Teaching Post-Classical Computation

or : CS UG degree considered harmful

Susan Stepney

Non-Standard Computation Group
Department of Computer Science
University of York

June 2004

youth

- hands up everyone whose first degree is *not* in CS
- second degree?
- *any* degree?

- CS is a young subject
  - when many of the more "mature" CS academics went to university, there was no such thing as a CS degree
    - so they have their first degree, and often their second, in some other subject
  - not true of the younger generation
    - younger CS academics have both first and second degrees in CS
    - is this an improvement?

Is Computing Science?

- Christopher Strachey
  - 1969 : is computing a suitable subject for teaching in universities and if so at what level?
- categorised relevance
  - clearly relevant / peripheral / another subject
- categorised state of development
  - firm knowledge / no underlying theory / exploratory
- and the 1969 status
  - clearly relevant ⇔ exploratory
  - another subject ⇔ firm knowledge

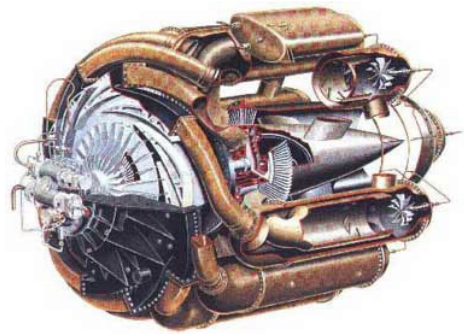
> "**we do not have enough material of the right sort to teach the full three year course to a BSc in computing**"

status in 2004

- clearly, there *is* now enough to teach
  - Algorithms and Data Structures
  - Principles of Programming
  - Computer Architecture
  - Mathematics for Computer Science
  - Digital and Analogue Circuit Design
  - Database Theory and Design
  - Human Computer Interaction
  - Lexical and Syntax Analysis
  - Operating Systems
  - Real Time Systems
  - Theory of Computation
  - …

Interdisciplinarity of subject

- application domains
  - from embedded engineering
    - controllers for jet engines, space craft, …
  - via the Web
    - e-business
    - e-everything
  - to finance
    - classical payroll
    - devising and selling novel "instruments"
    - ATMs

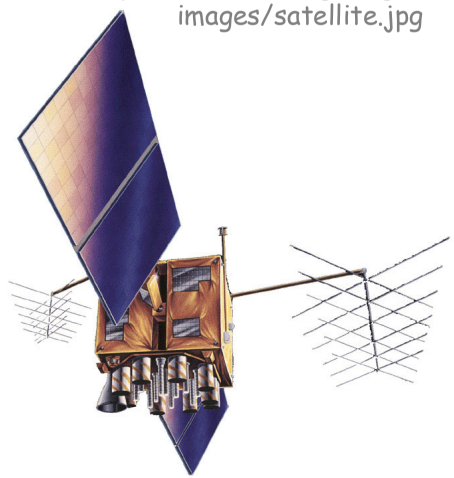


http://www.fiddlersgreen.net/aircraft/jets/whittle-jet/engine-dwg.jpg

http://www.mongoliatourism.gov.mn/images/image/bank.jpg

http://www.strasburgpa.com/sbimages/bank.jpg

http://www.scec.org/scign/images/satellite.jpg



21st Century computation

- breaking free of the classical Turing and von Neumann paradigms
  - Real World as inspiration
    - natural computation : physics-inspired, bio-inspired
    - massive parallelism, emergence, "more is different"
  - Real World as a computer
    - all computation and all data is ***embodied***
      - it's not merely a branch of mathematics
      - physical effects – particularly quantum
    - analogue computation
      - » the great missed opportunity of the 20th Century?
      - protein folding
  - Open systems
    - no Halting, rather ongoing developing interactive processes

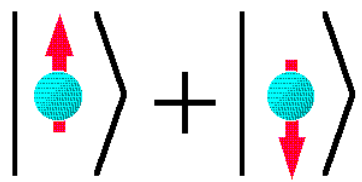
"non von" : massive concurrency

- the real world is *massively* parallel
  - with no central point of control
- our classical programming paradigm is the sequential von Neumann architecture
  - our traditional concurrent programming approaches are clumsy
    - (a few dozen) threads, etc
    - we take an intrinsically parallel world, sequentialise it, then add the wrong sort of concurrency back on!
- we need arbitrary decentralised concurrency as a fundamental computational paradigm
  - a complete rethink of the primitives

