ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8.
Available from http://www.i6doc.com/en/.

Discovering temporally extended features for reinforcement learning in domains with delayed causalities

Robert Lieck and Marc Toussaint

Universität Stuttgart – Machine Learning and Robotics Lab
Universitätsstraße 38 – 70569 Stuttgart – Germany

Abstract. Discovering temporally delayed causalities from data raises notoriously hard problems in reinforcement learning. In this paper we define a space of *temporally extended features*, designed to capture such causal structures, using a generating operation. Our discovery algorithm *PULSE* exploits the generating operation to efficiently discover a sparse subset of features. We provide convergence guarantees and apply our method to train a model-based as well as a model-free agent in different domains. In terms of achieved rewards and the number of required features our methods can achieve much better results than other feature expansion methods.

1 Introduction

Temporally delayed causalities are a natural aspect of real-world problems. If the latent variable of the causality is not observed this leads to a partially observable Markov decision process (POMDP), which for a reinforcement learning agent implies the fundamentally hard problem of discovering and representing these causalities based on interaction with the environment. For instance, whether a door is locked or not may depend on whether the robot turned the key some time ago, likewise, switching on the electric kettle produces hot water only at a later time. Existing approaches for solving POMDPs either define an abstract Markov state representation that subsumes information from past observations, such as *belief states* [1], *predictive state representations* [2], or *finite state controllers* [3], or work with history-based features, such as *context tree* methods [4, 5, 6]. In the important case of temporally delayed causalities history-based features have the advantage of being capable of explicitly representing the causal structure, which gives a better structural insight and allows a more intuitive integration of prior domain knowledge – one of the few means to improve autonomous artificial agents.

Contribution We propose a method for defining a feature space and an associated learning algorithm on a common basis by using a generating operation N^+ that spans a space of *temporally extended features* (TEFs) with increasing complexity and temporal extent, tailored to represent delayed causalities. Our learning algorithm *PULSE* makes use of N^+ to iteratively discover a sparse subset of TEFs. We provide convergence guarantees and use *PULSE* for solving POMDPs in model-based and model-free fashion showing that, in terms of achieved rewards as well as the number of required features, the agents trained with *PULSE* can achieve much better results than their competitors.

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8.
Available from http://www.i6doc.com/en/.

Content We will first establish the connection to related work on context trees, feature expansion, and L_1-regularized reinforcement learning. We will then introduce our method by defining *temporally extended features*, describing our discovery algorithm *PULSE* in detail, and discussing convergence properties and the richness of the generated representation. Finally, we discuss our empirical evaluations and possible future research.

2 Related work

As said above, only methods using history-based features are able to explicitly represent temporally delayed causalities. The most widely used approach for defining such features are *context tree* (CT) methods [4, 5, 6], which build a decision tree with each leaf node corresponding to a complex feature defined as the conjunction of basis features along the path to the root. We will compare our method with the *utility tree* (U-Tree) algorithm by McCallum [4] in its original model-free version as well as a proposed model-based variation, which we denote by *U-Tree (value)* and *U-Tree (model)*, respectively. For U-Tree (model) we replaced the Kolmogorov-Smirnov test by the chi-square test to measure the divergence of observation-reward distributions instead of value distributions.

By using a decision tree, CT methods are subject to three major constraints that we attempt to overcome with our method proposed in this paper: (1) CT methods require discrete valued basis features to build the decision tree while our method uses a linear combination of features thus allowing for any scalar valued features. (2) Features in a decision tree are mutually exclusive, that is, at any time only one of them can be active. In contrast, a linear combination of $|\mathcal{F}|$ binary features can take the exponentially larger number of $2^{|\mathcal{F}|}$ different values. (3) Restructuring a decision tree (as needed for transfer learning tasks when part of the tree should be reused to represent different data) is a non-trivial task while in our learning algorithm *PULSE* restructuring the feature set by growing and shrinking it is a natural part of the learning process.

Our method combines two other branches of research: that of feature expansion techniques and that of L_1-regularization applied to temporal difference (TD) learning. Feature expansion techniques were successfully applied to learn (conditional) random field models for text [7, 8] and, in reinforcement learning, for linear approximations of the value or transition function [9, 10]. In all cases a monotonically growing feature set is learned by scoring and including conjunctions of basis features. In contrast, the approaches for L_1-regularized TD learning [11, 12] start with a large feature set that is monotonically shrunk during the learning process.

