Lecture on

New approaches to nonlinear control of electric power systems:

Lyapunov methods

Gerasimos Rigatos

Unit of Industrial Automation
Industrial Systems Institute
26504, Rion Patras, Greece

email: grigat@ieee.org

1. Outline

- The functioning of **distributed nonlinear dynamical systems** in real conditions is characterized by **model uncertainty**, **parametric changes** and **external perturbations**.

- Control schemes must perform **simultaneously identification** and **stabilization** of such uncertain dynamics.

- This is a **dual optimization problem** since **modelling errors** and **deviation of the system's state vector elements from the associated setpoints** have to be **minimized** in real-time.

- To achieve these objectives an initial **transformation** (diffeomorphism) of the system's dynamic model to an **equivalent linearized form**, is proposed.

- The **transformed control inputs** consist of **unknown nonlinear functions** which are **identified** with the use of nonlinear regressors.

- **Learning in such networks** is performed through **gradient algorithms** in which the **adaptation rate** (step for the search of an optimum) is defined by conditions for the **minimization of an aggregate energy function** (Lyapunov function).

1. Outline

- In each iteration of the control algorithm, the **estimates of the nonlinear functions** that constitute the system's dynamics are **fed into a state feedback controller**.

- It has been proven that this control approach assures the **minimization of the aforementioned energy function** and thus the nonlinear system becomes **a globally asymptotically stable one**.

- The proposed method can be applied to **all distributed dynamical systems** which satisfy the **differential flatness property**.

- This is the **widest class of nonlinear dynamical systems** to which one can apply **optimization and control with gradient methods**, while assuring the convergence of the optimization procedure and the stability of the control loop.

- The efficiency of the proposed **Lyapunov theory-based control approac**h has been confirmed in several complex nonlinear dynamical systems

- In particular, the method has been applied to the problem of synchronization and stabilization of **distributed power generators**

2. Differential flatness of MIMO nonlinear systems

- **Differential flatness theory** has been developed as a **global linearization control method** by M. Fliess (Ecole Polytechnique, France) and co-researchers (Lévine, Rouchon, Mounier, Rudolph, Petit, Martin, Zhu, Sira-Ramirez et. al)

- A dynamical system can be written in the ODE form $S_i(w, \dot{w}, \ddot{w}, \ldots, w^{(i)}),\ i=1,2,\ldots,q$

  where $w^{(i)}$ stands for the i-th derivative of either a state vector element or of a control input

- The system is said to be **differentially flat** with respect to the **flat output**

$$y_i = \phi(w, \dot{w}, \ddot{w}, \ldots, w^{(a)}),\ i=1,\ldots,m \quad \text{where} \quad y=(y_1, y_2, \ldots, y_m)$$

if the following two conditions are satisfied

(i) There does not exist any differential relation of the form

$$R(y, \dot{y}, \ddot{y}, \ldots, y^{(\beta)}) = 0$$

which means that **the flat output and its derivatives are linearly independent**

(ii) All system variables are **functions of the flat output and its derivatives**

$$w^{(i)} = \psi(y, \dot{y}, \ddot{y}, \ldots, y^{(\gamma_i)})$$

2. Differential flatness of MIMO nonlinear systems

The proposed Lyapunov theory-based control method is based on the **transformation** of the nonlinear system's model into the **linear canonical form**, and this transformation is succeeded by exploiting the system's differential flatness properties

- **All single input nonlinear systems** are differentially flat and can be transformed into the linear canonical form

One has to define also which are the **MIMO nonlinear systems** which are differentially flat.

- Differential flatness holds for **MIMO nonlinear systems** that admit **static feedback linearization.**and which can be transformed into the linear canonical form through a change of variables (diffeomorphism) and feedback of the state vector.

- Differential flatness holds for **MIMO nonlinear models** that admit **dynamic feedback linearization**, This **is the case of specific underactuated robotic models.** In the latter case the state vector of the system is extended by considering as additional flat outputs some of the control inputs and their derivatives

- Finally, a more rare case is the so-called **Liouvillian systems**. These are systems for which differential flatness properties hold for part of their state vector (constituting a flat subsystem) while the non-flat state variables can be obtained by integration of the elements of the flat subsystem.

3. State-space modelling of MIMO nonlinear systems

3.1. Transformation of MIMO nonlinear systems into the Brunovsky form

The **initial MIMO nonlinear system** is taken to be in the generic form:

$$\dot{x} = f(x,u)$$

It is assumed now that **after defining the flat outputs of the initial MIMO nonlinear system**, and after expressing the system state variables and control inputs as functions of the flat output and of the associated derivatives, the system can be **transformed in the Brunovsky canonical form**

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= x_3 \\
&\cdots \\
\dot{x}_{r_1-1} &= x_{r_1} \\
\dot{x}_{r_1} &= f_1(x) + \sum_{j=1}^{p} g_{1_j}(x) u_j + d_1 \\
\\
\dot{x}_{r_1+1} &= x_{r_1+2} \\
\dot{x}_{r_1+2} &= x_{r_1+3} \\
&\cdots \\
\dot{x}_{p-1} &= x_p \\
\dot{x}_p &= f_p(x) + \sum_{j=1}^{p} g_{p_j}(x) u_j + d_p
\end{aligned}$$

$$\begin{aligned}
y_1 &= x_1 \\
y_2 &= x_{r_1-1} \\
&\cdots \\
y_p &= x_{n-r_p+1}
\end{aligned}$$

$x = [x_1, \cdots, x_n]^T$: is the state vector

$u = [u_1, \cdots, u_p]^T$: is the inputs vector

$y = [y_1, \cdots, y_p]^T$: is the outputs vector

3. State-space modelling of MIMO nonlinear systems

3.1. Transformation of MIMO nonlinear systems into the Brunovsky form

Next **the following vectors and matrices** can be defined

$$f(x)=[f_1(x), \quad \dots, \quad f_n(x)]^T$$

$$g(x)=[g_1(x), \quad \dots, \quad g_n(x)]^T$$

with $g_i(x)=[g_{1i}(x), \quad \dots, \quad g_{pi}(x)]^T$

$$A=diag[A_1,\dots,A_p], \; B=diag[B_1,\dots,B_p]$$

$$C^T=diag[C_1,\dots,C_p], \; d=[d_1,\dots,d_p]^T$$

where matrix A has the **MIMO canonical form**, i.e. with elements

$$A_i=\begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}_{r_i\times r_i}$$

$$B_i^T=[0 \quad 0 \quad \dots \quad 0 \quad 1]_{1\times r_i} \qquad C_i=[1 \quad 0 \quad \dots \quad 0 \quad 0]_{1\times r_i}$$

Thus, the initial nonlinear system can be written in the **state-space form**

$$\dot{x}=Ax+B[f(x)+g(x)u+\tilde{d}]$$

$$y=Cx$$

or equivalently in the state space form

$$\dot{x}=Ax+Bv+B\tilde{d}$$

$$y=Cx$$

where $v=f(x)+g(x)u$

For the generic case of the **MIMO nonlinear system** it is assumed that the functions $f(x)$ and $g(x)$ are unknown and have to be approximated by **nonlinear regressors** (e.g. neuro-fuzzy networks)

