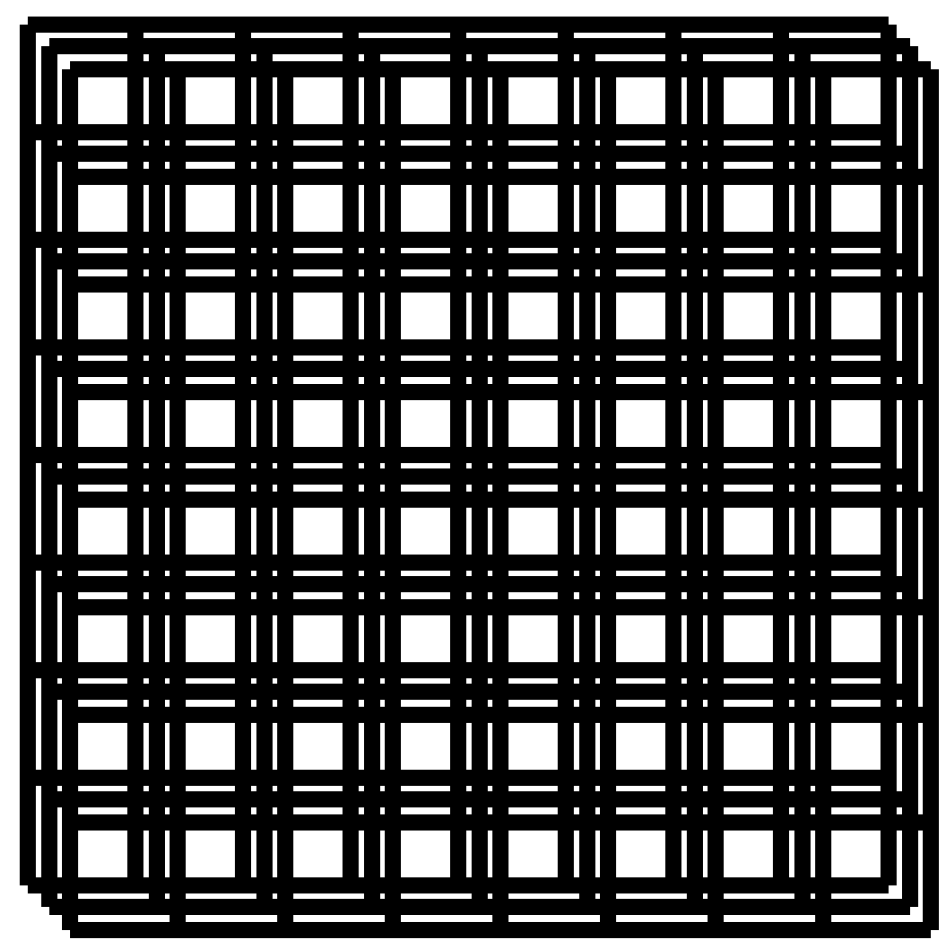


Motivation for Online Generalization

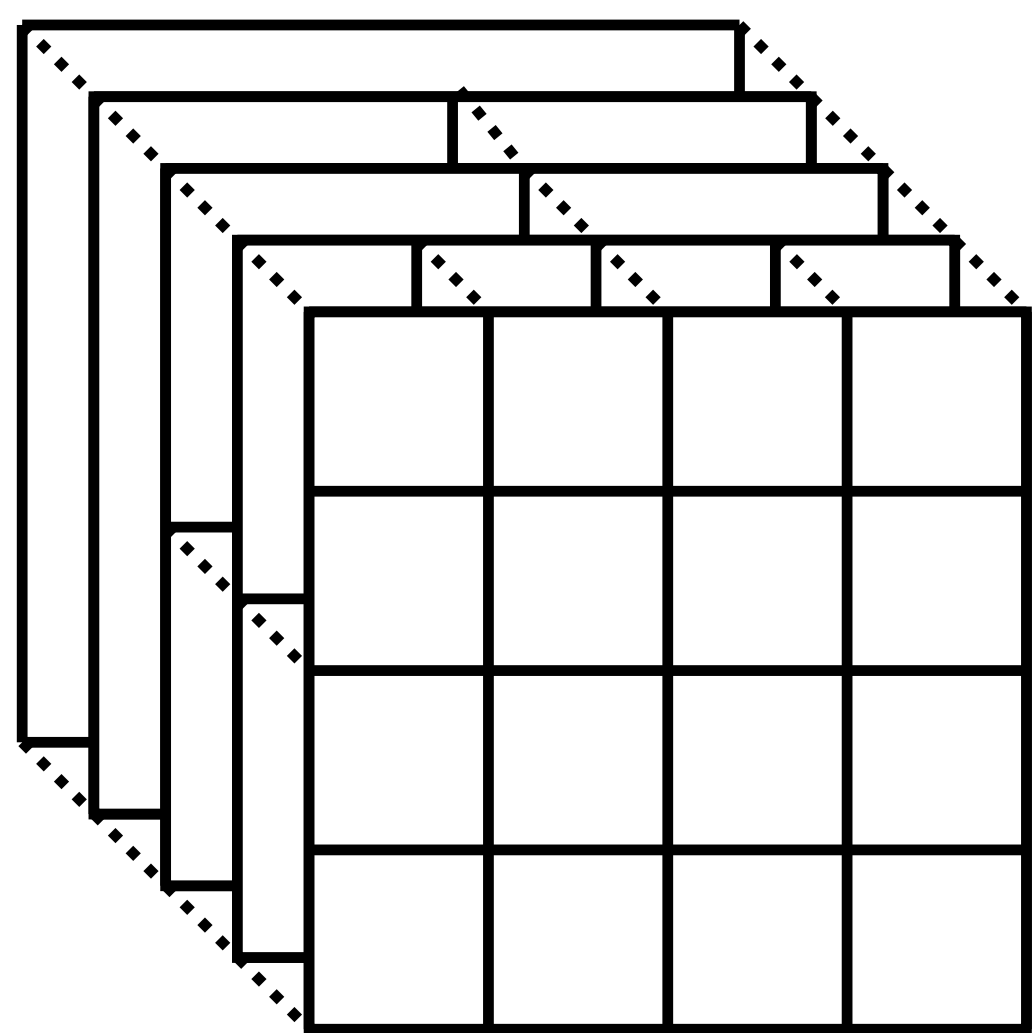
- ▶ One-to-one mappings are not viable
