Function Approximation via Tile Coding: Automating Parameter Choice

Alexander Sherstov and **Peter Stone**

Department of Computer Sciences
The University of Texas at Austin

About the Authors

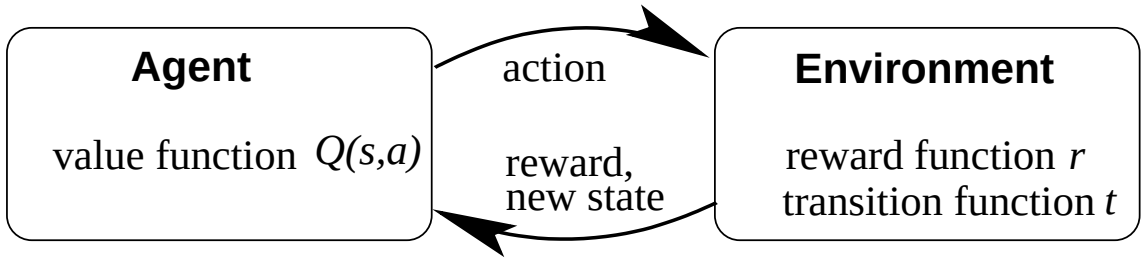
Alex Sherstov

Peter Stone

Thanks to Nick Jong for presenting!

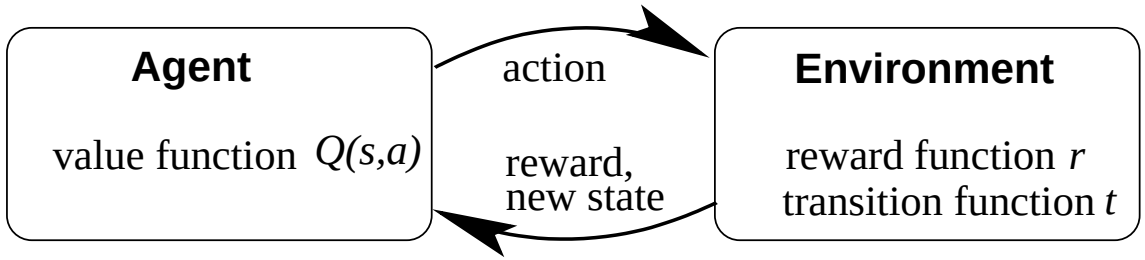
Overview

- TD reinforcement learning
  - Leading abstraction for decision making
  - Uses ***function approximation*** to store ***value function***



Overview

- TD reinforcement learning
  - Leading abstraction for decision making
  - Uses ***function approximation*** to store ***value function***



- Existing methods
  - Discretization, neural nets, radial basis, case-based, ...

