Learning Multi-Step Predictive State Representations

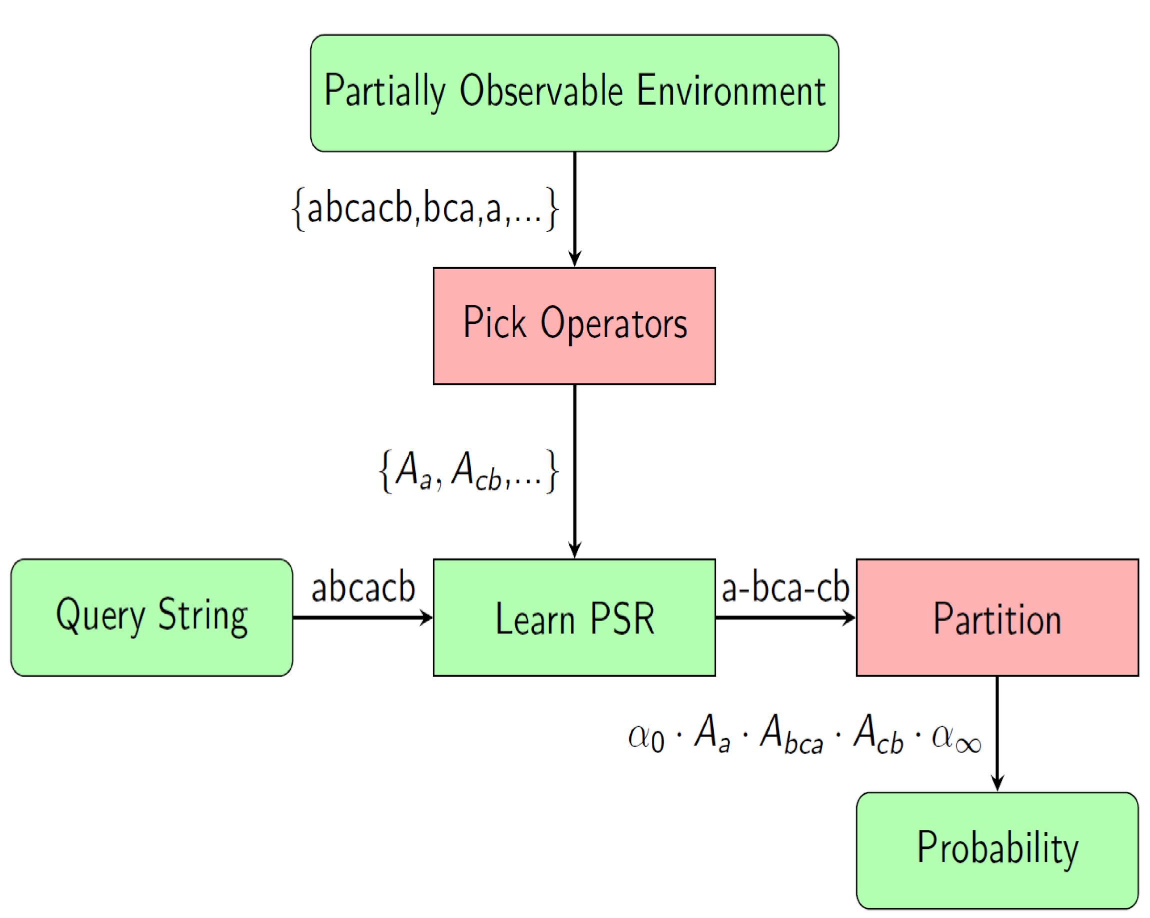
Lucas Langer, Borja Balle, Doina Precup
Reasoning and Learning Lab, McGill University

McGill

Abstract

With spectral learning algorithms, we consider the problem of making predictions in partially observable environments. Spectral algorithms learn a linear representation of a system's internal states. State representations can be used to make predictions of future observations conditioned on the past. Such predictions have been applied to robotics, where expectations of the future guide planning decisions (Pierre Luc). Spectral algorithms have the advantage that they are statistically consistent with well known error bounds (Citation). In this work, we develop a novel extension to a previous spectral based representation. We augment Predictive State Representations (PSRs) with multi-step transition operators to get Multi-Step Predictive State Representations (M-PSRs). In experiments, the novel representation far outperforms its predecessor when using regularized models.

