CS394R
Reinforcement Learning:
Theory and Practice

Peter Stone

Department of Computer Science
The University of Texas at Austin
Good Morning Colleagues

• Are there any questions?
Logistics

- Please do the class midterm Survey
Logistics

- Please do the class midterm Survey
- Schedule for rest of the semester
Options

- Extension of RL to temporal abstraction
Options

- Extension of RL to temporal abstraction
- State abstraction vs. temporal abstraction...
Options

- Extension of RL to temporal abstraction
- State abstraction vs. temporal abstraction...
  - ... Week 0 task!
Options

- Extension of RL to temporal abstraction
- State abstraction vs. temporal abstraction...
  - ... Week 0 task!
- They don’t address what temporal abstraction to use — they just show how it can fit into the RL formalism
Options

- Extension of RL to temporal abstraction

- State abstraction vs. temporal abstraction...
  - ... Week 0 task!

- They don’t address what temporal abstraction to use — they just show how it can fit into the RL formalism
  - Why couldn’t it before?
Options

- Extension of RL to temporal abstraction

- State abstraction vs. temporal abstraction...
  - ... Week 0 task!

- They don’t address **what** temporal abstraction to use — they just show how it can fit into the RL formalism
  - Why couldn’t it before?

- Markov vs. Semi-markov:
  - states, actions
  - mapping from (s, a) to expected discounted reward
  - well-defined distribution of next state, transit time
Discussion Points

• What happens when initial value functions are optimistic? (slides)
Discussion Points

• What happens when initial value functions are optimistic? (slides)

• Option discovery
  – bottleneck states
  – novelty
  – changed useful state abstractions (slides)
Discussion Points

• What happens when initial value functions are optimistic? (slides)

• Option discovery
  – bottleneck states
  – novelty
  – changed useful state abstractions (slides)
MAXQ

- Defines how to learn given a task hierarchically
MAXQ

- Defines how to learn given a task hierarchically
- Does not address how to construct the hierarchy
MAXQ

- Defines how to learn given a task hierarchically
- Does not address how to construct the hierarchy
- Strives for recursive optimality
MAXQ

- Defines how to learn given a task hierarchically
- Does not address how to construct the hierarchy
- Strives for **recursive optimality**— local optimality given subtask policies
MAXQ

- Defines how to learn given a task hierarchically
- Does not address how to construct the hierarchy
- Strives for **recursive optimality**— local optimality given subtask policies
  - Weaker or stronger than hierarchical optimality?
MAXQ

• Defines how to learn given a task hierarchically

• Does not address how to construct the hierarchy

• Strives for recursive optimality—local optimality given subtask policies
  – Weaker or stronger than hierarchical optimality?

• Enables reuse of subtasks
MAXQ

- Defines how to learn given a task hierarchically
- Does not address how to construct the hierarchy
- Strives for **recursive optimality**—local optimality given subtask policies
  - Weaker or stronger than hierarchical optimality?
- Enables reuse of subtasks
- Enables useful state abstraction (how?)
Some details

- A means both primitive actions and subtasks (options)
Some details

- \( a \) means both primitive actions and subtasks (options)

- Context-dependent vs. context-independent
Some details

- $a$ means both primitive actions and subtasks (options)

- Context-dependent vs. context-independent

- Higher-level subtasks are essentially policies over options
  - But subtasks are learned too
  - And the values propagate correctly
Some details

- $a$ means both primitive actions and subtasks (options)

- Context-dependent vs. context-independent

- Higher-level subtasks are essentially policies over options
  - But subtasks are learned too
  - And the values propagate correctly

- What does $C^\pi_i (s, a)$ mean?
Some details

- $a$ means both primitive actions and subtasks (options)
- Context-dependent vs. context-independent
- Higher-level subtasks are essentially policies over options
  - But subtasks are learned too
  - And the values propagate correctly
- What does $C_i^\pi(s, a)$ mean? (Nick slides)
Some details

- $a$ means both primitive actions and subtasks (options)
- Context-dependent vs. context-independent
- Higher-level subtasks are essentially policies over options
  - But subtasks are learned too
  - And the values propagate correctly
- What does $C_i^\pi(s, a)$ mean? (Nick slides)
Discussion Points

• What does MAXQ-Q buy you over flat?
Discussion Points

- What does MAXQ-Q buy you over flat?
- What does polling buy you over flat?