Cours EDSYS - Commande Adaptative

Jean-Marc BIANNIC - DCSD-ONERA - Toulouse

Denis EFIMOV - IMS - Université de Bordeaux 1

Dimitri PEAUCELLE - LAAS-CNRS - Université de Toulouse

Toulouse EDSYS Mai 2011
INDIRECT ADAPTIVE CONTROL

Denis EFIMOV - IMS - Université de Bordeaux 1

OUTLINE

1. Introduction
   a. Main properties
   b. Running example
2. Adaptive parameter estimation
   a. Parameterized system model
   b. Linear parametric model
   c. Normalized gradient algorithm
   d. Normalized least-squares algorithm
   e. Discrete-time version of adaptive algorithms
3. Identification and robustness
   a. Parametric convergence and persistency of excitation
   b. Robustness of adaptive algorithms
4. Indirect adaptive control
   a. Model reference control
   b. Pole placement control
5. Adaptive observers
1. INTRODUCTION

Dynamic systems are characterized by their structures and parameters:

**Linear:**
\[
\Sigma_l : \begin{cases} 
\dot{x} = A(\theta)x + B(\theta)u + d; \\
y = Cx + Du + v,
\end{cases}
\]

**Nonlinear:**
\[
\Sigma_n : \begin{cases} 
\dot{x} = f(x, u, d, \theta); \\
y = h(x, u) + v,
\end{cases}
\]

- \(x\) is state vector,
- \(u\) is control input,
- \(d\) is disturbance,
- \(y\) is output,
- \(v\) is noise,
- \(\theta\) is parameters.

Control system design steps:

1. **Modeling**
   \[ u \xrightarrow{P} y \]

2. **Control design**
   \[ u \xrightarrow{P_m} y \]
   \(P_m \in \{\Sigma_l, \Sigma_n\}\)

3. **Implementation**
   \[ u \xrightarrow{P_m} y \]
   \[ \Delta \text{ uncertainty} \]

Stability, robustness, performance???
a. Main properties

Parameter estimation is to use a collection of available system signals \( y \) and \( u \), based on certain system structure information \( \Sigma_l \) or \( \Sigma_n \), to produce estimates \( \hat{\theta}(t) \) of the system parameters \( \theta \). Appears on the step 1.

Adaptive parameter estimation is a dynamic estimation procedure that produce updated parameter estimates on-line. Appears on the step 2&3.

Adaptive parameter estimation is crucial for indirect adaptive control design where controller parameters \( \theta_c(t) \) are some continuous functions of the estimates \( \hat{\theta}(t) \):

The general scheme of adaptive control. The scheme of indirect adaptive control.
Key issues in the classical adaptive parameter estimation:

- linear parameterization of system models,
- linear representation of parametric error models,
- stable design of adaptive estimation algorithms,
- analytical proof of system stability,
- parameter convergence,
- robustness of adaptive estimation.

Realization:

- continuous-time,
- discrete-time.
b. Running example

Moving vehicle:

\[ V_d \rightarrow F_e = kN_e \]

\[ V \rightarrow F_{f+} + F_l \]

\[ F_{f<} \]

\[ F_{f>} \]

\[ V \text{ is velocity (regulating variable), } \dot{V} = dV / dt \text{ is acceleration, } m \text{ is unknown vehicle mass,} \]

\[ F_e \text{ is engine force, } F_e = kN_e \text{, where } N_e \text{ is torque, } k \text{ is unknown conversion coefficient,} \]

\[ F_f \text{ is friction force, } F_f = \rho V \text{, where } \rho \text{ is unknown friction coefficient,} \]

\[ F_l \text{ is load force (unknown, dependent on the road profile).} \]

The first order dynamics (Newton's Second Law):

\[ m\dot{V} = F_e - (F_f + F_l) = kN_e - \rho V - F_l. \]
Define the state variable \( x = V \), the control input \( u = N_e \), the disturbance \( d = -F_l / m \):

\[
\dot{x} = -ax + bu + d, \\
y = x + \nu,
\]

where \( y \) is the output, \( \nu \) is the measurement noise, \( a = \rho / m \), \( b = k / m \).

**Note:** the engine from the introduction lecture has the same model \( I\dot{\omega} = -f\omega + Ku \).

**Features:**
- the constant parameters \( a > 0 \) and \( b > 0 \) are unknown \( \Rightarrow \) (1) is a variant of \( \Sigma_l \);
- the time-varying signals \( d \) and \( \nu \) are unknown, but bounded;
- the unperturbed noise-free case: \( d = \nu = 0 \);
- the reference signal \( r = V_d \), where \( V_d \) – desired velocity.

**Control problem** (the asymptotic tracking):

\[ x(t) \to r(t) \text{ with } t \to +\infty. \]
A variant of the solution:

\[ u = b^{-1}[ay - a_m y + b_m r], \]

where \( a_m > 0 \) and \( b_m \) are parameters of the reference model:

\[ \dot{x}_m = -a_m x_m + b_m r. \]

The closed loop system has form:

\[ \dot{x} = -a_m x + b_m r + \tilde{d}, \quad \tilde{d} = d + (a - a_m)v. \]

In the noise-free case \((d = v = 0 \Rightarrow \tilde{d} = 0)\) the variable \( x \) has the desired dynamics!

To design the control \( u \) we have to estimate the unknown parameters \( a \) and \( b \)!

Let us try to solve this problem for the noise-free case. We will analyze the robustness issue later. In this case the model (1) can be rewritten as follows:

\[ \dot{y} = -ay + bu. \quad (1') \]
2. ADAPTIVE PARAMETER ESTIMATION

a. Parameterized system model

Consider a **linear time-invariant SISO system** described by the differential equation:

\[ P(s)[y](t) = Z(s)[u](t), \quad (2) \]

\( y(t) \in \mathbb{R}, u(t) \in \mathbb{R} \) are the measured output and input as before;

\[
P(s) = s^n + p_{n-1}s^{n-1} + \ldots + p_1s + p_0,
\]

\[
Z(s) = z_ms^m + z_{m-1}s^{m-1} + \ldots + z_1s + z_0,
\]

are polynomials in \( s \), with \( s \) being the differentiation operator 

\[ s[x](t) = \dot{x}(t); \]

\( p_i, i = 0, \overline{n-1} \) and \( z_j, j = 0, \overline{m} \) are the unknown but constant parameters to be estimated.

**Note:** \( n = 1, m = 0 \Rightarrow (1') \) with \( p_0 = a \) and \( z_0 = b \).

**The objective:** estimate the values \( p_i, i = 0, \overline{n-1} \) and \( z_j, j = 0, \overline{m} \) using available for on-line measurements signals \( y(t) \) and \( u(t) \) (no \textit{a priori} accessible datasets).
**Parameterization:**

let $\Lambda(s) = s^n + \lambda_{n-1}s^{n-1} + \ldots + \lambda_1 s + \lambda_0$ be a stable polynomial (all zeros are in Re$[s] < 0$).

Then (2) can be represented as follows:

$$\frac{P(s)}{\Lambda(s)}[y](t) = \frac{Z(s)}{\Lambda(s)}[u](t) \Rightarrow \left(1 - \frac{\Lambda(s)}{\Lambda(s)}\right)[y](t) + \frac{P(s)}{\Lambda(s)}[y](t) = \frac{Z(s)}{\Lambda(s)}[u](t) \Rightarrow$$

$$\left.\right.$$

$$y(t) = \frac{Z(s)}{\Lambda(s)}[u](t) + \frac{\Lambda(s) - P(s)}{\Lambda(s)}[y](t). \quad (3)$$

Define **parameter vector**

$$\theta^* = [z_0, z_1, \ldots, z_{m-1}, z_m, \lambda_0 - p_0, \lambda_1 - p_1, \ldots, \lambda_{n-2} - p_{n-2}, \lambda_{n-1} - p_{n-1}]^T \in \mathbb{R}^{n+m+1}$$

and **regressor function**

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

Then (3) can be expressed in the equivalent form

$$y(t) = \theta^{*T} \phi(t). \quad (4)$$
In (4):
- the vector $\theta^*$ contains all unknown parameters of the system (2);
- the regressor $\phi(t)$ can be computed using the filters $\frac{s^i}{\Lambda(s)}$, $i = 0, n-1$.

Another variant of implementation:

\[
\begin{align*}
\dot{\omega}_1(t) &= A_\lambda \omega_1(t) + bu(t), \\
\dot{\omega}_2(t) &= A_\lambda \omega_2(t) + by(t),
\end{align*}
\]

where $\omega_1(t) \in \mathbb{R}^n$, $\omega_2(t) \in \mathbb{R}^n$ and

\[
A_\lambda = 
\begin{bmatrix}
0 & 1 & 0 & \cdots & 0 & 0 \\
0 & 0 & 1 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & \cdots & 0 & 1 \\
-\lambda_0 & -\lambda_1 & \cdots & \cdots & -\lambda_{n-2} & -\lambda_{n-1}
\end{bmatrix},
\quad
b = 
\begin{bmatrix}
0 \\
\vdots \\
0 \\
1
\end{bmatrix}.
\]

Then, we generate the regressor $\phi(t)$ from

\[
\phi(t) = \left[ \begin{bmatrix} C_m \omega_1(t) \end{bmatrix}^T, \omega_2(t)^T \right]^T,
\]

where

\[
C_m = [I_{m+1}, \mathbf{0}_{(m+1) \times (n-m-1)}] \in \mathbb{R}^{(m+1) \times n} \quad (\phi(t) = [\omega_1(t)^T, \omega_2(t)^T]^T \text{ for } m = n - 1).
\]
b. Linear parametric model

**Linear parametric model** has the form

\[ y(t) = \theta^* T \phi(t), \ t \geq t_0, \]  

(4)

where \( \theta^* \in \mathbb{R}^{n_\theta} \) is an unknown parameter vector, \( y(t) \in \mathbb{R} \) is a known (measured) signal, \( \phi(t) \in \mathbb{R}^{n_\theta} \) is a known vector signal (regressor), \( n_\theta = n + m + 1 \) is the dimension of the model.

**Features:**

1) The model (4) is commonly seen in system modeling when unknown system parameters can be separated from known signals.

2) The components of \( \phi(t) \) may contain nonlinear and/or filtered functions of \( y(t) \) and \( u(t) \) (or some other system signals).

3) **Adaptive parameter estimation** based on \( y(t), u(t) \) \( \Leftrightarrow \) **Linear parametric model**.