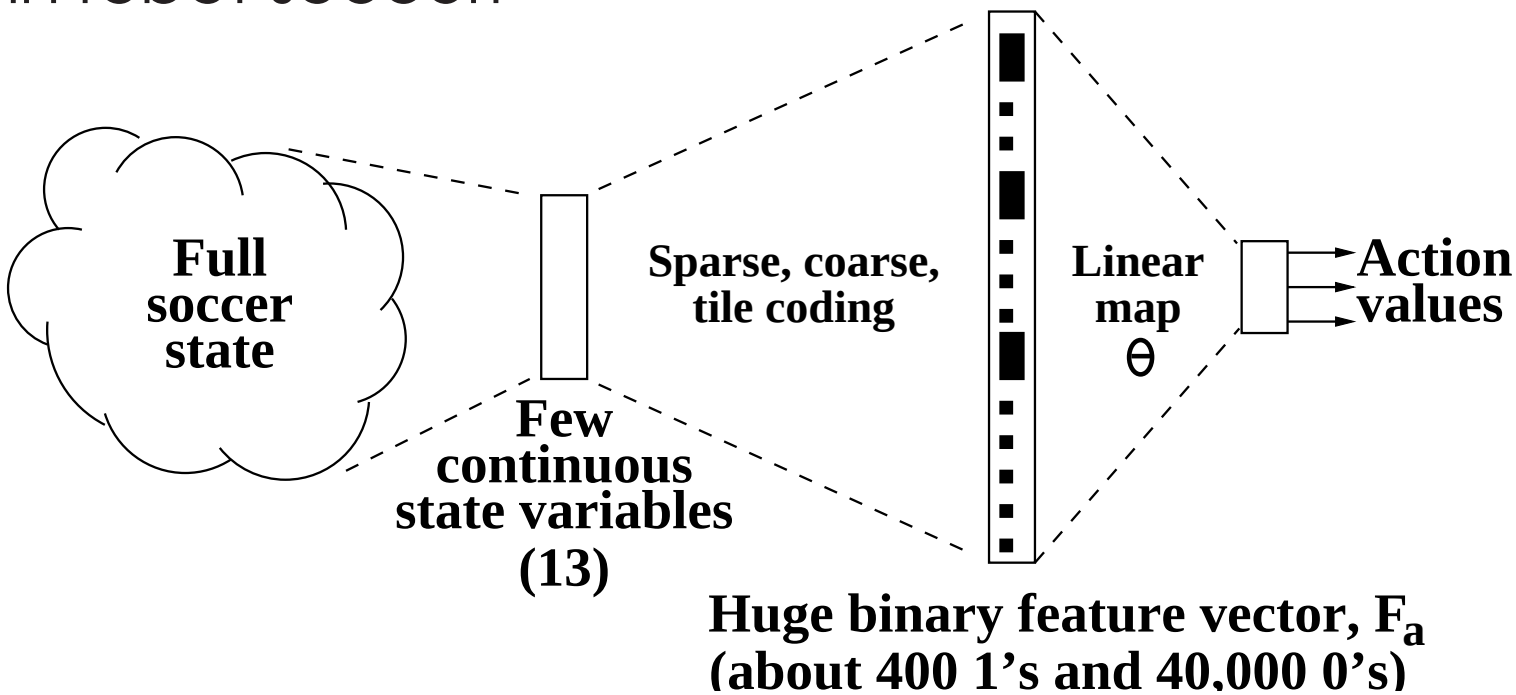
    (Santamaria et al., 1997)
  - Trade-offs:

    representational power, ***time/space*** req's, ***ease*** of use

Overview, cont.

- "Happy medium": ***tile coding***
  - Widely used in RL

    (Stone and Sutton, 2001, Santamaria et al., 1997, Sutton, 1996).
  - Use in robot soccer:



Our Results

- We show that:
  - Tile-coding is ***parameter-sensitive***
  - Optimal parameterization depends on the ***problem*** and ***elapsed training time***

Our Results

- We show that:
  - Tile-coding is ***parameter-sensitive***
  - Optimal parameterization depends on the ***problem*** and ***elapsed training time***

- We contribute:
  - An ***automated*** parameter-adjustment scheme
  - Empirical validation

Background: Reinforcement Learning

- RL problem given by $\langle \mathcal{S}, \mathcal{A}, t, r \rangle$:
  - $\mathcal{S}$, set of ***states***;
  - $\mathcal{A}$, set of ***actions***;
  - $t : \mathcal{S} \times \mathcal{A} \to \Pr(\mathcal{S})$, ***transition*** function;
  - $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, ***reward*** function.

Background: Reinforcement Learning

- RL problem given by $\langle \mathcal{S}, \mathcal{A}, t, r \rangle$:
  - $\mathcal{S}$, set of ***states***;
  - $\mathcal{A}$, set of ***actions***;
  - $t : \mathcal{S} \times \mathcal{A} \to \Pr(\mathcal{S})$, ***transition*** function;
  - $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, ***reward*** function.

