Readings Summary

Papert, Seymour. (1988). "One AI or many?" *Daedalus* 117(1)1-14

- Papert is not a neutral party in the connectionist debate
- Papert addresses, "the nature of artificial intelligence and its appeal to people more interested in the human mind than in building robots"
- Papert speaks of "psychologically relevant" saying, "Artificial intelligence should become the methodology for thinking about ways of knowing"
- Papert urges us to attend to the requirements of specific kinds of tasks, not to the search for universal mechanisms
- Papert argues that what can (and can not) easily be done by parallel and distributed machines is technically non trivial to sort out. The qualities parallel and distributed "are in tension rather than sweet harmony"
- For systems that need to interact with (and survive in) the real world, time often is an important task constraint

On December 5, 2006, while crossing a street in Hanoi, Papert (then aged 78) was struck by a motorcycle and suffered a brain injury. A blog website, Seymour Papert's Recovery, (last updated June 27, 2012) reports, "He [Seymour Papert] is doing well, with round the clock care and rehabilitation that will be needed indefinitely."

Today's Learning Goals

1. To provide examples that demonstrate the challenges that dealing with time raise in computer science
2. To understand *causality* and the difference between causal (i.e., *"on-line"*) and acausal (i.e., *"off-line"*) computation
3. To understand *scalability*, here taken to mean how the time complexity of a computation grows with the size of the problem considered
4. To begin to explore constraints on *embodied* robotic systems so that they can interact with (and survive in) the real world

Examples Involving Time

Activities cannot be scheduled concurrently when they contend for the same resources.

Example 1: Concurrency and Resource Contention

Contrast:

(1.1) John reviewed the paper, played the piano, and vacuumed the floor; he had no time left to cook dinner

(1.2) John sprawled on the sofa, read a magazine, and listened to music; he had lots of time left to cook dinner

Gerevini, 1997

In (1.1), reviewing a paper, playing the piano and vacuuming the floor contend for common cognitive and sensorimotor resources (with each task requiring substantial time to complete). On the other hand, in (1.2), sprawling on the sofa, reading a magazine and listening to music do not contend for common resources and thus are presumed to be activities that can be done in parallel leaving lots of time left to cook dinner.

Example 2: Time Dependent Knowledge

Complete the following: "God save the ____________"

Top ten in the line of succession to Queen Elizabeth II

1. Prince Charles, Prince of Wales
2. Prince William, Duke of Cambridge
3. Prince George of Cambridge
4. Prince Henry of Wales
5. Prince Andrew, Duke of York
6. Princess Beatrice of York
7. Princess Eugenie of York
8. Prince Edward, Earl of Wessex
9. James Mountbatten-Windsor, Viscount Severn
10. Lady Louise Mountbatten-Windsor

Note: The Duke and Duchess of Cambridge are expecting a second child who, upon birth, will be fourth in the line of succession, behind older brother Prince George of Cambridge.

In Canada, it has been "God save the Queen" since 1952. Given the line of succession, it is almost certain that the next monach will be male. At that instant, it will become, "God save the King."

Mathematics, including formal logic, is "monotonic." Progress in mathematics adds to the set of established theorems. But, once a theorem always a theorem. This makes formal logic problematic as a mechanism for knowledge representation and reasoning when one must deal with propositions that are contingent on states of the world that can change with time.

Example 3: Truth Maintenance

Suppose a red object A is near a blue object B. Suppose a robot picks up object A and moves it to a different location

Question: How does the robot "know" that object A is no longer near object B?

Question: How does the robot "know" that object A still is red and object B still is blue?

Answer: A robot doesn't "know" that movement typically alters nearness but preserves colour unless this is represented in the robot's internal model of the world

Alternative answer: A robot (with vision) doesn't need an internal model of the world. If it needs to reason further about colour or nearness, it can always look. That is, let the world be its own model

People have extended formal logic to reason about time and actions. For each action one can explicitly assert rules for each property that can change as a consequence of that action (with the default being that all other properties are preserved). Alternatively, for each action one can explicitly assert rules for each property that is preserved as a consequence of that action (with the default being that all other properties can change). Properties that can change are eliminated from the knowledge base. This elimination process is recursive in that properties that depend on properties that can change also must be eliminated from the knowledge base. Neither option scales well with the size and the complexity of the world modeled, even with the assumption (usually made) that the only source of change (i.e., the only source of actions) is the robot itself.

Time in Distributed Systems

Time in Distributed Systems

> "A system is distributed if the message transmission delay is not negligible compared to the time between events in a single process."

