CISC 499 Project Descriptions

Winter 2020

Generally, I am amenable to supervise any project in the area of Automated Planning. Below is a short collection of ideas I am currently seeking students for, and will be given priority over self-proposed ideas. Reach out if you would like to work on something not described below.

(1) Action Reachability via Deadend Detection

Objective
To develop a suite of debugging tools for analyzing automated planning models. Emphasis will be placed on detecting useless actions via problem reformulation and unsolvability detection.

Learning Outcomes
At the end of the semester, the student should have a firm grasp on the space of deadend detection techniques that exist in the automated planning literature, and additionally have a strong foundation in the AI field of Automated Planning.

Required Expertise
Basic understanding of classical planning, as provided by CISC 352.

Description
During the process of modeling a planning problem, the vast majority of models are incorrect, incomplete, or simply inconsistent. It is only the final model that is actually correct and concise. This project focuses on the crucial and understudied step of improving the modeling process directly through a complex analysis of reachability.

Deadend detection (aka unsolvability) in the field of Automated Planning (AP) is the task of determining if the goal can be reached from a given state. During the modeling process, the preconditions of each action currently being authored can be used as a temporary goal, and the initial state used to detect if the problem is now unsolvable. If so, the action in question can *never* be executed, and is thus useless (likely indicating a bug in the domain).

The project will involve exploring the various techniques for unsolvability, and applying them in novel ways for on-the-fly diagnostics of planning models.

Deliverables
Survey of existing unsolvability techniques; an online service for performing action reachability; integration of the service with pre-existing modeling software.

(2) Label-free Action Inference

Objective
To develop both the theoretical underpinning and practical implementation of inferring potential action schemas from discrete time series data.

Learning Outcomes
By the end of the semester, the student will have acquired a strong foundation in dynamical systems, and the area of action model induction. Further expertise in data-driven techniques for analyzing discrete time series data will be a natural outcome of this project.

Required Expertise
Understanding of classical planning, as provided by CISC 352. Additional expertise in data analytics, such as that provided by CISC 351/372, will be considered a strong asset.

Description
Understanding *how* an environment works by observation alone is a long-standing grand challenge for Artificial Intelligence. Action model induction refers to the task of automatically or semi-automatically synthesizing action descriptions from observation data only, thus providing a description of how an environment works. Typically, partial action schemas or action labels are provided as guidance, but this project aims to remove that assumption.

From discrete sequences of actions, the goal is to infer (1) which actions exist and are being executed; and (2) what are the potential preconditions and effects of those actions. Both perfect information and noisy variants of this problem will be considered, and the student is free to explore different approaches to solving the problem (an initial short-list of ideas will be provided).

Deliverables
A brief but complete survey of existing model induction methods; a stand-alone implementation for action model induction from discrete time series data; an analysis of the approach(es) developed on the existing benchmarks in the field.

(3) Interactive Forward Search Planner

Objective
To create a framework for the interactive exploration of heuristic forward search planning technology.

Learning Outcomes
The student will acquire an in-depth understanding of automated planning technology, and the methods used to create state-of-the-art planners.

Required Expertise
Strong web development skills are essential (e.g., JavaScript frameworks such as d3 or similar). A background in automated planning (such as that provided by CISC 352) would be an asset, but not essential.

Description
Modern AI planners are large and complex systems with decades of advanced engineering practices built in. This project aims to highlight the major components of modern planning systems in a visual and interactive way; integrated into the leading editor for planning problems at Planning.Domains.

This project aims to accomplish two key things:

1. Fill the gap in the field of automated planning for educational experiences that explain the core concepts of modern planners.

2. Construct a flexible framework for researchers in the field to visualize and test their ideas for new heuristics, search procedures, etc.

Tackling the task of visually explaining difficult concepts in Artificial Intelligence is at the heart of this project. The final product will have a life far beyond the duration of the course, and provide a valuable resource to an entire research field.

Deliverables
The following key deliverables are expected as an outcome of this project:

- A flexible framework integrated into the online Planning.Domains editor allowing for:
    - search visualization & analysis,
    - heuristic computation, and
    - custom data-structure exploration.

- Example implementations of 1-2 common search methods and 1-2 common heuristics

(4) RL Agent & Mathematical Model of the Tak Board Game

Objective
Tak is a board game simple to learn & play, but difficult to master. The objective of this project is to explore a variety of techniques for modeling and solving the game of Tak, and ultimately produce an agent capable of human-level play.

Learning Outcomes
As part of this project, the student will gain a thorough knowledge of modern (Deep) Reinforcement Learning and Automated Planning solving / modeling techniques.

Required Expertise
A firm knowledge of AI and RL techniques are essential, as provided by CISC 352 and CISC 474. Experience with deep learning libraries (such as PyTorch or TensorFlow) will be an asset.

Description
Tak is a strategic board game for 2 players that involves placing and moving stacks of tokens around a grid of cells. It is inspired by a game described in the fictional series known as **The King Killer's Chronicles**, and was co-designed by the book's author. The game is designed to be simple to play, but rich in the space of game-play and strategy.

The added element of (physical) depth in the game gives it a unique flavour compared to other traditional board games (such as Go, Checkers, and Chess). Exploring a variety of mathematical models for the representation of the game state will play a central role for this project. The languages considered will include both custom representations that modern reinforcement frameworks are capable of consuming (i.e., convertible to tensor-based representations), as well as planning models to capture the game mechanics.

The techniques explored for solving the game is intentionally left open to allow for custom exploration of ideas as the project progresses. That said, as a basis for comparison the student will be expected to implement and test common RL approaches such as DQN and those ideas found in the work of AlphaZero.

Deliverables
At the conclusion of this project, the student should have produced (1) a model of the Tak game in both a planning-like language as well as custom tensor-based representation; (2) a variety of (D)RL implementations to solve the game; and (3) a detailed comparison of the effectiveness of each of the approaches.