Let \( \theta(t) \) be **the estimate** of \( \theta^* \) obtained from an adaptive update law, \( \tilde{\theta}(t) = \theta(t) - \theta^* \) is **the parametric error**, then define **the estimation error**

\[ \varepsilon(t) = \theta(t)^T \phi(t) - y(t) = \theta(t)^T \phi(t) - \theta^T \phi(t) = \tilde{\theta}(t)^T \phi(t). \]  

(5)
Example 1

\[ \dot{y} = -ay + bu. \quad (1') \]

It has the form (2) for \( P(s) = s + p_0, \ Z(s) = z_0 \) with \( p_0 = a, \ z_0 = b, \ m = n-1, \ n = 1. \)

The filter

\[ \frac{1}{\Lambda(s)} = \frac{1}{s+1}. \]

The parameter vector

\[ \theta^* = [\theta_1^*, \theta_2^*]^T = [b, 1-a]^T \in \mathbb{R}^2, \ n_{\theta} = 2. \]

The regressor

\[ \phi(t) = \left[ \frac{1}{s+1}[u](t), \frac{1}{s+1}[y](t) \right]^T \in \mathbb{R}^2. \]

The fast implementation \( \phi(t) = [\omega_1(t), \omega_2(t)]^T \) for

\[ \dot{\omega}_1(t) = -\omega_1(t) + u(t), \quad \dot{\omega}_2(t) = -\omega_2(t) + y(t), \quad \mathbf{A}_\lambda = -1, \quad \mathbf{b} = 1. \]

The estimation error for the estimate \( \theta(t) = [\theta_1(t), \theta_2(t)]^T \in \mathbb{R}^2: \)

\[ \varepsilon(t) = \theta(t)^T \phi(t) - y(t) = \omega_1(t)\theta_1(t) + \omega_2(t)\theta_2(t) - y(t) = \omega_1(t)(\theta_1(t) - b) + \omega_2(t)(\theta_2(t) - 1 + a) = \hat{\theta}(t)^T \phi(t). \]
c. Normalized gradient algorithm

How to update $\theta(t)$? How to minimize the error $\epsilon(t) = \theta(t)^T \phi(t) - y(t) = \tilde{\theta}(t)^T \phi(t)$?

The idea is to choose the derivative of $\theta(t)$ in a steepest descent direction in order to minimize a normalized quadratic cost functional

$$J(t, \theta) = \frac{\epsilon(t)^2}{2m(t)^2} = \tilde{\theta}(t)^T \phi(t) \phi(t)^T \tilde{\theta}(t) = \frac{(\theta(t) - \theta^*)^T \phi(t) \phi(t)^T (\theta(t) - \theta^*)}{2m(t)^2},$$

where $m(t)$ is a normalizing signal not depending (explicitly) on $\theta(t)$.

The idea of $m(t)$ choice: $\phi(t) \phi(t)^T / m(t)^2$ has to be bounded (return later to this issue).

The steepest descent direction of $J(t, \theta)$ is

$$\frac{\partial J(t, \theta)}{\partial \theta} = -\frac{\epsilon(t)}{m(t)^2} \frac{\partial \epsilon}{\partial \theta} = -\epsilon(t) \frac{\phi(t)}{m(t)^2},$$

therefore:

$$\dot{\theta}(t) = -\epsilon(t) \Gamma \frac{\phi(t)}{m(t)^2}, \quad \theta(t_0) = \theta_0, \quad t \geq t_0,$$

where $\Gamma = \Gamma^T > 0$ is a design matrix gain, $\theta_0$ is an initial estimate of $\theta^*$. 

EDSYS
For (6) an admissible choice of the normalizing function \( m(t) \) is

\[
m(t) = \sqrt{1 + \kappa \phi(t)^T \phi(t)},
\]

where \( \kappa > 0 \) is a design parameter.

**Example 1**

The estimation error and the regressor:

\[
\varepsilon(t) = \omega_1(t)\theta_1(t) + \omega_2(t)\theta_2(t) - y(t), \quad \phi(t) = [\omega_1(t), \omega_2(t)]^T.
\]

The cost functional and derivative:

\[
J(t, \theta) = \frac{\varepsilon(t)^2}{2m(t)^2} = \frac{[\omega_1(t)\theta_1(t) + \omega_2(t)\theta_2(t) - y(t)]^2}{2m(t)^2}, \quad \frac{\partial J(t, \theta)}{\partial \theta} = -\frac{\varepsilon(t)}{m(t)^2} \begin{bmatrix} \omega_1(t) \\ \omega_2(t) \end{bmatrix}.
\]

The normalized gradient algorithm for \( \Gamma = \gamma I_2 \), \( \gamma > 0 \) and \( \kappa = 1 \):

\[
\dot{\theta}(t) = -\gamma \frac{\varepsilon(t)}{1 + \omega_1^2(t) + \omega_2^2(t)} \begin{bmatrix} \omega_1(t) \\ \omega_2(t) \end{bmatrix}.
\]
Lemma 1. The adaptive algorithm (6) guarantees:

(i) \( \theta(t), \dot{\theta}(t) \) and \( \varepsilon(t)/m(t) \) are bounded (belong to \( L_{\infty} \));

(ii) \( \varepsilon(t)/m(t) \) and \( \dot{\theta}(t) \) belong to \( L_2 \).

Proof. Introduce the positive definite (Lyapunov) function \( V(\tilde{\theta}) = \tilde{\theta}^T \Gamma^{-1} \tilde{\theta} \), then \( (\dot{\theta} = \dot{\theta}) \)

\[
\dot{V} = 2\tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}} = 2\tilde{\theta}^T \Gamma^{-1} \left[ -\varepsilon(t) \Gamma \frac{\phi(t)}{m(t)^2} \right] = -2\varepsilon(t) \frac{\tilde{\theta}^T \phi(t)}{m(t)^2} = -2 \frac{\varepsilon(t)^2}{m(t)^2}, \quad t \geq t_0. \tag{7}
\]

Since \( \dot{V} \leq 0 \) we have: \( V(t) \in L_{\infty} \Rightarrow \tilde{\theta}(t) \in L_{\infty} \Rightarrow \theta(t) \in L_{\infty} = \) all these signals are bounded.

The boundedness of \( \varepsilon(t)/m(t) \) follows the boundedness of \( \tilde{\theta}(t) \) and the inequality

\[
\frac{|\varepsilon(t)|}{m(t)} = \frac{|\tilde{\theta}^T \phi(t)|}{m(t)} \leq \frac{||\phi(t)||}{\sqrt{1 + \kappa \phi^T(t) \phi(t)}} ||\tilde{\theta}(t)||.
\]

Then boundedness of \( \dot{\theta}(t) \) follows from the inequality

\[
||\dot{\theta}|| = \left\| \varepsilon(t) \Gamma \frac{\phi(t)}{m(t)^2} \right\| \leq \|\Gamma\| \left\| \frac{|\varepsilon(t)| ||\phi(t)||}{m(t)} \right\| \leq \|\Gamma\| \frac{||\phi(t)||}{\sqrt{1 + \kappa ||\phi(t)||^2}} \frac{|\varepsilon(t)|}{m(t)}. \Rightarrow (i)
\]
Lemma 1. The adaptive algorithm (6) guarantees:

(i) $\theta(t), \dot{\theta}(t)$ and $\varepsilon(t) / m(t)$ are bounded (belong to $L_\infty$);
(ii) $\varepsilon(t) / m(t)$ and $\dot{\theta}(t)$ belong to $L_2$.

Proof. Let us rewrite the equality (7) in the form $2 \frac{\varepsilon(t)^2}{m(t)^2} = -\dot{V}(t)$ and integrate it:

$$-2 \int_{t_0}^{t} \frac{\varepsilon(t)^2}{m(t)^2} dt = \int_{t_0}^{t} \dot{V}(t) dt = V(t_0) - V(t) \leq V(t_0) = (\theta_0 - \theta^*)^T \Gamma^{-1} (\theta_0 - \theta^*) < \infty, \ t \geq t_0,$$

therefore $\frac{\varepsilon(t)}{m(t)} \in L_2$. From the inequality

$$||\dot{\theta}|| \leq ||\Gamma|| \frac{||\phi(t)||}{\sqrt{1 + \kappa ||\phi(t)||^2}} \frac{|\varepsilon(t)|}{m(t)},$$

we obtain that $\dot{\theta}(t)$ belongs to $L_2$. $\Rightarrow$ (ii) $\Rightarrow$ The Lemma 1 is proven.

Note:

We did not prove that $\lim_{t \to \infty} \theta(t) = \theta^*$!
Discussion:

1) The algorithm has equilibriums when $||\dot{\theta}(t)|| = 0$, from (6) we have $||\dot{\theta}(t)|| = ||\epsilon(t)\Gamma \frac{\phi(t)}{m(t)^2}||$:

$$||\phi(t)|| = 0 \Rightarrow ||\dot{\theta}(t)|| = 0 \iff \epsilon(t) = 0 \iff \theta(t) = \theta^*!$$

$\theta(t) = \theta^*$ is not unique equilibrium of (6) (the usual drawback of any gradient algorithm)!

2) $V(t) = \tilde{\theta}(t)^T \Gamma^{-1} \tilde{\theta}(t)$ is a measure of deviation of $\theta(t)$ from $\theta^*$, and from (7)

$$\dot{V}(t) \leq 0 \Rightarrow [\theta(t) - \theta^*]^T \Gamma^{-1} [\theta(t) - \theta^*] = V(t) \leq V(t_0) = [\theta_0 - \theta^*]^T \Gamma^{-1} [\theta_0 - \theta^*].$$

3) From Lemma 1 we have that $\epsilon(t)/m(t) \in L_\infty \cap L_2$ and $\lim_{t \to \infty} \epsilon(t)/m(t) = 0$.

4) From (7) we have that the function is nonincreasing ($\dot{V}(t) \leq 0$) and bounded from below ($V(t) \geq 0$), thus there exists $\lim_{t \to \infty} V(t) = V_\infty$ for some constant $V_\infty \geq 0$:

- $V_\infty = 0 \Rightarrow \lim_{t \to \infty} \theta(t) = \theta^*$;
- $V_\infty > 0 \Rightarrow \lim_{t \to \infty} \theta(t) = \theta_\infty$ for some constant vector $\theta_\infty = \mathbb{R}^n_\theta$.

5) if $\ddot{\theta}(t) \in L_\infty \Rightarrow \dot{\theta}(t) \in L_\infty \cap L_2$ (Lemma 1) $\Rightarrow \lim_{t \to \infty} \dot{\theta}(t) = 0 \Rightarrow \lim_{t \to \infty} \theta(t) = \theta_\infty$.

