An Analysis of Laplacian Methods for Value Function Approximation in MDPs

Marek Petrik
Department of Computer Science
University of Massachusetts
Amherst, MA 01003
petrik@cs.umass.edu

Abstract

Recently, a method based on Laplacian eigenfunctions was proposed to automatically construct a basis for value function approximation in MDPs. We show that its success may be explained by drawing a connection between the spectrum of the Laplacian and the value function of the MDP. This explanation helps us to identify more precisely the conditions that this method requires to achieve good performance. Based on this, we propose a modification of the Laplacian method for which we derive an analytical bound on the approximation error. Further, we show that the method is related the augmented Krylov methods, commonly used to solve sparse linear systems. Finally, we empirically demonstrate that in basis construction the augmented Krylov methods may significantly outperform the Laplacian methods in terms of both speed and quality.

1 Introduction

Markov Decision Processes (MDP) [Puterman, 2005] are a widely-used framework for planning under uncertainty. In this paper, we focus on the discounted infinite horizon problem with discount γ such that $0 < \gamma < 1$. We also assume finite state and action spaces. While solving this problem requires only polynomial time, many practical problems are too large to be solved precisely. This motivated the development of approximate methods for solving very large MDPs that are sparse and structured.

Value Function Approximation (VFA) is a method for finding approximate solutions for MDPs, which has received a lot of attention [Bertsekas and Tsitsiklis, 1996]. Linear approximation is a popular VFA method, because it is simple to analyze and use. The representation of the value function in linear schemes is a linear combination of basis vectors. The optimal policy is usually calculated using *approximate value iteration* or *approximate linear programming* [Bertsekas and Tsitsiklis, 1996].

The choice of the basis plays an important role in solving the problem. Usually, the basis used to represent the space is hand-crafted using human insight about some topological properties of the problem [Sutton and Barto, 1998]. Recently, a new framework for automatically constructing the basis was proposed in [Mahadevan, 2005]. This approach is based on analysis of the neighborhood relation among the states. The analysis is inspired by spectral methods, often used in machine-learning tasks. We describe this framework in more detail in Section 2.

In the following, we use v to denote the value function, and r to denote the reward vector. The matrix I denotes an identity matrix of an appropriate size. We use n to denote the number of states of the MDP.

An important property of many VFA methods is that they are guaranteed to converge to an approximately optimal solution. In methods based on *approximate policy iteration* the maximal distance of the *optimal* approximate value function $\hat{v}$ from the optimal value function v^* is bounded by [Munos, 2003]:

$$\limsup_{k\to\infty} \|v^* - \hat{v}\|_\mu \leq \frac{2\gamma}{(1-\gamma)^2} \sup_k \|v_k - \tilde{v_k}\|_{\mu_k},$$

where v_k and $\tilde{v_k}$ are true and approximated value at step k given the current policy. The norm $\|\cdot\|_\mu$ denotes a weighted *quadratic norm* with distribution μ. The distribution μ is arbitrary, and μ_k depends on μ and the current transition matrix [Munos, 2003]. Similar bounds hold for algorithms based on Bellman residual minimization, when $\|v_k - \tilde{v_k}\|$ is replaced by $\|r - (I - \gamma P_{\pi_k})\tilde{v_k}\|$, where P_{π_k} is a transition matrix of the current policy. In addition, these bounds hold also for the *max norm* [Bertsekas and Tsitsiklis, 1996]. In the following, we refer to the value function v_k and its approximation $\tilde{v}_k$ as v and $\tilde{v}$ respectively, without using the index k.

The main focus of the paper is the construction a good basis for algorithms that minimize $\|v - \tilde{v}\|_2$ in each iteration. We focus only on the quadratic approximation bound, because the other bounds are related. Notice that $\|\cdot\|_\infty \leq \|\cdot\|_2$.