Leslie Lamport,
"Time, Clocks, and the Ordering of Events in a Distributed System,"
Communications of the ACM 21(7)558–565, 1978

Corollary: All (non trivial) systems these days are distributed systems

The systems Lamport had in mind in 1978 were, "a collection of distinct processes which are spatially separated, and which communicate with each other by exchanging messages." The prototypical example in 1978 was the ARPAnet network of computers, the precursor to today's internet. Lamport was prescient enough also to note that, "A single computer can also be viewed as a distributed system in which the central control unit, the memory units, and the input-output channels are separate processes."

Time in Distributed Systems (cont'd)

- *Time (in the small):*
  - clock skew
  - (multi-core) clock synchronization
- *Time (in the large):*
  - networks of computers, GPS
  - reservation systems
  - stock trading
  - on-line shopping
- *Time (in the very large):*
  - outside on a sunny day
  - outside on a starry night

Light travels $\sim$29.979 cm in 1 nanosecond. This is equivalent to $\sim$11.8 inches. Consequently, a nanosecond is sometimes referred to as a light-foot. Admiral Grace Hopper (1906-1992) used to give out pieces of wire about a foot long to illustrate the eventual problem of building very high speed computers. A computer built with parts connected by half this distance (i.e., $\sim$15 cm) of wire, would take at least a nanosecond to send data to a part and get a response.

Clock skew occurs in synchronous circuits when the central clock signal arrives at different parts of the circuit at different times. There are many causes: different wire path length, temperature variation, timing variation in intermediate devices, capacitive coupling, material imperfections, and differences in input capacitance on the clock inputs of devices using the clock. As the clock rate of a circuit increases, timing becomes more critical and less variation can be tolerated if the circuit is to function properly. (See http://en.wikipedia.org/wiki/Clock_skew).

One solution is to use multiple clocks that themselves are synchronized. (See http://en.wikipedia.org/wiki/Clock_synchronization).

At the other extreme, it takes light on average about 8 minutes and 20 seconds to travel from the sun to the earth. Light from the next nearest star (Alpha Centauri) takes 4.37 years to reach the earth. Further afield, M109 is a barred spiral galaxy approximately 83.5 $\pm$24 million light-years away in the constellation Ursa Major.

Aside: CBC Time Signal

November 5, 2014, marked the 75th anniversary of the National Research Council's official time signal, broadcast on CBC radio

> "To determine the official time, the NRC uses atomic clocks, which are instruments that use microwave signals and atoms to provide the most accurate time known in the world…

…They probably gain only a few microseconds in a year…

…The NRC also sends out a time signal through the Network Time Protocol, which is what's used to set time on most personal computers"

The beginning of the long dash indicates…

Newton–Raphson

Isaac Newton (1642 – 1726) and Joseph Raphson (c. 1648 – c. 1715) were contemporary Englishmen. Newton, of course, was a well known physicist and mathematician. Raphson was a less well known mathematician.

Newton-Raphson

Let $f(x)$ be a function defined over the real numbers, x, with derivative $f'(x)$. A root of $f(x)$ is a value of x for which $f(x) = 0$

Let x_0 be a first guess for a root of $f(x)$. Provided the function satisfies certain assumptions, a better approximation, x_1, is given by

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

The process is repeated as

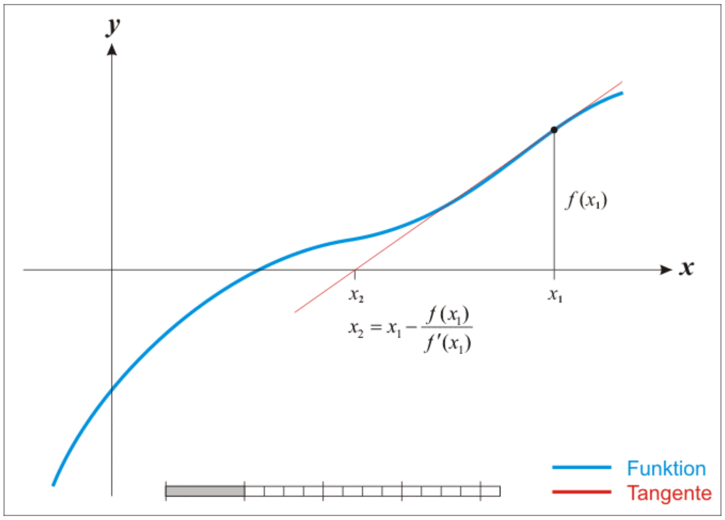
$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

until a sufficiently accurate value is reached

http://en.wikipedia.org/wiki/Newton%27s_method

Newton-Raphson (Geometric Interpretation)

The "slides" version of the notes shows all 18 frames in the animation. Here, only the fifth frame is shown.



http://en.wikipedia.org/wiki/Newton%27s_method

Consider the specific example of calculating the square root of 2. We can apply Newton-Raphson to determine a root of the equation $f(x) = x^2 - 2$. For this equation, $f'(x) = 2x$ and our update rule becomes

$$x_{n+1} = \frac{x_n}{2} + \frac{1}{x_n}$$

The property of most interest is the rate of convergence. The surprising result is that the rate of convergence of Newton-Raphson is quadratic, which means that the number of correct digits roughly doubles at each step.