- ▶ Difficult to design a value function a priori
- ▶ Tight reactivity constraints

Tile Coding / CMAC



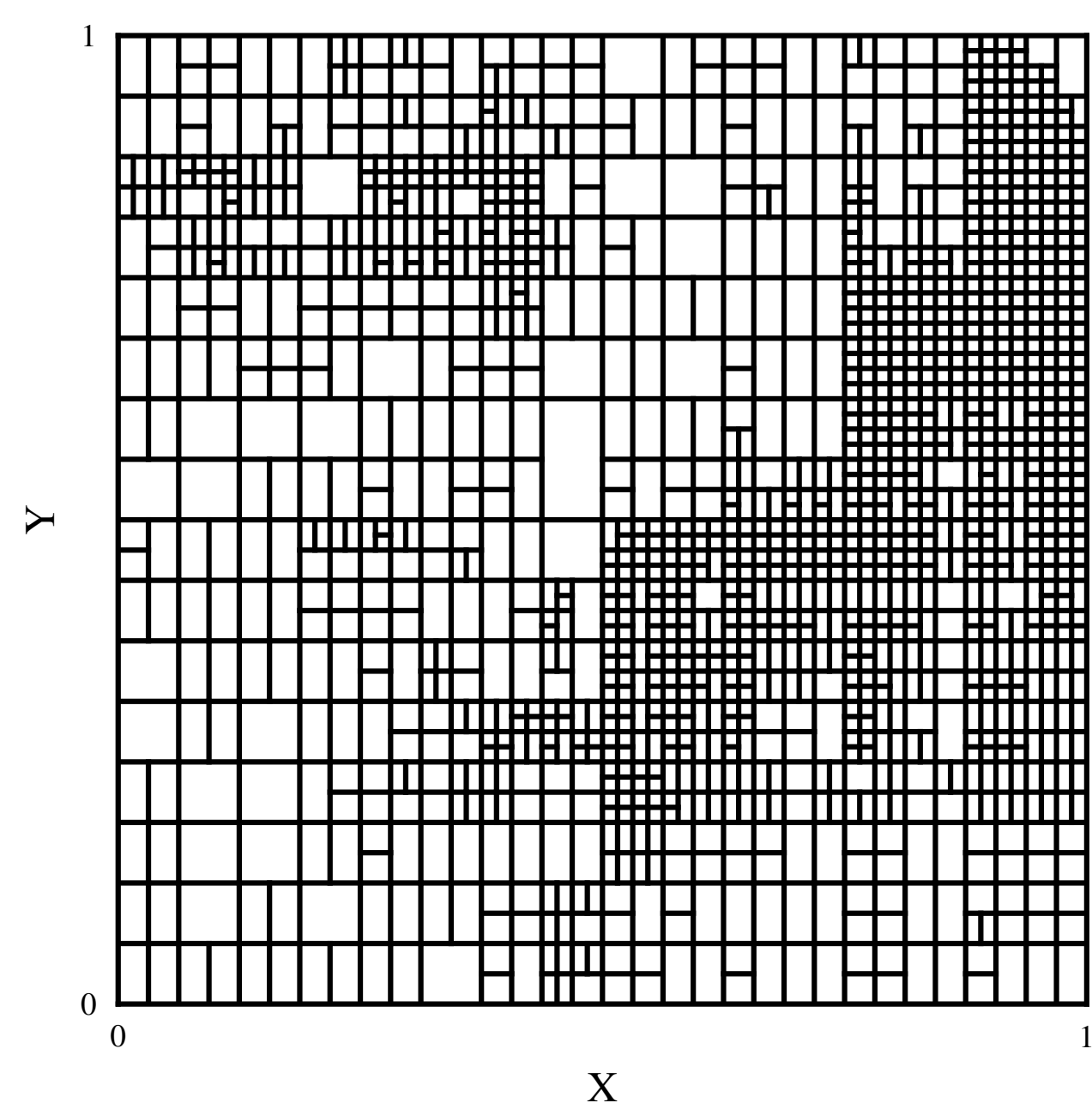
Multiple tilings partition the state-space