$\theta(t) = \sin(\sqrt{t+1})$, $\dot{\theta}(t) = 0.5 \frac{\cos(\sqrt{t+1})}{\sqrt{t+1}} \Rightarrow \dot{\theta}(t) \in L_\infty \cap L_2$, $\lim_{t \to \infty} \dot{\theta}(t) = 0$, $\lim_{t \to \infty} \theta(t) = ?$
Example 1

Plant:

\[ \dot{y} = -ay + bu. \]

Adaptive estimator:

\[ \dot{\theta}(t) = -\gamma \frac{\omega_1(t)\theta_1(t) + \omega_2(t)\theta_2(t) - y(t)}{1 + \omega_1^2(t) + \omega_2^2(t)} \begin{bmatrix} \omega_1(t) \\ \omega_2(t) \end{bmatrix}, \quad \dot{\omega}_1(t) = -\omega_1(t) + u(t), \quad \dot{\omega}_2(t) = -\omega_2(t) + y(t). \]

Simulation 1:

\[ a = 0.5, \; b = 1, \; \gamma = 20 \text{ and } u(t) = \sin(t), \]

Simulation 2:

\[ a = 1.5, \; b = 2, \; \gamma = 20 \text{ and } u(t) = \sin(t), \]
Simulation 3: \( a = 1.5, \ b = 2, \ \gamma = 20 \) and \( u(t) = 1 - e^{-t} \cos(t), \)

Conclusions:

- the convergence of adjusted estimates \( \theta(t) \) to their ideal values \( \theta^* \) depends on the input \( u \);
- \( y, u \) are oscillating \( \Rightarrow \theta(t) \rightarrow \theta^* \); \( y \rightarrow \text{const}, u \rightarrow \text{const} \) (set-point) \( \Rightarrow \theta(t) \rightarrow \theta^*. \)
d. Normalized least-squares algorithm

\[
\dot{\theta}(t) = -\varepsilon(t) \frac{P(t)\phi(t)}{m(t)^2}, \quad \theta(t_0) = \theta_0, \quad t \geq t_0, \quad (8)
\]

\[
\dot{P}(t) = -\frac{P(t)\phi(t)\phi(t)^T P(t)}{m(t)^2}, \quad P(t_0) = P_0 = P_0^T > 0, \quad t \geq t_0, \quad (9)
\]

where \(\kappa > 0\) is a design parameter, \(\theta_0\) is the initial estimate of \(\theta^*\) and \(P_0\) is the initial value of the gain matrix \(P(t) \in \mathbb{R}^{n_\theta \times n_\theta}\).

**Note:**
- if \(P(t) = \Gamma\) for all \(t \geq t_0\), then \((8) \Rightarrow (6)\);
- the dimension of \((6)\) is \(n_\theta = n + m + 1\), as far as the dimension of \((8)\), \((9)\) is \(n_\theta + n_\theta^2\).

**Example 1**

\[
\dot{\theta} = -\frac{\varepsilon}{m^2} \left[ \begin{array}{c} P_{1,1}^2 \omega_1 + P_{1,2}^2 \omega_2 \\ P_{2,1}^2 \omega_1 + P_{2,2}^2 \omega_2 \end{array} \right], \quad \dot{P} = -\frac{1}{m^2} \left[ \begin{array}{c} P_{1,1}^2 \omega_1 + P_{1,2}^2 \omega_2 \\ P_{2,1}^2 \omega_1 + P_{2,2}^2 \omega_2 \end{array} \right]^T 
\]

\[
m^2 = 1 + P_{1,1}^2 \omega_1^2 + P_{1,2}^2 \omega_1 \omega_1 + P_{2,1}^2 \omega_1 \omega_1 + P_{2,2}^2 \omega_2^2.
\]
Lemma 2. The adaptive algorithm (8),(9) guarantees:

(i) \( P(t) = P^T(t) > 0 \) for all \( t \geq t_0 \), \( P(t), \dot{P}(t) \) are bounded;

(ii) \( \theta(t) \) and \( \varepsilon(t) / \bar{m}(t) \) are bounded (belong to \( L_\infty \)), where \( \bar{m}(t) = \sqrt{1 + \phi(t)^T \phi(t)} \);

(iii) \( \varepsilon(t) / m(t), \varepsilon(t) / \bar{m}(t) \) and \( \dot{\theta}(t) \) belong to \( L_2 \);

(iv) there exist a constant matrix \( P_\infty \in \mathbb{R}^{n_\theta \times n_\theta} \), and a constant vector \( \theta_\infty \in \mathbb{R}^{n_\theta} \) such that

\[
\lim_{t \to \infty} P(t) = P_\infty, \lim_{t \to \infty} \theta(t) = \theta_\infty.
\]

Proof. First, \( P(t) = P^T(t) \) and \( \dot{P}(t) \) is bounded by the algorithm (9) construction:

\[
\dot{P}(t) = -\frac{P(t)\phi(t)\phi(t)^T}{1 + \kappa \phi(t)^T P(t)\phi(t)} P(t).
\]

Second, the identity \( P(t)P(t)^{-1} = I_{n_\theta} \) implies

\[
\frac{d}{dt}(P(t)^{-1}) = -P(t)^{-1}\dot{P}(t)P(t)^{-1} = m(t)^{-2}\phi(t)\phi(t)^T,
\]

then integrating this equality we obtain:

\[
P(t)^{-1} = P(t_0)^{-1} + \int_{t_0}^{t} m(\tau)^{-2}\phi(\tau)\phi(\tau)^T d\tau, \quad t \geq t_0.
\] (10)

\[
P(t_0)^{-1} > 0 \Rightarrow P(t)^{-1} \geq P(t_0)^{-1} > 0 \Rightarrow P(t) > 0 \text{ and } P(t) \text{ is bounded.} \Rightarrow (i)
\]
Consider the positive definite function $V(t, \tilde{\theta}) = \tilde{\theta}^T P(t)^{-1} \tilde{\theta}$, then ($\varepsilon(t) = \tilde{\theta}(t)^T \phi(t)$)

$$
\dot{V} = \tilde{\theta}(t)^T P(t)^{-1} \tilde{\theta}(t) + \tilde{\theta}(t)^T P(t)^{-1} \tilde{\varepsilon}(t) + \tilde{\theta}(t)^T \frac{d}{dt} \left( P(t)^{-1} \right) \tilde{\theta}(t) = 
$$

$$
= -\varepsilon(t) \frac{\phi(t)^T P(t)}{m(t)^2} P(t)^{-1} \tilde{\theta}(t) - \tilde{\theta}(t)^T P(t)^{-1} \varepsilon(t) \frac{P(t)\phi(t)}{m(t)^2} + \tilde{\theta}(t)^T \frac{\phi(t)\phi(t)^T}{m(t)^2} \tilde{\theta}(t) = 
$$

$$
= -\varepsilon(t) \frac{\phi(t)^T \tilde{\theta}(t)}{m(t)^2} - \varepsilon(t) \frac{\tilde{\theta}(t)^T \phi(t)}{m(t)^2} + \frac{\tilde{\theta}(t)^T \phi(t)\phi(t)^T}{m(t)^2} \tilde{\theta}(t) = -\frac{\varepsilon(t)^2}{m(t)^2}, t \geq t_0. 
$$

Hence, $V(t) = V[t, \tilde{\theta}(t)]$ is bounded, and using (10) we obtain:

$$
V(t) = \tilde{\theta}(t)^T P(t_0)^{-1} \tilde{\theta}(t) + \tilde{\theta}(t)^T \left( \int_{t_0}^t m(\tau)^{-2} \phi(\tau)\phi(\tau)^T d\tau \right) \tilde{\theta}(t) < \infty, t \geq t_0.
$$

Therefore

$$
\tilde{\theta}(t)^T P(t_0)^{-1} \tilde{\theta}(t) \text{ is bounded } \Rightarrow \tilde{\theta}(t) \text{ and } \theta(t) \text{ are bounded.}
$$

Boundedness of $\varepsilon(t) / \bar{m}(t)$ follows the proven property $\tilde{\theta}(t) \in L_\infty$ and the inequality

$$
\frac{|\varepsilon(t)|}{\bar{m}(t)} = \frac{|\tilde{\theta}(t)^T \phi(t)|}{\sqrt{1 + \phi(t)^T \phi(t)}} \leq \frac{\|\phi(t)\|}{\sqrt{1 + \phi(t)^T \phi(t)}} \|\tilde{\theta}(t)\|. \Rightarrow (\text{ii})
$$
Rewriting the equality (11) in the form $2\varepsilon(t)^2 / m(t)^2 = -\dot{V}(t)$ and integrating it, we obtain:

$$-2\int_{t_0}^{t} \frac{\varepsilon(t)^2}{m(t)^2} dt = \int_{t_0}^{t} \dot{V}(t) dt = V(t_0) - V(t) \leq V(t_0) = (\theta_0 - \theta^*)^T \mathbf{P}_0^{-1}(\theta_0 - \theta^*) < \infty, \quad t \geq t_0,$$

therefore $\frac{\varepsilon(t)}{m(t)} \in L_2$ and

$$\frac{\varepsilon(t)}{\bar{m}(t)} = \frac{\varepsilon(t)}{m(t)} \frac{m(t)}{\bar{m}(t)} + \frac{\varepsilon(t)}{m(t)} \in L_2, \quad \frac{m(t)}{\bar{m}(t)} \in L_\infty \Rightarrow \frac{\varepsilon(t)}{\bar{m}(t)} \in L_2.$$

Since $\mathbf{P}(t) = \mathbf{P}(t)^T$ is bounded and $\mathbf{P}(t) = \mathbf{P}_s(t)\mathbf{P}_s(t)$ ($\mathbf{P}_s(t)$ is also bounded) we have

$$\| \dot{\theta}(t) \| = \left\| \varepsilon(t) \frac{\mathbf{P}(t)\phi(t)}{m(t)^2} \right\| = \left\| \frac{\mathbf{P}(t)\phi(t)}{\sqrt{1 + \kappa\phi(t)^T \mathbf{P}(t)\phi(t)}} \right\| \frac{\varepsilon(t)}{m(t)} =$$

$$= \left\| \frac{\mathbf{P}_s(t)\mathbf{P}_s(t)\phi(t)}{\sqrt{1 + \kappa\phi(t)^T \mathbf{P}_s(t)\mathbf{P}_s(t)\phi(t)}} \right\| \frac{\varepsilon(t)}{m(t)} = \| \mathbf{P}_s(t) \| \frac{\left\| \mathbf{P}_s(t)\phi(t) \right\|}{\sqrt{1 + \kappa \left\| \mathbf{P}_s(t)\phi(t) \right\|^2}} \frac{\varepsilon(t)}{m(t)},$$

therefore, $\dot{\theta}(t) \in L_2. \Rightarrow (iii)$
The integration of the differential equation (9) gives for \( t \geq t_0 \):

\[
P(t) = P(t_0) - \int_{t_0}^{t} \frac{P(\tau)\phi(\tau)\phi(\tau)^T P(\tau)}{m(\tau)^2} d\tau > 0 \Rightarrow P(t_0) > \int_{t_0}^{t} \frac{P(\tau)\phi(\tau)\phi(\tau)^T P(\tau)}{m(\tau)^2} d\tau.
\]

For any \( z \in \mathbb{R}^{n_\theta} \) we have \( \infty > z^T P(t_0) z > \int_{t_0}^{t} z^T \frac{P(\tau)\phi(\tau)\phi(\tau)^T P(\tau)}{m(\tau)^2} z d\tau \geq 0 \), consequently, the scalar function \( f(t, z) = \int_{t_0}^{t} z^T \frac{P(\tau)\phi(\tau)\phi(\tau)^T P(\tau)}{m(\tau)^2} z d\tau \) has properties:

- it is a nondecreasing function of \( t \geq t_0 \);
- it is upper and lower bounded,

then there exists \( f_z \in \mathbb{R} \) such that \( \lim_{t \to \infty} f(t, z) = f_z \). \( \Rightarrow \lim_{t \to \infty} P(t) = P_\infty, P_\infty \in \mathbb{R}^{n_\theta \times n_\theta} \).