The following table demonstrates.

Example 4: Newton-Raphson $\sqrt{2}$

The value of $\sqrt{2}$ (55 digit precision) is:

1.414213562373095048801688724209698078569671875376948073

With Newton-Raphson and an initial guess of 1.5, we obtain

Iteration	Estimate	Digits Accuracy
0	*1.5*	1
1	*1.41*666666666666666666666666666666...	3
2	*1.41421*568627450980392156862745098...	6
3	*1.41421356237*468991062629557889013...	12
4	*1.41421356237309504880168*962350253...	24
5	*1.41421356237309504880168872420969*...	48
⋮	⋮	⋮

More Examples Involving Time

Example 5: You be the Judge

The BC provincial government, fearful that there aren't enough lawyers in the province, decides to launch a new private (but accredited) law school named, "Suits 'R Us University." In order to entice students, Suits 'R Us offers the following guarantee to its graduates, "Win your first case or your tuition is free"

Ariel, a COGS alumna, decides to enrol. Law school turns out to be easy for her and she graduates with a high GPA. Suits 'R Us accountants notice, however, that despite repeated letters of reminder and even referral to a collection agency, Ariel never paid any tuition

Suits 'R Us proceeds to sue Ariel for the tuition owing. Ariel represents herself in this her first case. The case comes before the judge

Example 5 (cont'd): You be the Judge

The lawyer for Suits 'R Us approaches the judge and argues, "Your honour, this case is simple. Suits 'R Us is suing for payment of tuition. If you rule in our favour then Ariel must pay. Should you rule in Ariel's favour then she has won her first case. Thus, according to the terms of her agreement with Suits 'R Us, she owes us tuition. Either way, Ariel must pay"

Ariel approaches the judge. She says, "Your honour, I agree with my learned opponent that the case is simple. Unfortunately, opposing counsel has it quite wrong. Suits 'R Us is suing me for payment of tuition. If you rule in my favour then, according to that ruling, I don't have to pay. Should you rule in favour of Suits 'R Us then I will have lost my first case. Thus, according to the terms of the guarantee, I don't have to pay. Either way, I don't have to pay"

Question: How does the judge rule?

If this example were real, it's not clear how the judge would rule. One can imagine different possibilities. Three are:

1. The judge might rule in favour of Ariel simply on the grounds that Suits 'R Us waited too long to try to collect the tuition owing. Suits 'R Us would have been within its rights to have suspended Ariel's registration any year while she was still in the program. But, now that she's graduated there's nothing (in law) that allows them to recover. That is, the judge rules that Suits 'R Us is the author of its own misfortune for failing to act in a timely manner.
2. The judge might rule in favour of Suits 'R Us based on tuition owing (without prejudice with respect to enforcement of the guarantee) and note that using the guarantee to recover tuition is a separate matter to be pursued by Ariel as an independent case (hopefully before another judge).

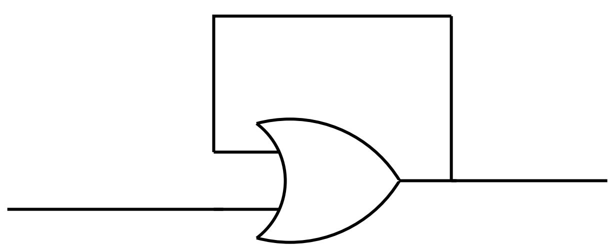
   In this scenario, Ariel will have used her knowledge to her advantage because we would expect her subsequently to recover her tuition based on the terms of the guarantee.
3. Alternatively, the judge might suspend judgment pending the outcome of Ariel's first case. That is, the judge would withhold judgment until there is a first case of record, distinct from this case.

   Should Ariel never complete a "first case" then the judge would never rule (and Ariel will have used her knowledge to her advantage because she would never have to pay). On the other hand, if Ariel were to have a completed "first case" then the judge would then rule on this case based on the outcome of that case.

