

Intelligent Planning for Autonomous Underwater Vehicles

Zeyn Saigol

January 31, 2007

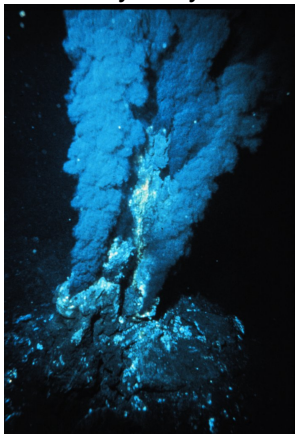
Supervisors: Richard Dearden
Jeremy Wyatt

- ▶ Introduction
- ▶ Oceanography and Black Smokers
- ▶ Autonomous Underwater Vehicles
- ▶ Classical planning systems
- ▶ Problem specification
- ▶ Markov Decision Processes

- ▶ Plan: sequence of actions to achieve a desired goal, given a starting state
- ▶ Action: method from getting from state A to state B
- ▶ Classical AI planning makes several assumptions:
 - ▶ Instantaneous actions
 - ▶ Deterministic universe
 - ▶ Discrete state
 - ▶ State is fully observable
 - ▶ Unlimited resources
- ▶ Recently, much interest in relaxing these \Rightarrow planning under uncertainty
- ▶ Application area: Autonomous Underwater Vehicles (AUVs) used in scientific research

Oceanography and Black Smokers

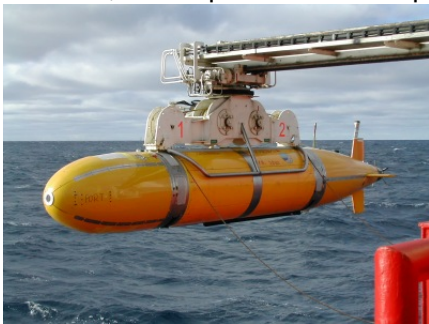
- ▶ Plate tectonics - 1960s
- ▶ Mechanism for most interesting ocean behaviour
- ▶ Discovery of hydrothermal vents (Black Smokers) - 1977



- ▶ Formed at spreading centres such as the Mid-Atlantic Ridge

Autonomous Underwater Vehicles

- ▶ Unmanned, untethered submersibles
- ▶ Autosub, developed in Southampton

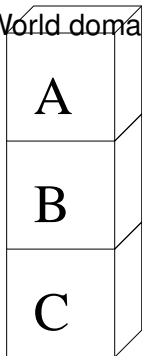


- ▶ Cheaper than manned vehicles
- ▶ Can get to places tethered vehicles can't
- ▶ No need for human supervision - extended missions
- ▶ Current control systems are simple if-then-else rules

Classical planning systems

- ▶ Familiar STRIPS planner with Blocks World domain
- ▶ Action: Pickup(x)

Preconditions	Clear(x) OnTable(x) HandEmpty
Add List	Holds(x)
Delete List	Clear(x) OnTable(x) HandEmpty

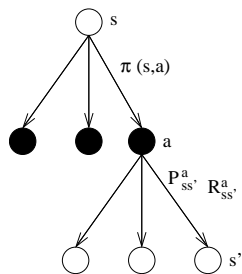


- ▶ Later systems more sophisticated, but same restrictions

- ▶ Results of actions are stochastic
- ▶ Actions can have a finite duration
- ▶ State is not completely or reliably observable
- ▶ State is made of continuous (rather than discrete) variables
- ▶ There are limited resources for executing the plan

Markov Decision Processes

- ▶ Dealing with uncertainty - use MDPs
- ▶ Mathematical framework for solving reinforcement learning and planning problems
- ▶ Models uncertainty in the outcome of actions



- ▶ $\pi(s, a)$ - policy of agent: probability of choosing action a from state s
- ▶ $P_{ss'}^a$ - probability of going from state s to state s' when taking action a
- ▶ $R_{ss'}^a$ - reward given to agent when transitions from state s to state s' via action a

Solving MDPs

- ▶ Usually solved by Dynamic Programming (or related method) - give a "score" $V(s)$ to each state s
- ▶ Iteratively apply the Bellman equation

$$V(s) = \sum_a \pi(s, a) \sum_{s'} P_{ss'}^a [R_{ss'}^a + \gamma V(s')]$$

- ▶ Needs iteration over every state \Rightarrow discrete state spaces only
- ▶ Various options for continuous state spaces
 - ▶ Discretise the state space
 - ▶ Use the structure of transition and reward functions to create a piecewise linear value function (Feng, Dearden, Meuleau, Washington)
 - ▶ Use structured functions to formulate the problem as an Approximate Linear Programming task (Hauskrecht and Kveton)

- ▶ AUVs - very useful in Oceanography
- ▶ Need more intelligence to maximise usefulness
- ▶ Good fit to new planning technologies
- ▶ My aim - to extend these new technologies to cover partially-observable uncertain state-spaces, and apply them in the AUV domain