Note that \( (\varepsilon(t) = \phi(t)^T \tilde{\theta}(t)) \)

\[
\dot{\tilde{\theta}} = \dot{\theta} = -\frac{P\phi}{m^2} \varepsilon = -\frac{P\phi}{m^2} \phi^T PP^{-1} \tilde{\theta} = \dot{P}P^{-1} \tilde{\theta} \Rightarrow \tilde{\theta}(t) = P(t)P(t_0)^{-1} \tilde{\theta}(t_0) \Rightarrow
\]

\[
\lim_{t \to \infty} \theta(t) = \theta^* + \lim_{t \to \infty} P(t)P(t_0)^{-1} \tilde{\theta}(t_0) = \theta^* + P_\infty P(t_0)^{-1} \tilde{\theta}(t_0) = \theta_\infty \in \mathbb{R}^{n_\theta}.
\]
Discussion:

1) The algorithm (8)–(9) can be presented in the form

\[
\dot{\theta}(t) = \dot{\theta}(t) = -\varepsilon(t)\Gamma \frac{\phi(t)}{m(t)^2} = -\Gamma \frac{\phi(t)\phi(t)^T}{m(t)^2} \tilde{\theta}(t) = B(t)\tilde{\theta}(t).
\]

thus it is a linear time-varying system!!! The same as the algorithm (6):

\[
\dot{\theta}(t) = \dot{\theta}(t) = -\varepsilon(t)\Gamma \frac{\phi(t)}{m(t)^2} = -\Gamma \frac{\phi(t)\phi(t)^T}{m(t)^2} \tilde{\theta}(t) = B(t)\tilde{\theta}(t).
\]

2) Uniform stability: \[ ||\tilde{\theta}(t)|| \leq || P(t)P(t_0)^{-1}\tilde{\theta}(t_0)|| \leq c_0 ||\tilde{\theta}(t_0)|| \] for some \( c_0 > 0 \).

3) The least-squares algorithm (8), (9) minimizes a cost function which is an integral of squared errors at many time instants with a penalty on the initial estimate \( \theta(t_0) = \theta_0 \):

\[
J(t, \theta) = \frac{1}{2} \int_{t_0}^{t} \frac{(\theta(\tau)^T \phi(\tau) - y(\tau))^2}{m(\tau)^2} d\tau + \frac{1}{2} [\theta(t) - \theta_0]^T P_0^{-1} [\theta(t) - \theta_0] =
\]

\[
= \frac{1}{2} \int_{t_0}^{t} \frac{\varepsilon(\tau)^2}{m(\tau)^2} d\tau + \frac{1}{2} \tilde{\theta}(t_0)^T P_0^{-1}\tilde{\theta}(t_0).
\]

Compare with the gradient descent algorithm (6): \( J(t, \theta) = \frac{1}{2} \frac{\varepsilon(t)^2}{m(t)^2} \).
Example 1

Plant: \[ \dot{y} = -ay + bu. \]

Estimator: \[ \hat{\theta} = -\frac{\epsilon}{m^2} \begin{bmatrix} R_{1,1}\omega_1 + R_{1,2}\omega_2 \\ P_{2,1}\omega_1 + P_{2,2}\omega_2 \end{bmatrix}, \quad \dot{\hat{P}} = -\frac{1}{m^2} \begin{bmatrix} R_{1,1}\omega_1 + R_{1,2}\omega_2 \\ P_{2,1}\omega_1 + P_{2,2}\omega_2 \end{bmatrix}^T \begin{bmatrix} R_{1,1}\omega_1 + R_{1,2}\omega_2 \\ P_{2,1}\omega_1 + P_{2,2}\omega_2 \end{bmatrix} \]

Simulation 1: \[ a = 0.5, \ b = 1, \ P_0 = 20I_2 \text{ and } u(t) = \sin(t), \]

Simulation 2: \[ a = 1.5, \ b = 2, \ P_0 = 50I_2 \text{ and } u(t) = \sin(t), \]
Simulation 3: $a = 1.5$, $b = 2$, $P_0 = 50I_2$ and $u(t) = 1 - e^{-t} \cos(t)$, $\sigma(t) = \|P(t)\|$, 

Conclusions:
- the rate of convergence in the algorithm (8), (9) is a more complex issue than in (6);
- the convergence of adjusted estimates $\theta(t)$ to their ideal values $\theta^*$ depends on the input $u$;
- $y, u$ are oscillating $\Rightarrow \theta(t) \rightarrow \theta^*$; $y \rightarrow \text{const}$, $u \rightarrow \text{const}$ (set-point) $\Rightarrow \theta(t) \nrightarrow \theta^*$. 

EDSYS
e. Discrete-time version of adaptive algorithms

Continuous time \( t \geq t_0 \Rightarrow \) Discrete time \( t \in \{t_0, t_0 + T, t_0 + 2T\ldots\} \), \( T > 0 \) is the period.

The normalized gradient algorithm:

\[
\theta(t + 1) = \theta(t) - \Gamma \frac{\phi(t) \varepsilon(t)}{m(t)^2}, \quad \theta(t_0) = \theta_0, \quad 2I_{n\theta} > \Gamma = \Gamma^T > 0,
\]

\[
m(t) = \sqrt{\kappa + \phi(t)^T \phi(t)}, \quad \kappa > 0.
\]

The normalized least-squares algorithm:

\[
\theta(t + 1) = \theta(t) - P(t - 1) \frac{\phi(t) \varepsilon(t)}{m(t)^2}, \quad \theta(t_0) = \theta_0,
\]

\[
P(t) = P(t - 1) - P(t - 1) \frac{\phi(t) \phi(t)^T}{m(t)^2} P(t - 1), \quad P(t_0 - 1) = P_0 = P^T_0 > 0,
\]

\[
m(t) = \sqrt{\kappa + \phi(t)^T P(t - 1) \phi(t)}, \quad \kappa > 0.
\]

Properties:

- \( \theta(t), \varepsilon(t)/m(t), \varepsilon(t)/\bar{m}(t) \) and \( P(t) = P(t)^T > 0 \) are bounded;
- \( \varepsilon(t)/m(t), \varepsilon(t)/\bar{m}(t) \) and \( \theta(t + 1) - \theta(t) \) belong to \( L_2 \).
3. IDENTIFICATION AND ROBUSTNESS

- identification ⇔ parameter convergence;
- robustness ⇔ \|d\| ≠ 0, \|v\| ≠ 0.

**a. Parametric convergence and persistency of excitation**

**Lemma 3.** For the gradient algorithm (6) or least-squares algorithm (8)–(9), if \(m(t) \in L_\infty\) and \(\dot{\phi}(t) \in L_\infty\), then \(\lim_{t \to \infty} \epsilon(t) = 0\).

**Proof.** \(\frac{\epsilon(t)}{m(t)} \in L_2 \cap L_\infty\) and \(\tilde{\theta}(t), \dot{\tilde{\theta}}(t) \in L_\infty\) from lemmas 1, 2. Since \(\epsilon(t) = \phi(t)^T \tilde{\theta}(t)\) we have

\[\dot{\epsilon}(t) = \phi(t)^T \dot{\tilde{\theta}}(t) + \phi(t)^T \ddot{\tilde{\theta}}(t)\]. Hence: \(\dot{\phi}(t) \in L_\infty \Rightarrow \dot{\epsilon}(t) \in L_\infty\), \(m(t) \in L_\infty \Rightarrow \epsilon(t) \in L_2 \cap L_\infty\).

Under conditions of lemma 3 asymptotically \(\epsilon(t) = \sum_{i=1}^{n_\theta} [\theta_i(t) - \theta_i^*] \phi_i(t) = 0, t \geq t_1\):

a) \(\phi(t) = [1, 0, \ldots, 0]^T \Rightarrow \theta_1(t) - \theta_1^* = 0, \theta_i(t) \) for \(2 \leq i \leq n_\theta\) —?

b) \(\phi(t) = [1, 1, \ldots, 1]^T \Rightarrow \sum_{i=1}^{n_\theta} [\theta_i(t) - \theta_i^*] = 0\) —?

c) \(\phi_i(t) = \sin(\omega it), i = 1, \ldots, n_\theta, \omega > 0 \Rightarrow \sum_{i=1}^{n_\theta} [\theta_i(t) - \theta_i^*] \sin(\omega it) = 0 \Rightarrow \theta_i(t) = \theta_i^*, i = 1, \ldots, n_\theta\).
Definition 1. A bounded vector signal \( \varphi(t) \in \mathbb{R}^q, q \geq 1 \), is exciting over the finite time interval \([\sigma_0, \sigma_0 + \delta_0], \delta_0 > 0, \sigma_0 \geq t_0\), if for some \( \alpha_0 > 0 \)

\[
\int_{\sigma_0}^{\sigma_0 + \delta_0} \varphi(\tau)\varphi(\tau)^T d\tau \geq \alpha_0 I_q.
\]

\[\square\]

Definition 2. A bounded vector signal \( \varphi(t) \in \mathbb{R}^q, q \geq 1 \), is Persistently Exciting (PE) if there exist \( \delta > 0 \) and \( \alpha > 0 \) such that

\[
\int_{\sigma}^{\sigma + \delta} \varphi(\tau)\varphi(\tau)^T d\tau \geq \alpha I_q, \ \forall \sigma \geq t_0.
\]

\[\square\]

\( \varphi(t) \in \mathbb{R}^q \) is PE \( \Leftrightarrow \exists \rho > 0, \delta > 0: \int_{t_0}^{t} \varphi(\tau)\varphi(\tau)^T d\tau \geq \rho(t-t_0)I_q, \ \forall t \geq t_0 + \delta \)

(positive definite in average).

The idea: \( \text{rank}[\varphi(t)\varphi(t)^T] = 1, \ t \geq t_0 \Rightarrow \text{rank}[\int_{t_0}^{t} \varphi(\tau)\varphi(\tau)^T d\tau] = q. \)

Example 2.

