Projection Operator in Adaptive Systems

Eugene Lavretsky* Travis E. Gibson† and Anuradha M. Annaswamy‡

October 18, 2012

Abstract

The projection algorithm is frequently used in adaptive control and this note presents a detailed analysis of its properties.

1 Introduction

These notes started in [2] as a personal communication from Eugene to colleagues in the field of adaptive control and summarized results from [5, 3, 1, 4]. Properties of the projection operator are explored in detail in the following section.

2 Properties of Convex Sets and Functions

Definition 1. A set $E \subset \mathbb{R}^k$ is *convex* if

$$\lambda x + (1-\lambda)y \in E$$

whenever $x \in E$, $y \in E$, and $0 \leq \lambda \leq 1$

Remark. Essentially, a convex set has the following property. For any two points $x, y \in E$ where E is convex, all the points on the connecting line from x to y are also in E.

Definition 2. A function $f : \mathbb{R}^k \to \mathbb{R}$ is *convex* if

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y)$$

$\forall 0 \leq \lambda \leq 1$.

*The Boeing Company, Huntington Beach CA, USA

†Department of Mechanical Engineering, Massachusetts Institute of Technology,Cambridge MA 02139 USA

‡Department of Mechanical Engineering, Massachusetts Institute of Technology,Cambridge MA 02139 USA

Lemma 3. *Let $f(\theta) : \mathbb{R}^k \to \mathbb{R}$ be a convex function. Then for any constant $\delta > 0$ the subset $\Omega_\delta = \{\theta \in \mathbb{R}^k | f(\theta) \le \delta\}$ is convex.*

Proof. Let $\theta_1, \theta_2 \in \Omega_\delta$. Then $f(\theta_1) \le \delta$ and $f(\theta_2) \le \delta$. Since $f(x)$ is convex then for any $0 \le \lambda \le 1$

$$f\big(\underbrace{\lambda\theta_1 + (1-\lambda)\theta_2}_{\theta}\big) \le \lambda \underbrace{f(\theta_1)}_{\le\delta} + (1-\lambda) \underbrace{f(\theta_2)}_{\le\delta} \le \lambda\delta + (1-\lambda)\delta = \delta$$

$\therefore f(\theta) \le \delta$, thus, $\theta \in \Omega_\delta$. □

Lemma 4. *Let $f(\theta) : \mathbb{R}^k \to \mathbb{R}$ be a continuously differentiable convex function. Choose a constant $\delta > 0$ and consider $\Omega_\delta = \{\theta \in \mathbb{R}^k | f(\theta) \le \delta\} \subset \mathbb{R}$. Let θ^* be an interior point of Ω_δ, i.e. $f(\theta^*) < \delta$. Choose θ_b as a boundary point so that $f(\theta_b) = \delta$. Then the following holds:*

$$(\theta^* - \theta_b)^T \nabla f(\theta_b) \le 0 \tag{1}$$

where $\nabla f(\theta_b) = \left(\frac{\partial f(\theta)}{\partial \theta_1} \cdots \frac{\partial f(\theta)}{\partial \theta_k} \right)^T$ evaluated at θ_b.

Proof. $f(\theta)$ is convex $\therefore$

$$f\left(\lambda\theta^* + (1-\lambda)\theta_b\right) \le \lambda f(\theta^*) + (1-\lambda) f(\theta_b)$$

equivalently,

$$f\left(\theta_b + \lambda(\theta^* - \theta_b)\right) \le f(\theta_b) + \lambda\left(f(\theta^*) - f(\theta_b)\right)$$

For any $0 < \lambda \le 1$:

$$\frac{f\left(\theta_b + \lambda(\theta^* - \theta_b)\right) - f(\theta_b)}{\lambda} \le f(\theta^*) - f(\theta_b) \le \delta - \delta = 0$$

and taking the limit as $\lambda \to 0$ yields (1). □

3 Projection

Definition 5. The *Projection Operator* for two vectors $\theta, y \in \mathbb{R}^k$ is now introduced as

$$\mathrm{Proj}(\theta, y, f) = \begin{cases} y - \frac{\nabla f(\theta)(\nabla f(\theta))^T}{\|\nabla f(\theta)\|^2} y f(\theta) & \text{if } f(\theta) > 0 \wedge y^T \nabla f(\theta) > 0 \\ y & \text{otherwise.} \end{cases} \tag{2}$$

where $f : \mathbb{R}^k \to \mathbb{R}$ is a convex function and $\nabla f(\theta) = \left(\frac{\partial f(\theta)}{\partial \theta_1} \cdots \frac{\partial f(\theta)}{\partial \theta_k} \right)^T$. Note that the following are notationally equivalent $\mathrm{Proj}(\theta, y) = \mathrm{Proj}(\theta, y, f)$ when the exact structure of the convex function f is of no importance.

