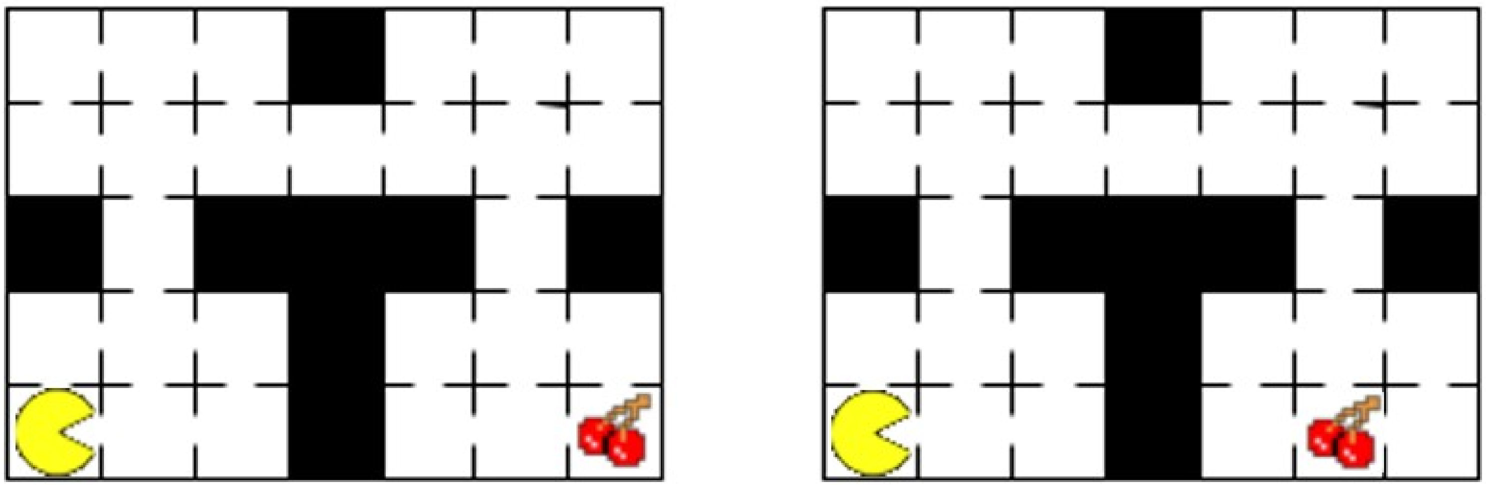
Transfer Learning – With similar MDPs

Advanced Topics in Reinforcement Learning Seminar

Mike Smyk



[Phillips, 2006]