3. State-space modelling of MIMO nonlinear systems

3.1. Transformation of MIMO nonlinear systems into the Brunovsky form

Thus, the nonlinear system can be written in **state-space form**

$$\dot{x} = Ax + B[f(x) + g(x)u + \tilde{d}]$$
$$y = C^T x$$

which equivalently can be written as

$$\dot{x} = Ax + Bv + B\tilde{d}$$
$$y = C^T x$$

where

$$v = f(x) + g(x)u.$$

The **reference setpoints** for the system's outputs $y_1, \cdots, y_p$ are denoted as $y_{1m}, \cdots, y_{pm}$ and the associated tracking errors are defined as

$$e_1 = y_1 - y_{1m}$$
$$e_2 = y_2 - y_{2m}$$
$$\cdots$$
$$e_p = y_p - y_{pm}$$

The **error vector of the outputs** of the transformed MIMO system is denoted as

$$E_1 = [e_1, \cdots, e_p]^T$$
$$y_m = [y_{1m}, \cdots, y_{pm}]^T$$
$$\cdots$$
$$y_m^{(n)} = [y_{1m}^{(n)}, \cdots, y_{pm}^{(n)}]^T$$

3. State-space modelling of MIMO nonlinear systems

3.2. Control law under measurable state vector

The **control signal** $v = f(x) + g(x)u$ **of the MIMO nonlinear system** contains the **unknown nonlinear functions** *f(x)* and *g(x)* which can be approximated by

$$\hat{f}(x|\theta_f) = \Phi_f(x)\theta_f, \quad \hat{g}(x|\theta_g) = \Phi_g(x)\theta_g$$

where $\Phi_f(x) = \left(\xi_f^1(x), \xi_f^2(x), \cdots \xi_f^n(x)\right)^T,$

$$\xi_f^i(x) = \left(\phi_f^{i,1}(x), \phi_f^{i,2}(x), \cdots, \phi_f^{i,N}(x)\right)$$

thus giving

$$\Phi_f(x) = \begin{pmatrix} \phi_f^{1,1}(x) & \phi_f^{1,2}(x) & \cdots & \phi_f^{1,N}(x) \\ \phi_f^{2,1}(x) & \phi_f^{2,2}(x) & \cdots & \phi_f^{2,N}(x) \\ \cdots & \cdots & \cdots & \cdots \\ \phi_f^{n,1}(x) & \phi_f^{n,2}(x) & \cdots & \phi_f^{n,N}(x) \end{pmatrix}$$

while the weights vector is defined as $\theta_f^T = \left(\theta_f^1, \theta_f^2, \cdots \theta_f^N\right).$

3. State-space modelling of MIMO nonlinear systems

3.2. Control law under measurable state vector

Similarly, it holds $\Phi_g(x) = (\xi_g^1(x), \xi_g^2(x), \cdots \xi_g^N(x))^T$.

$$\xi_g^i(x) = (\phi_g^{i,1}(x), \phi_g^{i,2}(x), \cdots, \phi_g^{i,N}(x)).$$

thus giving

$$\Phi_g(x) = \begin{pmatrix} \phi_g^{1,1}(x) & \phi_g^{1,2}(x) & \cdots & \phi_g^{1,N}(x) \\ \phi_g^{2,1}(x) & \phi_g^{2,2}(x) & \cdots & \phi_g^{2,N}(x) \\ \cdots & \cdots & \cdots & \cdots \\ \phi_g^{n,1}(x) & \phi_g^{n,2}(x) & \cdots & \phi_g^{n,N}(x) \end{pmatrix}$$

while the weights vector is defined as $\theta_g = (\theta_g^1, \theta_g^2, \cdots, \theta_g^p)^T$.

However, here each row of θ_g is vector thus giving

$$\theta_g = \begin{pmatrix} \theta_{g1}^1 & \theta_{g1}^2 & \cdots & \theta_{g1}^p \\ \theta_{g2}^1 & \theta_{g2}^2 & \cdots & \theta_{g2}^p \\ \cdots & \cdots & \cdots & \cdots \\ \theta_{gN}^1 & \theta_{gN}^2 & \cdots & \theta_{gN}^p \end{pmatrix}$$

If the state variables of the system are available for measurement then a **state-feedback control law can be formulated** as

$$u = \hat{g}^{-1}(x|\theta_g)[-\hat{f}(x|\theta_f) + y_m^{(r)} + K_c^T e + u_c]$$

3. State-space modelling of MIMO nonlinear systems

3.2. Control law under non-measurable state vector

The control of the system $\dot{x} = f(x,u)$ becomes more complicated **when the state vector x is not directly measurable** and has to be reconstructed through a state observer. The following definitions are used

$e = x - x_m$: is the error of the state vector

$\hat{e} = \hat{x} - x_m$ is the error of the estimated state vector

$\tilde{e} = e - \hat{e} = (x - x_m) - (\hat{x} - x_m)$ is the observation error

When an **observer is used to reconstruct the state vector**, the control law

$$u = \hat{g}^{-1}(\hat{x}|\theta_g)[-\hat{f}(\hat{x}|\theta_f) + y_m^{(r)} - K^T\hat{e} + u_c]$$

By applying the previous feedback control law one obtains the closed-loop dynamics

$$y^{(n)} = f(x) + g(x)\hat{g}^{-1}(\hat{x})[-\hat{f}(\hat{x}) + y_m^{(n)} - K^T\hat{e} + u_c] + d \Rightarrow$$
$$y^{(n)} = f(x) + [g(x) - \hat{g}(\hat{x}) + \hat{g}(\hat{x})]\hat{g}^{-1}(\hat{x})[-\hat{f}(\hat{x}) + y_m^{(n)} - K^T\hat{e} + u_c] + d \Rightarrow$$
$$y^{(n)} = [f(x) - \hat{f}(\hat{x})] + [g(x) - \hat{g}(\hat{x})]u + y_m^{(n)} - K^T\hat{e} + u_c + d$$

It holds $e = x - x_m \Rightarrow y^{(n)} = e^{(n)} + y_m^{(n)}$

and by substituting $y^{(n)}$ in the **previous feedback control loop dynamics** gives

3. State-space modelling of MIMO nonlinear systems

3.2. Control law under non-measurable state vector

the **tracking error dynamics**

$$e^{(r)} + y_m^{(r)} = y_m^{(r)} - K^T\hat{e} + u_c + [f(x) - \hat{f}(\hat{x})] + [g(x) - \hat{g}(\hat{x})]u + d$$

or equivalently

$$\dot{e} = Ae - BK^T\hat{e} + Bu_c + B\{[f(x) - \hat{f}(\hat{x})] + [g(x) - \hat{g}(\hat{x})]u + d\}$$ (A)

$$e_1 = C^T e$$

where $e = [e^1, e^2, \cdots, e^p]^T$ with $e^i = [e_i, \dot{e}_i, \ddot{e}_i, \cdots, e_i^{r_i-1}]^T, i = 1, 2, \cdots, p$

and equivalently $\hat{e} = [\hat{e}^1, \hat{e}^2, \cdots, \hat{e}^p]^T$ with $\hat{e}^i = [\hat{e}_i, \dot{\hat{e}}_i, \ddot{\hat{e}}_i, \cdots, \hat{e}_i^{r_i-1}]^T, i = 1, 2, \cdots, p.$

A **state observer** is designed as:

$$\dot{\hat{e}} = A\hat{e} - BK^T\hat{e} + K_o[e_1 - C^T\hat{e}]$$ (B)

$$\hat{e}_1 = C^T\hat{e}$$

4. An application example of optimization-based control

4.1. Dynamics of the tracking error

Without loss of generality consider a two-input MIMO system:

By **applying differential flatness theory, and in the presence of disturbances**, the dynamic model of the system comes to the form

$$\ddot{x}_1 = f_1(x,t) + g_1(x,t)u + d_1$$
$$\ddot{x}_3 = f_2(x,t) + g_2(x,t)u + d_2 \qquad (C)$$

The following **control input** is defined:

$$u = \begin{pmatrix} \hat{g}_1(x,t) \\ \hat{g}_2(x,t) \end{pmatrix}^{-1} \left\{ \begin{pmatrix} \ddot{x}_1^d \\ \ddot{x}_3^d \end{pmatrix} - \begin{pmatrix} \hat{f}_1(x,t) \\ \hat{f}_2(x,t) \end{pmatrix} - \begin{pmatrix} K_1^T \\ K_2^T \end{pmatrix} e + \begin{pmatrix} u_{c_1} \\ u_{c_2} \end{pmatrix} \right\} \qquad (D)$$

where: $[u_{c_1} \; u_{c_2}]^T$ is a **robust control term** that is used for the compensation of the model's uncertainties as well as of the external disturbances

and: $K_i^T = [k_1^i, k_2^i, \cdots, k_{n-1}^i, k_n^i]$ is the feedback gain

Substituting the control input (D) into the system (C) one obtains

$$\begin{pmatrix} \ddot{x}_1 \\ \ddot{x}_3 \end{pmatrix} = \begin{pmatrix} f_1(x,t) \\ f_2(x,t) \end{pmatrix} + \begin{pmatrix} g_1(x,t) \\ g_2(x,t) \end{pmatrix} \begin{pmatrix} \hat{g}_1(x,t) \\ \hat{g}_2(x,t) \end{pmatrix}^{-1} \cdot \left\{ \begin{pmatrix} \ddot{x}_1^d \\ \ddot{x}_3^d \end{pmatrix} - \begin{pmatrix} \hat{f}_1(x,t) \\ \hat{f}_2(x,t) \end{pmatrix} - \begin{pmatrix} K_1^T \\ K_2^T \end{pmatrix} e + \begin{pmatrix} u_{c_1} \\ u_{c_2} \end{pmatrix} \right\} + \begin{pmatrix} d_1 \\ d_2 \end{pmatrix}$$

4. An application example of optimization-based control

4.1. Dynamics of the tracking error

Moreover, using again Eq. (D) one obtains the **tracking error dynamics**

$$\begin{pmatrix} \ddot{e}_1 \\ \ddot{e}_3 \end{pmatrix} = \begin{pmatrix} f_1(x,t) - \hat{f}_1(x,t) \\ f_2(x,t) - \hat{f}_2(x,t) \end{pmatrix} + \begin{pmatrix} g_1(x,t) - \hat{g}_1(x,t) \\ g_2(x,t) - \hat{g}_2(x,t) \end{pmatrix} u - \begin{pmatrix} K_1^T \\ K_2^T \end{pmatrix} e + \begin{pmatrix} u_{c_1} \\ u_{c_2} \end{pmatrix} + \begin{pmatrix} d_1 \\ d_2 \end{pmatrix}$$

The **approximation error** is defined as:

$$w = \begin{pmatrix} f_1(x,t) - \hat{f}_1(x,t) \\ f_2(x,t) - \hat{f}_2(x,t) \end{pmatrix} + \begin{pmatrix} g_1(x,t) - \hat{g}_1(x,t) \\ g_2(x,t) - \hat{g}_2(x,t) \end{pmatrix} u$$

Using matrices A,B;K, and considering that **the estimated state vector is used in the control loop** the following description of the tracking error dynamics is obtained:

$$\dot{e} = Ae - BK^T\hat{e} + Bu_c + B\{ \begin{pmatrix} f_1(x,t) - \hat{f}_1(\hat{x},t) \\ f_2(x,t) - \hat{f}_2(\hat{x},t) \end{pmatrix} + \begin{pmatrix} g_1(x,t) - \hat{g}_1(\hat{x},t) \\ g_2(x,t) - \hat{g}_2(\hat{x},t) \end{pmatrix} u + \tilde{d} \}$$

When the **estimated state vector** is used in the loop the **approximation error is written as**

$$w = \begin{pmatrix} f_1(x,t) - \hat{f}_1(\hat{x},t) \\ f_2(x,t) - \hat{f}_2(\hat{x},t) \end{pmatrix} + \begin{pmatrix} g_1(x,t) - \hat{g}_1(\hat{x},t) \\ g_2(x,t) - \hat{g}_2(\hat{x},t) \end{pmatrix} u$$

while the **tracking error dynamics becomes**

$$\dot{e} = Ae - BK^T\hat{e} + Bu_c + Bw + B\tilde{d}$$

4. An application example of optimization-based control

4.2. Dynamics of the observation error

The **observation error** is defined as: $\bar{e} = e - \hat{e} = x - \hat{x}.$

By subtracting Eq. (B) from Eq. (A) one obtains:

$$\dot{e} - \dot{\hat{e}} = A(e-\hat{e}) + Bu_c + B\{[f(x,t) - \hat{f}(\hat{x},t)] + [g(x,t) - \hat{g}(\hat{x},t)]u + \bar{d}\} - K_o C^T(e-\hat{e})$$

$$e_1 - \hat{e}_1 = C^T(e-\hat{e})$$

or equivalently:

$$\dot{\bar{e}} = A\bar{e} + Bu_c + B\{[f(x,t) - \hat{f}(\hat{x},t)] + [g(x,t) - \hat{g}(\hat{x},t)]u + \bar{d}\} - K_o C^T\bar{e}$$

$$\bar{e}_1 = C^T\bar{e}$$

which can be also written as:

$$\dot{\bar{e}} = (A - K_o C^T)\bar{e} + Bu_c + Bw + \bar{d}\}$$

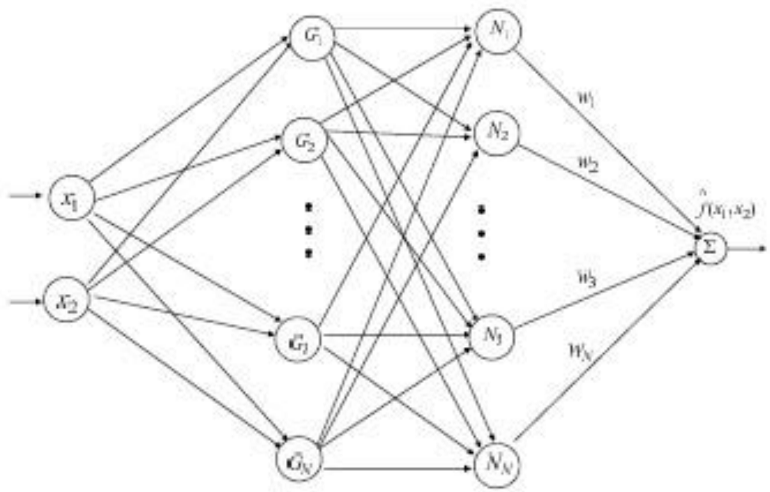
$$\bar{e}_1 = C^T\bar{e}$$

4. An application example of optimization-based control

4.3. Approximation of the unknown system dynamics

Next, **the first of the approximators** of the unknown system dynamics is defined

