CPS331 Lecture: Biologically-Inspired AI; Brooks Discussion

last revised Nov 16, 2016

Objectives:

1. To overview other biologically-inspired approaches to learning

Materials:

1. Brooks Discussion
2. Floreano book to show
3. Projectables of Floreano Figures 2.1, 2.2
4. Game of Life Demo
5. CA Maze demo with Floreanou Figure 2.23 problem
6. Projectable of Wolfram Rule 30 and Cone Snail that exhibits this
7. CA Traffic demo
8. Projectables of Floreano Figures 2.18, 2.30 left column, right column
9. Projectable of Floreano Figure 4.1
10. Projectable of Floreano Figure 4.11
11. L-System demo program
12. Boids demo (or use web)
13. Projectable of Figure 1 from "Swarm Intelligence" *CACM* article

I. **Introduction**

   A. So far, we have talked about several approaches to learning, including various symbolic approaches and two inspired by biology (Genetic Algorithms and Neural Networks).

   B. We have also looked at the view espoused by Fogel to the effect that symbolic AI has led the field down a wrong path.

   1. Recall that Fogel created a program that learned to play expert-level checks by making use of a genetic algorithm to learn the weights for a neural network that performs static evaluation in a min-max game tree algorithm!

   2. For today, you read a paper by Rodney Brooks

   Brooks discussion

C. We have looked at two biologically-inspired approaches to learning, and will look at four more today. What all have in common being loosely inspired by what we find in living systems.

  1. When we say "inspired" by biological systems, we mean something very different from "simulating" biological systems. While there is certainly some value in simulations of life, the work we have considered recently and are considering here builds loosely on principles observed in nature, without in any sense attempting to produce simulations of nature, and often includes strategies that are significantly different from what is found in nature.

  2. Moreover, the approaches we are considering here deal view "intelligence" quite broadly, as we shall see.

     a) It may well be that there are some fundamental idea underlying intelligence that we just don't see at this time.

        Rodney Brooks put it this way: "My hypothesis is that we may simply not be seeing some fundamental mathematical description of what is going on in living systems. Consequently we are leaving out the necessary generative components to produce the processes that give rise to those descriptions as we build our artificial intelligence and artificial life models"

        Brooks elsewhere refers to this concept ("provocatively", he admits) as "the juice".

        (*Flesh and Machines* p. 188)

     b) The different approaches we will look at today are basically various ways of looking at what is going on in living systems - not paths to achieving general artificial intelligence per se, but things we need to better understand.

c) A common thread that runs through them is the notion of complexity arising from simplicity. It has certainly been true in other fields that fundamental understandings involve a simple, clean, elegant explanation for what is otherwise complex.

Example: A key factor in a heliocentric description of the solar system replacing a geocentric one was that it made possible a clean and elegant description of the observed motions of the planets without resorting to inelegant notions such as "epicycles".

3. The "Embodied AI" topic which will be addressed by one of the groups also falls in this area - in fact, one of the key figures in this field is Rodney Brooks.

4. Much of what I will be sharing in this lecture comes from: *Bio-Inspired Artificial Intelligence* by Dario Floreano and Claudio Mattiussi (MIT Press, 2008)

SHOW book

II. **Cellular Automata**

A. As you know (and Newell and Simon reminded us) the basic building block of living systems is the cell.

1. Complex living creatures are composed of huge numbers of cells. For example, there are more cells in your little finger than there are human beings in the whole world.

2. The functioning of a cell may be influenced by its immediate neighbors through chemical or electrical interaction. All interaction is local - even transmission of messages from the brain is actually handled by passing a message along from cell to cell along the a nerve.

B. A cellular automaton is a computational system consisting of an array of basic building blocks or "cells".

1. Commonly, the cells are arranged in a two-dimensional pattern, though it is also possible to create models of other dimensionality.

2. Commonly cells are square or cubical, though other shapes are possible.

   PROJECT: Floreano Figure 2.1

C. Each cell is in a particular state drawn from a set of possible states. (Or, in some cases, the state may be a real number)

1. Each cell periodically updates its own state based on the states of its neighbors - where various definitions of "neighbor" are possible.

   PROJECT: Floreano Figure 2.2

   (In the case of boundary cells, some mechanism must be specified to deal with the absence of certain neighbors)

2. Typically, all cells update their states synchronously at regular time intervals. (This, of course, differs from what occurs in living systems.)

