Full text available at: http://dx.doi.org/10.1561/2200000003

Learning Representation and Control in Markov Decision Processes: New Frontiers

Full text available at: http://dx.doi.org/10.1561/2200000003

Learning Representation and Control in Markov Decision Processes: New Frontiers

Sridhar Mahadevan

University of Massachusetts — Amherst
Amherst, MA 01003
USA
mahadeva@cs.umass.edu

the essence of knowledge

Boston – Delft

Foundations and Trends® in Machine Learning

Published, sold and distributed by:
now Publishers Inc.
PO Box 1024
Hanover, MA 02339
USA
Tel. +1-781-985-4510
www.nowpublishers.com
sales@nowpublishers.com

Outside North America:
now Publishers Inc.
PO Box 179
2600 AD Delft
The Netherlands
Tel. +31-6-51115274

The preferred citation for this publication is S. Mahadevan, Learning Representation and Control in Markov Decision Processes: New Frontiers, Foundation and Trends® in Machine Learning, vol 1, no 4, pp 403–565, 2008

ISBN: 978-1-60198-238-4

Foundations and Trends® in Machine Learning
Volume 1 Issue 4, 2008

Editorial Board

Editor-in-Chief:
Michael Jordan
Department of Electrical Engineering and Computer Science
Department of Statistics
University of California, Berkeley
Berkeley, CA 94720-1776

Editors

Editorial Scope

Foundations and Trends® in Machine Learning will publish survey and tutorial articles in the following topics:

- Adaptive control and signal processing
- Applications and case studies
- Behavioral, cognitive and neural learning
- Bayesian learning
- Classification and prediction
- Clustering
- Data mining
- Dimensionality reduction
- Evaluation
- Game theoretic learning
- Graphical models
- Independent component analysis
- Inductive logic programming
- Kernel methods
- Markov chain Monte Carlo
- Model choice
- Nonparametric methods
- Online learning
- Optimization
- Reinforcement learning
- Relational learning
- Robustness
- Spectral methods
- Statistical learning theory
- Variational inference
- Visualization

Information for Librarians
Foundations and Trends® in Machine Learning, 2008, Volume 1, 4 issues. ISSN paper version 1935-8237. ISSN online version 1935-8245. Also available as a combined paper and online subscription.

Foundations and Trends® in
Machine Learning
Vol. 1, No. 4 (2008) 403–565

DOI: 10.1561/2200000003

now
the essence of knowledge

Learning Representation and Control in Markov Decision Processes: New Frontiers

Sridhar Mahadevan

Department of Computer Science, University of Massachusetts — Amherst, 140 Governor's Drive, Amherst, MA 01003, USA, mahadeva@cs.umass.edu

Abstract

This paper describes a novel machine learning framework for solving sequential decision problems called Markov decision processes (MDPs) by iteratively computing low-dimensional representations and approximately optimal policies. A unified mathematical framework for learning representation and optimal control in MDPs is presented based on a class of singular operators called Laplacians, whose matrix representations have nonpositive off-diagonal elements and zero row sums. Exact solutions of discounted and average-reward MDPs are expressed in terms of a generalized spectral inverse of the Laplacian called the *Drazin inverse*. A generic algorithm called *representation policy iteration* (RPI) is presented which interleaves computing low-dimensional representations and approximately optimal policies. Two approaches for dimensionality reduction of MDPs are described based on geometric and reward-sensitive regularization, whereby low-dimensional representations are formed by *diagonalization* or *dilation* of Laplacian operators. Model-based and model-free variants of the RPI algorithm are presented; they are also compared experimentally on discrete and continuous MDPs. Some directions for future work are finally outlined.

Contents

1

Introduction

In this section, we introduce the problem of representation discovery in sequential decision problems called Markov decision processes (MDPs), whereby the aim is to solve MDPs by automatically finding "low-dimensional" descriptions of "high-dimensional" functions on a state (action) space. The functions of interest include policy functions specifying the desired action to take, reward functions specifying the immediate payoff for taking a particular action, transition distributions describing the stochastic effects of doing actions, as well as value functions that represent the long-term sum of rewards of acting according to a given policy. Our aim is to illustrate the major ideas in an informal setting, leaving more precise definitions to later sections. The concept of a Laplacian operator is introduced, and its importance to MDPs is explained. The general problem of dimensionality reduction in MDPs is discussed. A roadmap to the remainder of the paper is also provided.

1.1 Motivation

A variety of problems of practical interest to researchers across a diverse range of areas, from artificial intelligence (AI) [117] to operations

research (OR) [109, 110], can be abstractly characterized as "sequential decision-making." Namely, in all these problems, the task can be formulated in terms of a set of discrete or continuous set of states, in each of which a decision maker has to select one of a discrete set of actions, which incurs a reward or cost. The objective of the decision maker is to choose actions "optimally," that is, to compute a policy function that maps states to actions maximizing some long-term cumulative measure of rewards. Examples range from game-playing [132] and manufacturing [33] to robotics [81, 97], and scheduling [143]. MDPs [56, 110] have emerged as the standard mathematical framework to model sequential decision-making. A MDP is mathematically defined in terms of a set of states S; a set of actions A (which may often be conditionally defined in terms of choices available in the current state as A_s); a stochastic transition distribution $P^a_{ss'}$ describing the set of outcomes s' of performing action a in state s; and a payoff or "reward" function $R^a_{ss'}$. The optimization objective is to find a mapping or policy from states to actions that maximize some cumulative measure of rewards. Commonly used objective measures include maximizing the expected "discounted" sum of rewards (where rewards in the future are geometrically attenuated by powers of a fixed positive scalar value $\gamma < 1$), or maximizing the average reward or expected reward per decision. Crucially, the optimization goal takes into account the uncertainty associated with actions. Every policy defines a value function on the state space, where the value of a state is the sum of the immediate reward received and the expected value of the next state that results from choosing the action dictated by the policy. A MDP is typically solved through the well-known Bellman equation [110], which is a recursive equation relating the value of the current state to values of adjacent states.

Exact solution methods, such as linear programming [36], policy iteration [56], and value iteration [110], assume value functions are represented using a table (or more generally on a fixed set of *basis functions*). The complexity of these algorithms is typically polynomial (cubic) in the size of the discrete state space $|S|$ (or exponential in the size of any compact description of the state space). When the number of states is large or if the state space is continuous, exact representations become infeasible, and some parametric or nonparametric function

approximation method needs to be used. For example, if the states S of a discrete MDP are enumerated from $s = 1, \ldots, s = |S|$, where $|S| = n$, then functions over this discrete state space can be viewed as vectors that lie in a Euclidean space $\mathbb{R}^{|S|}$. Most previous work on approximately solving large MDPs surveyed in books on *approximate dynamic programming* [109], *neuro-dynamic programming* [12], and *reinforcement learning* [129], assume that MDPs are solved approximately by a set of handcoded "features" or basis functions mapping a state s to a k-dimensional real vector $\phi(s) \in \mathbb{R}^k$, where $k \ll |S|$.