$$\hat{f}(\hat{x}) = \begin{pmatrix} \hat{f}_1(\hat{x}|\theta_f) & \hat{x}\in R^{4\times 1} & \hat{f}_1(\hat{x}|\theta_f) & \in & R^{1\times 1} \\ \hat{f}_2(\hat{x}|\theta_f) & \hat{x}\in R^{4\times 1} & \hat{f}_2(\hat{x}|\theta_f) & \in & R^{1\times 1} \end{pmatrix}$$



containing kernel functions $\phi_f^{l,j}(\hat{x}) = \frac{\prod_{j=1}^{n}\mu_{A_j^l}(\hat{x}_j)}{\sum_{l=1}^{N}\prod_{j=1}^{n}\mu_{A_j^l}(\hat{x}_j)}$

where $\mu_{A_j^l}(\hat{x})$ are fuzzy membership functions

appearing in the antecedent part of the *l-th* fuzzy rule

4. An application example of optimization-based control

4.3. Approximation of the unknown system dynamics

Similarly, **the second of the approximators** of the unknown system dynamics is defined

$$\hat{g}(\hat{x}) = \begin{pmatrix} \hat{g}_1(\hat{x}|\theta_g) & \hat{x}\in R^{4\times 1} & \hat{g}_1(\hat{x}|\theta_g) & \in & R^{1\times 2} \\ \hat{g}_2(\hat{x}|\theta_g) & \hat{x}\in R^{4\times 1} & \hat{g}_2(\hat{x}|\theta_g) & \in & R^{1\times 2} \end{pmatrix}$$

The **values of the weights that result in optimal approximation** are

$$\theta_f^* = arg\ min_{\theta_f \in M_{\theta_f}} [sup_{\hat{x}\in U_{\hat{x}}} (f(x) - \hat{f}(\hat{x}|\theta_f))]$$
$$\theta_g^* = arg\ min_{\theta_g \in M_{\theta_g}} [sup_{\hat{x}\in U_{\hat{x}}} (g(x) - \hat{g}(\hat{x}|\theta_g))]$$

The variation ranges for the weights are given by

$$M_{\theta_f} = \{\theta_f \in R^h : \ ||\theta_f|| \le m_{\theta_f}\}$$
$$M_{\theta_g} = \{\theta_g \in R^h : \ ||\theta_g|| \le m_{\theta_g}\}$$

The **value of the approximation error** that corresponds to the optimal values of the weights vectors is

$$w = \left(f(x,t) - \hat{f}(\hat{x}|\theta_f^*)\right) + \left(g(x,t) - \hat{g}(\hat{x}|\theta_g^*)\right) u$$

4. An application example of optimization-based control

4.3. Approximation of the unknown system dynamics

which is next written as

$$w = \left(f(x,t) - \hat{f}(\hat{x}|\theta_f) + \hat{f}(\hat{x}|\theta_f) - \hat{f}(\hat{x}|\theta_f^*)\right) + \\ + \left(g(x,t) - \hat{g}(\hat{x}|\theta_g) + \hat{g}(\hat{x}|\theta_g) - \hat{g}(\hat{x}|\theta_g^*)\right) u$$

which can be also written in the following form

$$w = (w_a + w_b)$$

with

$$w_a = \{[f(x,t) - \hat{f}(\hat{x}|\theta_f)] + [g(x,t) - \hat{g}(\hat{x}|\theta_g)]\} u$$

and

$$w_b = \{[\hat{f}(\hat{x}|\theta_f) - \hat{f}(\hat{x}|\theta_f^*)] + [\hat{g}(\hat{x},\theta_g) - \hat{g}(\hat{x}|\theta_g^*)]\} u$$

Moreover, the following **weights error vectors are defined**

$$\tilde{\theta}_f = \theta_f - \theta_f^* \\ \tilde{\theta}_g = \theta_g - \theta_g^*$$

and these denote the distance of the **weights vectors** from the values that provide optimal model estimation

It will be shown that these **weights are updated through a gradient method**

5. Convergence proof for the optimization method

The **following Lyapunov (energy) function** is considered:

$$V = \frac{1}{2}\hat{e}^T P_1 \hat{e} + \frac{1}{2}\bar{e}^T P_2 \bar{e} + \frac{1}{2\gamma_1}\bar{\theta}_f^T \bar{\theta}_f + \frac{1}{2\gamma_2} tr[\bar{\theta}_g^T \bar{\theta}_g]$$

The selection of the **Lyapunov function** is based on the following principle of indirect adaptive control

$$\hat{e} : \lim_{t\to\infty} \hat{x}(t) = x_d(t)$$
$$\bar{e} : \lim_{t\to\infty} \hat{x}(t) = x(t).$$

this results into

$$\lim_{t\to\infty} x(t) = x_d(t)$$

By deriving the **Lyapunov function** with respect to time one obtains:

$$\dot{V} = \frac{1}{2}\dot{\hat{e}}^T P_1 \hat{e} + \frac{1}{2}\hat{e}^T P_1 \dot{\hat{e}} + \frac{1}{2}\dot{\bar{e}}^T P_2 \bar{e} + \frac{1}{2}\bar{e}^T P_2 \dot{\bar{e}} + \\ + \frac{1}{\gamma_1}\dot{\bar{\theta}}_f^T \bar{\theta}_f + \frac{1}{\gamma_2} tr[\dot{\bar{\theta}}_g^T \bar{\theta}_g] \Rightarrow$$

$$\dot{V} = \frac{1}{2}\{(A - BK^T)\hat{e} + K_o C^T \bar{e}\}^T P_1 \hat{e} + \frac{1}{2}\hat{e}^T P_1 \{(A - BK^T)\hat{e} + K_o C^T \bar{e}\} + \\ + \frac{1}{2}\{(A - K_o C^T)\bar{e} + B u_c + B\bar{d} + Bw\}^T P_2 \bar{e} + \\ + \frac{1}{2}\bar{e}^T P_2 \{(A - K_o C^T)\bar{e} + B u_c + B\bar{d} + Bw\} + \\ + \frac{1}{\gamma_1}\dot{\bar{\theta}}_f^T \bar{\theta}_f + \frac{1}{\gamma_2} tr[\dot{\bar{\theta}}_g^T \bar{\theta}_g] \Rightarrow$$

5. Convergence proof for the optimization method

The previous equation is rewritten as:

$$\dot{V} = \frac{1}{2}\{\hat{e}^T(A-BK^T)^T + \tilde{e}^T C K_o^T\} P_1 \hat{e} + \frac{1}{2}\hat{e}^T P_1 \{(A-BK^T)\hat{e} + K_o C^T \tilde{e}\} + $$
$$+ \frac{1}{2}\{\tilde{e}^T (A - K_o C^T)^T + u_c^T B^T + w^T B^T + \tilde{d}^T B^T\} P_2 \tilde{e} + $$
$$\frac{1}{2}\tilde{e}^T P_2 \{(A - K_o C^T)\tilde{e} + B u_c + B w + B\tilde{d}\} + \frac{1}{\gamma_1}\dot{\tilde{\theta}}_f^T \tilde{\theta}_f + \frac{1}{\gamma_2} tr[\dot{\tilde{\theta}}_g^T \tilde{\theta}_g] \Rightarrow$$

which finally takes the form:

$$\dot{V} = \frac{1}{2}\hat{e}^T(A-BK^T)^T P_1 \hat{e} + \frac{1}{2}\tilde{e}^T C K_o^T P_1 \hat{e} + $$
$$+ \frac{1}{2}\hat{e}^T P_1 (A - BK^T)\hat{e} + \frac{1}{2}\hat{e}^T P_1 K_o C^T \tilde{e} + $$
$$+ \frac{1}{2}\tilde{e}^T (A - K_o C^T)^T P_2 \tilde{e} + \frac{1}{2}(u_c^T + w^T + \tilde{d}^T) B^T P_2 \tilde{e} + $$
$$+ \frac{1}{2}\tilde{e}^T P_2 (A - K_o C^T)\tilde{e} + \frac{1}{2}\tilde{e}^T P_2 B (u_c + w + \tilde{d}) + $$
$$+ \frac{1}{\gamma_1}\dot{\tilde{\theta}}_f^T \tilde{\theta}_f + \frac{1}{\gamma_2} tr[\dot{\tilde{\theta}}_g^T \tilde{\theta}_g]$$

