The Neglected Pillar of Material Computation

Susan Stepney

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

July 2007

The UK Grand Challenges in Computing

- UK Computing Research Committee (UKCRC) initiative
  - to discuss opportunities for advancement of computing science
  - (Nov 2002) original call resulted in 109 submissions, merged and refined into seven "Grand Challenges"

    http://www.ukcrc.org.uk/grand_challenges/index.cfm/

1. In Vivo -- In Silico : Andrew Bangham

   The Worm, the Weed, and the Bug : breathing life into biological data

2/4. Global Ubiquitous Computing: Science & Design : Morris Sloman

3. Memories for Life : Nigel Shadbolt

5. Architecture of Brain and Mind : Murray Shanahan

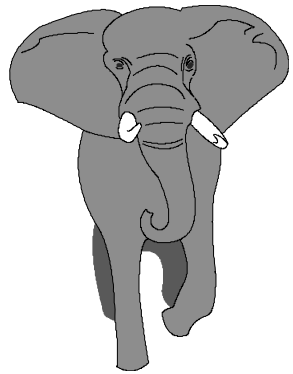
6. Dependable Systems Evolution : Jim Woodcock

7. Journeys in Non-Classical Computation : Susan Stepney

   Robust, adaptable, powerful computation, as inspired by Nature

"non-classical" computation?

like defining the bulk of zoology
by calling it the study of
'non-elephant animals'



- Stan Ulam (attrib)
on the name "non-linear science"

non-linear science /
non-classical computation

Here be Dragons

linear science /
classical computation

components of computation

non-halting
continuous, open
stochastic
massively parallel,
robust

biology

in materio
(substrate)

Turing machines:
Halting, Universality

discrete, symbolic
closed

deterministic
sequential, fragile

classical
infor-
matics

computation : logical *and* physical

- virtual/logical constraints
  - computability
    - maybe also a *physical* constraint? [philosophical arguments]
  - feasibility (ie, not NP-complete, or worse)
    - theoretical constraint of a class of problems
      - approximations, particular instances, may be feasible

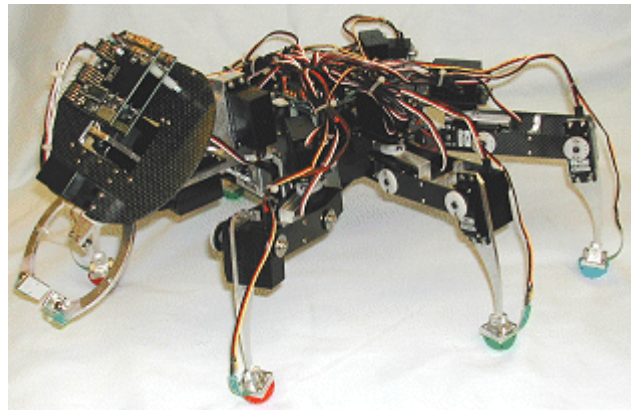
- physical/material constraints
  - fundamental physical laws (affect computability)
    - speed of light; conservation of energy/matter; entropy; …
  - specific laws (affect feasibility)
    - material composition: strength; elasticity; resistance; …
    - may give some solutions "for free"

the power of the substrate

- the logical system can transfer some of its computational burden (memory / state / processing) *to the substrate*
  - eg, analogue computation
- may allow new problems to be solved *in new ways*
  - exploiting vast computational power of rich substrate
- both richness *and* constraints of substrate contribute to this power
  - richness gives substrate vast computational power
  - constraints give some solutions "for free" (or much cheaper)
    - constraining the space to a smaller region
    - constraining the computation to particular trajectories

aside: substrate v. analogy

- nature-inspired computing …
  - simulated annealing
  - artificial chemistry
  - artificial neural networks
  - evolutionary algorithms
  - artificial immune systems
  - ant colony optimisation
  - …



http://i44.photobucket.com/albums/f16/daisydexterdobbs/Blog/robotant.gif

- … has to ***implement (simulate) the substrate***
  - thermal jiggling, conservation of mass, signal propagation, selection and fitness, shape matching, pheromone decay, …
    - inefficient!
    - advantage: can do it ***differently***

Biological substrates: "wet" computation

- DNA
  - base pair matching
    - graph problems
- bacteriorhodopsin protein
  - 3D optical memory
- leech neurons
  - hybrid NNs
- bacteria and slime moulds
  - robot controllers
  - graph problems
  - …

