Piazza course website is up:
piazza.com/cornell/spring2017/cs4700
The researchers found that when apples were scarce, the agents quickly learned to attack one another -- zapping, or “tagging” their opponent with a ray that temporarily immobilized them. When apples were abundant, the agents preferred to co-exist more peacefully.

Rather chillingly, however, the researchers found when they tried this same game with more intelligent agents that drew on larger neural networks -- a kind of machine intelligence designed to mimic how certain parts of the human brain work -- they would “try to tag the other agent more frequently, i.e. behave less cooperatively, no matter how we vary the scarcity of apples,” they wrote in a blog post on DeepMind’s website.

New research from DeepMind, Alphabet Inc.’s London-based artificial intelligence unit could ultimately shed light on this fundamental question.
Leibo said that the agents used in the apple-gathering and Wolfpack experiments had no short-term memory, and as a result could not make any inferences about the intent of the other agent. “Going forward it would be interesting to equip agents with the ability to reason about other agent’s beliefs and goals,” he said.

In the meantime, it might be wise to keep a few spare apples around.
Artificial intelligence changes the way it behaves based on the environment it is in, much like humans do, according to the latest research from DeepMind.

[...] During the work, they found it is possible for AI to act in an "aggressive manner" when it feels it is going to lose out, but agents will work as a team when there is more to be gained.
So, depending on the situation, having a greater capacity to implement complex strategies may yield either more or less cooperation. The new framework of sequential social dilemmas allows us to take into account not only the outcome of the interaction (as in the Prisoner’s dilemma), but also the difficulty of learning to implement a given strategy.

Self-interested people often work together to achieve great things. Why should this be the case, when it is in their best interest to just care about their own wellbeing and disregard that of others?
Robots

Actors, teachers, therapists - think your job is safe from robots? Think again

Thanks to advances in artificial intelligence, many jobs that weren't considered ripe for automation suddenly are

Dan Tynan in San Francisco

Thursday 9 February 2017 03.00 EST
Is AI a Threat to Christianity?

Are you there, God? It's I, robot.

JONATHAN MERRITT | FEB 3, 2017 | TECHNOLOGY

In his relatively short tenure, Pope Francis has been hard at work welcoming spiritual seekers into the Catholic Church. He’s refused to judge LGBT people, sought to integrate divorced couples, and extended priests’ ability to forgive abortion. But Francis’s wide arcs have arguably never stretched further than a mass in 2014 when he suggested the church would baptize Martians.

"If—for example—tomorrow an expedition of Martians came ... and one says, ‘But I want to be baptized!’ What would happen?” Pope Francis asked. "When the Lord shows us the way, who are we to say, ‘No, Lord, it is not prudent! No, let’s do it this way!’"
“It’s 2017, and the new thing is called AI,” he said. “And it’s more powerful than the Internet, because the Internet didn’t think, the Internet didn’t know you. So that shift is about to happen again, where there is a new type of way that we connect and get information. It’s about to enter society and it knows a lot. It’s coming and it’s going to be awesome.”

**William and Farfetch’s Jose Neves**

Topics ranged from AI to the importance of data to Will.i.am’s regrets about not having studied computer science.

By Lorelei Marfil on February 7, 2017
THE COMING DECADES, artificial intelligence will replace a lot of human jobs, from driving trucks to analyzing X-rays. But it will also work with us, taking over mundane personal tasks and enhancing our cognitive capabilities.
Datamation Hangout:

**Using Artificial Intelligence to Speak with a Lost Friend**

**Event Description:** We talk with an artificial intelligence developer working on what may be the final frontier of AI: recreating the human...

**Date:** Friday, February 10, 2017  
**Time:** 12:00 - 12:30 PST  
**Location:** Watch live: right here on this page or [Google+](https://plus.google.com)

**Video streams on this page on Friday, February 10, 12 noon PT.**

Artificial Intelligence. It seems, now accomplishes nearly any task: it drives cars, helps us shop, decides which of our friends to display on Facebook. But Eugenia Kuyda is pushing AI into a still untouched space: talking to the lost.
Realdoll builds artificially intelligent sex robots with programmable personalities

Sex doll manufacturer Realdoll is dipping its toe (and we don’t want to know which other body parts) into the world of artificial intelligence and robotics with a forthcoming robot sex assistant that promises to form a “real bond” with its, er, users.

