Convergence of Synchronous Reinforcement Learning with Linear Function Approximation

Artur Merke ARTUR.MERKE@UDO.EDU
Lehrstuhl Informatik 1, University of Dortmund, 44227 Dortmund, Germany

Ralf Schoknecht RALF.SCHOKNECHT@ILKD.UNI-KARLSRUHE.DE
Institute of Logic, Complexity and Deduction Systems, University of Karlsruhe, 76128 Karlsruhe, Germany

Abstract

Synchronous reinforcement learning (RL) algorithms with linear function approximation are representable as inhomogeneous matrix iterations of a special form (Schoknecht & Merke, 2003). In this paper we state conditions of convergence for general inhomogeneous matrix iterations and prove that they are both necessary and sufficient. This result extends the work presented in (Schoknecht & Merke, 2003), where only a sufficient condition of convergence was proved. As the condition of convergence is necessary and sufficient, the new result is suitable to prove convergence *and* divergence of RL algorithms with function approximation. We use the theorem to deduce a new concise proof of convergence for the synchronous residual gradient algorithm (Baird, 1995). Moreover, we derive a counterexample for which the uniform RL algorithm (Merke & Schoknecht, 2002) diverges. This yields a negative answer to the open question if the uniform RL algorithm converges for arbitrary multiple transitions.

1. Introduction

Reinforcement Learning (RL) is concerned with learning optimal policies for the interaction of an agent with its environment. This problem is formulated as a dynamic optimisation problem and modelled as a Markov Decision Process (MDP). This MDP corresponds to the following learning situation. An agent interacts with the environment by selecting an action a from the available finite action set A and receiving feedback about the resulting immediate reward r. As a consequence of the action the environment makes a transition from a state s to a state s'. Accumulated over time the obtained rewards yield an evaluation of every state concerning its long-term desirability. The objective is to find an optimal policy that corresponds to an optimal value function. One algorithm to compute such an optimal policy is *policy iteration* (Bertsekas & Tsitsiklis, 1996). This algorithm consists of two steps that are executed in turn. The *policy evaluation* step determines the value function for a fixed policy. From this value function an improved policy is derived in the *policy improvement* step. In this paper we only consider policy evaluation.

As long as a tabular representation for the value function is used, RL algorithms like TD(λ) (Sutton, 1988) or the residual gradient algorithm (Baird, 1995) are known to converge to the optimal solution of the policy evaluation task. For large problems, however, a tabular representation of the value function is no longer feasible with respect to time and memory considerations. Therefore, linear feature-based function approximation is often used. Such function approximation makes the issue of convergence more complicated. On the one hand algorithms like TD(λ) may not converge at all (Baird, 1995). And on the other hand, even if they converge, different algorithms may converge to different solutions (Schoknecht, 2003).

For the TD(λ) algorithm with general linear function approximation convergence with probability one can be proved under the crucial condition that the states are sampled according to the steady state distribution of the underlying Markov chain (Tsitsiklis & Van Roy,

Appearing in *Proceedings of the 21st International Conference on Machine Learning*, Banff, Canada, 2004.

1997). This requirement may be disadvantageous for policy improvement as shown in (Koller & Parr, 2000) because it may lead to bad action choices in rarely visited parts of the state space. Moreover, in practical reinforcement learning it is desirable to take transition data from arbitrary sources, e.g. from on-line behaviour, archived data or from observing the system while under the control of some other policy. In this case a certain sampling distribution cannot be assured which may prevent convergence. Therefore, we need a convergence analysis for RL algorithms with linear function approximation when the transition data is arbitrary.

In (Schoknecht & Merke, 2003) a unified framework for synchronous RL algorithms with linear function approximation was considered. It was shown that the update rules of synchronous RL algorithms can be written as inhomogeneous iterations with a special common structure. Moreover, sufficient conditions of convergence for this class of iterations were derived. Our main theorem in this paper extends this result in two important ways. First, we consider a more general iteration that contains the above iterations as a special case. And second, we give a necessary *and* sufficient condition for the convergence of this general iteration. We apply our new theorem to prove the convergence of the synchronous residual gradient algorithm for a fixed set of arbitrary multiple transitions.

