonic relational operations—for example, set difference—stand out in the language's syntax. In addition, Bloom's type system includes CRDT-like lattices that provide object-level commutativity, associativity and idempotence, which can be composed into larger monotonic structures.¹⁷

The advantages of the Bloom design are twofold. First, Bloom makes set-oriented, monotonic (and hence confluent) programming the *easiest constructs for programmers to work with in the language*. Contrast this with imperative languages, in which assignment and explicit sequencing of instructions—two non-monotone constructs—are the most natural and familiar building blocks for programs. Second, Bloom can leverage simple forms of static analysis—syntactic checks for non-monotonicity and dataflow analysis for the taint of nonmonotonicity—to certify when programs provide the eventual consistency properties desired for CRDTs, as well as confirming when those properties are preserved across *compositions* of modules. This is the power of a language-based approach to monotonic programming: local, state-centric guarantees can be verified and automatically composed into global, outcome-oriented, program-level guarantees.


With Bloom as a base, we have developed tools including declarative testing frameworks,⁴ verification tools,⁶ and program transformation libraries that add coordination to programs that cannot be statically proven to be confluent.²

Coordination in its place. Pragmatically, it can sometimes be difficult to find a monotonic implementation of a full-featured application. Instead, a good strategy is to keep coordination off the critical path. In the shopping cart example, coordination was limited to checkout, when user performance expectations are lower. In the garbage collection example (assuming adequate resources) the non-monotonic task can run in the background without affecting users.


It can take creativity to move coordination off the critical path and into a background task. The most telling example is the use of tombstoning for low-latency deletion. In practice, memory for tombstoned items must be reclaimed, so eventually all machines need to agree to delete certain tombstoned items. Like garbage collection, this distributed deletion can be coordinated lazily in the background on a rolling basis. In this case, monotonic design does not stamp out coordination entirely, it moves it off the critical path.

Another non-obvious use of CALM analysis is to identify when to *compensate* (“apologize”²⁷) for inconsistency, rather than prevent it via coordination. For example, when a retail site allows you to purchase an item, it should decrement the count of items in inventory. This non-monotonic action suggests that coordination is required, for example, to ensure that the supply is not depleted before an item is allocated to you. In practice, this requires too much integration between systems for inventory, supply chain, and shopping. In the absence of such coordination, your purchase may fail non-deterministically after checkout. To account for this possibility, additional compensation code must be written to detect the out-of-stock exception and handle it by—for example—sending you an apologetic email with a loyalty coupon. Note that a coupon is not a clear mathematical inverse of any action in the original program; domain-aware compensation often goes beyond typical type system logic.

In short, we do not advocate pure



Our question is one of computability ... it asks what is (im)possible for a clever programmer to achieve.



monotonic programming as the only way to build efficient distributed systems. Monotonicity also has utility as an analysis framework for identifying nondeterminism so that programmers can address it creatively.

Additional Results

Many questions remain open in understanding the implications of the CALM Theorem on both theory and practice; we overview these in a longer version of this article.²⁹ The deeper questions include whether all PTIME is practically computable without coordination, and whether monotonicity in the CALM sense maps to stochastic guarantees for machine learning and scientific computation.

The PODS keynote talk that introduced the CALM conjecture included a number of related conjectures regarding coordination, consistency and declarative semantics.²⁸ Following the CALM Theorem result,⁹ the database theory community continued to explore these relationships, as summarized by Ameloot.⁷ For example, in the batch processing domain, Koutris and Suciu,³¹ and Beame et al.¹² examine massively parallel computations with rounds of global coordination, considering not only the number of coordination rounds needed for different algorithms, but also communication costs and skew.

In a different direction, a number of papers discuss tolerating memory inconsistency while maintaining program invariants. Bailis et al. define a notion of Invariant Confluence^{11,42} for replicated transactional databases, given a set of database invariants. Many of the invariants they propose are monotonic in flavor and echo intuition from CALM. Gotsman et al.²⁴ present program analyses that identify which pairs of potentially concurrent operations must be synchronized to avoid invariant violations. Li et al. define RedBlue Consistency,³⁶ requiring that users “color” operations based on their ordering requirements; given a coloring they choose a synchronization regime that satisfies the requirements.

Blazes² similarly elicits programmer-provided labels to more efficiently avoid coordination, but with the goal of guaranteeing full program consistency as in CALM.

Conclusion

Distributed systems theory is dominated by fearsome negative results, such as the Fischer/Lynch/Patterson impossibility proof,²¹ the CAP Theorem,²³ and the two generals problem.²⁵ These results identify things that are not possible to achieve in general in a distributed system. System builders, of course, are more interested in the complement of this space—those things that *can* be achieved, and, importantly, how they can be achieved while minimizing complexity and cost.

The CALM Theorem presents a positive result that delineates the frontier of the possible. CALM proves that if a problem is monotonic, it has a coordination-free program that guarantees consistency—a property of all possible executions of that program. The inverse is also true: any program for a non-monotonic problem will require runtime enforcement (coordination) to ensure consistent outcomes. CALM enables reasoning via static analysis, and limits or eliminates the use of runtime consistency checks. This is in contrast to storage consistency like linearizability or serializability, which requires expensive runtime enforcement.

CALM falls short of being a constructive result—it does not actually tell us how to write consistent, coordination-free distributed systems. Even armed with the CALM Theorem, a system builder must answer two key questions. First, and most difficult, is whether the problem they are trying to solve has a monotonic specification. Most programmers begin with pseudocode of some implementation in mind, and the theory behind CALM would appear to provide no guidance on how to extract a monotone specification from a candidate implementation. The second question is equally important: given a monotonic specification for a problem, how can I implement it in practice? Languages such as Bloom point the way to new paradigms for programming distributed systems that favor and (conservatively) test for monotonic specification. There is remaining work to do making these languages attractive to developers and efficient at runtime.

Acknowledgments. Thanks to Jeffrey Chase, our reviewers, as well as

Eric Brewer, Jose Faleiro, Pat Helland, Frank Neven, Chris Ré, and Jan Van den Bussche for their feedback and encouragement. C

References

1. Abiteboul, S., Vianu, V., Fordham, B. and Yesha, Y. Relational transducers for electronic commerce. *J. Computer and System Sciences* 61, 2 (2000), 236–269.
2. Alvaro, P., Conway, N., Hellerstein, J. and Maier, D. Blazes: Coordination analysis for distributed programs. In *Proceedings of the IEEE 30th Intern. Conf. on Data Engineering*, 2014, 52–63.
3. Alvaro, P., Conway, N., Hellerstein, J. and Marczak, W. Consistency analysis in Bloom: A CALM and collected approach. In *Proceedings of the 5th Biennial Conf. Innovative Data Systems Research (Asilomar, CA, USA, Jan. 9–12, 2011)* 249–260.
4. Alvaro, P., Hutchinson, A., Conway, N., Marczak, W. and Hellerstein, J. BloomUnit: Declarative testing for distributed programs. In *Proceedings of the 5th Intern. Workshop on Testing Database Systems*. ACM, 2012., 1.
5. Alvaro, P., Marczak, W., Conway, N., Hellerstein, J., Maier, D. and Sears, R. Dedalus: Datalog in time and space. *Datalog Reloaded*. Springer, 2011, 262–281.
6. Alvaro, P., Rosen, J. and Hellerstein, J. Lineage-driven fault injection. In *Proceedings of the 2015 ACM SIGMOD Intern. Conf. Management of Data*. ACM, 2015, 331–346.
7. Ameloot, T. Declarative networking: Recent theoretical work on coordination, correctness, and declarative semantics. *ACM SIGMOD Record* 43, 2 (2014), 5–16.
8. Ameloot, T., Ketsman, B., Neven, F. and Zinn, D. Weaker forms of monotonicity for declarative networking: A more fine-grained answer to the CALM-conjecture. *ACM Trans. Database Systems* 40, 4 (2016), 21.
9. Ameloot, T., Neven, F. and den Bussche, J.V. Relational transducers for declarative networking. *J. ACM* 60, 2 (2013), 15.
10. Backus, J. Can programming be liberated from the Von Neumann style? A functional style and its algebra of programs. *Commun. ACM* 21, 8 (Aug. 1978).
11. Bailis, P., Fekete, A., Franklin, M., Ghodsi, A., Hellerstein, J. and Stoica, I. Coordination avoidance in database systems. In *Proceedings of the VLDB Endowment* 8, 3 (2014), 185–196.
12. Beame, P., Koutris, P. and Suciu, D. Communication steps for parallel query processing. In *Proceedings of the 32nd ACM SIGMOD-SIGACT-SIGAI Symp. Principles of Database Systems*. ACM, 2013, 273–284.
13. Birman, K., Chockler, G. and van Renesse, R. Toward a cloud computing research agenda. *SIGACT News* 40, 2 (2009).
14. Brewer, E. CAP twelve years later: How the “rules” have changed. *Computer* 45, 2 (2012), 23–29.
15. Chrysanthis, P.K. and Ramamritham, K. Acta: A framework for specifying and reasoning about transaction structure and behavior. *ACM SIGMOD Record* 19, 2 (1990), 194–203.
16. Clements, A.T., Kaashoek, M.F., Zeldovich, N., Morris, R.T., and Kohler, E. The scalable commutativity rule: Designing scalable software for multicore processors. *ACM Trans. Computer Systems* 32, 4 (2015), 10.
17. Conway, N., Marczak, W., Alvaro, P., Hellerstein, J. and Maier, D. Logic and lattices for distributed programming. In *Proceedings of the 3rd ACM Symp. Cloud Computing*. ACM, 2012, 1.
18. Corbett, J. et al. Spanner: Google’s globally distributed database. *ACM Trans. Computer Systems* 31, 3 (2013), 8.
19. DeCandia, G. et al. Dynamo: Amazon’s highly available key-value store. *ACM SIGOPS Operating Systems Rev.* 41, 6 (2007), 205–220.
20. Eswaran, K., Gray, J., Lorie, R. and Traiger, I. The notions of consistency and predicate locks in a database system. *Commun. ACM* 19, 11 (1976), 624–633.
21. Fischer, M., Lynch, N. and Paterson, M. Impossibility of distributed consensus with one faulty process. *J. ACM* 32, 2 (1985), 374–382.
22. Garcia-Molina, H. and Salem, K. Sagas. In *Proceedings of the 1987 ACM SIGMOD Intern. Conf. Management of Data*. ACM, 249–259.
23. Gilbert, S. and Lynch, N. Brewer’s conjecture and the feasibility of consistent, available, partition-tolerant web services. *ACM SIGACT News* 33, 2 (2002), 51–59.
24. Gotsman, A., Yang, H., Ferreira, C., Najafzadeh, M. and Shapiro, M. ‘Cause I’m strong enough: Reasoning about consistency choices in distributed systems. *ACM SIGPLAN Notices* 51, 1 (2016), 371–384.
25. Gray, J. Notes on data base operating systems.

Operating Systems. Springer, 1978, 393–481.

26. Hamilton, J. Keynote talk. The 3rd ACM SIGOPS Workshop on Large-Scale Distributed Systems and Middleware. ACM, 2009.
27. Helland, P. and Campbell, D. Building on quicksand. In *Proceedings of the Conference on Innovative Data Systems Research*. ACM, 2009.
28. Hellerstein, J. The Declarative Imperative: Experiences and conjectures in distributed logic. *SIGMOD Record* 39, 1 (2010), 5–19.
29. Hellerstein, J. and Alvaro, P. Keeping CALM: When distributed consistency is easy. 2019; arXiv:1901.01930.
30. Herlihy, M. and Wing, J. Linearizability: A correctness condition for concurrent objects. *ACM Trans. Programming Languages and Systems* 12, 3 (1990), 463–492.
31. Koutris, P. and Suciu, D. Parallel evaluation of conjunctive queries. In *Proceedings of the 30th ACM SIGMOD-SIGACT-SIGART Symp. Principles of Database Systems*. ACM, 2011, 223–234.
32. Kuper, L. and Newton, R. LVARS: Lattice-based data structures for deterministic parallelism. In *Proceedings of the 2nd ACM SIGPLAN Workshop on Functional High-Performance Computing*. ACM, 2013, 71–84.
33. Lamport, L. The part-time parliament. *ACM Trans. Computer Systems* 16, 2 (1998), 133–169.
34. Lamson, B. and Sturgis, H. Crash recovery in a distributed system. Technical report, Xerox PARC Research Report, 1976.
35. Lausen, G., Ludäscher, B. and May, W. On active deductive databases: The state log approach. In *Workshop on (Trans) Actions and Change in Logic Programming and Deductive Databases*. Springer, 1997, 69–106.
36. Li, C., Porto, D., Clement, A., Gehrke, J., Prego, N. and Rodrigues, R. Making geo-replicated systems fast as possible, consistent when necessary. *OSDI 12* (2012), 265–278.
37. Meiklejohn, C. and Van Roy, P. Lasp: A language for distributed, coordination-free programming. In *Proceedings of the 17th Intern. Symp. Principles and Practice of Declarative Programming*. ACM, 2015, 184–195.
38. Milano, M., Recto, R., Magrino, T. and Myers, A. A tour of Gallifrey, a language for geo-distributed programming. In *Proceedings of the 3rd Summit on Advances in Programming Languages*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2019.
39. Shapiro, M., Prego, N., Baquero, C. and Zawirski, M. Conflict-free replicated data types. In *Proceedings of the Symp. Self-Stabilizing Systems*. Springer, 2011, 386–400.
40. Valiant, L. A bridging model for parallel computation. *Commun. ACM* 33, 8 (Aug. 1990), 103–111.
41. Wadler, P. Deforestation: Transforming programs to eliminate trees. In *Proceedings of the 2nd European Symp. Programming*, 1988.
42. Whittaker, M. and Hellerstein, J. Interactive checks for coordination avoidance. In *Proceedings of the VLDB Endowment* 12, 1 (2018), 14–27.
43. Wu, C., Faleiro, J., Lin, Y. and Hellerstein, J. Anna: A KVS for any scale. In *Proceedings of the 34th IEEE Intern. Conf. on Data Engineering*, 2018.
44. Zinn, D., Green, T. and Ludäscher, B. Win-move is coordination-free (sometimes). In *Proceedings of the 15th Intern. Conf. Database Theory*. ACM, 2012, 99–113.

Joseph M. Hellerstein (hellerstein@berkeley.edu) is the Jim Gray Professor of Computer Science at the University of California at Berkeley, CA, USA.

Peter Alvaro (palvaro@cs.ucsc.edu) is an assistant professor at the University of California at Santa Cruz, CA, USA.

© 2020 ACM 0001-0782/20/9 \$15.00



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

Attention: Undergraduate and Graduate Computing Students

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



Association for Computing Machinery
Advancing Computing as a Science & Profession

SPONSORED BY  Microsoft

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



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

David Mueller
North Carolina State University | SIGDOC 2018



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

Manuel Rigger
Johannes Kepler University Linz, Austria | Programming 2018



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

Matthias Springer
Tokyo Institute of Technology | SPLASH 2018



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

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



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

Elizaveta Tremsina
UC Berkeley | TAPIA 2018



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

Justin Lubin
University of Chicago | SPLASH 2018



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

Gemma Catolino
University of Salerno | MobileSoft 2018

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

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

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



research highlights

P. 84

**Technical
Perspective**
**Computing the Value
of Location Data**

By Cyrus Shahabi

P. 85

Computing Value of Spatiotemporal Information

By Heba Aly, John Krumm, Gireeja Ranade, and Eric Horvitz

P. 93

**Technical
Perspective**
**Progress in Spatial
Computing
for Flood Prediction**

By Shashi Shekhar

P. 94

Flood-Risk Analysis on Terrains

By Aaron Lowe, Pankaj K. Agarwal, and Mathias Rav

Technical Perspective

Computing the Value of Location Data

By Cyrus Shahabi

THE MOBILE COMPUTING landscape is witnessing an unprecedented number of devices that can acquire geo-tagged (aka location-based) data, for example, mobile phones, wearable sensors, in-vehicle dashcams, and IoT sensors. These devices can collect large amounts of data such as images, videos, movement parameters, or environmental measurements along with the data collectors' location data. However, we are giving away our location data to large Web search companies and social media companies for free. In addition, some of our smartphone apps gather our location data to sell to other companies for targeted advertising.

This data may be useful to third-party entities interested in gathering information from a certain location. For example, journalists may want to gather images around an event of interest for their newspaper; law enforcement may seek images taken soon before or after a crime occurred; and city authorities may be interested in travel patterns during heavy traffic.

An emerging trend is therefore to create data marketplaces where *owners* advertise their geo-tagged data objects to potential *buyers*, dubbed *geo-marketplaces*.¹ The following paper describes a technique for computing the monetary value of a person's location data for a potential geo-marketplace. The authors postulate that buyers pay people for their location data, perhaps through a geo-marketplace, and it shows how to compute the monetary value of such locations (represented as GPS points).


The paper applies established decision theory and "value of information" (VOI) techniques to determine the worth of a GPS point, illustrated with three scenarios. The first scenario is for a buyer that wants to deliver advertisements to people who live in a certain geographic region. The buyer uses GPS points to gradually narrow down the location of each person's home, ideally gaining confidence in the location

An emerging trend is to create data marketplaces where owners advertise their geo-tagged data objects to buyers, dubbed geo-marketplaces.

with each new point. The buyer must decide, for each user, which points to buy (if any) and make the ad delivery decision (that is, decide to deliver the ad to that user or not). The dilemma for the buyer is that it cannot see the coordinates of the available GPS points until it actually pays for them, so it must make its purchasing decisions based only on other available data about the points, such as their timestamps. The paper's analysis gives a principled approach for the buyer to compute the expected value of unseen GPS data, leading to purchasing decisions that are demonstrably better than more simplistic approaches. By estimating the VOI of each point, the buyer can place a numerical value on each point giving its worth. The paper gives a second example scenario about using GPS data to estimate traffic speeds on roads, and a third scenario about predicting a person's future location. In all three scenarios, buying points with high VOI leads to more efficient use of the data resources by identifying the most valuable points for the task.