Hierarchical Tile Coding



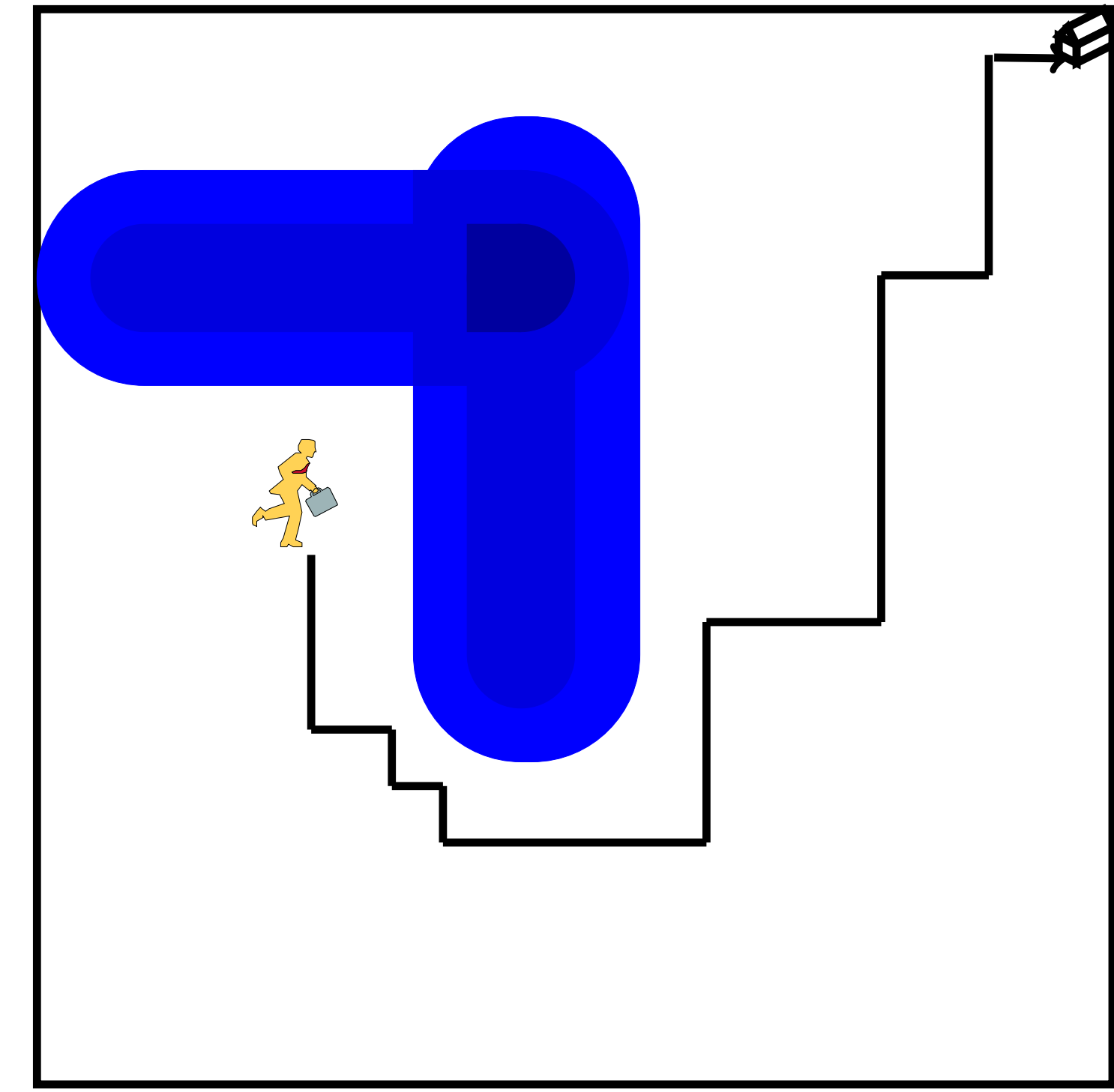
A variable resolution CMAC

Incremental Hierarchical Tile Coding

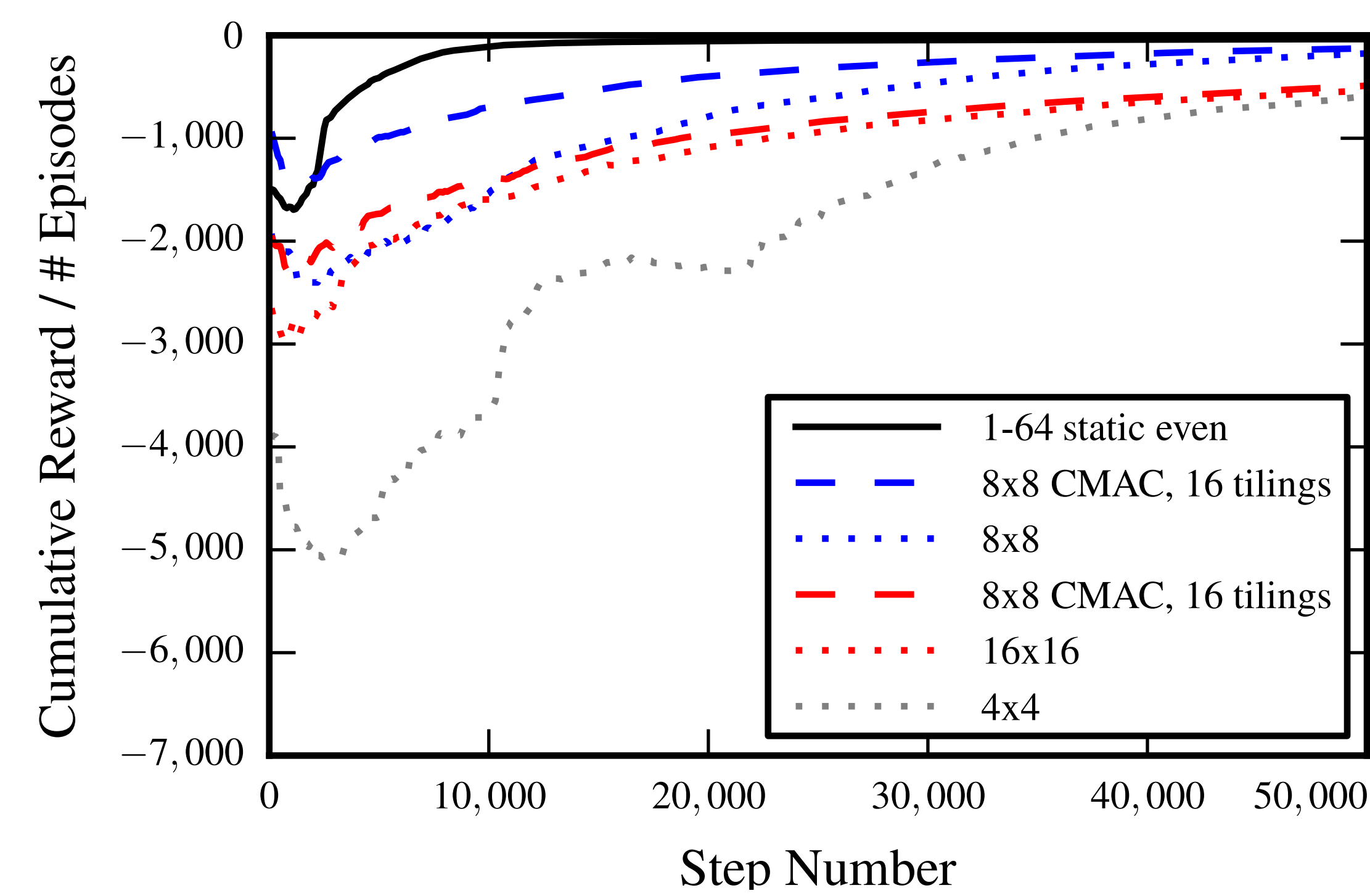


A flattened incremental hierarchical tile coding

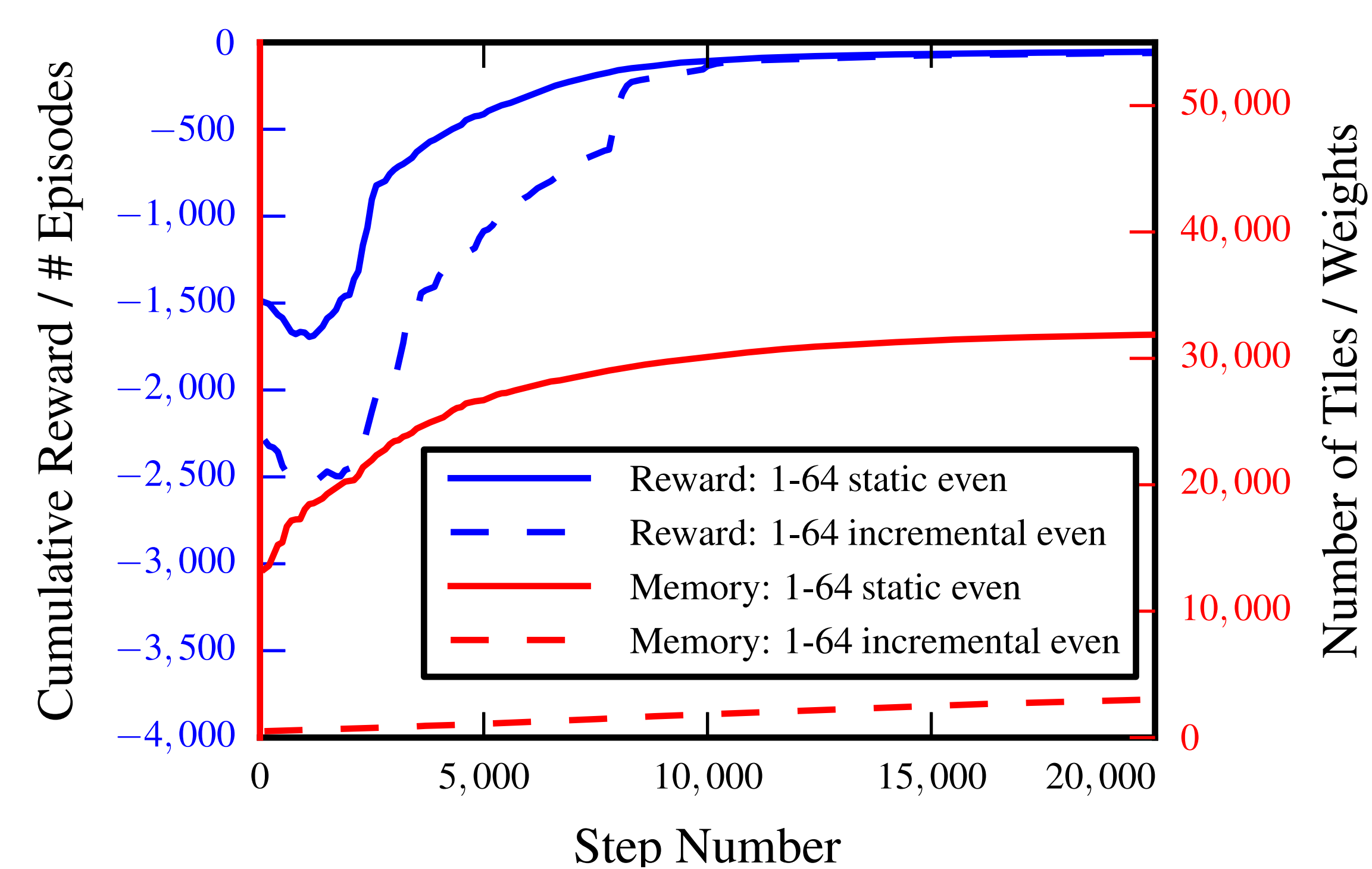
Puddle World



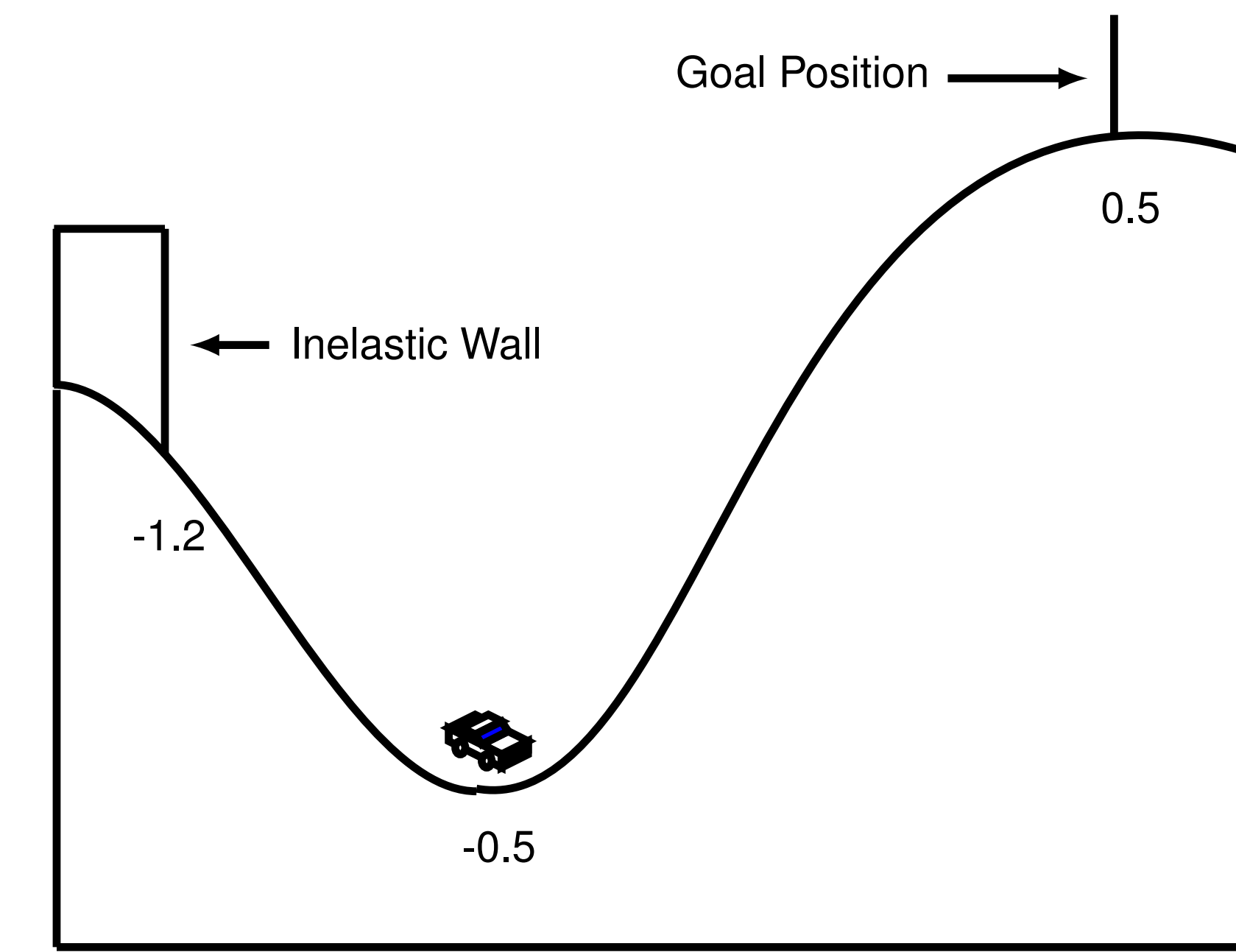