In (Merke & Schoknecht, 2002) the *uniform RL algorithm* in combination with interpolating grid based function approximators was introduced. In contrast to the synchronous TD(0) algorithm the uniform RL algorithm converges for arbitrary single transitions. However, it was an open question if the uniform RL algorithm also converges for arbitrary multiple transitions. Using the new theorem we construct a counterexample with two transitions that violates the necessary condition of convergence. This shows that the uniform RL algorithm diverges in the general case.

2. Convergence Results

We consider the general inhomogeneous matrix iteration of the form

$$x(k+1) = Ax(k) + b \tag{1}$$

were $A \in \mathbb{K}^{n\times n}$, $x(k)$, $b \in \mathbb{K}^n$ and $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$ denotes the field of real or complex numbers respectively. The iteration (1) can equivalently be written as

$$x(k+1) = \phi_{A,b}(x(k)) \tag{2}$$

with the affine mapping $\phi_{A,b}(x) := Ax + b$. The k-th element in the sequence is then given by $x(k) = \phi^k_{A,b}(x(0))$. There are four possible behaviours for the equivalent iterations (1) and (2)

Convergence

$$\texttt{Conv}(A,b) :\Longleftrightarrow \forall\, x \in \mathbb{K}^n : \lim_{k\to\infty} \phi^k_{A,b}(x) = a < \infty$$

Conditional convergence (the limit depends on the start value x)

$$\texttt{CondConv}(A,b) :\Longleftrightarrow \forall\, x \in \mathbb{K}^n : \lim_{k\to\infty} \phi^k_{A,b}(x) = a(x) < \infty$$

Boundedness (also known as Lagrange stability)

$$\texttt{Bounded}(A,b) :\Longleftrightarrow \forall\, x \in \mathbb{K}^n : \limsup_{k\to\infty} \|\phi^k_{A,b}(x)\| < \infty$$

Divergence

$$\texttt{Diverge}(A,b) :\Longleftrightarrow \exists\, x \in \mathbb{K}^n : \limsup_{k\to\infty} \|\phi^k_{A,b}(x)\| = \infty.$$

Obviously, the following relations between the different properties of the iterations hold: $\texttt{Conv}(A,b) \Rightarrow \texttt{CondConv}(A,b) \Rightarrow \texttt{Bounded}(A,b)$ and $\texttt{Bounded}(A,b) \Leftrightarrow \neg\texttt{Diverge}(A,b)$. The objective is to state necessary and sufficient conditions for the above properties that depend on A and b. For property $\texttt{Conv}(A,b)$ there exist standard numerical results which provide such necessary and sufficient conditions (Greenbaum, 1997). For the special case $b = 0$ conditions for the properties $\texttt{CondConv}(A,0)$ and $\texttt{Bounded}(A,0)$ are given in (Ludyk, 1985) (see proposition 1 below). However, there exist no necessary and sufficient conditions for the properties $\texttt{CondConv}(A,b)$ and $\texttt{Bounded}(A,b)$ with an arbitrary translation vector b. In this paper we state such conditions and prove that they are both necessary and sufficient. This result is especially important for approximate reinforcement learning (RL) where questions about convergence or boundedness of RL algorithms in conjunction with linear function approximators naturally arise (Bertsekas & Tsitsiklis, 1996; Schoknecht & Merke, 2003).

Our analysis will rely on spectral properties of the matrix A. In the following $\sigma(A)$ denotes the spectrum of A, i.e. the set of its eigenvalues. The eigenvalue with maximal absolute value defines the spectral radius $\rho(A) := \max\{|\lambda| \mid \lambda \in \sigma(A)\}$ of A. The eigenspace of A corresponding to the eigenvalue λ is denoted by $\mathcal{E}^A_\lambda = \{x \mid Ax = \lambda x\}$, and the generalised eigenspace by $\mathcal{H}^A_\lambda = \{x \mid \exists k : (\lambda I - A)^k x = 0\}$. From the definitions it follows that $\mathcal{E}^A_\lambda \subset \mathcal{H}^A_\lambda$, but in general $\mathcal{E}^A_\lambda \neq \mathcal{H}^A_\lambda$. The case $\mathcal{E}^A_\lambda = \mathcal{H}^A_\lambda$ means that if a vector is annihilated by $(\lambda I - A)^k$ for some k, then it is already annihilated for $k = 1$ (cf. (Gohberg et al., 1986)).