The authors introduce a general framework that can be adopted by any buyer or geo-marketplace requiring GPS data from users. The general framework allows the buyer to estimate the VOI of the offered GPS points, and

it aids in making its purchasing and business decisions. The introduction of these principles means both buyers and sellers can set fair prices for GPS data, which today is still largely given away for no cost. Establishing these prices is a first step toward building a geo-marketplace, where sellers can be fairly compensated, and buyers can determine which data is most valuable for their application.

Geo-marketplaces, however, raise unique concerns. Publishing geo-tags reveal owners' whereabouts, which may lead to serious privacy breaches such as leakage of one's health status or political orientation. In addition, one must also protect the interests of buyers, and ensure they receive data objects satisfying their spatial requirements. Owners must be held accountable for their advertised data and not be able to change the geo-tag of an object after its initial advertisement. This can prevent situations where owners change geo-tags to reflect ongoing trends in buyers' interest. For example, when a certain high-profile event occurs at a location, dishonest owners may attempt to change their geo-tags closer to that location in order to sell their images at higher prices. Furthermore, the system must provide strong disincentives to prevent spam behavior, where dishonest participants flood the system with fake advertisements. These are possible future work for geo-marketplaces, some of which we discussed in a recent paper.¹ 

Reference

1. Nguyen, K., Ghinita, G., Naveed, M., and Shahabi, C. A privacy-preserving, accountable and spam-resilient geo-Marketplace. In *Proceedings of the 27th ACM SIGSPATIAL Intern. Conf. in Geographic Information Systems* (Chicago, IL, USA, Nov. 5–8, 2019).

Cyrus Shahabi is the Helen N. and Emmett H. Jones Professor of computer science, electrical engineering, and spatial sciences at the University of Southern California, Los Angeles, CA, USA. He is also chair of the CS department, director of the Integrated Media Systems Center and director of InfoLAB.

Copyright held by author.

Computing Value of Spatiotemporal Information

By Heba Aly, John Krumm, Gireeja Ranade, and Eric Horvitz

Abstract

Location data from mobile devices is a sensitive yet valuable commodity for location-based services and advertising. We investigate the intrinsic value of location data in the context of strong privacy, where location information is only available from end users via purchase. We present an algorithm to compute the expected value of location data from a user, without access to the specific coordinates of the location data point. We use decision-theoretic techniques to provide a principled way for a potential buyer to make purchasing decisions about private user location data. We illustrate our approach in three scenarios: the delivery of targeted ads specific to a user's home location, the estimation of traffic speed, and the prediction of location. In all three cases, the methodology leads to quantifiably better purchasing decisions than competing approaches.

1. INTRODUCTION

As people carry and interact with their connected devices, they create spatiotemporal data that can be harnessed by them and others to generate a variety of insights. Proposals have been made for creating markets for personal data¹ rather than for people either to provide their behavioral data freely or to refuse sharing. Some of these proposals are specific to location data.⁶ Several studies have explored the price that people would seek for sharing their GPS data.^{5,13,9} However, little has been published on determining the value of location data from a buyer's point of view. For instance, a Wall Street Journal blog says¹⁰:

“What groceries you buy, what Facebook posts you ‘like’ and how you use GPS in your car:

Companies are building their entire businesses around the collection and sale of such data. The problem is that no one really knows what all that information is worth. Data isn't a physical asset like a factory or cash, and there aren't any official guidelines for assessing its value.”

We present a principled method for computing the value of spatiotemporal data from the perspective of a buyer. Knowledge of this value could guide pursuit of the most informative data and would provide insights about potential markets for location data.

We consider situations where a buyer is presented with a set of location data points for sale, and we provide estimates of the value of information (VOI) for these points. Because the coordinates of the location data points are unknown, we compute the VOI based on the prior knowledge that is available to the buyer and on side information that a user may provide (e.g., the time of day or location granularity). The VOI computation is customized to the specific goals of

the buyer, such as targeting ad delivery for home services, offering efficient driving routes, or predicting a person's location in advance. We account for the fact that location data and user state are both uncertain. Additional data purchases can help reduce this uncertainty, and we quantify this reduction as well.

In the next section, we introduce a decision-making framework with a detailed analysis of geo-targeted advertising. We focus on the buyer's goal of delivering ads to people living within a certain region. We show that our method performs better than alternate approaches in terms of inferential accuracy, data efficiency, and cost. In Section 3, we apply the methodology to a traffic estimation scenario using real and simulated spatiotemporal data. We present our last scenario in Section 4, where we show how to make good data-buying decisions for predicting a person's future location.

Our contributions are as follows:

- We present a methodology to calculate the expected monetary value of a user's location coordinates, even when the detailed coordinates are unknown to the buyer a priori.
- We provide an algorithm for a buyer to make purchasing decisions about location data that may be sold by owners of the data, despite the specific location uncertainty.
- We demonstrate how the algorithm behaves in three scenarios: targeted ad delivery, crowdsourced traffic information, and location prediction.

To the best of our knowledge, this is the first principled method to compute the value of unseen crowdsourced location data from a buyer's point of view.

2. SCENARIO 1: HOME TARGETED ADS

Our first illustrative scenario is called “Home Targeted Ads” because it focuses on a business that wants to deliver ads to people who have homes within a certain geospatial region. For instance, a local roofing business may be licensed only in a certain geographic area and wish their ads to only be delivered to people who live in that area. A mobile dog grooming service may want to limit its advertising to a region that they

The original versions of this paper were “On the Value of Spatiotemporal Information: Principles and Scenarios” published in the *Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, (Nov. 2018) and “To Buy or Not to Buy: Computing Value of Spatiotemporal Information” published in *ACM Transactions on Spatial Algorithms and Systems* 12 (Sept. 2019).

can reach efficiently. We will refer to this target region as \mathcal{R} . The region can be any closed region on the ground, such as the boundary around a particular area.

The buyer in this case could be the business itself or an advertising specialist who can find the best recipients for the ads. In either case, the buyer seeks to find the home locations of potential ad recipients. There are multiple ways to find a person's home location: a telephone directory usually gives names and addresses, and many people give their home city as part of their social media profiles. However, the telephone directory can be incomplete or out-of-date, and social media profiles usually give only city-level resolution. Location measurements, such as those from GPS, are usually very precise, and they can be used to infer the location of a person's home. In this scenario, the buyer will seek to buy a small number of time-stamped location measurements from potential ad recipients and use the measurements to decide who should receive the ad.

2.1. Decision to deliver an advertisement

In this scenario, a buyer must choose whether or not to deliver an ad to a potential recipient, and the crux of this decision depends on whether or not the potential recipient lives in the targeted region. We model the costs to the buyer with a payoff matrix. The matrix describes the monetary gain or loss depending on the decision of whether or not to deliver an ad to the potential recipient and depending on whether or not the recipient lives in the region \mathcal{R} , as shown in Table 1. The buyer always has some uncertainty about the home location of the potential ad recipient.

The four cases in Table 1 represent the following scenarios:

- **Ad not delivered when home is *not* in region \mathcal{R}** (payoff b_{11}): This is a neutral outcome, because an ad was correctly withheld from a person who does not live in the targeted region. The cost (and benefit) is normally zero in this case; thus, $b_{11} = 0$.
- **Ad not delivered when home is in region \mathcal{R}** (payoff b_{12}): This is a negative outcome, because the ad should have been delivered, but was not. The cost is the lost opportunity and the possibility that a competitor may acquire the person as a customer; thus, $b_{12} \leq 0$.
- **Ad delivered when home is *not* in region \mathcal{R}** (payoff b_{21}): This is a negative outcome, because the ad was mistakenly delivered to a person whose home is not in the target region. The cost is the wasted cost of the ad plus the annoyance caused to the targeted person, so $b_{21} \leq 0$.
- **Ad delivered when home is in region \mathcal{R}** (payoff b_{22}): This is a positive outcome, because it could generate a purchase

Table 1. The payoff matrix for home targeted ads.

		Home location	
		Not in region	In region
Ad	Do not deliver	$b_{11} (0)$	$b_{12} (\beta)$
	Deliver	$b_{21} (\gamma)$	$b_{22} (1.0)$

The values in parentheses are used for our experiments.

from the business. The value would be the expected profit from a successful ad minus the cost of the ad, so $b_{22} \geq 0$.

We assume the payoff matrix values are given or can be learned.¹¹

Based on location data collected from the potential ad recipient, the buyer computes a probability distribution $P_H(\mathbf{h})$, where \mathbf{h} is a two-dimensional vector, $[x, y]^T$, that describes the location of the potential recipient's home. In Aly et al.,² we give a method to compute this distribution based on time-stamped location measures, such as the ones a buyer would purchase. From this distribution, we can compute the probability $p_{\mathcal{R}}$ that the home is inside the targeted region \mathcal{R} :

$$p_{\mathcal{R}} = \int_{\mathcal{R}} P_H(\mathbf{h}) d\mathbf{h}. \quad (1)$$

Based on this, we can compute the expected value of the revenue, V , given our decision on ad delivery:

$$\mathbb{E}[V | \text{no ad}] = (1 - p_{\mathcal{R}})b_{11} + p_{\mathcal{R}}b_{12},$$

$$\mathbb{E}[V | \text{ad}] = (1 - p_{\mathcal{R}})b_{21} + p_{\mathcal{R}}b_{22}.$$

The advertiser would choose whichever alternative has the largest expected revenue:

$$\mathbb{E}[V] = \max(\mathbb{E}[V | \text{no ad}], \mathbb{E}[V | \text{ad}]). \quad (2)$$

2.2. Decision to buy a GPS point

We consider the case where the buyer is presented with a list of points to evaluate buying, where each of these points has been recorded at a different time. The buyer is allowed to see the time stamps, but not the points' spatial coordinates.

The buyer will compute VOI to decide whether or not to buy a measured location point, having knowledge of only the point's time stamp. The buyer has already purchased n points, denoted by the random variables L_1, L_2, \dots, L_n or as the collection L_1^n . An instance of this random location variable is $l_i = [x_i, y_i, t_i, \sigma_i, c_i]^T$, which is a 5D vector with $[x_i, y_i]^T$ representing the point's 2D location at time t_i and the location precision represented as the standard deviation σ_i . We could optionally represent a varying precision for each measurement, but we assume all the users have similar location sensors with the same precision. The price of the point is c_i , which is the amount the buyer would have to pay the seller (potential ad recipient) to know (x, y) . This price is determined by the seller. Using these points, the buyer computes $P_{H|L_1^n}(\mathbf{h})$, which is a probability distribution of the home location based on location measurements 1 through n . We give a method for this computation in Aly et al.² The buyer then computes the probability that the home is in the target region (Equation (1)) and the expected revenue $\mathbb{E}[V | L_1^n]$, as described above.

The buyer has the option of buying another location measurement L_{n+1} . The VOI can then be defined as the gain in revenue by receiving the $n + 1$ th location $L_{n+1} = \ell_{n+1}$:

$$VOI(\ell_{n+1} | L_1^n = \ell_1^n) = \mathbb{E}[V | L_1^{n+1} = \ell_1^{n+1}] - \mathbb{E}[V | L_1^n = \ell_1^n]. \quad (3)$$

The location of this new point is unknown to the buyer, but it follows a distribution $P_{L_{n+1}}(\ell_{n+1})$. This distribution is the buyer's guess about where the unseen point L_{n+1} may be. We give a principled way to compute this in Aly et al.² It is based on experimental data about how a person's distance from home varies over the day. In the middle of the night, people are normally close to home, but they are normally farther away at noon. Because of the uncertainty surrounding the location of the new point, the buyer is reduced to computing the expected VOI. This comes from Equation 3, but it includes an expectation integral over $P_{L_{n+1}}(\ell_{n+1})$, which is the probability density of all possible locations of the new point. This expected VOI is $EVOI(L_{n+1} | L_1^n = \ell_1^n)$.

The decision to buy the $n + 1$ th point will be based on whether the value of the point in expectation, that is, $EVOI(L_{n+1} | L_1^n = \ell_1^n)$, is larger than the cost of the point, c_{n+1} . Thus, we will buy the point that maximizes the expected profit:

$$\mathbb{E}[\text{Profit} | L_1^{n+1} = \ell_1^{n+1}] = EVOI(L_{n+1} | L_1^n = \ell_1^n) - c_{n+1}. \quad (4)$$

Here we assume that the potential ad recipients have placed a price on their location data. This price could also be set by a location broker who acts as a representative of the potential ad recipient. We note that although this equation accounts for the price of the location point, the price of the ad has already been accounted for in the values of the payoff matrix.

If we assume zero expected profit for the buyer, Equation 4 can be rearranged to show a fair price for the location point as

$$c_{n+1} = EVOI(L_{n+1} | L_1^n = \ell_1^n). \quad (5)$$

Note that the price is independent of the actual location of the data. However, as the seller knows the location, a deeper analysis could adjust the price based on location. However, this price adjustment could in turn convey extra information to the seller about the potential value of the point, that is, if it is near the seller's home.

2.3. Algorithm for decisions

The final algorithm followed by the data requester, and illustrated in Figure 1, consists of repeated computations of the

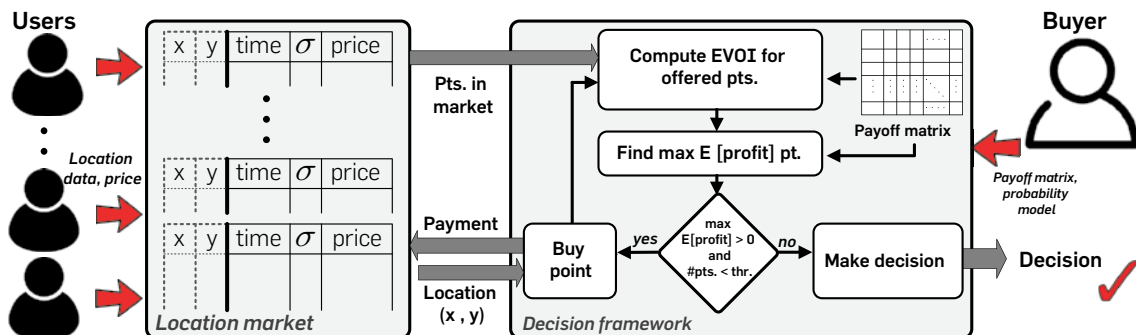
expected profit from Equation 4 over all the available points from the user. The buyer repeatedly buys the point with the maximum expected profit (Equation 4) as long as at least one point has an expected profit greater than zero, and as long as the number of points purchased does not exceed a preset threshold. When there are no more profitable points, or if the threshold has been exceeded, the buyer harnesses the information collected to decide whether or not to send the ad according to Equation 2.

2.4. Evaluation experiments

To evaluate the proposed decision framework, we used a GPS dataset of 66 participants living in Seattle, Washington, USA. The trajectories were collected for an average of 40.12 days ($\sigma = 24.43$) and have an average sampling rate of 0.77 samples/minute. The trajectories represent data offered by the user to the data buyer. We define three regions to test our framework. We have 13, 14, and 18 users living in regions R_1 , R_2 , and R_3 , respectively. To find the ground truth home location for each user, we leverage each user's full trajectory and the American Time Use Survey¹² (ATUS). ATUS points out that users are most likely to be at their homes at midnight. Thus, we apply density-based clustering (DBSCAN) on the user's time-stamped location trajectory. Then, the largest collection of data points (cluster) at midnight is identified as the user's home.⁸

We have compared the described methods to two other techniques that represent simple, practical methods to decide whether or not to send an ad to a user. For the first of these techniques, the advertiser simply makes a random decision to send the ad or not, with the probability of sending the ad set to 0.5. We call this technique "No points." In the second comparison technique, the data requester buys a number of points from the user at random times of day. Then, the ad is sent to the user only if the majority of the purchased points are inside the region. This method reflects an assumption that users tend to spend most of their time around their homes. Using our default price of 0.01 per point, our new, proposed method recommends buying no more than 20 points in about 85% of the cases, when the expected profit per point reaches zero. Thus, in our second comparison method, we have the data requester buy 20 points regardless of their expected benefit. We call this

Figure 1. Proposed data-sharing mechanism and decision framework: users offer their passively crowdsourced, time-stamped data with a certain location accuracy for a fixed price, while hiding the actual coordinates. Data buyers estimate the value of the offered data, buy points with the maximum expected profit, and make a business decision based on the points they have purchased.



second technique “20 points.” In addition, for our proposed new method, we set a maximum threshold of 20 points in the evaluation to represent a realistic case where the buyer is interested in buying a bounded amount of data. We refer to our proposed method as “VOI decision.”

Evaluation metrics. To evaluate the proposed decision framework, we employ three metrics: (1) *The true positive rate* (TPR) measures the proportion of correctly sent ads (i.e., ads sent to people with homes in the region); (2) *the false positive rate* (FPR) measures the proportion of incorrectly sent ads (i.e., ads sent to people with homes outside the region); and (3) *the revenue ratio* measures the ratio of the revenue gained to the maximum revenue the advertiser can gain by making perfectly correct decisions about which users should receive the ad without buying any location points.