The new system is made up of several components, which will roll out over the course of this year and next. It will begin with the Harmony AI app, scheduled for release in the spring, which will allow users to control the robot’s movements and responses. A few months later, the robot itself, which will be around 3 feet tall, will be released. It will come with a library of scripts that can be activated by voice commands or through the Harmony app. The robot will be able to respond to a variety of scenarios, including role-playing, intimate encounters, and even virtual sex. The Harmony AI app will allow users to customize the robot’s routines and responses to their liking.

The company is confident that the new system will revolutionize the sex doll market, offering users a more personalized and interactive experience. The Harmony AI app will also allow for remote control, so users can have their virtual sex partner anywhere in the world. The company is currently working on a number of other features, including a virtual world where users can interact with their robot in a variety of settings.

Realdoll is also working on a range of other innovations, including a line of robotic pets and a virtual assistant that can help with household chores. The company is committed to developing new technologies that will make life easier and more enjoyable for its users.
Google uses AI to sharpen low-res images

'Blade Runner' style image enhancement is just a neural network away.
Today

• Uninformed Search (R&N Ch 3)

Tuesday, February 14

• Informed Search (R&N Ch 3)
Depth-First Search
Breadth-First Search
Depth-First Search

DFS(s,ops,queue) =
   If goal(s) then return(s);
   Else
      successors ← {}; queue’ ← queue;
      For each o ∈ ops that applies to s
         successors ← successors + apply(o,s);
      queue’ ← append(queue’,successors);
      If not(empty(queue’))
         s’ ← last(queue);
         queue’ ← remove(s’,queue’);
         DFS(s’,ops,queue’)
      Else return(FAIL)
   Initial call: DFS(initialstate,ops,{})
Depth-First Search

\[
\text{DFS}(s, \text{ops}, \text{open}) = \\
\begin{align*}
\text{If } \text{goal}(s) \text{ then return}(s); \\
\text{Else} \\
\text{successors} \leftarrow \{\}; \quad \text{open'} \leftarrow \text{open}; \\
\text{For each } o \in \text{ops} \text{ that applies to } s \\
\quad \text{successors} \leftarrow \text{successors} + \text{apply}(o, s); \\
\quad \text{open'} \leftarrow \text{append} (\text{open'}, \text{successors}); \\
\text{If not}(\text{empty}(\text{open'})) \\
\quad s' \leftarrow \text{last} (\text{open}); \\
\quad \text{queue'} \leftarrow \text{remove}(s', \text{open'}); \\
\quad \text{DFS}(s', \text{ops}, \text{open'}) \\
\text{Else return}(\text{FAIL})
\end{align*}
\]

Initial call: DFS(initialstate, ops, \{\})
Handling Repeated States

DFS(s,ops,open,closed) =
   If goal(s) then return(s);
   Else
      open’ ← open;
      If s ∉ closed then
         closed’ ← closed + s;
         successors ← {};
         For each o ∈ ops that applies to s
            successors ← successors + apply(o,s);
         open’ ← append(open’,successors);
         Else closed’ ← closed;
      Else closed’ ← closed;
      If not(empty(open’))
         s’ ← last(open’);
         open’ ← remove(s’,open’);
         DFS(s’,ops,open’,closed’);
      Else return(FAIL)
   Initial call: DFS(initialstate,ops,{},{})
Handling Repeated States

DFS(s,ops,open,closed) =
    If goal(s) then return(s);
    Else
        open’ ← open;
        If s ∉ closed then
            closed’ ← closed + s;
            successors ← {};
            For each o ∈ ops that applies to s
                successors ← successors + apply(o,s);
                open’ ← append(open’,successors);
            Else closed’ ← closed;
            If not(empty(open’))
                s’ ← last(open’);
                open’ ← remove(s’,open’);
                DFS(s’,ops,open’,closed’)
        Else return(FAIL)
    End If
End DFS

Initial call: DFS(initialstate,ops,{},{})
Iterative Deepening Search
Iterative Deepening Search

Depth-first search for 1 level
Iterative Deepening Search

Depth-first search for 2 levels
Iterative Deepening Search

Depth-first search for 3 levels
Iterative Deepening Search

Depth-first search for 3 levels

Etc.
Iterative Deepening Search

Need a depth-first search with a depth bound
Handling Repeated States

$$\text{DFS}(s, \text{ops}, \text{open}, \text{closed}) =$$

If goal(s) then return(s);
Else

open' $\leftarrow$ open;
If $s \notin \text{closed}$ then
closed' $\leftarrow$ closed + s;
successors $\leftarrow$ {};
For each $o \in \text{ops}$ that applies to $s$
  successors $\leftarrow$ successors + apply(o,s);
open' $\leftarrow$ append(open',successors);
Else closed' $\leftarrow$ closed;
If not(empty(open'))
  s' $\leftarrow$ last(open');
  open' $\leftarrow$ remove(s',open');
  DFS(s', \text{ops}, \text{open}', \text{closed}')
Else return(FAIL)