Our discovery algorithm *PULSE* combines these two basic ideas by both growing *and* shrinking the feature set in each iteration. *PULSE* can thus be regarded as a generalization of these approaches. Furthermore, the generating operation N^+ and the objective function can be chosen freely, which makes *PULSE* very flexible and applicable to a wide range of problems.

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8.
Available from http://www.i6doc.com/en/.

3 Discovering temporally extended features

Temporally extended features The set $\mathcal{T}$ of *temporally extended features* (TEFs) is the set of all maps from histories $(\mathcal{A} \times \mathcal{O} \times \mathcal{R})^*$, that is, sequences of action-observation-reward triplets, to the real numbers $\mathbb{R}$. In practice we will always work with a subset of $\mathcal{T}$ that is generated by an operation $N^+ : \mathcal{P}(\mathcal{T}) \to \mathcal{P}(\mathcal{T})$ where $\mathcal{P}(\mathcal{T})$ is the power set of $\mathcal{T}$. This means, for a given set $\mathcal{F}$ of features $N^+(\mathcal{F})$ is a set of *candidate* features and we work with the smallest subset $\mathcal{T}_{N^+} \subseteq \mathcal{T}$ that is closed under N^+. Our discovery algorithm *PULSE* will in each iteration make use of the candidate features to explore the feature set $\mathcal{T}_{N^+}$.

The specific choice of N^+ depends on the problem and learning method used with *PULSE*. To represent temporally delayed causalities we use a set of basis features $\mathcal{B}$ consisting of indicator features for all actions, observations, and rewards at all different times in the past and define N^+ to generate candidate features by taking all possible conjunctions of an existing feature with one of the basis features. To introduce a bias towards the near past we modify N^+ to only use basis features that go one step further into the past than the existing features and not farther than a fixed time horizon $t_{min} = -k$. Also to initiate learning from an empty feature set the candidate features will in that case be all basis features with time index $t = 0$.

Note that this specific definition of N^+ is at the same time a restriction of the general definition of TEFs and (for this very reason) a means to tame their complexity and tailor the used subset $\mathcal{T}_{N^+}$ to our needs. Our learning algorithm *PULSE* presented below does not rely on this definition and can equally well be used with any other definition of N^+.

The *PULSE* algorithm *PULSE* stands for ***P****eriodical* ***U****nconvering of* ***L****ocal* ***S****tructure* ***E****xtensions*. The idea is to exploit the structure that N^+ induces on the feature set $\mathcal{T}_{N^+}$ to discover local extensions of the current feature set. By using an L_1-regularization within the objective function superfluous features are eliminated. Repeating this procedure results in a pulsating dynamic of the feature set, guided by N^+ and the objective, driving the feature set to an optimum.

The *PULSE* algorithm is detailed in Alg. 1. The main loop consists of (1) growing the feature set using N^+ and assigning zero weight to any new features, (2) optimizing the objective function $\mathcal{O}$ with respect to the feature weights Θ given the current feature set $\mathcal{F}$ and data D, and (3) shrinking $\mathcal{F}$ by eliminating any zero-weight features. The crucial part is the optimization step (line 6) where the feature weights are optimized. In order for *PULSE* to work properly, the objective function $\mathcal{O}$ should fulfill two properties: (1) $\operatorname{argmin}_\Theta \mathcal{O}(\mathcal{F}, \Theta, D)$ should be sparse, that is, after optimizing $\mathcal{O}$ many Θ_f should be zero. (2) Features with zero weight should not affect the objective value, which ensures monotone convergence throughout the growth and shrinkage operations. $\mathcal{O}$ can otherwise be chosen freely.

Convergence guarantees *PULSE* is guaranteed to converge to a locally optimal feature set with globally optimal weights (provided $\mathcal{O}$ is convex and fulfills condition (2) above). For the important case of predicting an event that depends

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8.
Available from http://www.i6doc.com/en/.

Algorithm 1 The *PULSE* algorithm

Input: $N^+, \mathcal{O}, D$
Output: $\mathcal{F}$, Θ
Initialize: $\mathcal{F} \leftarrow \emptyset$, $\Theta \leftarrow \emptyset$
repeat
  grow_feature_set$(\mathcal{F}, \Theta, N^+)$
  $\Theta \leftarrow \text{argmin}_\Theta\ \mathcal{O}(\mathcal{F}, \Theta, D)$
  shrink_feature_set$(\mathcal{F}, \Theta)$
until $\mathcal{O}, \mathcal{F}, \Theta$ do not change