Results. To test our proposed framework for different payoff matrices, we created a payoff matrix with the values in parentheses as shown in Table 1. Here, we have $b_{11} = 0$, which represents the neutral result of not sending an ad to someone whose home is outside the region \mathcal{R} . To reduce the size of the parameter space, we normalize by setting $b_{22} = 1$, which represents the reward for correctly delivering an ad to someone whose home is inside the region. The other two outcomes are negative: $b_{21} = \gamma$ represents the penalty for delivering an ad to someone not in the region, and $b_{12} = \beta$ represents the penalty for not delivering an ad to someone who does live in the region. We let both γ and β vary over $[0.0, -0.9]$. These normalizations mean we can show results over just two payoff parameters (γ and β) rather than four.

We compared the performance of our method to other methods in Figure 2. Figure 2 shows the average results over the three regions for the different payoff matrices for a GPS point cost of 0.01. The two comparative methods (“No points” and “20 points”) TPR and FPR are independent of the payoff matrix values, because they are neither considering the costs and benefits of buying points nor making ad decisions. The algorithm “No points” (red surface) has a TPR and FPR of around 0.5. The algorithm “20 points” (yellow surface) generally performs better for both TPR and FPR, but comes with the penalty of buying 20 points for every decision. Our price sensitive “VOI decision” algorithm (blue surface) is superior to both the comparison algorithms for TPR. For FPR in Figure 2(b), the “VOI decision” algorithm (blue surface) is superior over most of the payoff range. Its FPR rises dramatically when γ is zero, where the penalty for sending an ad outside the region is zero. Finally, Figure 2(c) shows the revenue ratios of the three methods, where “VOI decision” is again significantly superior.

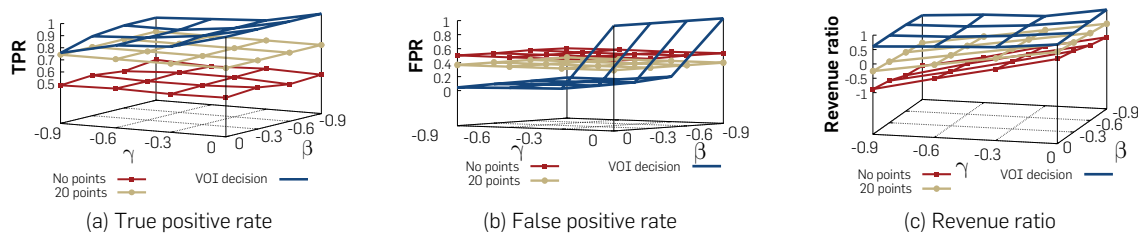
The other two algorithms actually lose money in some regions of the payoff matrix, whereas the “VOI decision” algorithm is always positive. Specifically, “VOI decision” relatively improves the TPR on average by 80.2% and 20.9% and up to 107.9% (when $\gamma = 0$ and $\beta = -0.6$) and 43.7% (when $\gamma = 0$) as compared to the “No points” and “20 points,” respectively. Also, “VOI decision” relatively improves the FPR on average by 38.2% and 15.8% and up to 91.1% (when $\gamma = -0.9$ and $\beta = 0$) and 78.7% (when $\gamma = -0.9$ and $\beta = 0$) as compared to the “No points” and “20 points,” respectively. Moreover, “VOI decision” reduces the number of points bought to make the decision on average by 60% as compared to “20 points.”

3. SCENARIO 2: TRAFFIC STATE ESTIMATION

We now focus on a second scenario, which is a service that provides traffic state estimates for a given road segment using crowdsourced spatiotemporal data. In particular, the traffic state estimator service buys time-stamped location data from people traveling through the road network and uses it to estimate their speed. Then, this uncertain speed estimate is used to infer the road segment’s discrete traffic state. For instance, we assume three levels for a highway road segment: **green** representing free flow/smooth traffic with speed greater than 60 km/hr, **red** representing congested traffic with speed less than 30 km/hr, and **yellow** representing medium congested traffic with speed between 30 and 60 km/hr. The service uses the points it buys to decide which level to assign to the road segment.

For clarity of illustration, we assume that the vehicle is on a single road segment for the duration of the analysis. The procedure described here can be generalized to the use of data from multiple vehicles traversing multiple road segments. In steady state, we assume the service has at least one previously purchased location measurement from the vehicle. This purchased data is used to place the vehicle on the road segment of interest, and it means that any subsequent point purchased from the vehicle can be used to estimate the speed of the segment using the points’ time stamps. The service provider must decide whether or not to buy a new location point from the vehicle as well as which point to buy with only knowledge of the points’ time stamps and location precision. Although crowdsourcing traffic speeds is a familiar idea, we show how to choose intelligently which points to buy and to compute their value. Throughout the rest of the section, we will describe how the service provider will use the proposed framework to make two decisions: (1) congestion-level descriptor (color) for the road segment and (2) whether to buy a new point from travelers.

Figure 2. Home targeted ads (Scenario 1) experiment results using the proposed framework (“VOI decision”) as compared to two other methods (“No points” and “20 points”).



3.1. Congestion level decision

As in the first scenario, we model the decision costs of the data buyer using a payoff matrix. The matrix describes the monetary gain and loss depending on the provider's choice of which color to display and the road segment's actual traffic state, as shown in Table 2. There are nine different possible cases: b_{rr} , b_{yy} , and b_{gg} represent positive outcomes where the service provider is choosing the correct traffic congestion level (red, yellow, and green, respectively); thus, b_{rr} , b_{yy} , and $b_{gg} > 0$. The remaining cases represent negative outcomes as the service provider is choosing a wrong congestion level descriptor. For example, payoff b_{gr} represents choosing smooth traffic (green) although actually it is congested (red). Thus, these payoffs are less than b_{rr} , b_{yy} , and b_{gg} and are generally less than zero. When the actual road speed is red (severely congested), choosing green (free-flowing) would have a relatively large cost, $b_{gr} < 0$, because it could mistakenly entice drivers toward the segment only to find slow speeds. We assume the payoff matrix is given or can be learned.¹¹

To choose the congestion level from the noisy location data, we again employ decision theory principles.¹¹ Specifically, the service provider uses the purchased location data to model their belief about the traffic segment's speed. This distribution is $P_U(u)$, where u represents the vehicle's speed. We give a method to compute this distribution in Aly et al.² From this distribution, we can compute the probability that the road segment's congestion level is green as follows:

$$p_g = \int_{\mathcal{R}(g)} P_U(u) du$$

where $\mathcal{R}(g)$ represents the range of speeds for the green road coloring, which is $[60, \infty]$ in our scenario. Similar equations are used to compute the probabilities of the yellow and red states, p_y and p_r .

With these probabilities, we can compute the expected revenue V for any congestion level display choice from the payoff matrix in Table 2. This is as below for the decision "r", and the decisions "g" and "y" can be evaluated similarly.

$$\mathbb{E}[V | \text{decision is } r] = p_r b_{rr} + p_y b_{ry} + p_g b_{rg},$$

We assume the service provider will choose to display the congestion level that gives maximum revenue, and thus the expected revenue ($\mathbb{E}[V]$) will be

$$\mathbb{E}[V] = \max(\mathbb{E}[V | r], \mathbb{E}[V | y], \mathbb{E}[V | g]).$$

In Aly et al.,² we discuss how the service provider computes $P_U(u)$ from individual time-stamped location measurements.

Table 2. Payoff matrix for traffic state estimation.

		Actual traffic state		
		Red	Yellow	Green
Traffic State Decision	Red	b_{rr}	b_{ry}	b_{rg}
	Yellow	b_{yr}	b_{yy}	b_{yg}
	Green	b_{gr}	b_{gy}	b_{gg}

The fundamental method is a Kalman filter, which gives the probability distribution $P_U(u)$ representing the speed estimate and its uncertainty as well as a distribution giving a prediction of the speed in the future, which gives a buyer an idea of what the next speed value will be. The next section discusses how to make decisions about the location points to buy.

3.2. Decision to buy a GPS point

The buyer must decide whether to buy a new point based on its time stamp and accuracy. In this scenario, we formulate the decision as one of buying a new speed estimate. We leverage VOI to compute the value of knowing the traveler's unknown speed and use it to make the buying decision. Having already purchased n speed estimates, this data forms a list of speeds, denoted by the random variables U_1, U_2, \dots, U_n or as U_1^n . Using these speeds, the data requester uses a Kalman filter to compute $P_{U|U_1^n}(u)$, which is a probability distribution of the road segment speed based on speed measurements 1 through n . The buyer also computes their expected revenue $\mathbb{E}[V | U_1^n]$, as described in section 3.1, using $P_{U|U_1^n}(u) \sim \mathcal{N}(\hat{u}_n, (\hat{\sigma}_n^n)^2)$ as the speed distribution. The mean \hat{u}_n and variance $(\hat{\sigma}_n^n)^2$ of this normal distribution are predicted by the Kalman filter. Because we are assuming the user is traveling at a locally constant speed, the Kalman estimate serves as the anticipated distribution of the as yet unknown next speed that the buyer is considering.

The value of information at time n can then be defined as the gain in revenue by receiving the $n + 1$ th speed measurement $U_{n+1} = u_{n+1}$:

$$VOI(u_{n+1} | U_1^n = u_1^n) = \mathbb{E}[V | U_1^{n+1} = u_1^{n+1}] - \mathbb{E}[V | U_1^n = u_1^n]. \quad (6)$$

Hence, the expected value of information for the $n + 1$ th speed is given by the expected value of (6):

$$EVOI(U_{n+1} | U_1^n = u_1^n) = \int_u VOI(u | U_1^n = u_1^n) \cdot P_{U_{n+1}}(u | U_1^n = u_1^n) du \quad (7)$$

where $u \in \mathbf{R}$ and the integral is taken over the full domain of u .

The decision to buy the $n + 1$ th speed will be based on whether the value of the point in expectation, that is, $EVOI(U_{n+1} | U_1^n = u_1^n)$, is larger than the cost of the speed (c_{n+1}):

$$\mathbb{E}[\text{Profit}] = EVOI(U_{n+1} | U_1^n = u_1^n) - c_{n+1}. \quad (8)$$

Here, we are assuming that the driver/data provider has placed a price on their location (speed) data.

We give results of detailed experiments in the next section. To build intuition about these computations, we present results of a simple simulation experiment in Figure 3. For different vehicle speeds, Figure 3 displays the number of points purchased using the methodology. Note that we buy more points whose speeds are near the congestion level thresholds, that is, 30 and 60. In effect, the method is trying to resolve the ambiguity of speeds near the speed boundaries to avoid the cost of mistakes as expressed in the payoff matrix. In addition, as the location precision σ_l decreases, the method buys more points as needed to resolve the speed uncertainty.

3.3. Evaluation experiments

We evaluated our proposed framework in two ways: First, we used simulation studies to evaluate the effect of points' cost on the performance of the proposed methodology across the entire speed spectrum (0–140 km/hr). In addition, we show the effect of the payoff matrix on the accuracy and compare the performance to a mean filter with different window sizes as our baseline technique. For each speed in a range from 0 to 140 km/hr with an increment of 1 km/hr, we ran 500 experiments. We estimate speeds from noisy location data with precision σ_l as described in the experiments, and we sample locations every 3 seconds. We report the average results of the experiments for each speed in the experimental range. The default payoff matrix is $[b_{rr} \ b_{ry} \ b_{rg}; b_{yr} \ b_{yy} \ b_{yg}; b_{gr} \ b_{gy} \ b_{gg}] = [1 \ -0.1 \ -0.1; -0.1 \ 1 \ -0.1; -0.1 \ -0.1 \ 1]$, and the default point cost is $c_i = 0.001$. We show the effect of the point cost, point precision, and the decision-maker's payoff matrix on the proposed framework as compared to the baseline technique. Second, we test the performance of our framework against real driving traces.

Effect of point cost and precision. Using simulated data, Figure 4 shows the effect of the point cost on the performance of the proposed framework in terms of congestion level decision accuracy for different location precisions, that is, $\sigma_l \in \{3m, 10m, 20m\}$ in parts a, b, and c of Figure 4, respectively. The blue bars show the percentage of correct speed interval inferences. We see that less expensive points lead to higher system accuracy, because the blue bars grow as the points become less expensive. This is because the system is more willing to buy additional points. As the price of the location points exceeds their value, the buyer refrains from buying. Comparing parts a, b, and c of Figure 4, we also see that lower precision (larger σ_l) leads to more error, as the blue bars generally shrink from a to b to c. In Figure 4, the error assigned to choosing the correct speed interval

Figure 3. Average number of points bought at different possible speeds for location points with an accuracy of 3m, 10m, and 20m. The model buys more points near the traffic state boundaries. The payoff matrix is $[1 \ -0.1 \ -0.1; -0.1 \ 1 \ -0.1; -0.1 \ -0.1 \ 1]$, cost = 0.01 and $\Delta t = 3s$.

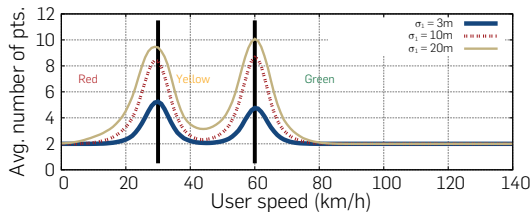
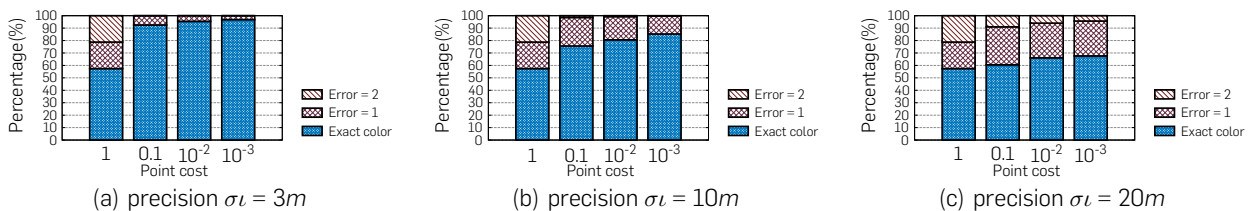


Figure 4. Effect of point cost on congestion level/color decision accuracy while users are driving at different possible speeds (0–140 km/hr) for location points with a precision of 3m, 10m, and 20m.



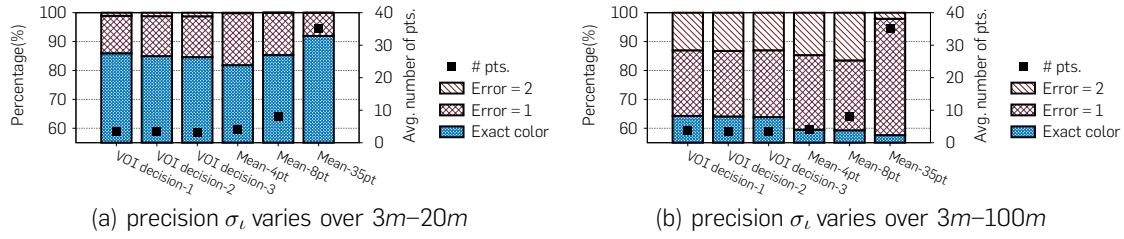
for the road segment is zero, represented by the blue bars. Choosing an adjacent interval (e.g., red instead of yellow) has an error of one, and choosing the interval at the other end of the spectrum (e.g., green instead of red) has an error of two.

Comparative analysis. Figure 5(a) compares the performance of our framework to the mean window filter over different window sizes (baseline technique). The bars in Figure 5 show the error rates in the same way as shown in Figure 4. We also show the mean number of points purchased in these figures as small, black boxes. For relatively accurate location points (with precision σ_l varying uniformly at random from 3 to 20 m), Figure 5(a) shows that our proposed framework identifies the exact traffic congestion level at least 84.6% of the time (“VOI decision-3” bar in the figure); this is better than the baseline technique with window size 4 points by 3.4% and with a reduction in the average number of purchased points by 20%. In addition, our approach has comparable performance to the baseline technique with window sizes 8 and 35 points along with a reduction in the number of purchased points by 60% and 90.9%, respectively.

For more noisy location estimates (with σ_l varying uniformly at random from 3 to 100 m), our proposed framework estimates the exact traffic congestion level at least 63.9% of the time (“VOI decision-3” bar), as shown in Figure 5(b). This is better than the baseline technique with windows sizes 4, 8, and 35 points by 7.3%, 7.10%, and 10.8%, respectively. Moreover, this comes with a reduction in number of purchased points of 15%, 57.5%, and 90.2%, respectively. Our framework gives higher accuracy with fewer location points. Figure 5 also shows that varying the payoff matrix resulted in a small change in the accuracy and the average number of purchased points as seen in the first three bars. With a larger penalty for making a wrong decision, the framework buys more points and gives higher accuracy.

Validation experiments with real data. Using the same GPS data as we did for the experiments in Section 2.4, we extracted 20 traces from drivers on the I-90 interstate highway and State Route 520 in Seattle, WA, at different dates and times of day. All 20 traces had more than 8 points on the road in order to compare with a mean filter with window size 8. The traces' speeds varied from 10 to 133 km/hr ($\mu = 89.4$ km/hr and $\sigma = 36.5$), covering the three congestion levels. We estimate the road congestion level ground truth by applying an alpha-trimmed filter to remove speed outliers and estimate the speed from the full traces. Using the default payoff matrix, our framework was able to identify the road segment's congestion levels accurately

Figure 5. The black squares show the average number of points bought while users are driving at different possible speeds for location points with randomly varying precision in the range 3–20 m and 3–100 m. This is compared to a mean filter with window sizes of 4, 8, and 35 location points. The payoff matrix for VOI decision-1 is $[b_{rr} \ b_{ry} \ b_{rg}; b_{yr} \ b_{yy} \ b_{yg}; b_{gr} \ b_{gy} \ b_{gg}] = [1 \ -0.9 \ -0.9; -0.9 \ 1 \ -0.9; -0.9 \ -0.9 \ 1]$, for VOI decision-2 is $[1 \ -0.4 \ -0.9; -0.4 \ 1 \ -0.9; -0.9 \ -0.4 \ 1]$, and for VOI decision-3 is $[1 \ -0.1 \ -0.1; -0.1 \ 1 \ -0.1; -0.1 \ -0.1 \ 1]$.