Motivation

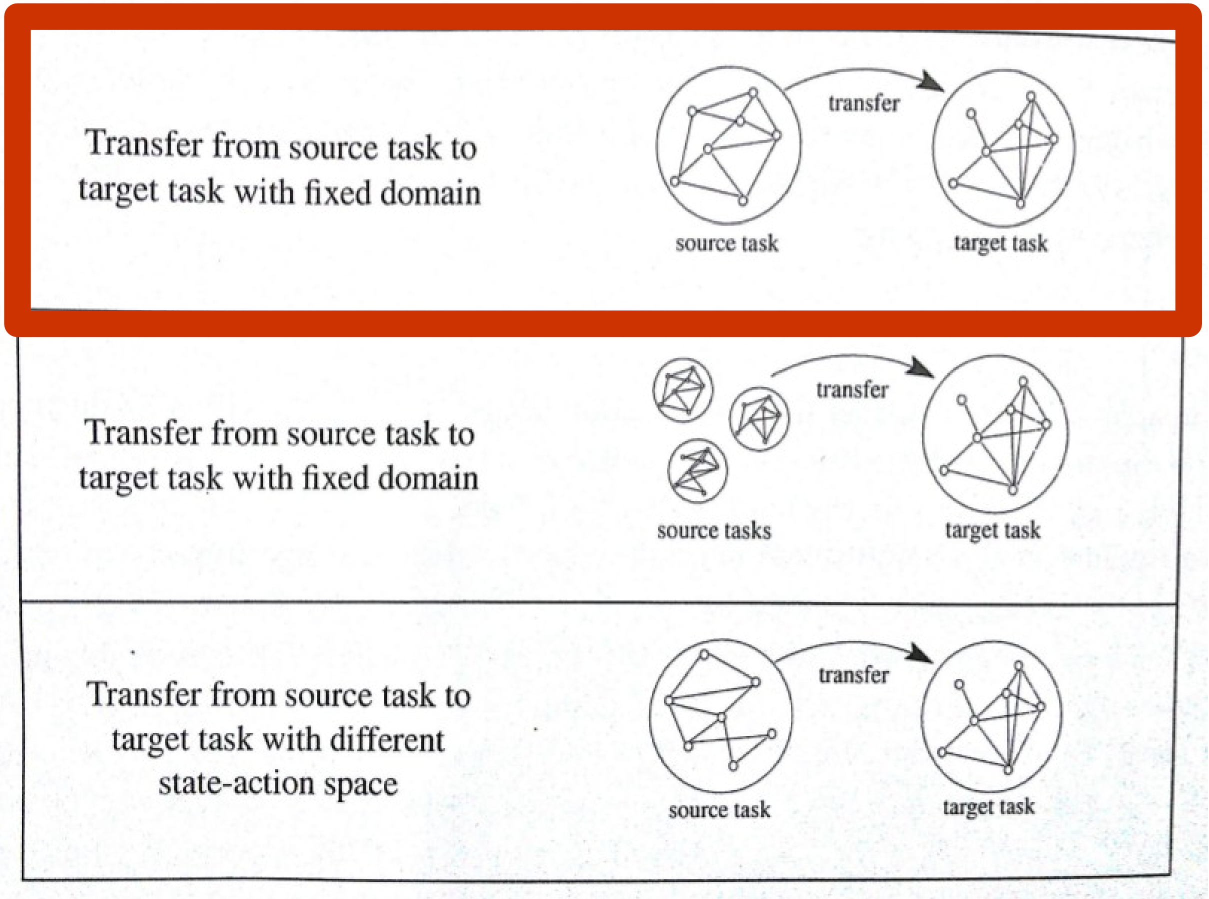
- Learning optimal policy is time-consuming
- Requires lots of data

→ Use computed policies from other **similar** MDP(s)

- Problem:
  - What are ***similar*** MDPs?

But first: How to transfer knowledge

Main Transfer Settings



[Wiering, 2012]

Definition - MDP

Markov Decision Process:

$$\mathrm{M} = (\mathrm{S},\ \mathrm{A},\ \mathrm{P},\ \mathrm{R})$$

Bellman equation:

$$V^{\pi}(s) = \sum_{a \in A} \pi(s,a) * [(R(s,a) + \gamma \sum_{s' \in S} P(s'|s,a) V^{\pi}(s'))]$$

Optimal Policy:

$$V^{*}(s) = \max_{a \in A}(R(s,a) + \gamma \sum_{s' \in S} P(s'|s,a) V^{*}(s'))$$

MDP - Transfer policy

TECHNISCHE UNIVERSITÄT DARMSTADT

$$M_s = (S_s, A, P_s, R_s)$$ (source)

$$M_t = (S_t, A, P_t, R_t)$$ (target)

Define mapping (does not have to be one-to-one):
$\rho : S_s \mapsto S_t$

$$\pi_s(s, a) = \pi_t(\rho(s), a), \text{ with } s \in S_s$$

But how good will this work? → Need a metric

[Phillips, 2006]

Definition – Bisimulation Relation

TECHNISCHE UNIVERSITÄT DARMSTADT

"[...] two states of a process are deemed equivalent if all the transitions of one state can be matched by transitions of the other state, and the results are themselves bisimilar."

i.e.:

$$s \sim s' \Leftrightarrow \forall a \in A.(R(s,a) = R(s',a) \\ \wedge \forall C \in S/\sim .P_s^a(C) = P_{s'}^a(C))$$

Where: $S/\sim$ is the state partition induced by $\sim$ and $P_s^a(C) = \sum_{c \in C} P(c|s,a)$