Initial call: $$\text{DFS}(\text{initialstate}, \text{ops}, \{\}, \{\})$$
Depth-Bounded Depth-First Search

\[
DBDFS(s,ops,open,maxdepth) = \\
\begin{array}{l}
\text{If goal}(s) \text{ then return}(s) ; \\
\text{Else} \\
\quad \text{open'} \leftarrow \text{open}; \\
\quad \text{if depth}(s) < \text{maxdepth} \\
\quad \{ \text{depth of each successor is set} \}
\end{array}
\]

\[
\begin{array}{l}
\quad \text{Do a test before successor generation} \\
\quad \{ \text{to 1 more than the depth of s} \}
\end{array}
\]

\[
\begin{array}{l}
\quad \text{successors} \leftarrow \{} ; \\
\quad \text{For each } o \in \text{ ops that applies to } s \\
\quad \quad \{ \text{depth of each successor is set} \}
\end{array}
\]

\[
\begin{array}{l}
\quad \text{successors} \leftarrow \text{successors + apply}(o,s) ; \\
\quad \text{open'} \leftarrow \text{append}(\text{open'},\text{successors}); \\
\quad \text{If not}(\text{empty}(\text{open'})) \\
\quad \quad \text{s'} \leftarrow \text{last}(\text{open}); \\
\quad \quad \text{queue'} \leftarrow \text{remove}(\text{s'},\text{open'}); \\
\quad \quad \text{DBDFS}(\text{s'},\text{ops,open',maxdepth}) \\
\quad \text{Else return}(\text{FAIL}) \\
\end{array}
\]

Initial call: \text{DBDFS}(\text{initial state,ops,{},maxdepth})
Iterative Deepening Search

IDS(s,ops) =
    i ← 1;
    repeat
        result ← DBDFS(s,ops,{},i);
        i ← i + 1;
    until result ≠ FAIL;
    return(result)
Iterative Deepening Search
Iterative Deepening Search
Iterative Deepening Search

1, 2
Iterative Deepening Search
Iterative Deepening Search
Iterative Deepening Search
Iterative Deepening Search

1, 2, 6

3

4

5
Iterative Deepening Search
Iterative Deepening Search
Iterative Deepening Search

Diagram:

- Root node with 1,2,6
  - Child node 3,7
    - Grandchild nodes 8, 9
  - Child node 4
  - Child node 5
Iterative Deepening Search
Iterative Deepening Search

1, 2, 6

3, 7

4, 11

5

8, 9, 10
Iterative Deepening Search
Iterative Deepening Search
Iterative Deepening Search

1,2,6

3,7

4,11

5

8 9 10 12 13 14
Iterative Deepening Search

```
1,2,6
/   /
3,7 4,11
/  /  /
8 9 10 12 13 14
/ / / / / /
5 15
```
Iterative Deepening Search

1, 2, 6

3, 7

4, 11

5, 15

8, 9, 10, 12, 13, 14, 16
Iterative Deepening Search
Iterative Deepening Search
## Analysis: Iterative Deepening

<table>
<thead>
<tr>
<th>Criterion</th>
<th>DFS</th>
<th>BFS</th>
<th>Iterative Deepening</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complete?</td>
<td>No</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Optimal?</td>
<td>No</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Time?</td>
<td>∞</td>
<td>O(b^n)</td>
<td></td>
</tr>
<tr>
<td>Space?</td>
<td>O(db)</td>
<td>O(b^d)</td>
<td></td>
</tr>
</tbody>
</table>
### Analysis: Iterative Deepening

<table>
<thead>
<tr>
<th>Criterion</th>
<th>DFS</th>
<th>BFS</th>
<th>Iterative Deepening</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complete?</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Optimal?</td>
<td>No</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Time?</td>
<td>$\infty$</td>
<td>$O(b^n)$</td>
<td></td>
</tr>
<tr>
<td>Space?</td>
<td>$O(db)$</td>
<td>$O(b^d)$</td>
<td></td>
</tr>
</tbody>
</table>
## Analysis: Iterative Deepening