Remark. A geometrical interpretation of (2) follows. Define a convex set Ω_0 as

$$\Omega_0 \triangleq \left\{\theta \in \mathbb{R}^k | f(\theta) \leq 0\right\} \tag{3}$$

and let Ω_1 represent another convex set such that

$$\Omega_1 \triangleq \left\{\theta \in \mathbb{R}^k | f(\theta) \leq 1\right\} \tag{4}$$

From (3) and (4) $\Omega_0 \subset \Omega_1$. From the definition of the projection operator in (7) θ is not modified when $\theta \in \Omega_0$. Let

$$\Omega_{\mathcal{A}} \triangleq \Omega_1 \backslash \Omega_0 = \left\{\theta | 0 < f(\theta) \leq 1\right\}$$

represent an annulus region. Within $\Omega_{\mathcal{A}}$ the projection algorithm subtracts a scaled component of y that is normal to boundary $\left\{\theta | f(\theta) = \lambda\right\}$. When $\lambda = 0$, the scaled normal component is 0, and when $\lambda = 1$, the component of y that is normal to the boundary Ω_1 is entirely subtracted from y, so that $\text{Proj}(\theta, y, f)$ is tangent to the boundary $\left\{\theta | f(\theta) = 1\right\}$. This discussion is visualized in Figure 1.

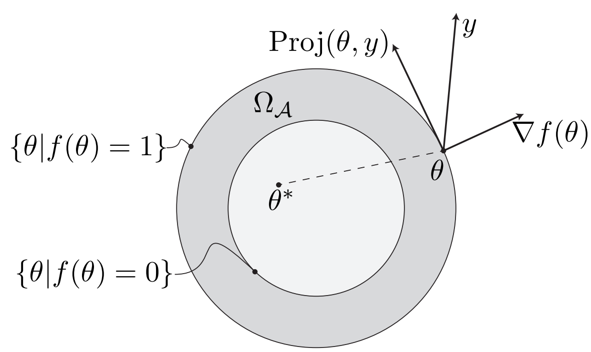

Figure 1: Visualization of Projection Operator in $\mathbb{R}^2$.

Remark. Note that $(\nabla f(\theta))^T \text{Proj}(\theta, y) = 0 \forall \theta$ when $f(\theta) = 1$ and that the general structure of the algorithm is as follows

$$\text{Proj}(\theta, y) = y - \alpha(t) \nabla f(\theta) \tag{5}$$

for some time varying α when the modification is triggered. Multiplying the left hand side of the equation by $(\nabla f(\theta))^T$ and solving for α one finds that

$$\alpha(t) = \left((\nabla f(\theta))^T \nabla f(\theta)\right)^{-1} (\nabla f(\theta))^T y \tag{6}$$

and thus the algorithm takes the form

$$\text{Proj}(\theta, y) = y - \nabla f(\theta) \left((\nabla f(\theta))^T \nabla f(\theta)\right)^{-1} (\nabla f(\theta))^T y f(\theta) \tag{7}$$

where the modification is active. Notice that the $f(\theta)$ has been added to the definition, making (7) continuous.

Lemma 6. *One important property of the projection operator follows. Given $\theta^* \in \Omega_0$,*

$$(\theta - \theta^*)^T (Proj(\theta, y, f) - y) \leq 0. \tag{8}$$

Proof. Note that

$$(\theta - \theta^*)^T(\text{Proj}(\theta, y, f) - y) = (\theta^* - \theta)^T(y - \text{Proj}(\theta, y, f))$$

If $f(\theta) > 0 \wedge y^T \nabla f(\theta) > 0$, then

$$(\theta^* - \theta)^T \left(y - \left(y - \frac{\nabla f(\theta)(\nabla f(\theta))^T}{\|\nabla f(\theta)\|^2} y f(\theta) \right) \right)$$

and using Lemma 4

$$\frac{\underbrace{(\theta^* - \theta)^T \nabla f(\theta)}_{\leq 0} \underbrace{(\nabla f(\theta))^T y}_{>0}}{\|\nabla f(\theta)\|^2} \underbrace{f(\theta)}_{\geq 0} \leq 0$$

otherwise $\text{Proj}(\theta, y, f) = y$. □

Definition 7 (Projection Operator)**.** The general form of the projection operator is the $n \times m$ matrix extension to the vector definition above.

