kind461 Intro to Artificial Intelligence

TOPICS OVERVIEW – EXAM 1

Intro/Chapter 1

* Define artificial intelligence --- <https://www.youtube.com/watch?v=ad79nYk2keg>
* Strong vs Weak AI
* Agents <https://www.tutorialspoint.com/artificial_intelligence/artificial_intelligence_agents_and_environments.htm>
  + Precept
  + Percept Sequences
  + Agent Functions
* Turing Test --- <https://www.youtube.com/watch?v=3wLqsRLvV-c> , <https://www.youtube.com/watch?v=4VROUIAF2Do>
* Objections to Turing test
  + Searles Chinese Room --- <https://www.youtube.com/watch?v=htrsnpwzhmI>
  + Blocks
* Rationality -- <https://www.youtube.com/watch?v=MqXfz65vPDw>
  + Inference in Uncertainty
  + Rational Decisions
  + **Acting** Rationally
* Foundations of AI
* Heuristics --- <https://www.youtube.com/watch?v=ReFqFPJHLhA>
  + Compared to Algorithms
  + Definitions
  + Kinds of heuristics
* Algorithm vs Heuristic ---- <https://www.youtube.com/watch?v=jJ3CV3yhajM>
* Properties of Heuristics
  + What should a heuristic never do.
  + Monotonic heuristic
* More informed search algorithm
  + Searches what?
* Problems Suitable for AI
  + What they require
  + What are they?
* Properties of Environments -- <https://www.youtube.com/watch?v=yHXatUO6h28>
  + Fully Observable vs Partially Observable vs Unobservable
  + Single agent vs multi agent
    - Cooperative vs competitive
  + Deterministic vs Stochastic choices
    - Deterministic vs non-Deterministic
    - Uncertain
  + Episodic vs Sequential
    - What matters?
  + Static vs Dynamic Environments
    - Semi-Dynamic
  + Discrete vs Continuous
    - State and time
  + Known vs Unknown.
    - Programmed in or learnable.
* More foundations of AI – where do we draw from?

Chapter 2

* Searching --- <https://www.youtube.com/watch?v=gRK5BUw7TCk> ( Linear search vs Binary search)
  + What are the aims and goals of **searching**?
* Uninformed / Unsupervised Searches (Uninformed vs informed)
  + Brute Force
* Path length --- <https://www.youtube.com/watch?v=aU_qPL6VN6o>
* Lowest cost path ---- <https://www.youtube.com/watch?v=t1shZ8_s6jc>
  + Shortest?
* State Space Graph ---- <https://www.youtube.com/watch?v=ngkXAcjeNWE>
* Branching Factor ----- <https://www.youtube.com/watch?v=BUnoSIp0HbA&t=16s> (focus on **properties**)
* False Coin Problem ---- <https://www.youtube.com/watch?v=tE2dZLDJSjA> (another version of the solutions by parts coin problem)
* Greedy Algorithms ---- <https://www.youtube.com/watch?v=HzeK7g8cD0Y> [this one is ok but includes unnecessary information – rely on your notes for topics]
  + What’s the goal.
  + Where does it look?
  + What causes failure?
* Depth First Search DFS ------ <https://www.youtube.com/watch?v=h1RYvCfuoN4> (numerical solution on a whiteboard – good if you need to see the numbers to get a better picture)
  + How does it work?
  + Strengths / Weaknesses compared to others.
  + Storage Needs
  + Optimal?
* Breadth First Search BFS ----- <https://www.youtube.com/watch?v=1wu2sojwsyQ> , <https://www.youtube.com/watch?v=n3fPL9q_Nyc>
  + How does it work?
  + Strengths / Weaknesses compared to others.
  + Storage Needs
  + Optimal?
* Iterative Deepening IDDFS --- <https://www.youtube.com/watch?v=Y85ECk_H3h4>
  + How does it work?
  + Strengths / Weaknesses compared to others.
  + Storage Needs
  + Optimal
* Bi-Directional Search ---- <https://www.youtube.com/watch?v=RS9r7FzYKDk> (up to 3:33 – past that is **specific** code level implementation – good if you want to ground it by method)
  + How does it work?
  + Strengths / Weaknesses compared to others
  + Storage Needs
  + Complete?
  + Optimal?
* Comparison of Path finding Algorithms -- <https://www.youtube.com/watch?v=GC-nBgi9r0U>
* Admissible Heuristics
* Monotonic / Consistent Heuristics ---- <https://www.youtube.com/watch?v=kAsVvS02T_U>
  + How does it relate to admissible?
* More informed heuristic
  + What makes a heuristic more Informed?
  + What makes a heuristic dominate?
* Relaxed Heuristics
* Pattern Database Heuristics ---- <https://www.youtube.com/watch?v=HZWV4uOJWk8> (deeper than we went in our discussion, but good info)
  + Disjoint Pattern Database
* Actual + Heuristic Values
  + Monotonic heuristics
* Branch and Bound / Uniform Cost Search --- <https://www.youtube.com/watch?v=dRMvK76xQJI>
  + How does it order nodes?
  + Is it optimal? Why?
* Branch and Bound with Underestimates -----<https://www.youtube.com/watch?v=qqPcWZ0vV7g>
* -- LONG [MiT OCW] <https://www.youtube.com/watch?v=gGQ-vAmdAOI> ( starts at ~15 min + other items)
  + What do we use heuristics for?
  + How does this differ from above?
  + Is this informed?
  + What if we find a lower cost path?
* A\* Search ------ <https://www.youtube.com/watch?v=ySN5Wnu88nE> (very good for conceptual understanding – visually interesting/numerical breakdown)
  + Why does it combine actual with heuristic costs
  + What is the advantage of A\* over other informed searches
  + Is it complete / Optimal?
  + Downsides?
* Beam Search --- <https://www.youtube.com/watch?v=KVR8J3iPszw> (solution starts @ 2:12)
  + What method does it use to start?
  + How does it continue?
  + Does a wider beam produce better solutions?
* Best First Search---- <https://www.youtube.com/watch?v=dv1m3L6QXWs>
  + What lists are maintained, what goes in them?
  + How do we order these lists?
  + Can it recover from dead ends?
* Constraint Satisfaction ---- <https://www.youtube.com/watch?v=_e64FiDWvqs>   
  (with backtrack solution)
* And-Or Trees [methods/terms only]
  + What do these typically solve?
* Goals of Heuristics
  + Do heuristics learn by guiding or experience?
  + Definition of Heuristic
  + Do they speed up path solutions?
* What do heuristics decide for us?