High Level View



Multi-Step Predictive State Representations (M-PSRs)

As with traditional PSRs, M-PSRs are a linear predictive model which computes a function on strings. A traditional PSR is defined by the tuple: $\langle \Sigma, \boldsymbol{\alpha}_\lambda, \boldsymbol{\alpha}_\infty, \{\mathbf{A}_\sigma\}_{\sigma\in\Sigma}\rangle$. We defined an M-PSR by the tuple: $\langle \Sigma, \Sigma', \kappa, \boldsymbol{\alpha}_\lambda, \boldsymbol{\alpha}_\infty, \{\mathbf{A}_\sigma\}_{\sigma\in\Sigma'}\rangle$. Here, Σ is the set of observation symbols, α_λ is an initial weighting on states, α_∞ is a normalizing vector, and $\{\mathbf{A}_\sigma\}_{\sigma\in\Sigma'}$ is a set of transition operators. The additions in the M-PSR are $\Sigma' \subset \Sigma^*$ which is a set of important transition sequences and an encoding function κ. Combined, these two allow the M-PSR to express long transitions with a few operators. Intuitively, this should help prevent compounding errors and allow for faster queries.

Probability Queries with M-PSRs

To illustrate the difference between PSRs and M-PSRs, we will show probability queries using both representations. Let x be an observation sequence and σ_i being characters in Σ with $x = \sigma_1 \cdots \sigma_n$. For PSRs, we perform the following matrix product to compute the probability of x:

$$f_{\mathcal{A}}(x) = f_{\mathcal{A}}(\sigma_1 \cdots \sigma_n) = \boldsymbol{\alpha}_\lambda^\top \mathbf{A}_{\sigma_1} \cdots \mathbf{A}_{\sigma_n} \boldsymbol{\alpha}_\infty = \boldsymbol{\alpha}_\lambda^\top \mathbf{A}_x \boldsymbol{\alpha}_\infty \ .$$

With M-PSRs, we use the encoding $\kappa(x)$ to determine the matrices used in the product. The encoding is a splitting of the string x, $x = x_1 \cdots x_m$. Here, m is the length of the encoding and depends on x.

$$f_{\mathcal{A}}(x) = f_{\mathcal{A}}(x_1 \cdots x_m) = \boldsymbol{\alpha}_\lambda^\top \mathbf{A}_{x_1} \cdots \mathbf{A}_{x_m} \boldsymbol{\alpha}_\infty = \boldsymbol{\alpha}_\lambda^\top \mathbf{A}_x \boldsymbol{\alpha}_\infty \ .$$

Data-Driven M-PSRs

For Data-Driven M-PSRs we use iterative greedy algorithm for selecting Σ'. We take $\kappa(x)$ to be the encoding of minimal length using operators in Σ'. In the equations below, ∂ is the decoding function for κ, $train$ is the observation set, and sub_M are all sub-strings in $train$.

$$\kappa(x) = \underset{y \in \Sigma'^*,\ \partial(y) = x}{\operatorname{argmin}} |y|$$

$$\Sigma'_{i+1} = \underset{u \in \mathrm{sub}_M}{\operatorname{argmin}} \sum_{x \in \mathrm{train}} |\kappa_{\Sigma'_i \cup \{u\}}(x)|$$

Experiments

We conducted simulations of robot trajectories in labyrinth environments. Observations considered are time and wall colors. We compared the performances of PSRs with M-PSRs.