In the following we define two properties of a matrix A with eigenvalues $\lambda \in \sigma(A)$ that will be useful to state our results:

$$\texttt{Cond1}(A) :\Longleftrightarrow |\lambda| = 1 \text{ implies } \lambda = 1$$
$$\texttt{CondE}(A) :\Longleftrightarrow |\lambda| = 1 \text{ implies } \mathcal{E}_\lambda^A = \mathcal{H}_\lambda^A$$

The first condition ensures that 1 is the only eigenvalue with modulus one. The second condition ensures that eigenspaces of eigenvalues with modulus one are identical to the corresponding generalised eigenspaces. The following proposition reviews the results for a homogeneous iteration

Proposition 1 *For the homogeneous iteration*

$$x(k+1) = Ax(k)$$

with $A \in \mathbb{K}^{n\times n}$ and $x(k) \in \mathbb{K}^n$ the following holds

$$\begin{aligned}\texttt{Conv}(A,0) &\Longleftrightarrow \rho(A) < 1\\ \texttt{CondConv}(A,0) &\Longleftrightarrow \rho(A) \le 1 \wedge \texttt{CondE}(A) \wedge \texttt{Cond1}(A)\\ \texttt{Bounded}(A,0) &\Longleftrightarrow \rho(A) \le 1 \wedge \texttt{CondE}(A).\end{aligned}$$

Proof: The proof uses the Jordan canonical form of the matrix A and can be adapted from (Ludyk, 1985; Elaydi, 1999). ♢

Thus, in order to obtain conditional convergence from boundedness the only eigenvalue of the iteration matrix with modulus one may be $\lambda = 1$. In the following we present our main theorem. It completely clarifies the behaviour of the inhomogeneous iteration (1) and states the most general conditions for its convergence or boundedness. The idea behind the proof is to reduce the inhomogeneous case (1) via homogenisation to the homogeneous iteration $\tilde{x}(k+1) = \tilde{A}\tilde{x}(k)$. Then, we will use proposition 1 to obtain conditions for the different behaviours of the iteration. However, $\texttt{CondE}(A) \Leftrightarrow \texttt{CondE}(\tilde{A})$ generally does not hold. Therefore, we will need an additional condition $b \in \mathrm{Im}\{I - A\}$, where $\mathrm{Im}\{A\} = \{Ax \mid x \in \mathbb{K}^n\}$ denotes the range of the linear mapping A.

Theorem 1 *Consider the inhomogeneous matrix iteration (1). Then the following equivalences hold*

$$\texttt{Conv}(A,b) \Longleftrightarrow \rho(A) < 1 \tag{3}$$

Case 1: $1 \notin \sigma(A)$

$$\texttt{CondConv}(A,b) \Longleftrightarrow \rho(A) \le 1 \wedge \texttt{CondE}(A) \wedge \texttt{Cond1}(A) \tag{4}$$

$$\texttt{Bounded}(A,b) \Longleftrightarrow \rho(A) \le 1 \wedge \texttt{CondE}(A) \tag{5}$$

Case 2: $1 \in \sigma(A)$

$$\texttt{CondConv}(A,b) \Longleftrightarrow \rho(A) \le 1 \wedge \texttt{CondE}(A) \wedge \texttt{Cond1}(A) \wedge b \in Im\{I - A\} \tag{6}$$

$$\texttt{Bounded}(A,b) \Longleftrightarrow \rho(A) \le 1 \wedge \texttt{CondE}(A) \wedge b \in Im\{I - A\}. \tag{7}$$

