Parameter Estimation II

Contents

- Vector Case
- Gradient-based Algorithms
- Least-square Algorithms

Generalized SPM

Consider a LTI SISO system

$$y = G(s)u, \quad G(s) = \frac{Z(s)}{R(s)} = k_p \frac{\bar{Z}(s)}{R(s)} \tag{1}$$

with

$$\begin{aligned} R(s) &= s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0 \\ Z(s) &= b_m s^m + \cdots + b_1 s + b_0 \end{aligned}$$

and $k_p = b_m$ is a.k.a. high-frequency gain. Express the system as an nth-order differential equation, we obtain

$$y^{(n)} + a_{n-1}y^{(n-1)} + \cdots + a_1\dot{y} + a_0 y = b_m u^{(m)} + \cdots + b_1\dot{u} + b_0 u.$$

Generalized SPM

Consider a LTI SISO system

$$y = G(s)u, \quad G(s) = \frac{Z(s)}{R(s)} = k_p \frac{\bar{Z}(s)}{R(s)} \tag{1}$$

with

$$\begin{aligned} R(s) &= s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0 \\ Z(s) &= b_m s^m + \cdots + b_1 s + b_0 \end{aligned}$$

and $k_p = b_m$ is a.k.a. high-frequency gain. Express the system as an nth-order differential equation, we obtain

$$y^{(n)} + a_{n-1}y^{(n-1)} + \cdots + a_1\dot{y} + a_0 y = b_m u^{(m)} + \cdots + b_1\dot{u} + b_0 u.$$

Generalized SPM

Lumping all the parameters in the vector and filtering both sides with $\frac{1}{\Lambda(s)}$ with $\Lambda(s) = s^n + \lambda_{n-1}s^{n-1} + \cdots + \lambda_1 s + \lambda_0$ is a monic Hurwitz polynomial, we obtain the parametric model

$$z = \theta^{*\top}\phi$$

where

$$z = \frac{1}{\Lambda(s)}y^{(n)} = \frac{s^n}{\Lambda(s)}y$$

$$\theta^* = [b_m, \ldots, b_0, a_{n-1}, \ldots, a_0]^T \in \mathcal{R}^{n+m+1}$$

$$\phi = \left[\frac{s^m}{\Lambda(s)}u, \ldots, \frac{1}{\Lambda(s)}u, -\frac{s^{n-1}}{\Lambda(s)}y, \ldots, -\frac{1}{\Lambda(s)}y\right]^T$$

Generalized SPM

The objective is to process the signals z and ϕ in order to generate an estimate $\theta(t)$ for θ^* at each time t, as follows

$$\dot{\theta} = \Phi(z, \phi)$$

Different choices of $\Phi(\cdot)$ lead to a wide class of adaptive laws with, sometimes, different convergence properties, as demonstrated in the following lectures.

Questions:

1. What if a_0, a_1, b_m are known?

2. If we use the previous gradient-based method, What kind of u shall we choose to ensure the exponential convergence?

Generalized SPM

The objective is to process the signals z and ϕ in order to generate an estimate $\theta(t)$ for θ^* at each time t, as follows

$$\dot{\theta} = \Phi(z, \phi)$$

Different choices of $\Phi(\cdot)$ lead to a wide class of adaptive laws with, sometimes, different convergence properties, as demonstrated in the following lectures.

Questions:

1. What if a_0, a_1, b_m are known?

2. If we use the previous gradient-based method, What kind of u shall we choose to ensure the exponential convergence?

Generalized SPM

The objective is to process the signals z and ϕ in order to generate an estimate $\theta(t)$ for θ^* at each time t, as follows

$$\dot{\theta} = \Phi(z, \phi)$$

Different choices of $\Phi(\cdot)$ lead to a wide class of adaptive laws with, sometimes, different convergence properties, as demonstrated in the following lectures.

Questions:

1. What if a_0, a_1, b_m are known?

2. If we use the previous gradient-based method, What kind of u shall we choose to ensure the exponential convergence?

Contents

Gradient-based Algorithms

Start with SPM

$$z = \theta^{*\top}\phi,$$

with $\theta^* \in \mathbb{R}^n, \phi \in \mathbb{R}^n$ and the estimation error is constructed as

$$\varepsilon = \frac{z - \hat{z}}{m_s^2} = \frac{z - \theta^T\phi}{m_s^2}$$

where m_s^2 is the normalizing signal. The gradient algorithm is developed by using the gradient method to minimize some appropriate functional $J(\theta)$.

$$\dot{\theta} = -\Gamma\nabla J$$

Different choices for $J(\theta)$ lead to different algorithms.

Gradient-based Algorithms

Start with SPM

$$z = \theta^{*\top}\phi,$$

with $\theta^* \in \mathbb{R}^n, \phi \in \mathbb{R}^n$ and the estimation error is constructed as

$$\varepsilon = \frac{z - \hat{z}}{m_s^2} = \frac{z - \theta^T\phi}{m_s^2}$$

where m_s^2 is the normalizing signal.The gradient algorithm is developed by using the gradient method to minimize some appropriate functional $J(\theta)$.

$$\dot{\theta} = -\Gamma\nabla J$$

Different choices for $J(\theta)$ lead to different algorithms.