Assumption 1: For given positive definite matrices Q1 and Q2 there exist positive definite matrices P1 and P2, which are the solution of the following **Riccati equations**

$$(A - BK^T)^T P_1 + P_1 (A - BK^T) + Q_1 = 0$$

$$(A - K_o C^T)^T P_2 + P_2 (A - K_o C^T) -$$
$$- P_2 B (\frac{2}{r} - \frac{1}{\rho^2}) B^T P_2 + Q_2 = 0$$

5. Convergence proof for the optimization method

By substituting the relations described by the previous **Riccati equations** into the derivative of the Lyapunov function one gets:

$$\dot{V} = \frac{1}{2}\hat{e}^T\{(A-BK^T)^T P_1 + P_1(A-BK^T)\}\hat{e} + \tilde{e}^T C K_o^T P_1 \hat{e} + \\ + \frac{1}{2}\tilde{e}^T\{(A-K_oC^T)^T P_2 + P_2(A-K_oC^T)\}\tilde{e} + \\ + \tilde{e}^T P_2 B(u_c + w + \tilde{d}) + \frac{1}{\gamma_1}\dot{\tilde{\theta}}_f^T \tilde{\theta}_f + \frac{1}{\gamma_2} tr[\dot{\tilde{\theta}}_g^T \tilde{\theta}_g]$$

or:

$$\dot{V} = -\frac{1}{2}\hat{e}^T Q_1 \hat{e} + \tilde{e}^T C K_o^T P_1 \hat{e} - \frac{1}{2}\tilde{e}^T\{Q_2 - P_2 B(\frac{2}{r} - \frac{1}{\rho^2})B^T P_2\}\tilde{e} + \\ + \tilde{e}^T P_2 B(u_c + w + \tilde{d}) + \frac{1}{\gamma_1}\dot{\tilde{\theta}}_f^T \tilde{\theta}_f + \frac{1}{\gamma_2} tr[\dot{\tilde{\theta}}_g^T \tilde{\theta}_g]$$

The **supervisory control term** u_c consists of two terms u_a and u_b

The first term u_c is

$$u_a = -\frac{1}{r}\tilde{e}^T P_2 B + \Delta u_a$$

where assuming that the measurable elements of vector $\tilde{e}$ are $\{\tilde{e}_1, \tilde{e}_3, \cdots, \tilde{e}_k\}$

5. Convergence proof for the optimization method

The term Δu_a is such that

$$-\frac{1}{r}\tilde{e}^T P_2 B + \Delta u_a = -\frac{1}{r}\begin{pmatrix} p_{11}\tilde{e}_1 + p_{13}\tilde{e}_3 + \cdots + p_{1k}\tilde{e}_k \\ p_{13}\tilde{e}_1 + p_{33}\tilde{e}_3 + \cdots + p_{3k}\tilde{e}_k \\ \cdots \cdots\cdots \\ p_{1k}\tilde{e}_1 + p_{3k}\tilde{e}_3 + \cdots + p_{kk}\tilde{e}_k \end{pmatrix}$$

u_a is an H_∞ control used for the **compensation of the approximation error** w and the additive disturbance $\tilde{d}$ (the control term u_a has been chosen so as to satisfy the condition

The previous relation finally stands for a product between the measurable state vector elements $\{\tilde{e}_1, \tilde{e}_3, \cdots, \tilde{e}_k\}$ and the elements of matrix P_2 which is obtained from the solution of the previous Riccati equation.

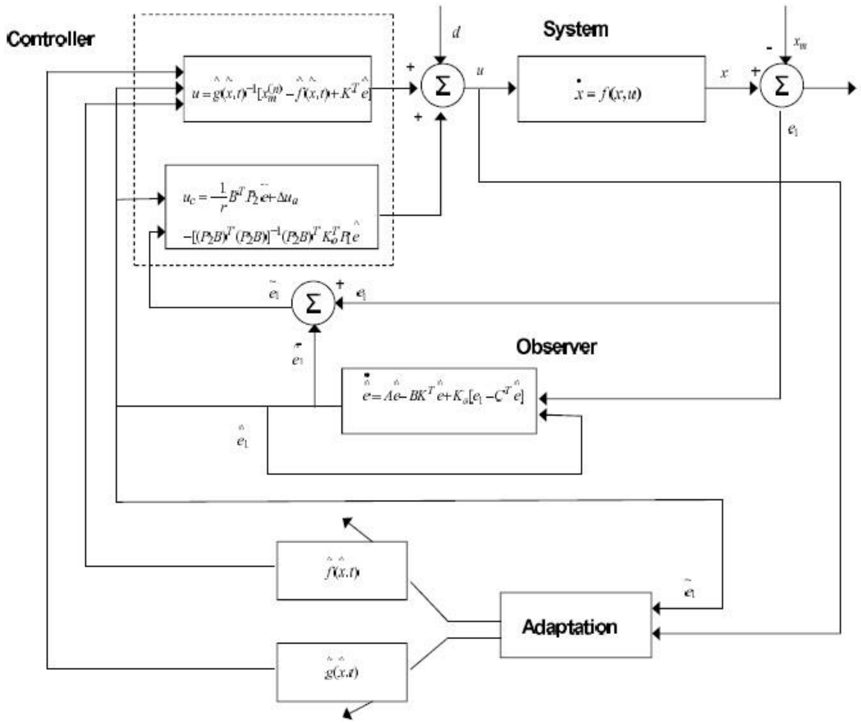
The control term u_b is given by

$$u_b = -[(P_2B)^T(P_2B)]^{-1}(P_2B)^T C K_o^T P_1 \hat{e}$$

u_b is a control used for the **compensation of the observation error** (the control term u_b has been chosen so as to satisfy the condition $\tilde{e}^T P_2 B u_b = -\tilde{e}^T C K_o^T P_1 \hat{e}$.