\( \varphi(t) = [1,1]^T \Rightarrow \varphi(t)\varphi(t)^T = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \Rightarrow \int_{0}^{\delta} \varphi(\tau)\varphi(\tau)^T d\tau = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \delta \geq 0 \Rightarrow \text{not PE}. \)
\[
\varphi(t) = [1, e^{-t}]^T \Rightarrow \int_0^\delta \varphi(\tau) \varphi(\tau)^T \, d\tau = \begin{bmatrix}
\delta & -e^{-\delta} \\
-e^{-\delta} & -0.5e^{-2\delta}
\end{bmatrix}
\Rightarrow \text{exciting over some finite intervals.}
\]

\[
\varphi(t) = [1, \sin(t)]^T \Rightarrow \int_0^\delta \varphi(\tau) \varphi(\tau)^T \, d\tau = \begin{bmatrix}
\delta & -\cos(\delta) \\
-\cos(\delta) & -0.5\delta - 0.25\sin(2\delta)
\end{bmatrix} \geq \lambda(\delta)I_2,
\]

\[
\lambda(\delta) = \frac{6\delta - \sin(2\delta)}{8} - \sqrt{\frac{[2\delta + \sin(2\delta)]^2}{64} + \cos(\delta)^2} \geq \rho\delta, \rho = 0.4 \text{ for } \delta > 5:
\]

\[
\varphi(t) = \begin{bmatrix}
\cos(t) \\
\sin(t)
\end{bmatrix} \Rightarrow \int_0^\delta \varphi(\tau) \varphi(\tau)^T \, d\tau = \frac{1}{2} \begin{bmatrix}
\delta + 0.5\sin(2\delta) & \sin(\delta)^2 \\
\sin(\delta)^2 & \delta - 0.5\sin(2\delta)
\end{bmatrix} \geq \frac{1}{2} [\delta - \sin(\delta)]I_2 \Rightarrow \text{PE.}
\]
Normalized gradient algorithm (6) \( (\tilde{\theta}(t) = \theta(t) - \theta^*, \, \varepsilon(t) = \phi(t)^T \tilde{\theta}(t)) \):

\[
\dot{\tilde{\theta}}(t) = -\varepsilon(t) \Gamma \frac{\phi(t)}{m(t)^2} = -\Gamma \frac{\phi(t)}{m(t)^2} \phi(t)^T \tilde{\theta}(t) = B(t) \tilde{\theta}(t), \, B(t) = -\Gamma \frac{\phi(t)\phi(t)^T}{m(t)^2}.
\]

Let \( \Phi(t_0, t) \) be the state transition matrix of the linear time-varying system (6), then

\[ -\tilde{\theta}(t) = \Phi(t_0, t)\tilde{\theta}(t_0); \]
\[ -\text{PE } \phi(t) \Rightarrow \phi(t) / m(t), \, m(t) = \sqrt{1 + \kappa \phi(t)^T \phi(t)} \text{ is PE } \Rightarrow \eta(t) = \Phi(t_0, t)^T \phi(t) / m(t) \text{ is PE:} \]

\[ \exists \rho > 0, \, \delta > 0: \int_{t_0}^{t} \eta(\tau)\eta(\tau)^T d\tau \geq \rho (t-t_0) I_{n_{\theta}}, \, \forall t \geq t_0 + \delta. \]

Consider the Lyapunov function \( V(\tilde{\theta}) = \tilde{\theta}^T \Gamma^{-1} \tilde{\theta} \):

\[
\dot{V} = -2 \frac{\varepsilon(t)^2}{m(t)^2} = -2\tilde{\theta}(t)^T \frac{\phi(t)\phi(t)^T}{m(t)^2} \tilde{\theta}(t) = -2\tilde{\theta}_0^T \eta(t)\eta(t)^T \tilde{\theta}_0,
\]

integrating this equality for \( t \geq t_0 + \delta \) we obtain \( (V(t_0) = \tilde{\theta}_0^T \Gamma^{-1} \tilde{\theta}_0) \):

\[ V(t) = V(t_0) - 2\tilde{\theta}_0^T \int_{t_0}^{t} \eta(\tau)\eta(\tau)^T d\tau \tilde{\theta}_0 \leq V(t_0) - 2\rho (t-t_0)\tilde{\theta}_0^T \tilde{\theta}_0 = \tilde{\theta}_0^T [\Gamma^{-1} - 2\rho (t-t_0)]\tilde{\theta}_0 \Rightarrow \]

\[ \lim_{t \to \infty} V(t) = 0 \Rightarrow \lim_{t \to \infty} \theta(t) = \theta^*. \]
Normalized least-squares algorithm (8)–(9):

\[ \tilde{\theta}(t) = P(t)P(t_0)^{-1}\tilde{\theta}(t_0), \; t \geq t_0. \]

Properties:

- \( \lim_{t \to \infty} P(t) = 0 \iff \lim_{t \to \infty} \tilde{\theta}(t) = 0 \);
- \( P(t) = P(t_0) - \int_{t_0}^{t} \frac{P(\tau)\phi(\tau)\phi(\tau)^T P(\tau)}{m(\tau)^2} d\tau, \; P(t) = P^T(t) > 0 \) for all \( t \geq t_0 \);
- \( \phi(t) \) is PE \( \Rightarrow \phi(t)/m(t), \; m(t) = \sqrt{1 + \kappa \phi(t)^T \phi(t)} \) is PE \( \Rightarrow \eta(t) = P(t)\phi(t)/m(t) \) is PE:
  \[ \exists \rho > 0, \; \delta > 0: \int_{t_0}^{t} \eta(\tau)\eta(\tau)^T d\tau \geq \rho(t-t_0)I_{n_\theta}, \; \forall t \geq t_0 + \delta. \]

Then

\[ 0 < P(t) = P(t_0) - \int_{t_0}^{t} \eta(\tau)\eta(\tau)^T d\tau \leq P(t_0) - \rho(t-t_0)I_{n_\theta} \leq 0 \text{ for some } t \geq t_0 \Rightarrow \]

\[ \lim_{t \to \infty} \theta(t) = \theta^*. \]

**Lemma 4.** For the gradient algorithm (6) or least-squares algorithm (8)–(9), if \( \phi(t) \) is PE, then \( \lim_{t \to \infty} \theta(t) = \theta^*. \)
**Discussion:**

What is PE property of the regressor $\phi(t)$:

$$\phi(t) = \left[ \{ C_m \omega_1(t) \}^T, \omega_2(t)^T \right]^T,$$

where $\omega_1(t) \in \mathbb{R}^n$, $\omega_2(t) \in \mathbb{R}^n$ and for a Hurwitz matrix $A_\lambda$:

$$\dot{\omega}_1(t) = A_\lambda \omega_1(t) + bu(t),$$
$$\dot{\omega}_2(t) = A_\lambda \omega_2(t) + by(t).$$

PE of $\phi(t) \leftarrow$ PE of $\omega_1(t)$ and $\omega_2(t) \leftarrow$ PE of $u(t)$ and $y(t)$.

(2) is a linear system $\Rightarrow$ PE of $y(t)$ is determined by the input $u(t)$!

**PE of $u(t) \Rightarrow$ PE of $\phi(t)$**

(that we already observed in the example).
Example 1

Plant: \[ \dot{y} = -ay + bu, \]
\[ a = 0.5, \ b = 1 \text{ and } u(t) = \sin(t). \]

Gradient algorithm: \[ \gamma = 20 \]

Least-squares algorithm: \[ P_0 = 20I_2 \]

\[ u(t) = \sin(t) \Rightarrow y(t) = \alpha \sin(t + \beta) \Rightarrow \omega_i(t) = \alpha_i \sin(t + \beta_i) \text{ due to } \dot{\omega}_1(t) = -\omega_1(t) + u(t), \dot{\omega}_2(t) = -\omega_2(t) + y(t) \]

\[ \phi(t) = [\omega_1(t), \omega_2(t)]^T \Rightarrow \varphi(t) = [\cos(t), \sin(t)]^T \Rightarrow \text{PE.} \]
b. Robustness of adaptive algorithms

Before the noise free case with $d(t) = 0$ and $v(t) = 0$ has been considered for

$$\Sigma_l : \begin{cases} \dot{x} = A(\theta)x + B(\theta)u + d; \\ y = Cx + Du + v. \end{cases}$$

**What happens if $d(t) \neq 0$ or $v(t) \neq 0$?**

(only the case $d(t) \neq 0$ will be considered)

**Example 1**

Plant:

$$\dot{y} = -ay + bu + d(t),$$

$a = 1.5$, $b = 2$ and $u(t) = \sin(t)$, $d(t) = 0.5\sin(3t)$. 

$\phi(t)$ is PE $\Rightarrow$ Robustness!!!
\[ u(t) = 1 - e^{-t} \cos(t) \]

\[ u(t) = \sin(t), \quad d(t) = 0.5 \sin(t) \]

**Conclusion:** the disturbance can seriously modify the system behavior.
Linear parametric model with modeling errors:

\[ y(t) = \phi(t)^T \theta^* + \delta(t), \quad t \geq t_0, \]

where \( \theta^* \in \mathbb{R}^{n_\theta} \) is an unknown parameter vector, \( \phi(t) \in \mathbb{R}^{n_\theta} \) is a known regressor, \( y(t) \in \mathbb{R} \) is a measured output, \( \delta(t) \in \mathbb{R} \) represents system modeling errors:

\[ |\delta(t)| \leq c_1 \| \phi(t) \| + c_2, \quad c_1 > 0, \quad c_2 > 0. \]

Let \( \theta(t) \in \mathbb{R}^{n_\theta} \) be the estimate of \( \theta^* \) and define the estimation error

\[ \varepsilon(t) = \phi(t)^T \theta(t) - y(t) = \phi(t)^T \tilde{\theta}(t) + \delta(t), \quad t \geq t_0, \]

where \( \tilde{\theta}(t) = \theta(t) - \theta^* \) is the parametric error.