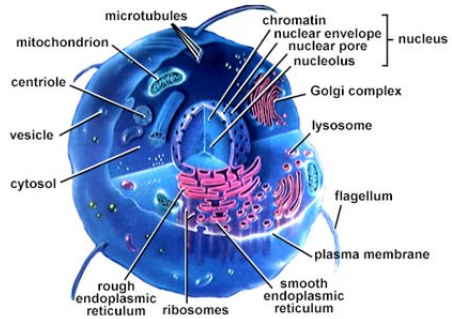
http://www.newscientisttech.com/article/dn8718.html

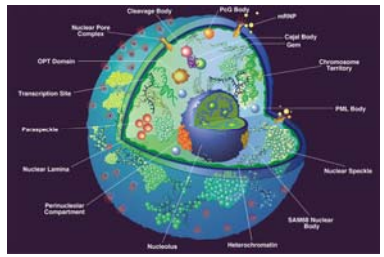
wanted: theory of substrate computation

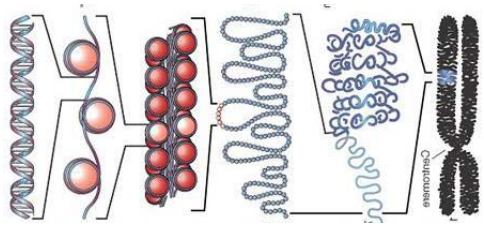
- *how does wet computation help us get new **theories** of computation, new **insights** into computation?*

- **problem**: bio-materials have *5 bn years of evolution* behind them
  - extremely complex, extremely complicated
  - not seeing the "primitive" substrate
  - masking any *general principles* of substrate computation

- so first, look to simpler substrates
  - physics and chemistry -- *unevolved*







http://www-math.mit.edu/~lippert/18.417/lectures/01_Intro/ http://spectorlab.cshl.edu/images/NucleusModel.jpg

http://www.gla.ac.uk/cancerpathology/genemech/awest/chromatin.jpg

components of computation

non-halting
continuous, open
stochastic
massively parallel,
robust

biology

in materio
(substrate)

Turing machines:
Halting, Universality

discrete, symbolic
closed

deterministic
sequential, fragile

classical
infor-
matics

in materio
compu-
tation

non-halting,
non-universal

continuous,
stochastic,
massively parallel,
robust (?)

physical substrates

- optics
- chemical reactions
  - reaction-diffusion computers
    - eg: Voronoi diagrams
- quantum
  - liquid NMR
  - semiconductor quantum dots
- plastic foam
- jelly

http://uncomp.uwe.ac.uk/delacycostello/index_files/image065.jpg

Mills et al. "Empty Space" Computes

"I wouldn't start from here"

- classical approach to substrate (ab)use
  - starts from a particular logical-mathematical model of computation (Boolean logic, UTMs, etc), then tries to find (force, engineer) a physical substrate to behave like this

- why should *anything* material naturally behave like this?
  - we have spent the last 50 or so years *torturing* silicon to implement Boolean logic

"... is like a dog's walking on his hind legs.
It is not done well; but you are surprised to find it done at all."

[Boswell. *Life of Johnson.* 1791]

virtual machines all the way down

- classical approach:
  - engineer the physical substrate to ***implement the model of computation***
- and recall:
  - (classical) bio-inspired computing has to ***implement (simulate) the (bio) substrate***

bio-inspired VM (simulation)
Boolean VM (computing hardware)
silicon substrate (physical material)

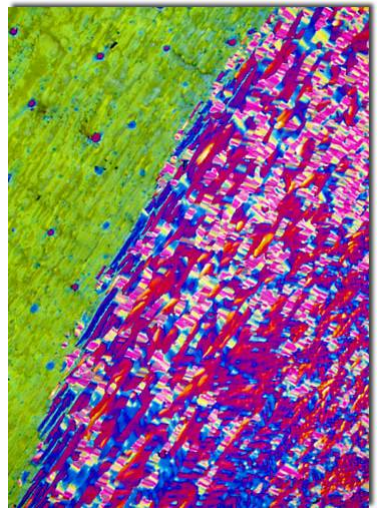
- no wonder these novel paradigms can be inefficient!

not another NOT gate, please!

- the whole thrust of "unconventional" computation is to be just that – ***unconventional*** – not only in the choice of substrate, but also in what that substrate *does*
- why use diffusing chemicals to implement logic gates?
  - or other weird ways of getting extremely inefficient UTMs?
    - other than to show we can torture this other material, too?
- *how do chemical logic gates help us get new **theories** of computation, new **insights** into computation?*
- unconventional approach : abstract a model of computation from ***what the substrate does naturally***
    - rarely (never?) is this to implement Boolean logic, or a UTM!