Gradient-based Algorithms

1. Instantaneous cost function

$$J(\theta) = \frac{\varepsilon^2 m_s^2}{2} = \frac{\left(z - \theta^\top \phi\right)^2}{2m_s^2}$$

Adaptive Law for $\theta(t)$:

$$\dot{\theta} = \Gamma \varepsilon \phi, \quad \theta(0) = \theta_0$$

guarantees the following properties:

- $\varepsilon, \varepsilon m_s, \dot{\theta} \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\theta \in \mathcal{L}_\infty$
- If there exists $T_0 > 0$ such that $\frac{\phi}{m_s}$ is P.E. with level of α_0, then $\theta(t)$ converges to θ^* exponentially fast.

Gradient-based Algorithms

1. Instantaneous cost function

$$J(\theta) = \frac{\varepsilon^2 m_s^2}{2} = \frac{\left(z - \theta^\top \phi\right)^2}{2m_s^2}$$

Adaptive Law for $\theta(t)$:

$$\dot{\theta} = \Gamma \varepsilon \phi, \quad \theta(0) = \theta_0$$

guarantees the following properties:

- $\varepsilon, \varepsilon m_s, \dot{\theta} \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\theta \in \mathcal{L}_\infty$
- If there exists $T_0 > 0$ such that $\frac{\phi}{m_s}$ is P.E. with level of α_0, then $\theta(t)$ converges to θ^* exponentially fast.

Gradient-based Algorithms

1. Instantaneous cost function

$$J(\theta) = \frac{\varepsilon^2 m_s^2}{2} = \frac{\left(z - \theta^\top \phi\right)^2}{2m_s^2}$$

Adaptive Law for $\theta(t)$:

$$\dot{\theta} = \Gamma \varepsilon \phi, \quad \theta(0) = \theta_0$$

guarantees the following properties:

- $\varepsilon, \varepsilon m_s, \dot{\theta} \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\theta \in \mathcal{L}_\infty$
- If there exists $T_0 > 0$ such that $\frac{\phi}{m_s}$ is P.E. with level of α_0, then $\theta(t)$ converges to θ^* exponentially fast.

Gradient-based Algorithms

In addition, for all $t \geq nT_0$, $n = 0, 1...$, it holds

$$(\theta(t) - \theta^*)^\top \Gamma^{-1} (\theta(t) - \theta^*) \leq (1 - \gamma_1)^n (\theta(0) - \theta^*)^\top \Gamma^{-1} (\theta(0) - \theta^*)$$

with $\gamma_1 = \frac{2\alpha_0 T_0 \lambda_{\min}(\Gamma)}{2+\beta^4 \lambda_{\max}^2(\Gamma) T_0^2}$, $\beta = \sup_t \left| \frac{\phi}{m_s} \right|$.

Furthermore, if the regressor signal ϕ is of the form

$$\phi = H(s)u$$

with $H(s) = \left[\frac{s^m}{\Lambda(s)}, \ldots, \frac{1}{\Lambda(s)}, -\frac{s^{n-1}G(s)}{\Lambda(s)}, \ldots, -\frac{G(s)}{\Lambda(s)} \right]^\top$ and $G(s)$ is Hurwitz defined in (1) and has no pole-zero cancellation, then given a sufficiently rich u signal of order $n + m + 1$, we have ϕ and $\frac{\phi}{m_s}$ are P.E., $\tilde{\theta}, \varepsilon, \dot{\theta}$ all converge to zero exponentially fast.

Gradient-based Algorithms

In addition, for all $t \geq nT_0$, $n = 0, 1...$, it holds

$$(\theta(t) - \theta^*)^\top \Gamma^{-1} (\theta(t) - \theta^*) \leq (1 - \gamma_1)^n (\theta(0) - \theta^*)^\top \Gamma^{-1} (\theta(0) - \theta^*)$$

with $\gamma_1 = \frac{2\alpha_0 T_0 \lambda_{\min}(\Gamma)}{2+\beta^4 \lambda_{\max}^2(\Gamma) T_0^2}$, $\beta = \sup_t \left| \frac{\phi}{m_s} \right|$.

Furthermore, if the regressor signal ϕ is of the form

$$\phi = H(s)u$$

with $H(s) = \left[\frac{s^m}{\Lambda(s)}, \ldots, \frac{1}{\Lambda(s)}, -\frac{s^{n-1}G(s)}{\Lambda(s)}, \ldots, -\frac{G(s)}{\Lambda(s)} \right]^\top$ and $G(s)$ is Hurwitz defined in (1) and has no pole-zero cancellation, then given a sufficiently rich u signal of order $n + m + 1$, we have ϕ and $\frac{\phi}{m_s}$ are P.E., $\tilde{\theta}, \varepsilon, \dot{\theta}$ all converge to zero exponentially fast.

Remark:

The rate of convergence can be improved if we choose the design parameters so that $1 - \gamma_1$ is as small as possible. Examining the expression for γ_1

$$\gamma_1 = \frac{2\alpha_0 T_0 \lambda_{\min}(\Gamma)}{2 + \beta^4 \lambda_{\max}^2(\Gamma) T_0^2}$$

The only free design parameter is the adaptive gain matrix Γ.

However, very small or very large values of Γ lead to slower convergence rates. In general, the convergence rate depends on the signal input and filters used in addition to Γ in a way that is not understood quantitatively.