The paper is organized as follows. Section 2 describes the spectral methods for VFA in greater detail. The main contribution of the paper is an explanation of the good performance of spectral methods for VFA and its connection to methods for solving sparse linear systems. This is described in Section 3, where we also propose two new alternative algorithms. In Section 4, we show theoretical error bounds of one of the methods, and then in Section 6 we demonstrate that our method may significantly outperform the previously proposed spectral methods.

2 Proto-Value Functions

In this section, we describe in greater detail the proto-value function methods proposed in [Mahadevan, 2005]. These methods use the spectral graph framework to construct a basis for linear VFA that respects the topology of the problem. The states of the MDP for a fixed policy represent nodes of an *undirected* weighted graph (N, E). This graph may be represented by a *symmetric* adjacency matrix W, where W_{xy} represents the weight between the nodes. A diagonal matrix of the node degrees is denoted by D and defined as $D_{xx} = \sum_{y \in N} W_{xy}$. There are several definitions of the graph Laplacian, with the most common being the *normalized Laplacian*, defined for W as $L = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$. The advantage of the normalized Laplacian is that it is symmetric, and thus its eigenvectors are orthogonal. Its use is closely related to the random walk Laplacian $L_r = I - D^{-1}W$ and the combinatorial Laplacian $L_c = D - W$, which are commonly used to motivate the use of the Laplacian by making a connection to random walks on graphs [Chung, 1997].

A function f on a graph (N, E) is a mapping from each vertex to $\mathbb{R}$. One well-defined measure of smoothness of a function on a graph is its Sobolev norm. The bottom eigenvectors of L_c may be seen as a good approximation to smooth functions, that is, functions with low Sobolev norm [Chung, 1997].

The application of the spectral framework to VFA is straightforward. The value function may be seen as a function on the graph of nodes that correspond to states. The edge weight between two nodes is determined by the probability of transiting either way between the corresponding states, given a fixed policy. Usually, the weight is 1 if the transition is possible either way and 0 otherwise. Notice that these weights must be symmetric, unlike the transition probabilities. If we assume the value function is smooth on the induced graph, the spectral graph framework should lead to good results. While the adjacency matrix with edge weights of 1 was usually used before, [Mahadevan, 2005] also discusses other schemes.

While the approach is interesting, it suffers from some problems. In our opinion, the good performance of this method has not been sufficiently well explained, because the construction of the adjacency matrix is not well motivated. In addition, the construction of the adjacency matrix makes the method very hard to analyze. As we show later, using the actual transition matrix instead of the adjacency matrix leads to a better motivated algorithm, which is easy to derive and analyze.

The requirement of the value function being smooth was partially resolved by using diffusion wavelets in [Mahadevan and Maggioni, 2005; Maggioni and Mahadevan, 2006]. Briefly, diffusion wavelets construct a low-order approximation of the inverse of a matrix. An disadvantage of using the wavelets is the high computational overhead needed to construct the inverse approximation. The advantage is, however, that once the inverse is constructed it may be reused for different rewards as long as the transition matrix is fixed. Thus, we do not compare our approach to diffusion wavelets due to the different goals and computational complexities of the two methods.

3 Analysis

In this section we show that if we use the actual transition matrix instead of the random walk Laplacian of the adjacency matrix, the method may be well justified and analyzed. We assume in the following that the transition matrix P and the reward function r are available.

It is reasonable to explain the performance using P instead of L_r, because in the past applications P was usually very similar to $I - L_r$. This is because the transition in these problems were symmetric, and the adjacency matrix was based on a random policy. Notice that the eigenvectors of L_r and $I - L_r$ are the same. Moreover, if λ is an eigenvalue of L_r, then $1 - \lambda$ is an eigenvalue of $I - L_r$.

3.1 Spectral Approximation

Assuming that P is the transition matrix for a fixed Markov policy, we can express the value function as [Puterman, 2005]:

$$v = (I - \gamma P)^{-1} r = \sum_{i=0}^{\infty} (\gamma P)^i r. \tag{3.1}$$

The second equality follows from the Neumann series expansion. In fact, synchronous backups in the modified policy iteration calculate this series adding one term in each iteration.