5. Convergence proof for the optimization method

The **optimization-based control scheme** is depicted in the following diagram



By substituting the supervisory control term in the **derivative of the Lyapunov function** one obtains

$$\dot{V} = -\frac{1}{2}\hat{e}^T Q_1 \hat{e} + \bar{e}^T C K_o^T P_1 \hat{e} - \frac{1}{2}\bar{e}^T Q_2 \bar{e} + \frac{1}{r}\bar{e}^T P_2 B B^T P_2 \bar{e} - \frac{1}{2\rho^2}\bar{e}^T P_2 B B^T P_2 \bar{e} +$$
$$+ \bar{e}^T P_2 B u_a + \bar{e}^T P_2 B u_b + \bar{e}^T P_2 B(w + \bar{d}) + \frac{1}{\gamma_1}\dot{\bar{\theta}}_f^T \bar{\theta}_f + \frac{1}{\gamma_2} tr[\dot{\bar{\theta}}_g^T \bar{\theta}_g]$$

5. Convergence proof for the optimization method

or equivalently

$$\dot{V} = -\frac{1}{2}\hat{e}^T Q_1 \hat{e} - \frac{1}{2}\tilde{e}^T Q_2 \tilde{e} - \frac{1}{2\rho^2}\tilde{e}^T P_2 B B^T P_2 \tilde{e} + \tilde{e}^T P_2 B(w + \tilde{d} + \Delta u_a) + \frac{1}{\gamma_1}\dot{\tilde{\theta}}_f^T \tilde{\theta}_f + \frac{1}{\gamma_2} tr[\dot{\tilde{\theta}}_g^T \tilde{\theta}_g]$$

Besides, about the **adaptation of the weights** of the neurofuzzy approximator it holds

$$\dot{\tilde{\theta}}_f = \dot{\theta}_f - \dot{\theta}_f^* = \dot{\theta}_f \qquad \dot{\tilde{\theta}}_g = \dot{\theta}_g - \dot{\theta}_g^* = \dot{\theta}_g.$$

A **gradient-based update** is applied to the approximator's weights

$$\dot{\theta}_f = -\gamma_1 \Phi(\hat{x})^T B^T P_2 \bar{e}$$
$$\dot{\theta}_g = -\gamma_2 \Phi(\hat{x})^T B^T P_2 \bar{e} u^T$$

Gradient-based optimization

The **gradient update scheme** is defined in a manner that assures that **the first derivative of the Lyapunov function will remain negative**, and thus the Lyapunov function will be monotonously decreasing.

By substituting the above relations in the derivative of the Lyapunov function one obtains

$$\dot{V} = -\frac{1}{2}\hat{e}^T Q_1 \hat{e} - \frac{1}{2}\tilde{e}^T Q_2 \tilde{e} - \frac{1}{2\rho^2}\tilde{e}^T P_2 B B^T P_2 \tilde{e} + B^T P_2 \tilde{e}(w + \tilde{d} + \Delta u_a) + \frac{1}{\gamma_1}(-\gamma_1)\tilde{e}^T P_2 B \Phi(\hat{x})(\theta_f - \theta_f^*) + \frac{1}{\gamma_2}(-\gamma_2) tr[u\tilde{e}^T P_2 B(\hat{g}(\hat{x}|\theta_g) - \hat{g}(\hat{x}|\theta_g^*))]$$

5. Convergence proof for the optimization method

To continue with the **convergence proof for the proposed optimization method** it is taken into account that

$$u \in R^{2\times 1} \text{ and } \tilde{e}^T PB(\hat{g}(x|\theta_g) - \hat{g}(x|\theta_g^*)) \in R^{1\times 2}$$

one gets

$$\dot{V} = -\frac{1}{2}\hat{e}^T Q_1 \hat{e} - \frac{1}{2}\tilde{e}^T Q_2 \tilde{e} - \frac{1}{2\rho^2}\tilde{e}^T P_2 B B^T P_2 \tilde{e} + B^T P_2 \tilde{e}(w + \tilde{d} + \Delta u_a) + \frac{1}{\gamma_1}(-\gamma_1)\tilde{e}^T P_2 B \Phi(\hat{x})(\theta_f - \theta_f^*) + \frac{1}{\gamma_2}(-\gamma_2) tr[\tilde{e}^T P_2 B(\hat{g}(\hat{x}|\theta_g) - \hat{g}(\hat{x}|\theta_g^*))u]$$

Since

$$\tilde{e}^T P_2 B(\hat{g}(\hat{x}|\theta_g) - \hat{g}(\hat{x}|\theta_g^*))u \in R^{1\times 1}$$

it holds

$$tr(\tilde{e}^T P_2 B(\hat{g}(x|\theta_g) - \hat{g}(x|\theta_g^*)u) = \\ = \tilde{e}^T P_2 B(\hat{g}(x|\theta_g) - \hat{g}(x|\theta_g^*))u$$

Therefore, one finally obtains

$$\dot{V} = -\frac{1}{2}\hat{e}^T Q_1 \hat{e} - \frac{1}{2}\tilde{e}^T Q_2 \tilde{e} - \frac{1}{2\rho^2}\tilde{e}^T P_2 B B^T P_2 \tilde{e} + B^T P_2 \tilde{e}(w + \tilde{d} + \Delta u_a) + \frac{1}{\gamma_1}(-\gamma_1)\tilde{e}^T P_2 B \Phi(\hat{x})(\theta_f - \theta_f^*) + \frac{1}{\gamma_2}(-\gamma_2)\tilde{e}^T P_2 B(\hat{g}(\hat{x}|\theta_g) - \hat{g}(\hat{x}|\theta_g^*))u$$

Next, the following **approximation error** is defined

$$w_a = [\hat{f}(\hat{x}|\theta_f^*) - \hat{f}(\hat{x}|\theta_f)] + [\hat{g}(\hat{x}|\theta_g^*) - \hat{g}(\hat{x}|\theta_g)]u$$

5. Convergence proof for the optimization method

Thus, one obtains

$$\dot{V} = -\frac{1}{2}\hat{e}^T Q_1 \hat{e} - \frac{1}{2}\tilde{e}^T Q_2 \tilde{e} - \frac{1}{2\rho^2}\tilde{e}^T P_2 B B^T P_2 \tilde{e} + \\ + B^T P_2 \tilde{e}(w + \tilde{d} + \Delta u_a) + \tilde{e}^T P_2 B w_\alpha$$

Denoting the **aggregate approximation error** and disturbances vector as

$$w_1 = w + \tilde{d} + w_\alpha + \Delta u_a$$

the derivative of the Lyapunov function becomes

$$\dot{V} = -\frac{1}{2}\hat{e}^T Q_1 \hat{e} - \frac{1}{2}\tilde{e}^T Q_2 \tilde{e} - \frac{1}{2\rho^2}\tilde{e}^T P_2 B B^T P_2 \tilde{e} + \tilde{e}^T P_2 B w_1$$

which in turn is written as

$$\dot{V} = -\frac{1}{2}\hat{e}^T Q_1 \hat{e} - \frac{1}{2}\tilde{e}^T Q_2 \tilde{e} - \frac{1}{2\rho^2}\tilde{e}^T P_2 B B^T P_2 \tilde{e} + \\ + \frac{1}{2}\tilde{e}^T P B w_1 + \frac{1}{2} w_1^T B^T P_2 \tilde{e}$$

Lemma: The following inequality holds

$$\frac{1}{2}\tilde{e}^T P_2 B w_1 + \frac{1}{2} w_1^T B^T P_2 \tilde{e} - \frac{1}{2\rho^2}\tilde{e}^T P_2 B B^T P_2 \tilde{e} \\ \leq \frac{1}{2}\rho^2 w_1^T w_1$$

5. Convergence proof for the optimization method

Proof:

The binomial $(\rho a - \frac{1}{\rho} b)^2 \geq 0$ is considered. Expanding the left part of the above inequality one gets

$$\rho^2 a^2 + \frac{1}{\rho^2} b^2 - 2ab \geq 0 \Rightarrow$$
$$\frac{1}{2}\rho^2 a^2 + \frac{1}{2\rho^2} b^2 - ab \geq 0 \Rightarrow$$
$$ab - \frac{1}{2\rho^2} b^2 \leq \frac{1}{2}\rho^2 a^2 \Rightarrow$$
$$\frac{1}{2}ab + \frac{1}{2}ab - \frac{1}{2\rho^2} b^2 \leq \frac{1}{2}\rho^2 a^2$$