Remark:

The rate of convergence can be improved if we choose the design parameters so that $1 - \gamma_1$ *is as small as possible. Examining the expression for* γ_1

$$\gamma_1 = \frac{2\alpha_0 T_0 \lambda_{\min}(\Gamma)}{2 + \beta^4 \lambda_{\max}^2(\Gamma) T_0^2}$$

The only free design parameter is the adaptive gain matrix Γ*. However, very small or very large values of* Γ *lead to slower convergence rates. In general, the convergence rate depends on the signal input and filters used in addition to* Γ *in a way that is not understood quantitatively.*

Gradient-based algorithms

2. Integral cost function

$$J(\theta) = \frac{1}{2}\int_0^t e^{-\beta(t-\tau)}\varepsilon^2(t,\tau)m_s^2(\tau)d\tau$$

where $\beta > 0$ is a design constant acting as a forgetting factor and

$$\varepsilon(t,\tau) = \frac{z(\tau) - \theta^T(t)\phi(\tau)}{m_s^2(\tau)}, \quad \varepsilon(t,t) = \varepsilon, \quad \tau \le t$$

Remark:

- The forgetting factor $e^{-\beta(t-\tau)}$ is used to put more weight on recent data by discounting the earlier ones.
- It is clear that J is a convex function of θ at each time t

Gradient-based algorithms

2. Integral cost function

$$J(\theta) = \frac{1}{2}\int_0^t e^{-\beta(t-\tau)}\varepsilon^2(t,\tau)m_s^2(\tau)d\tau$$

where $\beta > 0$ is a design constant acting as a forgetting factor and

$$\varepsilon(t,\tau) = \frac{z(\tau) - \theta^T(t)\phi(\tau)}{m_s^2(\tau)}, \quad \varepsilon(t,t) = \varepsilon, \quad \tau \leq t$$

Remark:

- The forgetting factor $e^{-\beta(t-\tau)}$ is used to put more weight on recent data by discounting the earlier ones.
- It is clear that J is a convex function of θ at each time t

Gradient-based algorithms

Applying the gradient method, we have the adaptive law

$$\dot{\theta} = \Gamma \int_0^t e^{-\beta(t-\tau)} \frac{z(\tau) - \theta^\top(t)\phi(\tau)}{m_s^2(\tau)} \phi(\tau) d\tau$$

this can be implemented as

$$\begin{aligned}
\dot{\theta} &= -\Gamma(R(t)\theta + Q(t)), & \theta(0) &= \theta_0 \\
\dot{R} &= -\beta R + \tfrac{\phi\phi^T}{m_s^2}, & R(0) &= \mathbf{0} \in \mathbb{R}^{n\times n} \\
\dot{Q} &= -\beta Q - \tfrac{z\phi}{m_s^2}, & Q(0) &= \mathbf{0} \in \mathbb{R}^{n}
\end{aligned}$$

with n is the dimension of the vector θ^*.

Gradient-based algorithms

Applying the gradient method, we have the adaptive law

$$\dot{\theta} = \Gamma \int_0^t e^{-\beta(t-\tau)} \frac{z(\tau) - \theta^\top(t)\phi(\tau)}{m_s^2(\tau)} \phi(\tau) d\tau$$

this can be implemented as

$$\begin{aligned} \dot{\theta} &= -\Gamma(R(t)\theta + Q(t)), & \theta(0) &= \theta_0 \\ \dot{R} &= -\beta R + \tfrac{\phi\phi^T}{m_s^2}, & R(0) &= \mathbf{0} \in \mathbb{R}^{n\times n} \\ \dot{Q} &= -\beta Q - \tfrac{z\phi}{m_s^2}, & Q(0) &= \mathbf{0} \in \mathbb{R}^n \end{aligned}$$

with n is the dimension of the vector θ^*.

Gradient-based algorithm

Lemma: For SPM, the gradient-based algorithm with integral cost function guarantees that

- $\varepsilon, \varepsilon m_s, \dot{\theta} \in \mathcal{L}_2 \cap \infty$ and $\theta \in \mathcal{L}_\infty$
- $\lim_{t\to\infty} |\dot{\theta}(t)| = 0$
- If $\frac{\phi}{m_s}$ is PE, then $\theta(t) \to \theta^*$ exponentially fast. Furthermore, for $\Gamma = \gamma I$, the rate of convergence increases with γ.

Gradient-based algorithm

Proof Because $\frac{\phi}{m} \in \mathcal{L}_\infty$, it follows that $R, Q \in \mathcal{L}_\infty$ and substituting for $z = \phi^\top \theta^*$ in the differential equation for Q we verify that

$$Q(t) = -\int_0^t e^{-\beta(t-\tau)} \frac{\phi(\tau)\phi^\top(\tau)}{m^2} d\tau\theta^* = -R(t)\theta^*$$

and, therefore,

$$\dot{\theta} = \dot{\tilde{\theta}} = -\Gamma R(t)\tilde{\theta}$$

Considering the Lyapunov-like function

$$V(\tilde{\theta}) = \frac{\tilde{\theta}^\top \Gamma^{-1} \tilde{\theta}}{2}$$

Gradient-based algorithm

Proof whose time derivative along the solution of $\dot{\tilde{\theta}}$ is given by

$$\dot{V} = -\tilde{\theta}^\top R(t)\tilde{\theta}$$

it follows that:

- $\dot{V} \leq 0$ implies $\tilde{\theta}, \theta \in \mathcal{L}_\infty$
- $\left(\tilde{\theta}^\top R\tilde{\theta}\right)^{\frac{1}{2}} = \left|R^{\frac{1}{2}}\tilde{\theta}\right| \in \mathcal{L}_2$.
- From $\varepsilon = -\frac{\tilde{\theta}^\top \phi}{m_s^2}$ and $\tilde{\theta}, \frac{\phi}{m} \in \mathcal{L}_\infty$ we conclude that $\varepsilon, \varepsilon m_s \in \mathcal{L}_\infty$.
- Since $\dot{\theta} = \dot{\tilde{\theta}} = -\Gamma R(t)\tilde{\theta}$, we have $|\dot{\theta}| \leq \|\Gamma R^{\frac{T}{2}}\||R^{\frac{1}{2}}\tilde{\theta}\ |$ which together with $R \in \mathcal{L}_\infty$ and $\left|R^{\frac{1}{2}}\tilde{\theta}\right| \in \mathcal{L}_\infty \cap \mathcal{L}_2$ imply that $\dot{\theta} \in \mathcal{L}_\infty \cap \mathcal{L}_2$.