Popular choices of parametric bases include radial basis functions (RBFs), neural networks, CMACs, and polynomials. Concretely, a polynomial basis can be viewed as an $|S| \times k$ matrix, where the ith column represents the basis function $1, 2^i, 3^i, \ldots, |S|^i$. A radial basis function $\phi_k(s) = e^{-\frac{||s-s_k||^2}{2\sigma^2}}$, where σ is a scaling factor, and s_k is the "center" of the basis function. A value function V is approximated as a linear combination of basis functions, namely: $V \approx \Phi w$, where Φ is a matrix whose columns are the specified basis functions, and w is a weight vector. If the number of columns of Φ is $k \ll |S|$, then Φ can be viewed as providing a low-dimensional projection of the original value function $\in \mathbb{R}^{|S|}$ to a subspace $\in \mathbb{R}^k$.

It has long been recognized that traditional parametric function approximators, such as RBFs, may have difficulty accurately approximating value functions due to nonlinearities in a MDP's state space (see Figure 1.1). Dayan [35] and Drummond [40] have noted that states close in Euclidean distance may have values that are very far apart (e.g., two states on opposite sides of a wall in a spatial navigation task). A traditional parametric architecture, such as an RBF, makes the simplifying assumption that the underlying space has Euclidean geometry.

The same issues arise in continuous MDPs as well. Figure 1.2 shows a set of samples produced by doing a random walk in a 2D inverted pendulum task. Here, the state variables are θ, the pole angle, and $\dot{\theta}$, the angular velocity. Note that in this task, and in many other continuous control tasks, there are often physical constraints that limit the "degrees of freedom" to a lower-dimensional manifold, resulting in motion along highly constrained regions of the state space. Figure 1.2

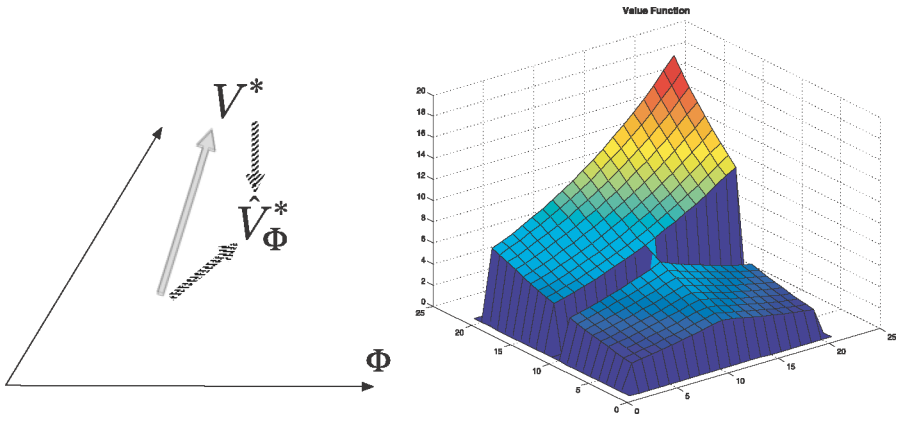


Fig. 1.1 *Left*: Dimensionality reduction of a MDP M involves finding a set of bases Φ such that any function on a MDP's state space, such as its optimal value function V^*, can be compressed effectively. *Right*: The optimal value function in a "two-room" discrete MDP with 400 states. The agent can take actions in the four compass directions. Each action succeeds with probability 0.9, otherwise leaves the agent in the same state. The agent is "rewarded" only for reaching a corner "goal" state. Access to each room from the other is available only through a central door, and this "bottleneck" results in a nonlinear optimal value function. This value function is ostensibly a high dimensional vector $\in \mathbb{R}^{400}$, but can be compressed onto a much lower-dimensional subspace.

also shows an approximation to the optimal value function constructed using a linear combination of "proto-value" basis functions [77], or eigenfunctions obtained by diagonalizing a random walk operator on a graph connecting nearby samples.

Both the discrete MDP shown in Figure 1.1 and the continuous MDP shown in Figure 1.2 have "inaccessible" regions of the state space, which can be exploited in focusing the function approximator to accessible regions. Parametric approximators, as typically constructed, do not distinguish between accessible and inaccessible regions. The approaches described below go beyond modeling just the reachable state space, in that they also build representations based on the transition matrix associated with a specific policy and a particular reward function [103, 106]. By constructing basis functions adapted to the nonuniform density and geometry of the state space, as well as the transition matrix and reward function, the approaches described in this paper are able to construct adaptive representations that can outperform parametric bases.

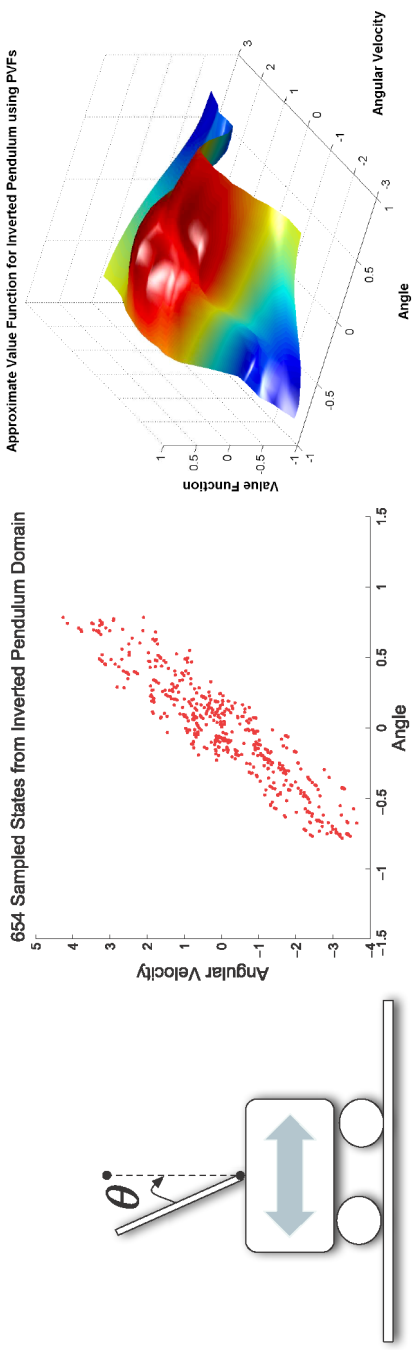


Fig. 1.2 *Left*: The inverted pendulum is a continuous MDP that involves keeping the pole upright by applying equal and opposite forces on the cart. *Middle*: Samples from a series of random walks in a 2D inverted pendulum task. Due to physical constraints, the samples are largely confined to a narrow manifold in the state space. *Right*: By explicitly modeling this manifold by a graph, one of the basis construction methods described in this paper derives customized basis functions called proto-value functions (PVFs) [83] by diagonalizing a random walk operator on a graph connecting nearby samples. An approximation of the optimal value function using proto-value functions is illustrated.