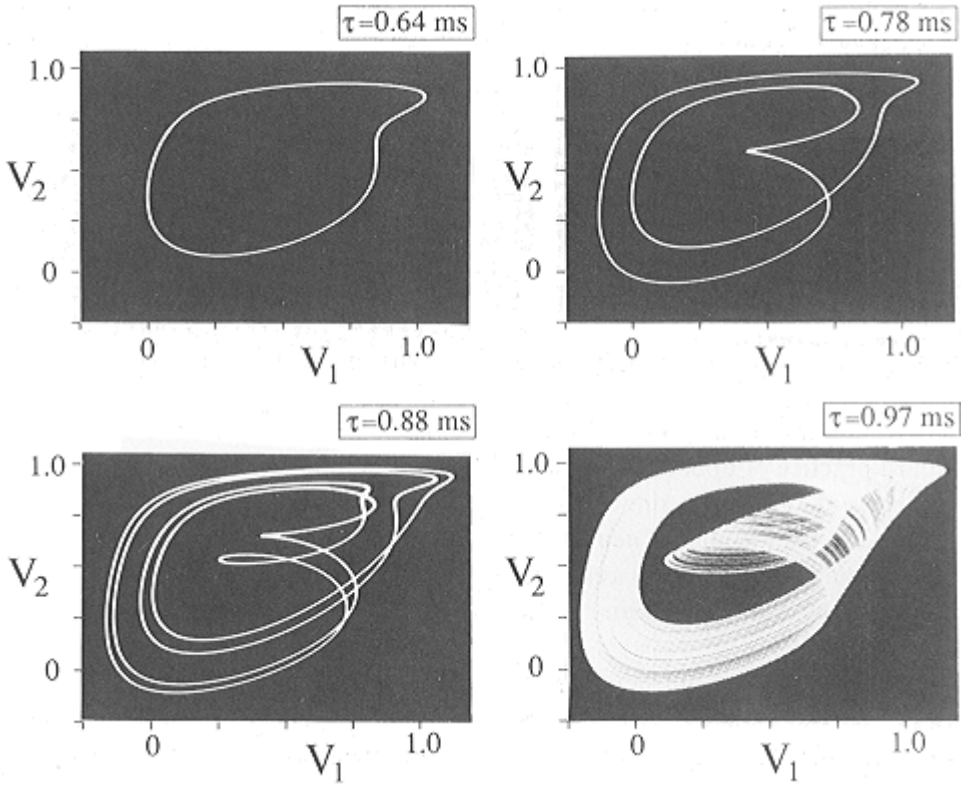
computation from physics

- physics *inspired*
  - simulated annealing
    - search algorithm
  - dynamical systems
    - control algorithms

