

ASPLOS 2015 Debate

Onur Mutlu

onur@cmu.edu

<http://users.ece.cmu.edu/~omutlu/>

March 17, 2014

Istanbul

Carnegie Mellon

Debate Statement

- *It's time: systems venues **should** require authors to make their code and data publicly available; those that do not will be held to **a higher standard**.*
- Clarification: in PLDI, for example, papers by PC members are "held to a higher standard". **The term doesn't have a strict definition.** I'm told, however, that it means that no weak rejects are allowed (e.g. AAAC will get rejected) and that AABB papers get occasionally rejected (A=accept, B=weak accept). In the discussion, the PC notes that the paper being discuss is written by a member, and they consider whether the higher bar criterion is met.

Debate Abstract

- **Abstract:** Most scientific disciplines take reproducibility of experimental results much more seriously than computer science. (See, for example, the policies concerning supporting artifacts—such as code and data—of the [Nature](#) and [Science](#) journals.) In this panel, two teams of highly-opinionated experts will debate whether it is time to adopt a similar policy in top-tier systems conferences and journals. **The idea is to change the review process such that papers that do not make their code and data available will be held to a higher standard when making the accept/reject decision, thereby incentivizing authors to share.** Attendees will be asked to vote whether they are in favor or against at the beginning and end of the panel.

My Bottom Line

- I like the **spirit** of sharing source and data with publications.
- **Everyone should want to do it.**
- Everyone should be **educated** to do it (perhaps over time and with other incentives).

- But ...

- I disagree with the “debate statement.”
- **Having to share** source code and data **should not be a requirement** for publication.
- Authors should not be **forced** to do it in our field.

Many Reasons for My Opposition

1. The main goal of publishing is to contribute insight (quickly)
2. Strict rules have unintended consequences
3. Double standards are a bad idea
4. Do we not want industry to publish papers?
5. There is great value to developing one's own infrastructure

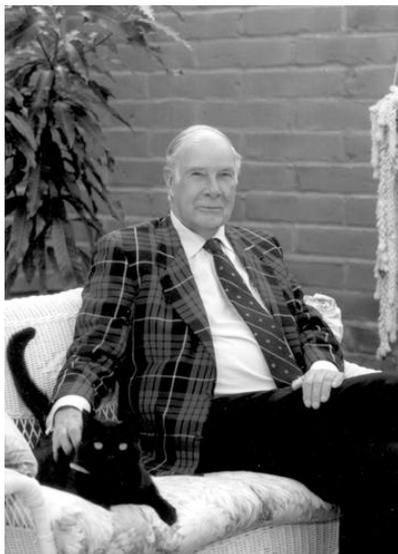
I will not have time to cover my suggestion for alternative models, but there are many:

encouragement, education, and rewards for publishing source code and data

without tying paper publication to source code publication

Some Basic Beliefs

- Research (in engineering) is a hunt for insight that can eventually impact the world; evaluation is secondary
 - “New insight” (not numbers) should be the bar for publication
 - “Evaluation craze” can hinder our ability to reach big insights



“The purpose of computing
is insight, not numbers”
Richard Hamming

What transfers is *insight*

Not academic design

Not performance numbers

Do the minimum analysis and
experimentation necessary to make a
point



Some Basic Beliefs

- Research (in engineering) is a hunt for insight that can eventually impact the world; evaluation is secondary
 - “New insight” (not numbers) should be the bar for publication
 - “Evaluation craze” can hinder the ability to reach big insights
- “Insight” is the most important determinant for publication
 - Any other criterion that goes against a purely merit based process will lead to unintended consequences and unfairness
- Industry publications are valuable in our field and we should make it easier (not harder) for industry to share
- Developing and exploiting good infrastructure should not be penalized by *requiring* authors to give it out right away
- **The proposed statement goes against all these basic beliefs**

1. Unintended Consequences

- Strict rules governing publication → unintended consequences
- What about authors who cannot make the source code and often times even absolute data available?
 - E.g., industry, government, industry-academia collaborations, ...
- Will they stop publishing?
- Will they be forced to hide even more such that they can publish with minimal exposure?
- Will they be forced to form their own conferences/tracks?
- What about industry/academia collaborative papers?
- Will we have to sign NDAs to see some source code and data?
- Will we have to deal with more and more bureaucracy?
- Add your own questions here...

2. Double Standards

a set of principles that applies differently and usually more rigorously to one group of people or circumstances than to another

- Requiring a higher bar for some papers over others is a terrible idea
- Our goal is to publish the works that can advance the field
- Any other criterion than merit leads to unintended consequences and some form of unfairness
- It could deter some people from publishing
- It could skew the balance of papers
- It could slow us down (collectively as a field) from reaching bigger and other new insights
- It's just bad for the spirit of fairness
- ...

My Favorite Double Standard Example

- PC Summary of a Rejected paper:

“The PC discussed this paper at length. The reviewers felt the solution was incremental with respect to the prior work, but that there was a contribution here. However, the fact that there was an overlap in the authorship of the two papers caused some concern, and **it was felt that the minimum research increment when the papers come from the same group should be higher than if they came from separate groups.**”

Rejected!

My Suggestion

- Do not allow double standards -- ever
- And, if you were not affected so far, you may be in the future



3. Industry Papers

- Requiring source code would deter industry from publishing
- It is already difficult for the industry to publish (for various reasons)
- Making it even harder would reduce progress in our field
- Subjecting industry papers to a higher bar would make it harder for industry to publish
- Could we make it work?
 - Not without unintended consequences

4. Freeloading

- Putting out source code may penalize researchers who take the pains to develop infrastructure to get to new insights
- Developing and exploiting good infrastructure should not be penalized by *requiring* authors to give it out right away
- A big idea may be barred from publication due to small versions of it being published (thx to the free infrastructure)
- Plus, there is great educational and research value in developing one's own infrastructure
- If you have lots of free options available, you may not want to develop your own infrastructure
- Thus, you may not be able to look at problems from a different angle
- And, you may actually inherit the bugs of other people!

That Said ...

SAFARI

SAFARI Research Group at Carnegie Mellon University

Site for source code and tools distribution from SAFARI Research Group at Carnegie Mellon University.

Ramulator: A Fast and Extensible DRAM Simulator

rowhammer

Source code for [Base-Delta-Immediate Compression:](#)

Memory tester for RowHammer. (Built on top of Memtest86+ v5.01.)

RowHammer is a new type of **memory failure** that is found only in recent generations of DRAM chips

Source code for [Mem-Sim](#)

Full data sets for [Adaptive-Latency DRAM:](#)

ASPLOS 2015 Debate

Onur Mutlu

onur@cmu.edu

<http://users.ece.cmu.edu/~omutlu/>

March 17, 2014

Istanbul

Carnegie Mellon

The Solution?

- A good reward system
- Positive reinforcement instead of negative reinforcement and double standards
 - Awards, education, culture change

