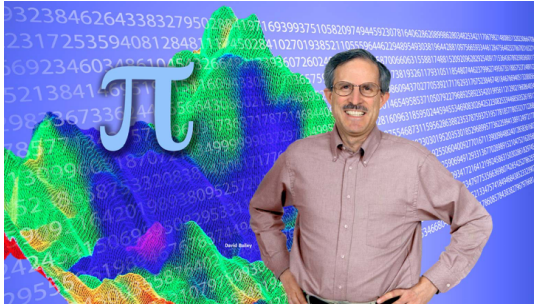
Quantum computer performance: Standards now or scandals later?

David H. Bailey
http://www.davidhbailey.com
Lawrence Berkeley National Lab (retired) and Univ. of California, Davis
This talk: http://www.davidhbailey.com/dhbtalks/dhb-dwave-2019.pdf



Reproducibility crises in psychology and economics

- In August 2015, the Reproducibility Project in Virginia reported that they were able to reproduce only 39 of 100 psychology studies.
- In November 2018, a separate consortium of researchers was above to reproduce only 15 of 28 psychology studies.
- In September 2015, the U.S. Federal Reserve was able to reproduce only 29 of 67 economics studies.

Reproducibility Project staff

Credit: NY Times

1. A. P. Taylor, "Half the time, psychology results not reproducible," *The Scientist*, 20 Nov 2018, https://www.the-scientist.com/news-opinion/half-the-time--psychology-results-not-reproducible--study-65117.
2. A. C. Chang and P. Li, "Is economics research replicable? Sixty published papers from thirteen journals Say 'Usually Not'," *U.S. Federal Reserve*, Washington, 2015, https://www.federalreserve.gov/econresdata/feds/2015/files/2015083pap.pdf.

Reproducibility crises in the social sciences, cont.

The "blank slate" paradigm (1920–1990):

- The human mind at birth is a *tabula rasa* ("blank slate").
- Heredity and biology play no significant role in human psychology; all personality and behavioral traits are socially constructed.
- Pre-modern societies were peaceful, devoid of psychological and social problems.

Current consensus, based on latest research in behavioral genetics and anthropology:

- Humans at birth possess sophisticated facilities for social interaction, language acquisition, pattern recognition, navigation and counting.
- Heredity, evolution and biology are major factors in human personality.
- Some personality traits are more than 50% heritable.
- Pre-modern societies had more crime, war and social problems than today.

How did the 20th century social scientists get it so wrong?

- Sloppy experimental methodology, with little concern for reproducibility.
- Ignoring or dismissing data that runs counter to predisposition.

3. S. Pinker, *The Blank Slate: The Modern Denial of Human Nature*, Penguin Books, 2003.

Reproducibility crises in biomedicine

The biomedical field has been stung by numerous cases where pharma products look good based on clinical trials, but later disappoint in real-world usage, or the results cannot be reproduced in separate studies. Examples:

- In 2004, GlaxoSmithKline acknowledged that while some trials of Paxil found it effective for depression in children, other unpublished studies showed no benefit.
- In 2011, Bayer researchers reported that they were able to reproduce the results of only 17 of 67 published studies they examined.
- In 2012, Amgen researchers reported that they were able to reproduce the results of only 6 of 53 published cancer studies.
- In 2014, a review of Tamiflu found that while it made flu symptoms disappear a bit sooner, it did not stop serious complications or keep people out of the hospital.

Only publicizing the results of successful trials introduces a bias into the results.

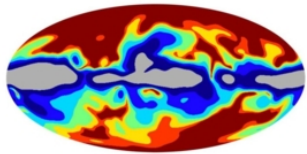
The AllTrials movement requires all results to be public: `http://www.alltrials.net`.