Static Tile Coding: Puddle World



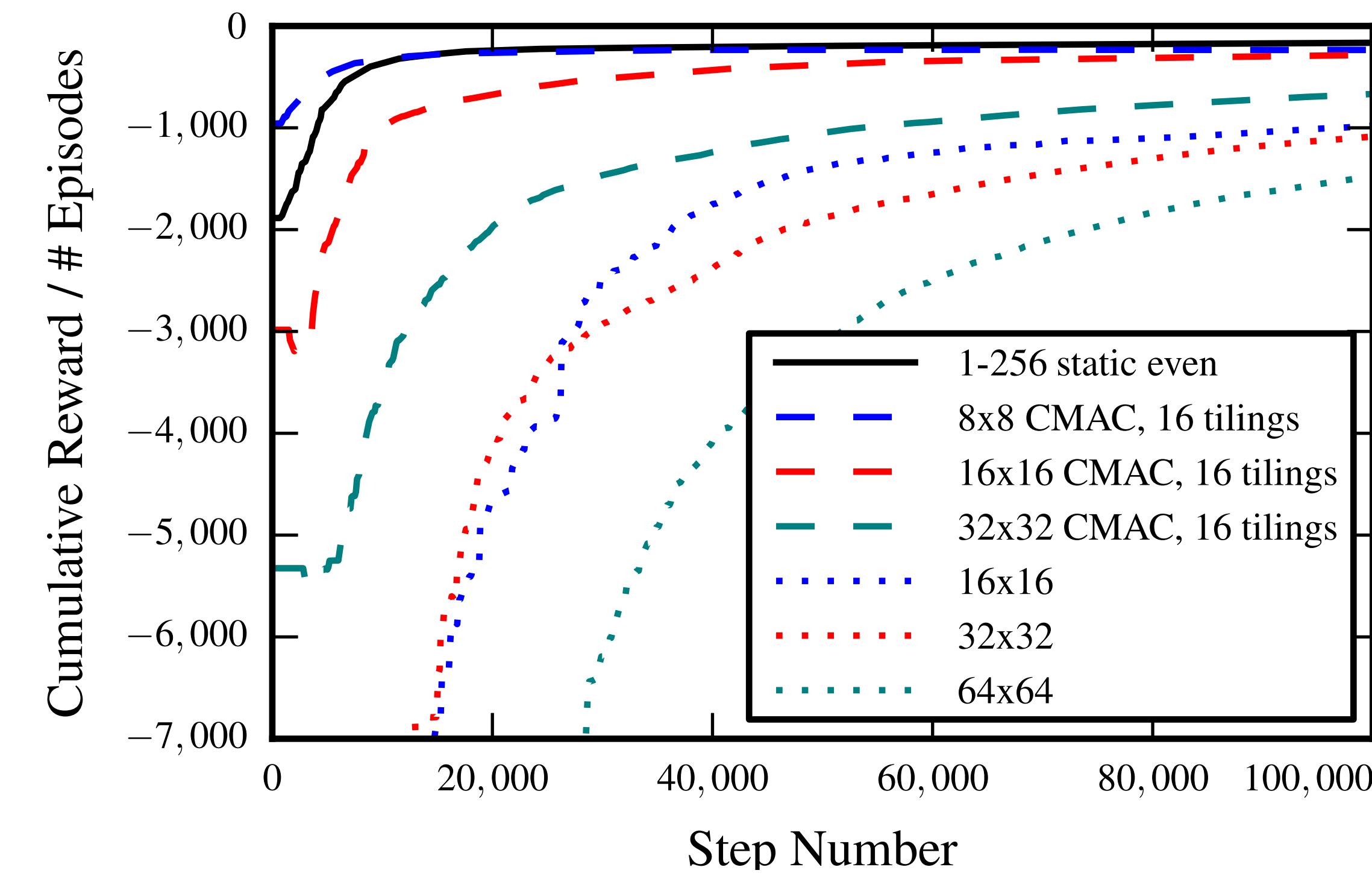
Incremental Tile Coding: Puddle World



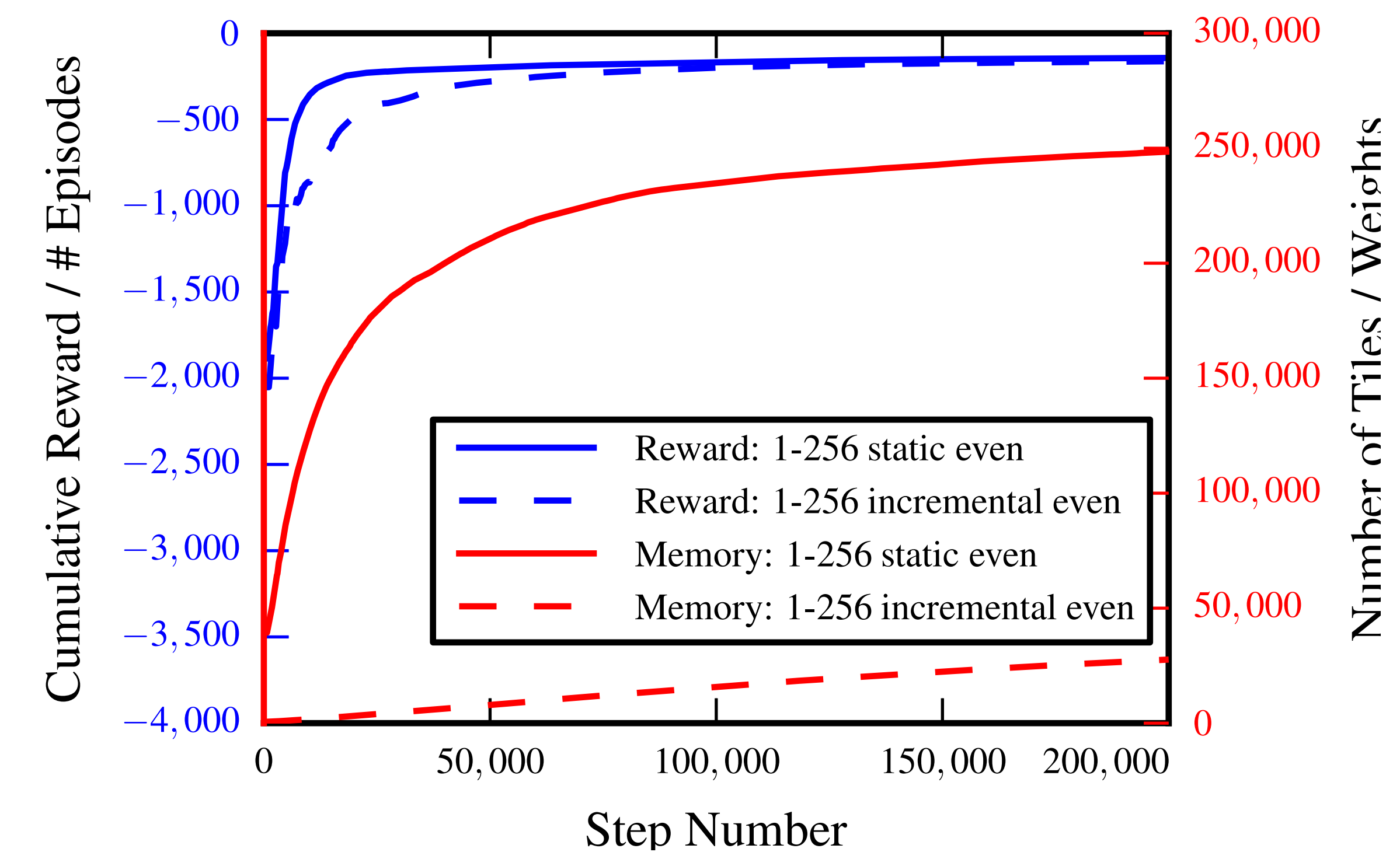
Mountain Car



Static Tile Coding: Mountain Car



Incremental Tile Coding: Mountain Car



Developments

- ▶ Hierarchical tile coding
 - ▶ A CMAC variation
 - ▶ Uses linear function approximation