By substituting $a = w_1$ and $b = \bar{e}^T P_2 B$ one gets

$$\frac{1}{2} w_1^T B^T P_2 \bar{e} + \frac{1}{2} \bar{e}^T P_2 B w_1 - \frac{1}{2\rho^2} \bar{e}^T P_2 B B^T P_2 \bar{e} \leq \frac{1}{2}\rho^2 w_1^T w_1$$

Moreover, by substituting the above inequality into the **derivative of the Lyapunov function** one gets

$$\dot{V} \leq -\frac{1}{2}\hat{e}^T Q_1 \hat{e} - \frac{1}{2}\bar{e}^T Q_2 \bar{e} + \frac{1}{2}\rho^2 w_1^T w_1$$

which is also written as $\dot{V} \leq -\frac{1}{2} E^T Q E + \frac{1}{2}\rho^2 w_1^T w_1$

with

$$E = \begin{pmatrix} \hat{e} \\ \bar{e} \end{pmatrix}, \quad Q = \begin{pmatrix} Q_1 & 0 \\ 0 & Q_2 \end{pmatrix} = diag[Q_1, Q_2]$$

5. Convergence proof for the optimization method

Hence, the H_∞ performance criterion is derived. **For sufficiently small *ρ* the inequality will be true** and the H_∞ tracking criterion will be satisfied. In that case, the integration of $\dot{V}$ from 0 to T gives

$$\int_0^T \dot{V}(t)\,dt \leq -\frac{1}{2}\int_0^T ||E||^2\,dt + \frac{1}{2}\rho^2 \int_0^T ||w_1||^2\,dt \Rightarrow$$
$$2V(T) - 2V(0) \leq -\int_0^T ||E||_Q^2\,dt + \rho^2 \int_0^T ||w_1||^2\,dt \Rightarrow$$
$$2V(T) + \int_0^T ||E||_Q^2\,dt \leq 2V(0) + \rho^2 \int_0^T ||w_1||^2\,dt$$

It is assumed that there exists a positive constant $M_w > 0$ such that

$$\int_0^\infty ||w_1||^2\,dt \leq M_w$$

Therefore for the integral $\int_0^T ||E||_Q^2\,dt$ one gets

$$\int_0^\infty ||E||_Q^2\,dt \leq 2V(0) + \rho^2 M_w$$

Thus, the integral $\int_0^\infty ||E||_Q^2\,dt$ is bounded and **according to Barbalat's Lemma**

$$\lim_{t\to\infty} e(t) = 0$$

and thus **global asymptotic stability** is also shown for the control loop.

6. Case studies on robotic and electric power systems

6.2 The model of distributed power generators

The **dynamic model of the distributed power generation units** is assumed to be that of synchronous generators. The modelling approach is also applicable to PMSGs (permanent magnet synchronous generators) which are a special case of synchronous electric machines.

$$\dot{\delta} = \omega$$
$$\dot{\omega} = -\frac{D}{2J}(\omega - \omega_0) + \frac{\omega_0}{2J}(P_m - P_e) \qquad (1)$$

δ : turn angle of the rotor

ω : turn speed of the rotor

ω_0 : synchronous speed

J : moment of inertia of the rotor

P_e : active electrical power of the machine

P_m : mechanical power of the machine

D : damping coefficient

T_e : electromagnetic torque

The **generator's electrical dynamics** is:

$$\dot{E}_q' = \frac{1}{T_{d_o}}(E_f - E_q) \qquad (2)$$

E_q' is the quadrature-axis transient voltage (a variable related to the magnetic flux)

E_q is quadrature axis voltage of the generator

T_{d_o} is the direct axis open-circuit transient time constant

E_f is the equivalent voltage in the excitation coil

6. Case studies on robotic and electric power systems

6.2 The model of distributed power generators

The **synchronous generator's model** is complemented by a set of algebraic equations:

$$E_q = \frac{x_{d\Sigma}}{x'_{d\Sigma}} E'_q - (x_d - x'_d)\frac{V_s}{x'_{d\Sigma}} cos(\Delta\delta)$$
$$I_q = \frac{V_s}{x'_{d\Sigma}} sin(\Delta\delta)$$
$$I_d = \frac{E'_q}{x'_{d\Sigma}} - \frac{V_s}{x'_{d\Sigma}} cos(\Delta\delta)$$
$$P_e = \frac{V_s E'_q}{x'_{d\Sigma}} sin(\Delta\delta)$$
$$Q_e = \frac{V_s E'_q}{x'_{d\Sigma}} cos(\Delta\delta) - \frac{V_s^2}{x_{d\Sigma}}$$
$$V_t = \sqrt{(E'_q - X'_d I_d)^2 + (X'_d I_q)^2}$$

3

where: $x_{d\Sigma} = x_d + x_T + x_L$ $\quad x'_{d\Sigma} = x'_d + x_T + x_L$

x_d : direct-axis synchronous reactance

x_T : reactance of the transformer

x'_d : direct-axis transient reactance

x_L : transmission line reactance

I_d and I_q : direct and quadrature axis currents

V_s : infinite bus voltage

Q_e : reactive power of the generator

V_t : terminal voltage of the generator

6. Case studies on robotic and electric power systems

6.2 The model of distributed power generators

From Eq. (1) and Eq. (2) one obtains the **dynamic model of the synchronous generator**:

$$\dot{\delta} = \omega - \omega_0$$
$$\dot{\omega} = -\frac{D}{2J}(\omega - \omega_0) + \omega_0 \frac{P_m}{2J} - \omega_0 \frac{1}{2J} \frac{V_s E_q'}{x_{d_\Sigma}'} sin(\Delta\delta)$$
$$\dot{E}_q' = -\frac{1}{T_d'} E_q' + \frac{1}{T_{d_o}} \frac{x_d - x_d'}{x_{d_\Sigma}'} V_s cos(\Delta\delta) + \frac{1}{T_{d_o}} E_f$$

Moreover, the generator can be written in a **state-space form**:

$$\dot{x} = f(x) + g(x)u$$

where the state vector is $x = (\Delta\delta \quad \Delta\omega \quad E_q')^T$ and

$$f(x) = \begin{pmatrix} \omega - \omega_0 \\ -\frac{D}{2J}(\omega - \omega_0) + \omega_0 \frac{P_m}{2J} - \omega_0 \frac{1}{2J} \frac{V_s E_q'}{x_{d\Sigma}'} sin(\Delta\delta) \\ -\frac{1}{T_d'} E_q' + \frac{1}{T_{do}} \frac{x_d - x_d'}{x_{d\Sigma}'} V_s cos(\Delta\delta) \end{pmatrix}$$

