# Biorthogonalization Techniques for Least Squares Temporal Difference Learning

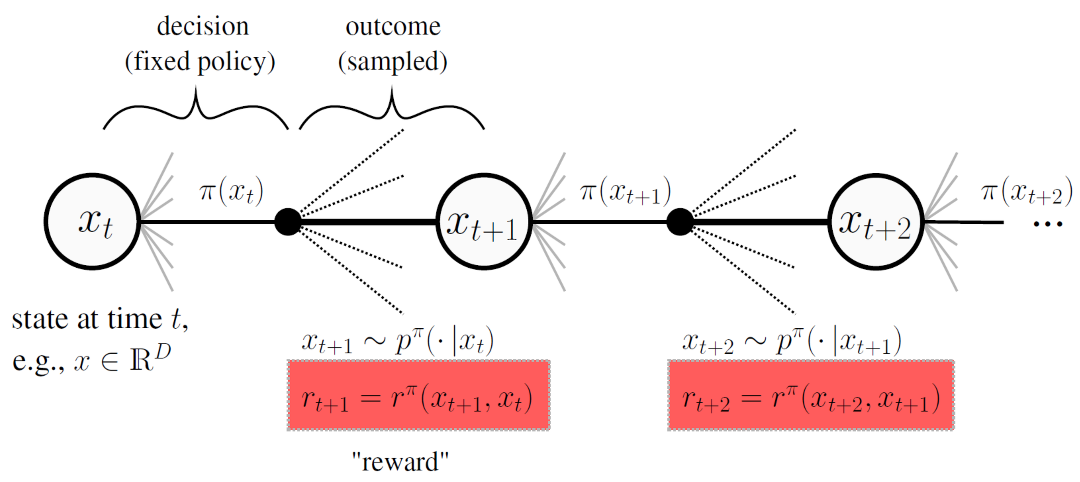
**Tobias Jung**
Montefiore Institute, University of Liège, Belgium
email: tjung@ulg.ac.be

## Abstract

We consider Markov reward processes and study **OLS-LSTD**, a framework for selecting basis functions from a set of candidates to obtain a sparse representation of the value function in the context of least squares temporal difference learning. To support efficient both updating and downdating operations, OLS-LSTD uses a biorthogonal representation for the selected basis vectors. Empirical comparisons with the recently proposed MP and LARS frameworks for LSTD are made.

## Introduction

### Markov reward processes




A Markov reward process over state space $X$ (e.g., $X \subset \mathbb{R}^D$) is specified through the two components

- $p^\pi(\cdot|x_t)$ (transitions)
- $r^\pi(x_{t+1}, x_t)$ (reward)

Assume in the following that we don't know $p^\pi, r^\pi$.

### The problem

We are given a history of observations obtained from executing $\pi$ over a sequence of $N$ steps:

$$\boxed{\{x_0, r_1, x_1, \ldots, x_{N-1}, r_N, x_N\} \quad \text{("training" data)}} \qquad (1)$$

where $x_i \sim p^\pi(\cdot|x_{i-1})$ and $r_i = r^\pi(x_i, x_{i-1})$, $i = 2, \ldots, N$. Our goal is to be able to estimate how "good" *any* given state $x \in X$ is, provided that the system will always develop according to $\pi$. The "goodness" of a state is measured in terms of the expected accumulated reward that we will receive from it. More precisely, we want to estimate the quantity (infinite horizon discounted sum of expected rewards) $V^\pi(x) = \mathbb{E}\{\sum_{t=0}^\infty \gamma^t r^\pi(x_{t+1}, x_t) \,|\, x_0 = x\}$, where $\gamma \in (0,1)$ is a discount factor, and which obeys the linear fixed point equation ($\forall x$)

$$V^\pi(x) = \mathbb{E}_{x' \sim p^\pi(\cdot|x)} \left\{ r^\pi(x', x) + \gamma V^\pi(x') \right\}. \qquad (2)$$

Since we do not know $p^\pi$ and $r^\pi$ (and even if we did, the state space is typically too large to solve the equation exactly), we have to somehow use the training data to infer a function $\tilde{V}(\cdot; w)$, parameterized by $w$, such that $\tilde{V}(x; w)$ is close to $V^\pi(x)$.