Proof: Equivalence (3) is a standard numerical result (see (Greenbaum, 1997)), and was included for completeness. The iteration (1) can also be written in the homogeneous form

$$\tilde{x}(k+1) = \tilde{A}\tilde{x}(k) \tag{8}$$

where $\tilde{A} \in \mathbb{K}^{(n+1)\times(n+1)}$ and $\tilde{x}(k) \in \mathbb{K}^{n+1}$:

$$\tilde{A} = \left(\begin{array}{ccc|c} & A & & b \\ \hline 0 & \dots & 0 & 1 \end{array}\right), \quad \tilde{x} = \left(\begin{array}{c} x \\ \hline 1 \end{array}\right)$$

Let $p(\lambda)$ and $\tilde{p}(\lambda)$ be the the characteristic polynomials of A and $\tilde{A}$ respectively. Then, $\tilde{p}(\lambda) = p(\lambda)(1-\lambda)$ holds and we obtain

$$\sigma(\tilde{A}) = \sigma(A) \cup \{1\}. \tag{9}$$

Moreover, due to the block structure of $\tilde{A}$ the vector $(x_1, \dots, x_n)^\top$ is a (generalised) eigenvector of A corresponding to eigenvalue λ if and only if $(x_1, \dots, x_n, 0)^\top$ is a (generalised) eigenvector of $\tilde{A}$ corresponding to the same eigenvalue λ. Thus,

$$\mathcal{E}_\lambda^A = \mathcal{H}_\lambda^A \Longleftrightarrow \mathcal{E}_\lambda^{\tilde{A}} = \mathcal{H}_\lambda^{\tilde{A}}, \quad \text{for} \quad \lambda \neq 1. \tag{10}$$

As iteration (8) is just the equivalent homogenised form of iteration (1) the relations $\texttt{CondConv}(A,b) \Leftrightarrow \texttt{CondConv}(\tilde{A},0)$ and $\texttt{Bounded}(A,b) \Leftrightarrow \texttt{Bounded}(\tilde{A},0)$ hold.

Let us first consider the case $1 \notin \sigma(A)$. Together with (9) $\lambda = 1$ is a simple eigenvalue of $\tilde{A}$. Therefore, the corresponding eigenspace must at least be one-dimensional. On the other hand, the generalised eigenspace can be at most one-dimensional. Thus $\mathcal{E}_1^{\tilde{A}} = \mathcal{H}_1^{\tilde{A}}$. Together with (10) we obtain $\texttt{CondE}(A) \Leftrightarrow \texttt{CondE}(\tilde{A})$. From (9) we directly obtain $\rho(A) \le 1 \Leftrightarrow \rho(\tilde{A}) \le 1$. Thus, we have $\rho(A) \le 1 \wedge \texttt{CondE}(A) \Longleftrightarrow \rho(\tilde{A}) \le 1 \wedge \texttt{CondE}(\tilde{A})$. Proposition 1 states that this equivalent to $\texttt{Bounded}(\tilde{A},0)$ which is equivalent to $\texttt{Bounded}(A,b)$. This yields (5). Due to (9) it holds that $\texttt{Cond1}(A) \Leftrightarrow \texttt{Cond1}(\tilde{A})$. Together with (5) this gives (4).

We now consider the more interesting case $1 \in \sigma(A)$. *Assume that* $\texttt{CondE}(\tilde{A})$ *holds. With* (9) *we directly obtain* $\mathcal{E}_1^A = \mathcal{H}_1^A$. *And with* (10) *this yields* $\texttt{CondE}(A)$. *Moreover, the new eigenvalue 1 of* $\tilde{A}$ *yields a new eigenvector* $\tilde{x} = (x, x_{n+1})^\top$ *with* $x_{n+1} \neq 0$. *Without loss of generality assume that* $x_{n+1} = 1$. *It holds that*

$$\begin{aligned} \tilde{A}\tilde{x} = \tilde{x} &\iff Ax + b = x \\ &\iff b = (I - A)x \\ &\iff b \in Im\{I - A\}. \end{aligned} \tag{11}$$