- Solution:
  - ***policy*** $\pi^* : \mathcal{S} \to \mathcal{A}$ that maximizes ***return*** $\sum_{i=0}^{\infty} \gamma^i r_i$
  - Q-learning: find π^* by approximating optimal ***value function*** $Q^* : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$

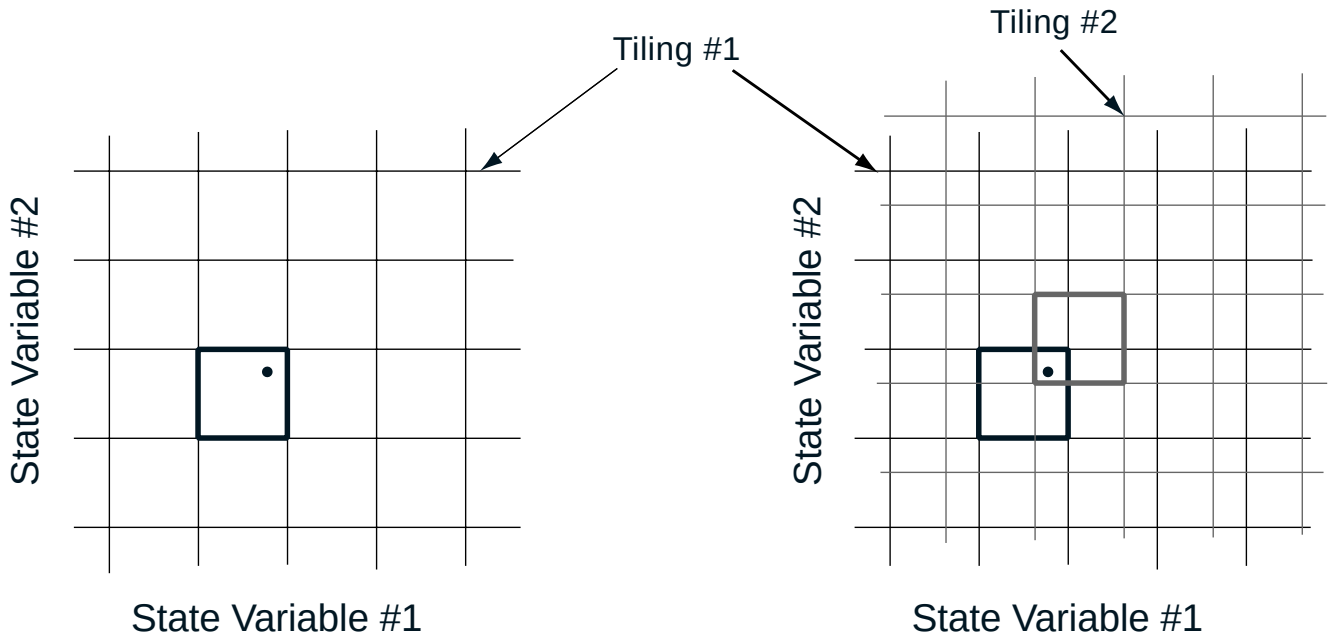
Background: Reinforcement Learning

- RL problem given by $\langle \mathcal{S}, \mathcal{A}, t, r \rangle$:
  - $\mathcal{S}$, set of ***states***;
  - $\mathcal{A}$, set of ***actions***;
  - $t : \mathcal{S} \times \mathcal{A} \to \Pr(\mathcal{S})$, ***transition*** function;
  - $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, ***reward*** function.

- Solution:
  - ***policy*** $\pi^* : \mathcal{S} \to \mathcal{A}$ that maximizes ***return*** $\sum_{i=0}^{\infty} \gamma^i r_i$
  - Q-learning: find π^* by approximating optimal ***value function*** $Q^* : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$

- Need FA to generalize Q^* to unseen situations

Background: Tile Coding



- Maintaining arbitrary $f : \mathcal{D} \to \mathbb{R}$ (often $\mathcal{D} = \mathcal{S} \times \mathcal{A}$):
  - $\mathcal{D}$ partitioned into ***tiles***, each with a ***weight***
  - Each partition is a ***tiling***; several used
  - Given $x \in \mathcal{D}$, sum weights of participating tiles
    $\Longrightarrow$ get $f(x)$

Background: Tile Coding Parameters

- We study ***canonical univariate*** tile coding:
  - w, tile width (same for all tiles)
  - t, # of tilings ("generalization breadth")
  - $r = w/t$, resolution
  - tilings uniformly offset