$$\text{Proj}(\Theta, Y, F) = [\text{Proj}(\theta_1, y_1, f_1) \ \ldots \ \text{Proj}(\theta_m, y_m, y_m)]$$

where $\Theta = [\theta_1 \ \ldots \ \theta_m] \in \mathbb{R}^{n \times m}, Y = [y_1 \ \ldots \ y_m] \in \mathbb{R}^{n \times m}$, and $F = [f_1(\theta_1) \ \ldots \ f_m(\theta_m)]^T \in \mathbb{R}^{m \times 1}$. Recalling (2)

$$\text{Proj}(\theta_j, y_j, f_j) = \begin{cases} y_j - \frac{\nabla f_j(\theta_j)(\nabla f_j(\theta_j))^T}{\|\nabla f_j(\theta_j)\|^2} y_j f_j(\theta_j) & \text{if } f_j(\theta_j) > 0 \wedge y_j^T \nabla f_j(\theta_j) > 0 \\ y_j & \text{otherwise} \end{cases}$$

$j = 1$ to m.

Lemma 8. *Let $F = [f_1 \ \ldots \ f_m]^T \in \mathbb{R}^{m \times 1}$ be a convex vector function and $\hat{\Theta} = [\hat{\theta}_1 \ \ldots \ \hat{\theta}_m], \Theta = [\theta_1 \ \ldots \ \theta_m], Y = [y_1 \ \ldots \ y_m]$ where $\hat{\Theta}, \Theta, Y \in \mathbb{R}^{n \times m}$ then,*

$$\mathit{trace}\left\{ \left(\hat{\Theta} - \Theta\right)^T \left(\mathit{Proj}(\hat{\Theta}, Y, F) - Y\right) \right\} \leq 0.$$

Proof. Using (8),

$$\text{trace}\left\{ \left(\hat{\Theta} - \Theta\right)^T \left(\text{Proj}(\hat{\Theta}, Y, F) - Y\right) \right\} = \sum_{j=1}^{m} (\hat{\theta}_j - \theta_j)^T (\text{Proj}(\hat{\theta}_j, y_j, f_j) - y_j)$$
$$\leq 0. \square$$

The application of the projection algorithm in adaptive control is explored below.

Lemma 9. *If an initial value problem, i.e. adaptive control algorithm with adaptive law and initial conditions, is defined by:*

1. $\dot{\theta} = Proj(\theta, y, f)$
2. $\theta(t = 0) = \theta_0 \in \Omega_1 = \{\theta \in \mathbb{R}^k | f(\theta) \leq 1\}$

3. $f(\theta) : \mathbb{R}^k \to \mathbb{R}$ *is convex*

Then $\theta(t) \in \Omega_1 \forall t \geq 0$.

Proof. Taking the time derivative of the convex function

$$\dot{f}(\theta) = (\nabla f(\theta))^T \dot{\theta} = (\nabla f(\theta))^T \mathrm{Proj}(\theta, y, f) \tag{9}$$

Substitution of (9) into (2) leads to

$$\begin{aligned}\dot{f}(\theta) &= (\nabla f(\theta))^T \mathrm{Proj}(\theta, y, f) \\ &= \begin{cases} (\nabla f(\theta))^T y(1 - f(\theta)) & \text{if } f(\theta) > 0 \wedge y^T \nabla f(\theta) > 0 \\ (\nabla f(\theta))^T y & \text{if } f(\theta) \leq 0 \vee y^T \nabla f(\theta) \leq 0 \end{cases}\end{aligned}$$

therefore

$$\begin{cases} \dot{f}(\theta) > 0 & \text{if } 0 < f(\theta) < 1 \wedge y^T \nabla f(\theta) > 0 \\ \dot{f}(\theta) = 0 & \text{if } f(\theta) = 1 \wedge y^T \nabla f(\theta) > 0 \\ \dot{f}(\theta) < 0 & \text{if } f(\theta) \leq 0 \vee y^T \nabla f(\theta) \leq 0 \end{cases}.$$

Thus $f(\theta_0) \leq 1 \Rightarrow f(\theta) \leq 1 \forall t \geq 0$. □

Remark. Given $\theta_0 \in \Omega_0$, θ may increase up to the boundary where $f(\theta) = 1$. However, θ never leaves the convex set Ω_1.