Therefore, $\texttt{CondE}(\tilde{A}) \Rightarrow \texttt{CondE}(A) \wedge b \in Im\{I - A\}$. *On the other hand let* $\texttt{CondE}(A) \wedge b \in Im\{I - A\}$ *be valid. According to* (11) *the latter condition implies that* $\tilde{A}$ *has an eigenvector of the form* $\tilde{x} = (x, x_{n+1})^\top$ *with* $x_{n+1} \neq 0$. *Thus,* $\dim \mathcal{E}_1^{\tilde{A}} = \dim \mathcal{E}_1^A + 1$. *From* $\texttt{CondE}(A)$ *it follows that* $\mathcal{E}_1^A = \mathcal{H}_1^A$. *And due to* $\mathcal{E}_1^{\tilde{A}} \subset \mathcal{H}_1^{\tilde{A}}$ *it follows that* $\mathcal{E}_1^{\tilde{A}} = \mathcal{H}_1^{\tilde{A}}$. *Together with* (10) *this yields* $\texttt{CondE}(\tilde{A}) \Leftarrow \texttt{CondE}(A) \wedge b \in Im\{I - A\}$. *Hence we have shown the equivalence* $\texttt{CondE}(\tilde{A}) \Leftrightarrow \texttt{CondE}(A) \wedge b \in Im\{I - A\}$. *Together with proposition 1 this yields* (7). *With the same argument as above equivalence* (6) *follows from identity* (9) *and* (7). ♢

To get a better intuition for the conditions in theorem 1 we consider a simple example. The translation $x(k+1) = Ix(k) + b$ for $b \neq 0$ clearly diverges because of $x(k) = kb$. This can also be seen by observing that $\sigma(I) = \{1\}$ and $b \notin \text{Im}\{I - I\} = \{0\}$ which is a violation of equivalence (7) in theorem 1.

3. Application to Reinforcement Learning

The framework for synchronous RL algorithms in the policy evaluation case was set up in (Schoknecht & Merke, 2003). For a Markov decision process (MDP) with n states $S = \{s_1, \ldots, s_n\}$, action space A, state transition probabilities $p(s_i|a,s_j)$ and reward function $r : (S, A) \to \mathbb{R}$ policy evaluation is concerned with solving the Bellman equation

$$V^\pi = \gamma P^\pi V^\pi + R^\pi \tag{12}$$

for a fixed policy $\pi : S \to A$. V_i^π denotes the value of state s_i, $P_{i,j}^\pi = p(s_i|s_j, \pi(s_i))$ and $\gamma \in [0, 1)$ is the discount factor.

In practice due to the large number of states, the value function (vector) V is approximated by a linear combination of basis functions $\{\Phi_1, \ldots \Phi_F\}$ which can be written in matrix form as $V = \Phi w$ with $\Phi = [\Phi_1| \ldots |\Phi_F] \in \mathbb{R}^{n \times F}$. We can also write $\Phi^\top = [\varphi(s_1)| \ldots |\varphi(s_n)]$, where $\varphi(s_i) \in \mathbb{R}^F$ contains the feature vector of state s_i (see also (Bertsekas & Tsitsiklis, 1996)).

In (Schoknecht & Merke, 2003) it was shown that for one transition $x \to z$ with reward ρ the TD(0) algorithm for the solution of (12) can be written as

$$w(k+1) = (I + \alpha c d^\top) w(k) + \alpha c \rho \tag{13}$$

with $c = \varphi(x)$ and $d = \gamma\varphi(z) - \varphi(x)$.

This can be generalised for a set $T = \{(x_i, z_i, \rho_i) \mid i = 1, \ldots, m\}$ of m transitions

$$w(k+1) = (I + \alpha C D^\top) w(k) + \alpha C r, \tag{14}$$

where in the case of the synchronous TD(0) algorithm $r = (\rho_1, \ldots, \rho_m)^\top$, $C = [\varphi(x_1)| \ldots |\varphi(x_m)]$ and $D = [\gamma\varphi(z_1) - \varphi(x_1)| \ldots |\gamma\varphi(z_m) - \varphi(x_m)]$.

