Two Computer Sciences: a Branch of Physics and Naur's Dataology

Steve Meyer

Tachyon Design Automation
San Francisco, CA 94111
smeyer@tdl.com

Presented July 4, 2014 at IACAP Thessaloniki Greece

Abstract and slides will posted on my web page www.tdl.com/~smeyer

Computing as science should be physics not mind engineering

- Peter Naur realized this in the 1980s (published in 1990s).
- According to Naur, CS is not "cognitive information processing." His 1995 *Knowing and the Mystique of Logic and Rules*, criticizes the 1990 ACM proposed CS curriculum (Appendix 2, p. 208).
- Naur saw the problem with computer science (EECS) as studying the computer as a tool. Instead, he considered many characterizations of science.
- In particular Naur analyzed Turing's original Turing Machine (TM) paper in detail from the perspective of experimental physics.

Naur continued ...

- His solution was to create a dataology department at Copenhagen University. Dataology is similar to astronomy (Naur's training). In astronomy various kinds of indirect and contradictory stellar data are analyzed without preconceived rules.
- The dataology analogy is good and provides a better way of studying "big data", social effects of data, more data versus more computation etc. Naur has written extensively on this so I will not discuss that aspect here.
- There is another science of computing that studies the nature of calculation. It uses an experimental methodology from the development of quantum physics.

19th century development of natural philosophy of calculating

- Rest of talk discusses this other aspect of calculating as physics natural philosophy. This other CS is important because there are many computer scientists who claim physics is nothing more than cognitive information processing (axiomatization of what computing is).
- The second half of the 19th century witnessed huge progress in calculating physical quantities: Maxwell's equations, combinatorics of Boltzmann gas, thermodynamics, etc. but many experimental measurements continued to disagree with calculated values.

Natural philosophy of calculating continued ...

- It was not clear if the problem was the calculations (the experimentalists view) or experiments and their interpretations (phenomenology of experiments).
- Result was the development of physics that used theories and experiments to determine how to calculate.
- I think this method is what computer science should be.

Skepticism toward logic in natural philosophy

- An important step in the development of natural science that uses experiments to evaluate formal logic was a proof by Max Planck's 1895 assistant Ernst Zermelo (later developer of Zermelo Frankel set theory) that reversible physical processes were impossible. Planck as a thermodynamics expert knew the proof was somehow wrong.
- The conflict continues because mathematicians (logicians) are still trying to prove that Zermelo's proof is correct or calling it a paradox in physics not a disproof of formalism (a web search will find hundreds of mostly unrefereed papers on this).

What I am trying to say - choosing mathematical models is empirical

- The clearest example is 19th development of non Euclidean geometry. Non Euclidean geometry removed the parallel line axiom: at most one line can be drawn through any point not on the line parallel to the given line.
- Experimentation has determined that non Euclidean geometry is the correct characterization of physical reality at least up to the current state of knowledge.
- A twentieth century example is Paul Finsler's proof that the continuum hypothesis is true (not undecidable as shown by Cohen). The proof is developed using analogy with non Euclidean geometry.

What I am trying to say continued ...

- Finsler also rejected Zermelo Frankel set theory. His view was that there is objective mathematical (calculating) reality that is experimentally determined.
- Called Platonism (Plato's cave wall image connects to reality) by Finsler. Called quasi-empirical mathematics by Lakatos.
- I am arguing mathematics used in characterizing computation needs to be experimental in Finsler's sense. Whether TMs model the physical reality of computation requires experimentation as carried out in physics, not blind application of math used by engineering.

Aaronson's hierarchy of the sciences

- In the book *Quantum Computing Since Democritus* Scott Aaronson characterizes philosophy of computer science as nothing more than formal mathematics. One way to put this is that physical reality is Turing Machines (TM).
- Aaronson starts with an assumption about the hierarchy of scientific explanation (p. 200). **" In science there is the traditional hierarchy where you have biology on top, and chemistry underlies it, and then physics. If physicists are in a generous mood, they'll say that math underlies physics."**
- I view this assumption as the problem with Aaronson. Truth is formal mathematics that flows upward to all other knowledge.
- I think Aaronson is wrong, but it is positive that an advocate of formalist CS discusses philosophical issues.

Aaronson's CS

- **"... computer science is what mediates between the physical world and the Platonic world. With this in mind, 'computer science' is a bit of a misnomer, maybe it should be called 'quantitative epistemology.' It is sort of the study of the capacity of finite beings such as us to learn mathematical truths."**
- Problem with this definition is that Aaronson's mathematical truths do not exist because assumptions determine what is provable.

Aaronson's CS continued ...

- Returning to Finsler's continuum hypothesis proof and his rejection of Zermelo Frankel set theory, Finsler believed (showed?) that mathematical structures exist that can not be generated by Zermelo Frankel set theory.
- Logician Paul Bernay's discussed Finsler's proof by saying Finsler's continuum is different from ours.
- Bernay's writes that he is not sure if continuum hypothesis is different from the parallel line axiom and believes **"a mathematical theory can not be characterized by one formal system, but only by an open succession of formal systems."** (Lakatos archive 13/75 9 July 1965).

Turing Machines need parallel speed up axiom

- Once one believes scientific theories and experiments are needed to understand physical reality of computation, CS as formal mathematics can be falsified and computation models corresponding to reality can be chosen.
- Here is an example. I think TMs should be replaced by many TM like machines with different basic operations because physical reality of calculating needs to model increased speed and class size from parallelism.