- embodied physics
  - quantum computing



  - analogue computing

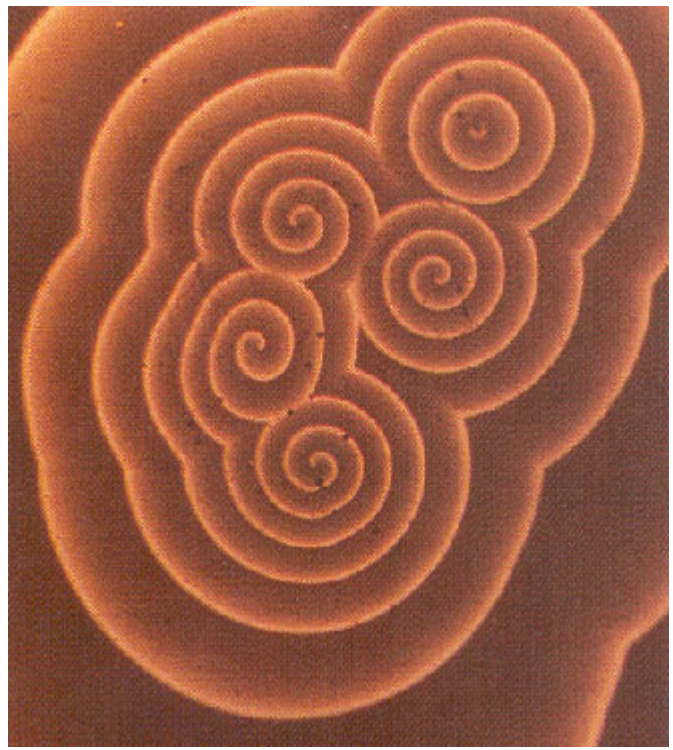


C. M. Marcus *et al.* Nonlinear dynamics and stability of analog neural networks. *Physica D* 51 234-247 1991

computation from chemistry

- chemistry *inspired*
  - cellular automata

- wet chemistry
  - Reaction-diffusion systems
    - computing with chemical waves
  - Belousov-Zhabotinsky (BZ) reaction



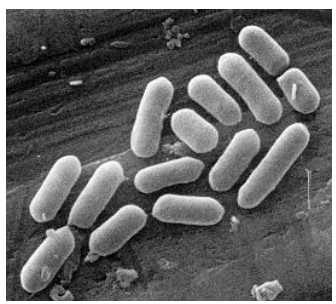
[Gaponov-Grekhov & Rabinovich, plate 34]

computation from biology

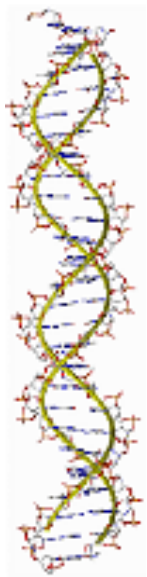
- bio-*inspired*
  - artificial neural networks
  - evolutionary algorithms
  - artificial immune systems
  - L-systems (plant growth), GRNs, …
  - swarm algorithms, ant colony optimisation, …

[Prusinkiewicz & Lindenmeyer, fig 1.25]

- wet biology
  - DNA computing
  - computing with membranes, cells, bacteria

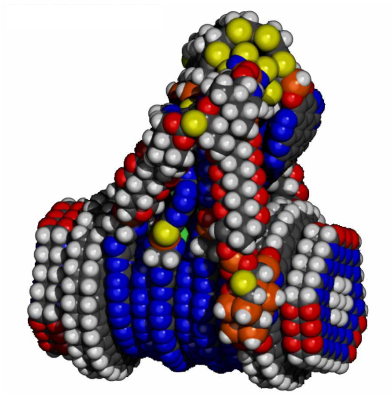


http://www.uga.edu/caur/bacteria.jpg



example : nanite assemblers

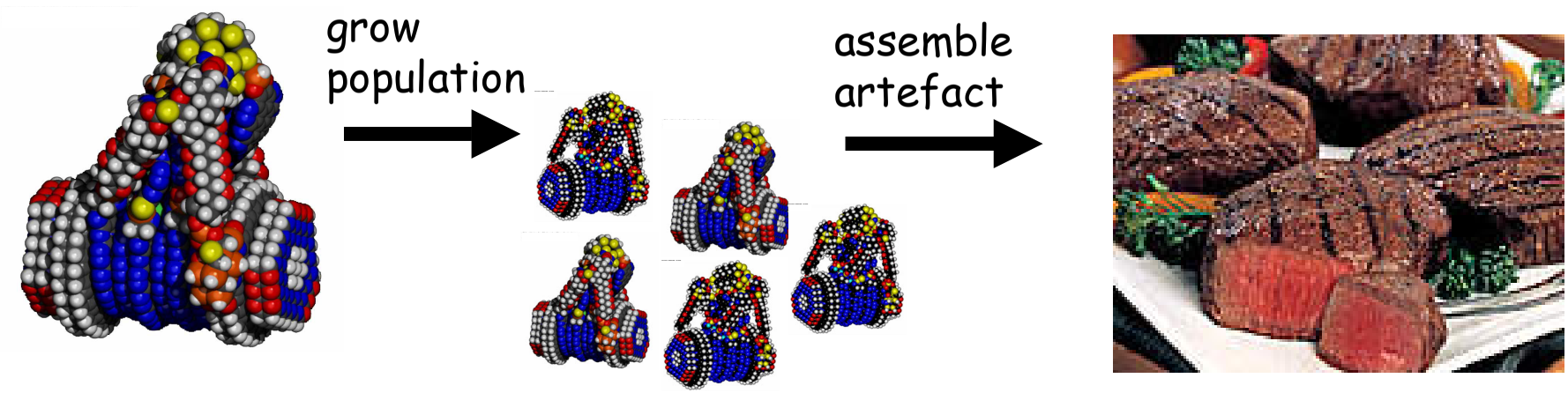
- ***molecular nanotechnology (MNT)*** [K. Eric Drexler, 1986, 1992]
  - molecular scale programmable "robots"
    - assemblers, nanites, nanobots
  - mechanically positioning reactive molecules
  - making macroscopic artefacts
    - assembling anything, from steaks to spaceships
  - assemblers make conventional factories unnecessary