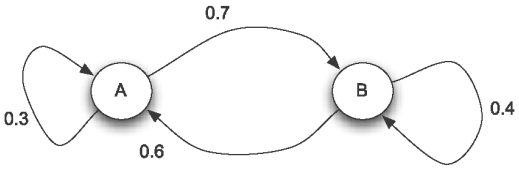
1.2 Laplacian Operators

A unique perspective adopted in this paper is based on exploring links between a family of singular matrices, termed *Laplacians*, and the solution of MDPs.[1] In continuous spaces, the (symmetric) Laplacian operator has been the object of study for almost two centuries: it has been called "the most beautiful object in all of mathematics and physics" [95] as it has played a central role in physics and in many areas of mathematics. On graphs, the discretized (symmetric and non-symmetric) Laplacian has been studied extensively in graph theory, where its spectra reveal structural properties of undirected and directed graphs [26, 27]. Stated in its most general form, the (nonsymmetric) Laplacian matrix is one whose off-diagonal elements are nonpositive and whose row sums are equal to 0 [2, 22, 24]. As we show in this paper, there are strong connections between Laplacian matrices and MDPs. In particular, for any MDP, either in the discounted or average-reward seting, its solution can be shown to involve computing a generalized Laplacian matrix.

Since their row sums are equal to 0, Laplacian matrices are singular, and they do not have a direct inverse (the nullspace is nontrivial since the constant vector of all 1s is an eigenvector associated with the 0 eigenvalue). However, a family of generalized inverses exist for low-rank and singular matrices. The well-known *Moore-Penrose* pseudo-inverse is widely used in least-squares approximation, which will be useful later in this paper. However, a less well-known family of *spectral* inverses — the Drazin inverse (and a special instance of it called the group inverse) is of foundational importance to the study of Markov chains. Indeed, Campbell and Meyer [20] state that:

> For an m-state [Markov] chain whose transition matrix is T, we will be primarily concerned with the matrix $A = I - T$. Virtually everything that one wants to know about a chain can be extracted from A and its Drazin inverse.

[1] In this paper, we use the term "operator" to mean a mapping on a finite or infinite-dimensional space, and the term "matrix" to denote its representation on a specific set of bases.



$$\mathbb{L} = I - P = \begin{bmatrix} 0.7 & -0.7 \\ -0.6 & 0.6 \end{bmatrix}$$

Fig. 1.3 Illustration of a Laplacian operator associated with a MDP. *Left*: A simple two-state Markov chain with transition matrix P. *Right*: Its associated Laplacian matrix. Exact (and approximate) solutions to MDPs can be expressed in terms of a generalized spectral inverse of such singular Laplacian matrices.

We denote transition matrices generally as P, and define the Laplacian associated with a transition matrix as $\mathbb{L} = I - P$ [2, 22, 24] (see Figure 1.3). It has long been known that the Drazin inverse of the singular Laplacian matrix $\mathbb{L}$ reveals a great deal of information about the structure of the Markov chain [92, 121]. In particular, the states in a Markov chain can be partitioned into its various recurrent classes or transient classes based on the Drazin inverse. Also, the sensitivity of the invariant distribution of an ergodic Markov chain to perturbations in the transition matrix can be quantified by the size of the entries in the Drazin inverse of the Laplacian.[2] The solution to average-reward and discounted MDPs can be shown to depend on the Drazin inverse of the Laplacian [21, 110].

As we will show in this paper, Laplacian matrices play a crucial role in the approximate solution of MDPs as well. We will explore a specific set of bases, called Drazin bases, to approximate solutions to MDPs (see Figure 1.4). In continuous as well as discrete MDPs, approximation requires interpolation of noisy samples of the true value function. A growing body of work in machine learning on nonlinear dimensionality reduction [73], manifold learning [8, 30, 116, 131], regression on graphs [99] and representation discovery [80] exploit the remarkable properties of the symmetric Laplacian operator on graphs and manifolds [115]. We will describe how regularization based on symmetric and nonsymmetric graph Laplacians can be shown to provide an automatic method of constructing basis functions for approximately solving

[2] Meyer [92] defines the *condition number* of a Markov chain with transition matrix P by the absolute value of the largest element in the Drazin inverse of $I - P$.

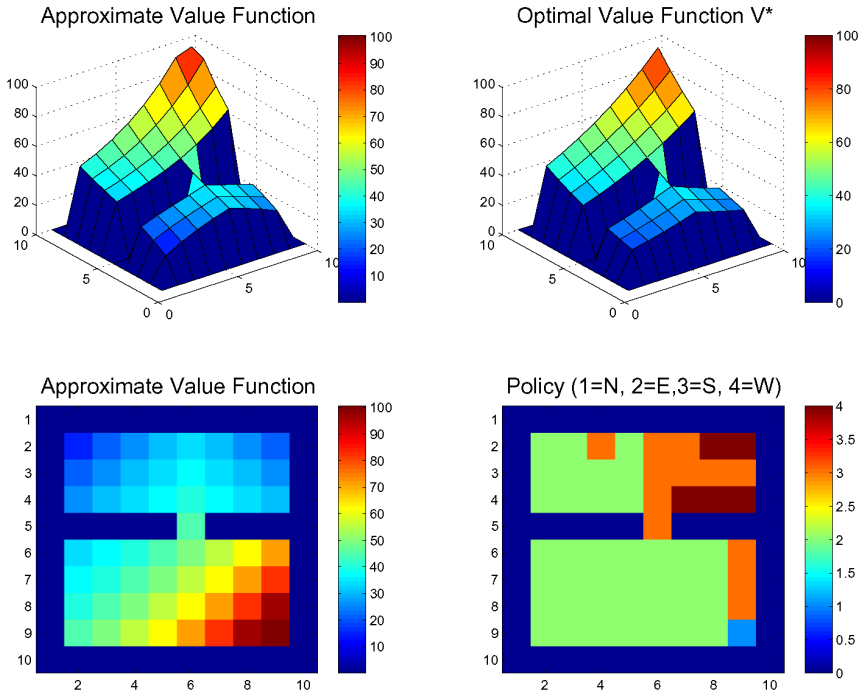


Fig. 1.4 *Top right*: The optimal value function in a two-room MDP with 100 states. *Top left*: Using just 4 Drazin bases, the original MDP is compressed onto a 4D problem, whose solution yields an optimal policy. *Bottom left*: The approximation plotted in 2D showing the state space layout. *Bottom right*: The learned policy using the approximation is optimal.