Reproducibility crises in biomedicine (references for previous page)

4. S. Foley, "GlaxoSmithKline pays $3bn for illegally marketing depression drug," *U. K. Independent*, 3 Jul 2012, http://www.independent.co.uk/news/business/news/glaxosmithkline-pays-3bn-for-illegally-marketing-depression-drug-7904555.html.
5. E. S. Reich, "Cancer trial errors revealed," *Nature*, 11 Jan 2011, http://www.nature.com/news/2011/110111/full/469139a.html.
6. F. Prinz, T. Schlange and K. Asadullah, "Believe it or not: How much can we rely on published data on potential drug targets?," *Nature Reviews Drug Discovery*, vol. 10 (Sep 2011), pg. 712, https://www.nature.com/articles/nrd3439-c1.
7. C. G. Begley and L. M. Ellis, "Drug development: Raise standards for preclinical cancer research," *Nature*, vol. 483 (29 Mar 2012), pg. 531-533, https://www.nature.com/articles/483531a.
8. B. Goldacre, "What the Tamiflu saga tells us about drug trials and big pharma," *U. K. Guardian*, 9 Apr 2014, http://www.theguardian.com/business/2014/apr/10/tamiflu-saga-drug-trials-big-pharma.

Reproducibility crises in physics

In March 2014, the BICEP2 team announced that they had discovered a "twisting" pattern in cosmic microwave background data, which fits the most common hypothesized model of the inflation era just after big bang.



Press reports trumpeted this result as the first experimental evidence of the inflationary big bang.

But other researchers had difficulty reconstructing the claimed results. Finally, two teams challenged the BICEP2 findings, saying that the results could more readily be explained by dust in the Milky Way.

Now the consensus is that the detection was false.

9. Ron Cowen, "Doubt grows about gravitational waves detection," *Scientific American*, 2 Jun 2014, https://www.scientificamerican.com/article/doubt-grows-about-gravitational-waves-detection/.

Reproducibility crises in finance

Finance has been stung with many instances of investment strategies that look great on paper, but fall flat in practice. A primary cause is backtest overfitting – statistical overfitting of historical market data.

When a computer can analyze thousands or millions of variations of a given strategy or fund design, it is almost certain that the best such strategy, measured by backtests, will be overfit and thus of dubious value.

In two 2014 papers by myself and colleagues, we show that a broad range of financial strategies and fund designs are compromised by backtest overfitting.

10. D. H. Bailey, J. M. Borwein, M. Lopez de Prado and Q. J. Zhu, "Pseudo-mathematics and financial charlatanism: The effects of backtest over fitting on out-of-sample performance," *Notices of the American Mathematical Society*, May 2014, pg. 458–471, http://ssrn.com/abstract=2308659.
11. D. H. Bailey, J. M. Borwein, M. Lopez de Prado and Q. J. Zhu, "The probability of backtest overfitting," 12 Feb 2014, http://ssrn.com/abstract=2326253.

Email exchange between DHB and a finance colleague

Email from DHB to finance colleague, 10 June 2013:

One thing that has always puzzled me about the financial world is the following sort of thing: [press examples cited]. Excuse me for being "dumb," but this sort of thing seems to me to be outright nonsense. ... When people like those above say that they "know" where the stock market is heading, this cannot have any scientific basis. ...

So why doesn't somebody blow this whistle on this sort of thing? Am I missing something?

Response from finance colleague to DHB, 17 June 2013:

It is not a dumb question at all. It is a question I have struggled with and which answer makes me an unhappy man. The truth is, most people in this industry are charlatans. They do not have any particular model or theory to understand the world. They are not scientists. ...

I completely agree with your assessment. The amount of nonsense ... is incredible.

Financial news commentary

Excerpt from a recent article on a widely read financial news site:

> *As I highlighted during the past week, the bottom we struck this past week seems to best count as a third wave bottom in the (c) wave of the a-wave. That still has me looking for a lower low yet to come. Moreover, there were no divergences on any of the technicals present on the 60-minute chart when we struck that bottom. The great majority of the time, that suggests that only the third wave of the (c) wave has completed. For this reason, I was looking for this bounce to be a fourth wave, which we began to anticipate when we identified the bottoming in the market in our chat room on Tuesday night.*

Why the silence in the mathematical finance community?