Example 10 (Projection Algorithm in Adaptive Control Law)**.** Let $\Theta(t) : \mathbb{R}^+ \to \mathbb{R}^{m \times n}$ represent a time varying feedback gain in a dynamical system. This feedback gain is implemented as:

$$u = \Theta(t)^T x$$

where $u \in \mathbb{R}^n$ represents the control input and $x \in \mathbb{R}^m$ the state vector. The time varying feedback gain is adjusted using the following adaptive law

$$\dot{\Theta} = \mathrm{Proj}(\Theta, -xe^T PB, F)$$

where $e \in \mathbb{R}^m$ is an error signal in the state vector space, $P \in \mathbb{R}^{m \times m}$ is a square matrix derived from a Lyapunov relationship and $B \in \mathbb{R}^{m \times n}$ is the input Jacobian for the LTI system to be controlled and $F(\Theta) = [f_1(\theta_1) \ \ldots \ f_m(\theta_m))]^T$. The projection algorithm operates with the family of convex functions

$$f(\theta; \vartheta, \varepsilon) = \frac{\|\theta\|^2 - \vartheta^2}{2\varepsilon\vartheta + \varepsilon^2}.$$

Then, the components of the convex vector function F are chosen as

$$f_i(\theta_i) = f(\theta_i; \vartheta_i, \varepsilon_i). \tag{10}$$

Each i–th component of F is associated with two constant scalar quantities ϑ_i and ε_i. From (10), $f_i(\theta_i) = 0$ when $\|\theta_i\| = \vartheta_i$, and $f_i(\theta_i) = 1$ when $\|\theta_i\| = \vartheta_i + \varepsilon_i$. If the initial condition for Θ is such that $\Theta(t = 0) \in \Theta_0 = [\theta_{0,1} \ \ldots \ \theta_{0,m}]$ where $\{\theta_{0,i} | f_i(\theta_i) \leq 0 \, i = 1 \text{ to } m\}$, then each θ_i satisfies all three conditions for Lemma 9. Thus $\|\theta_i(t)\| \leq \vartheta_i + \epsilon_i \forall t \geq 0$.

4 Γ–Projection

Definition 11. A variant of the projection algorithm, Γ–projection, updates the parameter along a symmetric positive definite gain Γ as defined below

$$\mathrm{Proj}_\Gamma(\theta, y, f) = \begin{cases} \Gamma y - \Gamma \frac{\nabla f(\theta)(\nabla f(\theta))^T}{(\nabla f(\theta))^T \Gamma \nabla f(\theta)} \Gamma y f(\theta) & \text{if } f(\theta) > 0 \wedge y^T \Gamma \nabla f(\theta) > 0 \\ \Gamma y & \text{otherwise.} \end{cases} \tag{11}$$

This method was first introduced in [1].

Lemma 12. *Given* $\theta^* \in \Omega_0$*,*

$$(\theta - \theta^*)^T (\Gamma^{-1} Proj_\Gamma(\theta, y, f) - y) \leq 0. \tag{12}$$

Proof. If $f(\theta) > 0 \wedge y^T \Gamma \nabla f(\theta) > 0$, then

$$(\theta^* - \theta)^T \left(y - \Gamma^{-1} \left(\Gamma y - \Gamma \frac{\nabla f(\theta)(\nabla f(\theta))^T}{(\nabla f(\theta))^T \Gamma \nabla f(\theta)} \Gamma y f(\theta) \right) \right)$$

and using Lemma 4

$$\frac{\underbrace{(\theta^* - \theta)^T \nabla f(\theta)}_{\leq 0} \underbrace{(\nabla f(\theta))^T \Gamma y}_{> 0}}{(\nabla f(\theta))^T \Gamma \nabla f(\theta)} \underbrace{f(\theta)}_{\geq 0} \leq 0$$

otherwise $\mathrm{Proj}_\Gamma(\theta, y, f) = \Gamma y$. □

References

[1] Petros A. Ioannou and Jing Sun. *Robust Adaptive Control*. Prentice-Hall, 1996.

[2] E. Lavretsky. The projection operator. In *Personal Notes*, 2006.

[3] J. Pomet and L. Praly. Adaptive nonlinear regulation: Estimation from the lyapunov equation. *IEEE Transactions on Automatic Control*, 37(6), June 1992.

[4] S. Seshagiri and Hassan K. Khalil. Adaptive nonlinear regulation: Output feedback control of nonlinear systems using rbf neural networks. *IEEE Transactions on Automatic Control*, 11(1), January 2000.

[5] J.-J. E. Slotine and J. A. Coetsee. Adaptive sliding controller synthesis for non-linear systems. *International Journal of Control*, 43(6):1631–1651, 1986.