

CS394R
Reinforcement Learning:
Theory and Practice

Peter Stone

Department of Computer Science
The University of Texas at Austin

Good Afternoon Colleagues

- Are there any questions?

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 for each subtask

MAXQ

- Defines how to learn given a task hierarchically
- Does not address how to construct the hierarchy
- Strives for **recursive optimality**— local optimality for each subtask
 - 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 for each subtask
 - 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 for each subtask
 - 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 $C_i^\pi(s, a)$ 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?(Dietterich 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? (Dietterich slides)
- How does equation (2) relate to flat Q?

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? (Dietterich slides)
- How does equation (2) relate to flat Q?
- Theorem 2

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? (Dietterich slides)
- How does equation (2) relate to flat Q?
- Theorem 2

- Polling: Why the dip in the graph in Figure 6?

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?

Discussion Points

- What does MAXQ-Q buy you over flat?
- What does polling buy you over flat?
- Would learning the subtasks from the bottom up help?

Discussion Points

- What does MAXQ-Q buy you over flat?
- What does polling buy you over flat?
- Would learning the subtasks from the bottom up help?