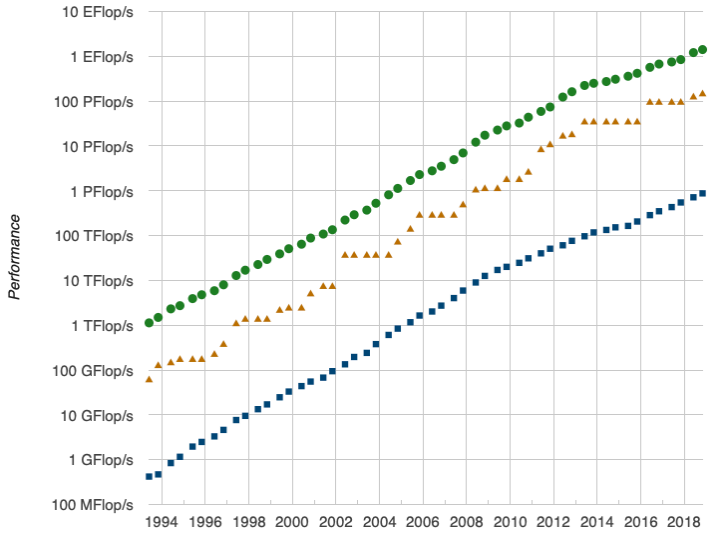
Historically scientists have led the way in exposing those who utilize pseudoscience to extract a commercial benefit.

Yet financial mathematicians in the 21st century have remained disappointingly silent with the regards to those in the community who, knowingly or not:

1. Fail to disclose the number of models or variations that were used to develop an investment strategy or fund.
2. Make vague predictions that do not permit rigorous testing and falsification.
3. Misuse probability theory, statistics and stochastic calculus.
4. Suggest in countless press reports and promotions that investors can achieve above-market returns via unsophisticated products and chart-watching strategies.
5. Use dubious technical jargon: "stochastic oscillators," "Fibonacci ratios," "cycles," "Elliott waves," "golden ratio," "parabolic SAR," "pivot point," "momentum," etc.

Message to community: Our silence is consent, making us accomplices in these abuses.

Increasing performance of the top 500 supercomputers (1994 – present)



Orange = #1
Blue = #500
Green = sum #1 thru #500

Courtesy: Top500.org

DANGER AHEAD

Supercomputers can generate nonsense faster than ever before!

Are computational results reproducible?

- Are the algorithms, data sources and processing methods well documented?
- Are all source code files, make files and data files available in a secure public repository?
- Are the results statistically sound?
- Are the results numerically reliable?
- Have the results been validated using separate tests?
- Have the results been validated by independent researchers?

Reproducibility in scientific computing

A December 2012 workshop on reproducibility in computing, held at Brown University in Rhode Island, U.S.A., found that

> *Science is built upon the foundations of theory and experiment validated and improved through open, transparent communication. With the increasingly central role of computation in scientific discovery, this means communicating all details of the computations needed for others to replicate the experiment.*
> ...
> *The "reproducible research" movement recognizes that traditional scientific research and publication practices now fall short of this ideal, and encourages all those involved in the production of computational science ... to facilitate and practice really reproducible research.*

12. V. Stodden, D. H. Bailey, J. Borwein, R. J. LeVeque, W. Rider and W. Stein, "Setting the default to reproducible: Reproducibility in computational and experimental mathematics," Jan 2013, http://www.davidhbailey.com/dhbpapers/icerm-report.pdf.

Reproducibility in scientific computing, continued

Issues identified in the ICERM report and other studies include:

- Researchers must carefully document the full context of computational experiments—system environment, input data, code used, computed results, etc.
- Researchers must save the code and output data in a permanent repository.
- Reviewers, research institutions and funding agencies need to recognize the importance of computing and computing professionals, and to allocate funding for after-the-grant support and repositories.
- Researchers need to be more careful with numerical reproducibility.
- The community must encourage the publication of negative results — other researchers can often learn from them.
- The community must ensure the responsible reporting of performance.

