Transerring Instances for Model-Based Reinforcement Learning

Seminar aus Künstlicher Intelligenz – Felix Klose

Overview

- Transfer Learning
- Model-Based Reinforcement Learning
- TIMBREL
- 2D/3D Mountain Car
- Experiments

Transfer Learning

- Reinforcement Learning Tasks often need significant amount of data
- Obtaining data can be very time intensive
- Transfer Learning: Exploit data from related, simpler tasks
- Example: Learn 2D Mountain Car and learn 3D Mountain Car using the learned data from 2D Mountain Car
- Paper: „Transferring Instances for Model-Based Reinforcement Learning“ by Matthew E. Taylor, Nicholas K. Jong and Peter Stone

Overview

- Transfer Learning
- Model-Based Reinforcement Learning
- TIMBREL
- 2D/3D Mountain Car
- Experiments

Model-Based Reinforcement Learning

- Model-Free learning: Only utility of actions in states are learned, not the effects of actions
- Examples: Q-Learning, Sarsa
- Model-Based learning: Effects of actions and reward-function are learned
- For example: R-Max

Markov Decision Processes (MDP)

- $s \in S, s = \langle x_1, x_2 \ldots, x_k \rangle$: States with k state variables
- Initial state $x_{initial}$ and goal state x_{goal}
- Set of available actions A
- Transision function $T: S \times A \rightarrow S$
- Reward function $R: S \times A \rightarrow \mathbb{R}$
- Policy $\pi: S \rightarrow A$

R-Max

- Model-Based learning algorithm
- Stochastically approximates an MDP and Action-Value-Function
  - By estimating transitions and rewards
- NOT guaranteed to converge to an optimal policy
- Used as learning algorithm in TIMBREL
- Any other Model-Based algorithm can be used, R-Max used for it‘s high learning performance

Overview

- Transfer Learning
- Model-Based Reinforcement Learning
- TIMBREL
- 2D/3D Mountain Car
- Experiments

TIMBREL (Transferring Instances for Model-Based Reinforcement Learning)

1. Learn the source task
2. Provide learned transitions to the target task
3. **While** training in target task **do**

   If RL algorithm has insufficiont data for learning

   While RL algorithm has insufficient data

   transfer transitions from the source task to the target task

   if no unused source task transitions are availbale

   exit inner loop

   Use transferred data in target task

TIMBREL

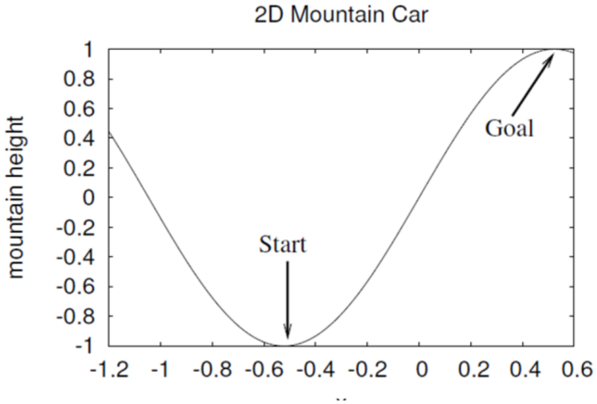
- Transfer function hand-made per problem
- Example for transfer function will be given later
- „Insufficient data“ not precisely defined in the paper
  - Depends on problem

Overview

- Transfer Learning
- Model-Based Reinforcement Learning
- TIMBREL
- 2D/3D Mountain Car
- Experiments

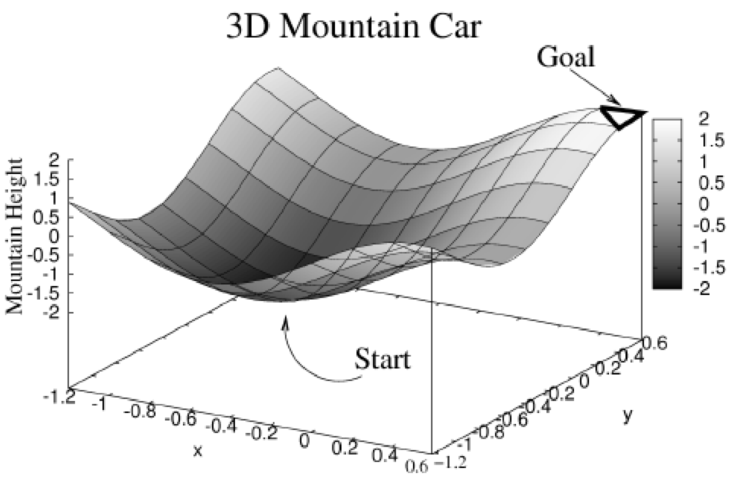
2D Mountain Car

- Agent starts at Start
- Task: reach goal
- Actions: *Left*, *Neutral*, *Right*
- State variables: horizontal position x, velocity $\dot{x}$
- Each timestep, $-0.025(\cos(3x))$ is added to $\dot{x}$ (gravity)
- Actions *Left*, *Neutral*, *Right* add -0.0007, 0 and 0.0007 to $\dot{x}$
- Agent can not reach goal state by just going *Right* all the time
- Each action has the reward -1