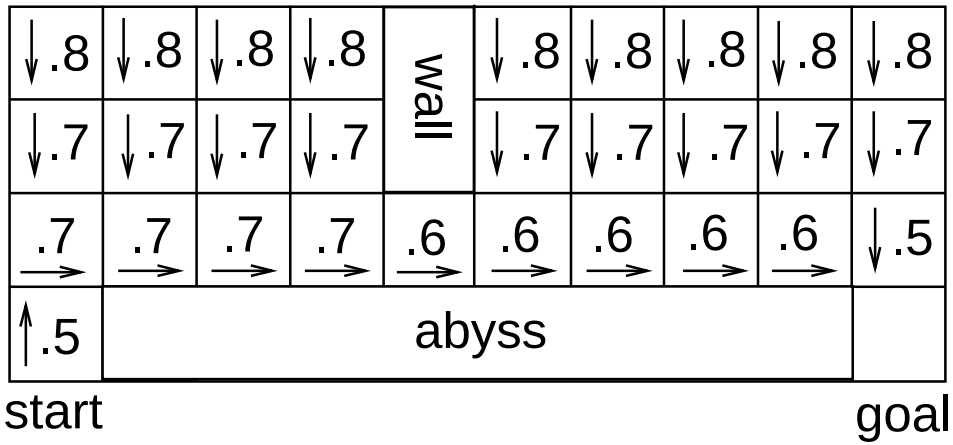
Background: Tile Coding Parameters

- We study ***canonical univariate*** tile coding:
  - w, tile width (same for all tiles)
  - t, # of tilings ("generalization breadth")
  - $r = w/t$, resolution
  - tilings uniformly offset

- Empirical model:
  - ***Fix*** resolution r, ***vary*** generalization breadth t
  - Same resolution $\implies$ same rep power, asymptotic perf
  - But: t affects intermediate performance
  - ***How to set*** t***?***

Testbed Domain: Grid World

- Domain and optimal policy:



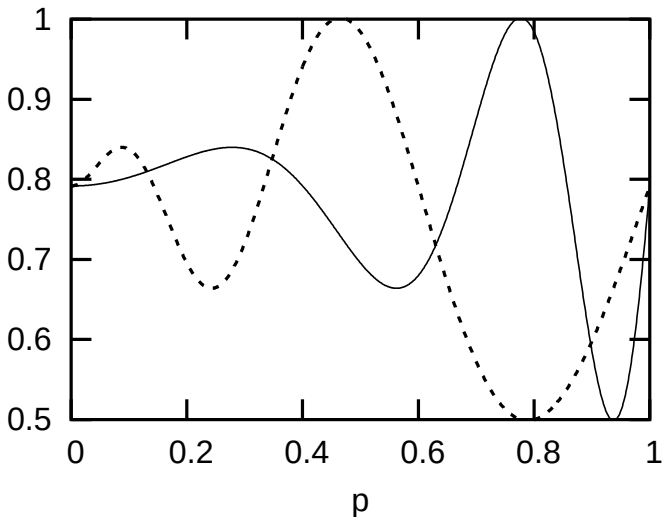
- Episodic task (cliff, goal cells terminal)

- Actions:

$$(d, p) \quad \in \quad \{\uparrow, \downarrow, \rightarrow, \leftarrow\} \times [0, 1]$$

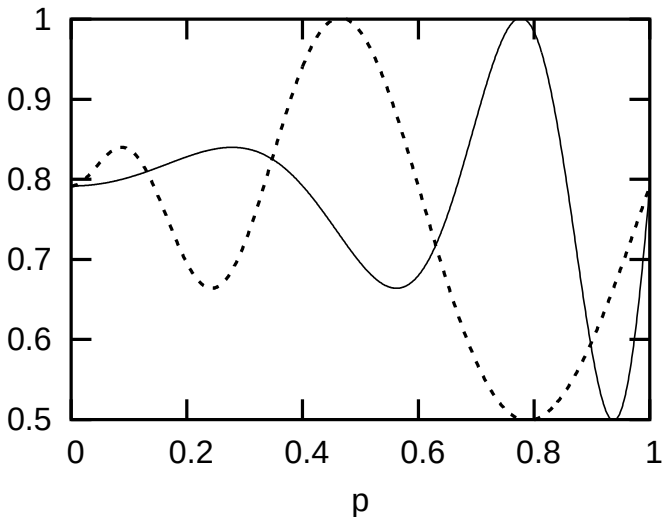
Testbed Domain, cont.