But: equivalence for stochastic processes is problematic since it requires the transition probabilities to agree exactly

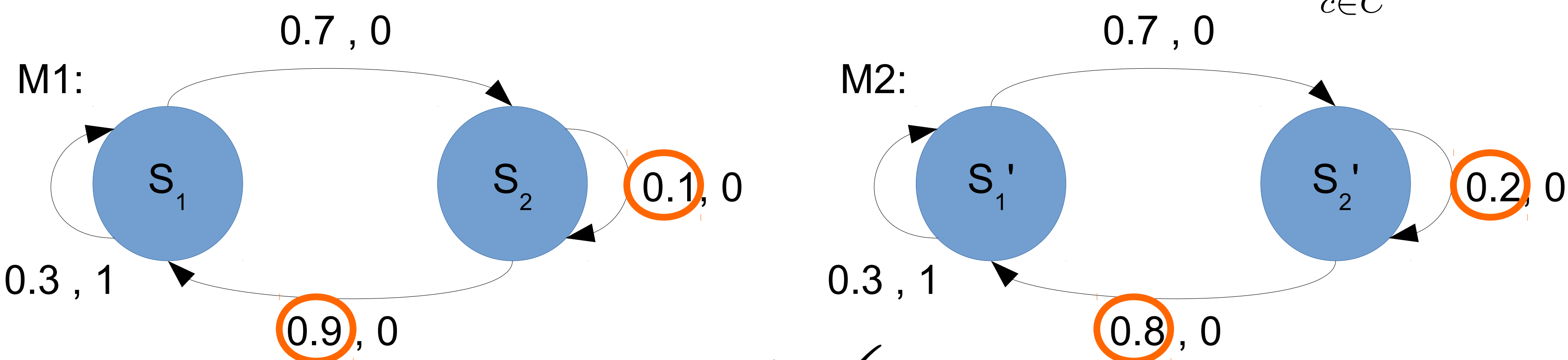
[Ferns, 2004]

Example – Bisimulation Relation

TECHNISCHE UNIVERSITÄT DARMSTADT

$$s \sim s' \Leftrightarrow \forall a \in A.(R(s,a) = R(s',a) \\ \wedge \forall C \in S/\sim .P_s^a(C) = P_{s'}^a(C))$$

Where: $S/\sim$ is the state partition induced by $\sim$ and $P_s^a(C) = \sum_{c \in C} P(c|s,a)$



$S_1 \sim S_1'$ ✓

$S_2 \not\sim S_2'$ ✗

Definition - Metric

1. $d(x, y) \geq 0$
2. $s = s' \Leftrightarrow d(s, s') = 0$
3. $d(s, s') = d(s' s)$
4. $d(s, s'') \leq d(s, s') + d(s', s'')$

State similarity metric

Bisimulation relation:

$$s \sim s' \Leftrightarrow \forall a \in A.(R(s,a) = R(s',a) \\ \wedge \forall C \in S/\sim .P_s^a(C) = P_{s'}^a(C))$$

- We need distance for reward and transition probabilities

$$d(s,s') = \max_{a \in A}(|R(s,a) - R(s',a)| \\ + \gamma T_K(d)(P(\cdot|s,a), Q(\cdot|s',a))$$

Discount factor

Kantorovich probability metric

[Ferns, 2004], [Phillips, 2009]

Definition – Kantorovich Metric $T_K(d)(P,Q)$

$$d(s,s') = \max_{a \in A}(|R(s,a) - R(s',a)| + \gamma T_K(d)(P(\cdot|s,a), Q(\cdot|s',a))$$

$$\max_{u_i, i=1\ldots|S|} \quad \sum_{i=1}^{|S|}(P(s_i) - Q(s_i))u_i$$

$$\text{subject to:} \quad \forall i,j.u_i - u_j \leq d(s_i, s_j)$$

$$\forall i.0 \leq u_i \leq 1$$

[Ferns, 2004]

Intuition: “[The metric] reflects the minimal amount of work that must be performed to transform one distribution into the other by moving “distribution mass” around.”
[Rubner, 1998]

a.k.a. “Earth mover's distance”

Similarity Calculation

- What we have:

  State distance measure

$$d(s,s') = \max_{a \in A}(|R(s,a) - R(s',a)| \\ + \gamma T_K(d)(P(\cdot|s,a), Q(\cdot|s',a))$$

- What we need:

  Measure for performance loss when transferring policy

MDP - Transfer policy

TECHNISCHE UNIVERSITÄT DARMSTADT

$M_s = (S_s, A, P_s, R_s)$ (source)

$M_t = (S_t, A, P_t, R_t)$ (target)

Mapping: $\rho : S_s \mapsto S_t$

$$\pi_s(s, a) = \pi_t(\rho(s), a), \text{ with } s \in S_s$$

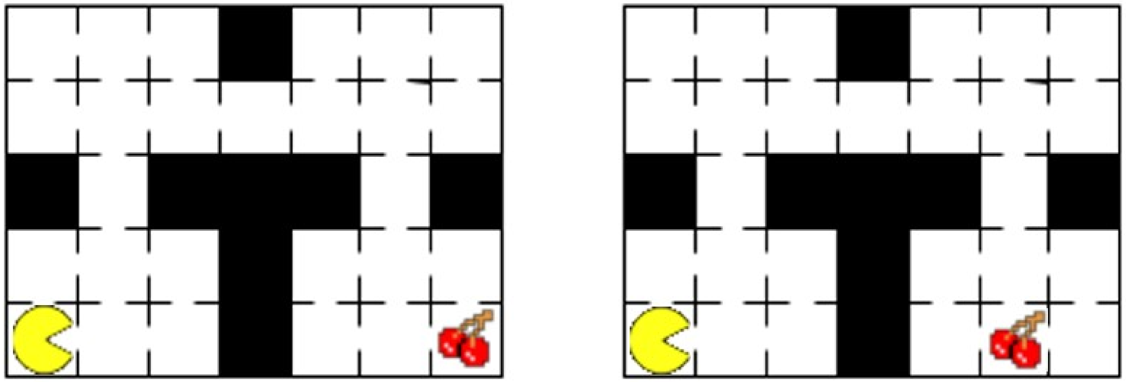
Now we can upper bound the performance loss by:

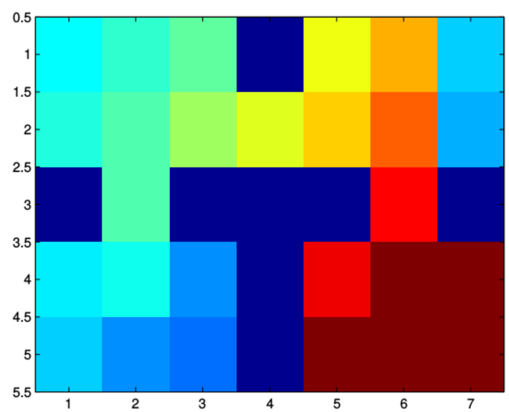
$$||V_t^{\pi_s} - V_t^{\pi_t^*}|| \leq \frac{2}{1-\gamma} \max_{s \in S_s} d(s, \rho(s)) + \frac{1+\gamma}{1-\gamma} ||V_s^{\pi_s} - V_s^{\pi_s^*}||$$

Proof: See [Phillips, 2006]

Note: Upper bound depends on quality of π_s and the mapping ρ

Example





Upper bound for

$$||V_t^{\pi_s} - V_t^{\pi_t^*}||$$

[Phillips, 2006]

Summary

- Goal: Transfer a policy from one MDP to a similar one
- Problems:
  - How to transfer?
  - How to measure the quality of the transfer?
- Solutions:
  - Transfer by mapping the states and induce the new poilcy
  - Use upper bound of performance loss as quality measure
- Conclusion:
  - This was just one special case of transfer learning
  - But: “[...] the problem of transfer in RL is far from being solved.” [Wiering, 2012]
  - Even in 2016 still an open problem (e.g. [Behbood, 2015], [Saito, 2016])

References

[Behbood, 2015] J. Lu, V. Behbood, P. Hao, H. Zuo, S. Xue, and G. Zhang, "Transfer learning using computational intelligence: A survey," Knowledge-Based Systems, vol. 80, pp. 14–23, May 2015.

