Embodied computation

Susan Stepney

Non-Standard Computation Group

June 2006

disembodied computation

"brain in a vat"

"black box"

0, 0, \rightarrow, 1, 1
0, 1, \leftarrow, 2, 0
1, 1, \downarrow, 1, 1
2, 0, \leftarrow, 5, 1
3, 0, \leftarrow, 4, 1
natural embodied computation

continually sensing the ever-changing world, acting based on the current state of the world, and further changing the world
Simon’s ant walking on a beach

Viewed as a geometric figure, the ant’s path is irregular, complex, hard to describe. But its complexity is really a complexity in the surface of the beach, not a complexity in the ant . . .

The apparent complexity of its behaviour over time is largely a reflection of the complexity of the environment in which it finds itself.

[Simon, 1996]
embodied computation -- 5

Department of Computer Science

embodied: in an environment

continual rich sensory environmental input

the environment is a rich and active resource—a partner in the production of adaptive behavior.

http://www.piaggio.cci.unipi.it/robotics/images/hand.jpg

sophisticated outputs; environmental manipulation (stigmergy)

[Clark, 1997]

continual rich sensory environmental input

[http://www.24hourmuseum.org.uk/content/images/2004_0170.jpg]
embodiment yesterday ...

“What are you going to do?  *Sucker* me to death?”
... tomorrow?

Moravec: fractal “bush” robot

http://www.frc.ri.cmu.edu/~hpm/project.archive/robot.papers/1999/NASA.report.99/9901.NASA.S3.html
the power of embodiment

• rich interaction allows the system to transfer some of its computational burden (memory / state / processing) to the environment

• may allow new classes of problems to be solved
  - exploiting vastly more computational power

• both richness and constraints of environment contribute to this power
  - richness gives environment 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
virtual embodiment

- embodiment offers advantages (to the embodied)
  - can we engineer these, including *software* systems in *software* environments?
    - *embodied* software agents on the Web?
    - *embodied* AIS on a network?

- “*having some (physical) body interacting with some (physical) environment*”
  - says both too little, and too much!
    - little insight into nature of interactions, etc: no design help
    - assumes a *material situatedness*: seems to rule out software/virtual embodiment
“ontologically neutral” definitions

• “structural coupling” \[\text{[Maturana & Varela, 1980]}\]

• “continuous reciprocal causation” \[\text{[Clark, 1997]}\]

• “non-destructive perturbations between a system and its environment, each having an effect on the dynamical trajectory of the other, and this in turn effecting the generation of and responses to subsequent perturbations.” \[\text{[Quick and Dautenhahn, 1999]}\]
embodiment model

computational system

---

“suitably rich” complex internal dynamics

analogue / digital open / predefined

sensors

rich dynamical perturbatory interactions

environment

---

complex dynamics (assumed?)

material / virtual open

actuators

embodied via rich complex feedback
rich feedback

- actuators change the environment ...
- ... thereby changing what the system senses

- minimal interaction:
  move the system (position, orientation)
  - change the part of the environment that is sensed
    - environment may react to changed location
- rich stigmergic interaction:
  alter the environment
  - mark, erase, build, dismantle, ...
  - sense these alterations, and act accordingly
    - communication with other embodied systems
physical and virtual constraints

- **physical/material constraints**
  - fundamental physical laws
    - speed of light; conservation of energy/matter; entropy; ...
  - specific laws
    - material composition: strength; resistance; ...
  - may give some solutions “for free”

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

• interaction timescales need to be comparable to environmental changes
  - not too fast: environment cannot react in time
  - not too slow: environment has “moved on”
    • (part of) the solution to Searle’s “Chinese Room” fallacy

• real time constraints essential
  - high bandwidth interactions on relevant timescales
    • animals and plants are both embodied
      - animals have vision: fast interaction (motion)
      - plants do not: much longer interaction timescales

My vegetable love should grow
Vaster than empires and more slow.  [Marvell]
rich physical environment

• “complex dynamics (assumed?)”
  - of sufficient complexity that it is possible to transfer some of the computational burden to it
• “open”, exhibit continual novelty
  - which can be exploited by the computational system
  - can exploit any feature of the environment
    • not just those in our abstract models
  - evolve to exploit “extra-logical properties”
    • Adrian Thompson’s FPGA experiments
    • cryptanalysis side channels
    • analogue properties
• hence “the world is its own best model” [Brooks]
impoverished environments

