

Knowledge-based systems in Bioinformatics, 1MB602

Lecture 12: Cellular Automata

Lecture overview

- Cellular automaton
- *The game of life*
- Bioinformatics examples
- Summary of the course

Cellular automaton

- Idealization of a physical system
 - Space and time are discrete
 - Physical quantities (states of the automaton) take only a finite set of values
- Invented by von Neumann (late 1940s)
- Extract the abstract mechanisms leading to **self-reproduction** of biological organisms
 - a system having the capability to produce another organism of equivalent complexity with only its own resources
- Can mimic or simulate physical system

Cellular automaton

- Fully discrete universe made up of cells
- Each cell characterized by an internal state
 - Consists of a finite number of information bits
- The system of cells evolves in discrete time steps
- The rule determining the evolution of this system is the same for all cells
 - A function of the states of the neighboring cells
- The activity of cells takes place simultaneously
 - The same clock drives the evolution of each cell
 - The updating of each cell's internal state occurs synchronously

The game of life (John Conway, 1970)

- Motivation:
 - Find a simple rule leading to complex behaviors
- Representation:
 - 2-dimensional square lattice
 - Each cell can be either alive (state 1) or dead (state 0)
- Updating rules:
 - A dead cell surrounded by exactly three living cells gets back to life (in the next timestep)
 - A living cell surrounded by less than two or more than three living cells dies of isolation or overcrowding



von Neumann neighborhood



Moore neighborhood

Used by Conway

The game of life

- The R-pentomino pattern gets stabilized after 1103 iterations



- Still life objects (examples)



- Oscillators (examples)



- Gliders (example)



The game of life

- *The game of life* is one of the simplest examples of what is sometimes called **emergent complexity** or **self-organizing systems**
- It is the study of how elaborate patterns and behaviors can emerge from very simple rules
 - E.g. explains how the stripes on a zebra can arise from a tissue of living cells growing together
- In Nature we do *not* know all the rules. *The game of life* lets us observe a system where we do
- What can be learnt? E.g.
 - Behavior of cells or animals can be better understood using simple rules
 - Behavior that seems intelligent, such as we see in **ant colonies**, might just be simple rules that we don't understand yet
- More: <http://www.math.com/students/wonders/life/life.htm>



THE LINNAEUS CENTRE FOR BIOINFORMATICS
<http://www.lcb.uu.se>

Cellular automata – theoretical capabilities

- *The game of Life* is a CA capable of universal computations
 - A computer can be built inside the Life "universe"
 - Streams of gliders can be used to send information just as electrical signals are used to send information in a physical computer
 - These streams of gliders can react in a way to perform all of the logical functions on which a modern computer is based
 - Several interesting special-purpose computers have been constructed
 - Possible to mimic any computation process
- CAs have the capacity to be non-restrictive computational technique



THE LINNAEUS CENTRE FOR BIOINFORMATICS
<http://www.lcb.uu.se>

Cellular automata

- Provide simple models of complex systems
 - A collective behavior can emerge out of the sum of many, simply interacting, components
 - A global behavior may obey new laws that are not obviously extrapolated from the individual properties
 - The whole is more than the sum of all parts
- CAs can do more than just behaving similarly to natural dynamical processes
 - Can represent actual models of a given physical system leading to macroscopic predictions
 - The macroscopic behavior of many systems is quite disconnected from its microscopic reality
 - E.g. flows of fluid and gas are very similar at a macroscopic scale, in spite of their microscopic nature
- Many physical processes are well suited to the cellular automata approach
 - Pattern formation, growth phenomena, etc.



THE LINNAEUS CENTRE FOR BIOINFORMATICS
<http://www.lcb.uu.se>

Four classes of CA

1. After a finite number of iterations, the CA achieve a unique state from nearly all possible starting conditions
2. The CA creates patterns that repeat periodically or are stable
3. From nearly all starting conditions, the CA leads to aperiodic-chaotic patterns
4. After a finite number of steps, the CA dies