- challenges of software, tools, techniques, models, …
  - hardware/wetware too : physicists, engineers, biologists

assembling artefacts

- growth and development on two levels
  - bootstrap a small initial assembler population
    - pool of raw material (mainly carbon)
    - assemble trillions of nanites (exponential growth)
  - eg, to build a new nano-fabrication plant
    - which then assembles, or "grows", the artefact



http://www.imm.org/

http:// www.omahasteaks.com/

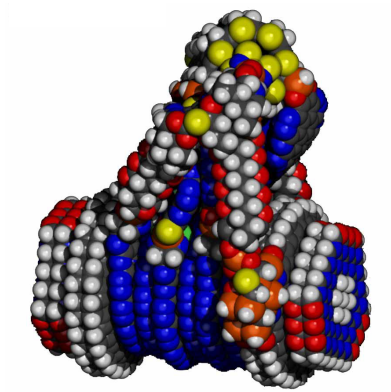
disassemblers

- as part of assembly
  - disassembly of raw materials required for assembly
  - disassembly of "scaffolding" required during assembly
- medical applications
  - scouring cholesterol from arteries
  - filtering blood toxins
  - removing damaged cells
  - repairing damaged nerves
- environmental applications
  - disassembling toxic chemicals into safe constituents
  - concentrating heavy metals
  - disassembling unwanted artefacts

the MNT design challenge

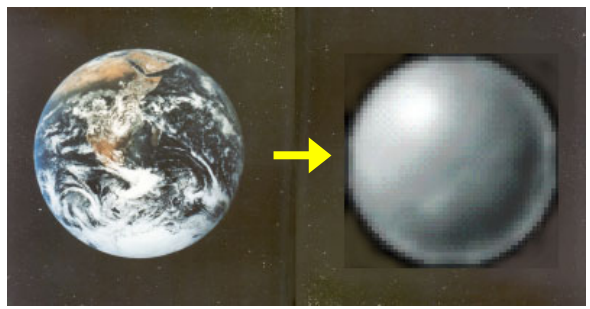
- assembled artefact is ***emergent property***
  - of actions of vast number of nanites, "growing" the artefact
- design requires "reverse emergence"
  - from desired emergent artefact
  - to behaviour of nanite assemblers
    - extreme example of "non-classical refinement"

design appropriate assemblers



the MNT safety challenge

- given vast numbers of nanites, some will go wrong
  - if they are self-replicating, they will evolve
    - evolution is an inevitable consequence of "reproduction, variation, selection"
- safety critical application
  - "Grey Goo" scenario
  - current approaches totally inadequate
    - "proof of correctness" doesn't help with a mutant
      - the mutant is a *different system*
  - new safety techniques and tools required
    - evolution will exploit *anything*
      - even (especially) things ***outside your abstract model***
      - particularly the embodied properties of the substrate



