PHY 411-506 Computational Physics II

Chapter 12: Interdisciplinary Topics

Lecture 7

Monday April 21, 2008

Lecture Outline

Cellular Automata

- One of the first *Cellular Automaton* models was John von Neumann's *self-reproducing automaton* which he constructed in 1948 in an attempt to explain biological reproduction.
  - ◇ A copy of *Theory of Self-Reproducing Automata* by John von Neumann, edited and completed by Arthur W. Banks is available online
- The model had many interesting features, including the fact that it was equivalent to a universal Turing Machine.
  - ◇ MathWorld Turing Machine
- The most famous cellular automaton is probably Conway's Game of Life.

Basic ingredients of cellular automaton models

- Cellular automata model dynamical systems using discrete approximations including
  - ◇ Continuous space x, y, z is replaced by a finite number of *cells* fixed in space, usually in a regular array or lattice.

◇ Continuous dynamical functions are also approximated by a discrete set of values at each cell site.

◇ Continuous time t is made discrete.

◇ The dynamical equation of motion is replaced by a *local rule*: at each time step, cell values are given new values which depend on the cell values in a small *local neighborhood*.

◇ The cell values are updated *simultaneously* or *synchronously*

One-dimensional Cellular Automaton

- A simple 1-d cellular automaton consists of cells arranged in a line. Each cell is assumed to have two values which can be labeled with the binary digits 0 and 1. This is therefore called a *Boolean* cellular automaton.

- A simple choice for the neighborhood of a cell is the cell itself and its two nearest neighbors.

- The next value of a cell depends on the values of its neighboring cells. Since each neighbor can take 2 values, the total number of values of the neighbors is $2 \times 2 \times 2 = 8$. An example of a local rule is

```
    t:  111   110   101   100    011    010    001    000
-----------------------------------------------------------
t + 1:   0     1     0     1      1      0      1      0
```

- The total number of such rules is $2^8 = 256$, which is the number of different 8-digit binary numbers, i.e., the number of different possibilities on the second line.
  - ◇ The above rule is called rule 90 from the decimal representation of the second line

$$
\begin{aligned}
01011010 &= 0 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 \\
&\quad +0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 \\
&= 64 + 16 + 8 + 2 = 90 \ .
\end{aligned}
$$

- Wolfram studied analyzed cellular automaton models like this one and concluded that there are 4 possible types of behavior for such automata:
  - ◇ **Limit point behavior:** The cell values tend to a unique fixed state independent of the initial state.
  - ◇ **Limit cycle behavior:** Stable periodic structures emerge.
  - ◇ **Chaotic behavior:** The time evolution is non-periodic.

- ◇ **Complex behavior:** Complex and localized propagating structures are formed.
  - ○ Can perform Universal Computation!

Conway's Game of Life

- The basic features of cellular automata are nicely illustrated by the *Game of Life* automaton, which was invented by the mathematician John Conway in 1970.
  - ◇ For a simple introduction, see What is the Game of Life?
- This can be thought of as a model of an ecological system of creatures living on a 2-D substrate.
  - ◇ Space is discrete: the cells are taken to form a 2-D square grid.
  - ◇ Each cell can be in one of only two states:
    - ○ dead (0), or
    - ○ alive (1).
- This makes it a *Boolean cellular automaton*.

◇ The local neighborhood of a cell is taken to be the *Moore neighborhood*:

```
X  X  X
X  O  X
X  X  X
```

which consists of the cell itself and 8 surrounding neighbors. Von Neumann's original automaton used the *von Neumann neighborhood*

```
   X
X  O  X
   X
```

◇ The cell value is updated by counting the number of live neighbors, which can take values $0, 1, \ldots, 8$:

```
t             t + 1
-    -------------------------
0 -> 0  0  0  1  0  0  0  0  0
     0  1  2  3  4  5  6  7  8  <= No. of live neighbors
1 -> 0  0  1  1  0  0  0  0  0
```

- The Game of Life Java Applet implements these rules.

- This automaton has many fascinating properties including a variety of *life forms* and the fact that it is capable of universal computation!
  - ◇ Alan Hensel's Life Patterns
  - ◇ A Turing Machine implemented using the Game of Life