<table>
<thead>
<tr>
<th>Criterion</th>
<th>DFS</th>
<th>BFS</th>
<th>Iterative Deepening</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complete?</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Optimal?</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Time?</td>
<td>$\infty$</td>
<td>$O(b^n)$</td>
<td></td>
</tr>
<tr>
<td>Space?</td>
<td>$O(db)$</td>
<td>$O(b^d)$</td>
<td></td>
</tr>
<tr>
<td>Level</td>
<td>Best-First</td>
<td>Iterative Deepening</td>
<td>Ratio</td>
</tr>
<tr>
<td>-------</td>
<td>------------</td>
<td>---------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1.000000</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>4</td>
<td>1.333333</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
<td>11</td>
<td>1.571429</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>26</td>
<td>1.733333</td>
</tr>
<tr>
<td>4</td>
<td>31</td>
<td>57</td>
<td>1.838710</td>
</tr>
<tr>
<td>5</td>
<td>63</td>
<td>120</td>
<td>1.904762</td>
</tr>
<tr>
<td>6</td>
<td>127</td>
<td>247</td>
<td>1.944882</td>
</tr>
<tr>
<td>7</td>
<td>255</td>
<td>502</td>
<td>1.968627</td>
</tr>
<tr>
<td>8</td>
<td>511</td>
<td>1013</td>
<td>1.982387</td>
</tr>
<tr>
<td>9</td>
<td>1023</td>
<td>2036</td>
<td>1.990225</td>
</tr>
<tr>
<td>10</td>
<td>2047</td>
<td>4083</td>
<td>1.994626</td>
</tr>
<tr>
<td>11</td>
<td>4095</td>
<td>8178</td>
<td>1.997070</td>
</tr>
<tr>
<td>12</td>
<td>8191</td>
<td>16369</td>
<td>1.998413</td>
</tr>
<tr>
<td>13</td>
<td>16383</td>
<td>32752</td>
<td>1.999145</td>
</tr>
<tr>
<td>14</td>
<td>32767</td>
<td>65519</td>
<td>1.999542</td>
</tr>
<tr>
<td>15</td>
<td>65535</td>
<td>131054</td>
<td>1.999756</td>
</tr>
<tr>
<td>16</td>
<td>131071</td>
<td>262125</td>
<td>1.999870</td>
</tr>
<tr>
<td>17</td>
<td>262143</td>
<td>524268</td>
<td>1.999931</td>
</tr>
<tr>
<td>18</td>
<td>524287</td>
<td>1048555</td>
<td>1.999964</td>
</tr>
<tr>
<td>19</td>
<td>1048575</td>
<td>2097130</td>
<td>1.999981</td>
</tr>
<tr>
<td>20</td>
<td>2097151</td>
<td>4194281</td>
<td>1.999990</td>
</tr>
</tbody>
</table>

\[
n = 2^{n+1} - 1 = \frac{2^{n+2} - n - 3}{2^{n+1} - 1} \rightarrow 2
\]
<table>
<thead>
<tr>
<th>Level</th>
<th>Best-First</th>
<th>Iterative Deepening</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1.000000</td>
</tr>
<tr>
<td>1</td>
<td>11</td>
<td>12</td>
<td>1.090909</td>
</tr>
<tr>
<td>2</td>
<td>111</td>
<td>123</td>
<td>1.108108</td>
</tr>
<tr>
<td>3</td>
<td>1111</td>
<td>1234</td>
<td>1.110711</td>
</tr>
<tr>
<td>4</td>
<td>11111</td>
<td>12345</td>
<td>1.110611</td>
</tr>
<tr>
<td>5</td>
<td>111111</td>
<td>123456</td>
<td>1.111050</td>
</tr>
<tr>
<td>6</td>
<td>1111111</td>
<td>1234567</td>
<td>1.111105</td>
</tr>
<tr>
<td>7</td>
<td>11111111</td>
<td>12345678</td>
<td>1.111111</td>
</tr>
<tr>
<td>8</td>
<td>1.11E+08</td>
<td>123456789</td>
<td>1.111111</td>
</tr>
<tr>
<td>9</td>
<td>1.11E+09</td>
<td>1.235E+09</td>
<td>1.111111</td>
</tr>
<tr>
<td>10</td>
<td>1.11E+10</td>
<td>1.235E+10</td>
<td>1.111111</td>
</tr>
<tr>
<td>11</td>
<td>1.11E+11</td>
<td>1.235E+11</td>
<td>1.111111</td>
</tr>
<tr>
<td>12</td>
<td>1.11E+12</td>
<td>1.235E+12</td>
<td>1.111111</td>
</tr>
<tr>
<td>13</td>
<td>1.11E+13</td>
<td>1.235E+13</td>
<td>1.111111</td>
</tr>
<tr>
<td>14</td>
<td>1.11E+14</td>
<td>1.235E+14</td>
<td>1.111111</td>
</tr>
<tr>
<td>15</td>
<td>1.11E+15</td>
<td>1.235E+15</td>
<td>1.111111</td>
</tr>
<tr>
<td>16</td>
<td>1.11E+16</td>
<td>1.235E+16</td>
<td>1.111111</td>
</tr>
<tr>
<td>17</td>
<td>1.11E+17</td>
<td>1.235E+17</td>
<td>1.111111</td>
</tr>
<tr>
<td>18</td>
<td>1.11E+18</td>
<td>1.235E+18</td>
<td>1.111111</td>
</tr>
<tr>
<td>19</td>
<td>1.11E+19</td>
<td>1.235E+19</td>
<td>1.111111</td>
</tr>
<tr>
<td>20</td>
<td>1.11E+20</td>
<td>1.235E+20</td>
<td>1.111111</td>
</tr>
</tbody>
</table>