Chapter 3 --- <https://ktiml.mff.cuni.cz/~bartak/constraints/consistent.html> [overview]  
(Guide to constraint *programming if you like things in a more* ***code driven*** *approach this should help*)

* Constraint Satisfaction
  + How is it different than normal searches?
  + What is an advantage?
  + Can these be general?
* Solution
  + Definition
  + What makes a solution complete
  + What makes a solution consistent
  + What makes a solution correct
* Constraint Graph
  + How do nodes connect?
* Types of constraints
  + Urnary
  + Binary
  + N-Ary
  + Global
    - Preference
      * How/when do we apply preference constraints
      * Is this optimal?
* Constraint Propagation ----- <https://www.youtube.com/watch?v=vQ7ucWQOQ4s>
  + What is it? What does it do?
* (Covered CSP Variations, Constraint Propagation, Local Consistencies, Node, Arc, Path)
* Arc Consistency ----- <https://www.youtube.com/watch?v=5rlIYGJdPy4&t=128s>
* Network Arc Consistency
* AC-3
  + Understand the algorithm -> [Video](https://www.youtube.com/watch?v=4cCS8rrYT14&t=250s&ab_channel=JohnLevine)
  + What do we end up with after AC-3
  + Time Complexity
* Local Consistency vs Node-Consistency --- <https://www.youtube.com/watch?v=ehXK8EYcYM0>K
* Consistency
  + Node / Path / Arc Consistency
* Global Constraints
  + Effects all variables?
* Domain
  + Possible assignments
  + What does this mean we can do with the domain?
* Resource Constraints (**at most** constraints)
* Bounds Consistency
* Back Tracking
  + What kind of search is this?
  + Describe the process
  + How do we speed uniformed DFS without problem specific heuristics?
* Variable Ordering
  + Minimum Remain Values Heuristic
  + Degree Heuristic
* Value Ordering
  + Least Constraining Value Heuristic
  + Is choosing values: fail first or fail last
* Local Search for Constraint Problems
  + What heuristic should be used?
* And Or Trees
* Heuristic Bidirectional Search --- <https://www.youtube.com/watch?v=iW-OjmxSNeo>

Chapter 4

* States not Paths
* Focus on local search.
* State Space Landscapes
* Local Hill Climbing Search ---- <https://www.youtube.com/watch?v=VoUotaCmDk4> , <https://www.youtube.com/watch?v=rA3a8QDtYLs>
  + Greedy?
  + Issues?
  + Solutions?
* Local Beam Search --- <https://www.youtube.com/watch?v=JaG9dGhcP3w>
  + How many do we choose down each path to follow?
* Heuristic Beam Search
* Genetic Algorithms -- <https://www.youtube.com/watch?v=uQj5UNhCPuo>
  + Initialization
  + Selection
    - Fitness
  + Reproduction
    - Crossover
    - Mutation