Table 1.1 Some Laplacian operators on undirected graphs. W is a symmetric weight matrix reflecting pairwise similarities. D is a diagonal matrix whose entries are row sums of W. All these operators are represented by matrices whose row sums are 0 and have non-positive off-diagonal entries.

Operator	Definition	Spectrum
Combinatorial Laplacian	$L = D - W$	$\lambda \in [0, 2\max_v d_v]$
Normalized Laplacian	$\mathcal{L} = I - D^{-1/2} W D^{-1/2}$	$\lambda \in [0, 2]$
Random Walk Laplacian	$L_r = I - D^{-1} W$	$\lambda \in [0, 2]$

MDPs [57, 77, 83, 102]. Table 1.1 describes a few examples of graph Laplacian matrices. The spectral properties of the graph Laplacian reveal a great deal of information about the structure of a graph. In particular, the eigevectors of the symmetric Laplacian yield a low-dimensional representation of a MDP, generating an orthogonal basis

that reflects the nonlinear geometry of the state space. We turn to describe the problem of dimensionality reduction in MDPs next.

1.3 Dimensionality Reduction of MDPs

Constructing a low-dimensional representation of a MDP means finding a *basis* Φ with respect to which the original MDP can be represented "compactly" and solved "efficiently."

Definition 1.1. *Basis Construction Problem in MDPs*: Given a Markov decision process M, find an "optimal" basis matrix Φ that provides a "low-dimensional" representation of M, and enables solving M as "accurately" as possible with the "least" computational effort.

Notions like "optimal," "accurately," and "least" will for now be left somewhat vague, but will be defined more precisely later. Note that the solution to the basis construction problem involves managing a set of mutually incompatible trade-offs. For example, a discrete MDP can be solved exactly using the unit vector ("table lookup") representation: this choice of basis optimizes the "accuracy' dimension, and requires no effort in finding the basis, but incurs a sizable computational cost. Exact algorithms like policy iteration [56] have a computational complexity cubic in the size of the state space $|S|$, or exponential in the size of any compact encoding of a state. On the other extreme, it is easy to project a high-dimensional value function $V \in \mathbb{R}^{|S|}$ on a low-order basis space of dimension $\mathbb{R}^k$, where $k \ll |S|$ by trivially choosing a set of random vectors (e.g., each vector is normalized to have length 1 and whose entries are distributed uniformly between 0 and 1). In this case, the cost of solving the MDP may be dramatically reduced, and the effort in finding the basis matrix is again trivial, but the resulting solution may be far from optimal.

It is possible to design an extremely compact basis matrix Φ if the optimal value function V^* is known — namely, use V^* itself! However, knowing V^* presupposes solving the original MDP (presumably on some initial basis, say the unit vectors). This latter solution illustrates the somewhat paradoxical situation that the basis construction

problem may require as much or more computational effort than that required to solve the original MDP. An example of an efficient basis is given in Figure 1.4. Here, the optimal policy is found by compressing a 100 state MDP into an effectively 4D space, whose solution gives an optimal policy. However, the cost of finding the Drazin bases is quite significant, since it involves finding the generalized inverse of the Laplacian. In many applications, a decision maker is required to solve many instances of the same problem. An example may be a robot that is tasked to retrieve a set of objects in a given environment, where each object is located in a different room. Thus, the cost of finding such low-dimensional representations may be amortized over the solution of a range of MDPs M_i, say all of which are defined on the same state (action) space, and differ only in the reward function. Finally, in the fully general setting of learning to solve MDPs, the decision maker may only have access to *samples* from the underlying MDP, say by simulation whereby training data are available in the form of trajectories (s_t, a_t, r_t, s_{t+1}). Here, s_t is the state at time t, a_t is the action selected, r_t is the payoff or reward received, and s_{t+1} is the resulting state from performing action a_t. This setting is commonly studied in a variety of areas, such as approximate dynamic programming [12, 109] and reinforcement learning [129]. The methods described later will illustrate these competing trade-offs and how to balance them. It is worthwhile to point out that these similar issues often arise in other domains, e.g., the use of wavelet methods to compress images [86].

1.3.1 Invariant Subspaces of a MDP

This paper describes a range of methods for constructing *adaptive* bases that are customized to the nonlinear *geometry* of a state space, or to a particular policy and reward function. The overarching theme underlying the various methods described in this paper is the notion of constructing representations by decomposing the effect of a linear operator T on the space of functions on a state (or state-action) space, principally by finding its *invariant* subspaces.[3] There are many reasons to find invariant subspaces of an operator T. The solution to a MDP can

[3] If $T : \mathbb{X} \to \mathbb{X}$ is an operator, a subspace $\mathbb{Y} \subseteq \mathbb{X}$ is called invariant if for each $x \in \mathbb{Y}$, $Tx \in \mathbb{Y}$.

be expressed abstractly in terms of finding the fixed point of an operator on the space of value functions. More formally, it can be shown that the value function V^π associated with a policy π is a fixed point of the Bellman operator T^π:

$$T^\pi(V^\pi)(x) = V^\pi(x). \tag{1.1}$$

Thus, the value function V^π forms a 1D invariant subspace of the Bellman operator. We will see, however, that there are compelling reasons for finding other larger invariant spaces. The invariant subspaces associated with a transition matrix P have the attractive property of eliminating prediction errors [11, 104]. Two main principles for constructing invariant subspaces of operators are explored: *diagonalization* and *dilation*.

1.3.2 Diagonalization and Dilation

In this paper, we explore two broad principles for solving the basis construction problem in MDPs by finding invariant subspaces, based on widely used principles in a variety of subfields in mathematics from group theory [123], harmonic analysis [52, 86], linear algebra [127], and statistics [59]. Diagonalization corresponds to finding eigenvectors of an operator: it reduces a possibly full matrix to a diagonal matrix. For example, in linear algebra [127], eigenvectors form invariant subspaces of T since $Tx = \lambda x = x\lambda$. Here, λ is the representation of T on the space spanned by x.