Numerical reproducibility in high-performance computing

The report mentioned above on reproducibility in high-performance computing noted:

> *Numerical round-off error and numerical differences are greatly magnified as computational simulations are scaled up to run on highly parallel systems. As a result, it is increasingly difficult to determine whether a code has been correctly ported to a new system, because computational results quickly diverge from standard benchmark cases. And it is doubly difficult for other researchers, using independently written codes and distinct computer systems, to reproduce published results.*

Analysis of collisions at the Large Hadron Collider

- The 2012 discovery of the Higgs boson at the ATLAS experiment in the LHC relied crucially on the ability to track charged particles with exquisite precision (10 microns over a 10m length) and high reliability (over 99% of roughly 1000 charged particles per collision correctly identified).
- Software: 5 millions line of C++ and python code, developed by roughly 2000 physicists and engineers over 15 years.
- Recently, in an attempt to speed up the calculation, researchers found that merely changing the underlying math library resulted in some collisions being missed or misidentified.

Questions:

- How serious are these numerical difficulties?
- How can they be tracked down?
- How can the library be maintained, producing numerically reliable results?

Are performance reports reproducible?

- Are the algorithms, data sources, and complete system configuration fully documented?
- Are all source code files, make files and data files available in a secure public repository?
- Is the exact benchmark being tested fully defined?
- Is the performance comparison a fair, apples-to-apples comparison, with comparable levels of precision, accuracy, tuning and analysis?
- Are the performance results statistically sound (e.g., are only the most favorable results reported)?
- Have the performance results been validated using separate tests?
- Have the performance results been validated by independent researchers?

The same standards apply for performance results as for other scientific results.

Reproducibility crisis in HPC performance: 1990-1994

Background:

- Many new parallel systems had been introduced; each claimed theirs was best.
- Many researchers were excited about the potential of highly parallel systems.
- Few standard benchmarks and testing methodologies had been established.
- It was hard to reproduce published performance results; much confusion reigned.
- Overall, the level of rigor and peer review in the field was rather low.

In response, DHB published a humorous essay "Twelve ways to fool the masses when giving performance results on parallel computers," poking fun at some of the abuses.

Since abuses continued, DHB presented a talk at Supercomputing 1992 and published a paper with specific examples culled from peer-reviewed papers.

1991 paper: "Twelve ways to fool the masses"

1. Quote 32-bit performance results, not 64-bit results, but don't mention this in paper.
2. Present performance figures for an inner kernel, then represent these figures as the performance of the entire application.
3. Quietly employ assembly code and other low-level language constructs.
4. Scale up the problem size with the number of processors, but omit any mention of this.
5. Quote performance results projected to a full system.
6. Compare your results against scalar, unoptimized code on conventional systems.
7. When run times are compared, compare with an old code on an obsolete system.
8. Base Mflop/s rates on the operation count of the parallel implementation, instead of the best practical serial algorithm.
9. Quote performance as processor utilization, parallel speedups or Mflop/s per dollar.
10. Mutilate the algorithm used in the parallel implementation to match the architecture.
11. Measure parallel run times on a dedicated system, but measure conventional run times in a busy environment.
12. If all else fails, show pretty pictures and animated videos, and don't discuss performance.

NY Times (22 Sep 1991): “Measuring How Fast Computers Really Are”

Excerpts:

- “Rival supercomputer and work station manufacturers are prone to hype, choosing the performance figures that make their own machines look best.”
- “It’s like the Wild West.” [quoting David J. Kuck of UIUC].
- “It’s not really to the point of widespread fraud, but if people aren’t a little more circumspect, the entire field could start to get a bad name.” [quoting DHB].