Gradient-based algorithm

Proof whose time derivative along the solution of $\dot{\tilde{\theta}}$ is given by

$$\dot{V} = -\tilde{\theta}^\top R(t)\tilde{\theta}$$

it follows that:

- $\dot{V} \leq 0$ implies $\tilde{\theta}, \theta \in \mathcal{L}_\infty$
- $\left(\tilde{\theta}^\top R\tilde{\theta}\right)^{\frac{1}{2}} = \left|R^{\frac{1}{2}}\tilde{\theta}\right| \in \mathcal{L}_2$.
- From $\varepsilon = -\frac{\tilde{\theta}^\top \phi}{m_s^2}$ and $\tilde{\theta}, \frac{\phi}{m} \in \mathcal{L}_\infty$ we conclude that $\varepsilon, \varepsilon m_s \in \mathcal{L}_\infty$.
- Since $\dot{\theta} = \dot{\tilde{\theta}} = -\Gamma R(t)\tilde{\theta}$, we have $|\dot{\theta}| \leq \|\Gamma R^{\frac{T}{2}}\| |R^{\frac{1}{2}}\tilde{\theta}\ |$ which together with $R \in \mathcal{L}_\infty$ and $\left|R^{\frac{1}{2}}\tilde{\theta}\right| \in \mathcal{L}_\infty \cap \mathcal{L}_2$ imply that $\dot{\theta} \in \mathcal{L}_\infty \cap \mathcal{L}_2$.

Gradient-based algorithm

Proof whose time derivative along the solution of $\dot{\tilde{\theta}}$ is given by

$$\dot{V} = -\tilde{\theta}^\top R(t)\tilde{\theta}$$

it follows that:

- $\dot{V} \le 0$ implies $\tilde{\theta}, \theta \in \mathcal{L}_\infty$
- $\left(\tilde{\theta}^\top R\tilde{\theta}\right)^{\frac{1}{2}} = \left|R^{\frac{1}{2}}\tilde{\theta}\right| \in \mathcal{L}_2$.
- From $\varepsilon = -\frac{\tilde{\theta}^\top \phi}{m_s^2}$ and $\tilde{\theta}, \frac{\phi}{m} \in \mathcal{L}_\infty$ we conclude that $\varepsilon, \varepsilon m_s \in \mathcal{L}_\infty$.
- Since $\dot{\theta} = \dot{\tilde{\theta}} = -\Gamma R(t)\tilde{\theta}$, we have $|\dot{\theta}| \le \|\Gamma R^{\frac{T}{2}}\| |R^{\frac{1}{2}}\tilde{\theta}\ |$ which together with $R \in \mathcal{L}_\infty$ and $\left|R^{\frac{1}{2}}\tilde{\theta}\right| \in \mathcal{L}_\infty \cap \mathcal{L}_2$ imply that $\dot{\theta} \in \mathcal{L}_\infty \cap \mathcal{L}_2$.

Gradient-based algorithm

Proof whose time derivative along the solution of $\dot{\tilde{\theta}}$ is given by

$$\dot{V} = -\tilde{\theta}^{\top} R(t) \tilde{\theta}$$

it follows that:

- $\dot{V} \leq 0$ implies $\tilde{\theta}, \theta \in \mathcal{L}_\infty$
- $\left(\tilde{\theta}^{\top} R \tilde{\theta}\right)^{\frac{1}{2}} = \left| R^{\frac{1}{2}} \tilde{\theta} \right| \in \mathcal{L}_2$.
- From $\varepsilon = -\frac{\tilde{\theta}^{\top}\phi}{m_s^2}$ and $\tilde{\theta}, \frac{\phi}{m} \in \mathcal{L}_\infty$ we conclude that $\varepsilon, \varepsilon m_s \in \mathcal{L}_\infty$.
- Since $\dot{\theta} = \dot{\tilde{\theta}} = -\Gamma R(t) \tilde{\theta}$, we have $|\dot{\theta}| \leq \|\Gamma R^{\frac{T}{2}}\| |R^{\frac{1}{2}} \tilde{\theta}\ |$ which together with $R \in \mathcal{L}_\infty$ and $\left| R^{\frac{1}{2}} \tilde{\theta} \right| \in \mathcal{L}_\infty \cap \mathcal{L}_2$ imply that $\dot{\theta} \in \mathcal{L}_\infty \cap \mathcal{L}_2$.

