

Do We Need Robust Tools?

Igor Markov
U. Michigan



Back to Beta

David S

Gmail isn't the only one that can enjoy a BETA tag.

Enable

Disable

Incentives, Near-term & Long-term Issues

- Incentives in the greater software industry
- EDA tools
 - Basic tools vs. novelty
 - In-house vs. commercial tools
- Incentives for researchers
 - Plural (“anecdote”) \neq data
 - Robust tools in academic research?
- What can be done ...



Incentives for Commercial Software

- **Note:** Microsoft draws most of its revenues from Windows and Office sales
 - New versions - additional sales



- Why do users buy Microsoft software ?
 - It is bundled with new computers
 - It supports a great variety of hardware
 - Standards, inertia

Incentives for Commercial Software

- **SW cost vs. hardware vs. entire budget**
 - German Govt is migrating to Linux, OpenOffice
 - US Military is buying a lot of expensive SW, still use Windows
- **UI, ease of use, integration & aesthetic value trump robustness, features & cost**
 - Windows vs Linux
 - Apple vs Microsoft



Think different.



Novelty Items

- Basic OS and Office SW are *good enough*
- iPhone apps – *novelty*



Incentives for EDA Tools?

- Novelty items – the most interesting segment
 - New features, cost
 - Performance on benchmarks



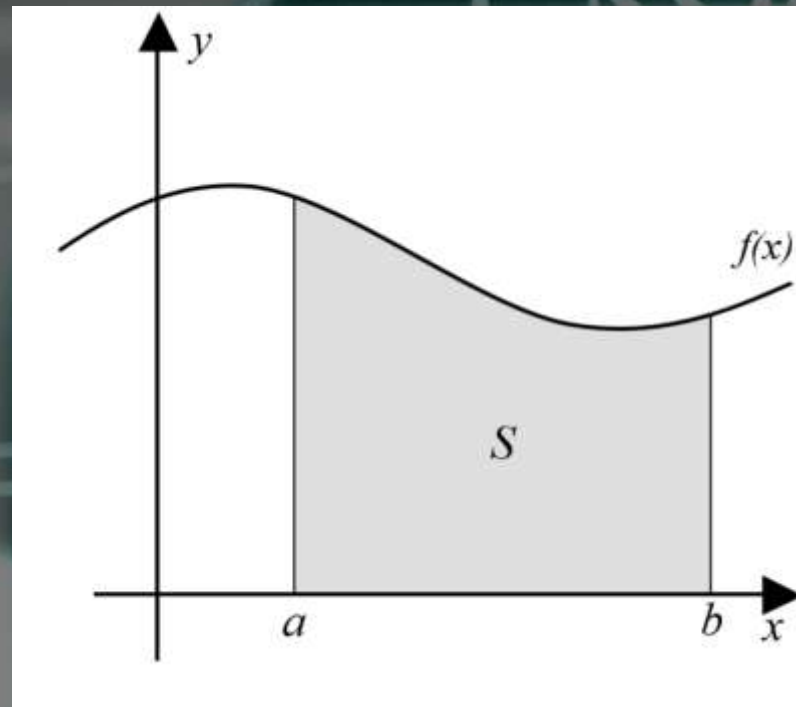
Performance Metrics in EDA

- Fundamental reasons why no EDA tool will be ideal for all possible inputs
 - Many EDA problems are NP-hard or worse
 - It's all about trade-offs: capture practical aspects of the problem that are solvable



Ideal Metrics for EDA Tool Users

- The integral of performance over all inputs that will be given to the tool



- Instantaneous execution

Ideal Metrics for Established EDA Vendors

- Beat the competition on key benchmarks
 - Not too much, say 5-10%
- Run just a little faster
 - Say 20%
- Keep on improving runtime & performance every year
 - Watch out for start-ups



Cost & Availability of Tech Support

FPGA tool users
“fire and forget”