Turing Machines need parallelism continued ...

- The foundation of CS as mathematics is the Church Turing thesis that anything algorithmic can be calculated by the "universal" TM.
- In my view a problem with Turing Machines (TM) as a model for computability is that extra tapes (parallelism) does not increase polynomial speed (class membership) and maybe even size of the class of calculatable functions. It seems to me intuitively using physical thinking that parallelism needs to increase at least the size of the class of problems solvable in polynomial bounded time.
- PRAM (parallel ram) models capture parallelism but were turned into engineering although for the first abstract PRAM models P = NP. Juri Hartmanis performed the original analysis.

Turing Machines versus hypercomputation ...

- The machine unit step processing time assumption of TMs is too narrow. In my view, scientific computational complexity needs a new TM axiom (machine step definition) that causes more tapes to increase the size of the class of problems that can be solved in polynomial time.
- For speed up, what I have in mind is sometimes called hypercomputation. Namely, adding extra tapes modifies basic steps duration to be shorter possibly by a very small amount (epsilon). There are many possible extra tape time step speed up functions. Theorizing and experimentation is needed to determine which one(s) fit reality.
- The proofs that all characterizations of the class of recursive functions (TM computable functions) are the same would then be seen to abstract out the wrong properties of TMs.

Aaronson's View of TM modification

- I think Aaronson may understand his formalism problem (pp. 31-32). In defending the Church-Turing thesis, he writes **"There have been plenty of nonserious challenges"** such as **"hypercomputation"**. Aaronson only considers the exponential TM speed up function (each step reduces step time by 1/2) which certainly seems incorrect physical reality, but he neglects other functions which don't **"sound a bit silly."**.
- Doesn't the current TM model where adding resources (parallel tapes) does not increase calculation speed seem intuitively wrong?

Parallelism in X86 processors as one table like TM function

- My thinking is probably close to Naur's because we both work on compiler software.
- Simulating electronic hardware on computers is a difficult computer time consuming problem that should benefit from parallelism.
- There have been many attempts to develop parallel simulation using a large number of simple processors (GPUs) or parallel hardware which have not worked (bad TM hypercomputation step time change functions?).
- The fastest accurate hardware simulators use the low level parallelism in complex X86 multiple instruction execution units, pipelines and branch processing units to gain maybe a 10x speed up.

Parallelism in X86 processors continued ...

- The rules for filling X86 instruction units and pipelines define a physically better table like TM non unit processing step function.
- People are trying to find better parallel TM speed up step functions by building faster hardware than the X86 parallel slot step rules.
- I am imagining here that the X86 pipeline arrangement can be increased in the abstract sense without physically being able to build so large a number of separate instruction processing units.

Impossibility of Quantum computers - hidden momentum

- Once experimental physics becomes methodological choice for CS, engineering to build computers can be replaced by scientific study.
- For example, physics provides an argument why quantum computers (QC) can not exist while studying the nature of computation may help solve an important computational problem in modern physics.
- The problem involves hidden mechanical momentum. One way to look at hidden momentum is that it is an intrinsic property - a property that can not be turned into macroscopic force. This is in contrast to an extrinsic property that can produce physical force (moving of a dial in a magnetic field say).

Impossibility of QC - hidden momentum continued ...

- Entanglement is almost certainly an intrinsic property (can't provide direct force needed in physical switching "gates"). If so, QCs can't be built.
- The engineering history of this is that Mansuripur's claim that hidden momentum contradicts special relativity was refuted (trace back from K. McDonald's *Mansuripur's Paradox* paper on his Princeton web page). William Shockley anticipated this intrinsic/extrinsic property (maybe a more descriptive name is needed) in a 1960s paper that introduced the concept of hidden mechanical momentum.

My prescription - connect CS to physics

- My prescription to continued scientific progress would be to require that to be admitted to a CS graduate program, a student must have an undergraduate degree in physics, i.e. students that have learned physical experimentation.
- Restore the close connection between CS and physics that existed until the late 1970s.

Scientific problems in need of calculating method experimentation

- Von Mises economic theory claims that only a free market can calculate optimum resource allocation versus socialist planners (Neurath). Thomas Uebel writes on testing the computation issues.
- In quantum synthetic chemistry there a conflict between something called Woodward's rules for Diels Alder reactions that use measurements and qualitative calculating to define synthesis rules versus traditional quantum chemistry calculations. Calculated values are different but reaction behavior is not experimentally separable. Buhm Soon Park studies this calculation controversy which goes back to Linus Pauling.
- The Bitcoin system has shown the cryptographic axiomatized models of security are problematic. At very least the axioms are irrelevant. See Adi Shamir's analysis of Bitcoin papers.

What if cell biology had been studied as microscope engineering?

- As a thought experiment, imagine what might have happened if 19th biologists studied engineering of microscopes instead of experimental properties of cells (MESM departments - mechanical engineering science of microscopes).
- There would be catalogs of different cells according to their light emitting spectrum.
- There would have been huge progress in dyes to inject in cells to enhance light emitting properties.
- People would be predicting that in x years human eyes would be obsolete and replaced by microscopes.
- There would be (still?) no progress in the chemistry of cell reproduction or the wave particle duality of light.