Gradient-based algorithm

- Since $\dot{\tilde{\theta}}, \dot{R} \in \mathcal{L}_\infty$, it follows from $\dot{\theta} = -\Gamma R(t)\tilde{\theta}$ that

$$\ddot{\tilde{\theta}} = -\Gamma \dot{R}(t)\tilde{\theta} - \Gamma R(t)\dot{\tilde{\theta}} \in \mathcal{L}_\infty$$

which, together with $\dot{\tilde{\theta}} \in \mathcal{L}_2$, implies

$$\lim_{t\to\infty} |\dot{\theta}(t)| = \lim_{t\to\infty} |\Gamma R(t)\tilde{\theta}(t)| = 0$$

To show that $\varepsilon m_s \in \mathcal{L}_2$ we proceed as follows. We have

$$\begin{aligned}
\frac{d}{dt}\tilde{\theta}^\top R\tilde{\theta} &= -\tilde{\theta}^\top (R + R^\top)\Gamma R\tilde{\theta} + \tilde{\theta}^\top \dot{R}\tilde{\theta} \\
&= -2\tilde{\theta}^\top R\Gamma R\tilde{\theta} + \tilde{\theta}^\top \left(-\beta R + \frac{\phi\phi^T}{m_s^2}\right)\tilde{\theta} \\
&= \varepsilon^2 m_s^2 - 2\tilde{\theta}^\top R\Gamma R\tilde{\theta} - \beta\tilde{\theta}^\top R\tilde{\theta}
\end{aligned}$$

Gradient-based algorithm

- Since $\dot{\tilde{\theta}}, \dot{R} \in \mathcal{L}_\infty$, it follows from $\dot{\theta} = -\Gamma R(t)\tilde{\theta}$ that

$$\ddot{\tilde{\theta}} = -\Gamma \dot{R}(t)\tilde{\theta} - \Gamma R(t)\dot{\tilde{\theta}} \in \mathcal{L}_\infty$$

which, together with $\dot{\tilde{\theta}} \in \mathcal{L}_2$, implies

$$\lim_{t\to\infty} |\dot{\theta}(t)| = \lim_{t\to\infty} |\Gamma R(t)\tilde{\theta}(t)| = 0$$

To show that $\varepsilon m_s \in \mathcal{L}_2$ we proceed as follows. We have

$$\begin{aligned}
\frac{d}{dt}\tilde{\theta}^\top R\tilde{\theta} &= -\tilde{\theta}^\top (R + R^\top)\Gamma R\tilde{\theta} + \tilde{\theta}^\top \dot{R}\tilde{\theta} \\
&= -2\tilde{\theta}^\top R\Gamma R\tilde{\theta} + \tilde{\theta}^\top \left(-\beta R + \frac{\phi\phi^T}{m_s^2}\right)\tilde{\theta} \\
&= \varepsilon^2 m_s^2 - 2\tilde{\theta}^\top R\Gamma R\tilde{\theta} - \beta\tilde{\theta}^\top R\tilde{\theta}
\end{aligned}$$

Gradient-based algorithm

Therefore,

$$\int_0^t \varepsilon^2 m_s^2 d\tau = \tilde{\theta}^\top R\tilde{\theta} + 2\int_0^t \tilde{\theta}^\top R\Gamma R\tilde{\theta} d\tau + \beta \int_0^t \tilde{\theta}^\top R\tilde{\theta} d\tau$$

Because $\lim_{t\to\infty}\left[\tilde{\theta}^\top(t)R(t)\tilde{\theta}(t)\right] = 0$ and $\left|R^{\frac{1}{2}}\tilde{\theta}\right| \in \mathcal{L}_2$ it follows that

$$\lim_{t\to\infty}\int_0^t \varepsilon^2 m_s^2 d\tau = \int_0^\infty \varepsilon^2 m_s^2 d\tau < \infty$$

i.e., $\varepsilon m_s \in \mathcal{L}_2$.

Last, the proof of exponential convergence can be found in Ioannou' s *Robust Adaptive Control* Section 4.8.

Gradient-based algorithm

Therefore,

$$\int_0^t \varepsilon^2 m_s^2 d\tau = \tilde{\theta}^\top R\tilde{\theta} + 2\int_0^t \tilde{\theta}^\top R\Gamma R\tilde{\theta} d\tau + \beta \int_0^t \tilde{\theta}^\top R\tilde{\theta} d\tau$$

Because $\lim_{t\to\infty}\left[\tilde{\theta}^\top(t)R(t)\tilde{\theta}(t)\right] = 0$ and $\left|R^{\frac{1}{2}}\tilde{\theta}\right| \in \mathcal{L}_2$ it follows that

$$\lim_{t\to\infty}\int_0^t \varepsilon^2 m_s^2 d\tau = \int_0^\infty \varepsilon^2 m_s^2 d\tau < \infty$$

i.e., $\varepsilon m_s \in \mathcal{L}_2$.

Last, the proof of exponential convergence can be found in Ioannou' s *Robust Adaptive Control* Section 4.8.

Example: Consider the plant

$$y = \frac{b_1 s + b_0}{s^2 + 2s + 1} u$$

where the parameters b_0 and b_1 are unknown.

Contents

- Vector Case
- Gradient-based Algorithms
- Least-square Algorithms

Least-square Algorithms

The basic idea behind is: fitting a mathematical model to a sequence of observed data by minimizing the sum of the squares of the difference between the observed and computed data.

Also for SPM

$$z = \theta^{*\top}\phi$$

Recall the instantaneous cost function

$$J(\theta) = \frac{\varepsilon^2 m_s^2}{2} = \frac{\left(z - \theta^\top \phi\right)^2}{2m_s^2}$$

at each time, its minimum satisfies

$$\nabla J(\theta) = -\frac{\left(z - \theta^\top \phi\right)}{m_s^2}\phi = 0$$

Least-square Algorithms

The basic idea behind is: fitting a mathematical model to a sequence of observed data by minimizing the sum of the squares of the difference between the observed and computed data.