### Where does it arise?

Estimating the value function $V^\pi$ (from samples), which is also known as policy evaluation, arises as the fundamental computational substep in policy iteration for determining optimal control policies $\pi^*$ in the context of reinforcement learning / approximate dynamic programming. Furthermore, there are some applications where knowing $V^\pi$ by itself is of interest, e.g., intrusion detection or marketing.

### Solution via LSTD

Assume a linearly parameterized $\tilde{V} : X \times \mathbb{R}^k \to \mathbb{R}$

$$\tilde{V}(\cdot\,; w) = \sum_{i=1}^k w_i \phi_i(\cdot)$$

with coefficients $w_i \in \mathbb{R}$ and where $\pi_i : X \to \mathbb{R}$ is a basis function or "feature" that extracts a certain property from the state. Now there are two questions:

1. How to choose $w_i$, given the $\phi_i$?
2. How to choose the $\phi_i$?

The answer to the first is simple. Given $\phi_1, \ldots, \phi_k$, we form from the observed sample transitions (1) the two $N \times k$ matrices

$$\tilde{\Phi}_k = [\alpha_1 | \cdots | \alpha_k], \text{ where } \alpha_i = \begin{bmatrix} \phi_i(x_0) \\ \vdots \\ \phi_i(x_{N-1}) \end{bmatrix}$$

$$\tilde{\Phi}'_k = [\alpha'_1 | \cdots | \alpha'_k], \text{ where } \alpha'_i = \begin{bmatrix} \phi_i(x_1) \\ \vdots \\ \phi_i(x_N) \end{bmatrix}$$

and the $N \times 1$ vector $\tilde{R}$ with entries $[\tilde{R}]_i = r^\pi(x_i, x_{i-1})$. Let $\mathcal{V}_k = \operatorname{span}\{\alpha_1, \ldots, \alpha_k\}$ and let $P_{\mathcal{V}_k}$ denote the orthogonal projection onto $\mathcal{V}_k$. In **least squares temporal difference learning (LSTD)**, the coefficients $w$ are obtained by solving the following equation

$$\boxed{\tilde{\Phi}_k w = P_{\mathcal{V}_k}(\tilde{R} + \gamma \tilde{\Phi}'_k w),} \qquad (3)$$

the solution of which "converges" (with growing number of samples $N$) to the solution of a projected version of the original fixed point equation (2) weighted by the stationary state distribution (Bertsekas, 2007).

### Feature generation or feature selection?

The second question raises a conceptual issue and is more difficult to answer. The problem one faces is one of *feature selection* (i.e., given a large set of candidate $\phi_i$, which ones to use in the representation of $\tilde{V}$) and *feature generation* (i.e., where do the candidate $\phi_i$'s come from). Our aim with this contribution will be to only address the (easier) problem of feature selection.

## Feature selection in LSTD via OLS

Assume that we are given a large set of basis function *candidates* $\{\phi_1, \ldots, \phi_M\}$ together with our training samples (1). From the basis functions and samples we can compute the corresponding basis vectors $\{\alpha_i\}_{i=1}^M$, $\{\alpha'_i\}_{i=1}^M$ as defined above; note that only the $\alpha_i$ will be used to represent $\tilde{V}$. The framework we propose works incrementally by combining *forward selection* with *backward deletion* steps. At each step $k$ of the procedure, we maintain a list of currently selected basis vectors and will

- either **add** from the unselected basis vectors the one which contributes the most (reduces the most the norm of the Bellman residual wrt the current LSTD solution),
- or **delete** from the selected basis vectors the one which contributes the least,
- or do a combination of both