Modified gradient algorithm (6):

\[ \dot{\theta}(t) = -\varepsilon(t) \Gamma \frac{\phi(t)}{m(t)^2} + \Gamma f(t), \quad \theta(0) = \theta_0, \quad m(t) = \sqrt{1 + \kappa \phi(t)^T \phi(t)}, \quad \kappa > 0, \quad t \geq t_0, \quad (12) \]

where \( \Gamma = \Gamma^T > 0 \) is a design matrix gain, \( f(t) \in \mathbb{R}^{n_\theta} \) is the modification term for robustness.
Stability & robustness analysis for nonlinear systems ⇔ Lyapunov function theory

\[ V(\tilde{\theta}) = \frac{1}{2} \tilde{\theta}^T \Gamma^{-1} \tilde{\theta}, \quad \dot{V} = -\frac{\varepsilon(t)^2}{m(t)^2} + \frac{\varepsilon(t)\delta(t)}{m(t)^2} + \tilde{\theta}^T f(t) \]

Note:

\[ \frac{|\delta(t)|}{m(t)} \leq \frac{c_1 \|\phi(t)\| + c_2}{\sqrt{1 + \kappa \phi(t)^T \phi(t)}} \leq \frac{c_1}{\sqrt{\kappa}} + \frac{c_2}{m(t)} \leq \frac{c_1}{\sqrt{\kappa}} + c_2. \]

Then

\[ \dot{V} \leq -\frac{\varepsilon(t)^2}{m(t)^2} + \left[ \frac{c_1}{\sqrt{\kappa}} + \frac{c_2}{m(t)} \right] \frac{|\varepsilon(t)|}{m(t)} + \tilde{\theta}^T f(t), \]

and

\[ \frac{|\varepsilon(t)|}{m(t)} \geq \frac{c_1}{\sqrt{\kappa}} + \frac{c_2}{m(t)} \Rightarrow -\frac{\varepsilon(t)^2}{m(t)^2} \leq -\left[ \frac{c_1}{\sqrt{\kappa}} + \frac{c_2}{m(t)} \right] \frac{|\varepsilon(t)|}{m(t)} \Rightarrow \dot{V} \leq \tilde{\theta}^T f(t). \]

The simplest modification:

\[ f(t) = \frac{\phi(t)}{m(t)^2} f_s(t), \quad f_s(t) = \begin{cases} 0 & \text{if } |\varepsilon(t)| / m(t) \geq c_1 / \sqrt{\kappa} + c_2 / m(t), \Rightarrow \dot{V} \leq 0. \\ \varepsilon(t) & \text{otherwise.} \end{cases} \Rightarrow \dot{\theta}(t) = 0! \]
The simplest modification:

\[ f(t) = \frac{\phi(t)}{m(t)^2} f_s(t), \quad f_s(t) = \begin{cases} \varepsilon(t) & \text{if } |\varepsilon(t)| / m(t) < \frac{c_1}{\sqrt{\kappa}} + \frac{c_2}{m(t)}, \\ 0 & \text{otherwise.} \end{cases} \Rightarrow \dot{V} \leq 0. \]

A dead zone modification:

\[ f(t) = \frac{\phi(t)}{m(t)^2} f_d(t), \quad f_d(t) = \begin{cases} \varepsilon(t) & \text{if } |\varepsilon(t)| / m(t) < \frac{c_1}{\sqrt{\kappa}} + \frac{c_2}{m(t)}, \\ \left[\frac{c_1 m(t)}{\sqrt{\kappa} + c_2}\right] \text{sign}[\varepsilon(t)] & \text{otherwise.} \end{cases} \Rightarrow \dot{V} \leq 0. \]

\[ \dot{\theta}(t) = -\Gamma \frac{\phi(t)}{m(t)^2} [\varepsilon(t) - f_d(t)] \Rightarrow \]
**Projection:** assume that the set of admissible values for $\theta^*$ is given, i.e.

$$\theta^* \in \Omega = \{ \theta \in \mathbb{R}^{n\theta} : \| \theta \| \leq M \}, \ M > 0.$$  

Projection has to ensure that $\theta(t) \in \Omega$ for all $t \geq t_0$, therefore

$$f(t) = \begin{cases} 0 & \text{if } \| \theta(t) \| < M \text{ or } \| \theta(t) \| = M \text{ and } \theta(t)^T \Gamma \frac{\phi(t)}{m(t)^2} \epsilon(t) \leq 0, \\ \frac{\Gamma \theta(t) \theta(t)^T}{{\theta(t)}^T \Gamma \theta(t)} \Gamma \frac{\phi(t)}{m(t)^2} \epsilon(t) & \text{otherwise.} \end{cases}$$

Inside the circle doing nothing.

On an attempt to exit the circle.
The properties:

- boundedness of $\theta(t)$, $\dot{\theta}(t)$ and $\epsilon(t)/m(t)$ (belong to $L_\infty$);
- $\epsilon(t)/m(t)$ and $\dot{\theta}(t)$ belong to $L_2$;
- in the noise-free case ($d(t) = 0$) the quality is preserved? $\Rightarrow$ **ESTIMATION?**

**Example 1**

Plants:

$$\dot{y} = -ay + bu + d(t),$$

$a = 0.5$, $b = 1$ and $u(t) = u(t) = 1 - e^{-t} \cos(t)$; (6) with $\gamma = 20$.

$d(t) = 0.5 \sin(0.3t)$
Dead zone algorithm:

\[ \theta_2 \]

\[ \theta_1 \]

\[ \theta_2 \]

\[ \theta_1 \]

\[ \theta_2 \]

\[ \theta_1 \]

\[ d(t) = 0.5 \sin(0.3t) \]

σ-Modification (\( \sigma = 0.01 \)):

Projection (\( M = 1.5 \)):
Dead zone algorithm:

\[
\theta_1 - \sigma \leq x(t) \leq \theta_2 + \sigma
\]

\(\theta_1\) and \(\theta_2\) are the limits of the dead zone.

\(\sigma\)-Modification (\(\sigma = 0.01\)):

Projection (\(M = 1.5\)):
1. Adaptive parameter estimation:

   a. Parameterized system model \( y(t) = \phi(t)^T \theta^* \).

   b. Linear parametric model \( \varepsilon(t) = \phi(t)^T \theta(t) - y(t) = \phi(t)^T \tilde{\theta}(t), \ \tilde{\theta}(t) = \theta(t) - \theta^* \).

   c. Normalized gradient algorithm \( \dot{\theta}(t) = -\varepsilon(t) \Gamma \frac{\phi(t)}{m(t)^2} \).

   d. Normalized least-squares algorithm \( \dot{\theta}(t) = -\varepsilon(t) \frac{P(t)\phi(t)}{m(t)^2}, \ \dot{P}(t) = -\frac{P(t)\phi(t)\phi(t)^T P(t)}{m(t)^2} \).

   e. Discrete-time version of adaptive algorithms.

2. Identification and robustness:

   f. Parametric convergence and PE (PE \(\Rightarrow\) convergence/estimation \(\Rightarrow\) robustness).

   g. Robustness of adaptive algorithms (robustness \(\Leftrightarrow\) estimation).
Example 2

Oscillating pendulum:

\[ \varphi \in [-\pi, \pi] \] is the pendulum angle, \( f \in \mathbb{R} \) is the (controlling or exciting) input applied to the support, \( d \in \mathbb{R} \) is the disturbance influencing the support also.

Nonlinear model:

\[
\ddot{y} = -\omega^2 \sin(y) - \rho \dot{y} + b \cos(y) f(t) + d(t),
\]

\[ y = \varphi \in [-\pi, \pi] \] is the measured angle, \( \dot{y} \in \mathbb{R} \) and \( \ddot{y} \in \mathbb{R} \) are the angle velocity and acceleration; \( \rho > 0 \) is an unknown friction coefficient, \( \omega > 0 \) is an unknown natural frequency, \( b > 0 \) is an unknown control gain.

3 unknown parameters + nonlinearity. \( \Rightarrow \) Define \( u_1 = \sin(y) \) and \( u_2 = \cos(y)u \):

\[
\ddot{y} + \rho \dot{y} = -\omega^2 u_1(t) + bu_2(t) + d(t) \Rightarrow (2) \text{ for } n = 2, \ m = 1 \text{ and a vector } \mathbf{u} = [u_1, u_2]^T.
\]
Define the polynomials:

\[ P(s) = s^2 + p_1 s, \quad p_1 = \rho; \quad Z_1(s) = z_{1,0} = -\omega^2; \quad Z_2(s) = z_{2,0} = b, \]
then the noise-free model (13) has the form \( P(s)[y](t) = Z_1(s)[u_1](t) + Z_2(s)[u_2](t) \).

Parameterization for \( \Lambda(s) = s^2 + \lambda_1 s + \lambda_0 \):

\[
\frac{P(s)}{\Lambda(s)}[y](t) = \frac{Z_1(s)}{\Lambda(s)}[u_1](t) + \frac{Z_2(s)}{\Lambda(s)}[u_2](t) \Rightarrow
\]

\[
(1 - \frac{\Lambda(s)}{\Lambda(s)})[y](t) + \frac{P(s)}{\Lambda(s)}[y](t) = \frac{Z_1(s)}{\Lambda(s)}[u_1](t) + \frac{Z_2(s)}{\Lambda(s)}[u_2](t) \Rightarrow
\]

\[
y(t) = \frac{\Lambda(s) - P(s)}{\Lambda(s)}[y](t) + \frac{Z_1(s)}{\Lambda(s)}[u_1](t) + \frac{Z_2(s)}{\Lambda(s)}[u_2](t) \Rightarrow \tilde{y}(t) = \phi(t)^T \theta^*,
\]

the parameterized system model for \( \tilde{y}(t) = y(t) - \lambda_0 \Lambda^{-1}(s)[y](t), \quad \theta^* = [\lambda_1 - \rho, -\omega^2, b]^T \) and

\[
\phi(t) = \left[ \frac{s}{\Lambda(s)}[y](t), \frac{1}{\Lambda(s)}[u_1](t), \frac{1}{\Lambda(s)}[u_2](t) \right]^T = [\omega_{0,2}, \omega_{1,1}, \omega_{2,1}]^T, \quad \tilde{y}(t) = y(t) - \lambda_0 \omega_{0,1}(t),
\]

\[
\dot{\omega}_0(t) = A_\lambda \omega_0(t) + b y(t), \quad A_\lambda = \begin{bmatrix} 0 & 1 \\ -\lambda_0 & -\lambda_1 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.
\]
\( \omega = 1, \rho = 0.1, b = 0.5, f(t) = \sin(3t), \lambda_0 = 1, \lambda_1 = 2, \gamma = 100. \)

\[ d(t) = 0 \]

\[ d(t) = 0.5 \sin(0.3t) \]

**Normalized gradient algorithm**

**Dead zone modification**
4. INDIRECT ADAPTIVE CONTROL

Adjustment of control parameters:
- *direct* (from an adaptive control law/Lyapunov analysis);
- *indirect* (from adaptive estimates of the system parameters).

Indirect adaptive control design:
1) adaptive estimation of the plant parameters; 2) calculation of control parameters.

a. Model reference control

The main steps:
1) adaptive estimation algorithm design;
2) reference model selection;
3) controller structure construction;
4) controller parameter calculation;
5) stability and robustness analysis.
Example 1

Plant: \[ \dot{y} = -ay + bu + d. \]

Adaptive estimation algorithm (\( \theta^* = [\theta_1^*, \theta_2^*]^T = [b, 1-a]^T \)):

\[
\dot{\theta}(t) = -\gamma \frac{\varepsilon(t)}{m(t)^2} \begin{bmatrix} \omega_1(t) \\ \omega_2(t) \end{bmatrix}, \quad m(t) = \sqrt{1 + \omega_1^2(t) + \omega_2^2(t)}, \quad \dot{\omega}_1(t) = -\omega_1(t) + u(t), \quad \dot{\omega}_2(t) = -\omega_2(t) + y(t), \quad A_\lambda = -1, \quad b = 1.
\]

Reference model: \[ \dot{y}_m = -a_m y_m + b_m r(t) \]
where \( r(t) \in \mathbb{R} \) is the reference signal to be tracked, \( a_m > 0 \) (the reference model is stable).