We assume that the transition matrix is diagonalizable. The analysis for non-diagonalizable matrices would be similar, but would have to use Jordan decomposition. Let $x_1 \dots x_n$ be the eigenvectors of P with corresponding eigenvalues $\lambda_1 \dots \lambda_n$. Moreover, without loss of generality, $\|x_j\|_2 = 1$ for all j. Since the matrix is diagonalizable, $x_1 \dots x_n$ are linearly independent, and we can decompose the reward as:

$$r = \sum_{j=1}^{n} c_j x_j,$$

for some $c_1 \dots c_n$. Using $P^i x_j = \lambda_j^i x_j$, we have

$$v = \sum_{j=1}^{n} \sum_{i=0}^{\infty} (\gamma \lambda_j)^i c_j x_j = \sum_{j=1}^{n} \frac{1}{1 - \gamma \lambda_j} c_j x_j = \sum_{j=1}^{n} d_j x_j.$$

Considering a subset U of eigenvectors x_j as a basis will lead to the following bound on approximation error:

$$\|v - \tilde{v}\|_2 \leq \sum_{j \notin U} |d_j|. \tag{3.2}$$

Therefore, the value function may be well approximated by considering those x_j with greatest $|d_j|$. Assuming that all c_j are equal, then the best choice to minimize the bound (3.2) is to consider the eigenvectors with high λ_j. This is identical to taking the low order eigenvectors of the random walk Laplacian, as proposed in the spectral proto-VFA framework, only that the transition matrix is used instead. Using the analysis above, we propose a new algorithm in Subsection 3.3.

3.2 Krylov Methods

There are also other well-motivated base choices besides the eigenvectors. Another choice is to use some of the vectors in the Neumann series (3.1). We denote these vectors as

$y_i = P^i r$ for all $i \in \langle 0, \infty)$. Calculating the value function by progressively adding all vectors in the series, as done in the modified policy iteration, potentially requires an infinite number of iterations. However, since the linear VFA methods consider *any* linear combination of the basis vectors, we need at most n linearly independent vectors from the sequence $\{y_i\}_{i=0}^{\infty}$. Then it is preferable to choose those y_i with small i, since these are simple to calculate.

Interestingly, it can be shown that we just need $y_0 \dots y_{m-1}$ to represent any value function, where m is the degree of the minimal polynomial [Ipsen and Meyer, 1998] of $(I - \gamma P)$. Moreover, even taking fewer vectors than that may be a good choice in many cases. To show that $y_0 \dots y_{m-1}$ vectors are sufficient to precisely represent any value function, let the minimal polynomial be: $p(A) = \sum_{i=0}^{m} \alpha_i A^i$. Then let

$$B = \frac{1}{\alpha_0} \sum_{i=0}^{m-1} \alpha_{i+1} A^i.$$

By algebraic multiplication, we have $BA = I$. Having this, the value function may be represented as:

$$v = Br = \frac{1}{\alpha_0} \sum_{i=0}^{m-1} \alpha_{i+1} (I - \gamma P)^i r = \sum_{i=0}^{m-1} \beta_i y_i,$$

for some β_i. A more rigorous derivation may be found for example in [Ipsen and Meyer, 1998; Golub and Loan, 1996]. The space spanned by $y_0 \dots y_{m-1}$ is known as *Krylov space* (or Krylov subspace), denoted as $\mathcal{K}(P, r)$. It has been previously used in a variety of numerical methods, such as GMRES, or Lancoz, and Arnoldi [Golub and Loan, 1996]. It is also common to combine the use of eigenvectors and Krylov space, what is known as an *augmented Krylov methods* [Saad, 1997]. These methods actually subsume the methods based on simply considering the largest eigenvectors of P. We discuss this method in Subsection 3.3.

3.3 Algorithms

In this section we propose two algorithms based on the previous analysis for constructing a good basis for VFA. These algorithms deal only with the selection of the basis and they can be arbitrarily incorporated into any algorithm based on approximate policy iteration.

