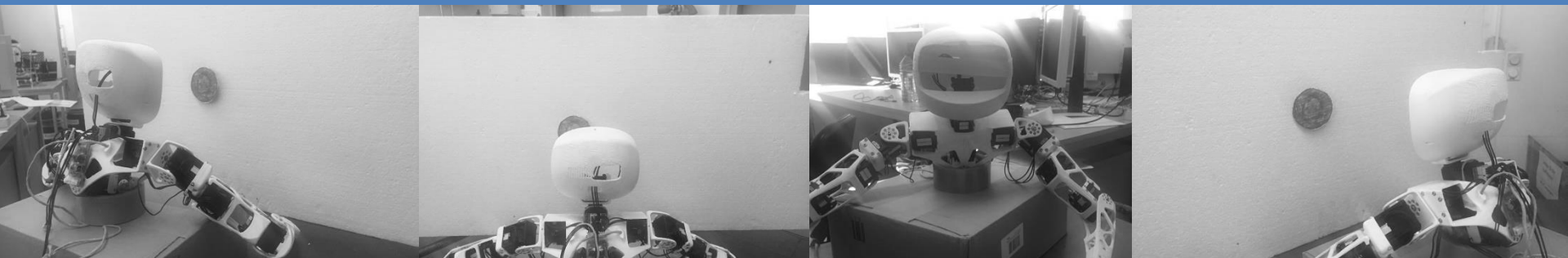
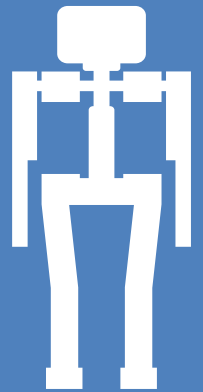


POPPY THE WATCHDOG



Applied Reinforcement Learning 2016
Erik Wannerberg, Zhiwei Han, Ben Pfirrmann





Structured Approach

Major Results

Major Learnings and Video Demo

Structured Approach and Task Modularization

Structured Approach

Structure supports Implementation

Framework

Relevant Classes

- Actor
- StateObserver
- Reward
- LearningAlgorithm
- GridStateActionSpace

Mathematical Domain

- Simply influencing and reading state variables according to what is mathematically intuitive for the scenario

VREP Domain

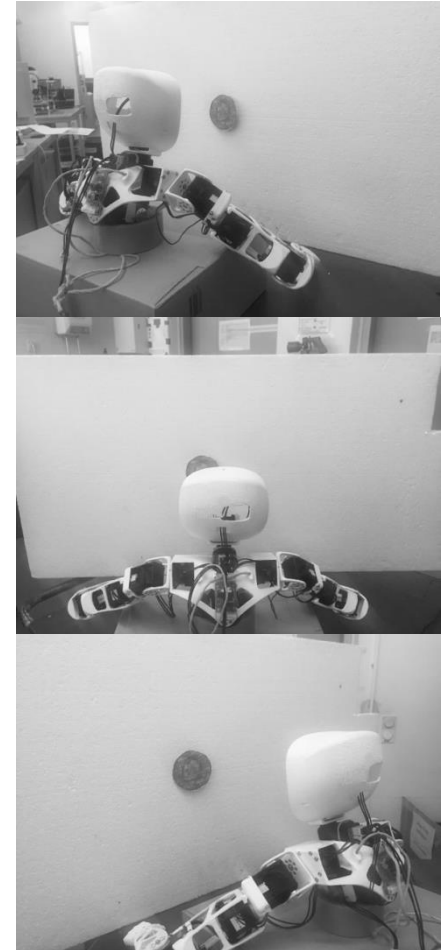
- Simulated camera-capture
- Controlling VREP-Poppy commands