- physical environments
  - robot mazes: sterile white-walled corridors
    - no rich structure to couple to
    - could you navigate in such an environment?
    - remember Simon’s ant: *The apparent complexity of its behaviour ... is ... a reflection of the complexity of the environment ...*

- (most) virtual environments
  - closed, pre-determined, finite, discrete representations
  - no “rich” dynamics
    - fragile, brittle, impoverished
      - true of most computational systems, too
  - no rich structure to couple to!
  - contrast the Internet/Web
    - open, constantly changing and growing

rich virtual dynamics

• sufficiently complex dynamics, at the “edge of chaos”
  - random noise is not “rich”; it is unstructured
• stable, persistent, yet “poised”, emergent patterns
  - sufficient stability for memory; sufficient changeability for adaptation
• fractal proteins [Bentley]
  - exploit non-linear properties of Mandelbrot set to simulate complex non-linear, rich behaviours
• artificial “gene regulatory networks”
  - with suitable complex non-linear, rich behaviours
co-evolution of sensors/actuators

- “rich dynamical perturbatory interactions”
  - coupling rich dynamics of computational system and environment
- cannot design in isolation
  - codesign/coevolve system and its sensors and actuators
    - in the context of the environment
      - embodied systems cannot be “transplanted” to very different environments
    - completely opposed to “well-defined interface” design philosophy
- adaptive system and environment will each learn to exploit the others’ inputs/outputs
  - depend on both, and both are changing!
    - adaptive system needs to change/adapt sensors and actuators
      - modalities, number, position, bandwidths, ...
developmental embodied systems

- simple designed system
  - static, non-adaptive, unchanging

- adaptive system
  - initially designed, then changes and adapts with environment

- developmental system
  - mature system “grows” from a “seed”
    - the seed itself may be designed
  - embodied: grows and adapts in an environment
    - same seed grows differently in different environments
  - environment shapes and is shaped by that growth
  - developmental systems cannot be “transplanted” to very different environments

http://www.artificialplants.co.uk/page2.htm
design principles I

1. design the system with sufficiently complex dynamics, that can execute this dynamics on the relevant timescale(s) of the environment
   - there may be several relevant timescales: immediate reaction; slower adaptation/learning; even slower evolution

2. design a sufficiently high interaction bandwidth on the relevant interaction timescale(s)
   - physical environments naturally offer high bandwidth
   - virtual environments need to be designed to offer sufficiently rich information
3. ensure that input from the environment is constantly available and up to date
   • continually sensing/filtering relevant data
     • not requesting or polling it
   • data needs limited spatial/temporal extent to be “up-to-date”

4. ensure that the system perturbs the environment, rather than being merely a passive observer
   • actuators as well as sensors
   • must change/affect the “data stream”
design principles III

5. ensure that the environment has sufficiently complex dynamics
   • a “given” for a physical environment
     • but beware those sterile artificial mazes!
   • add “edge of chaos”, or similar, to a virtual environment

6. allow the system to exploit structure and constraints in the environment in order to simplify its tasks
   • physical environment naturally has structure and constraints
   • virtual environment needs suitable structure and constraints designed in
     • “structure and constraints” ⇒ randomness not suitable “richness”
7. Apply embodied systems only in “softer” problem domains where approximate solutions are appropriate and acceptable
   • approximation is natural in the physical, analogue world
     • determine the acceptable degree of approximation
   • crisp digital problems are not appropriate
     • removes many classical computing applications!
       • including integer arithmetic, and the ubiquitous “2-bit adder” case study
8. co-design the system and its interface (sensor and actuator numbers, positions, data formats, etc)

9. design the system to develop, “grow”, in the relevant environment
   - the system should start as small as possible (“embryo”), and develop in the context of its environment
   - continual online learning
     - not offline learning followed by a frozen state deployment
where next?

- "BiGBIE": immuno-engineering principles (EPSRC proposal)
  - three diverse case studies, with different styles of embodiment
- YCCSA pump priming
  - embodied robot experiments