(with zero error) 95% of the time and within one level error 100% of the time. This is better than the mean filter, which gave accurate predictions (with zero error) 90% of the time. In addition, the model recommends purchase of 50% fewer points as compared to the mean filter.

4. SCENARIO 3: LOCATION PREDICTION

A third scenario centers on location prediction. The buyer in this case is interested in the future location of someone. For example, the buyer may want to know if a person will be near the buyer’s business place, which may prompt an ad delivery. A traffic authority may want to anticipate demand for the road network. In addition to introducing a new scenario, this section demonstrates a different form of the payoff matrix where the states and actions are continuous.

4.1. Location prediction

There are many existing techniques for predicting a person’s location based on location history. These include methods based on a Markov model⁴ and based on efficient driving and other cues.⁷ We introduce a new technique here that produces a continuous probability distribution over future locations, which meshes with our mathematical framework.

Using a single historical point ℓ_i taken at time t_i , the predicted location for a future time t_f is ℓ_f , given by the normal distribution:

$$P_{L_f|L_i} \sim \mathcal{N}(\ell_i, \sigma_f^2(t_i, t_f - t_i))$$

This implies that the normal distribution of future locations is centered around the measured location ℓ_i with a variance of $\sigma_f^2(t_i, t_f - t_i)$. This variance is a function of the current time t_i and the offset time into the future, $t_f - t_i$. Parameterizing the variance this way is intended to model the facts that (1) a person’s future location is a strong function of their current location, especially for the near future, and (2) prediction uncertainty changes with the current time and the time offset into the future. We computed a tabular approximation of $\sigma_f^2(t_i, t_f - t_i)$ from the data of all our test users, discretizing both t_i and $t_f - t_i$ to 30-min intervals.

Predicting ℓ_f from multiple purchased points ℓ_1^n gives a mixture of Gaussians:

$$P_{L_f|L_1^n}(\ell_f) = \frac{1}{n} \sum_{i=1}^n g(\ell_f | \ell_i, \sigma_f^2(t_i, t_f - t_i)) \quad (9)$$

Here, $g(\mathbf{x}|\mu, \sigma I)$ represents a two-dimensional Gaussian, centered at μ with a diagonal 2×2 covariance matrix $\sigma^2 I$. The accuracy of this prediction technique is given in Section 4.4.

Computing the VOI depends on anticipating the location of the next purchased point, L_{n+1} . We make a direct prediction of the location of the next purchased point, which is conveniently given by Equation 9, notated as $P_{L_{n+1}|L_1^n}(\ell)$.

4.2. Payoff and decision

We introduce a generic, continuous payoff function that depends on the distance between the predicted and actual future locations. If the buyer decides that the predicted location is ℓ_f^* , but the actual location is ℓ_f , then the payoff for this decision is $b^2 - \|\ell_f - \ell_f^*\|^2$. Here, b^2 is some base payoff for making an exact prediction, and the payoff decreases as the prediction error grows. This payoff function leads to a closed form for the expected revenue.

After some mathematics, detailed in Aly et al.³, it becomes apparent that the expected revenue for deciding a future location of ℓ_f^* then simplifies to

$$\mathbb{E}[V | \ell_f^*] = b^2 - \frac{1}{n} \sum_{i=1}^n (2\sigma_f^2(t_i, t_f - t_i) + \|\ell_i - \ell_f^*\|^2). \quad (10)$$

The buyer will want to maximize expected revenue by choosing the best value for ℓ_f^* . Differentiating the expected revenue in Equation 10 with respect to ℓ_f^* and setting it to zero gives the optimal location prediction as

$$\ell_f^* = \frac{1}{n} \sum_{i=1}^n \ell_i.$$

This shows the predicted location is simply the mean of the already purchased location points. Although this is a very simplistic location prediction, the key is choosing which points ℓ_i to buy for making an accurate prediction, which we describe next.

4.3. Value of information

By making the optimal prediction above, the expected revenue from previously purchased points ℓ_j^n would be

$$\mathbb{E}[V | L_1^n = \ell_1^n] = b^2 - \frac{1}{n} \sum_{i=1}^n (2\sigma_f^2(t_i, t_f - t_i) + \|\ell_i - \frac{1}{n} \sum_{j=1}^n \ell_j\|^2).$$

This shows that the expected revenue decreases with larger prediction variances and when the purchased points are more dispersed from their mean.

The VOI of an additional point ℓ^{n+1} is

$$\begin{aligned}
 \text{VOI}(\ell_{n+1} | L_1^n = \ell_1^n) &= \mathbb{E}[V | L_1^{n+1} = \ell_1^{n+1}] - \mathbb{E}[V | L_1^n = \ell_1^n] \\
 &= \frac{2}{n} \sum_{i=1}^n \sigma_f^2(t_i, t_f - t_i) - \frac{2}{n+1} \sum_{i=1}^{n+1} \sigma_f^2(t_i, t_f - t_i) \\
 &\quad + \frac{1}{n} \sum_{i=1}^n \left\| \ell_i - \frac{1}{n} \sum_{j=1}^n \ell_j \right\|^2 - \frac{1}{n+1} \sum_{i=1}^{n+1} \left\| \ell_i - \frac{1}{n+1} \sum_{j=1}^{n+1} \ell_j \right\|^2
 \end{aligned} \tag{11}$$

Two of the main terms in the equation above are independent of ℓ_{n+1} , that is, $\frac{2}{n} \sum_{i=1}^n \sigma_f^2(t_i, t_f - t_i)$ and $\frac{1}{n} \sum_{i=1}^n \left\| \ell_i - \frac{1}{n} \sum_{j=1}^n \ell_j \right\|^2$. The other two main terms depend on ℓ_{n+1} and thus affect the choice of which is the best point to buy next. The first of these terms, $-\frac{2}{n+1} \sum_{i=1}^{n+1} \sigma_f^2(t_i, t_f - t_i)$, encourages buying points that have a small associated prediction variance, $\sigma_f^2(t_{n+1}, t_f - t_{n+1})$. The second of these terms, $-\frac{1}{n+1} \sum_{i=1}^{n+1} \left\| \ell_i - \frac{1}{n+1} \sum_{j=1}^{n+1} \ell_j \right\|^2$, encourages buying points that help reduce the dispersion of the purchased points.

4.4. Evaluation experiments

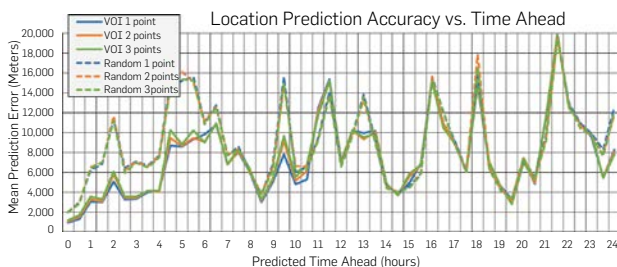
To test our prediction scenario, we used GPS data from the same 66 subjects as the ad delivery scenario described in Section 2.4. We used the temporal first half of each person’s data to compute one set of prediction variances, $\sigma_f^2(t_i, t_f - t_i)$, that pertain to all subjects. We represented t_i as the amount of time since the day’s previous midnight, discretized into 30-min intervals. The quantity $t_f - t_i$ represents the amount of time predicted into the future. We limited this to 24 hours and also discretized it to 30-min intervals.

For each subject, we randomly selected 100 test location points to predict from the temporal last half of their data. For each of these points, we randomly chose 20 prior points that were within our 24-hour prediction window as candidates for buying. With 66 subjects and 100 test predictions per subject, we tested our algorithm on 6600 different location prediction tasks.

Our primary test is to see if the algorithm is choosing good points to buy for making predictions. The next best point to buy is the one that maximized the expected VOI. As a comparison technique, we chose points randomly from the 20 available for each trial, repeating this 10 times for each of the 6600 prediction tasks.

Figure 6 shows the mean prediction error based on buying 1, 2, and 3 points. The solid lines show the VOI approach, and the correspondingly colored dashed lines show the random approach. From 0 to 7 hours into the future, the VOI technique

Figure 6. Using VOI to choose points to purchase is generally better than random choices in terms of prediction accuracy.



has noticeably smaller error than the random technique, after which the two techniques are approximately equal in error. Predicting ahead 0–30 min, the VOI technique reduces prediction error by 54%, 47%, and 40%, respectively for 1, 2, and 3 purchased points. This large reduction in error shows that the VOI technique is much better at choosing which location points to buy for increased location prediction accuracy.

5. CONCLUSION

We presented a principled method for buyers of location data to compute the value of users’ unseen location data. The approach relies on algorithms that consider probability distributions over locations based on data that has already been purchased, as well as the buyer’s pay-off matrix, to anticipate the value of future, as yet unpurchased data. As a by-product of the quantitative valuations, the methodology identifies which unseen data is likely the most valuable for the buyer. We considered three scenarios, home-targeted ads, traffic congestion inference, and location prediction, to illustrate how we estimate the value of location data obtained from end users in different settings. These techniques work significantly better than competing inference approaches, both by using less data and inferring more accurate results. We believe this work fills a gap in the pricing of location data and that the presented methods can help inform decisions by buyers and sellers of location data. □

References

- Adar, E., Huberman, B.A. A market for secrets. *First Monday* 8, 6 (2001).
- Aly, H., Krumm, J., Ranade, G., Horvitz, E. On the value of spatiotemporal information: principles and scenarios. In *Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems* (2018), ACM, 179–188.
- Aly, H., Krumm, J., Ranade, G., Horvitz, E. To buy or not to buy: Computing value of spatiotemporal information. *ACM Trans. Spat. Algor. Syst.* 4, 5 (2019), 22.
- Ashbrook, D., Starner, T. Using gps to learn significant locations and predict movement across multiple users. *Pers. Ubiquit. Comput.* 5, 7 (2003), 275–286.
- Cvrcek, D., Kumpost, M., Matyas, V., Danezis, G. A study on the value of location privacy. In *Proceedings of the 5th ACM Workshop on Privacy in Electronic Society* (2006), ACM, 109–118.
- Kanza, Y., Samet, H. An online marketplace for geosocial data. In *SIGSPATIAL* (2015), ACM, 10.
- Krumm, J., Horvitz, E. Predestination: Inferring destinations from partial trajectories. In *International Conference on Ubiquitous Computing* (2006), Springer, 243–260.
- Lv, M., Chen, L., Chen, G. Discovering personally semantic places from gps trajectories. In *CIKM* (2012), ACM, 1552–1556.
- Micro, T. How much is your personal data worth? survey says.... 2015.
- Monga, V. The big mystery: What’s big data really worth?, 2014.
- North, D.W. A tutorial introduction to decision theory. *IEEE Trans. Syst. Sci. Cybernet.* 3, 4 (1968), 200–210.
- U. B. of Labor Statistics. American time use survey, 2016.
- Staiano, J., Oliver, N., Lepri, B., de Oliveira, R., Caraviallo, M., Sebe, N. Money walks: A human-centric study on the economics of personal mobile data. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (2014), ACM, 583–594.

Heba Aly (hebaaly@amazon.com), Amazon, Seattle, WA, USA.

Gireeja Ranade (ranade@eecs.berkeley.edu), University of California, Berkeley, Berkeley, CA, USA.

John Krumm and **Eric Horvitz** (jkrumm, horvitz@microsoft.com), Microsoft Research, Redmond, WA, USA.

Technical Perspective

Progress in Spatial Computing for Flood Prediction

By Shashi Shekhar

IMAGINE YOU ARE considering buying a long-term place with a view of mountains or ocean. For due diligence, your partner asks about flood risk in the area. FEMA maps show the place is outside the 100-year flood zones (1% annual chance). However, you have heard that climate change is making extreme events more extreme and some places have seen multiple 100-year floods within a few years. Next, you browse information about climate change and its impact. These provide the projected climate change and precipitation (for example, rainfall, snowfall) under different policy scenarios, but not projected flood risk maps. How will you assess long-term flood risks for candidate places? This is an important societal use-case of and a challenging problem in spatial computing.^{3,4}

Flood forecast and risk assessment started as early as Egyptian civilization. The Nile monsoon floods nourished nearby fertile lands but also erased mud-based farm boundaries motivating the invention of land surveying, which was later formalized as geometry and trigonometry across the Mediterranean sea.

Communities started recording floods for forecasts and risk assessments to reduce loss of property and lives. Nile flood records go back hundreds of years and confound “one-size-fits-all” machine learning methods^{1,5} due to spatio-temporal auto-correlation, teleconnections (for example, upstream use and storage), and non-stationarity (for example, climate change). Physics-driven models¹ account for hydrological processes such as surface run-off, ground absorption, and evaporation using data about upstream precipitation, snow melt, water levels, flow rates, soil type, soil moisture, atmospheric humidity, temperature, river bathymetry, sewer networks, and so on. The model parameters are calibrated using historical data. Due to the high data needs and computational costs, hydro-


Flood forecasting and risk assessment started as early as the Egyptian civilization.

logical models are difficult to use for large areas and time-constrained use cases, for example, flash floods. Thus, many use Geographic Information Systems (GIS)-based approaches focusing on surface run-offs, sub-processes of flow accumulation, and depression overflows. For example, the ArcGIS hydro² uses maps of terrain elevation, precipitation and snow melt. These are converted to a flow graph, where nodes are locations and directed edges represent gravity-driven water-flows. For example, a single-flow graph sends flows towards the lowest neighbors.

Progress in spatial computing^{3,4} is made by either improving the approximation of spatial phenomena or by improving the computation cost or both. Spatial computing literature has investigated both for flood-modeling. It has investigated triangulated irregular networks (TINs) as an alternative to traditional gridded digital elevation models to not only reduce approximation error but also storage costs. It also advanced multi-flow graphs to allow water from a node to flow toward multiple downhill neighbors for more accurate representation of surface run-offs and downstream floods. However, there are few algorithms for multi-flow graphs beyond flow accumulation.

The authors of the following paper take a big step to fill this knowledge

gap. Design of scalable algorithms is more difficult for multi-flow graphs due to their weighted directed acyclic graph topology in contrast with the simpler tree or forest topology of single-flow graphs. While well-known transitive closure algorithms may estimate flow accumulations, new spatial algorithms are needed to efficiently compute edge weights and cascading depression overflows. This paper leverages properties of planar graphs, priority queues and fast matrix multiplication methods to address the challenges. Key results include new scalable (for example, linear or $n \log n$) algorithms for point-flood query, terrain-flood query and flood time query on multi-flow graphs. These were lauded by a best paper award at a recent ACM SIGSPATIAL conference and open doors for design of algorithms for next-generation use cases such as preparation of climate-change aware flood-plain maps with uncertainty quantification.

Going back to the opening story, you (or an insurance company) may use the algorithm for point-flood query to quickly assess flood risk for candidate properties. Smart cities and communities may use the terrain-flood query algorithms to identify flood-prone low-lying areas for remedies. 

References

- Jain, S. et al. A brief review of flood forecasting techniques and their applications. *Intl. J. of River Basin Management* 16, 3 (2018), 329–344. Taylor & Francis; doi: 10.1080/15715124.2017.1411920
- Maidment, D. A New Approach to Flood Mapping. ArcNews, ESRI Press, Summer 2018.
- Shekhar, S., Feiner, S. and Aref, W. Spatial computing. *Commun. ACM* 59, 1 (Jan. 2016), 72–81; <https://cacm.acm.org/magazines/2016/1/195727-spatial-computing/fulltext>.
- Shekhar, S. and Vold, P. Spatial computing. The MIT Press Essential Knowledge series, 2020; <https://mitpress.mit.edu/books/spatial-computing>
- University Consortium for Geographic Information Science A UCGIS Call to Action: Bringing the Geospatial Perspective to Data Science Degrees and Curricula, Summer 2018; <https://bit.ly/3iQoHoP>.

Shashi Shekhar is the McKnight Distinguished University Professor at the University of Minnesota, Minneapolis, MN, USA.

Copyright held by author.

Flood-Risk Analysis on Terrains

By Aaron Lowe, Pankaj K. Agarwal, and Mathias Rav

Abstract

An important problem in terrain analysis is modeling how water flows across a terrain and creates floods by filling up depressions. In this paper, we study a number of flood-risk related problems: given a terrain Σ , represented as a triangulated xy -monotone surface with n vertices, a rain distribution \mathcal{R} , and a volume of rain ψ , determine which portions of Σ are flooded. We give an overview of efficient algorithms for these problems as well as explore the efficacy and efficiency of these algorithms on real terrains.

1. INTRODUCTION

Flooding can be extremely dangerous and damaging. The United States experienced the wettest 12-month period from June 2018 to May 2019, with major flooding in the Midwest affecting millions of people and causing several billion dollars in damages. Being able to accurately and quickly model flooding can help predict and prepare for the risks. Flood-risk analysis has been studied widely across multiple research communities including environmental science, engineering, machine learning, and GIS communities: see Section 7.