The game of life corresponds to 2 and 4



THE LINNAEUS CENTRE FOR BIOINFORMATICS
<http://www.lcb.uu.se>

State transition rules

- The behavior of an CA is a function of the initial conformation and the transition rules
- Although the rules are simple, simulation is the only way to determine the CA's behavior
- Second-order rules
 - Use the historic state behavior of cells (current and neighbors) to compute a new state for the current cells
 - Short-term memory in the decision making
- Probabilistic rules
 - State changes are executed according to a probability
 - Could choose from a number of state changes based on their probability
 - CA behaves in a stochastic rather than deterministic manner



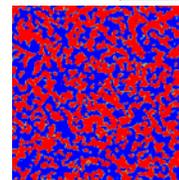
THE LINNAEUS CENTRE FOR BIOINFORMATICS
<http://www.lcb.uu.se>

Growth model

Twisted majority rule (G. Vichniac)

- Each cell considers its Moore neighborhood (2 dimensions) and computes the sum of the cells having a value 1
- The new state of each cell is determined from its local sum according to

sum	0	1	2	3	4	5	6	7	8	9
state	0	0	0	0	1	0	1	1	1	1



B. Chopard, La Complexité,
<http://theory.physics.unige.ch/complexity/>



THE LINNAEUS CENTRE FOR BIOINFORMATICS
<http://www.lcb.uu.se>

Competition models and cell differentiation

- Competitive dynamics
 - Cells compete for some resources at the expense of their nearest neighbors
 - A winner is a cell of state 1 and a loser of state 0
 - No two winner cells can be neighbors
 - Any loser cell must have at least one winner neighbor
- Direct application in biology
- Development of drosophila
 - 25% of the cells forming the embryo are evolving to the state of neuroblast (that develop into neurons)
 - How can we explain this differentiation and the observed fraction?
 - At the beginning, all cells are assumed equivalent

THE LINNAEUS CENTRE FOR BIOINFORMATICS
<http://www.lcb.uu.se>

Competition models and cell differentiation

- Competition takes place between adjacent biological cells
 - Each cell produces some substance S
 - The production rate is inhibited by the amount already present in the neighboring cells
 - Differentiation occurs when a cell reaches a level of S above a given threshold
- Hexagonal lattice (5 neighbors + current)
 - Approximation of the cell arrangement in drosophila embryos
 - S can be either 0 (inhibited) or 1 (active)

THE LINNAEUS CENTRE FOR BIOINFORMATICS
<http://www.lcb.uu.se>

Competition models and cell differentiation

- Rules
 - A $S=0$ cell will grow (i.e., turn to $S=1$) with probability p_{grow} provided that all its neighbors are 0
 - A cell in state $S=1$ will decay (i.e., turn to $S=0$) with probability p_{decay} if it is surrounded by at least one active cell
- Evolution stops when no $S=1$ cell feels any inhibition and when all $S=0$ cells are inhibited by their neighborhood
 - Cells with $S=1$ are those which will differentiate
- What is the expected fraction of these $S=1$ cells in the final configuration?

THE LINNAEUS CENTRE FOR BIOINFORMATICS
<http://www.lcb.uu.se>

Competition models and cell differentiation

- What is the expected fraction of these $S=1$ cells in the final configuration?
 - The maximum value is $1/3$ according to the inhibition condition
 - The minimum value is $1/6$ corresponding to a situation where all active cells are surrounded by 5 inhibited cells
- CA simulations show that the fraction of active cells s when the stationary state is reached is
 - $0.23 \leq s \leq 0.24$
 - Almost irrespectively of the values chosen for p_{grow} and p_{decay}
- Cell differentiation can be explained by a geometrical competition
 - No need to specify the inhibitory couplings of adjacent cells and the production rate

THE LINNAEUS CENTRE FOR BIOINFORMATICS
<http://www.lcb.uu.se>

CA model for enzyme kinetics (Kier et al., 1996)

- Model the reaction between an enzyme and substrate in water
- 110×110 grid of cells (12100 cells)
 - Can take values of E (enzyme), S (substrate), P (product), and W (water)
 - 69 % covered with water, 31% space
 - Added ingredients replace water
- Each cell has a probability associated with its movement and its interaction with other molecules
 - Enzymes can interact with S , P , and W , but not E
 - Extended von Neumann neighborhood (2 steps in each direction)