 - ▶ Subtilings learn more refined policies
 - ▶ Allows fast learning early on
 - ▶ Supports good terminal policies
- ▶ Incremental hierarchical tile coding
 - ▶ Still learns quickly
 - ▶ Achieves good terminal policies
 - ▶ Uses significantly less memory
- ▶ Cumulative absolute temporal difference error (CATDE) refinement criterion

Experiments

Our agents:

- ▶ Use an epsilon-greedy exploration strategy
- ▶ Learn using Q-learning and linear function approximation

Their value functions:

- ▶ Static hierarchical tile codings
 - ▶ Fixed number of complete tilings
 - ▶ Compared to individual tilings and CMACs
- ▶ Incremental hierarchical tile codings
 - ▶ Variable number of partial tilings

Discussion

- ▶ Hierarchical tile coding performs better than either a subset of its tilings or a classical CMAC implementation.
- ▶ Incremental hierarchical tile codings which perform nearly as well, but use significantly less memory.
- ▶ Hierarchical tile codings dominate simpler tile-splitting methods.

Future Work

- ▶ Moving toward relational reinforcement learning (RRL) domains, like Blocks World, Tetris, and eventually Liar's Dice
- ▶ Shifted from a trie-based value function representation to one based on a Rete
- ▶ Refocusing on modifying the value function where the policy is likely to improve as a result