- Move succeeds w/ prob. $F(p)$, random o/w; F varies from cell to cell:



Testbed Domain, cont.

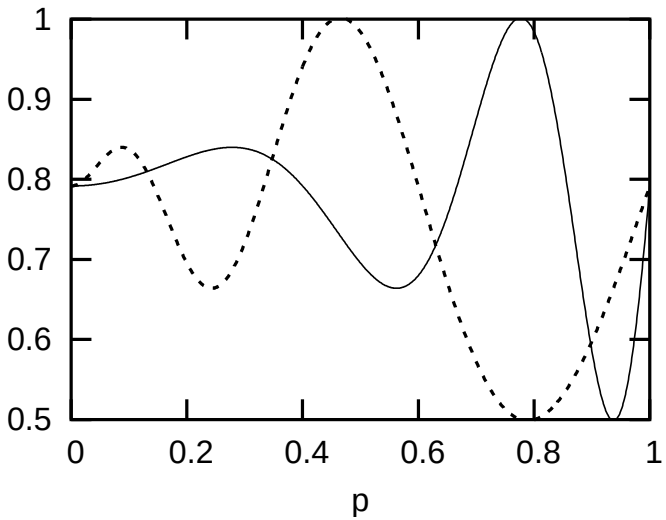
- Move succeeds w/ prob. $F(p)$, random o/w; F varies from cell to cell:



- 2 reward functions:
  -100 cliff, $+100$ goal, -1 o/w ("informative");
  $+100$ goal, 0 o/w ("uninformative")

Testbed Domain, cont.

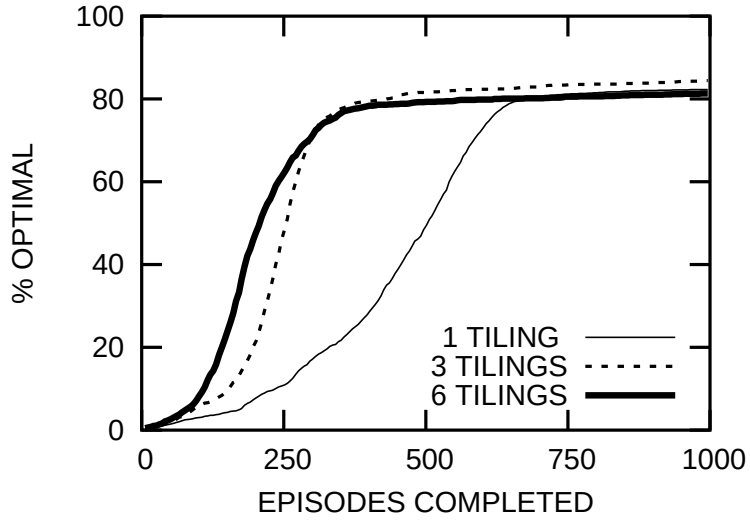
- Move succeeds w/ prob. $F(p)$, random o/w;
  F varies from cell to cell:



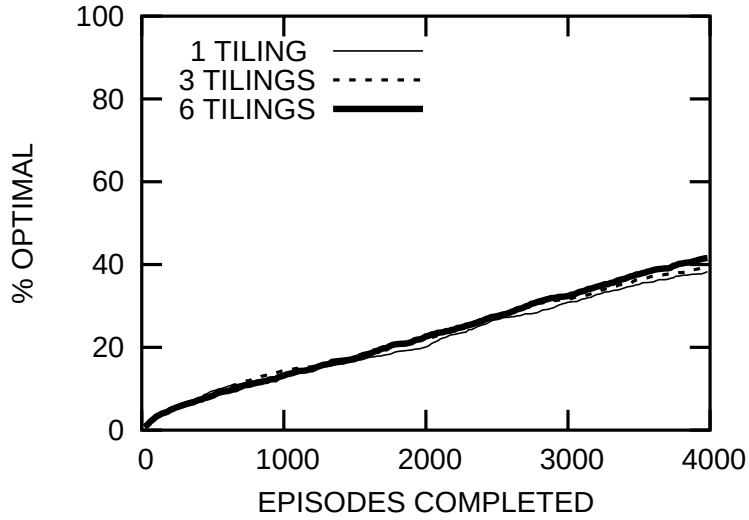
- 2 reward functions:
  -100 cliff, $+100$ goal, -1 o/w ("informative");
  $+100$ goal, 0 o/w ("uninformative")

- Use of tile coding: generalize over actions (p)

Generalization Helps Initially



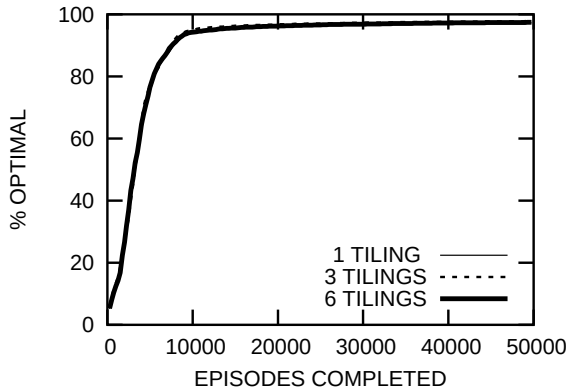
informative reward



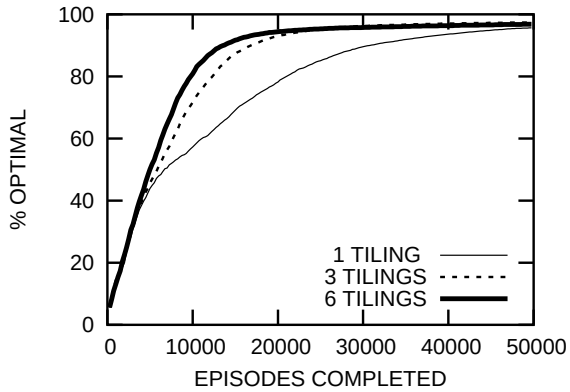
uninformative reward

Generalization improves cliff avoidance.

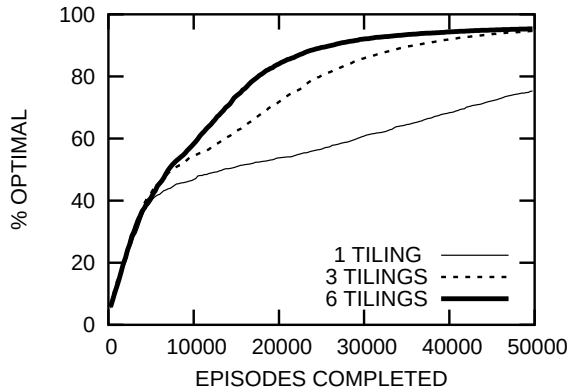
Generalization Helps Initially, cont.