"edge of chaos" substrates?

- motivation: "**edge of chaos**" = maximal complexity, maximal "computational power"?

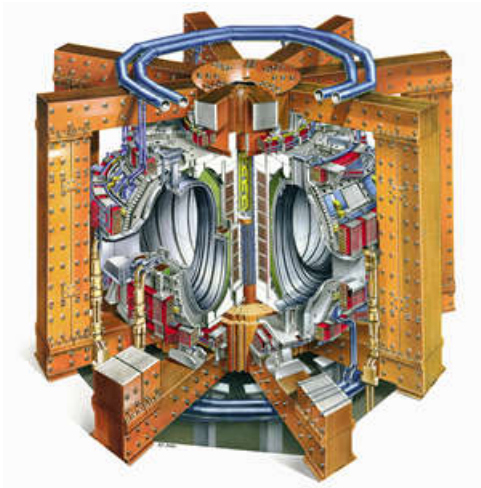


http://www.molecularexpressions.com/dna/magneticfield/magneticfield11.jpg

- critical region, phase transition
  - between "solid" : all structure/memory, no dynamics/processing and "fluid" : plenty of dynamics, no structure
- so: look at material substrates with *both* complex ***structure*** (memory) and complex ***dynamics*** (processing) over a wide range of length/timescales
  - also: start with materials that have existing commercial (laboratory) technology base

complex *structure* and *dynamics*

- **plasmas** (hot ionised gases)
  - experimental fusion reaction plasmas
  - dynamics galore
    - the problem is controlling it!
  - structure
    - turbulent vortices, shear flows



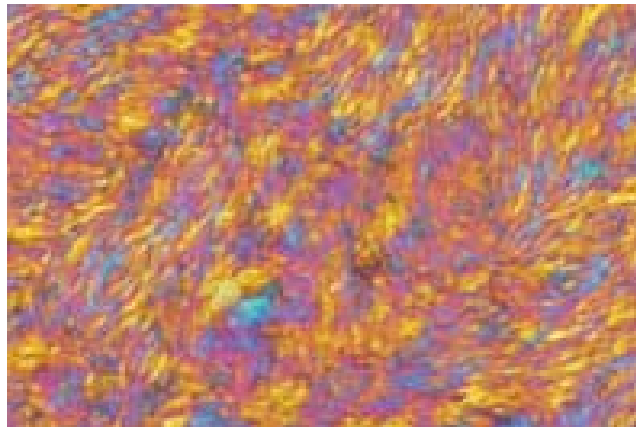
http://www.msm.cam.ac.uk/phase-trans/2006/Irradiated_Steel/Irradiated_Steel.html

- **NMR**
  - commercial laboratory NMR spectrometers
  - (*not* "quantum computing" NMR)
  - dynamics and structure
    - of interacting nuclear spins

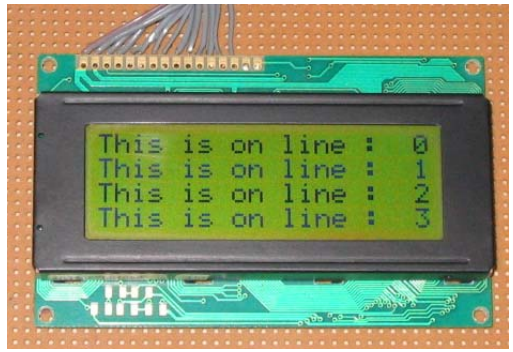
http://en.wikipedia.org/wiki/Image:Pacific_Northwest_National_Laboratory_800_MHz_NMR_Spectrometer.jpg

liquid crystals

- poised between **solid** and **liquid**
- dynamics
  - molecules flow / rotate
- structure
  - orientational order on wide range of length scales
- LCD chip packages
  - [Harding & Miller]
- soft condensed matter
  - from liquid crystals to bio-materials



http://www.trnmag.com/Stories/2004/022504/Nanotube_mix_makes_liquid_crystal_Brief_022504.html



http://home.iae.nl/users/pouweha/lcd/images/lcd/lcd-pic16c84.jpg

a computer?