The first algorithm, which we refer to as the *weighted spectral method*, is to form the basis from the eigenvectors of the transition matrix. This is in contrast with the method presented in [Mahadevan, 2005], which uses any of the Laplacians. Moreover, we propose to choose the vectors with greatest $|d_j|$ value, in contrast to choosing the ones with largest λ_j. The reason is that this leads to minimization of the bound in (3.2).

A practical implementation of this algorithm faces some major obstacles. One issue is that there is no standard eigenvector solver that can efficiently calculate the top eigenvectors with regard to $|d_j|$ of a sparse matrix. However, such a method may be developed in the future. In addition, when the transition matrix is not diagonalizable, we need to calculate the Jordan decomposition, which is very time-consuming and unstable. Finally, some of the eigenvectors and eigenvalues may contain complex numbers, what would require a revision of the approximate policy iteration algorithms. If these issues were resolved, this may be a viable alternative to other methods.

Require: P, r, k - number of eigenvectors in the basis, l - total number of vectors
  Let $z_1 \dots z_k$ be the top real eigenvectors of P
  $z_{k+1} \leftarrow r$
  for $i \leftarrow 1 \dots l + k$ **do**
    if $i > k + 1$ **then**
      $z_i \leftarrow P z_{i-1}$
    end if
    for $j \leftarrow 1 \dots (i - 1)$ **do**
      $z_i \leftarrow z_i - \langle z_j, z_i \rangle z_j$
    end for
    if $\|z_i\| \approx 0$ **then**
      break
    end if
    $z_i \leftarrow \frac{1}{\|z_i\|} z_i$
  end for

Figure 1: Augmented Krylov method for basis construction.

The second algorithm we propose is to use the vectors from the augmented Krylov method, that is, to combine the vectors in the Krylov space with a few top eigenvectors. A pseudocode of the algorithm is in Figure 1. The algorithm calculates an orthonormal basis of the augmented Krylov space using a modified Gram-Schmidt method.

Using the Krylov space eliminates the problems with non-diagonalizable transition matrices and complex eigenvectors. Another reason to combine these two methods is to take advantage of approximation properties of both methods. Intuitively, Krylov vectors capture the short-term behavior, while the eigenvectors capture the long-term behavior. Though, there is no reliable decision rule to determine the right number of augmenting eigenvectors, it is preferable to keep their number relatively low. This is because they are usually more expensive to compute than vectors in the Krylov space.

4 Approximation Bounds

In this section, we briefly present a theoretical error bound guaranteed by the augmented Krylov methods. The bound characterizes the worst-case approximation error of the value function, given that the basis is constructed using the current transition matrix and reward vector. We focus on bounding the quadratic norm, what also implies the max norm.

We show the bound for $\|r - \tilde{v} + \gamma P \tilde{v}\|_2$. This bound applies directly to algorithms that minimize the Bellman residual [Bertsekas and Tsitsiklis, 1996], and it also implies:

$$\begin{aligned} \|v - \tilde{v}\|_\infty &= \|(I - \gamma P)^{-1} r - (I - \gamma P)^{-1} (I - \gamma P) \tilde{v}\|_\infty \\ &\leq \frac{1}{1 - \gamma} \|r - (I - \gamma P) \tilde{v}\|_2. \end{aligned}$$

This follows from the Neumann series expansion of the inverse and from $\|P\|_\infty = 1$.

In the following, we denote the set of m Krylov vectors as K_m and the chosen set of top eigenvectors of P as U. We also use $E(c, d, a)$ to denote an ellipse in the set of complex numbers with center c, focal distance d, and major semi-axis a. The approximation error for a basis constructed for the current policy may be bounded as the following theorem states.