the MNT challenge

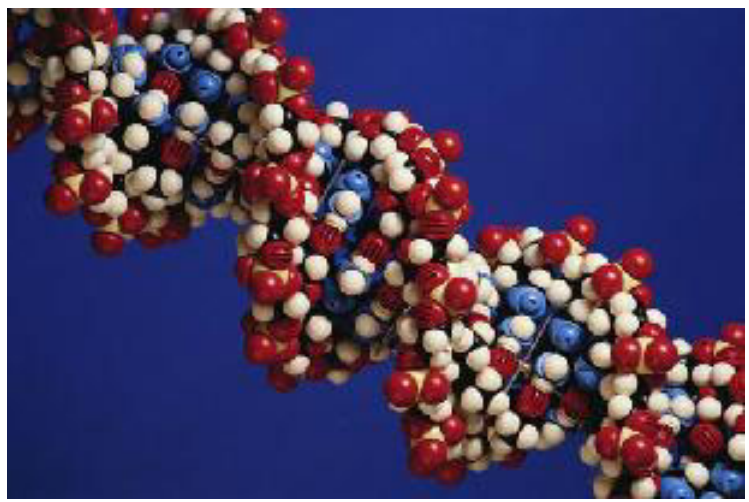
- simple rules give complex behaviour
  - but *which* simple rules give the *desired* complex behaviour?
  - designing the desired emergent properties
  - designing the lack of undesired emergent properties
  - thorough understanding of growth processes, self organising systems, …
    - will require a lot of biology, viewed from a computational perspective
- embodied nanites
  - strange physics at very small sizes
    - effects of friction, flow, gravity, etc all very different
  - inevitability of evolution
    - evolution exploiting embodied properties
      - properties ***outside the abstract model***

abstraction v. detail

- what is this:

AGGCATTGAAACGGCCTTTAA

- no, it's *not* DNA
  - it's a string of letters!
  - it's (part of) a VERY simple *model* of DNA
    - electron hopping
    - protein folding
    - genotype v. phenotype



http://www.columbia.edu/cu/opg/images/dna.jpg

too abstract

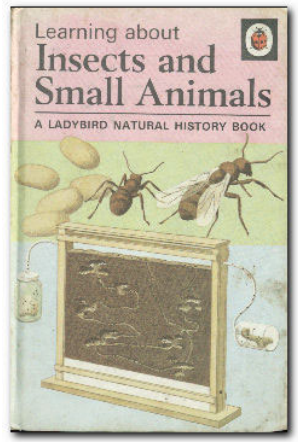
- I often complain how hard it is to teach abstraction
  - students have difficulty rising above the detail
- but a lot of the current nature inspired approaches abstract *too far*
  - rich interactive environment
  - rich physics
    - supporting hierarchies of emergence
- abstraction is an important skill
  - by the right amount, and in the right directions

Things should be made as simple as possible -- but no simpler.

Einstein

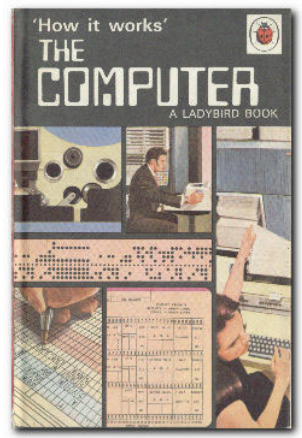
sources of information

- most bio-inspired computer algorithms appear to get their biological input from the **Ladybird book of Ants** / Genetics / Immunology / Neuroscience / whatever
  - if you think learning enough about Ants this way isn't as silly as learning about Immunology or Neuroscience … that means you don't know enough about ants!
  - all biology is incredibly intricate, subtle, complex, detailed, and difficult
- all teaching is "lies-to-children"
  - [Stewart & Cohen]



http://www.darkwoodonline.co.uk/si/13196.html

http://www.ladybird.co.uk/

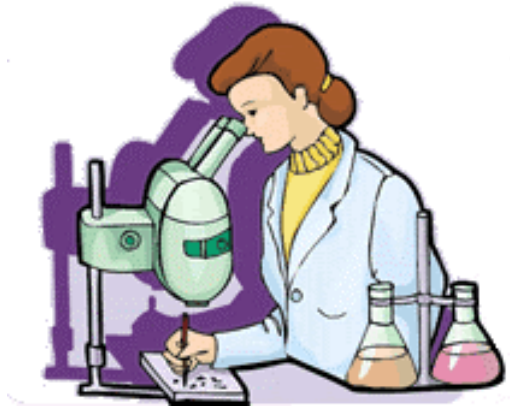


CS in the 21st century

- need some computation
  - but much of what is taught may be too "classical"
    - sequential, halting, mathematico-logical, …
    - some teachers, and hence their pupils, have an emotional attachment to the Turing machine
- need some real science
  - some specific area of physics, chemistry or biology
    - in some depth, to have an appreciation of the sheer complexity and potential
- need a "scientific mindset"

So, is Computing *Science*?