14 F THE NEW YORK TIMES, SUNDAY, SEPTEMBER 22, 1991

Technology

Measuring How Fast Computers Really Are

By JOHN MARKOFF

IN the world of scientific and technical computing, everyone agrees that computer speeds are increasing at a geometric rate. But measuring that speed is a vexing task. Rival supercomputer and work station manufacturers are prone to hype, choosing the performance figures that make their own machines look best.

"It's like the Wild West," said David J. Kuck, of the Center for Supercomputing Research and Development at the University of Illinois. "They say whatever they want to."

In fact, said David H. Bailey, a scientist at the National Aeronautics and Space Administration, "It's not really to the point of widespread fraud, but if people aren't a little more circumspect, the entire field could start to get a bad name."

The matter is complicated by a new generation of computers that have dozens, or even thousands, of separate processors. These parallel computers split problems into small parts and solve them simultaneously to reach greater speeds.

As a result, dozens of programs for determining benchmarks — measurements of computer speed — have been developed by scientists at universities and in government agencies. Some are based on how long a computer takes to solve a certain set of equations, while more sophisticated benchmarks attempt to match the operations required by real-world programs. But each benchmark generally measures only a single aspect of computer performance.

Just as a car buyer might buy a vehicle with the highest E.P.A. gas mileage rating for the price, a computer buyer could use benchmarks in deciding which machine to buy. But like their counterparts in the auto business, computer makers would do well to remind customers, "Your mileage may vary." The industry has no independent organization, analogous to the Environmental Protection Agency, to establish a single standard.

The proliferation of benchmarks is particularly problematic among the fastest scientific machines, where more than a dozen start-up companies compete to sell to university, corporate and Government laboratories.

These machines sell for hundreds of thousands of dollars or more, and the sale of only a few can mean success for a company. Supercomputers and smaller scientific work stations work on problems ranging from designing pharmaceuticals and weapons to weather modeling and the simulated crashing of automobiles.

Uneasy about the tendency for manufacturers to cite inflated claims, Mr. Bailey of NASA wrote a tongue-in-cheek indictment of performance claims for Supercomputing Review magazine in August. Titled "Twelve Ways to Fool the Masses When Giving Performance Results on Parallel Computers," it pokes fun at the tendency of computer makers to play fast and loose with speed claims.

It is common practice to "tune" computers and software to score better on benchmarks. "I know of a couple of companies who have full-time people, and all they do is optimize programs to achieve better benchmark results," said Gary Smaby, president of the Smaby Group, a consulting and market research firm in Minneapolis.

Such optimization is permissible under the rules established by benchmark designers to insure that computer makers can extract the full capability from their systems.

But some manufacturers go further and insert modules called "recognizers" into their compilers — software that translates a programmer's instructions into machine-readable form. When the recognizer finds a clue that points to a benchmark, it employs a specially written program that is designed to run a benchmark as quickly as possible.

Computer scientists say recognizers can be used to short-circuit the benchmark-measuring process and should be banned.

Today the most widely cited benchmark is Linpack. It is a set of linear algebra equations that requires a very large number of mathematical calculations to solve. Currently, the highest Linpack by a traditional supercomputer, measured in millions of mathematical calculations per second, is held by a Cray Y/MP. Its record: 403 million floating point calculations per second.

There is a similar Linpack measure for makers of massively parallel computers, and this summer the lead seesawed back and forth between Intel's Delta supercomputer and Thinking Machines' CM-200 Connection Machine. At the moment the Intel Delta has achieved the highest Linpack measure, 11.9 billion floating point calculations per second.

The Linpack benchmark results are expressed as several numbers, including the actual speed at solving the equations and another number called "theoretical peak performance," which represents a computer's absolute peak efficiency. In practice, computers generally reach only a small fraction of their theoretical peak speed.

Linpack was developed as part of an appendix to a book written by a University of Tennessee computer scientist, Jack Dongarra, in 1979. "The original benchmark was an accident," he said. "We were writing a book and were attempting to give the user a feeling for how fast particular computers could solve a problems."