Theorem 4.1. *Let $(I - \gamma P) = XSX^{-1}$ be a diagonalizable matrix. Further, let*

$$\phi = \max_{x \in U^{\perp} \|x\|_2 = 1} \|XTX^{-1}x\|_2,$$

where T is a diagonal matrix with 1 in place of eigenvalues of eigenvectors in $|U|$. The approximation error using the basis $K_m \cup U$ is bounded by:

$$\|r - (I - \gamma P)\tilde{v}\|_2 \leq \max\left\{\kappa(X)\frac{C_m(\frac{a_1}{d_1})}{C_m(\frac{c_1}{d_1})}, \phi\frac{C_m(\frac{a}{d})}{C_m(\frac{c}{d})}\right\},$$

where C_m is the Chebyshev polynomial of the first kind of degree m, and $\kappa(X) = \|X\|_2\|X^{-1}\|_2$. The parameters a, c, d and a_1, c_1, d_1 are chosen such that $E(c_1, d_1, a_1)$ includes the lower $n-|U|$ eigenvalues of $(I-\gamma P)$, and $E(c, d, a)$ includes all its eigenvalues.

The value of ϕ depends on the angle between the invariant subspace U and the other eigenvectors of P. If they are perpendicular, such as when P is symmetric, then $\phi = 0$.

Proof. To simplify the notation, we denote $A = (I - \gamma P)$. First, we show the standard bound on approximation by a Krylov space [Saad, 2003], ignoring the additional eigenvectors. In this case, the objective is to find such $w \in K_m$ that minimizes the following:

$$\|r - Aw\|_2 = \|r - \sum_{i=1}^{n} w(i)A^i r\|_2 = \|\sum_{i=0}^{n} -w(i)A^i r\|_2,$$

where $w(0) = -1$. Notice that this defines a polynomial in A multiplied by r with the constant factors determined by w. Let $\mathcal{P}_m$ denote the set of polynomials of degree at most m such that every $p \in \mathcal{P}_m$ satisfies $p(0) = 1$. The minimization problem may be then expressed as finding a polynomial $p \in \mathcal{P}_m$ that minimizes $\|p(A)r\|_2$. This is related to the value of the polynomial on complex numbers as [Golub and Loan, 1996]:

$$\|p(A)\|_2 \leq \kappa(X) \max_{i=1,\ldots,n} |p(\lambda_i)|,$$

where λ_i are the eigenvalues of A. A more practical bound may be obtained using Chebyshev polynomials, as for example in [Saad, 2003], as follows:

$$\min_{p \in \mathcal{P}_m} \max_{i=1,\ldots,n} |p(\lambda_i)| \leq \min_{p \in \mathcal{P}_m} \max_{\lambda \in E(c,d,a)} |p(\lambda)| \leq \frac{C_m(\frac{a}{d})}{C_m(\frac{c}{d})},$$

where the ellipse $E(c, d, a)$ covers all eigenvalues of A.

In the approximation with eigenvectors, the minimization may be expressed as follows:

$$\begin{aligned}
\min_v \|r - A\tilde{v}\|_2 &= \min_{w \in K_m} \min_{q \in U} \|r - Aw - AUq\|_2 = \\
&= \min_{p \in \mathcal{P}_m} \min_{q \in U} \|p(A)r + AUq\|_2 = \\
&= \min_{p \in \mathcal{P}_m} \min_{q \in U} \|p(A)(I - P_U)r + p(A)P_U r - AUq\|_2 \\
&= \min_{p \in \mathcal{P}_m} \|p(A)(I - P_U)r\|_2.
\end{aligned}$$

where P_U is the least squares projection matrix to U. Note that $\|p(A)P_U r - AUy\| = 0$, since U is an invariant space of A. Moreover, $(I - P_U) * r$ is perpendicular to U. The theorem then follows from the approximation by Chebyshev polynomials as described above, and the definition of matrix-valued function approximation [Golub and Loan, 1996]. □

This bound shows that it is important to choose an invariant subspace that corresponds to the top eigenvalues of P. It also shows that U should be chosen to be to the highest degree perpendicular to the remaining eigenvectors. Finally, the bound also implies that the approximation precision increases with lower γ, as this decreases the size of the ellipse to cover the eigenvalues.