$\alpha = 0.5$



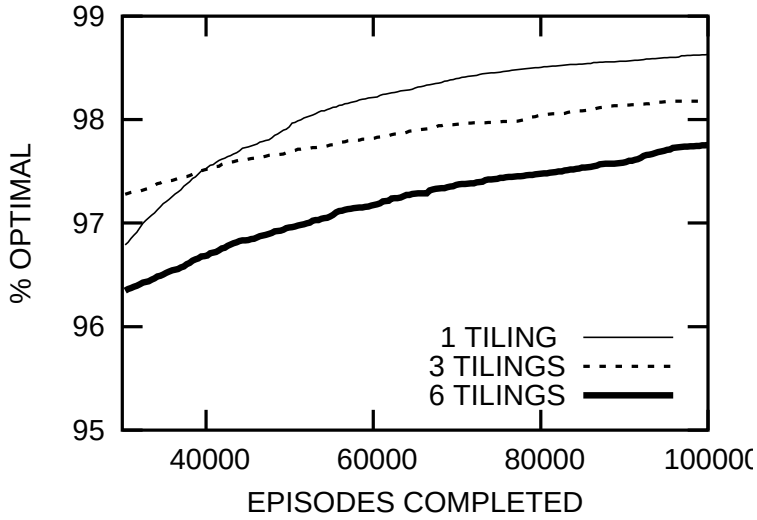
$\alpha = 0.1$



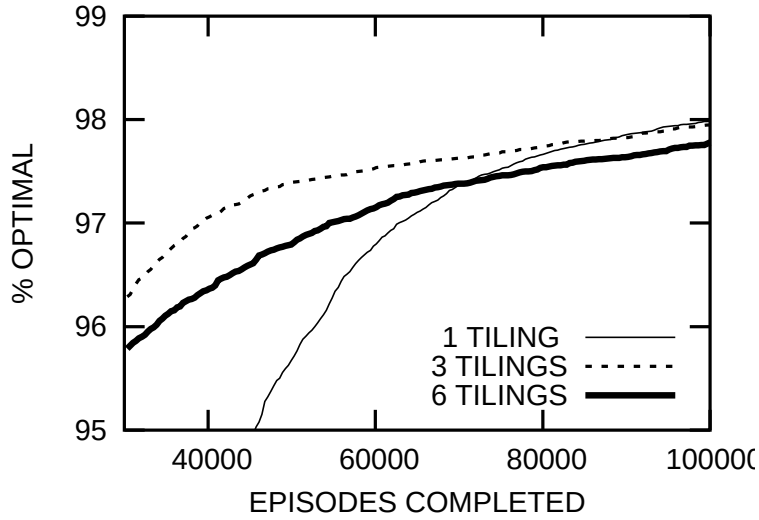
$\alpha = 0.05$

Generalization improves discovery of better actions.

Generalization Hurts Eventually



informative reward



uninformative reward

Generalization slows convergence.

Adaptive Generalization

- Best to ***adjust*** generalization over time

Adaptive Generalization

- Best to ***adjust*** generalization over time
- Solution: ***reliability index*** $\rho(s,a) \in [0,1]$
  - $\rho(s,a) \approx 1 \implies Q(s,a)$ reliable (and vice versa)
  - ***large*** backup error on (s,a) ***decreases*** $\rho(s,a)$ (and vice versa)

Adaptive Generalization

- Best to ***adjust*** generalization over time
- Solution: ***reliability index*** $\rho(s,a) \in [0,1]$
  - $\rho(s,a) \approx 1 \implies Q(s,a)$ reliable (and vice versa)
  - ***large*** backup error on (s,a) ***decreases*** $\rho(s,a)$ (and vice versa)
- Use of $\rho(s,a)$:
  - An update to $Q(s,a)$ is generalized to largest nearby region R that is ***unreliable on average***:

$$\frac{1}{|R|}\sum_{(s,a)\in R} \rho(s,a) \leq \frac{1}{2}$$

Effects of Adaptive Generalization

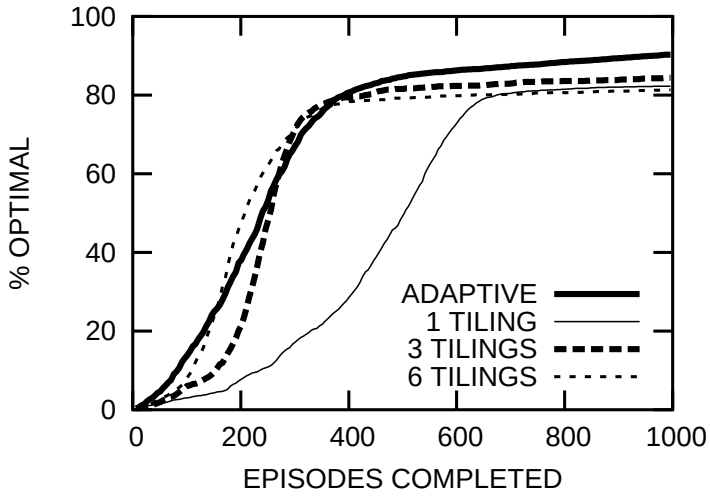
- ***Time-variant*** generalization
  - Encourages generalization when $Q(s, a)$ changing
  - Suppresses generalization near convergence