Diagonalization: One generic principle for basis construction involves remapping functions over the state space into a frequency-oriented coordinate system, generically termed *Fourier* analysis [125]. Examples include dimensionality reduction methods in statistics, such as principal components analysis (PCA) [59], low-rank approximations of matrices such as singular value decomposition (SVD) [49], and time-series and image-compression methods, such as the fast Fourier transform [136]. In the case of MDPs, the basis functions can be constructed by diagonalizing the state transition matrix. Often, these matrices are not diagonalizable or are simply not known. In this case, it is possible to construct bases by diagonalizing a "weaker" operator, namely a

random walk operator on a graph induced from a MDP's state space, similar to recent work on *manifold learning* [28, 98, 116, 131]. The graph Laplacian [26] is often used, since it is symmetric and its eigenvectors are closely related to that of the natural random walk. We call the bases resulting from diagonalizing the graph Laplacian "proto-value functions" or PVFs [77]. Unlike applications of graph-based machine learning, such as spectral clustering [96] or semi-supervised learning [98], approximating value functions on graphs involves new challenges as samples of the desired function are not readily available. Instead, an iterative procedure is used to sample from a series of functions $\hat{V}_t$, each of which is progressively closer to the desired optimal value function V^*. Furthermore, samples are not available *a priori*, but must be collected by exploration of the MDP's state space. The concept of invariant eigenspaces generalizes to infinite-dimensional Hilbert spaces [37]; one example of which is Fourier analysis in Euclidean spaces. We will explore building finite-dimensional Fourier bases on graphs, and see how to generalize these ideas to eigenfunctions on continuous spaces.

Dilation: Another general method for constructing invariant subspaces uses the principle of *dilation*. For example, a dilation operator on the space of functions on real numbers is $Tf(x) = f(2x)$. Several dilation-based approaches will be compared, including methods based on *Krylov* spaces, a standard approach of solving systems of linear equations [41, 118]. Applied to MDPs, this approach results in the reward function being "dilated" by powers of some operator, such as the transition matrix [106, 103]. A novel basis construction method called *Drazin bases* is described in this paper, which uses the Drazin inverse of the Laplacian $\mathbb{L}^D$. These are a new family of bases building on a theoretical result showing that the discounted value function of a MDP can be written in terms of a series of powers of the Drazin inverse.

Another dilation-based procedure involves a multiscale construction where functions over space or time are progressively remapped into time–frequency or space–frequency *atoms* [34, 86]. This multiscale construction is most characteristic of a family of more recent methods called *wavelets* [34, 86]. We will explore multiscale basis construction

on graphs and manifolds using a recent graph-based approach called *diffusion wavelets* [30]. There has been much work on multiscale wavelet methods in image compression and signal processing [87], which can also be viewed using the framework of invariance. We will construct multiscale wavelet bases on discrete graphs, building on the recently developed diffusion wavelet framework [76]. These approaches can be applied to a range of different operators, ranging from a model-based setting using the system dynamics transition matrix to "weaker" operators such as the natural random-walk on the (sampled) underlying state (action) space.

Combined with any of the procedures described above for constructing task-adaptive bases, it is possible to design a variety of architectures for simultaneously learning representation and control. One such framework is generically referred to as *representation policy iteration* [78], comprising of an outer loop where basis functions are constructed and an inner loop where the optimal policy within the linear span of the constructed bases is learned.

1.4 Roadmap to the Paper

The rest of the paper is organized as follows. Section 2 provides an overview of MDPs. Section 3 introduces a general family of Laplacian matrices, and shows how they are intimately connected to solving MDPs. Section 4 surveys various methods for approximately solving MDPs, including least-squares methods, linear programming methods, and reproducing kernel Hilbert space methods. Section 5 formulates the problem of constructing low-dimensional representations of MDPs more precisely, and describes a set of trade-offs that need to be balanced in coming up with effective solutions. Section 6 describes the first of the two main approaches to building basis functions by diagonalization. Section 7 describes methods for constructing representations by dilations of operators. Section 8 shows how these basis construction methods can be combined with methods for approximately solving MDPs to yield model-based techniques that simultaneously learn representation and control. Section 9 describes a generalization of the graph Laplacian operator to continuous sets called manifolds, as well as

an interpolation method for approximating continuous eigenfunctions of the manifold Laplacian. Section 10 describes a model-free version of the RPI framework, and evaluates its performance in continuous MDPs. Finally, Section 11 concludes with a brief survey of related work, and a discussion of directions for future work.

References

[1] D. Achlioptas, F. McSherry, and B. Scholkopff, "Sampling techniques for Kernel methods," in *Proceedings of the 14th International Conference on Neural Information Processing Systems (NIPS)*, pp. 335–342, MIT Press, 2002.

[2] R. Agaev and P. Cheboratev, "On the spectra of nonsymmetric Laplacian matrices," *Linear Algebra and Its Applications*, vol. 399, pp. 157–168, 2005.

[3] S. Amarel, "On representations of problems of reasoning about actions," in *Machine Intelligence 3*, (D. Michie, ed.), pp. 131–171, Elsevier/North-Holland, 1968.

[4] S. Axler, P. Bourdon, and W. Ramey, *Harmonic Function Theory*. Springer, 2001.

[5] J. Bagnell and J. Schneider, "Covariant Policy Search," in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1019–1024, 2003.

[6] C. T. H. Baker, *The Numerical Treatment of Integral Equations*. Oxford: Clarendon Press, 1977.

[7] A. Barto and S. Mahadevan, "Recent advances in hierarchical reinforcement learning," *Discrete Event Systems Journal*, vol. 13, pp. 41–77, 2003.

[8] M. Belkin and P. Niyogi, "Semi-supervised learning on Riemannian manifolds," *Machine Learning*, vol. 56, pp. 209–239, 2004.

[9] S. Belongie, C. Fowlkes, F. Chung, and J. Malik, "Spectral partitioning with indefinite Kernels using the Nyström extension," in *Proceedings of the 7th European Conference on Computer Vision*, pp. 531–542, 2002.

[10] A. Berman and R. Plemmons, *Nonnegative Matrices in the Mathematical Sciences*. SIAM Press, 1994.

[11] D. Bertsekas and D. Castanon, "Adaptive Aggregation Methods for infinite horizon dynamic programming," *IEEE Transactions on Automatic Control*, vol. 34, pp. 589–598, 1989.

[12] D. Bertsekas and J. Tsitsiklis, *Neuro-Dynamic Programming*. Belmont, MA: Athena Scientific, 1996.

[13] B. Bethke, J. How, and A. Ozdaglar, "Approximate dynamic programming using support vector regression," in *Proceedings of the IEEE Conference on Decision and Control*, 2008.

[14] G. Beylkin, R. R. Coifman, and V. Rokhlin, "Fast wavelet transforms and numerical algorithms," *Common Pure and Applied Mathematic*, vol. 44, pp. 141–183, 1991.

[15] L. Billera and P. Diaconis, "A geometric interpretation of the Metropolis-Hasting algorithm," *Statistical Science*, vol. 16, pp. 335–339, 2001.

[16] J. A. Boyan, "Least-squares temporal difference learning," in *Proceedings of the 16th International Conference on Machine Learning*, pp. 49–56, San Francisco, CA: Morgan Kaufmann, 1999.

[17] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[18] S. Bradtke and A. Barto, "Linear least-squares algorithms for temporal difference learning," *Machine Learning*, vol. 22, pp. 33–57, 1996.