5 Experiments

In this section we demonstrate the proposed methods on the two-room problem similar to the one used in [Mahadevan, 2005; Mahadevan and Maggioni, 2005]. This problem is a typical representative of some stochastic planning problems encountered in AI.

The MDP we use is a two-room grid with a single-cell doorway in the middle of the wall. The actions are to move in any of the four main direction. We use the policy with an equal probability of taking any action. The size of each room is 10 by 10 cells. The problem size is intentionally small to make explicit calculation of the value function possible. Notice that because of the structure of the problem and the policy, the random walk Laplacian has identical eigenvectors as the transition matrix in the same order. This allows us to evaluate the impact of choosing the eigenvectors in the order proposed by the weighted spectral method.

We used the problem with various reward vectors and discount factors, but with the same transition structure. The reward vectors are synthetically constructed to show possible advantages and disadvantages of the methods. They are shown projected onto the grid in Figure 2. Vector "Reward 1" represents an arbitrary smooth reward function. Vectors "Reward 2" and "Reward 3" are made perpendicular to the top 40 and the top 190 eigenvectors of the random walk Laplacian respectively.

A lower discount rate is generally more favorable to Krylov methods, because the value is closer to the value of the reward received in the state. This can also be seen from Theorem 4.1, because γ shrinks the eigenvalues. Therefore, we use problems that are generally less favorable to Krylov methods, we evaluate the approach using two high discount rates, $\gamma = 0.9$ and $\gamma = 0.99$.

In our experiments, we evaluate the Mean Squared Error (MSE) of the value approximation with regard to the number of vectors in the basis. The basis and the value function were calculated based on the same policy with equiprobable action choice. We obtained the true value function by explicitly evaluating $(I - \gamma P)^{-1}r$. Notice, however, that the true value does not need to be used in the approximation, it is only required to determine the approximation error. We compared the following 4 methods:

Laplacian The technique of using the eigenvectors of the random walk Laplacian of the adjacency matrix, as pro-

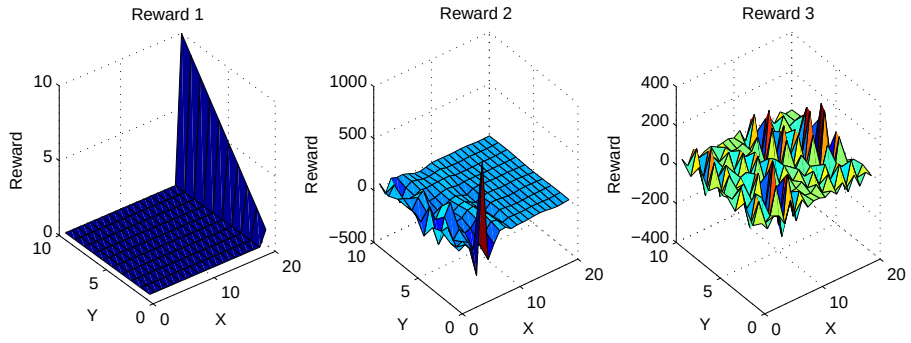

Figure 2: Reward vectors projected onto the grid

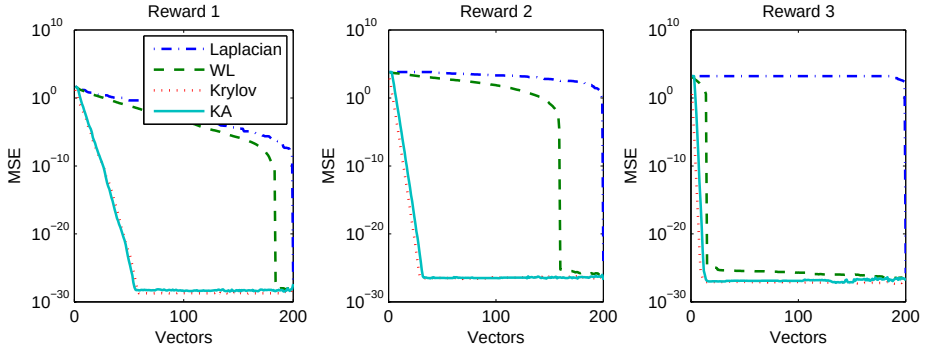

Figure 3: Mean squared error of each method, using discount of $\gamma = 0.9$. The MSE axis is in log scale.

posed in [Mahadevan, 2005]. The results of the normalized Laplacian are not shown because they were practically identical to the random walk Laplacian.