grow_feature_set$(\mathcal{F}, \Theta, N^+)$
  Initialize: $\mathcal{F}^+ \leftarrow N^+(\mathcal{F})$
  for all $f \in \mathcal{F}^+$
    if $f \notin \mathcal{F}$ **then** $\Theta_f \leftarrow 0$ **endif**
  end for
  $\mathcal{F} \leftarrow \mathcal{F} \cup \mathcal{F}^+$

shrink_feature_set$(\mathcal{F}, \Theta)$
  for all $f \in \mathcal{F}$
    if Θ_f is 0 **then** $\mathcal{F} \leftarrow \mathcal{F} \setminus f$ **endif**
  end for

on n each necessary and jointly sufficient conditions[1] *PULSE* is even guaranteed to converge to a globally optimal feature set with N^+ using only pairwise conjunctions. To see this, note that a predictor using m-fold conjunctions ($m < n$) will monotonically improve as $m \to n$ because each marginalized condition impairs the prediction. When greedy expansion does not guarantee global convergence simulated annealing approaches, where sub-optimal features are included with a certain probability, still guarantee *asymptotic* convergence to the global optimum.

Richness of the representation If N^+ is defined as above with a limit in the time horizon the resulting set of TEFs has the same descriptive power as a tabular k-MDP representations. In general, *PULSE* can be used with any N^+ that produces finite extensions $N^+(\mathcal{F})$. This provides a powerful framework, which can also be extended in various ways including the use of continuously parameterized features, products of basis functions for continuous domains, or finite state machines as basis features. Which choice of N^+ could generate, for instance, a set of TEFs equivalent to k-order predictive state representations is an interesting non-trivial question.

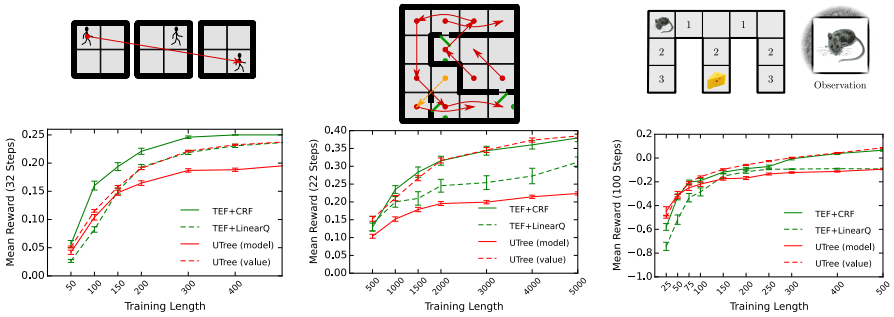
Model-based and model-free agents for experiments Our two agents trained with *PULSE*, referred to as *TEF+CRF* and *TEF+LinearQ*, use a conditional random field (CRF) [13] and a linear approximation of the state-action value function (Q-function), respectively

$$\begin{array}{cc} \text{TEF+CRF (model-based)} & \text{TEF+Linear } Q \text{ (model-free)} \\ p(\bar{o}, \bar{r} | \bar{h}, \bar{a}) = \frac{\exp \sum_{f \in F} \theta_f f(h)}{\sum_{\bar{o}, \bar{r}} \exp \sum_{f \in F} \theta_f f(h)} & Q(\bar{a}, \bar{h}) = \sum_{f \in \mathcal{F}} \theta_f f(h) \end{array} ,$$

where $(\bar{a}, \bar{o}, \bar{r})$ is the last action-observation-reward triplet in history h and $\bar{h}$ is the remaining part of h. TEF+CRF is trained by minimizing the neg-log-likelihood of the data via gradient descent using L-BFGS [14]. For TEF+Linear Q we use least-squares policy iteration [15]. Both use an additional L_1-regularization of variable strength.

[1] Such as "putting the key in the lock *and* turning it *and* pushing the handle *and* pulling" in order to open a door.

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8.
Available from http://www.i6doc.com/en/.



k-MDP states:	1 600	4 096 000	∞
latent states:	12	34 012 224	11

Fig. 1: Mazes and learning curves with size of k-MDP and latent state space.

4 Experiments

We performed evaluations in three different deterministic partially observable maze environments that we believe exhibit a prototypical structure for temporally delayed causalities. The 2×2 and 4×4-maze (left and center in Fig. 1) contain delayed rewards that are "activated" at one location and later "collected" at different one. The delay is $\Delta t = 2$ (red solid arrows) or $\Delta t = 3$ (orange dashed arrow). Additionally, the 4×4-maze contains doors that open for two time steps when the agent operates the nearby switch (semicircle) by stepping into the wall. The time horizon for these mazes was $t_{min} = -2$ and $t_{min} = -3$, respectively. The *Cheese Maze* maze (right in Fig. 1), where the agent only perceives adjacent walls, was introduced in [4]. The time horizon was $t_{min} = -2$.