THE LINNAEUS CENTRE FOR BIOINFORMATICS
<http://www.lcb.uu.se>

CA model for enzyme kinetics

- Affinities of certain molecules to other molecules
 - Determined by a probability of joining, breaking, and movement
- CA run with 50 enzyme cells and variable amount of substrate
 - Runs of 100 iterations were repeated 100 times and results were averaged
- The automaton agreed with well-established equations
- New insight: Affinity of substrate to water was more important than the affinity between the enzyme and substrate!
 - A lower affinity between the substrate and water led to increased conversion of substrate to products

THE LINNAEUS CENTRE FOR BIOINFORMATICS
<http://www.lcb.uu.se>

Simulation of an apoptosis reaction network (Siehs et al. 2002)

- Simulation of the molecular reaction pathways of apoptosis (cell death)
 - Potentially greater understanding of the mechanisms of cancerous cells
 - Cancer is often characterized by the inhibition of the apoptosis process
- 2-dimensional grid
 - Complex data structure at each cell
 - Registers storing variables relating the current state of the molecules within the cell and its surrounds
- Each of the cells could be in a large number of states due to combination of parameters in registers

Simulation of an apoptosis reaction network

- Registers:
 1. Type of molecular object(s) occupying the site
 2. Reaction rate constants for the occupying objects
 3. Molecular neighborhood (Moore). Types of molecules
 4. Distribution of local momentum (hard sphere collision model)
 5. Potential energy status of the molecules at the site (function of the attraction/repulsion of molecules on the site and in the neighborhood)
 6. Molecular reaction lists (what occurs if two molecules occupy the same site?)
 7. Reaction product lists (products of the reactions in 6)
 8. Moved direction (location of each of the molecules at time $t+1$ given information in 4 and 5)

Simulation of an apoptosis reaction network

- Each time the CA was updated the following steps were performed
 1. Evaluation of molecular collisions and redistribution of kinetic energies
 2. Propagation of type information from cells to register
 3. Computation of the local potential energy situation
 4. Evaluation of chemical reactions
 5. Computation of the grid positions of the molecules in the next timestep
 6. Full update of the grid based on 1-5

Simulation of an apoptosis reaction network

- Results:
 - Authors found a set of experiments where changes in concentration of certain proteins could affect the apoptosis
 - An equilibrium existed between several proteins
 - Confirmed what was known experimentally
 - Replicated expected results for different sets of stimuli
- Able to simulate the process of apoptosis under a number of artificial conditions with a small computational requirement

References

- E. Keedwell, A. Narayanan, Intelligent bioinformatics: the application of artificial intelligence techniques to bioinformatics problems. Chichester : John Wiley, cop. 2005
- B. Chopard, La Complexité, <http://theory.physics.unige.ch/complexity/>

Knowledge-based systems in Bioinformatics, 1MB602

Major topics

Scheme

- Programming in Scheme
 - Abstraction
 - Higher order function
 - Message-passing (object-oriented programming)
 - ...
- Concepts
 - Evaluation
 - Substitution model
 - Iterative versus recursive processes
 - Functional versus imperative programming

Knowledge representation and inference

- Knowledge bases
 - Properties
 - Types: Logics, ontologies
- Logics: propositional and first order logics
 - Syntax
 - Semantics
- Inference
 - Truth tables
 - Inference rules
 - Natural deduction proofs

Search

- Search representation
 - States
 - Edges
 - Costs
 - State list
- Uninformed search
- Informed search
 - Heuristics
- Local optimization
- Time/space complexity

Probabilistic approaches

- Bayes' rule
- Machine learning/supervised learning
- Bayesian (belief) networks
 - Representation
 - Reasoning

Genetic algorithms and programming

- Representation: chromosomes, populations, generations
- Genetic operations
 - Selection
 - Crossover
 - Mutation
- Objective functions / Multiple objective functions
- Genetic programming

Machine learning

- Decision trees
 - Algorithm
 - Information gain
 - Pruning
- Neural networks
 - Representation: Input function/activation functions
 - Optimal network structures
 - Perceptrons: linearly separable problems, learning
 - Multi-layer networks: back-propagation algorithm
- Principles
 - DT versus NN
 - Overfitting
 - Evaluation: cross validation

Cellular automatas

- Representation: cells, rules, discrete time
- *The game of Life*
- Purpose/power
- Possible applications