3D Mountain Car

- Agent starts at Start
- Task: reach goal
- Actions: *Neutral*, *West*, *East*, *South*, *North*
- State variables: Positions x and y, velocities $\dot{x}$ and $\dot{y}$
- West and East modify $\dot{x}$ by -0.0007 and 0.0007, North and South modify $\dot{y}$
- Gravity modifies $\dot{x}$ by $-0.025\cos(3x)$ and $\dot{y}$ 13by $-0.025\cos(3y)$



Mountain Car Transfer

- Source Task: 2D Mountain Car
- Target Task: 3D Mountain Car
- Actions and state variables are mapped according to the mapping table

Inter-task Mapping for Mountain Car

Action Mapping	State Variable Mapping
$\chi_A(\text{Neutral}) = \text{Neutral}$	$\chi_S(x) = x$
$\chi_A(\text{North}) = \text{Right}$	$\chi_S(\dot{x}) = \dot{x}$
$\chi_A(\text{East}) = \text{Right}$	or
$\chi_A(\text{South}) = \text{Left}$	$\chi_S(y) = x$
$\chi_A(\text{West}) = \text{Left}$	$\chi_S(\dot{y}) = \dot{x}$

Mountain Car Transfer

- Mapping is done 2 times per transfer:
  - x and $\dot{x}$ are mapped to the source task

    For transferring the transitions, x and $\dot{x}$ are adjusted according to the transferred instance and y and $\dot{y}$ stay fixed
  - y and $\dot{y}$ are mapped analogously

Overview

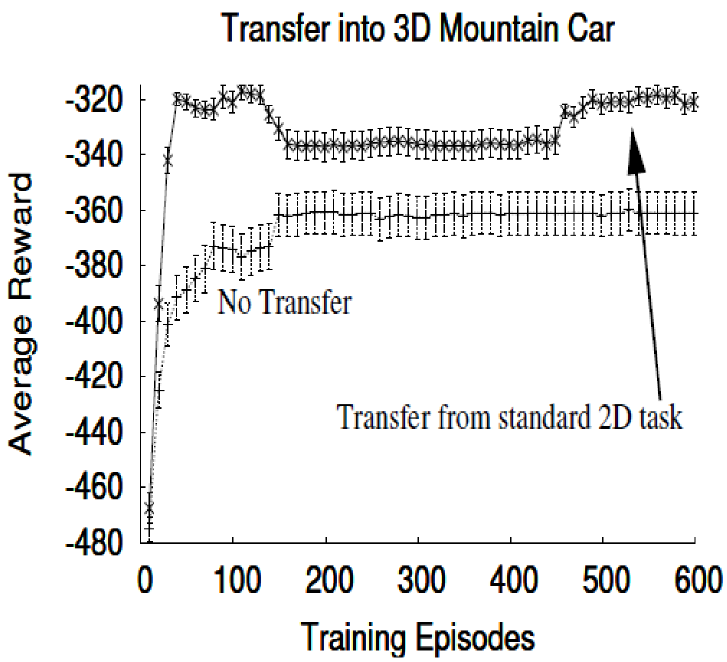
- Transfer Learning
- Model-Based Reinforcement Learning
- TIMBREL
- 2D/3D Mountain Car
- Experiments

Experiments

- R-Max 3D Mountain Car experiments to test learning speed
  - 50 Experiments
  - 12 Learning episodes with 4000 samples
  - 10 out of 12 runs found goal state

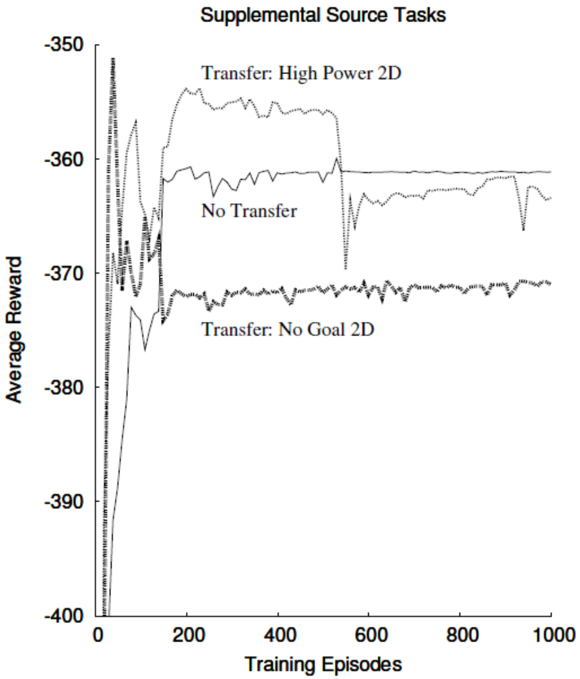
Experiments

- Transfer experiments
- TIMBREL used to transfer from standard 2D task
- Transfer significantly improves learning performance



Experiments

- High Power 2D: Velocity $\dot{x}$ is modified by higher value
- No Goal 2D: no goal state defined



Results

- Transfer can significantly improve learning rates for Reinforcement Learning
- Performance of transfer heavily depends on how similar source and target task are
- Performance in more complex tasks remains to be tested

Questions?