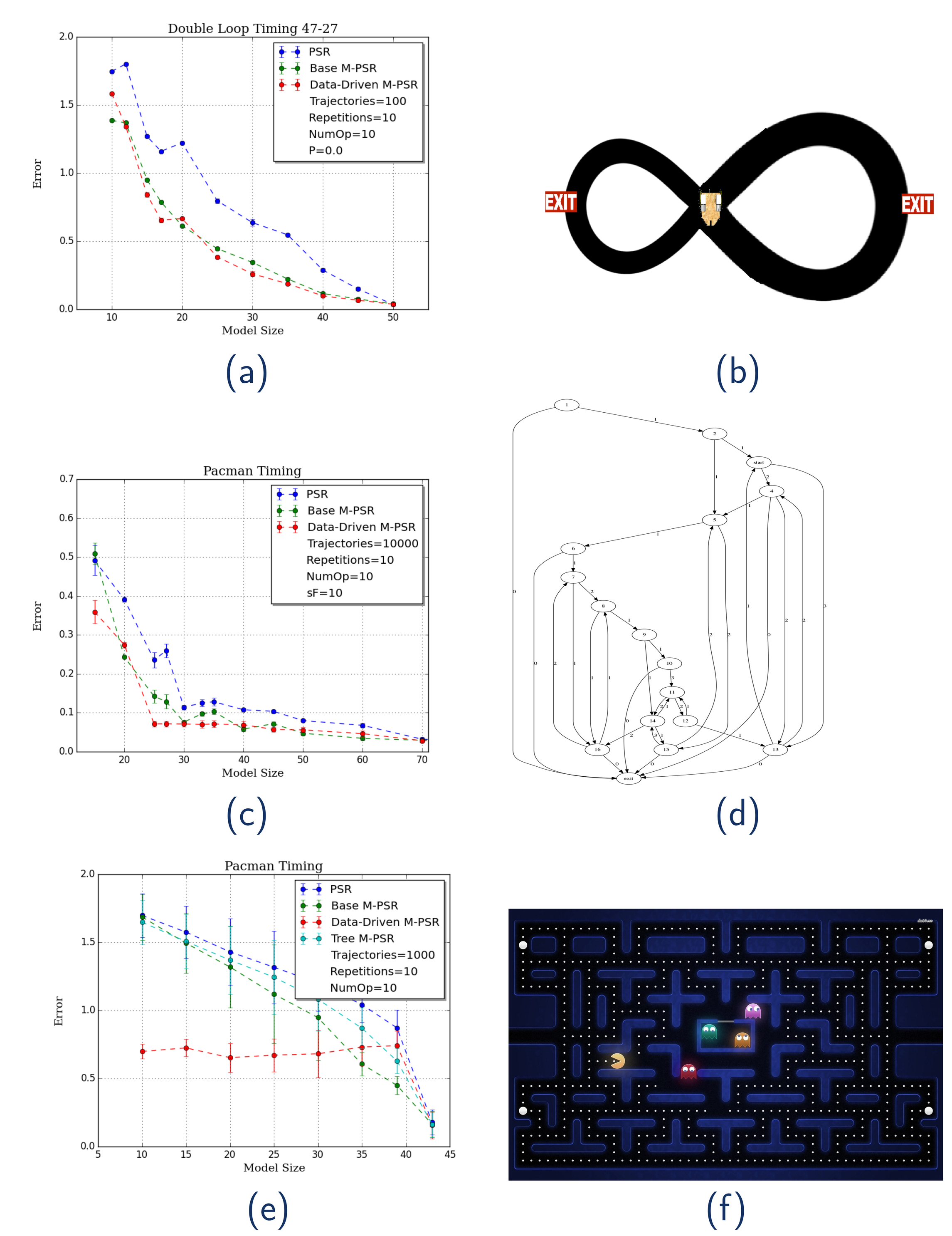


Figure : The plots compare the performance of the models as a function of the number of states represented. a) performance in double loops, c) performance in pacman, e) performance in colored loops. Figures b,d are images of the environments

Conclusion and Future Work

M-PSRs offer large improvements over traditional PSRs when using smaller model sizes, as is often desired in practice. These improvements offer various prospects for future work:

- Applications: planning, physiological signal recordings, and financial time-series
- Further optimization of the encoding function: κ, and selection of transition operators: Σ'
- Theoretical analysis of M-PSRs