3.1. Residual Gradient Algorithm

As shown in (Baird, 1995) the synchronous TD(0) algorithm can diverge. And therefore, the residual gradient (RG) algorithm was proposed as a convergent alternative. According to (Schoknecht & Merke, 2003) this algorithm can be represented with appropriately chosen matrices C and D, where $C = -D$. The sufficient condition of convergence in (Schoknecht & Merke, 2003) was tailored to iterations of the form (14). As (14) is a special case of the more general iteration (1) the new theorem 1 can also be used to obtain a concise convergence proof for the RG algorithm.

Corollary 1 *For an arbitrary matrix* $C \in \mathbb{K}^{n \times m}$ *and* $w(k) \in \mathbb{R}^n$, $r \in \mathbb{R}^m$ *there exist a range of positive* α*'s such that the iteration*

$$w(k+1) = (I - \alpha C C^\top) w(k) + C r \tag{15}$$

converges for every initial value $w(0)$ *(the limit depends on* $w(0)$ *if* $CC^\top$ *is singular).*

Proof: We write $A_\alpha := I - \alpha C C^\top$. *The matrix* $CC^\top$ *is positive semidefinite having real and nonnegative eigenvalues. If* $CC^\top$ *is nonsingular then there obviously exists a positive* α *such that* $\rho(A_\alpha) < 1$ *and we are done. In the singular case again because of the nonnegativeness of the eigenvalues we can choose an* α *such that the conditions* $\rho(A_\alpha) \leq 1$ *and* $\texttt{Cond1}(A_\alpha)$ *hold. As* A_α *is symmetric and therefore diagonalisable, it follows that* $\mathcal{E}_\lambda^A = \mathcal{H}_\lambda^A$, *which establishes property* $\texttt{CondE}(A_\alpha)$. *It remains to show that* $Cr \in Im\{I - A_\alpha\}$ *or equivalently that* $Cr = CC^\top x$ *for some* $x \in \mathbb{R}^n$. *Using the pseudo inverse* $(C^\top)^\dagger = C$ *of* $C^\top$ *(cf. (Björck, 1996)) yields* $CC^\top (C^\top)^\dagger = C$. *Therefore,* $x = (C^\top)^\dagger r$ *fulfils the above requirement.* ♢

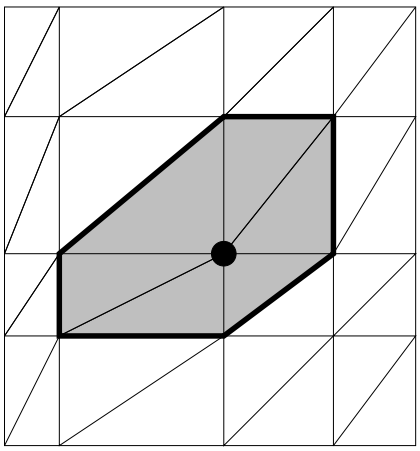

Figure 1. Support of a hat function on a two-dimensional grid with Kuhn triangulation.

3.2. Uniform RL Algorithm

In (Schoknecht & Merke, 2003) it was shown that a whole class of synchronous RL algorithms is generated by replacing C in (14) with some other matrix which depends on the x_i and z_i. This class contains the Kaczmarcz, RG, nearest neighbour and the uniform RL algorithm. In (Merke & Schoknecht, 2002) it was shown that only the RG algorithm and the uniform RL algorithm converge for single transitions.

It has been an open question if the uniform RL algorithm would also converge in the case of multiple synchronous transitions. In this section we will use theorem 1 to generate counterexamples which violate a necessary condition of convergence. This shows that the uniform RL algorithm generally diverges for more than one transition.

The uniform RL algorithm is applicable for linear grid based function approximators. This kind of function approximator assumes a d-dimensional grid and a triangulation into simplices given. In figure 1 we see an example of a two-dimensional grid with a Kuhn triangulation. The representable functions are linear on the simplices and continuous on the simplex boundaries. They can be described as a linear combination of generalised hat functions.