ASIC tool users



TECH SUPPORT

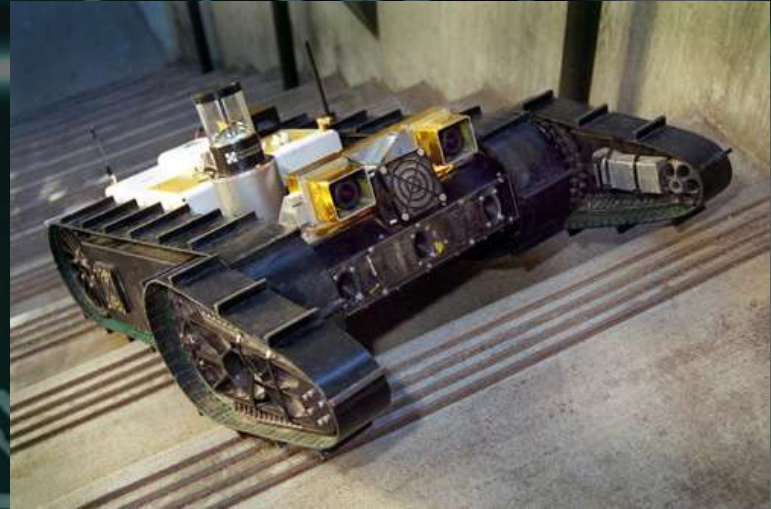
WE CAN'T FIX STUPID. BUT AT LEAST
YOU CAN STILL BLAME IT ON OUR PRODUCT.

Human in the Loop?

Intel and IBM
(CAD)



EDA industry



Robust Tools in the Industry?

- FPGA tools really need to be robust
 - More clients than tech-support staffers
 - Hardware purchases depend on them



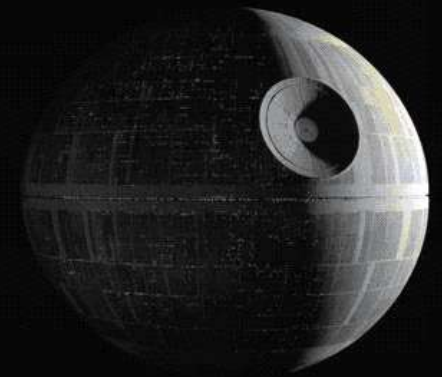
Robust Tools in the Industry?

- In-house ASIC tools (IBM, Intel) need to be *configurable* and *controllable*



Robust Tools in the Industry?

- Commercial ASIC tools need to be only somewhat robust (to sell upgrades) AND ONE OF
 1. Bundled, integrated or trusted
 2. Solve a new problem
 3. Give unquestionably better QOR



TOO BIG TO FAIL

Robust EDA Tools in the Academia?

- Ideal goals for academics
 - Discover new knowledge, useful algorithms
 - Develop techniques for new EDA problems
 - Develop *much better* techniques for known problems
- Tangible goals
 - Funding
 - Publications
- Proxies
 - Benchmarks & software
 - Student contests

A Problem with Relevance

- Academic research optimized for DOC, PPT
 - Entire lines of research shown bogus or wrong



FACEPALM

When words fail to describe the dismay, there is always Facepalm.

Examples

- **Multilevel routing 2002-2006**
 - Seemed like the most advanced algorithm for global routing, until ISPD routing contests
 - Turns out the authors didn't know about negotiated-congestion routing
- **Crosslink insertion in clock trees 2004-2009**
 - None of these algorithms worked in the ISPD '09 & '10 clock-network contests
 - Experiments didn't account for optimized trees

Step 1: Fix Benchmarks

- Heated discussions & much progress in the late 1990s and early 2000s
 - Complaints about insufficient benchmarks
 - Demonstrations of 10x difference in results due to wrong units
 - New benchmark releases
 - Comparisons to commercial tools
 - Software releases
 - Open-source software



Step 2: Dig into Tools

- From 2004, a number of new placement tools have shown increasingly better results
 - Some have been available in binary
 - None in source code
- Replication from papers may be impossible
 - Aplace has been replicated (in Taiwan)
 - mPl6: each mPl paper used different algorithms
 - FastPlace1-3 resisted many efforts
 - KraftWerk2 is replicable (with significant effort), but needs benchmark-specific tuning

Step 3 ...?