Flood risk analysis also has been a focus of a number of companies as well. SCALGO²² is a software development and services company that uses massive terrain dataprocessing technology to provide a flood risk platform for Scandinavian countries. 3Di Water Management¹ provides interactive hydrodynamic simulation software combining overland, channel, sewer and ground water flow. Fathom¹³ uses high-resolution global datasets and hydrological modeling to provide flood hazard data for many applications, including insurance and disaster response.

In this paper, we will focus on two key problems related to flood-risk analysis, which are defined more formally in later sections:

Terrain-flood query: given a terrain Σ and a rain pattern, determine which portions of Σ will be flooded. (See Figure 1 for an example.)

Point-flood query: In some applications, the terrain Σ is fixed and we wish to know whether a query point on Σ will be flooded for a given rain pattern. Preprocess Σ into a data structure so that for a given rain pattern and query point $q \in \Sigma$, one can quickly determine whether q is flooded. Alternatively, one can ask how long rain must fall before q is flooded.

When rain falls, the rate at which a depression fills up depends not only on its shape and the size of its watershed (the area of the terrain that contributes water to the depression), but also on other depressions filling up. Water falling on the watershed of a depression that is already filled flows to a neighboring depression, effectively making its watershed

The results described here originally appeared in “Flood-risk analysis on terrains under the multiflow-direction model”¹⁸ and “Flood risk analysis on terrains,”²¹ respectively.

Figure 1. A terrain-flood query over a region of Philadelphia, PA, USA. The areas marked in blue are flooded, with regions that water flows over marked in orange.



larger and thus making it fill up faster. Maintaining how depressions fill and spill into other depressions during a flash flood makes the above problem challenging.

We present an efficient algorithm for the terrain-flood query in Section 4. The algorithm works by sweeping descending contours on the terrain, tracking where water flows and determining which depressions become full. A key feature of the algorithm is that once we find a contour delimiting a flooded region, we can mark the enclosed region as flooded and prune it from consideration.

For the point-flood query, we present in Section 5 an algorithm that preprocesses the terrain into a data structure so that queries can be answered quickly. If we assume the single-flow direction (SFD) model in which water from a vertex flows along the steepest descending edge, we describe an algorithm that, after preprocessing the terrain, can answer point-flood queries in time logarithmic in the number of vertices on the terrain. We also briefly discuss algorithms for the *flood-time query* that asks *when*, rather than *if*, a point will become flooded. These algorithms all work by recognizing that not all depressions are equally important from the perspective of the query point. The terrain can be simplified into a set of depressions called the tributaries. The local behavior within each tributary can then be ignored. Instead, the algorithms depend only on the global behavior: does the tributary become full, and if so, which downstream tributaries does it spill into?

We have implemented the algorithms for terrain-flood and point-flood queries and tested them on real terrains. We show that the algorithms are efficient in practice, across a variety of queries, and give some analysis as to how varying

The original version of this paper was published in the *Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems* (Nov. 2017) Article No. 36.

the query affects the running time. We also compare terrain-flood queries qualitatively under two variants, the multi-flow direction (MFD) and single-flow direction (SFD) models, showing cases where the flooded regions are significantly different under the two (Section 6).

2. PRELIMINARIES

Terrains. Let \mathbb{M} be a triangulation of \mathbb{R}^2 , and let \mathbb{V} be the set of vertices of \mathbb{M} ; set $n = |\mathbb{V}|$. We assume that \mathbb{V} contains a vertex v_∞ at infinity and that each edge $\{u, v_\infty\}$ is a ray emanating from u ; the triangles in \mathbb{M} incident to v_∞ are unbounded. Let $h : \mathbb{M} \rightarrow \mathbb{R}$ be a height function. We assume that the restriction of h to each triangle of \mathbb{M} is a linear map, that h approaches $+\infty$ at v_∞ , and that the heights of all vertices are distinct. Given \mathbb{M} and h , the graph of h , called a *terrain* and denoted by $\Sigma = (\mathbb{M}, h)$, is an xy -monotone triangulated surface whose triangulation is induced by \mathbb{M} .

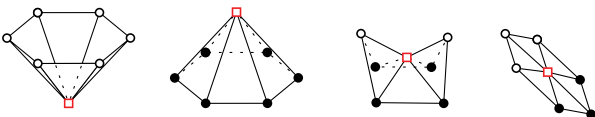
Critical vertices. There is a natural cyclic order on the neighbor vertices of a vertex v of \mathbb{M} , and each such vertex u is either an *upslope* or *downslope* neighbor, that is, $h(u) > h(v)$ or $h(u) < h(v)$, respectively. If v has no downslope (resp. upslope) neighbor, then v is a *minimum* (resp. *maximum*). We also refer to a minimum as a *sink*. If v has four neighbors w_1, w_2, w_3, w_4 in clockwise order such that $\max(h(w_1), h(w_3)) < h(v) < \min(h(w_2), h(w_4))$, then v is a *saddle* vertex. If a vertex is not a critical point, call it *regular*. See Figure 2.

Level sets, contours, depressions. Given $\ell \in \mathbb{R}$, the ℓ -*sublevel set* of h is the set $h_{\leq \ell} = \{x \in \mathbb{R}^2 \mid h(x) \leq \ell\}$, and the ℓ -*level set* of h is the set $h_{= \ell} = \{x \in \mathbb{R}^2 \mid h(x) = \ell\}$. Each connected component of $h_{\leq \ell}$ is called a *depression*. Each connected component of the boundary of a depression is a *contour*. A contour not passing through a critical vertex is a simple polygonal cycle. Note that a depression is not necessarily simply connected, as a maximum can cause a hole to appear.

For a point $x \in \mathbb{M}$, a depression β_x of $h_{\leq \ell}$ is said to be *delimited by the point x* if x lies on the boundary of β , which implies that $h(x) = \ell$. A depression β_1 is *maximal* if every depression $\beta_2 \supset \beta_1$ contains strictly more sinks than β_1 . A maximal depression that contains exactly one sink is called an *elementary depression*. Note that each maximal depression is delimited by a saddle, and a saddle that delimits more than one maximal depression is called a *negative saddle*. For a maximal depression β , let $\text{Sd}(\beta)$ denote the saddle delimiting β , and let $\text{Sk}(\beta)$ denote the set of sinks in β .

Merge tree. Suppose we sweep a horizontal plane from $-\infty$ to ∞ . As we vary ℓ , the depressions in $h_{\leq \ell}$ vary continuously, but their structure changes only at sinks and negative saddles. If we increase ℓ , then a new depression appears at a sink, and two depressions merge at a negative saddle. The merge tree, denoted by T_h , is a tree that tracks these changes.

Figure 2. From left to right: minimum (sink), maximum, saddle, and regular vertices. Upslope and downslope neighbors are marked in white and black, respectively.



Its leaves are the sinks of the terrain, and its internal nodes are the negative saddles. The edges of T_h are in one-to-one correspondence with the maximal depressions of Σ_h , that is, we associate each edge (u, v) , for $h(u) > h(v)$, with the maximal depression delimited by u and containing v . The point of height $\ell \in [h(v), h(u)]$ on edge (u, v) represents the depression of $h_{\leq \ell}$ contained in β_v . We assume that T_h has an edge from the root of T_h extending to $+\infty$, corresponding to the depression that extends to ∞ . See Figure 4. For simplicity, we assume that T_h is binary, that is, each negative saddle delimits exactly two depressions. Nonsimple saddles can be unfolded into a number of simple saddles.¹²

Let u be a negative saddle, let (u, v_1) and (u, v_2) be two down edges in T_h from u , and let (w, u) be the up edge from u . We call the depression associated with (u, v_2) (resp. with (w, u)) as the *sibling* (resp. *parent*) (depression) of that associated with (u, v_1) . Van Kreveld et al.¹⁶ gave an $O(n \log n)$ -time algorithm for constructing the merge tree in 2D. The algorithm was later extended to 3D by Tarasov and Vyalys,²³ and to arbitrary dimensions by Carr et al.⁸

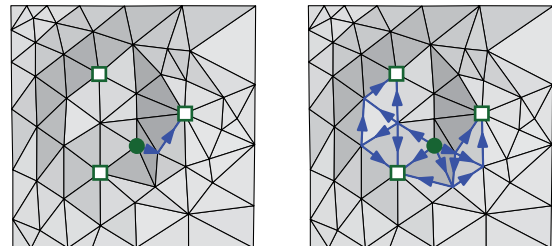
T_h can be preprocessed in $O(n)$ additional time so that for a point $x \in \mathbb{R}^2$, $\text{Vol}(\beta_x)$, the volume of the depression delimited by x can be computed in $O(\log n)$ time. In the following sections, we will be working with a fixed height function, so will drop the subscript h from T_h , Vol_h , etc.

3. FLOODING MODEL

Flow graph and flow functions. We transform \mathbb{M} into a directed acyclic graph \mathcal{M} , referred to as the *flow graph*, by directing each edge $\{u, v\}$ of \mathbb{M} in downward direction, that is, from u to v if $h(u) > h(v)$, and from v to u otherwise. For each (directed) edge (u, v) , we define the *local flow* $\lambda(u, v)$ to be the portion of water arriving at u that flows along the edge (u, v) to v .

The value of $\lambda(u, v)$ is, in general, based on relative heights of the downslope neighbors of u . We will refer to the general version in which water can flow along multiple downward edges from u as the *multi-flow direction* (MFD) model. If $\lambda(u, v) > 0$ for exactly one downslope edge from u , we will refer to this as the *single-flow direction* (SFD) model. See Figure 3 for an example of how these models differ. In some cases, notably for point-flood and flood-time queries, the SFD model will admit more efficient algorithms. We will not focus on the specifics of the local flow function, and only

Figure 3. Rain falls at the local maximum at the green circle toward local minima marked with squares. Left: an SFD model, water flows along a single path to a single minimum. Right: an MFD model, water flows along multiple paths to three local minima.



assume that it is specified and for a pair u, v can be retrieved in $O(1)$ time.

Following the edges of \mathcal{M} , water reaches a set of sinks of \mathbb{M} . We define a *flow function* $\phi: \mathbb{V} \rightarrow [0, 1]$, which specifies the proportion of water that flows from a vertex u to another vertex v . Note that under the MFD model, water can flow from u to v along many paths. The flow function is defined recursively as follows:

$$\phi(u, v) = \begin{cases} 1 & \text{if } u = v, \\ \sum_{(u, w) \in \mathbb{E}(\mathcal{M})} \lambda(u, w) \cdot \phi(w, v) & \text{otherwise.} \end{cases} \quad (1)$$

For a maximal depression β , we define

$$\phi(u, \beta) = \sum_{s \in \text{Sk}(\beta)} \phi(u, s)$$

to be the portion of water that reaches from a vertex u to β . Recall that $\text{Sk}(\beta)$ is the set of sinks in β .

If a maximal depression β delimited by saddle u is full, we define $\phi_\beta(u, v)$ to be the modified flow function, computed as if the flooded vertices have a height of $h(u)$.

Rain distribution. We let \mathcal{R} denote a *rain distribution*, which is specified as a probability distribution on the vertices of the terrain, that is, for each vertex $v \in \mathbb{V}$, $\mathcal{R}(v) \geq 0$ indicates the rate at which it rains on v , and we require that $\sum_v \mathcal{R}(v) = 1$. For a given depression β , let $\mathcal{R}(\beta) = \sum_{v \in \beta} \mathcal{R}(v)$ be the portion of rain falling directly into β . We denote by $|\mathcal{R}|$ the number of vertices with positive rainfall in \mathcal{R} , and we assume that \mathcal{R} is represented as a list of $|\mathcal{R}|$ pairs $(v, \mathcal{R}(v))$. In practice, $|\mathcal{R}| \ll n$.

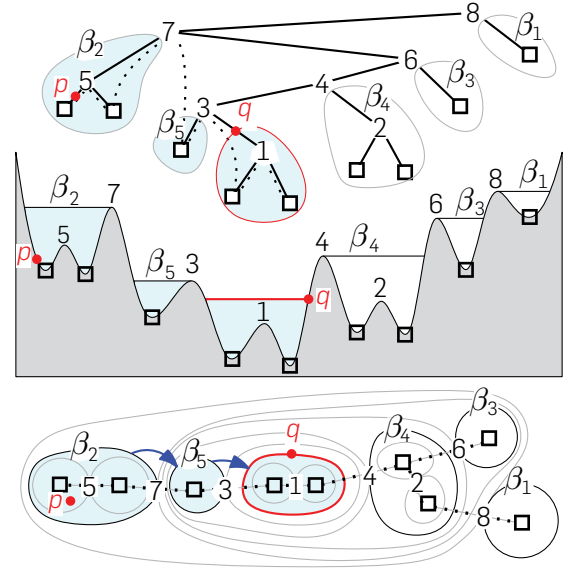
Flood propagation. Our flooding model follows a similar depression-filling model as Liu and Snoeyink.¹⁷ As rain falls according to a distribution \mathcal{R} on Σ , water flows along the downward edges according to the flow function and accumulates in depressions of Σ . When a maximal depression β_i fills up, water spills from the saddle v_i delimiting β_i toward sinks in the sibling depression. We refer to this event as a *spill event*.

The above process defines a sequence of spill events, each event marking a sink u as full, and redistributing the rain falling on u to other sinks. See Figure 4 for an example. In our model, the maximal depressions of Σ fill up at a constant rate between any two consecutive spill events. That is, after a spill event occurs at time t_1 and until the next occurs at time t_2 , the volume of water in each maximal depression is a nondecreasing linear function of time.

Tributaries. Any given point $q \in \mathbb{M}$ is contained in a sequence of maximal depressions $\alpha_1 \supset \dots \supset \alpha_k \ni q$, each α_i delimited by a saddle v_i with sibling depression β_i . These saddles form a path in \mathcal{T} from q to the root. We refer to the maximal depressions β_1, \dots, β_k as the *tributaries of q* and denote them by \mathcal{T}_q . See Figure 4.

Fill and spill rates. For a maximal depression β , we define the *fill rate* $F_\beta: \mathbb{R}_{\geq 0} \rightarrow [0, 1]$ as the rate at which water is arriving in the depression β as a function of time. That is, the rate at which rain is falling directly in β plus the rate at which other depressions are spilling water into β . The fill rate F_β is a monotonically nondecreasing piecewise-constant function. Similarly, we define the *spill rate* $S_\beta: \mathbb{R}_{\geq 0} \rightarrow [0, 1]$ as the rate (as a function of time) at which water spills from β through the saddle that delimits β . Let τ_β , called the *fill time*, be the

Figure 4. Example terrain and point-flood query (p, q) . Sinks are marked with squares, and saddles are marked with labels 1–8 indicating the saddle elevation. Dotted lines indicate spill events. **Top:** Merge tree with tributaries of q , β_1 – β_5 marked. **Middle:** Terrain seen from the side. **Bottom:** Terrain seen from above.



time at which β becomes full. Let β' be the sibling depression of β . Then,

$$S_\beta(t) = \begin{cases} 0 & \text{if } t < \tau_\beta \vee \tau_{\beta'} \leq \tau_\beta. \\ F_\beta(t) & \text{if } t > \tau_\beta \wedge \tau_{\beta'} > \tau_\beta. \end{cases}$$

By the definition of the fill rate, for any maximal depression β , its initial fill rate is

$$F_\beta(0) = \sum_{s \in \text{Sk}(\beta)} \sum_{v \in \mathbb{V}} \mathcal{R}(v) \phi(v, s), \quad (2)$$

which is how much rain water flows initially to the sinks of β . We can define $F_\beta(0)$ recursively using (1) as follows: abusing the notation a little, let $F_v(0)$ denote the water reaching a vertex v at time 0. Then,

$$F_v(0) = \mathcal{R}(v) + \sum_{(w, v) \in \mathbb{E}(\mathcal{M})} F_w(0) \lambda(w, v) \quad (3)$$

$$F_\beta(0) = \sum_{s \in \text{Sk}(\beta)} F_s(0). \quad (4)$$

For any $t \geq 0$, the fill rate of β at time t is the direct rain reaching β plus the water spilling into β from its tributaries that are full. That is,

$$F_\beta(t) = F_\beta(0) + \sum_{\alpha \in \mathcal{T}_\beta} S_\alpha(t) \phi(\text{Sd}(\alpha), \beta). \quad (5)$$

Fill and spill volumes. For a depression β , we similarly define $\mathcal{F}_\beta, \mathcal{S}_\beta: \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ as *fill- and spill-volume* functions of β , that is, $\mathcal{F}_\beta(t)$ tells how much water has arrived in depression β by time t , and $\mathcal{S}_\beta(t)$ tells how much water has spilled from β by time t .

$$\mathcal{F}_\beta(t) = \int_0^t F_\beta(x) dx \quad \text{and} \quad \mathcal{S}_\beta(t) = \int_0^t S_\beta(x) dx.$$

By definition of the spill rate, letting β' be the sibling depression of β , we have

$$S_{\beta}(t) = \begin{cases} 0 & \text{if } t < \tau_{\beta} \vee \tau_{\beta'} \leq \tau_{\beta}, \\ \mathcal{F}_{\beta}(t) - \text{Vol}(\beta) & \text{if } t > \tau_{\beta} \wedge \tau_{\beta'} > \tau_{\beta}. \end{cases}$$

4. TERRAIN-FLOOD QUERIES

In this section, we describe an algorithm for answering a terrain-flood query. That is, given a rain distribution \mathcal{R} and a volume ψ , determine which vertices of \mathbb{M} will be flooded if a volume of ψ rain falls according to the distribution \mathcal{R} . A key idea of the algorithm is that if a vertex v is flooded, then all points lying in the depression β_v are also flooded, so we do not have to process the vertices lying in β_v . We just need to know how much (if any) water will spill to its downstream tributaries. Using this simple observation, we compute the flooded vertices of \mathbb{M} for a given \mathcal{R} as follows.