- science
  - physics
  - chemistry
  - biology
- non-science
  - social science
    - sociology ?
  - political science
  - domestic science
  - creation science
  - …



you've got an ology, you're a scientist!

http://www.shef.ac.uk/staff/newsletter/vol24no1/page04.html

What is science?

- scientific method
- conjectures and refutations

http://www.filmmedical.co.uk/images/experiment.jpg

http://www.pigdogproductions.com/experiment.jpg

experimental design

- null hypothesis
  - what's the question?
- control
  - what would have happened otherwise?
- varying parameters
  - change one thing at a time
- training set v. evaluation set v. *independent* test data
  - why you shouldn't evaluate your results on your training data
  - *recognising tanks*

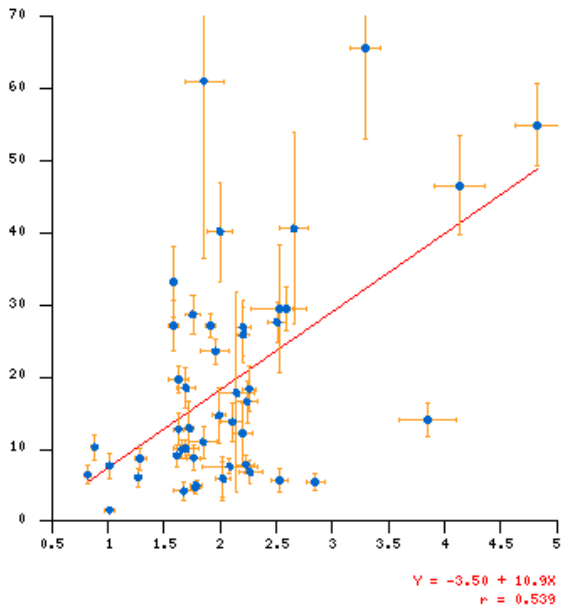
http://neil.fraser.name/writing/tank/

statistics

- significance tests
- confidence limits
- error bars

http://www.n-topos.gr/en/statistics.jpg

- *CS and the single datum*
  - "typical best result"



http://ploticus.sourceforge.net/gallery/scatterplot10.gif

multidisciplinary learning

- two first degrees
  - biology / …
  - computer science
- a first degree and a masters
  - which way round?
  - conversion Masters
- what to do after a CS degree?
  - advanced Masters often require same subject first degree

a Masters in Natural Computation

- bio-inspired
  - including some real biology
  - and scientific method, experiment design
- embodied
  - effect of laws of physics
  - using wet chemistry and biology
- complexity and emergence
  - chaos and complex adaptive systems
  - self organising systems

Bio-inspired computation

- Evolutionary
  - Genetic Algorithms, Genetic Programming, …
- Neural
  - ANNs, back-propagation, associative memories, spiking neurons, dynamic nets, …
- Further Bio-inspired Algorithms
  - artificial immune systems, growth (L-systems, embryology and developmental processes), social behaviour (ants, termites, swarms)
- Simulating Complex Biosystems
  - computational systems biology

Embodied computation

- Quantum Computation
  - basics, reversible computation, entanglement, algorithms, exponential speed-up
- Quantum Communication
  - protocols, teleportation, error correction, dense coding, crypto
- Computing with Biology and Chemistry
  - DNA, cell, and membrane computing, reaction-diffusion computers
- Unconventional Computational Hardware
  - FPGAs, Analogue Computation, FPAAs, evolvable hw, optical

Complexity and Emergence

- Dynamical Systems I
  - non-linear dynamics, trajectories, attractors, bifurcations
- Dynamical Systems II
  - Lorenz equation, logistic equation, fractals, IFS, Henon map, ...
- Adaptive and Learning Agents
- Emergence
  - cellular automata, artificial life, "edge of chaos", complex adaptive systems, self organising critical systems, nanotechnology

21st century CS

- will need more *science*
  - especially the scientific mindset
    - hypotheses, statistics, …
- preparation for multidisciplinarity
  - right levels and kinds of abstraction
- flexibility
  - rapid change in own subject
  - rapid change in other subjects