In the example in figure 1 the support of a two-dimensional hat function is shaded. The black dot is the top of the hat function, where it attains the value 1. At the boundaries of the support the hat function vanishes, and on each simplex it is linear. These conditions uniquely define the hat function centred on the black dot. The set of all hat functions $\{\varphi_1, \ldots, \varphi_F\}$ corresponding to the F grid nodes is

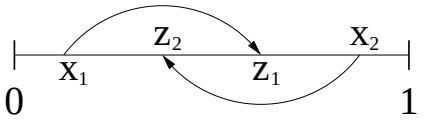

Figure 2. Two transitions $x_i \rightarrow z_i$ on a one-dimensional simplex

a basis of the space of representable functions, i.e. $f(x) = \sum_{i=1}^{F} \varphi_i(x) w_i$. The feature $\varphi_i(x)$, which is equivalent to the value of the i-th hat function at point x, determines the weight that is given to grid node i. The vector $\varphi(x) = (\varphi_1(x), \ldots, \varphi_F)^\top$ contains the barycentric coordinates of x. It satisfies $0 \leq \varphi_i(x) \leq 1$ and $\sum_{i=1}^{F} \varphi_i(x) = 1$.

If we now consider iteration (13) then we see that the magnitude of the change of the components i of $w(k)$ depends on the feature $\varphi_i(x)$. If x is near the centre of the i-th hat function, then the change will be greater than if x is near the boundary of the support of the hat function. For x outside the support of the i-th hat function $\varphi_i(x)$ will be zero, which reflects the local character of grid approximators. The idea behind the uniform update rule is to replace $c = \varphi(x)$ in (13) by a vector which equally weights all hat functions which have x in its support. Formally we define

$$\psi_i(x) = \begin{cases} \frac{1}{p} & \text{if } \varphi_i(x) \neq 0, \\ 0 & \text{else} \end{cases} \tag{16}$$

where p is the number of nonzero entries in $\varphi(x)$. Because of $\varphi_i(x) \neq 0$ for at least one i this is well defined.

For a single transition (x, z, ρ) we now obtain the uniform RL algorithm

$$\begin{aligned} w(k+1) = (I + \alpha\psi(x)(\gamma\varphi(z) - \varphi(x))^\top) w(k) \\ + \alpha\psi(x)\rho \end{aligned} \tag{17}$$

by replacing $c = \varphi(x)$ in (13) with $c = \psi(x)$. It was shown in (Merke & Schoknecht, 2002) that the iteration (17) converges for suitable choices of α. As we will show in the following, this is no longer true in the case of *multiple* transition updates. Our search for a divergent counterexample with two transitions will be guided by theorem 1. The objective is to find two transitions such that a necessary condition of convergence is violated. In this case the iteration diverges. For the sake of simplicity we consider the case of a one-dimensional simplex with two transitions as depicted in figure 2 The grid consists of just one simplex,

namely the interval $[0,1]$. For each $x \in [0,1]$ the feature vector contains the barycentric coordinates of x with respect the boundary points 0 and 1, and therefore $\varphi(x) = (1-x, x)^\top$.

The iteration (17) generalises in the same way to the case of arbitrary multiple transitions as iteration (14) was derived from (13). Thus, a two transitions iterations for the uniform RL algorithm can be written as

$$w(k+1) = (I + \alpha C D^\top) w(k) + \alpha C r \qquad (18)$$

with

$$C = [\psi(x_1)|\psi(x_2)], \quad r = (\rho_1, \rho_2)$$

and

$$\begin{aligned} D^\top &= [\gamma\varphi(z_1) - \varphi(x_1) \mid \gamma\varphi(z_2) - \varphi(x_2)]^\top \\ &= \begin{pmatrix} \gamma(1-z_1) - (1-x_1) & \gamma z_1 - x_1 \\ \gamma(1-z_2) - (1-x_2) & \gamma z_2 - x_2 \end{pmatrix} \end{aligned}$$