The problem, Mr. Dongarra said, was chosen so that it could be run on all of the 20 machines that were being compared. There are now Linpack numbers for more than 500 machines, from laptops to supercomputers.

Mr. Dongarra acknowledges that Linpack is a limited measure, and he is working on a second-generation measure named Lapack. But a number of other researchers have developed alternative benchmarks in an effort to provide a more complete profile of computer performance.

One recently introduced benchmark is called Slalom, developed by John Gustafson, a mathematician at the Department of Energy's Ames Laboratory. "I worked in industry for several years, and I was frustrated by inaccuracy and general lack of credibility in the ways we had of measuring computer performance," he said.

Rather than measuring how quickly a computer can solve a particular problem, Slalom attempts to measure how much work a computer can complete in one minute.

The problem computers must calculate as part of the Slalom benchmark is measuring the interplay of light in a room in which each surface is painted a different color. A particularly powerful computer will obtain higher resolution in one minute of computing time than a slower one.

One strength of Slalom is that it measures several aspects of computer performance, including the time taken to set up a problem. Hightest on the Slalom scale is the experimental Intel Delta supercomputer, installed at the California Institute of Technology.

ANOTHER approach has been taken by scientists at the Center for Supercomputing Research and Development at the University of Illinois. The group, called the Perfect Club, has composed a set of speed tests it calls the Perfect Benchmarks, a set of programs that are closely patterned after real programs run by scientific computers.

These benchmarks attempt to get as close as possible to the ultimate test of a computer's performance — the actual programs that it will run. "The only important benchmark for most customers is the code they intend to run, and there is no substitute for that," Mr. Smaby said. ■

Different Benchmarks, Different Winners

The six fastest computers according to various benchmarks. The Linpack benchmark expresses the results in Mflops, or millions of floating point operations per second. Slalom, the only one of these to cover massively parallel machines, measures the amount of work accomplished in a set amount of time. The Perfect benchmark is expressed as an average of 13 different Mflop measures.

LINPACK		SLALOM		PERFECT	
Cray Y-MP/16	403	Intel Delta	5,700	Cray Y-MP/832	128.4
NEC SX-3/14	314	Siemens S600/20	5,610	Cray Y-MP/416	75.9
Cray Y-MP/832	275	Cray Y-MP/8D	5,120	Siemens S400/10	26.2
Fujitsu VP2600/10	249	Cray 2S/4	4,204	Cray 2S/4-128	22.5
Cray X-MP/416	178	nCUBE 2	3,736	NEC SX-2	18.4*
Cray 2S/4-128	129	Fujitsu VP400-EX	2,598	Hitachi S-820/80	17.1*

Sources: Oak Ridge National Laboratory, Supercomputing Review, University of Illinois, University of Tennessee

* Computed on partial data as some data points missing.

The New York Times

Example 1: Scaling performance results to full-sized system

In some published papers and conference presentations, performance results on small-sized parallel systems were linearly scaled to full-sized systems, *often without even clearly disclosing this fact*.

Example: 8,192-CPU performance results were linearly scaled to 65,536-CPU results, simply by multiplying by eight.

Excuse: "We can't afford a full-sized system."

This and the other examples mentioned in the next few viewgraphs are presented in:

13. D. H. Bailey, "Misleading performance reporting in the supercomputing field," *Scientific Programming*, vol. 1., no. 2 (Winter 1992), pg. 141–151, https://www.davidhbailey.com/dhbpapers/mislead.pdf.

Example 2: Using inefficient algorithms on highly parallel systems

In many cases, inefficient algorithms were employed for the highly parallel implementation, requiring many more operations, thus producing artificially high Mflop/s rates:

- Numerous researchers cited parallel PDE performance based on explicit schemes, where implicit schemes were known to be much more efficient.
  Excuse: Explicit schemes "run better" on the researchers' parallel system.
- One paper cited performance for computing a 3D discrete Fourier transform by direct evaluation of the defining formula ($8n^2$ operations), rather than by using a fast Fourier transform ($5n \log_2 n$).
  Excuse: Direct computation of FFT was "more appropriate" for the architecture being analyzed.

