Artificial Intelligence Past Paper

- Adapted from the Lecturer's materials
- AI-I, AI-II

Part 1: Introduction and Search

- y2007p3q8
  - Search, Tree (priority queue / open list) vs Graph (+closed list, s.t. dedup)
  - Recursive Best First Search vs $A^\star$
- y2008p3q7
  - Recursive Best First Search vs $A^\star$
- y2009p4q3
  - $A^\star$, IDA
- y2016p4q2
  - IDA
- y2019p6q2 (d)
  - design admissible heuristic ($\leq$)
- y2012p4q1
  - heuristics
- y2013p4q1
  - local search

Part 2: Games and Constraint Satisfaction Problems

Games

- y2001p9q8
  - alpha-beta
- y2006p3q3
  - minimax algorithm

CSPs

$Vi \in D_i = \{l_0, l_1, l_2, l_3\}$, $\{constraints\}$, complete / consistent assignment

constraint propagation, back-jumping

- y2005p3q3
  - CSP vs $A^\star$
  - backtracking
  - minimum remaining values, the degree, the least constraining value heuristic
- y2010p4q1
  - forward checking

    - arc consistency
- y2012p4q2
    - Gaschnig's algorithm, graph-based backjumping vs forward checking
- y2013p4q1 (c)
    - local search extension
- y2014p4q2
    - arc consistency
    - AC-3 algorithm, 3-consistency
- y2015p4q2
    - backjumping (Gaschnig's algorithm vs graph-based backjumping)
- y2017p4q1 (a,b)
    - add row number and transfer to binary constraints
    - later used in Planning
- y2020p6q2
    - forward checking, AC-3, Constraint propagation
- y2021p6q2
    - forward checking

Part 3: Knowledge Representation and Reasoning

Situation Calculus

- y2003p9q8
    - ontological vs epistemological commitment
    - representational vs inferential frame problem
    - the qualification (pre), the ramification problem (effect)
- y2010p4q2
- y2014p4q1
    - unique names axiom, unique actions axiom
- y2006p4q4

Part 4: Planning

	situation space	**plan space**
Plan Representation	Sequence of actions / vars	Partial plans with flexible seqs / vars
Variable Commitment	Fixed before search	Least-commitment (delayed)
Search Space	Finite (states)	Infinite (plans)
Efficiency	Potentially faster when it works	Adaptive, potentially efficient

	situation space	plan space
Application	Smaller, well-defined problems	Complex, dynamic problems

For state-variable, given ground instances X, Domain, $\mathcal{D}_i^a$

	situation / plan	{(state-variable=c),$v \in X$ } / CSP
States	$s0 \rightarrow [s1 = result(grab, s)]...$ (ad hoc from 1 and onwards)	all states $RR : \mathcal{D}_i^a \times \mathcal{D}^a$ functions $f : \mathcal{D}_i^a \times S \rightarrow \mathcal{D}^a$
Logic	propositional	first-order logic
Action axioms	Probability (precondition) --- successor state Effect (for monetary action) Frame (for objects, persistent action☐)	$\{at(x, s) = c, v_2 = c'\}$ in s $\gamma(s, a) = \{(v = c) \vert v \in X\}$ c_{effect} c_{frame}
Goal	conjunction literals at timestamp T $at_T(v, row_{id}, col_{id})$	a set of state variable assignments $g \in \gamma(s_n, a_n)$
Sols	same	a sequence of actions from start state $(a_0, a_1, ...a_n)$

- y2003p8q8
  - STRIPS, States, Goals, Operators(Action, Pre, Effect)
  - plan, consistent, complete [all causal links / protection intervals $S' \rightarrow^c S$, precondition $c \in Effects(S')$]
  - initial plan with ordering constraint $Start < Finish$, no $v = x$
  - threat, $S' < S'' < S$ and $c \notin Effect(S'')$
    - promotion (after the threatened connection)
    - demotion (before the threatened connection)
- y2008p4q6
- y2009p4q4
  - situation space vs plan space
- y2011p4q2
- y2016p4q1

Propositional Logic

- y2018p6q1
  - sliding blocks puzzle, SAT
  - start state, goal state, axioms
  - action-exclusion (usually totally-order) vs state-constraint axioms (mutex link)

Planning Graph

- y2019p6q1
  - inconsistent effects, interfering actions, competing for preconditions mutex
  - the initial state level S_0, the first action level A_1, and the state level S_1 resulting from A_1
  - partial order planner, multiple actions can occur *simultaneously*
- y2022p6q1
  - GraphPlan
  - plan extraction as heuristic search

State-variable

- y2017p4q1 (c-e)
- y2023p7q1
  - translation to CSP (for each timestamp t)
  - CSP variables
    - $action^t, D^{action^t} = \{a\}$, where a is the ground instance of an action.
    - $sv_i^t(v_1, ...v_n), D^{sv_i^t} = range(sv_i^t) = \mathcal{D}^{sv_i^t}$, where a is the ground instance of an action.
  - CSP constraints
    - precondition, using binary constraint $(a^t = a, s_i^t = v)$
    - effect using binary constraint $(a^t = a, s_i^{t+1} = v)$
    - frame axioms using ternary constraint $(a^t = a, s_i^t = v, s_i^{t+1} = v)$
- y2019p6q2
  - State-variable (rigid relation, action, state, goals/solutions)
  - Heuristic search vs CSP

Part 5: Learning

Find a weight vector minimizing $E(w)$

Reuse δ_j from output layers, $y = \sigma(a_j), \delta_j = \frac{\partial E(\vec{w})}{\partial a_j} = \frac{\partial E(\vec{w})}{\partial y}\sigma'(a)$

- y2007p4q7
- y2011p4q1
  - gradient descent
- y2013p4q2
- y2015p4q1
  - application on $A^\star$
- y2017p4q2
  - MSE, cross-entropy loss
- y2018p6q2
  - $E(w)$ in 2D, convolution
- y2020p6q1
- y2021p6q1
  - heuristic search

- y2022p6q2
- y2023p7q2
    - n-class cross-entropy