According to iteration (18) the matrix A and the vector b in theorem 1 are given by $A = I + \alpha C D^\top$ and $b = Cr$. Let us assume that $x_1, x_2, z_1, z_2 \in [0,1]$ as depicted in figure 2. In the following we construct a counterexample that violates the necessary condition of convergence $b \in \mathrm{Im}\{I - A\}$. We set $x_1 = z_1 = 0$, $x_2 = \gamma z_2$ and $z_2 \in (0,1)$. Then we have $\varphi(x_1) = (1,0)^\top$. According to (16) $\psi(x_1) = (1,0)^\top$ because only one entry of $\varphi(x_1)$ is nonzero. With $\psi(x_2) = (\frac{1}{2}, \frac{1}{2})^\top$ this yields

$$C = \begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix}, \quad D^\top = \begin{pmatrix} \gamma - 1 & 0 \\ \gamma - 1 & 0 \end{pmatrix}$$

and

$$C D^\top = \frac{1}{2}(\gamma - 1) \begin{pmatrix} 3 & 0 \\ 1 & 0 \end{pmatrix}$$

Thus, $\mathrm{Im}\{I - A\} = \mathrm{Im}\{C D^\top\} = [(3,1)^\top]$, where $[v]$ denotes the linear subspace spanned by the vector v. We must now choose $r = (\rho_1, \rho_2)$ such that $b = Cr = (\rho_1 + \frac{1}{2}\rho_2, \frac{1}{2}\rho_2)^\top \notin [(3,1)^\top]$. This is equivalent to $\rho_1 + \frac{1}{2}\rho_2 \neq \frac{3}{2}\rho_2$ which is equivalent to $\rho_1 \neq \rho_2$. Thus for an arbitrary choice of ρ_1 and ρ_2 with $\rho_1 \neq \rho_2$ condition $b \in \mathrm{Im}\{I - A\}$ is violated and the iteration (18) diverges for all initial values $w(0)$.

4. Conclusions

We have proved necessary and sufficient conditions of convergence for a general matrix iteration. The update rules in a unified framework for synchronous reinforcement learning (RL) algorithms with linear function approximation can be seen as a special case of this iteration. Therefore, our results can be used to prove either convergence or divergence of the different algorithms in this framework. We have applied the new theorem to prove convergence of the synchronous residual gradient algorithm (Baird, 1995). Moreover, we have addressed the unresolved problem if the uniform RL algorithm (Merke & Schoknecht, 2002) converges for multiple arbitrary transitions. Our theorem allows to construct a counterexample for which this algorithm diverges. Therefore, the residual gradient algorithm remains the only RL algorithm for which convergence has been shown in the case of multiple transitions and linear function approximation.

References

Baird, L. C. (1995). Residual algorithms: Reinforcement learning with function approximation. *ICML*.

Bertsekas, D. P., & Tsitsiklis, J. N. (1996). *Neuro-dynamic programming*. Athena Scientific.

Björck, Å. (1996). *Numerical methods for least squares problems*. SIAM.

Elaydi, S. N. (1999). *An introduction to difference equations*. Springer. 2 edition.

Gohberg, I., Lancaster, P., & Rodman, L. (1986). *Invariant subspaces of matrices with applications*. A Wiley-Interscience publication. Wiley.

Greenbaum, A. (1997). *Iterative methods for solving linear systems*, vol. 17 of *Frontiers in Applied Mathematics*. SIAM.

Koller, D., & Parr, R. (2000). Policy iteration for factored mdps. *UAI*.

Ludyk, G. (1985). *Stability of time-variant discrete-time systems*. Vieweg.

Merke, A., & Schoknecht, R. (2002). A necessary condition of convergence for reinforcement learning with function approximation. *ICML* (pp. 411–418).

Schoknecht, R. (2003). Optimality of reinforcement learning algorithms with linear function approximation. *NIPS 15* (pp. 1555–1562). MIT Press.

Schoknecht, R., & Merke, A. (2003). Convergent combinations of reinforcement learning with linear function approximation. *NIPS 15* (pp. 1579–1586). MIT Press.

Sutton, R. S. (1988). Learning to predict by the methods of temporal differences. *Machine Learning, 3*.

Tsitsiklis, J. N., & Van Roy, B. (1997). An analysis of temporal-difference learning with function approximation. *IEEE Transactions on Automatic Control*.