- A peek into source codes of 3 winning routers at the ISPD`08 contest
 - Everyone tunes to individual benchmarks
 - ICCAD`08 papers report “very fast runtimes” with a straight face, while limiting #iterations for each benchmark differently
- ISPD`09 contest used hidden benchmarks
 - Many teams failed on harder benchmarks
 - But their conf. papers report 2x better results
 - Failed again in ISPD`10 contest

A Problem with Student Training

- The industry is salivating over fresh Ph.D.s who can develop robust tools
- The academic environment & *some industry experts* discourage such focus



Example: Obstacle-Avoiding Min-Length Steiner Trees

- DAC, ICCAD and ISPD for the last few years accepted many papers on this topic
 - Elegant geometric algorithms
 - Every paper claims a slight improvement on “industry” nets with 100s pins
- **None of these papers discuss applications**
 - % improvement for the entire netlist very small
 - Not useful in global routing !
 - Timing is not considered

Example: Multicore Programming

- Major challenge for the industry
- Funding agencies went bezerk: “multicore” is now a *de facto* requirement for funding

- A flurry of publications, follow the same template

- Take a *really slow*, uncompetitive algorithm
- Run it in parallel or on a GPUs
- Show a speed-up, ignore Amdahl’s law





DOUBLE FACEPALM

FOR WHEN ONE FACEPALM DOESN'T CUT IT

The Winner's Curse

- The more remarkable the claims, the more likely they are exaggerated
 - Documented in biomedical literature
 - Proven for auctions with incomplete information

Try Beta  Log in / create

article

discussion

edit this page

history

Winner's curse

From Wikipedia, the free encyclopedia

The **winner's curse** is a phenomenon akin to a [Pyrrhic victory](#) that occurs in [common value auctions](#) with [incomplete information](#). In short, the winner's curse says that in such an auction, the winner will tend to overpay. The winner may overpay or be 'cursed' in one of two ways: 1) the winning bid exceeds the true value of the auctioned asset such that the winner is worse off in absolute terms; or 2) the value of the asset is less than the bidder anticipated, so the winner may still have a net gain but will be worse off than anticipated.^[1] However, an actual overpayment will generally occur only if the winner fails to account for the winner's curse when bidding (an outcome that, according to the [Revenue Equivalence Theorem](#), need never occur). So despite its dire-sounding name, the winner's curse does not necessarily have ill effects in practice.

Contents [hide]

1 Related uses

2 Explanation

What Can Be Done ?

- Change incentives for academic research from PPT & DOC to *robust tools*
 - Or else deal with bogus results
(dramatic parallel speed-ups, improved QOR)



HONESTY

"When someone is lying, is it true that their pants are actually on fire?"

JERRY

Post-Funding vs. Pre-Funding

- Funding is now allocated based on proposals
 - @NSF results do not matter
 - @SRC results are reports & presentations
- Allocate at least some funding based on actual results, robust tools (may need to increase the overall \$\$ pot)



Do Results-based Incentives Work in Practice ?

- In several ISPD contests, the #1 team was the last team the previous year (!)
 - Multi-year strategy is less important than one may think
 - Barging into a new field is a good thing (for the researchers and for the field)
 - A surprising amount of development may be done in 2-3 months
- The only way to debunk status quo

Make Industry Reviewing (DAC, ICCAD, SRC) Less Fickle

- Knee-jerk reaction to practical research
“this is not how it is done”
- Like abstract / mathematical papers, w/o path to applications
 - Greedy works better in practice!
- Memory-less reviewing



What Else Can Be Done ?

- Open up industry to benchmarking of basic optimizations
 - IBM, Intel & TI are doing this
 - Many EDA development teams do not know where they stand



Conclusions: Do We Need Robust Tools?

- Industry mostly needs
 - cool new apps
 - *the ability to develop robust tools* when pressed by the competition
 - people who can develop robust tools
- US academia has little incentive to work on robust tools
 - Flaky tools can justify any “novel” idea
 - Things clear up a few years later
- Taiwan understands this very well