### Basic forward selection

Assume that at step $k$, $\mathcal{V}_k = \operatorname{span}\{\alpha_i\}_{i=1}^k$, i.e., that the indices are ordered such that the first $k$ correspond to the selected basis vectors. Let $\mathcal{V}_{k\oplus i} = \mathcal{V}_k \oplus \langle \alpha_i \rangle$, $i = k+1, \ldots, M$, be the $(k+1)$-dimensional space if unselected basis vector $\alpha_i$ is selected next, and let $P_{\mathcal{V}^\perp_{k\oplus i}} = I - P_{\mathcal{V}_{k\oplus i}}$ be the projection onto the orthogonal complement space (in $\mathbb{R}^N$). Each step of the forward selection now performs the following operations:

1. Find index $i^* \in \{k+1, \ldots, M\}$ which maximizes reduction of the Bellman residual in the current LSTD solution $w_k$ for $\mathcal{V}_k$:
$$i^* = \operatorname*{argmin}_{i=k+1,\ldots,M} err_{k\oplus i}, \qquad \text{where } err_{k\oplus i} = \|P_{\mathcal{V}^\perp_{k\oplus i}}(\tilde{R} + \gamma \tilde{\Phi}'_k w_k)\|^2$$
2. Add $\alpha_* = \alpha_{i^*}$ to the list of selected basis vectors. Set $\mathcal{V}_{k+1} = \operatorname{span}\{\alpha_1, \ldots, \alpha_k, \alpha_*\}$, append a column to $\tilde{\Phi}_{k+1} = [\tilde{\Phi}_k \,|\, \alpha_*]$, $\tilde{\Phi}'_{k+1} = [\tilde{\Phi}'_k \,|\, \alpha'_*]$ and swap elements such that index $k+1$ corresponds to index $i^*$.
3. Compute $w_{k+1}$ as LSTD solution for $\mathcal{V}_{k+1}$, i.e., solve $\tilde{\Phi}_{k+1} w = P_{\mathcal{V}_{k+1}}(\tilde{R} + \gamma \tilde{\Phi}'_{k+1} w)$.

### Orthogonal representation of unselected elements

To efficiently determine the novel contribution for each unselected basis vector in *Step 1*, we store $\psi_i^{(k)} = P_{\mathcal{V}^\perp_k} \alpha_i$, $i = k+1, \ldots, M$. The next best element to add is then simply

$$\textbf{Add: } i^* = \operatorname*{argmax}_{i=k+1,\ldots,M} |\langle \psi_i^{(k)}, \tilde{R} + \gamma \tilde{\Phi}'_k w_k \rangle| / \|\psi_i^{(k)}\|.$$

Whenever an unselected basis vector $\alpha_*$ is selected in *Step 2*, the remaining $\psi_i^{(k)}$ need to be reorthogonalized with respect to the new $\mathcal{V}_{k+1} = \mathcal{V}_k \oplus \langle \alpha_* \rangle$.

### Biorthogonal representation of selected elements

Each selected basis vector $\alpha_i$ spanning $\mathcal{V}_k$ is associated with a biorthogonal basis vector $\beta_i^{(k)}$ with the property $\langle \beta_i^{(k)}, \alpha_j \rangle = \delta_{ij}$ for $j = 1, \ldots, k$. The $\beta_i^{(k)}$ span the same space $\mathcal{V}_k$ and are chosen such they represent the projection onto $\mathcal{V}_k$ in terms of the original (non-orthogonalized) basis vectors $\alpha_i$; i.e.,

$$P_{\mathcal{V}_k} z = \sum_{i=1}^k \langle \beta_i^{(k)}, z \rangle \alpha_i.$$

With a biorthogonal basis representation, $P_{\mathcal{V}_k}$ can be easily updated *in both directions* $P_{\mathcal{V}_{k\oplus i}}$ (adding an element) and $P_{\mathcal{V}_{k\ominus i}}$ (deleting an element) [3].