- we've found matter with complex structure and dynamics
  - that naturally does interesting computational things
- and we've agreed *not* to torture it into implementing Boolean logic

- but matter just sloshing around in a complex manner isn't (yet) a *computer*

http://www1.istockphoto.com/file_thumbview_approve/1668638/1/istockphoto_1668638_slosh.jpg

programmability

- we need to be able to use the stuff to solve (a range of?) *our* problems
- we need to be able to configure it
- we need to be able to ***program*** it

- but, how to program complex stuff?

controlling fields

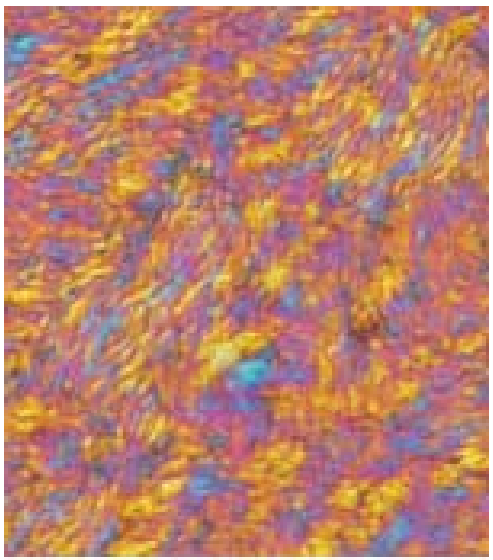
- *ex hypothesi* : the computation performed depends on the details of the complex structure and dynamics

- so, programming = altering those details in a controlled manner
- controlling the *initial conditions*
  - "ballistic" computation
- controlling the *boundary conditions*
  - "interactive" computation
  - by controllable "fields" that affect structure/dynamics permeating the device
    - control dependent on current path of computation
    - may control with a classical computer!

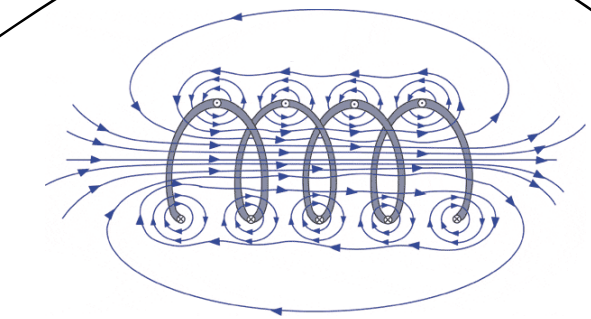
a programmable material computer?

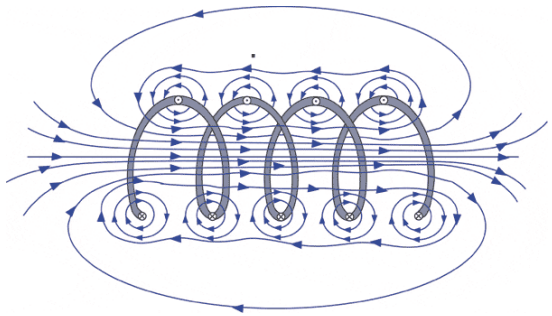
program steps (modulated fields)

field gradients



substrate with complex structure and dynamics





output (sub)state

controlling choices made *in silico*

what's still missing?

non-halting
continuous, open
stochastic
massively parallel,
robust

biology

in materio
(substrate)

Turing machines:
Halting, Universality

discrete, symbolic
closed

deterministic
sequential, fragile

classical
infor-
matics

in materio
compu-
tation

non-halting,
non-universal

continuous,
stochastic,
massively parallel,
robust (?)

laws of physics

understanding paper

"Turing hoped that his abstracted-paper-tape model was so simple, so transparent and well defined, that it would not depend on any assumptions about physics that could conceivably be falsified, and therefore that it could become the basis of an abstract theory of computation that was independent of the underlying physics. 'He thought,' as Feynman once put it, 'that he understood paper.' But he was mistaken."

[Deutsch, 1997]

physics matters !

- in other words, the entirety of classical computation is based on a ***demonstrably false premise***

- classical approach:
  - engineer the physical substrate to ***implement an*** **unphysical** ***model of computation***

- unconventional approach : the model of computation is just what ***the substrate does naturally***
  - as it naturally follows the actual laws of physics
    - (whatever they are)

unconventional computing

- a substrate with complex structure and dynamics
  - eg, soft condensed matter
  - ***biological systems have evolved this substrate***
- a computational model that encompasses this

- programmability, in terms of pervasive fields
  - controlling initial and boundary conditions
  - ***biological systems have evolved this control***

- eschew universality
  - do *in materio* what the material does best
  - do *in silico* the rest

The Grand Challenge

to produce a fully mature science of all forms of computation, that embraces the classical and the non-classical paradigms

- many journeys, one Challenge
- like all science, the Challenge is an ongoing journey

http://www.cs.york.ac.uk/nature/gc7/