D. One well-known example of a cellular automaton is Conway's Game of Life.

1. "Life" involves a 2-dimensional grid of square cells.

2. Each cell is in one of two states: living, or dead.

3. Each cell is regarded as having eight neighbors (the Moore neighborhood).

4. The boundary cells are always considered dead.

5. At regular intervals, each non-boundary cell updates itself according to the following rules:

   a) If the cell is living, it dies if it has fewer than 2 or more than 3 living neighbors (dies of loneliness or dies of overcrowding)

   b) If the cell is dead, it becomes living (is "born") if it has exactly three living neighbors.

6. "Life" is fascinating because it exhibits many of the properties of living systems, including reproduction.

   DEMO: Various options in Game of Life Demo (run Fast - switch Glider Gun to Hyper)

E. Another example (taken from the Floreano book)

1. Consider a maze like the following

   PROJECT Maze - load Floreano figure 2.23

   a) When we talked about search, we talked about how a maze like this might be solved by using a search algorithm.

   b) We now want to consider how we might solve a maze like this using purely local computation with Cellular Automata

2. How would we go about creating a CA solution to this maze?

   a) We might begin by defining a cell as having two states - (potentially) part of a solution path, or not part of a solution path.

(1) Initially, all the maze cells are considered to be potentially part of a solution path - but this will change as the solution progresses.

(2) Of course, from the beginning, the wall cells are not part of a solution path.

(3) We will show these as as gray and black in the diagram.

b) We could define an update process in terms of the four immediate neighbors of a cell (the Von Neumann neighborhood).

(1) A cell that is known to not be part of a solution path always remains such.

(2) What about a cell that is potentially part of a solution path?

Observe that a cell that is potentially part of a solution path, but has three (or four) neighbors that are not, represents a dead end.

(Point out examples)

What update rule might this suggest?

ASK

Change a potential solution cell to a not solution cell if it has three or more not solution neighbors

(3) This will result in potentially solution cells that cannot be part of the solution turning into not solution cells. Eventually, the only potential solution cells left will be the ones actually comprising the solution.

DEMO

F. One fascinating characteristic of much of nature is how very complex systems can arise from very simple models.

1. This can be illustrated by what is essentially the simplest possible form of Cellular Automaton - a 1-Dimensional Cellular Automaton having the following characteristics:

   a) Each cell is in one of two states - off (0) or on (1).

   b) All cells update simultaneously.

   c) The new value of each cell is calculated based on the value of the cell and its two neighbors on its immediate left and right.

2. Consider such a Cellular Automaton having the following update rule. (The rule specifies the new value of the cell based on the three values of its left neighbor, itself, and its right neighbor - hence it must consider 8 possibilities)

   111 -> 0
   110 -> 0
   101 -> 0
   100 -> 1
   011 -> 1
   010 -> 1
   001 -> 1
   000 -> 0

3.

   a) As an aside, how many such rules are possible?

   ASK

   $2^8 = 256$

   (There is a system for encoding such rules called Wolfram encoding. For example, the above is called rule 30 since 00011110_2 is 30_{10}.)

b) Consider how a system which starts out with a single 1 evolves over time. This can be diagrammed in two dimensions by representing each state by a single row.

Initial: 0 0 0 1 0 0 0
After one step: 0 0 1 1 1 0 0
After two steps: 0 1 1 0 1 1 0

c) Long term result:

PROJECT Wolfram Rule 30

d) Interestingly, such systems sometimes occur in nature

PROJECT Cone Snail

G. So what are some applications of Cellular Automata?

1. A picture can be encoded by a cellular automaton with one cell per pixel, with its state being the intensity of the pixel (hence one of 256 possible values if using standard gray-scale.)

Basic low-level vision operations - such as filtering and edge detection - can be defined as cellular operations involving a suitable neighborhood.

2. Various kinds of systems can be modeled by cellular automata.

a) Traffic

(1) Consider the simple case of traffic moving in one direction on a one-lane road.

(2) This can be modeled by a CA in which a car moves into a vacant cell in front of it, but otherwise remains where it is.

(a) A cell is in one of two states: vacant or occupied

(b) Cells use the following update rule

if a cell is vacant, it becomes occupied if the cell on its left is occupied; otherwise it stays vacant

if a cell is occupied, it becomes vacant if the cell on its right is vacant; otherwise it stays occupied

(Where cells are considered to wrap around so that the rightmost cell is considered "to the left of" the leftmost cell)

(3) DEMO - observe

(a) Traffic takes some time to settle into a steady state after density change

(b) Traffic flows smoothly at densities $<= 0.5$, but begins slowing down as soon as density exceeds 0.5. That is, the model predicts a behavior that is actually quite observable in practice!