$n = \frac{10^{n+1} - 1}{9}$

$\frac{10^{n+2} - (n + 2)}{10^{n+1} - 1} \rightarrow \frac{10}{9}$
Analysis: Iterative Deepening

\[ 1 + (1+b) + (1+b+b^2) + \cdots + (1+b+\cdots b^d) \]
Analysis: Iterative Deepening

\[ 1 + (1+b) + (1+b+b^2) + \cdots + (1+b+\cdots b^d) \]
\[ = \sum_{i=0}^{0} b^i + \sum_{i=0}^{1} b^i + \sum_{i=0}^{2} b^i + \cdots + \sum_{i=0}^{d} b^i \]
Analysis: Iterative Deepening

\[
\sum_{i=0}^{n} b^i = \frac{b^{n+1} - 1}{b - 1}
\]
Analysis: Iterative Deepening

\[
1 + (1+b) + (1+b+b^2) + \cdots + (1+b+\cdots+b^d)
= \sum_{i=0}^{0} b^i + \sum_{i=0}^{1} b^i + \sum_{i=0}^{2} b^i + \cdots + \sum_{i=0}^{d} b^i
= \frac{b^1 - 1}{b - 1} + \frac{b^2 - 1}{b - 1} + \cdots + \frac{b^{d+1} - 1}{b - 1}
\]
Analysis: Iterative Deepening

\[ 1 + (1+b) + (1+b+b^2) + \cdots + (1+b+\cdots b^d) \]
\[ = \sum_{i=0}^{0} b^i + \sum_{i=0}^{1} b^i + \sum_{i=0}^{2} b^i + \cdots + \sum_{i=0}^{d} b^i \]
\[ = \frac{b^1 - 1}{b - 1} + \frac{b^2 - 1}{b - 1} + \cdots + \frac{b^{d+1} - 1}{b - 1} \]
\[ = \frac{\sum_{i=1}^{d+1} b^i - 1}{b - 1} = \frac{\sum_{i=1}^{d+1} b^i - \sum_{i=1}^{d+1} 1}{b - 1} = \frac{b^{d+2} - 1}{b - 1} - 1 - (d+1) = O\left(\frac{b^{d+2}}{b^2}\right) = O(b^d) \]
### Analysis: Iterative Deepening

<table>
<thead>
<tr>
<th>Criterion</th>
<th>DFS</th>
<th>BFS</th>
<th>Iterative Deepening</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complete?</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Optimal?</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Time?</td>
<td>$\infty$</td>
<td>$O(b^d)$</td>
<td>$O(b^d)$</td>
</tr>
<tr>
<td>Space?</td>
<td>$O(bd)$</td>
<td>$O(b^d)$</td>
<td></td>
</tr>
</tbody>
</table>
## Analysis: Iterative Deepening

<table>
<thead>
<tr>
<th>Criterion</th>
<th>DFS</th>
<th>BFS</th>
<th>Iterative Deepening</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complete?</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Optimal?</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Time?</td>
<td>$\infty$</td>
<td>$O(b^d)$</td>
<td>$O(b^d)$</td>
</tr>
<tr>
<td>Space?</td>
<td>$O(bd)$</td>
<td>$O(b^d)$</td>
<td>$O(bd)$</td>
</tr>
</tbody>
</table>