Overview of the algorithm. As a preprocessing step, we construct the merge tree T , and in doing so, we augment T with a linear-size data structure so that for each point q (i) the edge of T containing q and (ii) $\text{Vol}(\beta_q)$ can each be computed in $O(\log n)$ time.

For a given rain distribution \mathcal{R} , we first compute how much rain directly falls in each maximal depression initially. For a maximal depression β , let $\hat{\mathcal{R}}(\beta)$ be the rate of rain falling on vertices in β which are not contained in any other maximal depression. Using the recurrence

$$\mathcal{R}(\alpha) = \hat{\mathcal{R}}(\alpha) + \sum_{(\alpha, \beta) \in T} \mathcal{R}(\beta),$$

$\mathcal{R}(\alpha)$ for all maximal depressions $\alpha \in T$ can be computed in $O(|\mathcal{R}| + m)$ time.

Next, we process the vertices in descending height order, and at each vertex v , we maintain the following:

- The set of *active depressions* that are not completely filled depressions in the $h(v)$ -sublevel set
- For each active depression α (i) the fill volume \mathcal{F}_{α} , that is, the volume of water in α , and (ii) the set of edges crossing into α , denoted $E(\alpha)$, and finally
- For each edge $e \in E(\alpha)$, the volume of rain flowing along e denoted by $\Lambda(e)$

As we sweep the vertices from top to bottom, we will find the height at which depressions are flooded. At this point, we mark the corresponding depression as flooded and do not consider any vertices contained in the flooded depression.

We now describe in detail how we process each vertex.

Processing a nonsaddle vertex. Let α_v be the smallest maximal depression containing β_v . A key observation is that \mathcal{F}_{α} is the same as \mathcal{F}_{α_v} . Therefore, it does not change at a nonsaddle vertex and thus has already been computed at an earlier step. Then, if $\text{Vol}(\beta_v) \leq \mathcal{F}_{\alpha_v}$, that is, β_v is flooded, we mark all vertices contained in β_v as flooded and remove β_v from the set of active depressions. If β_v is not flooded, for each edge (v, w) in T , we compute the water flowing along it as:

$$\Lambda(v, w) = \lambda(v, w) \left(\mathcal{R}(v)\psi + \sum_{(u, v) \in \mathbb{E}} \Lambda(u, v) \right). \quad (6)$$

Processing a saddle vertex. If v is a nonflooded saddle vertex delimiting two maximal depressions α, α' , in addition to the above process, we must also compute the volume of rain in each of the two depressions. To do so, partition the edges $E(\beta_v)$ into the two sets $E(\alpha)$ and $E(\alpha')$ and compute the volume of rain crossing into α (resp. α') as $\sum_{e \in E(\alpha)} \Lambda(e)$ (resp. $\sum_{e \in E(\alpha')} \Lambda(e)$). See Figure 5 for an example. These sums can be computed in a total of $O(n \log n)$ time summed over all saddles. Using this value along with the value of $\mathcal{R}(\alpha)$ (resp. $\mathcal{R}(\alpha')$) in (7), we can compute \mathcal{F}_{α} (resp. $\mathcal{F}_{\alpha'}$).

Then, the volume of water in a depression β is the amount falling directly in it, plus the amount of water flowing into it,

$$\mathcal{F}_{\beta} = \psi \mathcal{R}(\beta) + \sum_{e \in E(\beta)} \Lambda(e). \quad (7)$$

Finally, we use this value along with the depression volumes to check if α or α' is flooded. Note that because v is not flooded, at most, one of these depressions will be flooded. If one is flooded, without loss of generality, let it be α . In this case, mark α as flooded, and add α' to the set of active depressions. Then, the volume of rain spilling from α into α' will be $\mathcal{F}_{\alpha} - \text{Vol}(\alpha)$. Let $\lambda'(v, w)$ be the modified flow function, computed as if the flooded neighboring vertices have a height of ℓ . Then, we update the flow of water along the edges from v as follows:

$$\Lambda(v, w) = \lambda'(v, w) \left(\mathcal{R}(v) + \sum_{(u, v) \in \mathbb{E}} \Lambda(u, v) + (\mathcal{F}_{\alpha} - \text{Vol}(\alpha)) \right). \quad (8)$$

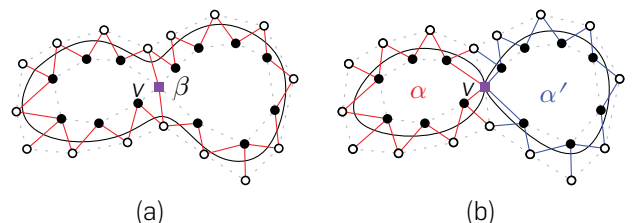
If neither α nor α' is full, add them both to the set of active depressions.

THEOREM 4.1. *Given a triangulation of \mathbb{M} of \mathbb{R}^2 with n vertices, a height function $h : \mathbb{M} \rightarrow \mathbb{R}$ that is linear on each face of \mathbb{M} , a rain distribution \mathcal{R} , and a volume of rain ψ , the flooded vertices of \mathbb{M} can be computed in $O(n \log n)$ time.*

5. POINT-FLOOD QUERIES

Given a rain distribution \mathcal{R} , a query point $q \in \mathbb{M}$, and a rain volume ψ , the point-flood query asks whether the point q is flooded if a volume ψ rain falls with distribution \mathcal{R} . Of course, the terrain-flood query procedure described in Section 4 can answer this query, but our goal is to answer this query more

Figure 5. (a) A single active depression β with active edges marked in red. (b) Saddle vertex v (marked in purple) delimits two maximal depressions α and α' . The active edges are partitioned into two sets: those crossing into α (resp. α') marked in red (resp. blue); the edges connecting to v in (a) are no longer active (corresponding to upslope edges), and downslope edges from v are now active and partitioned accordingly.



efficiently. We first discuss an algorithm for the MFD model and then describe how the running time can be further improved for the SFD model. We also briefly discuss a variant of the point-flood query, the *flood-time query*, which asks how much it must rain before a query point $q \in \mathbb{M}$ is flooded.

5.1. Point-flood query under MFD

Under the MFD model, the algorithm exploits two observations. First, we need not compute the fill volume of all maximal depressions. In particular, suppose q lies in a maximal depression β and there are two children depressions β_1 and β_2 of the sibling depression β' of β . We do not have to compute the fill volume of β_1 and β_2 . It suffices to compute if β' fills and how much, if any, water spills from β' to the depression β_q . In fact, the only depressions we need to consider are the tributaries of q . Second, we introduce the notion of tributary graph that describes how water flows between the tributaries of q . The tributary graph is used to quickly compute the fill and spill volumes of the tributaries of q .

Using these ideas, we describe an $O(nm)$ -size data structure for the MFD model that answers a point-flood query in $O(|\mathcal{R}|k + k^2)$ time, where m is the number of sinks in \mathbb{M} and k is the number of tributaries of q .

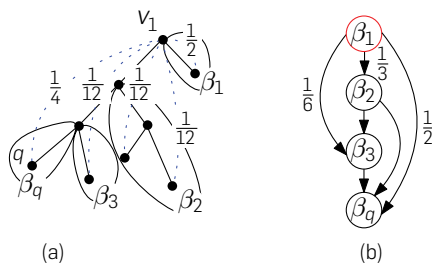
Tributary graph. For a point $q \in \mathbb{M}$, the tributary graph $G[q] = (X_q, E_q)$ is a directed acyclic graph. X_q is the depression β_q plus its tributaries. For a pair of depressions $\alpha, \beta \in X_q$, we add the directed edge (α, β) to E_q if $h(\text{Sd}(\alpha)) > h(\text{Sd}(\beta))$ and $\phi_\alpha(\text{Sd}(\alpha), \beta) > 0$ (if $\beta = \beta_q$; then, by $\text{Sd}(\beta_q)$, we mean q .) Recalling that ϕ_α is the flow function when the depression α is flooded, we set the weight of the edge (α, β) to be $\phi_\alpha(\text{Sd}(\alpha), \beta)$. For the MFD model, this can be computed as

$$w(\alpha, \beta) = \frac{\phi(\text{Sd}(\alpha), \beta)}{\sum_{(\alpha, \gamma) \in E_q} \phi(\text{Sd}(\alpha), \gamma)}, \quad (9)$$

that is, the weights are normalized so that the weighted out-degree of each node in $G[q]$ is 1. See Figure 6 for an example.

Overview of algorithm. It is expensive to compute the fill and spill volume functions \mathcal{F}_β and \mathcal{S}_β exactly for each tributary of q , so we define slightly different functions $\tilde{\mathcal{F}}_\beta, \tilde{\mathcal{S}}_\beta : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ for all $\beta \in X_q$. They are fill, spill volume functions under the assumption that every tributary of β_q fills before its sibling, that is, water spills from a tributary β to various sinks in the sibling β' of β ; note that β' is a depression containing q .

Figure 60 (a) A merge tree T , with tributaries $(\beta_1, \beta_2, \beta_3)$ of q delimited and flow from v_1 to each sink s , $\phi(v_1, s)$, marked. (b) Tributary graph $G[q]$, with the edge weights from β_1 .



We define $\tilde{\mathcal{F}}_\beta, \tilde{\mathcal{S}}_\beta$ recursively using the tributary graph $G[q]$ as follows:

$$\begin{aligned} \tilde{\mathcal{F}}_\beta &= F_\beta(0)\psi + \sum_{(\alpha, \beta) \in E_q} \tilde{\mathcal{S}}_\alpha w(\alpha, \beta), \\ \tilde{\mathcal{S}}_\beta &= \max\{0, \tilde{\mathcal{F}}_\beta - \text{Vol}(\beta)\}. \end{aligned} \quad (10)$$

For any given ψ , $\tilde{\mathcal{F}}_{\beta_q} < \text{Vol}(\beta_q)$ if and only if $\mathcal{F}_{\beta_q} < \text{Vol}(\beta_q)$. So the point-flood query can be answered by computing $\tilde{\mathcal{F}}_{\beta_q}$ and returning yes if this quantity is at least $\text{Vol}(\beta_q)$.

Preprocessing step. In the preprocessing step, we construct the merge tree T and preprocess it so that for a point $q \in \mathbb{M}$, (i) the edge of T containing q and (ii) $\text{Vol}(\beta_q)$ can be computed in $O(\log n)$ time.

Additionally, for each vertex $v \in \mathbb{M}$ and for each of $O(m)$ maximal depressions β , we store the value of $\phi(v, \beta)$. Actually, we need to store only nonzero values; in practice, the number of such pairs is much smaller than mn .

Preprocessing takes $O(n \log n + nm)$ time, and the size of the data structure is $O(nm)$.

Query procedure. For a query rain distribution \mathcal{R} and a query point, we first find the edge e of T containing q and $\text{Vol}(\beta_q)$. Given e , we compute the tributaries of q in $O(k)$ time by traversing T upward from e . We now construct the tributary graph $G[q] = (X_q, E_q)$ in $O(k^2)$ time, using the precomputed values of $\phi(v, \beta)$ in (9).

To compute $\tilde{\mathcal{F}}_\beta$ for all depressions $\beta \in X_q$, we first compute $F_\beta(0)$ for each $\beta \in X_q$ using the formula

$$F_\beta(0) = \sum_{u: \mathcal{R}(u) > 0} \mathcal{R}(u)\phi(u, \beta)$$

and then use the recurrence 10. The total time spent by the query procedure is $O(|\mathcal{R}|k + k^2)$. Putting everything together, we obtain the following:

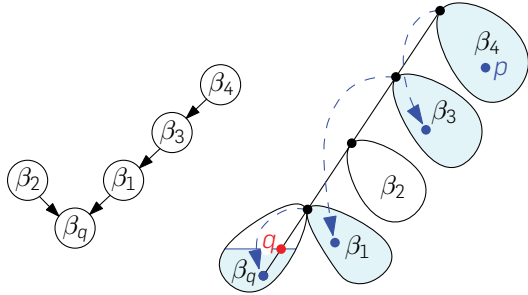
THEOREM 5.1. *Given a triangulation of \mathbb{M} of \mathbb{R}^2 with n vertices, a height function $h : \mathbb{M} \rightarrow \mathbb{R}$ that is linear on each face of \mathbb{M} , a data structure of size $O(mn)$ can be constructed in $O(n \log n + mn)$ time so that a point-flood query can be answered in $O(|\mathcal{R}|k + k^2)$ time, where $|\mathcal{R}|$ is the complexity of the query rain distribution, k is the number of maximal depressions containing the query point, and m is the number of sinks in the terrain (\mathbb{M}, h) .*

5.2. Point-flood query under SFD

If the water flows according to an SFD model, point-flood queries can be answered even more efficiently. Under the SFD model, a key observation is that the tributary graph $G[q]$ is a tree because each tributary has out degree 1; see Figure 7. To see why, note that for each vertex v , water flows from v to exactly one sink γ . Importantly, each tributary α upon becoming full will spill to a single sink γ , that is, $\phi_\alpha(\text{Sd}(\alpha), \gamma) = 1$, and for all other sinks, this will be 0. Letting β be the tributary containing γ , the edge (α, β) will have unit weight in $G[q]$, and there will be no other edges from α .

Single-point source. First consider the case when rain falls only at a single point p (contained in tributary, say β , of q). As $G[q]$ is a tree, there is a unique path π from β to β_q . When a tributary becomes full, the water begins spilling to the next

Figure 7. Left: $G[q]$ is a tree under the SFD model. Right: When rain falls at $p \in \beta_4$, the tributaries along the path from β_4 to β_q in $G[q]$ fill and spill in order.



tributary in π . For q to become flooded, all the tributaries in π must be flooded. We can answer the point-flood query by simply following the tributaries π , pushing excess water to the next tributary until either q is flooded or we come to a tributary that does not get filled. See Figure 7. However, if q has k tributaries, this algorithm takes $O(k)$ time, and in the worst case, $k = \Omega(n)$.

The query can be expedited using a well-known data structure called a *heavy-path tree decomposition* on T . In this data structure, T is partitioned into *heavy-paths*, such that every path intersects $O(\log n)$ heavy-paths. By precomputing prefix sums of tributary volumes along each heavy-path, we can process in amortized $O(1)$ time all the tributaries in π intersecting a given heavy-path. Therefore, point-flood queries for a single-point source can be answered in $O(\log n)$ time. We can again use the heavy-path decomposition data structure to add the volumes of tributaries, but the running time now depends on the number of tributaries in which rain is falling.

Region source. This approach can be extended to work for rain falling in multiple tributaries. When paths from two of these tributaries intersect, both spilling into a common tributary, we simply add the spill volume from both. See Figure 8.

THEOREM 5.2. *Given a triangulation of \mathbb{M} of \mathbb{R}^2 with n vertices, a height function $h: \mathbb{M} \rightarrow \mathbb{R}$ that is linear on each face of \mathbb{M} , a data structure of size $O(n)$ can be constructed in $O(n \log n)$ time so that a point-flood query under the SFD model can be answered in $O(|\mathcal{R}| + k \log n)$ time, where $|\mathcal{R}|$ is the complexity of the query rain distribution and k is the number of tributaries of q on which it is raining.*

5.3. Flood-time query

Given a rain distribution \mathcal{R} and a query point $q \in \mathbb{M}$, we can also ask how much it must rain before q becomes flooded. Now, instead of just maintaining the spill and fill volumes for a fixed volume of rain ψ , we must maintain the functions for spill and fill rates as defined in (2) and (5). Under the SFD model, the algorithm described above can be extended to answer the flood-time queries without increasing the space or time complexity. The main idea is that given the rates at the predecessors of a node α in $G[q]$, the

Figure 8. Top: Under the SFD model, water spilling from two tributaries intersects at a single downstream tributary. Bottom: Under the MFD model, (a fraction of) water spilling from two tributaries may intersect at many downstream tributaries.

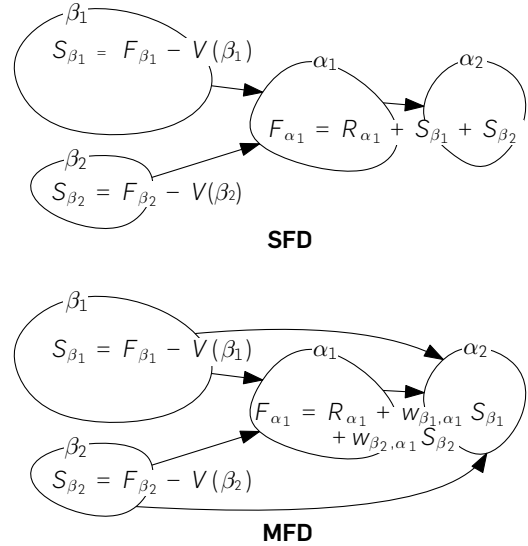
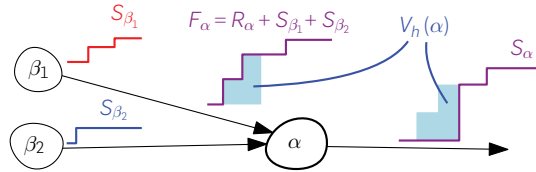


Figure 9. The fill rate F_α is computed from the spill rates S_{β_1} and S_{β_2} ; the spill rate S_α is computed from F_α .



fill and spill rates at α can be computed in $O(\log n)$ amortized time. See Figure 9.