(c) At densities $< 50\%$, traffic appears to be moving left to right; at densities $> 50\%$, "gaps appear to be moving right to left

b) Forest fires

PROJECT Floreano Figure 2.18

c) Social Interaction

(1) Segregation

PROJECT Floreano Figure 2.30 left column

(2) Suspicion

PROJECT Floreanou Figure 2.30 left column

III. **Developmental Systems**

A. The body of any living creature, though composed of a huge number of individual cells, develops from a single cell formed at conception. The genetic code that guides the development of the body is contained in the cell's DNA, which is present (identically) in every cell of the body.

B. The DNA code is actually remarkably compact.

   1. Human DNA contains about 20,000 - 25,000 genes, but the human body contains on the order of 50-100 trillion cells (approaching 10^{14}), of roughly 210 different types.

   2. A relatively small number of genes result in the development of these trillions of cells. That is, the DNA does not contain the "blueprints" for a body, but rather instructions that result in the development of a body.

   3. That is, the body is an incredibly complex system that is specified by a relatively codes.

C. In 1968, the biologist Aristid Lindenmayer proposed a class of formal systems that came to be know as Lindenmayer systems or L-Systems for short.

   1. An L-System includes

      a) An alphabet of symbols.

      b) An initial string of symbols (called the axiom)

      c) A set of production rules

      d) A stopping condition

2. The production rules are applied repeatedly to the produce new strings of symbols from the starting string, until the stopping condition is met.

3. The notation resembles the formal grammar notation we discussed in conjunction with natural languages.

4. For example, the following L-System models the development of bacterial filaments

   Alphabet: $\{ g_r, g_l. d_r, d_l.\}$

   Axiom: d_r

   Production rules: $\{$ $d_r \rightarrow d_l g_r$

   $\qquad\qquad\qquad d_l \rightarrow g_l d_r$

   $\qquad\qquad\qquad g_r \rightarrow d_r$

   $\qquad\qquad\qquad g_l \rightarrow d_l \}$

   Repeated application of these rules results in the development of a system that resembles a bacterial filament

   PROJECT Floreano figure 4.1

D. An L-System can be be linked to a graphical representation. For example, a system might include the following in its alphabet

   F = draw a line of some specified step size
   f = move the pen the same distance, without drawing a line
   + = turn the drawing position left by some specified angle
   - = turn the drawing position right by the same specified angle
   [ ] = save/restore the drawing position

E. An L-system might be stochastic - that is, it might have several rules that can be applied in the same situation, with relative probabilities so that the particular rule that is applied is randomly chosen from the possibilities.

F. With these capabilities, as very simple L-System can result in a very complex drawing.

Example: Consider the result of applying an L System with the above alphabet, axiom F, and the following productions

F → F [+ F] F [- F] F with probability 1/3
F → F [- F] F [+ F] F with probability 1/3
F → F [- F F - F] F with probability 1/3

PROJECT Floreano Figure 4.11

G. Here are some examples of other possibilities

DEMO: L-System Demo program

IV. Artificial Immune Systems

A. Vertebrates have sophisticated immune systems that allow them to cope with pathogens (viruses) that might otherwise kill them. Vertebrate immune systems are composed of two general types of cells

1. Detectors which recognize pathogens that need to be destroyed. There are many different types of detectors to allow for detecting many different types of pathogen

2. Effectors that actually destroy the pathogens.

B. A key component of the challenge for our immune system is that its detectors must recognize pathogens that it has never "seen" before, while not attacking cells that are normal parts of our bodies.

1. There is a whole class of diseases known as autoimmune disorders which involve the immune system falsely construing normal cells as pathogens, so that the body's immune system attacks the body itself.

2. When the immune system first encounters a new pathogen, it takes considerable time to mobilize a response against it, because only a few of the detectors will actually recognize it. (And the time required may allow the pathogen to overwhelm the body's defenses, possibly killing the individual)

3. But the immune system includes a "memory" capability which allows it to respond much more quickly to a pathogen that it has encountered previously. This is what makes vaccines useful. A vaccine activates the "memory" of the immune system so it will recognize the pathogen much more quickly the next time it sees it.