Also for SPM

$$z = \theta^{*\top}\phi$$

Recall the instantaneous cost function

$$J(\theta) = \frac{\varepsilon^2 m_s^2}{2} = \frac{\left(z - \theta^\top\phi\right)^2}{2m_s^2}$$

at each time, its minimum satisfies

$$\nabla J(\theta) = -\frac{\left(z - \theta^\top\phi\right)}{m_s^2}\phi = 0$$

LS Algorithms

However,

$$z - \theta^\top \phi = 0$$

is not solvable for non-scalar θ, since $\phi\phi^\top$ is singular at each time instant. For scalar case, if the measurement is corrupted by an additive disturbance d_n

$$z = \theta^{*\top} \phi + d_n$$

then the estimate given by

$$\theta(t) = \frac{z(\tau)}{\phi(\tau)} = \theta^* + \frac{d_n(\tau)}{\phi(\tau)}$$

may be far off from true value due to small disturbance.

LS Algorithms

However,

$$z - \theta^\top \phi = 0$$

is not solvable for non-scalar θ, since $\phi\phi^\top$ is singular at each time instant. For scalar case, if the measurement is corrupted by an additive disturbance d_n

$$z = \theta^{*\top} \phi + d_n$$

then the estimate given by

$$\theta(t) = \frac{z(\tau)}{\phi(\tau)} = \theta^* + \frac{d_n(\tau)}{\phi(\tau)}$$

may be far off from true value due to small disturbance.

LS Algorithms

Consider the integral cost function

$$J(\theta) = \frac{1}{2}\int_0^t \varepsilon^2(t,\tau)m_s^2(\tau)d\tau$$

then

$$\nabla J = -\int_0^t \frac{z(\tau) - \theta^\top(t)\phi(\tau)}{m_s^2(\tau)}\phi(\tau)d\tau = 0$$

admits a solution

$$\theta(t) = \left(\int_0^t \frac{\phi(\tau)\phi^\top(\tau)}{m_s^2(\tau)}d\tau\right)^{-1}\int_0^t \frac{z(\tau)\phi(\tau)}{m_s^2(\tau)}d\tau$$

for any persistent exciting $\frac{\phi}{m_s}$ and $t \geq T_0$

Recursive LS Algorithm

Consider the cost function

$$J(\theta) = \frac{1}{2}\int_0^t e^{-\beta(t-\tau)} \frac{\left[z(\tau) - \theta^\top(t)\phi(\tau)\right]^2}{m_s^2(\tau)} d\tau$$
$$+ \frac{1}{2} e^{-\beta t} \left(\theta - \theta_0\right)^\top Q_0 \left(\theta - \theta_0\right)$$

where $\beta > 0$ and $Q_0 = Q_0^\top > 0$. $J(\theta)$ is a convex function of θ over $\mathbb{R}^n$ space, hence the global minimum, i.e. the estimate of θ^*. is therefore obtained by solving $\nabla J(\theta) = 0$.

Recursive LS Algorithm

Consider the cost function

$$J(\theta) = \frac{1}{2}\int_0^t e^{-\beta(t-\tau)} \frac{\left[z(\tau) - \theta^\top(t)\phi(\tau)\right]^2}{m_s^2(\tau)} d\tau + \frac{1}{2} e^{-\beta t} \left(\theta - \theta_0\right)^\top Q_0 \left(\theta - \theta_0\right)$$

where $\beta > 0$ and $Q_0 = Q_0^\top > 0$. $J(\theta)$ is a convex function of θ over $\mathbb{R}^n$ space, hence the global minimum, i.e. the estimate of θ^*. is therefore obtained by solving $\nabla J(\theta) = 0$.

Recursive LS Algorithm

Consider the cost function

$$J(\theta) = \frac{1}{2}\int_0^t e^{-\beta(t-\tau)}\frac{\left[z(\tau) - \theta^\top(t)\phi(\tau)\right]^2}{m_s^2(\tau)}d\tau + \frac{1}{2}e^{-\beta t}\left(\theta - \theta_0\right)^\top Q_0\left(\theta - \theta_0\right)$$

where $\beta > 0$ and $Q_0 = Q_0^\top > 0$. $J(\theta)$ is a convex function of θ over $\mathbb{R}^n$ space, hence the global minimum, i.e. the estimate of θ^*. is therefore obtained by solving $\nabla J(\theta) = 0$.

Recursive LS Algorithm

Solving

$$\nabla J(\theta) = e^{-\beta t} Q_0 \left(\theta(t) - \theta_0\right) - \int_0^t e^{-\beta(t-\tau)} \frac{z(\tau) - \theta^\top(t)\phi(\tau)}{m_s^2(\tau)} \phi(\tau) d\tau = 0$$

yields the *nonrecursive* LS algorithm

$$\theta(t) = P(t) \left[e^{-\beta t} Q_0 \theta_0 + \int_0^t e^{-\beta(t-\tau)} \frac{z(\tau)\phi(\tau)}{m_s^2(\tau)} d\tau \right]$$

where

$$P(t) = \left[e^{-\beta t} Q_0 + \int_0^t e^{-\beta(t-\tau)} \frac{\phi(\tau)\phi^\top(\tau)}{m_s^2(\tau)} d\tau \right]^{-1}$$

To avoid the calculation of matrix inverse, we can express $\theta(t)$ and $P(t)$ in a recursive way.

Recursive LS Algorithm

The *recursive* LS algorithm

$$\begin{aligned} \dot{\theta} &= P\varepsilon\phi, & \theta(0) &= \theta_0 \\ \dot{P} &= \beta P - P\frac{\phi\phi^T}{m_s^2}P, & P(0) &= P_0 = Q_0^{-1} \end{aligned}$$

Theorem: If $\frac{\phi}{m_s}$ is PE and $\beta > 0$ then the recursive LS algorithm with forgetting factor guarantees that $P, P^{-1} \in \mathcal{L}_\infty$ and that $\theta(t) \to \theta^*$ as $t \to \infty$ exponentially fast.