However, under the MFD model, computing spill and fill rates becomes significantly more complex as the water spilling from a tributary may split and fill multiple downstream tributaries. We can, however, do better than the naive approach adapting ideas from the algorithm for the SFD model. The main idea is that by using matrix multiplication, the downstream effect of spilling from multiple tributaries can be computed in a single step. If the product of two $k \times k$ matrices can be computed in $O(k^\omega)$ time, a flood-time query can be answered in $O(|\mathcal{R}|k + k^\omega + k^2 \log n)$, where $|\mathcal{R}|$ is the complexity of the query rain distribution and k is the number of tributaries of q .

6. EXPERIMENTS

In this section, we present experiments we have conducted on real terrain data to demonstrate the efficiency of these algorithms and compare qualitatively the flooding under SFD and MFD models.

We have implemented the terrain-flood algorithm, described in Section 4, in C++, as well as a version of the point-flood algorithm.

We study the performance of the algorithms on three publicly available grid DEMs:

- The *Indiana dataset*, a 0.89 mi^2 model of a suburban area 0.5 mi northeast of Holland, IN, USA
- The *Philadelphia dataset*, a 225 km^2 model of an urban area in the northwest area of Philadelphia, PA, USA
- The *Norway dataset*, a $10,000 \text{ km}^2$ model of a mountainous region located in the Jotunheimen National Park, Norway

For the SFD model, water flows from a vertex to its lowest neighbor. For the MFD model, water flows from a vertex to all downslope neighbors with the relative rates proportional to the gradient of the slope.

SFD vs. MFD flooding. We considered the rain distribution \mathcal{R} to be rain falling: (i) on a single vertex or (ii) uniformly over a region.

Our experiments show that when rain is falling at a single point, the areas that are flooded under the SFD and MFD models can be quite different. Under the MFD model, some large areas may become flooded that would not under the SFD model (Figure 10). As we increase the region on which rain is falling, we still see differences in the areas flooded, although they may be less pronounced (Figure 11). For example, the same general regions may be flooded, but under the MFD model, more water might end up in one location as opposed to that in SFD model, or water may reach more depressions. When we expand the rain distribution to be falling over the whole terrain, the regions that are flooded tend to be very similar.

Another difference in the two models, irrespective of the area of the region where rain is falling, is how water flows over the terrain. Under the SFD model, water flows along disjoint paths, whereas under the MFD model, it spreads more on the terrain (Figures 10 and 11). We illustrate these observations here with a few examples.

For the case when rain falls at a single point, we computed the flooded areas with rain volume of 10^5 m^3 on the Indiana dataset and 10^7 m^3 on the Norway dataset. Figure 10 shows the terrain-flood query for two single point rain distributions under both the SFD and MFD models. In Figure 10(a), we see that under the SFD model water follows a single path from p north east, first filling a large region before

spilling and filling a series of smaller regions as the water flows west toward a feature corresponding to a river. In Figure 10(b), under the MFD model, the water splits at p and fills a number of depressions to the south west of p . We additionally note that under the MFD model in (b) water spreads out more and flows across a wider path between full depressions.

For the case where rain is falling uniformly over a small square, we set the rain distribution to be uniform over a square of size $1 \text{ km} \times 1 \text{ km}$ and set $\psi = 10^6 \text{ m}^3$ and computed the flooded area for the Norway dataset, under both SFD and MFD models. Figure 11 shows the queries along with enlarged images of the region on which it is raining. We see that although similar areas become flooded, water spreads out across the peak more under the MFD model, and a larger fraction of the water flows to the southwest. In contrast, under the SFD model, the rain follows narrow bands outside of the rain region, and a larger fraction of the water flows to the north.

Performance of algorithms. Building the merge tree and preprocessing it to compute the depression volume of any point as well as to perform lowest common ancestor queries took an average of 1.33 s over 5 trials for the Indiana dataset containing 10^6 vertices.

We also ran tests over the Philadelphia dataset, taking subregions with 1.6×10^7 , 9×10^6 and 10^6 vertices, and on the Norway dataset. Our experiments show that in practice, the preprocessing time is near linear in the size of the terrain. Although preprocessing does require sorting the nodes, which takes $O(n \log n)$ time in the worst case, in practice, the constant on this term is much smaller than that of the linear steps in the preprocessing.

We examined the running time of the terrain-flood query on the Indiana dataset as we change the amount of

Figure 10. 10^5 m^3 (resp. 10^7 m^3) of rain falls at p in the Indiana dataset. The flooded regions of Σ are marked in blue, with regions that water flows over marked in red. Water flows according to SFD in (a) and MFD in (b).

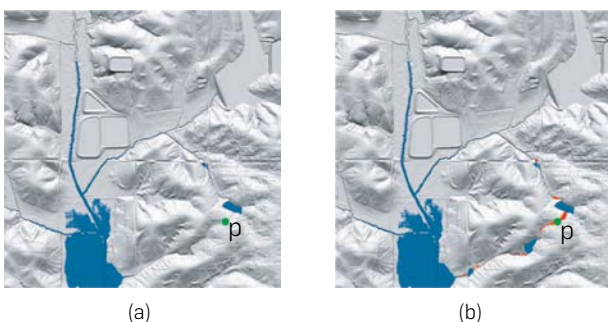
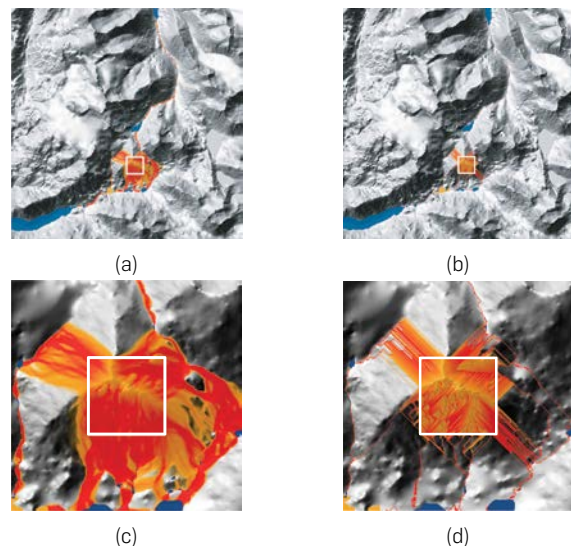


Figure 11. 100 m of rain falls uniformly over the square outlined in white in the Norway dataset. (a) Water flows according to MFD, (b) water flows according to SFD. (c) and (d) show a $3 \text{ km} \times 3 \text{ km}$ area around the region it is raining on.



rain ψ . Increasing the volume of rain first increases the running time until it reaches a peak and then decreases, becoming very fast with the largest volumes of rain. When a small amount of rain is falling in a small area, water reaches very few depressions and thus only a small portion of the merge tree is explored by the algorithm. As the volume and area of rain increases, the algorithm explores larger portions of the merge tree leading to an increase in the running time. However, once the volume of rain increases further, large depressions get filled and the algorithm succeeds in pruning large portions of the merge tree. Roughly speaking, the running time of the algorithm is proportional to the number of depressions that are partially filled.

Finally, we examined the running time of the terrain-flood query on the Indiana and Philadelphia datasets as we change the number of points with positive rain in \mathcal{R} , raining uniformly over squares of varying sizes. Although the running time increases with the number of vertices with positive rainfall, it grows slower than the linear dependence indicated by the worst-case time analysis.

7. RELATED WORK

The flood-risk problem has been widely studied in multiple research communities, and many different approaches have been proposed to tackle this problem. One such approach, coming from the hydrology community, simulates fluid dynamics, using nonlinear partial differential equations such as the Navier-Stokes equations.^{7, 14} They often account for additional factors, such as the effects of different terrain types and drainage networks. Although these models tend to be the most accurate, naive applications are computationally expensive and many techniques such as multiresolution representation of the terrain and approximate computation have been proposed to expedite the computation.^{7, 14, 25} Notwithstanding much work on to reducing the computational cost of these methods, these algorithms are hard to scale to large terrains.

Recently, machine-learning-based approaches have been proposed for predicting flood risk.^{10, 24} These approaches are relatively fast, while maintaining a reasonable level of predictive power. However, these methods generally are used for predicting fluvial floods (i.e., flooding of rivers) and rely on stream gauges or other sensors already being installed in the area of interest. Tehrany et al.²⁴ tested the efficacy of various support vector machine (SVM) kernels at predicting the overall flood hazard of points on a terrain using historical flood events and a number of terrain features such as slope, altitude, surface runoff, and distance from a river. Chang et al.¹⁰ used self-organizing maps and neural networks to forecast the flood inundation in the near future (1–3 h) given the current inundated areas.

There has been extensive work on modeling water flow on a terrain in the GIS community.^{2–6, 9, 11, 15, 17, 19–21} These approaches use simpler models, focusing on the geometry of the terrain. These tend to be more computationally efficient and suitable for large datasets. However, the simplifying assumptions mean that they may not be

as accurate as PDE-based models in all situations. For example, they do not take into account absorption of water into the ground and are thus more suitable for flash floods wherein most of the flooding occurs over a shorter timespan. Liu and Snoeyink¹⁷ (see also Arge et al.⁶) proposed an $O(n \log n)$ -time algorithm under the single-flow direction (SFD) model that computes the fill times of all depressions assuming rain is falling at a constant rate on the entire terrain.

Arge et al.⁴ described an $O(n \log n)$ -time algorithm, under the SFD model, to compute the set of flooded vertices when a given volume of rain $\psi \geq 0$ falls on a given region of the terrain.

8. CONCLUSION


In this paper, we have presented efficient algorithms for a few flood-risk queries: the *terrain-flood* query that asks which vertices of a terrain will be flooded and the *point-flood* query that asks if a given point will be flooded. The work is only a small step toward performing flood-risk analysis in real time over a large terrain, and there are many open questions:

Can these algorithms be extended to incorporate uncertainty in data as well as in the model? There have been some preliminary results for uncertainty of terrain height under the SFD model,²¹ but this problem remains largely open.

The flooding model described in this paper depends only on the elevation of the terrain data. In reality, there are other factors that affect flooding such as terrain type and permeability as well as drainage networks. Can a model be developed that incorporates additional terrain data? Furthermore, can historical flood data be incorporated into this model to more accurately compute flood risk?

The flooding model described here also assumes that water flows only along edges of the terrain and that water flows instantaneously to the sinks. Although these assumptions are reasonable in some settings (e.g., flash floods and high-resolution terrain models), can a model be developed that incorporates the velocity of the water and allows for channel flow?

Acknowledgments

Work by Lowe and Agarwal is supported by NSF under grants CCF-15-13816, CCF-15-46392, and IIS-14-08846, by ARO grant W911NF-15-1-0408, and by grant 2012/229 from the U.S.-Israel Binational Science Foundation. Work by Rav is partially supported by the Innovation Fund Denmark. 

References

- 3Di Water Management. <https://3diwatermanagement.com>, 2019
- Agarwal, P.K., Arge, L., Yi, K. I/O-efficient batched union-find and its applications to terrain analysis. In *Proceedings of the 22nd Annual Symposium on Computational Geometry* (2006), 167–176.
- Arge, L., Chase, J., Halpin, P., Toma, L., Vitter, J., Urban, D., Wickremesinghe, R. Efficient flow computation on massive grid terrain datasets. *GeoInformatica* 4, 7 (2003), 283–313
- Arge, L., Rav, M., Raza, S., Revsbæk, M. I/O-efficient event based depression flood risk. In *Proceedings of the 19th Workshop on Algorithm Engineering and Experiments* (2017), 259–269.
- Arge, L., Revsbæk, M. I/O-efficient contour tree simplification. In *International Symposium on*

Algorithms and Computation (2009), 1155–1165.

6. Arge, L., Revsbæk, M., Zeh, N. I/O-efficient computation of water flow across a terrain. In *Proceedings of the 26th Annual Symposium on Computational Geometry* (2010), 403–412.
7. Bates, P.D., De Roo, A. A simple raster-based model for flood inundation simulation. *J. Hydrol.* 1-2, 236 (2000), 54–77.
8. Carr, H., Snoeyink, J., Axen, U. Computing contour trees in all dimensions. *Comput. Geomet.* 2, 24 (2003), 75–94.
9. Carr, H., Snoeyink, J., Panne, M. Flexible isosurfaces: Simplifying and displaying scalar topology using the contour tree. *Comput. Geomet.* 1, 43 (2010), 42–58.
10. Chang, L.-C., Shen, H.-Y., Chang, F.-J. Regional flood inundation nowcast using hybrid som and dynamic neural networks. *J. Hydrol.*, 519 (2014), 476–489.
11. Danner, A., Mølhøve, T., Yi, K., Agarwal, P., Arge, L., Mitásová, H. Terrastream: From elevation data to watershed hierarchies. In *Proceedings of the 15th Annual ACM International Symposium on Advances in GIS* (2007), 28.
12. Edelsbrunner, H., Harer, J., Zomorodian, A. Hierarchical morse complexes for piecewise linear 2-manifolds. In *Proceedings of the 17th Annual Symposium on Computational Geometry* (2001), 70–79.
13. Fathom Global. <https://www.fathom.global/fathom-global>, 2019.
14. Ghimire, B., Chen, A.S., Guidolin, M., Keedwell, E.C., Djordjević, S., Savić, D.A. Formulation of a fast 2d urban pluvial flood model using a cellular automata approach. *J. Hydroinform.* 3, 15 (2013), 676–686.
15. Jenson, S., Domingue, J. Extracting topographic structure from digital elevation data for geographic information system analysis. *Photogramm. Eng. Rem. Sens.* 11, 54 (1988), 1593–1600.
16. Kreveld, M., Oostrum, R., Bajaj, C., Pascucci, V., Schikore, D. Contour trees and small seed sets for isosurface traversal. In *Proceedings of the 13th Annual Symposium on Computational Geometry* (1997), 212–220.
17. Liu, Y., Snoeyink, J. Flooding triangulated terrain. In *Proceedings of the 11th International Symposium on Spatial Data Handling* (2005), 137–148.
18. Lowe, A., Agarwal, P.K. Flood-risk analysis on terrains under the multiflow-direction model. In *Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, (ACM, 2018), 53–62.
19. O’Callaghan, J., Mark, D. The extraction of drainage networks from digital elevation data. *Comp. Vis. Graph. Image Process.* 3, 28 (1984), 323–344.
20. Quinn, P., Beven, K., Chevallier, P., Planchon, O. The prediction of hillslope flow paths for distributed hydrological modelling using digital terrain models. *Hydrol. Process.* 1, 5 (1991), 59–79.
21. Rav, M., Lowe, A., Agarwal, P. Flood risk analysis on terrains. In *Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in GIS* (ACM, 2017), 36.
22. SCALGO. www.scalgo.com, 2019.
23. Tarasov, S., Vyalyi, M. Construction of contour trees in 3d in $o(n \log n)$ steps. In *Proceedings of the 14th Annual Symposium on Computational Geometry* (1998), 68–75.
24. Tehrany, M.S., Pradhan, B., Mansor, S., Ahmad, N. Flood susceptibility assessment using gis-based support vector machine model with different kernel types. *Catena*, 125 (2015), 91–101.
25. Volp, N., Van Prooijen, B., Stelling, G. A finite volume approach for shallow water flow accounting for high-resolution bathymetry and roughness data. *Water Resour. Res.* 7, 49 (2013), 4126–4135.

Aaron Lowe and Pankaj K. Agarwal
 ({aaron, pankaj}@cs.duke.edu), Duke University, Durham, NC, USA.

Mathias Rav (mathias@scalgo.com), SCALGO, Aarhus, Denmark.

© 2020 ACM 0001-0782/20/9 \$15.00

Computing and the National Science Foundation, 1950-2016

Building a Foundation for Modern Computing

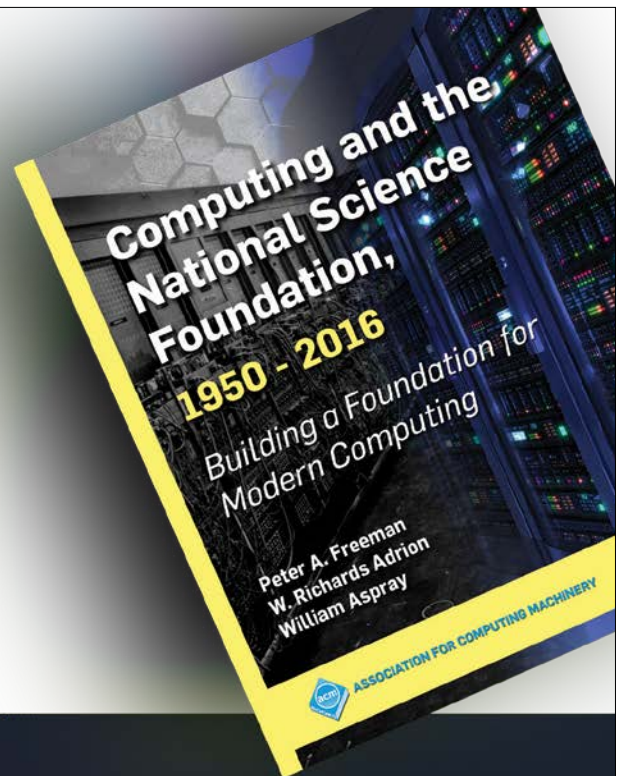
Peter A. Freeman
W. Richards Adrion
William Aspray

ISBN: 978-1-4503-7271-8

DOI: 10.1145/3335772

<http://books.acm.org>

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



ACM BOOKS
 Collection II

[CONTINUED FROM P. 104]

“Forget the WOW signal, then, you’ve got first contact, Ray. This is intelligent, there’s no doubt. Can you show me that as binary? Take the peaks as ones and troughs as zeroes.”