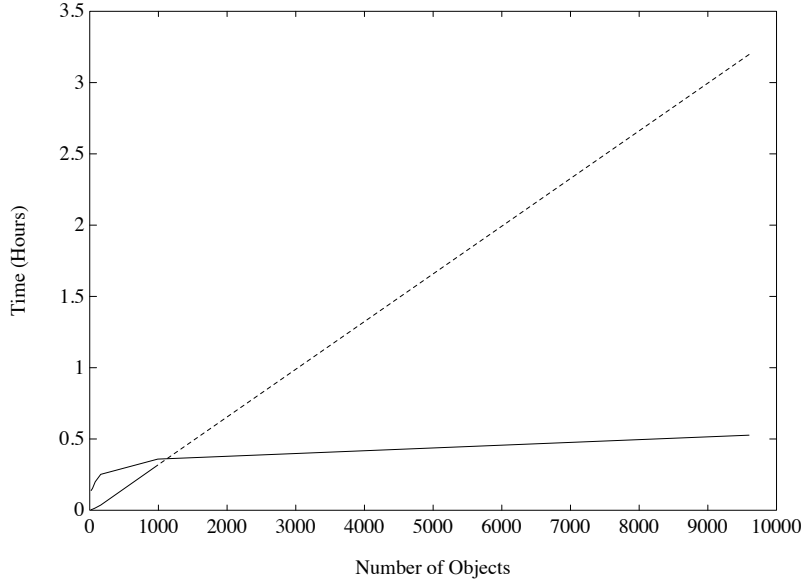
Both examples violate a rule of professional performance reporting, namely to base the operation count (when computing Mflop/s or Gflop/s rates) on the *best practical serial algorithm*, no matter what scheme was actually used on the parallel system.

Example 3: Not actually performing a computation on the claimed system

Abstract of published paper: "The current Connection Machine implementation runs at 300-800 Mflop/s on a full [64K] CM-2, or at the speed of a single processor of a Cray-2 on 1/4 of a CM-2."

- Excerpt from text: "This computation requires 568 iterations (taking 272 seconds) on a 16K Connection Machine."
  In other words, the computation was run on a 16K system, not on a 64K system; the figures cited in the Abstract were merely multiplied by four.
- Excerpt from text: "In contrast, a Convex C210 requires 909 seconds to compute this example. Experience indicates that for a wide range of problems, a C210 is about 1/4 the speed of a single processor Cray-2."
  In other words, the computation mentioned in the Abstract was not actually run on a Cray-2; instead, it was run on a Convex system, and a questionable rule-of-thumb scaling factor was used to produce the Cray-2 rate.

Example 4: Performance plot — parallel (lower) vs vector (upper)



Data for performance plot

Problem size (x axis)	Parallel system run time	Vector system run time
20	8:18	0:16
40	9:11	0:26
80	11:59	0:57
160	15:07	2:11
990	21:32	19:00
9600	31:36	3:11:50*

Details in text of paper:

- In last entry, the 3:11:50 figure is an "estimate."
- The vector system code is "not optimized."

Note that the parallel system is actually slower than the vector system for all cases, except for the last (estimated) entry. Except for the last entry, all real data in the graph is in the lower left corner (i.e., a log-log plot should have been used instead). Also, it is not fair to compared tuned vs. untuned performance.

Graveyard of failed HPC firms in 1990s

- Thinking Machines, Inc.: Founded 1983; popular in early 1990s; bankrupt 1994.
- Intel: Marketed Paragon system in late 1980s and early 1990s; exited 1994.
- Kendall Square Research: Founded 1986; bankrupt 1994.
- Convex Computers: Founded 1982; popular in early 1990s; sold to Hewlett-Packard in 1995.
- Cray Computer Corp.: Founded 1989 by Seymour Cray; bankrupt 1995.
- Cray Research, Inc.: Founded in 1970s by Seymour Cray; industry leader in 1970s and 1980s; declined in early 1990s; sold to SGI in 1995; then sold to Tera Computers in 2000, which changed name back to Cray; still in HPC market today.