Remark:

1. Stability cannot be established unless $\frac{\phi}{m_s}$ is PE.

2. $P(t)$ may grow without bound.

Recursive LS Algorithm

The *recursive* LS algorithm

$$\begin{aligned} \dot{\theta} &= P\varepsilon\phi, & \theta(0) &= \theta_0 \\ \dot{P} &= \beta P - P\frac{\phi\phi^T}{m_s^2}P, & P(0) &= P_0 = Q_0^{-1} \end{aligned}$$

Theorem: If $\frac{\phi}{m_s}$ is PE and $\beta > 0$ then the recursive LS algorithm with forgetting factor guarantees that $P, P^{-1} \in \mathcal{L}_\infty$ and that $\theta(t) \to \theta^*$ as $t \to \infty$ exponentially fast.

Remark:

1. Stability cannot be established unless $\frac{\phi}{m_s}$ is PE.
2. $P(t)$ may grow without bound.

Recursive LS Algorithm

The *recursive* LS algorithm

$$\dot{\theta} = P\varepsilon\phi, \qquad \theta(0) = \theta_0$$
$$\dot{P} = \beta P - P\frac{\phi\phi^T}{m_s^2}P, \quad P(0) = P_0 = Q_0^{-1}$$

Theorem: If $\frac{\phi}{m_s}$ is PE and $\beta > 0$ then the recursive LS algorithm with forgetting factor guarantees that $P, P^{-1} \in \mathcal{L}_\infty$ and that $\theta(t) \to \theta^*$ as $t \to \infty$ exponentially fast.

Remark:

1. Stability cannot be established unless $\frac{\phi}{m_s}$ is PE.
2. $P(t)$ may grow without bound.

Modified Recursive LS Algorithm

To avoid the unboundedness of $P(t)$, LS is modified as follows:

$$\begin{aligned}
\dot{\theta} &= P\varepsilon\phi, \quad \theta(0) = \theta_0 \\
\dot{P} &= \begin{cases} \beta P - \frac{P\phi\phi^\top P}{m_s^2} & \text{if } \|P(t)\| \leq R_0 \\ 0 & \text{otherwise} \end{cases}
\end{aligned}$$

where $P(0) > 0, \|P(0)\| \leq R_0, R_0$ is a constant that serves as an upper bound for $\|P\|$.

Theorem: The modified recursive LS algorithm guarantees that

(i) $\varepsilon, \varepsilon m_s, \dot{\theta} \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\theta \in \mathcal{L}_\infty$

(ii) If $\frac{\phi}{m_s}$ is PE, then $\theta(t) \to \theta^*$ as $t \to \infty$ exponentially fast.

Modified Recursive LS Algorithm

To avoid the unboundedness of $P(t)$, LS is modified as follows:

$$\begin{aligned}
\dot{\theta} &= P\varepsilon\phi, \quad \theta(0) = \theta_0 \\
\dot{P} &= \begin{cases} \beta P - \frac{P\phi\phi^\top P}{m_s^2} & \text{if } \|P(t)\| \leq R_0 \\ 0 & \text{otherwise} \end{cases}
\end{aligned}$$

where $P(0) > 0, \|P(0)\| \leq R_0, R_0$ is a constant that serves as an upper bound for $\|P\|$.

Theorem: The modified recursive LS algorithm guarantees that

(i) $\varepsilon, \varepsilon m_s, \dot{\theta} \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\theta \in \mathcal{L}_\infty$

(ii) If $\frac{\phi}{m_s}$ is PE, then $\theta(t) \to \theta^*$ as $t \to \infty$ exponentially fast.

Pure LS

When $\beta = 0$, the recursive LS estimation algorithm reduced to

$$\dot{\theta} = P\varepsilon\phi, \qquad \theta(0) = \theta_0$$
$$\dot{P} = -P\frac{\phi\phi^\top}{m_s^2}P, \quad P(0) = P_0$$

which is referred to as the *pure LS algorithm.*

Theorem The pure LS algorithm guarantees that

(i) $\varepsilon, \varepsilon m_s, \dot{\theta} \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\theta, P \in \mathcal{L}_\infty$

(ii) $\lim_{t\to\infty} \theta(t) = \bar{\theta}$. If $\frac{\phi}{m_s}$ is PE, then $\bar{\theta} = \theta^*$

Note that, only asymptotic convergence can be assured.

Pure LS

When $\beta = 0$, the recursive LS estimation algorithm reduced to

$$\begin{aligned} \dot{\theta} &= P\varepsilon\phi, & \theta(0) &= \theta_0 \\ \dot{P} &= -P\frac{\phi\phi^\top}{m_s^2}P, & P(0) &= P_0 \end{aligned}$$

which is referred to as the *pure LS algorithm.*

Theorem The pure LS algorithm guarantees that

(i) $\varepsilon, \varepsilon m_s, \dot{\theta} \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\theta, P \in \mathcal{L}_\infty$

(ii) $\lim_{t\to\infty} \theta(t) = \bar{\theta}$. If $\frac{\phi}{m_s}$ is PE, then $\bar{\theta} = \theta^*$

Note that, only asymptotic convergence can be assured.