Real Poppy Domain

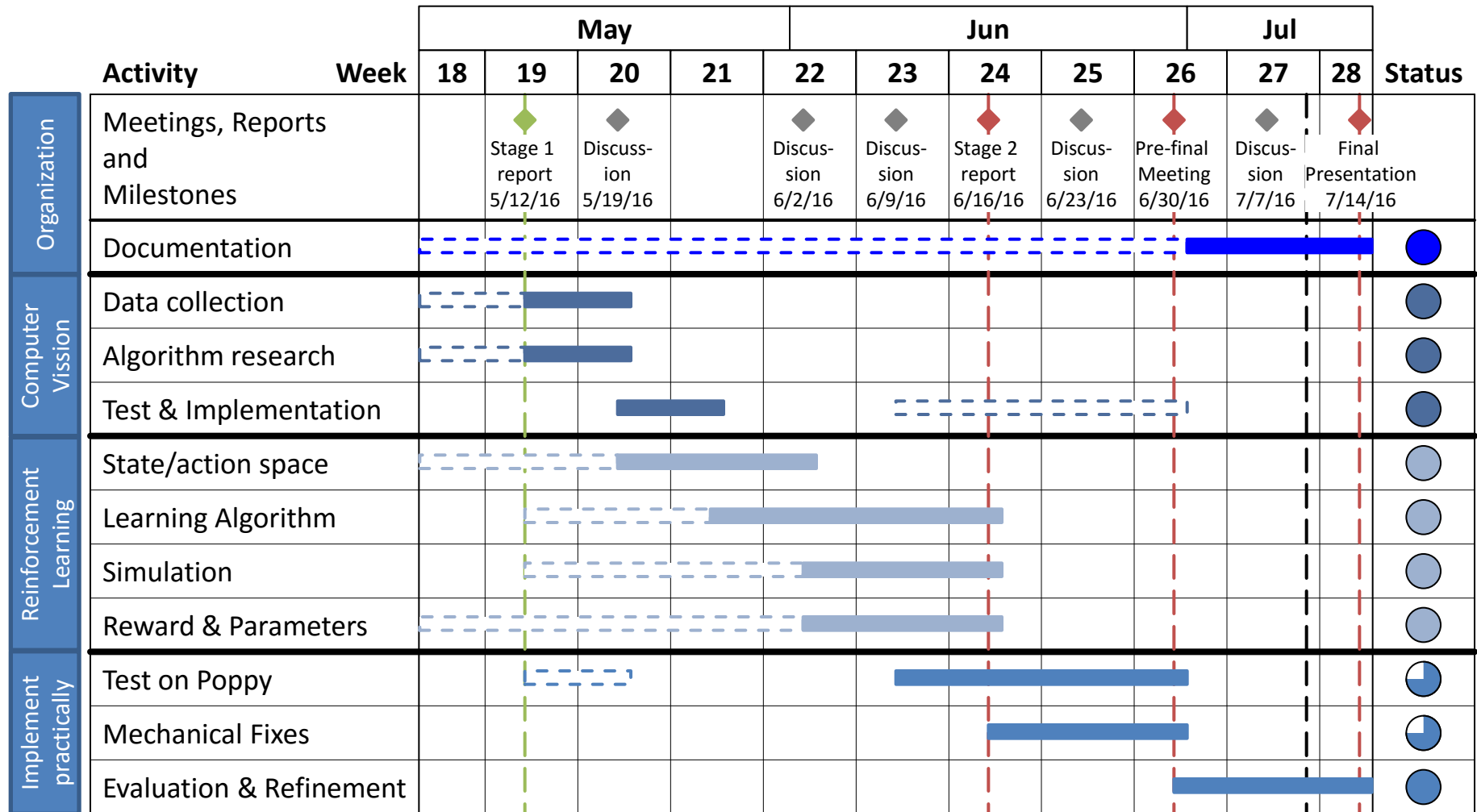
- Observing states based on camera images
- Actor controls real poppy motors
- Performance Evaluation

GitHub
(Version Control)

Task distribution made simple



Project plan - Visualized using a Gantt-Chart



7/10/16 ▲



Structured Approach

Major Results

Major Learnings and Video Demo

We implemented three strategies

Algorithm version	Mathematical	VREP	Real Poppy*
TD-Policy Iteration	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> *pre-final Version
SARSA(λ)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
SARSA(0)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>



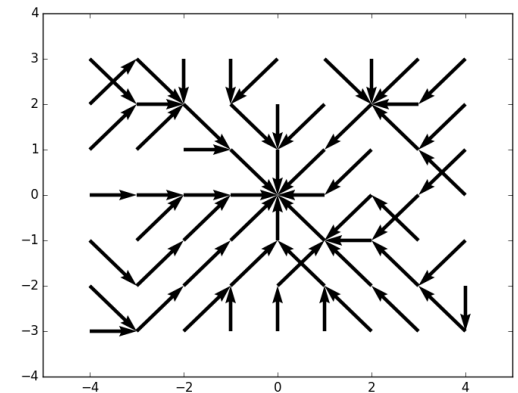
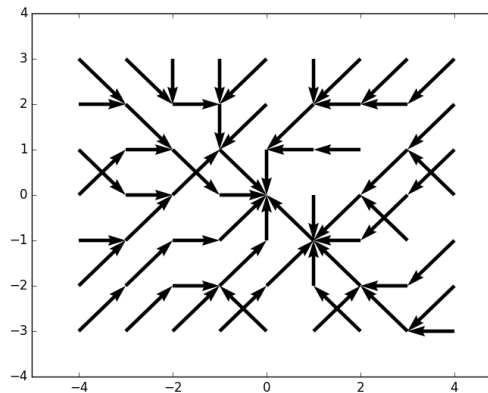
As the Camera broke we were unable to test and capture the behaviour after the pre-final version

Using different approaches, we also found different plotting techniques helpful

Figure 1: Value function of the TD Policy
Iteration - positionMatrix = (9, 7),
num_episodes = 50, epsilon = 0.1, gamma
= 0.7, learning_rate = 0.1



Figure 2: left: Policy for SARSA(λ), right: Policy for SARSA(0) - nsion = (9, 7), epsilonGreedy = 0.6,
learningRate = 0.1, gamma = 0.7, numEpisodes = 4500, lambdaDiscount = 0.5, iterNumLimit = 500





Structured Approach

Major Results

Major Learnings and Video Demo

General Project Management Learnings

- A software project requires a structured approach
 - GitHub was very helpful (if correctly used)
 - The Code-Framework was helpful
- People have different communication and work habits
 - A communication plan could be helpful
 - Writing down tasks is helpful
- A short time plan in the end may require over-time

Implementation Learnings

- Hardware may break and hence capturing pre-final results was very helpful
- Value function based approaches are not necessarily worse than Q-function based approaches
- A good and simple state-space model is very helpful to control the problem (we never needed to incorporate extensive Parameter tuning)



Great experience, that provided valuable experience with several relevant engineering and Reinforcement Learning Problems