[Ferns, 2004] N. Ferns, P. Panangaden, and D. Precup, "Metrics for Finite Markov Decision Processes," in Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence, Arlington, Virginia, United States, 2004, pp. 162–169.

[Ferns, 2012] N. Ferns, P. S. Castro, D. Precup, and P. Panangaden, "Methods for Computing State Similarity in Markov Decision Processes," Jun. 2012.

[Phillips, 2006] C. Phillips, J. Pineau, D. Precup, and P. Panangaden, "Knowledge Transfer in Markov Decision Processes (Technical Report)," 2006.

[Rubner, 1998] Y. Rubner, C. Tomasi, and L. J. Guibas, "A metric for distributions with applications to image databases," in Sixth International Conference on Computer Vision, 1998, 1998, pp. 59–66.

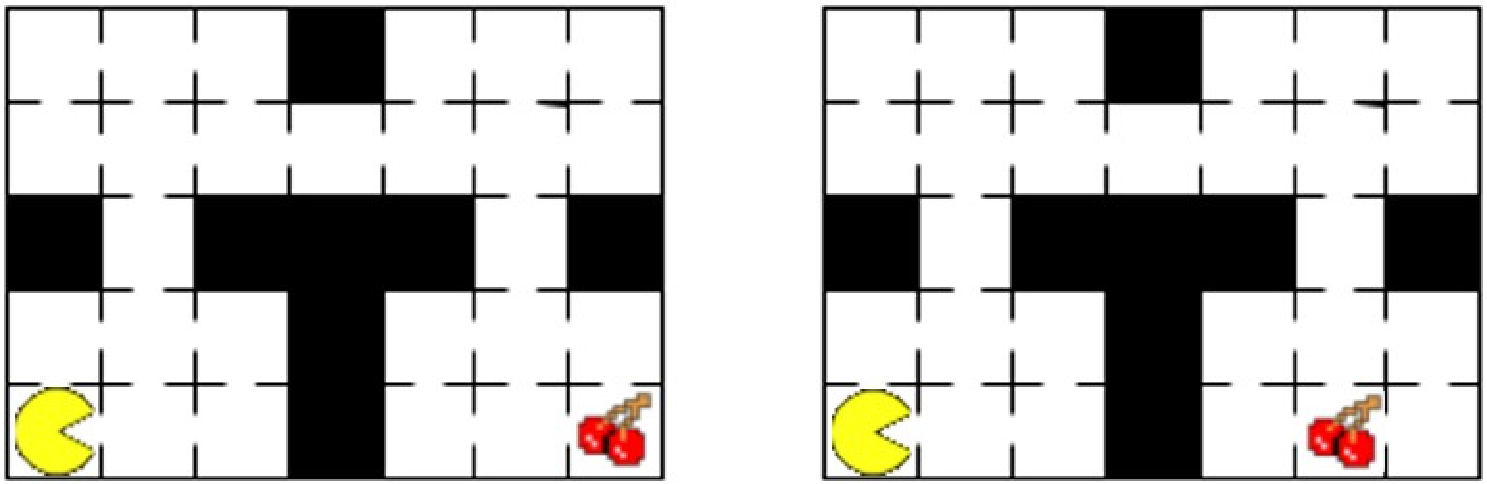
[Torrey, 2006] L. Torrey, J. Shavlik, T. Walker, and R. Maclin, "Skill Acquisition Via Transfer Learning and Advice Taking," in Machine Learning: ECML 2006, J. Fürnkranz, T. Scheffer, and M. Spiliopoulou, Eds. Springer Berlin Heidelberg, 2006, pp. 425–436.

[Wiering, 2012] M. Wiering and M. van Otterlo, Eds., Reinforcement Learning, vol. 12. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012.

[Saito, 2016] Saito, M., & Kobayashi, I., "A Study on Efficient Transfer Learning for Reinforcement Learning Using Sparse Coding," Journal of Automation and Control Engineering Vol, 4(4), 2016.

Thanks for your attention!

TECHNISCHE UNIVERSITÄT DARMSTADT



[Phillips, 2006]