

The Nature of Computing — Computing in Nature

I. Introduction

II. Weaning Ourselves Away from Church-Turing Computability

A. Every model presupposes a frame of relevance

B. Uncovering the FoR of Church-Turing Computation

- 1) Historical roots
 - (a) They were addressing issues of formal calculability and formal provability
 - (b) Principal assumptions included a finite number of steps requiring finite resources
- 2) Definitions in terms of classes of functions
- 3) Discreteness assumptions
 - (a) Information representation
 - (i) Information representation is formal, finite, and definite
 - (ii) Tokens, texts, and types defined
 - (iii) Mechanical determination of types is an imprecise notion
 - (iv) Texts are finite in breadth and depth
 - (v) Assumptions about texts and schemata
 - (b) Information processing
 - (i) Information processing is formal, finite, and definite
 - (ii) Mechanical processing
 - (iii) Deterministic and indeterministic processing
 - (iv) Termination is definite
 - (c) Algorithms and programs

C. Need for Alternative Models

- 1) In natural computation
 - (a) Definition
 - (b) Why TM is an inappropriate model
- 2) In analog computation
 - (a) Representations are continuous and poorly modeled by discrete spaces
 - (b) Processing is continuous and poorly modeled by discrete operations
- 3) In nanocomputation
 - (a) Error and noise are major, unavoidable factors
 - (b) Typically there are not well defined steps (asynchronous, continuous-time parallelism)
 - (c) Operations may be reversible

III. Alternative Frames of Relevance

A. Natural computation

- 1) Relevant issues & assumptions
- 2) Neural computation
- 3) Emergent computation

B. Nanocomputation

IV. Computation in General

A. Traditional notions of digital and analog computation

B. Definition of computation

How, then, can we define "computation" in sufficiently broad terms?

- 1) The essence of computation is manipulation of form independently of material instantiation
- 2) A computation must be physically realized, but it is independent of specific physical realization
- 3) Therefore, in nature or technology, a system may be considered computational if its purpose is manipulation of form independent of matter
- 4) Formally, a system is computational if its purposes is the mathematical manipulation of mathematical objects
- 5) The use of "purpose" is problematic, but reasonably well defined in technology and biology
- 6) Many systems, especially in biology, are not purely computational

C. Autonomy (informal)

D. Transduction

E. Classification of computational processes

F. Realization as homomorphism (informal)

G. Approximate realization

V. Expanding the Range of Physical Computation

A. General guidelines: matching natural processes and computations

- 1) The vicious cycle of conventional hardware and software technologies
- 2) In principle, any reasonably controllable, mathematically described physical process can be used for computation
- 3) Some requirements: speed, transducers, stability, etc.
- 4) To be able to use a wide variety of physical processes, we need to develop computational methods matched to mathematics of the process
- 5) We need a library of well-matched computational methods and physical technologies
- 6) General-purpose computation
 - (a) To make better use of computational methodologies and technologies, it is useful to have general-purpose computers in which computational process is controlled by easily modifiable parameters
 - (b) Notion of "universality" must be consistent with appropriate frame of relevance
 - (c) There has been work in GP analog computers, GP neural nets, etc.
 - (d) The "programs" need not be language-like (e.g., could be image-like)

B. Natural Computation

- 1) There is much that we can learn from computation in nature
- 2) The brains of humans as well as simpler animals have taught and will teach us much about natural computation
- 3) Emergent computation in populations of insects and microorganisms also has much to offer, especially for robustness, adaptability, etc.

C. Nanocomputation

- 1) Computing with a high probability of error and noise
- 2) Massive, fine-grain, asynchronous, continuous-time parallelism as the norm
- 3) Entropy and reversibility

D. Quantum and Quantum-like Computation

- 1) Our investigation of the possibilities of quantum computation should include the full representational capacity of the wave-function, although the problem of quantum incoherence will remain.
- 2) The problems of quantum incoherence may be avoided by quantum-like computation, that is, computation in other physical systems realizing the same mathematics

VI. Conclusions