[19] J. Bremer, R. Coifman, M. Maggioni, and A. Szlam, "Diffusion wavelet packets," *Applied and Computational Harmonic Analysis*, vol. 21, no. 1, pp. 95–112, July 2006.

[20] S. Campbell and C. Meyer, *Generalized Inverses of Linear Transformations*. Pitman, 1979.

[21] X. Cao, "The relations among potentials, perturbation analysis, and Markov decision processes," *Discrete-Event Dynamic Systems*, vol. 8, no. 1, pp. 71–87, 1998.

[22] J. Caughman and J. Veerman, "Kernels of directed graph Laplacians," *Electronic Journal of Combinatorics*, vol. 13, no. 1, pp. 253–274, 2006.

[23] I. Chavel, *Eigenvalues in Riemannian Geometry: Pure and Applied Mathematics*. Academic Press, 1984.

[24] P. Chebotarev and R. Agaev, "Forest matrices around the Laplacian matrix," *Linear Algebra and Its Applications*, vol. 15, no. 1, pp. 253–274, 2002.

[25] L. Chen, E. Krishnamurthy, and I. Macleod, "Generalized matrix inversion and rank computation by repeated squaring," *Parallel Computing*, vol. 20, pp. 297–311, 1994.

[26] F. Chung, *Spectral Graph Theory,* Number 92 in *CBMS Regional Conference Series in Mathematics*. American Mathematical Society, 1997.

[27] F. Chung, "Laplacians and the Cheeger inequality for directed graphs," *Annals of Combinatorics*, vol. 9, no. 1, pp. 1–19, April 2005.

[28] R. Coifman, S. Lafon, A. Lee, M. Maggioni, B. Nadler, F. Warner, and S. Zucker, "Geometric diffusions as a tool for harmonic analysis and structure definition of data. Part i: Diffusion maps," *Proceedings of National Academy of Science*, vol. 102, no. 21, pp. 7426–7431, May 2005.

[29] R. Coifman, S. Lafon, A. Lee, M. Maggioni, B. Nadler, F. Warner, and S. Zucker, "Geometric diffusions as a tool for harmonic analysis and structure definition of data. Part ii: Multiscale methods," *Proceedings of the National Academy of Science*, vol. 102, no. 21, pp. 7432–7437, May 2005.

[30] R. Coifman and M. Maggioni, "Diffusion wavelets," *Applied and Computational Harmonic Analysis*, vol. 21, no. 1, pp. 53–94, July 2006.

[31] R. Coifman, M. Maggioni, S. Zucker, and I. Kevrekidis, "Geometric diffusions for the analysis of data from sensor networks," *Curr Opin Neurobiol*, vol. 15, no. 5, pp. 576–584, October 2005.

[32] D. Cvetkovic, M. Doob, and H. Sachs, *Spectra of Graphs: Theory and Application*. Academic Press, 1980.

[33] T. Das, A. Gosavi, S. Mahadevan, and N. Marchalleck, "Solving semi-Markov decision problems using average-reward reinforcement learning," *Management Science*, vol. 45, no. 4, pp. 560–574, 1999.

[34] I. Daubechies, *Ten Lectures on Wavelets*, Society for Industrial and Applied Mathematics. 1992.

[35] P. Dayan, "Improving generalisation for temporal difference learning: The successor representation," *Neural Computation*, vol. 5, pp. 613–624, 1993.

[36] D. de Farias, "The linear programming approach to approximate dynamic programming," in *Learning and Approximate Dynamic Programming: Scaling Up to the Real World*, John Wiley and Sons, 2003.

[37] F. Deutsch, *Best Approximation in Inner Product Spaces*. Canadian Mathematical Society, 2001.

[38] T. Dietterich and X. Wang, "Batch value function approximation using support vectors," in *Proceedings of Neural Information Processing Systems*, MIT Press, 2002.

[39] P. Drineas and M. W. Mahoney, "On the Nyström method for approximating a Gram matrix for improved Kernel-based learning," *Journal of Machine Learning Research*, vol. 6, pp. 2153–2175, 2005.

[40] C. Drummond, "Accelerating reinforcement learning by composing solutions of automatically identified subtasks," *Journal of AI Research*, vol. 16, pp. 59–104, 2002.

[41] M. Eiermann and O. Ernst, "Geometric aspects of the theory of Krylov subspace methods," *Acta Numerica*, pp. 251–312, 2001.

[42] Y. Engel, S. Mannor, and R. Meir, "Bayes meets Bellman: The Gaussian process approach to temporal difference learning," in *Proceedings of the 20th International Conference on Machine Learning*, pp. 154–161, AAAI Press, 2003.

[43] K. Ferguson and S. Mahadevan, "Proto-transfer learning in Markov decision processes using spectral methods," in *International Conference on Machine Learning (ICML) Workshop on Transfer Learning*, 2006.

[44] M. Fiedler, "Algebraic connectivity of graphs," *Czechoslovak Mathematical Journal*, vol. 23, no. 98, pp. 298–305, 1973.

[45] D. Foster and P. Dayan, "Structure in the space of value functions," *Machine Learning*, vol. 49, pp. 325–346, 2002.

[46] A. Frieze, R. Kannan, and S. Vempala, "Fast Monte Carlo algorithms for finding low-rank approximations," in *Proceedings of the 39th Annual IEEE Symposium on Foundations of Computer Science*, pp. 370–378, 1998.

[47] M. Ghavamzadeh and S. Mahadevan, "Hierarchical average-reward reinforcement learning," *Journal of Machine Learning Research*, vol. 8, pp. 2629–2669, 2007.

[48] R. Givan and T. Dean, "Model minimization in Markov decision processes," *AAAI*, 1997.

[49] G. Golub and C. V. Loan, *Matrix Computations*. Johns Hopkins University Press, 1989.

[50] G. Gordon, "Stable function approximation in dynamic programming," Technical Report, CMU-CS-95-103, Department of Computer Science, Carnegie Mellon University, 1995.

[51] C. Guestrin, D. Koller, R. Parr, and S. Venkataraman, "Efficient solution algorithms for factored MDPs," *Journal of AI Research*, vol. 19, pp. 399–468, 2003.

[52] D. Gurarie, *Symmetries and Laplacians: Introduction to Harmonic Analysis, Group Representations and Laplacians*. North-Holland, 1992.

[53] M. Hein, J. Audibert, and U. von Luxburg, "Graph Laplacians and their convergence on random neighborhood graphs," *Journal of Machine Learning Research*, vol. 8, pp. 1325–1368, 2007.

[54] M. Herbster, M. Pontil, and L. Wainer, "Online learning over graphs," in *Proceedings of the Twenty-Second International Conference on Machine Learning*, 2005.