C. There is a whole area of research centering on the use of ideas from the vertebrate immune system to protect computer systems and networks from pathogens (i.e. computer viruses). Here the aim is to equip systems with a capability to recognize and respond quickly to things like dangerous network traffic that has never been seen before, but without waiting for human intervention.

V. **"Swarm" Intelligence**

A. There are many places in the natural world where groups of individuals exhibit a complex behavior that emerges from the behavior of individuals.

1. Examples: flocking of birds, schooling of fish, herd behavior of animals, food-seeking behavior of insects like ants and bees, etc.

2. At first glance, this behavior looks like it requires some kind of "central coordinator". But, in fact, there is no such role.

   Instead, these systems are examples of what is sometimes called "self-organizing" or emergent behavior.

B. For example, Craig Reynolds showed in 1987 that it is possible to generate the flocking behavior of birds (or similar behavior in fish and animal herds) by simple rules.

1. Each individual bird's motion is governed by three simple rules:

    a) Separation: steer to avoid crowding local flock-mates

    b) Alignment: steer toward the average heading of local flock-mates

    c) Cohesion: steer toward the average position of local flock-mates

2. If each individual bird behaves according to these three rules, flocking behavior for the whole flock will emerge naturally.

    a) DEMO: Program or `http://www.red3d.com/cwr/boids/applet/`

    b) Techniques like this have been used by the film industry for films like *The Lion King*, *Batman Returns* and *Star Trek*. (Floreano p. 533)

3. More complex examples include such things as the nest-building behavior of ant colonies and the high-building behavior of bees.

    a) One place where how this occurs is of interest in CS is in conjunction with the design of distributed systems using distributed, as opposed to central coordinator, control.

    b) Another place where this is of interest is "swarm robotics" - the use of swarms of simple robots in place of a single complex robot.

C. Now consider the food-seeking behavior of ants.

  1. The following description of this appeared in an article in the "Swarm Intelligence" (*Communications of the ACM* 45:8, August, 2002) (p. 64)

  "Although the capabilities of a single ant are very limited, ants can collectively establish the shortest route between a source of food and their nest, and efficiently move the food to their home. Ants communicate with each other through the use of pheromones, chemical substances that attract other ants. As the ants move, they lay down a trail of these pheromones that other ants can follow. Ants move randomly, but when they encounter a pheromone trail, they decide whether or not to follow it. If they do so, they lay down their own pheromone on the trail as well, reinforcing the pathway. The probability that an ant chooses one path over another increases with the amount of pheromone present. The more ants use a given trail, the more attractive that trail becomes to other ants. If a colony of ants is presented with a short path and a long path to a source of food, they will first use both paths in equal numbers, laying down pheromones as they move. But the ants taking the shorter path will return to the nest first. The shorter pathway will then be doubly marked with pheromone, and will be more attractive to those ants that return to the food source. This is illustrated by Figure 1, which shows a distribution of ants over a set of pathways between a nest and a food source over time. Early on, the ants are equally distributed, but eventually they favor the shorter route. There is, however, always a chance an ant will not follow a previous or well-marked trail. This allows for randomness and exploration, which is beneficial in that it allows for discovery of shorter or alternate pathways, or new sources of food. The pheromones also evaporate over time. Given this situation, pheromones will become less detectable on longer trails, since these trails take more time to traverse. The longer trails will hence be less attractive, another benefit to the colony as a whole."

  PROJECT Figure 1 from Article

2. This gives rise to an optimization technique called "Ant Colony Optimization" described in the same article

"Ant colony optimization (ACO) is a meta-heuristic that uses artificial ants to find good solutions to difficult combinatorial optimization problems. The behavior of artificial ants is based on the traits of real ants, plus additional capabilities that make them more effective, such as a memory of past actions. Each ant of the "colony" builds a solution to the problem under consideration, and uses information collected on the problem characteristics and its own performance to change how other ants see the problem".

VI. Conclusion

Floreano ended the book which I have referred to several times with the following statement: "A careful reader may have noticed that we have not yet defined what intelligence is ... The approaches and examples described in this book show that biology is a bewildering source of inspiration for the design of intelligent artifacts capable of efficient and autonomous operation in unknown and changing environments. ... Proper practice of bio-inspired artificial intelligence requires a *scientific effort* to extract the principles of biological intelligence from the data and theories provided by the biologists, and and *engineering effort* to translate these principles into functional artifacts and technologies ..."