“No problem.” Beach tapped at the keyboard and a string of zeroes and ones started to fill up the screen. “Is that too fast?”

Karlsson shook her head. “You know reading binary’s my party trick. So, first we’ve got the simple number sequences, but the rest of it, that’s not random garbage. It’s structured, I’m sure. It could be code, you know? It feels like code.”

“Don’t give me that,” said Beach. “If this is alien, it’ll be nothing like a recognizable programming language.”

“I realize that, but there’s something familiar in the structure. Like it’s doing something I should recognize. Print the first page off for me, I’m going to take a bath and think about it.”

Ng was on the phone when Karlsson burst from the bathroom, dripping water over the floor. Beach shook his head, pointing the phone and mouthed “The prof.” Karlsson didn’t speak, but waved the sodden printout at him.

“Sorry,” Beach said down the phone. “QRM.”

“The press office is very excited,” said Ng, “and they’ve got the provost involved too. You don’t tell anyone, okay? Not until we’re sure what this is.”

“I’ve already told Maya,” said Beach. “Code is her thing. She can help.”

“Put it on speaker!” Karlsson said, waving the sheet of paper again.

Beach shrugged. He flipped up the phone’s settings as Ng said something inaudible.

“Hi Wendy, it’s Maya here. Don’t blame Ray, he had to tell me, and I know what the signal is.”

“How can you know?” said Ng. “It’s got to be totally alien.”

“Is it, though?” said Karlsson. “It’ll take time to work out the detail, but I’ve already identified three basic operations for a Turing machine. Totally different from how we’d do it, but that’s what it is. A Turing machine is universal, though it’ll be friggling slow to untangle and execute.”

9/3/27, 10:00 A.M.

Sweat was dripping from Ng’s brow, though the a/c was turned up too

“Jocelyn and Hewish were joking when they called their source LGM for ‘Little Green Men.’ This is bound to have a natural explanation, too, but it sure looks like communication.”

high. She tapped Karlsson on the back. “Any progress?”

Karlsson shook her head. “There was no need to do it this way. Stand-alone computer, Faraday cage—it’s dumb. Alien code can’t be a computer virus that can escape onto the Internet and take over the world. You’ve seen too many movies. It’s like expecting a Windows virus to infect a person. It couldn’t happen.”

“The precautionary principle applies,” said Ng. “However small, there’s...”

“Whoa!” Beach jumped from his seat alongside Karlsson. They all stared at the screen. A collection of windows monitored progress of the Turing machine simulator running the alien code. Now, the windows seemed to melt, dripping down the screen to leave behind a blank nothing.

“Couldn’t happen?” said Ng.

“Okay, we don’t know... Oh, my God!” Karlsson was staring not at the screen, but the wall behind it. As they watched, the wall itself began to melt away, just as the image on the screen had done. In the gap they could see the building, the sky, everything dissolving.

“So, maybe it was a virus,” said Karlsson. “But for the universe’s operating sys ...”

0/0/0, 0:00 A.M.

UNIVERSE BOOT SEQUENCE RE-START □

Brian Clegg (www.brianclegg.net) is a science writer based in the U.K.

© 2020 ACM 0001-0782/20/9 \$15.00



Digital Government: Research and Practice

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



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

From the intersection of computational science and technological speculation, with boundaries limited only by our ability to imagine what could be.

DOI:10.1145/3409981

Brian Clegg

Future Tense

Little Green Message

A different kind of first-contact scenario.

8/11/27, 3:00 P.M.

“You need to look at this.” Ray Beach nodded toward the screen.

“Oh damn.” Wendy Ng, director of the radio astronomy observatory at the newly opened University of California, Las Vegas, slumped into the chair beside her grad student. “How long have you been picking it up?”

“Maybe half an hour. I’d nearly finished the final engineering run. I was testing the orientation locks on the antennae.”

“Could it be a blind injection?”

Beach grimaced. “No, our sadistic colleagues don’t deliver fake signals when we’re just testing. Anyway, I checked with them. It isn’t a pulsar, is it?”

Ng shook her head. “No. The frequency’s all wrong, and anyway pulsars don’t behave like that.” She moved closer, running a finger along the screen under the displayed string of pulses, collected together in groups—1, 2, 4, 8, 16—repeating over and over. “You’re recording this, I presume?”

Beach fought back the urge to be sarcastic. His thesis advisor had no sense of humor. “Yep. It’s standard procedure on engineering runs, for diagnostics.”

“This could be LGM-1 all over again.”

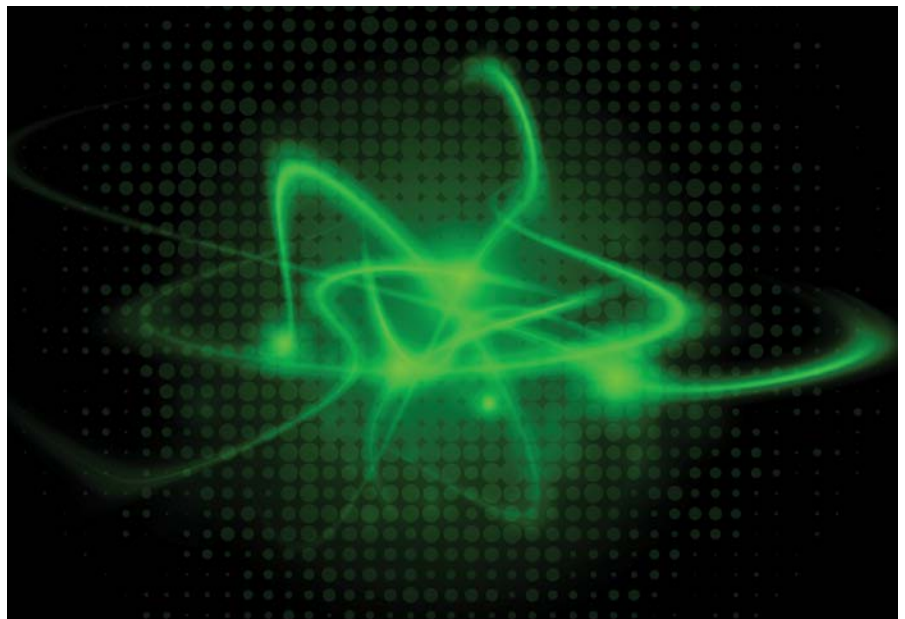
“As long as the discoverer gets treated better. I want my name on the Nobel.”

Ng snorted. “Yeah, Ray, but you have the advantage of being part of the patriarchy. When Jocelyn Bell discovered the first pulsar and her boss scooped the prize, she wasn’t so lucky.”

“It’s changing,” said Beach.

“Too slowly.”

“No,” Beach said, pointing at the screen. “Look.” After hundreds of cycles from one to sixteen, the pulses



were now displaying a pattern, anything but natural in appearance.

“Damn,” said Ng again, quieter this time. “Jocelyn and Hewish were joking when they called their source LGM for ‘Little Green Men.’ This is bound to have a natural explanation too, but it sure as hell looks like communication. Maybe there really is someone out there.” She bit at the flesh between her thumb and forefinger. “Get me the press office, Ray.”

“Are you sure? You know what they’ll do. However much we emphasize restraint, they’re going to pump out ‘Aliens contact Earth!’”

“It’s my job to make sure they don’t.”

8/11/27, 6:00 P.M.

When Beach got to his condo, his girlfriend Maya Karlsson was playing something on the PS6. He wasn’t sure what, but from the stream of in-

vective, she wasn’t winning. Seconds later, she tore off the headset and broke into a smile. “How long have you been there?”

“Not long. Have you got a minute? There’s something I’d like you to check out.”

“Sure. There’s only so much trashing I can take from teenagers.” Karlsson peered over Beach’s shoulder as he opened his laptop. “What’s this?”

“A signal we picked up today. See what you make of it.” Beach pulled up the recording and let it run from the point that the regular count changed to a more complex pattern.

Karlsson stared at the screen, her mouth slightly open. “Local interference? Picking up a broadcast?”

Beach grimaced. “Nope. It’s definitely from deep space. No clear source yet, but it’s for real.” [CONTINUED ON P. 103]



ACM BOOKS

Collection II

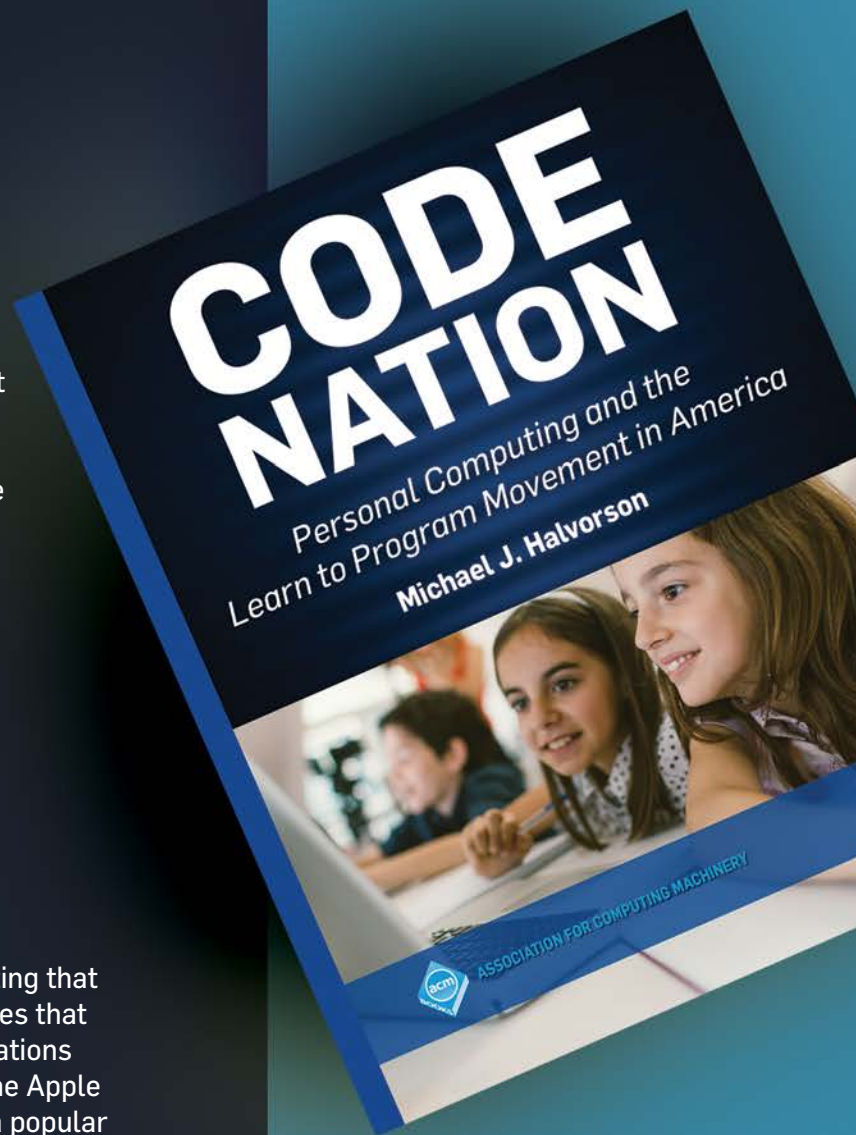
Code Nation explores the rise of software development as a social, cultural, and technical phenomenon in American history. The movement germinated in government and university labs during the 1950s, gained momentum through corporate and counterculture experiments in the 1960s and 1970s, and became a broad-based computer literacy movement in the 1980s. As personal computing came to the fore, learning to program was transformed by a groundswell of popular enthusiasm, exciting new platforms, and an array of commercial practices that have been further amplified by distributed computing and the Internet. The resulting society can be depicted as a “Code Nation”—a globally-connected world that is saturated with computer technology and enchanted by software and its creation.

Code Nation is a new history of personal computing that emphasizes the technical and business challenges that software developers faced when building applications for CP/M, MS-DOS, UNIX, Microsoft Windows, the Apple Macintosh, and other emerging platforms. It is a popular history of computing that explores the experiences of novice computer users, tinkerers, hackers, and power users, as well as the ideals and aspirations of leading computer scientists, engineers, educators, and entrepreneurs. Computer book and magazine publishers also played important, if overlooked, roles in the diffusion of new technical skills, and this book highlights their creative work and influence.

Code Nation offers a “behind-the-scenes” look at application and operating-system programming practices, the diversity of historic computer languages, the rise of user communities, early attempts to market PC software, and the origins of “enterprise” computing systems. Code samples and over 80 historic photographs support the text. The book concludes with an assessment of contemporary efforts to teach computational thinking to young people.

<http://books.acm.org>

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



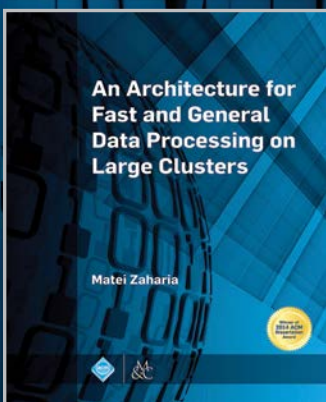
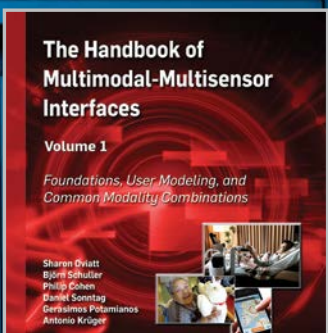
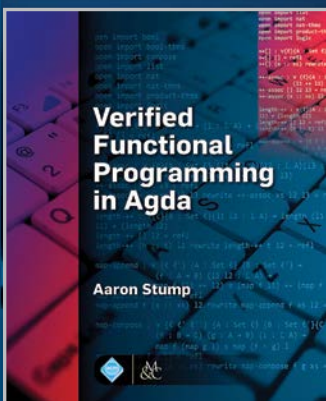
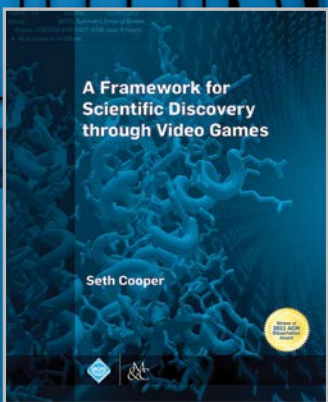
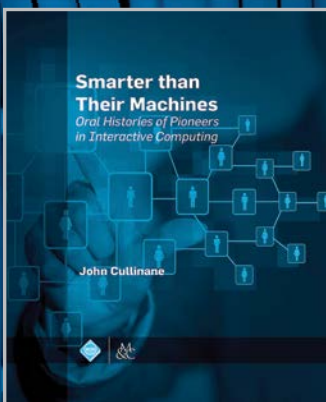
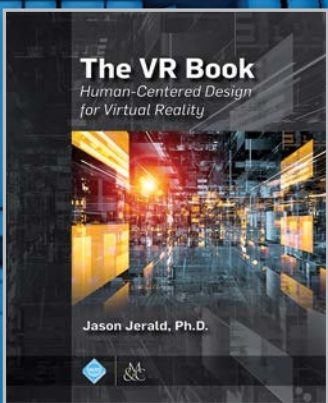
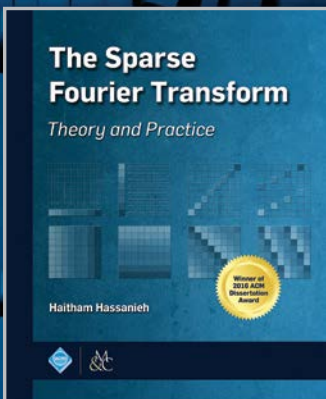
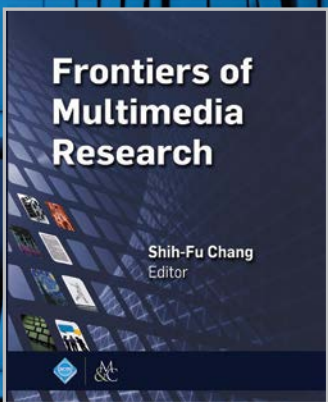
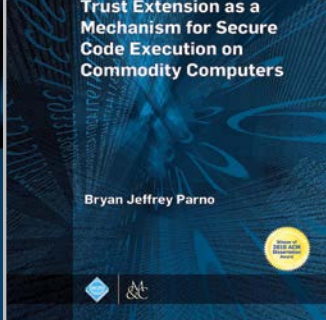
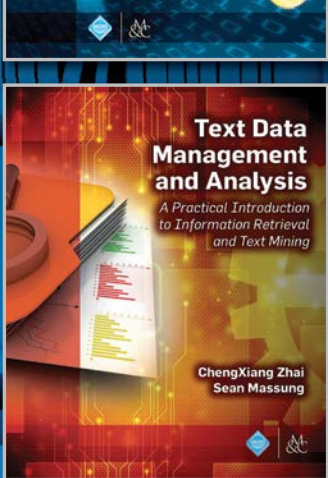
CODE NATION

*Personal Computing and
the Learn to Program
Movement in America*

Michael J. Halvorson

ISBN: 978-1-4503-7757-7

DOI: 10.1145/3368274



In-depth. Innovative. Insightful.

Inspired by the need for high-quality computer science publishing at the graduate, faculty, and professional levels, ACM Books are affordable, current, and comprehensive in scope.

**Full Collection | Title List
Now Available**

For more information, please visit
<http://books.acm.org>



Association for Computing Machinery
1601 Broadway, 10th Floor, New York, NY 10019-7434, USA
Phone: +1-212-626-0658 Email: acmbooks-info@acm.org