Fast forward to 2019: Fooling the masses with multicore/GPU systems

- Cite performance rates for a run with only one processor core active in a shared-memory multi-core node, producing artificially inflated performance (since there is no shared memory interference) and wasting resources (since most cores are idle).
  - Example: Cite performance on "1024 cores," even though the code was run on 1024 multicore nodes, one core per node, with 15 out of 16 cores idle on each node.
- Claim that since one is using a graphics processing unit (GPU) system, that efficient parallel algorithms must be discarded in favor of "more basic algorithms."
- Cite performance rates only for a core algorithm (such as FFT or linear system solution), even though full-scale applications have been run on the system.
- List only the best performance figure in the paper, omitting numerous less favorable results (recall the experience of pharma tests).
- Employ special hardware, operating system or compiler settings that are not appropriate for real-world production usage.
- Redefine "scalability" as successful execution on a large number of CPUs, regardless of performance.

Fooling the masses with quantum computing

A 2014 study (14) discussed comparisons between quantum computers and classical computers. However,

- Runtimes reported in Fig. 3 for the classical solvers SA and SQA are in fact measured runtimes *divided by the problem size N*.
- Thus the reported runtimes at the rightmost points of Fig. 3 are 512 times lower than actual measured CPU times.
- The analysis of the classical algorithms involved an extensive search of the parameter space to find the best combination of parameters, and then the results for the best cherry-picked combination were reported as measured performance at each problem size.
- The time to find the best parameter combination was not included in total runtimes.

The standards and expectations of the two communities differ. But why can't we agree on a single standard?

14. T. F. Ronnow, Z. Wang, J. Job and six others, "Defining and detecting quantum speedup," *arXiv*, 13 Jan 2014, https://arxiv.org/pdf/1401.2910.pdf.

Fooling the masses with quantum computing, cont.

A 2018 study (15) compared the Coherent Ising Machine (CIM) with a D-Wave 2000Q, asserting that there is an "exponential penalty" for the D-Wave system on dense graph Ising problems. However (16),

- The exponential performance gap is based on two regression models extrapolating the algorithm scaling on the systems.
- The resulting extrapolations *overestimate* the scaling of the 2000Q processor and *underestimate* the scaling of the CIM.
- If actual measured data are used, the CIM is approximately 10X and 8000X faster on the SK and MC problems.
- In a stricter apples-to-apples comparison, the CIM advantage drops to 5X on SK and 364X on MC.

15. R. Hamerly, T. Inagaki and 20 others, "Scaling advantages of all-to-all connectivity in physical annealers: the Coherent Ising Machine vs. D-Wave 2000Q," *arXiv*, 14 May 2018, https://authors.library.caltech.edu/86965/1/1805.05217.pdf.
16. C. C. McGeoch, W. Bernoudy and J. King, "Comment on 'Scaling advantages ... 2000Q'," *arXiv*, 3 Jul 2018, https://arxiv.org/pdf/1807.00089.pdf.

Standards now or scandals later?

Now is the time to establish performance standards for the quantum computing field:

- Benchmarks: The community carefully selects a set of kernels as well as full applications, representative of real-world, state-of-the-art usage.
- Thorough documentation: Test reports include algorithms, software, compilers, system environment, precision used, accuracy achieved and details of performance calculations.
- Full disclosure: Test reports clearly state any information that may affect the interpretation of the performance results.
- Fair comparisons: Tests reflect comparable tuning, comparable precision, comparable accuracy and comparable system environments.
- Public availability: All relevant files and input data are saved on a publicly available, persistent data repository.

An IEEE-sponsored group has begun work on benchmarking standards and metrics for quantum computing. All players should support this activity.

This talk is available at: http://www.davidhbailey.com/dhbtalks/dhb-dwave-2019.pdf.