Controller structure: \[ u = b^{-1}[(a-a_m)y + b_m r] \Rightarrow \dot{y} = -a_m y + b_m r + d. \]

Controller parameter calculation:

\[ u = \theta_1^c y + \theta_2^c r, \quad \theta_1^c = \theta_1^{-1}(1-\theta_2-a_m), \quad \theta_2^c = \theta_1^{-1}b_m. \]

Division on \( \theta_1 \) \( \Rightarrow \) projection modification of the adaptation algorithm:

\[
\dot{\theta}(t) = -\gamma \frac{\varepsilon(t)}{m(t)^2} \begin{bmatrix} \omega_1(t) \\ \omega_2(t) \end{bmatrix} + \begin{bmatrix} f_1(t) \\ 0 \end{bmatrix}, \quad f_1(t) = \begin{cases} 0 & \text{if } \theta_1(t) > b_{\min} \text{ or } \theta_1(t) = b_{\min} \text{ and } \varepsilon(t)\omega_1(t) \geq 0, \\ \gamma \varepsilon(t)\omega_1(t)m(t)^{-2} & \text{otherwise}. \end{cases}
\]

\( b_{\min} > 0 \) is the low bound for \( b \), i.e. \( b \geq b_{\min} \).
\[ a = 1.5, \ b = 2, \ a_m = 1, \ b_m = 1, \ b_{\text{min}} = 0.1 \]

\[ d(t) = 0 \]

\[ d(t) = 0.5 \sin(3t) \]

\[ d(t) = 0.5 \sin(0.3t) \]
The general procedure:

\[ P(s)[y](t) = k_p Z(s)[u](t) + d(t), \quad t \geq 0, \]  

(14)

\( y(t) \in \mathbb{R}, \ u(t) \in \mathbb{R} \) are the measured output and input as before;

\[ P(s) = s^n + p_{n-1}s^{n-1} + \ldots + p_1s + p_0, \quad Z(s) = s^m + z_{m-1}s^{m-1} + \ldots + z_1s + z_0, \]

\( k_p, \ p_i, \ i = 0, n-1 \) and \( z_j, \ j = 0, m-1 \) are the unknown but constant parameters.

**Assumption 1.** The constant \(| k_p | \geq k_{\text{min}} > 0 \) and \( \text{sign}(k_p) \) are given. \( \Rightarrow \) Necessary.

**Assumption 2.** \( k \leq k_p \leq \bar{k}; \ p_i \leq \bar{p}_i, \ i = 0, n-1; \ z_j \leq \bar{z}_j, \ j = 0, m-1. \) \( \Rightarrow \) Desired.

1) Adaptive estimation algorithm design:

\[ y(t) = k_p \frac{Z(s)[u](t) + \Lambda(s) - P(s)}{\Lambda(s)} [y](t), \quad \Lambda(s) = s^n + \lambda_{n-1}s^{n-1} + \ldots + \lambda_1s + \lambda_0 \Rightarrow \]

\[ y(t) = \theta^T \phi(t), \quad \theta^* = [k_p z_0, \ldots, k_p z_{m-1}, k_p, \lambda_0 - p_0, \ldots, \lambda_{n-1} - p_{n-1}]^T, \]

\[ \phi(t) = [\{C_m \omega_1(t)\}^T, \omega_2(t)^T]^T, \quad C_m = [I_{m+1}, \theta_{(m+1) \times (n-m-1)}], \]

\[ \dot{\omega}_1(t) = A_\lambda \omega_1(t) + bu(t), \]

\[ \dot{\omega}_2(t) = A_\lambda \omega_2(t) + by(t). \]  

(15)
Normalized gradient algorithm with projection (assumption 2):

\[
\dot{\theta}(t) = g(t) + f(t), \quad \theta(0) = \theta_0, \ t \geq 0, \\
g(t) = -\varepsilon(t) \Gamma \frac{\phi(t)}{m(t)^2}, \ \varepsilon(t) = \theta(t)^T \phi(t) - y(t), \ m(t) = \sqrt{1 + \kappa \phi(t)^T \phi(t)},
\]

\[
f_k(t) = \begin{cases} 
0 & \text{if } \theta_k < \theta_k(t) < \bar{\theta}_k \text{ or } \theta_k(t) = \theta_k \text{ and } g_k(t) \geq 0 \text{ or } \theta_k(t) = \bar{\theta}_k \text{ and } g_k(t) \leq 0, \\
-g_k(t) & \text{otherwise},
\end{cases}
\]

Properties:

\[
\theta(t), \dot{\theta}(t), \varepsilon(t) / m(t) \in L_{\infty} \text{ and } \dot{\theta}(t), \varepsilon(t) / m(t) \in L_2.
\]

2) Reference model selection:

\[
P_m(s)[y_m](t) = r(t),
\]

where \(P_m(s)\) is a stable polynomial of degree \(n - m\) and \(r(t)\) is a bounded and piecewise continuous reference input signal.
3) Controller structure construction:

\[ u(t) = \omega_1^c(t)^T \theta_1^c + \omega_2^c(t)^T \theta_2^c + \theta_3^c y(t) + \theta_4^c r(t), \]  

where \( \theta_1^c \in \mathbb{R}^n \), \( \theta_2^c \in \mathbb{R}^n \), \( \theta_3^c \in \mathbb{R} \), \( \theta_4^c \in \mathbb{R} \) are the controller parameters,

\[ \omega_1^c(t) = \frac{a(s)}{\Lambda_c(s)} [u](t), \quad \omega_2^c(t) = \frac{a(s)}{\Lambda_c(s)} [y](t), \quad a(s) = [1, s, \ldots, s^{n-2}]^T, \]

and \( \Lambda_c(s) = s^{n-1} + \lambda^c_{n-2} s^{n-2} + \ldots + \lambda^c_1 s + \lambda^c_0 \) is a stable polynomial. A variant of realization:

\[ \dot{\omega}_1^c(t) = A^c_\lambda \omega_1^c(t) + b^c u(t), \quad A^c_\lambda = \begin{bmatrix} 0 & 1 & 0 & \ldots & 0 & 0 \\ 0 & 0 & 1 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & \ldots & 0 & 1 \\ -\lambda^c_0 & -\lambda^c_1 & \ldots & \ldots & -\lambda^c_{n-3} & -\lambda^c_{n-2} \end{bmatrix}, \quad b^c = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}. \]
The controller parameter equation:

\[
a(s)^T \theta_1^c P(s) + [a(s)^T \theta_2^c + \theta_3^c \Lambda_c(s)] k_p Z(s) = \Lambda_c(s)[P(s) - k_p \theta_4^c Z(s) P_m(s)].
\]  

(19)

Multiply (19) on \( y(t) \) and substitute (14) for the case \( d(t) = 0 \):

\[
a(s)^T \theta_1^c P(s)[y](t) + [a(s)^T \theta_2^c + \theta_3^c \Lambda_c(s)] k_p Z(s)[y](t) =
\]

\[
= \Lambda_c(s) P(s)[y](t) - k_p \theta_4^c \Lambda_c(s) Z(s) P_m(s)[y](t),
\]

\[
\Rightarrow
\]

\[
a(s)^T \theta_1^c k_p Z(s)[u](t) + [a(s)^T \theta_2^c + \theta_3^c \Lambda_c(s)] k_p Z(s)[y](t) =
\]

\[
= \Lambda_c(s) k_p Z(s)[u](t) - k_p \theta_4^c \Lambda_c(s) Z(s) P_m(s)[y](t).
\]

Now divide both sides on \( \Lambda_c(s) k_p Z(s) \) (\( Z(s) \) and \( \Lambda_c(s) \) are stable polynomials):

\[
\frac{a(s)^T \theta_1^c}{\Lambda_c(s)} [u](t) + \frac{a(s)^T \theta_2^c + \theta_3^c \Lambda_c(s)}{\Lambda_c(s)} [y](t) = u(t) - \theta_4^c P_m(s)[y](t),
\]

\[
\Rightarrow
\]

\[
\omega_1^c(t)^T \theta_1^c + \omega_4^c(t)^T \theta_2^c + \theta_3^c y(t) = u(t) - \theta_4^c P_m(s)[y](t).
\]

Substitution of the control (18) gives

\[
\theta_4^c P_m(s)[y](t) = \theta_4^c r(t) \Rightarrow \theta_4^c P_m(s)[y](t) = \theta_4^c P_m(s)[y_m](t) \Rightarrow P_m(s)\{[y](t) - [y_m](t)\} = 0.
\]
4) Controller parameter calculation:

\[ \theta_4^c = k_p^{-1} \Rightarrow B(s) = \Lambda_c(s)\{P(s) - Z(s)P_m(s)\}, \text{ then (19) takes the form:} \]

\[ a(s)^T \theta_1^c P(s) + \left[a(s)^T \theta_2^c + \theta_3^c \Lambda_c(s)\right] k_p Z(s) = B(s). \]

The right hand side is a polynomial of degree \(2n-2\) with coefficients linearly dependent on \(\theta_1^c, \theta_2^c\) and \(\theta_3^c\). The left hand side is a polynomial of degree \(2n-2\) with constant coefficients.

Equating the coefficients with the same powers of \(s\) we obtain the solution:

\[ \theta_1^c = \Theta_1(p_{n-1}, ..., p_0; z_{m-1}, ..., z_0; \lambda_{n-2}, ..., \lambda_0), \theta_2^c = \Theta_1(p_{n-1}, ..., \lambda_0), \theta_3^c = \Theta_1(p_{n-1}, ..., \lambda_0) \]

\[ \theta_1^c = \Theta_1(\theta^*; \lambda_{n-2}, ..., \lambda_0), \theta_2^c = \Theta_1(\theta^*; \lambda_{n-2}, ..., \lambda_0), \theta_3^c = \Theta_1(\theta^*; \lambda_{n-2}, ..., \lambda_0). \]

**Example 1:**

\[ \theta_1^c = \theta_1^{-1}(1 - \theta_2 - a_m), \theta_2^c = \theta_1^{-1}b_m. \]

**Theorem 1.** Under assumption 2 and that all zeros of \(Z(s)\) are stable:

(i) \(y(t), \theta(t), \dot{\theta}(t), \omega_1(t), \omega_2(t) \in L_\infty\);

(ii) \(y(t) - y_m(t) \in L_2, \lim_{t \to \infty}[y(t) - y_m(t)] = 0.\)
b. Pole placement control

The pole placement equation:

\[ A^*(s) = C(s)Q(s)P(s) + D(s)Z(s), \]  

(20)

where \( A^*(s) \) is the desired polynomial of the closed loop system; \( C(s) \) and \( D(s) \) are polynomials of the pole placement control:

\[ u(t) = \{ \Lambda_c(s) - C(s)Q(s) \} \Lambda_c^{-1}(s)[u](t) + D(s)\Lambda_c^{-1}(s)[r - y](t), \]  

(21)

where \( r(t) \) is a bounded and piecewise continuous reference input signal, \( Q(s)[r](t) = 0 \Rightarrow \)

(a) \( r(t) = 0 \Rightarrow Q(s) = 1; \)

(b) \( r(t) = c \neq 0 \Rightarrow Q(s) = s; \)

(c) \( r(t) = ce^{-at} \Rightarrow Q(s) = s + a, \ a > 0. \)

According to (21) the control is a dynamical system:

\[ C(s)Q(s)[u](t) = D(s)[r - y](t). \]  

(22)

Controller structure \( (a_1(s) = [1, s, ..., s^{n+2}]^T) \):

\[ u(t) = \theta_1^{cT} a_1(s)\Lambda_c(s)[u](t) + \theta_2^{cT} a_1(s)\Lambda_c(s)[y - r](t) + \theta_3^c \{ y(t) - r(t) \}. \]
**Properties:**

1) multiplying both sides of (20) on \( y(t) \) we obtain:

\[
A^*(s)[y](t) = C(s)Q(s)P(s)[y](t) + D(s)Z(s)[y](t) =
\]

\[
= C(s)Q(s)P(s)[y](t) + Z(s)\{D(s)[r](t) - C(s)Q(s)[u](t)\} =
\]

\[
= Z(s)D(s)[r](t).
\]

\( r(t) \in L_\infty \) and \( A^* \) is stable \( \Rightarrow \) \( y(t) \in L_\infty \).