WL This is the *weighted spectral method*, described in Subsection 3.3, which determines the optimal order of the eigenvectors to be used based on the value of d_j.

Krylov This is the augmented Krylov method with no eigenvectors, as described in Subsection 3.3.

KA This is the augmented Krylov method with 3 eigenvectors, as proposed in Figure 1. The number of eigenvectors was chosen before running any experiments on this domain.

The results for $\gamma = 0.9$ are in Figure 3 and for $\gamma = 0.99$ are in Figure 4. "Reward 3" intentionally violates the smoothness assumption, but it is the easiest one to approximate for all methods except the Laplacian. In the case of "Reward 1" and $\gamma = 0.99$, the weighted spectral method and the random walk Laplacian outperform the ordinary Krylov method for the first 10 vectors. However, with additional vectors, both Krylov and augmented Krylov methods significantly outperform them.

Our results suggest that Krylov methods may offer superior performance compared to using the eigenvectors of the Laplacian. Since these methods have been found very effective in many sparse linear systems [Saad, 2003], they may perform well for basis construction also in other MDP problems. In addition, constructing a Krylov space is typically faster than constructing the invariant space of eigenvectors. Using Matlab on a standard PC, it took 2.99 seconds to calculate the 50 top eigenvectors, while it took only 0.24 second to calculate the first 50 vectors of the Krylov space for "Reward 1".

6 Discussion

We presented an alternative explanation of the success of Laplacian methods used for value approximation. This explanation allows us to more precisely determine when the method may work. Moreover, it shows that these methods are closely related to augmented Krylov methods. We demonstrated on a limited problem set that basis constructed from augmented Krylov methods may be superior to one constructed from the eigenvectors. In addition, both the weighted spectral method and the augmented Krylov method do not assume the value function to be smooth. Moreover, calculating vectors in the Krylov space is typically cheaper than calculating the eigenvectors.

An approach for state-space compression of Partially Observable MDPs (POMDP) that also uses Krylov space was proposed in [Poupart and Boutilier, 2002]. While POMDPs are a generalization of MDPs, the approach is not practical for large MDPs since it implicitly assumes that the number of observations is small. This is not true for MDPs, because the number of observations is equivalent to the number of states. In addition, the objectives of this approach are somewhat different, though the method is similar.

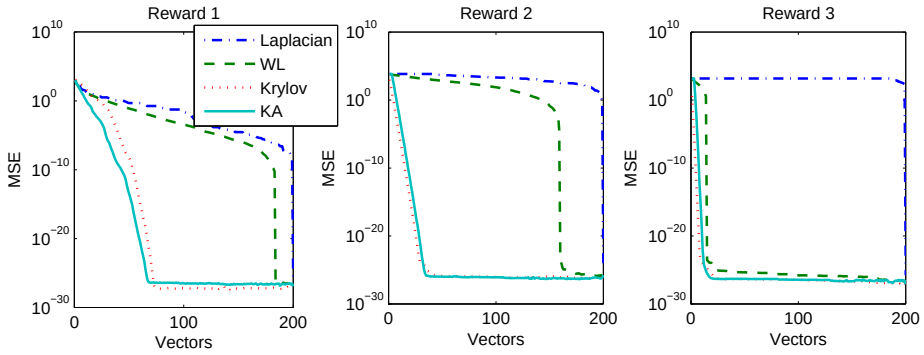

Figure 4: Mean squared error of each method, using discount of $\gamma = 0.99$. The MSE axis is in log scale.