[55] J. Hoey, R. St-aubin, A. Hu, and C. Boutilier, "SPUDD: Stochastic planning using decision diagrams," in *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pp. 279–288, Morgan Kaufmann, 1999.

[56] R. Howard, *Dynamic Programming and Markov Decision Processes*. MIT Press, 1960.

[57] J. Johns and S. Mahadevan, "Constructing basis functions from directed graphs for value function approximation," in *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 385–392, ACM Press, 2007.

[58] J. Johns, S. Mahadevan, and C. Wang, "Compact spectral bases for value function approximation using Kronecker factorization," in *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 2007.

[59] T. Jolliffe, *Principal Components Analysis*. Springer-Verlag, 1986.

[60] P. Jones, M. Maggioni, and R. Schul, "Universal parametrizations via Eigenfunctions of the Laplacian and heat kernels," Forthcoming 2007.

[61] S. Kakade, "A natural policy gradient," in *Proceedings of Neural Information Processing Systems*, MIT Press, 2002.

[62] G. Karypis and V. Kumar, "A fast and high quality multilevel scheme for partitioning irregular graphs," *SIAM Journal of Scientific Computing*, vol. 20, no. 1, pp. 359–392, 1999.

[63] A. Kaveh and A. Nikbakht, "Block diagonalization of Laplacian matrices of symmetric graphs using group theory," *International Journal for Numerical Methods in Engineering*, vol. 69, pp. 908–947, 2007.

[64] R. Kretchmar and C. Anderson, "Using temporal neighborhoods to adapt function approximators in reinforcement learning," in *International Work Conference on Artificial and Natural Neural Networks*, pp. 488–496, 1999.

[65] H. Kushner and G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*. Springer, 2003.

[66] B. Kveton, "Learning basis functions in hybrid domains," in *Proceedings of the 21st National Conference on Artificial Intelligence*, pp. 1161–1166, 2006.

[67] J. Lafferty and G. Lebanon, "Diffusion Kernels on statistical manifolds," *Journal of Machine Learning Research*, vol. 6, pp. 129–163, 2005.

[68] S. Lafon, "Diffusion maps and geometric harmonics," PhD thesis, Yale University, Department of Mathematics and Applied Mathematics, 2004.

[69] M. Lagoudakis and R. Parr, "Least-squares policy iteration," *Journal of Machine Learning Research*, vol. 4, pp. 1107–1149, 2003.

[70] A. Langville and C. Meyer, "Updating the stationary vector of an irreducible Markov chain with an eye on google's pagerank," *SIAM Journal on Matrix Analysis*, vol. 27, pp. 968–987, 2005.

[71] J. C. Latombe, *Robot Motion Planning*. Kluwer Academic Press, 1991.

[72] S. Lavalle, *Planning Algorithms*. Cambridge University Press, 2006.

[73] J. Lee and M. Verleysen, *Nonlinear Dimensionality Reduction*. Springer, 2007.

[74] J. M. Lee, *Introduction to Smooth Manifolds*. Springer, 2003.

[75] L. Li, T. Walsh, and M. Littman, "Towards a unified theory of state abstraction for MDPs," in *Proceedings of the Ninth International Symposium on Artificial Intelligence and Mathematics*, pp. 531–539, 2006.

[76] M. Maggioni and S. Mahadevan, "Fast direct policy evaluation using multiscale analysis of Markov diffusion processes," in *Proceedings of the 23rd International Conference on Machine Learning*, pp. 601–608, New York, NY, USA: ACM Press, 2006.

[77] S. Mahadevan, "Proto-value functions: Developmental reinforcement learning," in *Proceedings of the International Conference on Machine Learning*, pp. 553–560, 2005.

[78] S. Mahadevan, "Representation policy iteration," in *Proceedings of the 21st Annual Conference on Uncertainty in Artificial Intelligence (UAI-05)*, pp. 372–37, AUAI Press, 2005.

[79] S. Mahadevan, "Fast spectral learning using lanczos eigenspace projections," in *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 2008.

[80] S. Mahadevan, *Representation Discovery Using Harmonic Analysis*. Morgan and Claypool Publishers, 2008.

[81] S. Mahadevan and J. Connell, "Automatic programming of behavior-based robots using reinforcement learning," *Artificial Intelligence*, vol. 55, pp. 311–365, 1992. Appeared originally as *IBM TR RC16359*, December 1990.

[82] S. Mahadevan and M. Maggioni, "Value function approximation with diffusion wavelets and Laplacian eigenfunctions," in *Proceedings of the Neural Information Processing Systems (NIPS)*, MIT Press, 2006.

[83] S. Mahadevan and M. Maggioni, "Proto-value functions: A Laplacian framework for learning representation and control in Markov decision processes," *Journal of Machine Learning Research*, vol. 8, pp. 2169–2231, 2007.

[84] S. Mahadevan, M. Maggioni, K. Ferguson, and S. Osentoski, "Learning representation and control in continuous Markov decision processes," in *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 2006.

[85] S. Mahadevan, N. Marchalleck, T. Das, and A. Gosavi, "Self-improving factory simulation using continuous-time average-reward reinforcement learning," in *Proceedings of 14th International Conference on Machine Learning*, pp. 202–210, Morgan Kaufmann, 1997.

[86] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Transactions on Pattern Analysis of Machanical Intelligence*, vol. 11, no. 7, pp. 674–693, 1989.

[87] S. Mallat, *A Wavelet Tour in Signal Processing*. Academic Press, 1998.

[88] D. Malsen, M. Orrison, and D. Rockmore, "Computing isotypic projections with the lanczos iteration," *SIAM*, vol. 2, nos. 60/61, pp. 601–628, 2003.

[89] S. Mannor, I. Menache, A. Hoze, and U. Klein, "Dynamic abstraction in reinforcement learning via clustering," *International Conference on Machine Learning*, 2004.

[90] A. McGovern, "Autonomous discovery of temporal abstractions from interactions with an environment," PhD thesis, University of Massachusetts, Amherst, 2002.

[91] M. Meila and J. Shi, "Learning segmentation by random walks," *NIPS*, 2001.

[92] C. Meyer, "Sensitivity of the stationary distribution of a Markov chain," *SIAM Journal of Matrix Analysis and Applications*, vol. 15, no. 3, pp. 715–728, 1994.

[93] A. Moore, "Barycentric interpolators for continuous space and time reinforcement learning," in *Advances in Neural Information Processing Systems*, MIT Press, 1998.

[94] R. Munos, "Error bounds for approximate policy iteration," in *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 560–567, 2003.

[95] E. Nelson, *Tensor Analysis*. Princeton University Press, 1968.

[96] A. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," *NIPS*, 2002.