2) multiplying both sides of (20) on \( u(t) \) we obtain:

\[
A^*(s)[u](t) = C(s)Q(s)P(s)[u](t) + D(s)Z(s)[u](t) =
\]

\[
= P(s)D(s)[r - y](t) + D(s)Z(s)[u](t) = P(s)D(s)[r](t).
\]

\( r(t) \in L_\infty \) and \( A^* \) is stable \( \Rightarrow \) \( u(t) \in L_\infty \).

3) using (20)–(23) we get:

\[
A^*(s)[y - r](t) = 0 \Rightarrow \lim_{t \to \infty} \{y(t) - r(t)\} = 0.
\]

**Assumption 3.** \( Q(s)P(s) \) and \( Z(s) \) are coprime.

**Theorem 2.** Under assumption 3 all signals are bounded and \( \lim_{t \to \infty}[y(t) - r(t)] = 0. \)
SUMMARY

<table>
<thead>
<tr>
<th></th>
<th>Direct adaptive control</th>
<th>Indirect adaptive control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structure</td>
<td>+</td>
<td>–</td>
</tr>
<tr>
<td>Parameterization</td>
<td>–</td>
<td>+</td>
</tr>
<tr>
<td>Restrictions</td>
<td>(minimum phase)</td>
<td></td>
</tr>
</tbody>
</table>

Certainty equivalence
Example 1

Indirect adaptive control $\iff$ Robust control

Plant:

$$\dot{y} = -ay + bu + d.$$

Assumption:

$$0 < \underline{a} \leq a, \ 0 < \underline{b} \leq b + a_m > 0, \ r(t) = 0.$$

Normalized gradient descent algorithm with projection.

Robust control: $u = ky, \ k = \min\{b^{-1}(a - a_m), 0\}$.

$$a = 1.5, \ b = 2, \ \underline{a} = 0.5, \ \underline{b} = 0.1, \ a_m = 5, \ d(t) = 0, \ v(t) = 0.$$
\[ a_m = 1, \quad d(t) = 0, \quad v(t) = 0. \]

\[ a_m = 1, \quad d(t) = 5\sin(5t), \quad v(t) = 0. \]

\[ a_m = 1, \quad d(t) = 0, \quad v(t) = 0.1\sin(t). \]
5. ADAPTIVE OBSERVERS

A nonlinear system in state space presentation:

\[ \dot{x} = Ax + B(y)u + \phi(y), \quad y =Cx, \]  

(24)

\(x \in \mathbb{R}^n, \ u \in \mathbb{R}^m, \ y \in \mathbb{R}^p\) are the state, the input (control) and the measurable output;

\(A, \ C\) are constant and known, the functions \(B(y)\) and \(\phi(y)\) are continuous and known.

Everything is known except the state \(x\) (it is not measurable) \(\Rightarrow\) the state observer design:

\[ \hat{x} = A\hat{x} + B(y)u + \phi(y) + L[y - C\hat{x}], \]

\(\hat{x}\) is the estimate of \(x\); \(L\) is the observer matrix gain, \(A - LC\) is Hurwitz.

Assumption 1. \(x(t) \in L_\infty, \ u(t) \in L_\infty\) for all \(t \geq 0\).

The estimation error \(e = x - \hat{x}\):

\[ \dot{e} = \dot{x} - \dot{\hat{x}} = [Ax + B(y)u + \phi(y)] - [A\hat{x} + B(y)u + \phi(y) + L[y - C\hat{x}]] = [A - LC]e. \]

The matrix \(A - LC\) is Hurwitz (design of \(L\)) \(\Rightarrow \hat{x}(t) \in L_\infty, \ \lim_{t \to \infty} [\hat{x}(t) - x(t)] = 0.\)
A nonlinear system with parametric uncertainty:

$$\dot{x} = Ax + B(y)u + \varphi(y) + G(y,u)\theta, \quad y = Cx,$$

$$\theta \in \mathbb{R}^q$$ is the vector of unknown parameters, $$G(y,u)$$ is a known continuous function.

The adaptive observer:

$$\dot{x} = A\hat{x} + B(y)u + \varphi(y) + L[y - C\hat{x}] + G(y,u)\hat{\theta} - \Omega \dot{\theta},$$

$$\dot{\Omega} = [A - LC]\Omega - G(y,u),$$

$$\dot{\hat{\theta}} = -\gamma \Omega^T C^T[y - C\hat{x}], \quad \gamma > 0,$$

$$\hat{\theta} \in \mathbb{R}^q$$ is the estimate of $$\theta$$, $$\Omega \in \mathbb{R}^{n \times q}$$ is an auxiliary filter variable.

The state estimation error $$e = x - \hat{x}$$:

$$\dot{e} = [A - LC]e + G(y,u)[\theta - \hat{\theta}] + \Omega \dot{\theta}.$$  

$$A - LC$$ is Hurwitz + Properties of $$\hat{\theta}(t)$$ and $$\dot{\theta}(t) \Rightarrow$$ Properties of $$e(t)$$.
The **auxiliary error** $\delta = e + \Omega[\theta - \hat{\theta}]$:

$$
\dot{\delta} = \dot{e} + \dot{\Omega}[\theta - \hat{\theta}] - \Omega \dot{\theta} = \\
= \{[A - LC]e + G(y, u)[\theta - \hat{\theta}] + \Omega \dot{\theta}\} + \{[A - LC]\Omega - G(y, u)\}[\theta - \hat{\theta}] - \Omega \dot{\theta} = [A - LC] \delta.
$$

$A - LC$ is Hurwitz $\Rightarrow \delta(t) \in L_\infty$, $\lim_{t \to \infty} \delta(t) = 0$

$A - LC$ is Hurwitz + $y(t) \in L_\infty$, $u(t) \in L_\infty$ (assumption 1) $\Rightarrow \Omega(t) \in L_\infty$.

The **parameter estimation error** $\hat{\theta}(t) = \theta - \hat{\theta}(t)$:

$$
\dot{\theta} = -\dot{\theta} = \gamma \Omega^T C^T[y - C\bar{x}] = \gamma \Omega^T C^T e = \gamma \Omega^T C^T C[\delta - \Omega \hat{\theta}].
$$

**Intuition:** $\lim_{t \to \infty} \delta(t) = 0$ $\Rightarrow$ $\dot{\theta} = -\gamma h(t)h(t)^T \bar{\theta}$, $h(t) = \Omega^T \theta(t)C^T$ for $t \geq 0$ big enough.

**Assumption 2.** $h(t)$ is PE: $\exists \rho > 0$, $\delta > 0$: $\int_0^t h(\tau)h(\tau)^T d\tau \geq \rho t I_{nq}$, $\forall t \geq \delta$.

**Assumption 2 $\Rightarrow \hat{\theta}(t) \in L_\infty$, $\lim_{t \to \infty} \hat{\theta}(t) = 0$** + properties of $\delta(t) \Rightarrow e(t) \in L_\infty$, $\lim_{t \to \infty} e(t) = 0$.

**Theorem 1.** Under assumptions 1 and 2 all signals in (25)–(28) are bounded and

$$
\lim_{t \to \infty} [\hat{x}(t) - x(t)] = 0, \lim_{t \to \infty} [\hat{\theta}(t) - \theta] = 0.
$$
Example 2

Oscillating pendulum:

\[ \ddot{y} = -\omega^2 \sin(y) - \rho \dot{y} + b \cos(y) f(t) + d(t), \]  \quad (13)

\( y = \varphi \in [-\pi, \pi] \) is the measured angle, \( \dot{y} \in \mathbb{R} \) and \( \ddot{y} \in \mathbb{R} \) are the angle velocity and acceleration; \( \rho > 0 \) is a known friction coefficient, \( \omega > 0 \) is an unknown natural frequency, \( b > 0 \) is an unknown control gain.

Presentation in the form (25) for \( x_1 = y, x_2 = \dot{y}, u = f \) and \( d(t) = 0 \):

\[
\begin{align*}
\dot{x}_1 &= x_2, \quad y = x_1, \\
\dot{x}_2 &= -\rho x_2 - \omega^2 \sin(x_1) + b \cos(x_1)u(t).
\end{align*}
\]

\[ \Rightarrow A = \begin{bmatrix} 0 & 1 \\ 0 & -\rho \end{bmatrix}, \quad C = \begin{bmatrix} 1 \\ 0 \end{bmatrix}^T, \quad B(y) = 0, \quad \varphi(y) = 0, \]

\[ G(y,u) = \begin{bmatrix} 0 & 0 \\ -\sin(y) & \cos(y)u \end{bmatrix}, \quad \Theta = \begin{bmatrix} \omega^2 \\ b \end{bmatrix}. \]

Both assumptions are satisfied for this example.
\[ \omega = 1, \ \rho = 0.1, \ b = 0.5, \ f(t) = \sin(3t), \ L = [2, 1]^T, \ \gamma = 1000. \]

\[ d(t) = 0 \]

\[ d(t) = 0.5 \sin(10t) \]

\[ d(t) = 0.5 \sin(6t) \]
OUTLINE

1. Introduction
   a. Main properties
   b. Running example

2. Adaptive parameter estimation
   a. Parameterized system model
   b. Linear parametric model
   c. Normalized gradient algorithm
   d. Normalized least-squares algorithm
   e. Discrete-time version of adaptive algorithms

3. Identification and robustness
   a. Parametric convergence and persistency of excitation
   b. Robustness of adaptive algorithms

4. Indirect adaptive control
   a. Model reference control
   b. Pole placement control

5. Adaptive observers