Effects of Adaptive Generalization

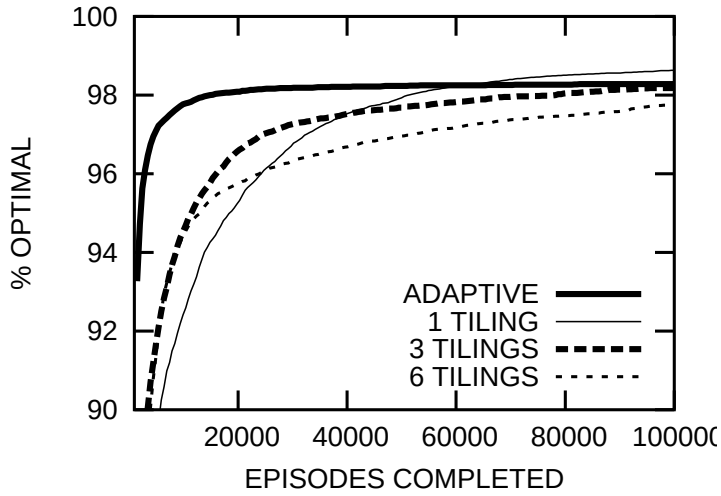
- ***Time-variant*** generalization
  - Encourages generalization when $Q(s, a)$ changing
  - Suppresses generalization near convergence

- ***Space-variant*** generalization
  - Rarely-visited states benefit from generalization for a longer time

Adaptive Generalization at Work



episodes 0–1000



episodes 1000–1000000

Adaptive generalization better than ***any*** fixed setting.

Conclusions

- Precise empirical study of ***parameter choice*** in tile coding

Conclusions

- Precise empirical study of ***parameter choice*** in tile coding

- ***No single setting*** ideal for all problems, or even throughout learning curve on the ***same*** problem

Conclusions

- Precise empirical study of ***parameter choice*** in tile coding

- ***No single setting*** ideal for all problems, or even throughout learning curve on the ***same*** problem

- Contributed algorithm for ***adjusting*** parameters as needed in different regions of $\mathcal{S} \times \mathcal{A}$ (***space-variant*** gen.) and at different learning stages (***time-variant*** gen.)

Conclusions

- Precise empirical study of ***parameter choice*** in tile coding

- ***No single setting*** ideal for all problems, or even throughout learning curve on the ***same*** problem

- Contributed algorithm for ***adjusting*** parameters as needed in different regions of $\mathcal{S} \times \mathcal{A}$ (***space-variant*** gen.) and at different learning stages (***time-variant*** gen.)

- Showed ***superiority*** of this adaptive technique to any fixed setting

References

(Santamaria et al., 1997) Santamaria, J. C., Sutton, R. S., and Ram, A. (1997). Experiments with reinforcement learning in problems with continuous state and action spaces. *Adaptive Behavior*, 6(2):163–217.

(Stone and Sutton, 2001) Stone, P. and Sutton, R. S. (2001). Scaling reinforcement learning toward RoboCup soccer. In *Proc. 18th International Conference on Machine Learning (ICML-01)*, pages 537–544. Morgan Kaufmann, San Francisco, CA.

(Sutton, 1996) Sutton, R. S. (1996). Generalization in reinforcement learning: Successful examples using sparse coarse coding. In Tesauro, G., Touretzky, D., and Leen, T., editors, *Advances in Neural Information Processing Systems 8*, pages 1038–1044, Cambridge, MA. MIT Press.