**Add:** Initially, set $\beta_1^{(1)} = \alpha_1 / \|\alpha_1\|$. Then, whenever in step $k$ unselected basis vector $\alpha_*$ gets selected, the current $\beta_i^{(k)}$ are modified as follows:

$$\beta_* = \psi_*^{(k)} / \|\psi_*^{(k)}\|^2, \qquad \beta_i^{(k+1)} = \beta_i^{(k)} - \beta_* \langle \beta_i^{(k)}, \alpha_* \rangle / \|\alpha_*\|$$

for $i = 1, \ldots, k$ and appending $\beta_{k+1}^{(k+1)} = \beta_*$.

**Del:** To decide whether we want to remove an element from the currently selected ones, we find the one with the minimum contribution

$$\textbf{Del: } i^* = \operatorname*{argmin}_{i=1,\ldots,k} |\langle \beta_i^{(k)}, \tilde{R} + \gamma \tilde{\Phi}'_k w_k \rangle| / \|\beta_i^{(k)}\|.$$

Whenever an element $j$ is deleted, we downdate the projection $P_{\mathcal{V}_{k\ominus j}}$ by setting:

$$\beta_* = \beta_j^{(k)} / \|\beta_j^{(k)}\|, \qquad \beta_i^{(k-j)} = \beta_i^{(k)} - \beta_* \langle \beta_*, \beta_i^{(k)} \rangle.$$

## Empirical comparison with LARS and MP

We examine variants of OLS-LSTD in the benchmark problem *mountain car* (nonlinear optimal control, deterministic, 2-dimensional state space), following the optimal policy $\pi^*$ during sample generation (starting from random initial states) and thus trying to estimate $V^*$.

### Methods compared

- **OLS-F($k_{max}$):** select $k_{max}$ basis vectors via forward selection.
- **OLS-FB($k_{max}$):** select $2 \times k_{max}$ basis vectors via forward selection, then remove $k_{max}$ via backward deletion.
- **OLS-FB2($\lambda$):** add $\ell 0$ regularization to (3). At each step, either add or delete a basis vector until no further improvement is possible.
- **MP-F($k_{max}$):** implemented as described in [1].
- **LARS($\beta$):** implemented as described in [2].

### Mountain car

Stats: 2504 training samples, 7513 test samples, 1365 basis function candidates (RBFs on a grid at various levels of coarseness).

LARS-LSTD

| $\beta$ = | Error (abs) | nBasis |
|---|---|---|
| 50 | 5.22 | 25 |
| 10 | 2.86 | 57 |
| 1 | 1.78 | 138 |
| 0.1 | 1.58 | 321 |

OLS-FB2-LSTD

| $\lambda$ = | Error (abs) | nBasis |
|---|---|---|
| 20 | 14.61 | 19 |
| 5 | 2.82 | 53 |
| 1 | 1.79 | 126 |
| 0.1 | 1.42 | 290 |

MP-LSTD

| nBasis | Error(abs) |
|---|---|
| 25 | 3.21 |
| 50 | 2.80 |
| 100 | 1.75 |
| 200 | 1.63 |
| 300 | 1.60 |

OLS-F-LSTD

| nBasis | Error (abs) |
|---|---|
| 25 | 7.75 |
| 50 | 2.83 |
| 100 | 1.96 |
| 200 | 1.57 |
| 300 | 1.37 |

OLS-FB-LSTD

| nBasis | Error (abs) |
|---|---|
| 25 | 2.89 |
| 50 | 1.94 |
| 100 | 1.91 |
| 200 | 1.50 |
| 300 | 1.34 |

## Related work

[1] C. Painter-Wakefield and R. Parr. Greedy algorithms for sparse reinforcement learning. In: *Proc. of ICML*, 2012
[2] J. Zico Kolter and Andrew Y. Ng. Regularization and feature selection in least-squares temporal difference learning. In: *Proc. of ICML*, 2009
[3] M. Andrle, L. Rebollo-Neira, and E. Sagianos. Backward-optimized orthogonal matching pursuit approach. *IEEE Signal Processing Letters*, 11(9):705-708, 2004