

Artificial Intelligence

Simulating The Other

ARTIFICIAL INTELLIGENCE

- There's no real intelligence, just a bunch of techniques. Sorry.
- Even the most advanced techniques are just tools.
- More complexity means more stuff to debug.
- Most important thing:

fun > realism

ARTIFICIAL INTELLIGENCE

- What is a good AI?
 - Perfect? (racing, tic-tac-toe..)
 - Challenging? (chess)
 - Non-cheating? (civilization)

..or perhaps, you know, it should just be fun?

ARTIFICIAL INTELLIGENCE

- How would **you** play?
 - Or better yet, how would a good player play?
- Can you define it as an algorithm?
- Would this be a fun opponent?
 - What can you do to make it so?

AI TOOLS: RANDOM

- Random is your friend.
 - Replayability.
 - Unpredictability.
- Random is not your friend.
 - Humans can detect patterns, even random ones.
 - Random can be boring.
 - Non-determinism can make debugging harder.
 - Randomness may not turn out to be fun.

AI TOOLS: RANDOM

- "The generation of random numbers is too important to be left to chance"
-- Robert R. Coveyou
- Many pseudorandom number generators are of poor quality (esp. C std lib ones)
- Better ones: Mersenne Twister, WELL

http://en.wikipedia.org/wiki/Well_Equidistributed_Long-period_Linear

AI TOOLS: STATE MACHINES

- State machines are extremely usable.
- Example; "Thief" AI
 - Idle
 - "Heard something"
 - "I saw you"
 - Attack
 - "I guess it was nothing" / "Must have run away"

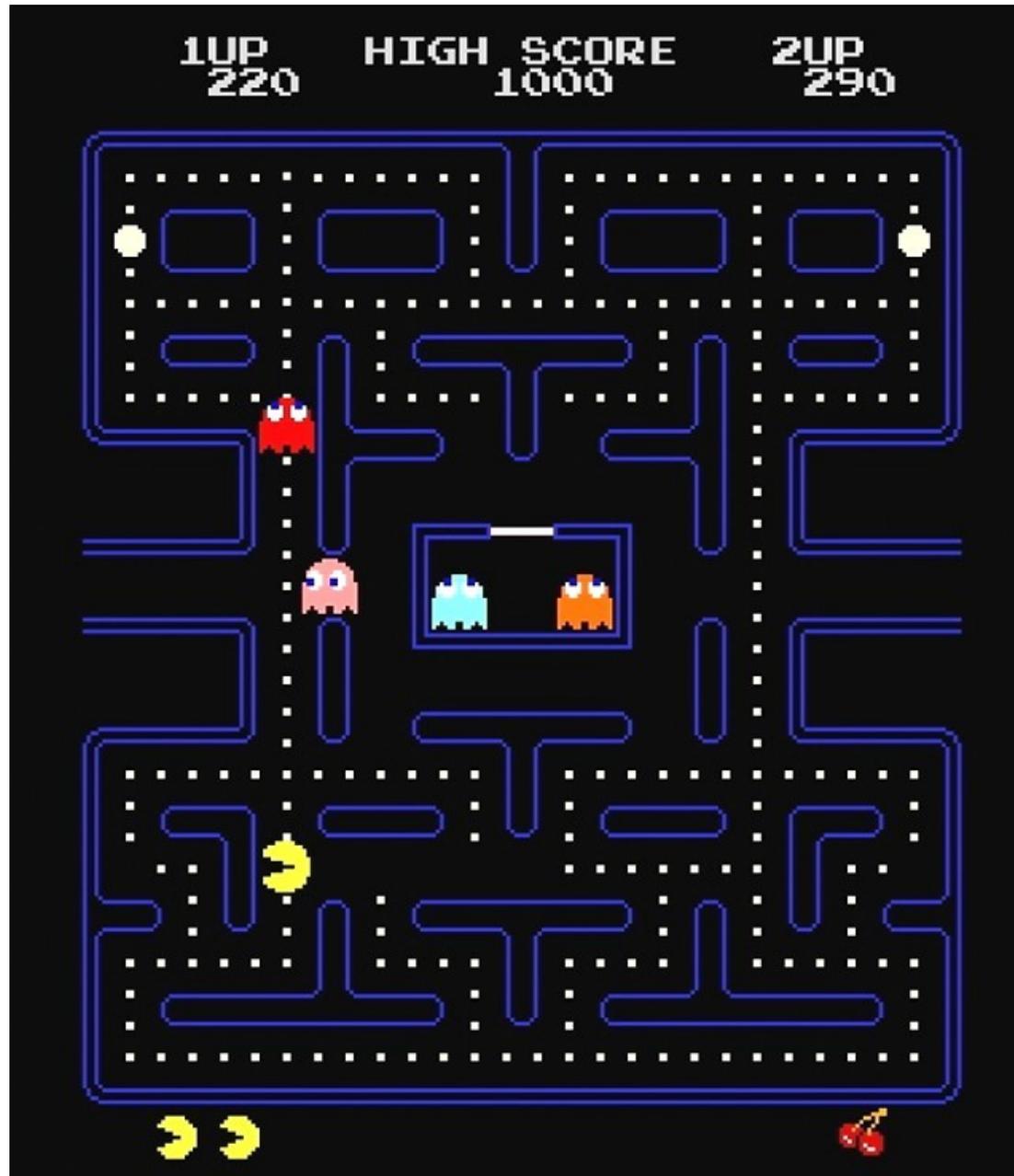
AI TOOLS: ADVANCED STUFF

- Fuzzy logic, genetic algorithms, neural nets.
 - Can simulate more complicated intelligence.
 - Learning.
 - Adapting.
 - Generally harder to tweak to be fun.
 - Usually can be avoided, achieving the same goals with more deterministic methods.

PATHFINDING - A*

- Fill graph (or grid) nodes.
 - Starting from point A.
 - Always filling nodes that are "closest" to target.
 - When B is reached, travel back using the cheapest route.
- Common, but not necessarily the best option.

EXAMPLES



EXAMPLES



EXAMPLES



EXAMPLES



EXAMPLES



EXAMPLES



EXAMPLES



HOMEWORK

- Create an AI algorithm that plays poker.
- Algorithm should consist of several pairs of:
 - WHEN (description of situation)
 - DO (what)
- http://en.wikipedia.org/wiki/List_of_poker_hands
- <http://en.wikipedia.org/wiki/Poker#Gameplay>
- Regular poker, not a variant, thanks.