[97] A. Ng, H. Kim, M. Jordan, and S. Sastry, "Autonomous helicopter flight via Reinforcement Learning," in *Proceedings of Neural Information Processing Systems*, 2004.

[98] P. Niyogi and M. Belkin, "Semi-supervised learning on Riemannian manifolds," Technical Report TR-2001-30, University of Chicago, Computer Science Deparment, November 2001.

[99] P. Niyogi, I. Matveeva, and M. Belkin, "Regression and regularization on large graphs," Techncial Report, University of Chicago, November 2003.

[100] D. Ormoneit and S. Sen, "Kernel-based reinforcement learning," *Machine Learning*, vol. 49, nos. 2–3, pp. 161–178, 2002.

[101] S. Osentoski, "Action-based representation discovery in Markov decision processes," PhD thesis, University of Massachusetts, Amherst, 2009.

[102] S. Osentoski and S. Mahadevan, "Learning state action basis functions for Hierarchical Markov decison processes," in *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 705–712, 2007.

[103] R. Parr, C. Painter-Wakefiled, L. Li, and M. Littman, "Analyzing feature generation for value function approximation," in *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 737–744, 2007.

[104] R. Parr, C. Painter-Wakefiled, L. Li, and M. Littman, "An analysis of linear models, linear value-function approximation, and feature selection for reinforcement learning," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2008.

[105] J. Peters, S. Vijaykumar, and S. Schaal, "Reinforcement learning for humanoid robots," in *Proceedings of the Third IEEE-RAS International Conference on Humanoid Robots*, 2003.

[106] M. Petrik, "An analysis of Laplacian methods for value function approximation in MDPs," in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2574–2579, 2007.

[107] P. Poupart and C. Boutilier, "Value Directed Compression of POMDPs," in *Proceedings of the International Conference on Neural Information Processing Systems (NIPS)*, 2003.

[108] P. Poupart, C. Boutilier, R. Patrascu, and D. Schuurmans, "Piecewise linear value function approximation for factored Markov decision processes," in *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pp. 285–291, 2002.

[109] W. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. Wiley, 2007.

[110] M. L. Puterman, *Markov Decision Processes*. New York, USA: Wiley Interscience, 1994.

[111] C. Rasmussen and M. Kuss, "Gaussian processes in reinforcement learning," in *Proceedings of the International Conference on Neural Information Processing Systems*, pp. 751–759, MIT Press, 2004.

[112] B. Ravindran and A. Barto, "SMDP homomorphisms: An algebraic approach to abstraction in semi-Markov decision processes," in *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, 2003.

[113] C. Robert and G. Casella, *Monte-Carlo Methods in Statistics*. Springer, 2005.

[114] K. Rohanimanesh and S. Mahadevan, "Coarticulation: An approach for generating concurrent plans in Markov decision processes," in *Proceedings of the International Conference on Machine Learning*, ACM Press, 2005.

[115] S. Rosenberg, *The Laplacian on a Riemannian Manifold*. Cambridge University Press, 1997.

[116] S. Roweis and L. Saul, "Nonlinear dimensionality reduction by local linear embedding," *Science*, vol. 290, pp. 2323–2326, 2000.

[117] S. Rusell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Prentice-Hall, 2002.

[118] Y. Saad, *Iterative Methods for Sparse Linear Systems*. SIAM Press, 2003.
[119] B. Sallans and G. Hinton, "Reinforcement learning with factored states and actions," *Journal of Machine Learning Research*, vol. 5, pp. 1063–1088, 2004.
[120] B. Scholkopf and A. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2001.
[121] P. Schweitzer, "Perturbation theory and finite Markov chains," *Journal of Applied Probability*, vol. 5, no. 2, pp. 410–413, 1968.
[122] P. Schweitzer and A. Seidmann, "Generalized polynomial approximations in Markov decision processes," *Journal of Mathematical Analysis and Applications*, vol. 110, pp. 568–582, 1985.
[123] J. Serre, *Linear Representations of Finite Groups*. Springer, 1977.
[124] R. St-Aubin, J. Hoey, and C. Boutilier, "Approximate policy construction using decision diagrams," *NIPS*, 2000.
[125] E. M. Stein and R. Shakarchi, *Fourier Analysis: An Introduction*. Princeton University Press, 2003.
[126] G. Stewart and J. Sun, *Matrix Perturbation Theory*. Academic Press, 1990.
[127] G. Strang, *Introduction to Linear Algebra*. Wellesley-Cambridge Press, 2003.
[128] D. Subramanian, "A theory of justified reformulations," PhD thesis, Stanford University, 1989.
[129] R. Sutton and A. G. Barto, *An Introduction to Reinforcement Learning*. MIT Press, 1998.
[130] R. S. Sutton, "Learning to predict by the methods of temporal differences," *Machine Learning*, vol. 3, pp. 9–44, 1988.
[131] J. Tenenbaum, V. de Silva, and J. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, pp. 2319–2323, 2000.
[132] G. Tesauro, "Td-gammon, a self-teaching backgammon program, achieves master-level play," *Neural Computation*, vol. 6, pp. 215–219, 1994.
[133] Y. Tsao, K. Xiao, and V. Soo, "Graph Laplacian based transfer learning in reinforcement learning," in *AAMAS '08: Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 1349–1352, 2008.
[134] B. Turker, J. Leydold, and P. Stadler, *Laplacian Eigenvectors of Graphs*. Springer, 2007.
[135] P. Utgoff and D. Stracuzzi, "Many-layered learning," *Neural Computation*, vol. 14, pp. 2497–2529, 2002.
[136] C. Van Loan, *Computational Frameworks for the Fast Fourier Transform*. SIAM Press, 1987.
[137] C. Van Loan and N. Pitsianis, "Approximation with Kronecker products," in *Linear Algebra for Large Scale and Real Time Applications*, pp. 293–314, Kluwer Publications, 1993.
[138] B. Van Roy, "Learning and value function approximation in complex decision processes," PhD thesis, MIT, 1998.
[139] G. Wahba, "Spline models for observational data," *Society for Industrial and Applied Mathematics*, 1990.

[140] C. Watkins, "Learning from delayed rewards," PhD thesis, King's College, Cambridge, England, 1989.

[141] Y. Wei, "Successive matrix squaring algorithm for computing the Drazin inverse," *Applied Mathematics and Computation*, vol. 108, pp. 67–75, 2000.

[142] C. Williams and M. Seeger, "Using the Nyström method to speed up Kernel machines," in *Proceedings of the International Conference on Neural Information Processing Systems*, pp. 682–688, 2000.

[143] W. Zhang and T. Dietterich, "A reinforcement learning approach to job-shop scheduling," in *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1114–1120, 1995.

[144] X. Zhou, "Semi-supervised learning with graphs," PhD thesis, Carnegie Mellon University, 2005.