Pure LS

When $\beta = 0$, the recursive LS estimation algorithm reduced to

$$\begin{aligned} \dot{\theta} &= P\varepsilon\phi, & \theta(0) &= \theta_0 \\ \dot{P} &= -P\tfrac{\phi\phi^\top}{m_s^2}P, & P(0) &= P_0 \end{aligned}$$

which is referred to as the *pure LS algorithm.*

Theorem The pure LS algorithm guarantees that

(i) $\varepsilon, \varepsilon m_s, \dot{\theta} \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\theta, P \in \mathcal{L}_\infty$

(ii) $\lim_{t\to\infty} \theta(t) = \bar{\theta}$. If $\frac{\phi}{m_s}$ is PE, then $\bar{\theta} = \theta^*$

Note that, only asymptotic convergence can be assured.

Pure LS

Drawbacks:

- covariance wind-up problem : covariance matrix P may become arbitrarily small and slow down adaptation in some directions.
- non-exponential convergence speed. see a scalar example:

$$\dot{p} = -p^2\phi^2, \quad p(0) = p_0 > 0$$

  let $\phi = 1$ which is PE in this case, then we have

$$p(t) = \frac{p_0}{1 + p_0 t}$$

  and

$$\tilde{\theta}(t) = \frac{\tilde{\theta}_0}{1 + p_0 t}$$

Pure LS

Drawbacks:

- covariance wind-up problem : covariance matrix P may become arbitrarily small and slow down adaptation in some directions.
- non-exponential convergence speed. see a scalar example:

$$\dot{p} = -p^2\phi^2, \quad p(0) = p_0 > 0$$

let $\phi = 1$ which is PE in this case, then we have

$$p(t) = \frac{p_0}{1 + p_0 t}$$

and

$$\tilde{\theta}(t) = \frac{\tilde{\theta}_0}{1 + p_0 t}$$

Pure LS

Drawbacks:

- covariance wind-up problem : covariance matrix P may become arbitrarily small and slow down adaptation in some directions.
- non-exponential convergence speed. see a scalar example:

$$\dot{p} = -p^2\phi^2, \quad p(0) = p_0 > 0$$

let $\phi = 1$ which is PE in this case, then we have

$$p(t) = \frac{p_0}{1 + p_0 t}$$

and

$$\tilde{\theta}(t) = \frac{\tilde{\theta}_0}{1 + p_0 t}$$

Pure LS

Drawbacks:

- covariance wind-up problem : covariance matrix P may become arbitrarily small and slow down adaptation in some directions.
- non-exponential convergence speed. see a scalar example:

$$\dot{p} = -p^2\phi^2, \quad p(0) = p_0 > 0$$

let $\phi = 1$ which is PE in this case, then we have

$$p(t) = \frac{p_0}{1 + p_0 t}$$

and

$$\tilde{\theta}(t) = \frac{\tilde{\theta}_0}{1 + p_0 t}$$

Modified pure LS

Avoid the covariance wind-up problem via resetting

$$\dot{\theta} = P\varepsilon\phi, \qquad \theta(0) = \theta_0$$
$$\dot{P} = -P\frac{\phi\phi^T}{m_s^2}P, \qquad P\left(t_r^+\right) = P_0 = \rho_0 I$$

where t_r^+ is the time at which $\lambda_{\min}(P(t)) \leq \rho_1$ and $\rho_0 > \rho_1 > 0$ are some design scalars. Therefore, P is guaranteed to be positive definite for all $t > 0$.

Theorem: The pure LS algorithm with covariance resetting guarantees that

(i) $\varepsilon, \varepsilon m_s, \dot{\theta} \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\theta \in \mathcal{L}_\infty$

(ii) If $\frac{\phi}{m_s}$ is PE, then $\theta(t) \to \theta^*$ as $t \to \infty$ exponentially fast.

Modified pure LS

Avoid the covariance wind-up problem via resetting

$$
\begin{aligned}
\dot{\theta} &= P\varepsilon\phi, & \theta(0) &= \theta_0 \\
\dot{P} &= -P\frac{\phi\phi^T}{m_s^2}P, & P\left(t_r^+\right) &= P_0 = \rho_0 I
\end{aligned}
$$

where t_r^+ is the time at which $\lambda_{\min}(P(t)) \leq \rho_1$ and $\rho_0 > \rho_1 > 0$ are some design scalars. Therefore, P is guaranteed to be positive definite for all $t > 0$.

Theorem: The pure LS algorithm with covariance resetting guarantees that

(i) $\varepsilon, \varepsilon m_s, \dot{\theta} \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\theta \in \mathcal{L}_\infty$

(ii) If $\frac{\phi}{m_s}$ is PE, then $\theta(t) \to \theta^*$ as $t \to \infty$ exponentially fast.

Modified pure LS

Avoid the covariance wind-up problem via resetting

$$\dot{\theta} = P\varepsilon\phi, \qquad \theta(0) = \theta_0$$
$$\dot{P} = -P\frac{\phi\phi^T}{m_s^2}P, \qquad P\left(t_r^+\right) = P_0 = \rho_0 I$$

where t_r^+ is the time at which $\lambda_{\min}(P(t)) \leq \rho_1$ and $\rho_0 > \rho_1 > 0$ are some design scalars. Therefore, P is guaranteed to be positive definite for all $t > 0$.

Theorem: The pure LS algorithm with covariance resetting guarantees that

(i) $\varepsilon, \varepsilon m_s, \dot{\theta} \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\theta \in \mathcal{L}_\infty$

(ii) If $\frac{\phi}{m_s}$ is PE, then $\theta(t) \to \theta^*$ as $t \to \infty$ exponentially fast.