For each maze we performed a number of trials with a training phase of varying length (using random policy) and an evaluation phase of fixed length (using the agent's optimal policy). The plots in Fig. 1 show the mean reward during evaluation with the standard error of the mean as error bars. We used forward tree search for planning with the model-based methods.

Results In all three environments our TEF+CRF method using *PULSE* outperforms U-Tree (model) by a large margin suggesting it to be the preferable method in a model-based setting. TEF+Linear Q, on the other hand, performs only equal or inferior to U-Tree (value) in the model-free setting. This suggests that focusing only on rewards makes it difficult to discover the relevant features. Note, however, that the model-based TEF+CRF even keeps up with the model-free U-Tree (value) while at the same time solving the significantly more complex task of learning a complete predictive model. Also, we observed *PULSE* to generally use fewer features of lower order (fewer conjunctions) than U-Tree and to learn more compact models for longer training lengths.

5 Conclusion and outlook

We considered the problem of uncovering temporally delayed causalities in partially observable reinforcement learning domains. To this end we introduced *tem-*

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8. Available from http://www.i6doc.com/en/.

porally extended features (TEFs) along with a training method called *PULSE* that efficiently and incrementally discovers a sparse set of relevant TEFs. We provided convergence guarantees and evaluated our approach empirically showing that in terms of achieved rewards as well as the number of required features *PULSE* clearly outperforms its competitors in a model-based setting. While in this paper we considered very simple basis features, we discussed how the general framework provided by *PULSE* can be extended to learn much richer representations.

References

[1] William S Lovejoy. A survey of algorithmic methods for partially observed markov decision processes. *Annals of Operations Research*, 28(1):47–65, 1991.

[2] Michael L. Littman, Richard S. Sutton, and Satinder Singh. Predictive Representations of State. In *In Advances In Neural Information Processing Systems 14*, pages 1555–1561. MIT Press, 2002.

[3] Eric A Hansen. An improved policy iteration algorithm for partially observable mdps. *Advances in Neural Information Processing Systems*, pages 1015–1021, 1998.

[4] Andrew Kachites McCallum. *Reinforcement Learning with Selective Perception and Hidden State*. PhD thesis, Computer Science Department, University of Rochester, 1996.

[5] Joel Veness, Kee Siong Ng, Marcus Hutter, and David Silver. Reinforcement learning via aixi approximation. In *AAAI*, 2010.

[6] Phuong Nguyen, Peter Sunehag, and Marcus Hutter. Context tree maximizing reinforcement learning. In *Proc. of the 26th AAAI Conference on Artificial Intelligence*, pages 1075–1082, 2012.

[7] S. Della Pietra, V. Della Pietra, and J. Lafferty. Inducing features of random fields. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(4):380 –393, 1997.

[8] A. McCallum. Efficiently inducing features of conditional random fields. In *Proceedings of the Nineteenth Conference on Uncertainty in Artificial Intelligence*, pages 403–410. Morgan Kaufmann Publishers Inc., 2002.

[9] Alborz Geramifard, Finale Doshi, Joshua Redding, Nicholas Roy, and Jonathan How. Online Discovery of Feature Dependencies . In Lise Getoor and Tobias Scheffer, editors, *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, ICML '11, pages 881–888, New York, NY, USA, June 2011. ACM.

[10] N Kemal Ure, Alborz Geramifard, Girish Chowdhary, and Jonathan P How. Adaptive planning for markov decision processes with uncertain transition models via incremental feature dependency discovery. In *Machine Learning and Knowledge Discovery in Databases*, pages 99–115. Springer, 2012.

[11] J Zico Kolter and Andrew Y Ng. Regularization and feature selection in least-squares temporal difference learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 521–528. ACM, 2009.

[12] Mayank Daswani, Peter Sunehag, and Marcus Hutter. Q-learning for history-based reinforcement learning. In *Asian Conference on Machine Learning*, pages 213–228, 2013.

[13] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

[14] Richard H Byrd, Jorge Nocedal, and Robert B Schnabel. Representations of quasi-newton matrices and their use in limited memory methods. *Mathematical Programming*, 63(1-3):129–156, 1994.

[15] Michail G. Lagoudakis and Ronald Parr. Least-squares policy iteration. *J. Mach. Learn. Res.*, 4:1107–1149, December 2003.