One important issue that we did not address is an application to state spaces that are too large to be enumerated, such as factored MDPs. Because the vectors in Krylov space are as large as the whole state space, they cannot be enumerated either. However, the approach may be extended along similar lines as discussed in [Poupart and Boutilier, 2002].

In reinforcement learning, the MDP needs to be solved using only an estimation of the transition matrix from sampling. An additional problem is that the basis is not defined for states that were not sampled. This was addressed for eigenvectors for example in [Mahadevan *et al.*, 2006]. Thus augmenting the Krylov space by the eigenvectors also has the advantage that these methods may be directly applied.

We focused mainly on the VFA for a fixed policy without explicitly considering the control part of the problem. While these methods may be combined arbitrarily, a future challenge is to determine an efficient combination.

Our results suggest that exploring the connection between the basis construction in VFA and sparse linear solvers may bring about interesting advances in the future.

Acknowledgements

The author was supported by the National Science Foundation under Grant No. IIS-0328601. I also thank Sridhar Mahadevan, Hala Mostafa, Shlomo Zilberstein, and the anonymous reviewers for valuable comments.

References

[Bertsekas and Tsitsiklis, 1996] Dimitri P. Bertsekas and John N. Tsitsiklis. *Neuro-dynamic programming*. Athena Scientific, 1996.

[Chung, 1997] Fang Chung. *Spectral graph theory*. American Mathematical Society, 1997.

[Golub and Loan, 1996] Gene H. Golub and Charles F. Van Loan. *Matrix computations*. John Hopkins University Press, 3rd edition, 1996.

[Ipsen and Meyer, 1998] Ilse C. F. Ipsen and Carl D. Meyer. The idea behind Krylov methods. *American Mathematical Monthly*, 105(10):889–899, 1998.

[Maggioni and Mahadevan, 2006] Mauro Maggioni and Sridhar Mahadevan. Fast direct policy evaluation using multiscale analysis of markov diffusion processes. In *Proceedings of the International Conference on Machine Learning*, pages 601–608. ACM Press, 2006.

[Mahadevan and Maggioni, 2005] Sridhar Mahadevan and Mauro Maggioni. Value function approximation with diffusion wavelets and Laplacian eigenfuctions. In *Proceedings of Advances in Neural Information Processing Systems*, 2005.

[Mahadevan *et al.*, 2006] Sridhar Mahadevan, Mauro Maggioni, Kimberly Ferguson, and Sarah Osentoski. Learning representation and control in continuous Markov decision processes. In *Proceedings of the National Conference on Artificial Intelligence*, 2006.

[Mahadevan, 2005] Sridhar Mahadevan. Samuel meets Amarel: Automating value function approximation using global state space analysis. In *Proceedings of the National Conference on Artificial Intelligence*, 2005.

[Munos, 2003] Remi Munos. Error bounds for approximate policy iteration. In *Proceedings of the International Conference on Machine Learning*, pages 560–567, 2003.

[Poupart and Boutilier, 2002] Pascal Poupart and Craig Boutilier. Value-directed compression of POMDPs. In *Proceedings of Advances in Neural Information Processing Systems*, pages 1547–1554, 2002.

[Puterman, 2005] Martin L. Puterman. *Markov decision processes: Discrete stochastic dynamic programming*. John Wiley & Sons, Inc., 2005.

[Saad, 1995] Yosef Saad. Preconditioned Krylov subspace methods for CFD applications. In W. G. Hasbashi, editor, *Solution Techniques for Large Scale CFD Problems*, page 141. Wiley, 1995.

[Saad, 1997] Yousef Saad. Analysis of augmented Krylov subspace methods. *SIAM Journal on Matrix Analysis and Applications*, 18(2):435–449, April 1997.

[Saad, 2003] Yousef Saad. *Iterative methods for sparse linear systems*. SIAM, 2nd edition, 2003.

[Sutton and Barto, 1998] Richard S. Sutton and Andrew Barto. *Reinforcement learning*. MIT Press, 1998.