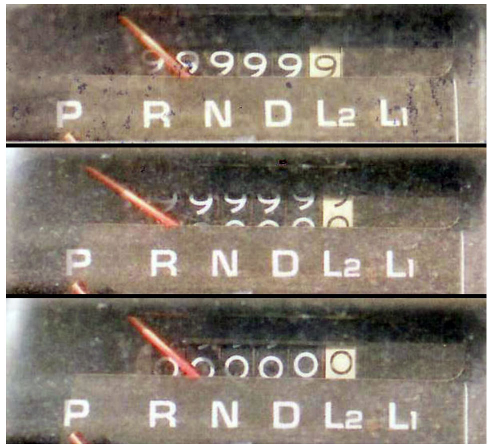
Note that time (and timing) play a key role in each of these three possibilities.

Recall this example from class on September 18, 2014:

What about this?



Example 6: Odometer Problem



http://en.wikipedia.org/wiki/Odometer

In this example, the odometer is "rolling over" from 99,999.9 miles to 00,0000.0 miles. The point to note here is that the roll over depicted requires all 6 odometer digits to change from 9 to 0. Similarly, the earlier roll over from 09,999.9 miles to 10,000.0 miles also required all 6 digits to change. In an odometer, all digits do not change simultaneously. (This certainly was evident in earlier odometers, which were mechanical). Obtaining an odometer reading when some of the digits are at the old value and some at the new can produce widely varying results, far from either the old or the new value. One strategy is to block reading of the odometer until it can be guaranteed that all digit transitions have been completed. Another strategy is to consider a "safe" design such that the odometer can be read at any time with the guarantee that, during a transition, the reading is either the old value or the new value, but never anything else. Both strategies are employed in the design of computer systems.

Aside: The precise determination of a ship's longitude was a "grand challenge" problem of the 18th century. The key was accurate measurement of time (in conditions found on ships at sea). In those days, tampering with a ship's chronometer was a capital offence. In the 20th century, the odometer reading was considered a reliable estimate of an automobile's worth (compared to other automobiles of the same make and year). Tampering with an automobile's odometer, while not a capital offence, was and remains an example of fraud.

Example 7: Toll Booth Problem

Consider an automated automobile toll booth. The gate is normally closed. When a driver makes the appropriate payment, the gate opens to let that automobile proceed. There is no human to decide when to open and close the gate

Question: How long should the gate stay open?

Too Long: Multiple cars can proceed through the gate on a single payment

Too Short: The gate can come down before the car that payed has proceeded through the gate or worse, it can come down on top of the car causing damage (and delays)

Example 7 (cont'd): MIT 6.004 Solution

The entire "Flakey Control Signals" sequence shown in class is posted on UBC Connect

Note that the key observation for the proposed fix for this (automobile) toll booth example is, "At no time is there an open path through both gates."

Example 8: Puck Location and Possession

Andrew Duan's M.Sc thesis (August, 2011) added the automatic determination of puck location and possession into the UBC LCI sports video analysis system

Here's what Andrew's system does with one test hockey video sequence:

Note: The 1000 frame broadcast hockey sequence shown in class is not embedded in the notes

Credit: Xin Duan (Andrew)
Additional credits: Kenji Okuma, Wei-Lwun Lu, Ankur Gupta

One design issue to discuss is the issue of "causality." A causal system is a system where the current output depends only on current and past inputs, not on future inputs. If a computer vision system is causal then, in principle, it can be made to run "on-line," in real-time, provided only that we have computing hardware of sufficient raw power. Conversely, a computer vision system that is not causal (i.e., is "acausal") can never run in real-time, regardless of computing power, since physical reality dictates that we can not build a system whose present output depends on future inputs.

Computer vision systems can be causal or acausal. When we want a computer vision system to interact directly with the world, as is the case in robotics and, for example, image-guided surgery, causality is an important design principle. On the other hand, if we are building a computer vision system for the "off-line" analysis of, say, a large video archive, then there is nothing in physical reality that prevents the system from looking ahead an arbitrary number of video frames in order to better interpret the current video frame.

For more discussion of causality, see `http://en.wikipedia.org/wiki/Causal_system`.

Today's Learning Goals

1. To provide examples that demonstrate the challenges that dealing with time raise in computer science
2. To understand *causality* and the difference between causal (i.e., *"on-line"*) and acausal (i.e., *"off-line"*) computation
3. To understand *scalability*, here taken to mean how the time complexity of a computation grows with the size of the problem considered
4. To begin to explore constraints on *embodied* robotic systems so that they can interact with (and survive in) the real world