Computation with Dynamics and Attractors

GC7 : Journeys in Non-Classical Computation

Susan Stepney

Non-Standard Computation Group
Department of Computer Science
University of York
phase space and attractors

- discrete trajectories in discrete *state space*
  - the states placed to make the structure explicit
  - attractors are cycles in state space, after any transients have died away
- dynamical systems moving in space
  - discrete state space $\rightarrow$ continuous *phase space*
    - one dimension for each parameter needed to specify the state
  - system trajectory is movement through this space
    - eg, system of $N$ particles as a point moving in a $3N$-dimensional positional space, or in a $6N$-dimensional "position+momentum" space
    - eg, pendulum in a 2D "displacement+velocity" space
random Boolean networks

- Stuart Kauffman
  - RBN as a very simplified model of gene regulatory networks

- network of $N$ binary valued nodes
  - each node takes binary inputs from $K$ randomly chosen nodes
  - inputs combined by a random Boolean function

- effect of $K$ on:
  - cycle length, number of attractors
  - stability
    - does it return to the same attractor after a small transient perturbation
  - reachability
    - how many other attractors reachable after a single small permanent perturbation
dynamic patterns in the brain

• spatio-temporal patterns
  - one attractor while idle
  - move into another when “thinking”

“the strange attractor characteristic of the attention state was replaced by a new, much more ordered attractor related to the recognition process”

--[Solé & Goodwin, 2000]

• attractor structure of the brain enables cognition
  - similar applications for general computation
  - set up a computational system (evolve, grow, learn) with the appropriate attractors
computational trajectories and attractors

- for fully dynamic computation with no halting, the computation emerges as (an observed projection of) the trajectory
  - dynamic associative memories, neural networks, artificial cognitive processes, ...
  - robotics controllers, ...
- the trajectory is governed by the structure of the underlying phase space and its attractors
  - whether discrete or continuous
- particularly important if the phase space itself is dynamic
  - parameters changing due to the environment, etc
open, far-from-equilibrium systems

- open -- constant addition of new resources
  - energy, matter, information flowing into and out of the system
    - new sand added to the pile (Per Bak 1987)

- far-from-equilibrium systems
  - not in a steady state, but at the computational “edge of chaos”
    - avalanches “on all scales”

- tend to form stable structures, patterns, that persist
  - stable, but not static
    - can change readily in response to stimuli, are “poised”
  - these patterns are emergent properties
    - levels of emergence -> hierarchical structures, in space and time

- impossibility of pre-defining the phase space
the proposed journey

- to develop a new computational paradigm expressed in dynamical terms of attractors and trajectories
  - can a computation be expressed as a trajectory amongst various attractors, each changing as the result of some parameter / input?
    - does the state transition analogy hold?
  - can a computation be expressed as movement between the unstable periodic orbits of a strange attractor?
    - how are these orbits selected?
  - how can we reason about “emergent” phase spaces?
    - where the phase space changes as the computation progresses
  - what kinds of algorithms are most suited to this paradigm?
  - what are the programming primitives and higher level languages?