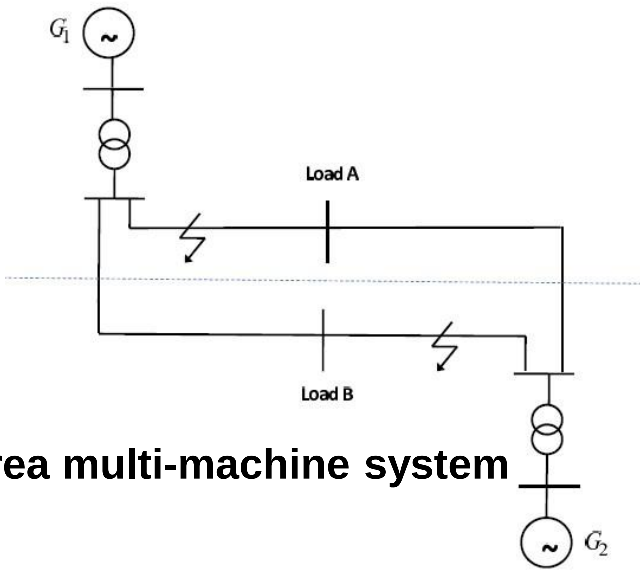
$$g(x) = (0 \quad 0 \quad \frac{1}{T_{do}})^T$$

while the system's output is $y = h(x) = \delta - \delta_0$

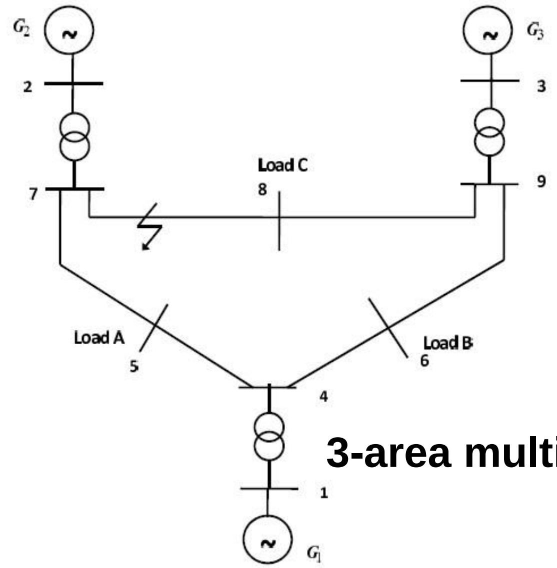
6. Case studies on robotic and electric power systems

6.2 The model of distributed power generators

The **interconnection between distributed power generators** results into a **multi-area multi-machine power system** model



2-area multi-machine system



3-area multi-machine system

The dynamic model of a power system that comprises **n-interconnected power generators** is

$$\dot{\delta}_i = \omega_i - \omega_0$$

$$\dot{\omega}_i = -\frac{D_i}{2J_i}(\omega_i - \omega_0) + \omega_0\frac{P_{m_i}}{2J_i} -$$

$$-\omega_0\frac{1}{2J_i}[G_{ii}E'^{2}_{qi} + E'_{qi}\sum_{j=1,j\neq i}^{n}E'_{qj}G_{ij}sin(\delta_i - \delta_j - \alpha_{ij})]$$

$$\dot{E}'_{q_i} = -\frac{1}{T'_{d_i}}E'_{q_i} + \frac{1}{T_{do_i}}\frac{x_{d_i} - x'_{d_i}}{x'_{d\Sigma_i}}V_{s_i}cos(\Delta\delta_i) + \frac{1}{T_{do_i}}E_{f_i}$$

6. Case studies on robotic and electric power systems

6.2 The model of distributed power generators

The **active power** associated with the i-th power generator is given by:

$$P_{e_i} = G_{ii}E_{qi}^{'^2} + E_{qi}^{'}\sum_{j=1,j\neq i}^{n} E_{qj}^{'} G_{ij} sin(\delta_i - \delta_j - \alpha_{ij})$$

The state vector of the distributed power system is given by $x = [x^1, x^2, \cdots, x^n]^T$

where $x^i = [x_1^i, x_2^i, x_3^i]^T$ with $x_1^i = \Delta\delta_i$ $x_2^i = \Delta\omega_i$ and $x_3^i = E_{qi}^{'}$ $i = 1, 2, \cdots, n$

Next, **differential flatness** is proven for the model of the **stand-alone synchronous generator**.

In state-space form one has:

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -\frac{D}{2J}x_2 + \omega_0\frac{P_m}{2J} - \frac{\omega_0}{2J}\frac{V_s}{x_{d\Sigma}^{'}}x_3 sin(x_1)$$

$$\dot{x}_3 = -\frac{1}{T_d^{'}}x_3 + \frac{1}{T_{do}}\frac{x_d - x_d^{'}}{x_{d\Sigma}^{'}}V_s cos(x_1) + \frac{1}{T_{do}}u$$

The **flat output** is taken to be $y = x_1$

It holds that $x_1 = y$ $x_2 = \dot{y}$ and for $x_1 \neq \pm n\pi$,

$$x_3 = \frac{\omega_0\frac{P_m}{2J} - \ddot{y} - \frac{D}{2J}\dot{y}}{\frac{\omega_0}{2J}\frac{V_s}{x_{d\Sigma}^{'}}sin(y)}, \text{ or } x_3 = f_a(y, \dot{y}, \ddot{y})$$

6. Case studies on robotic and electric power systems

6.2 The model of distributed power generators

while for the **generator's control input** one has

$$u = T_{do}[\dot{x}_3 + \frac{1}{T_d'}x_3 \frac{1}{T_{do}} \frac{x_d - x_d'}{x_{d\Sigma}'} V_s cos(x_1)], \text{ or}$$
$$u = f_b(y, \dot{y}, \ddot{y})$$

Consequently, **all state variables** and the **control input** of the synchronous generator are written as **differential functions** of the flat output and thus the differential flatness of the model is confirmed.

By defining the **new state variables** $y_1 = y,\ y_2 = \dot{y},\ y_3 = \ddot{y}$

the generator's model is transformed into the **canonical (Brunovsky) form**:

$$\begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} v$$

with $v = f_c(y, \dot{y}, \ddot{y}) + g_c(y, \dot{y}, \ddot{y})u$ where

$$f_c(y, \dot{y}, \ddot{y}) = (\frac{D}{2J})^2 \dot{y} - \omega_0 \frac{D}{2J} \frac{P_m}{2J} + \omega_0 \frac{D}{(2J)^2} \frac{V_s}{x_{d\Sigma}'} x_3 sin(\dot{y}) +$$
$$+ \frac{\omega_0}{2J} \frac{V_s}{x_{d\Sigma}'} \frac{1}{T_d'} x_3 sin(y) - \frac{\omega_0}{2J} \frac{V_s}{x_{d\Sigma}'} \frac{1}{T_{do}} \frac{x_d - x_d'}{x_{d\Sigma}'} V_s cos(y) sin(y) -$$
$$- \frac{\omega_0}{2J} \frac{V_s}{x_{d\Sigma}'} x_3 cos(y) \dot{y}$$

and $$g_c(y, \dot{y}, \ddot{y}) = -\frac{\omega_0}{2J} \frac{1}{T_{do}} \frac{V_s}{x_{d\Sigma}'} sin(y)$$

6. Case studies on robotic and electric power systems

6.2 The model of distributed power generators

Differential flatness can be also proven for the **model of the n-interconnected power generators**

The **flat output** is taken to be the vector of the turn angles of the n-power generators

$$y = [y_1^1, y_1^2, \cdots, y_1^n] \text{ or } y = \Delta\delta^1, \Delta\delta^2, \cdots, \Delta\delta^n$$

For the n-machines power generation system it holds

$$x_1^1 = y^1,\ x_1^2 = y^2,\ x_1^3 = y^3,\ \cdots,\ x_1^n = y^n$$

$$x_2^1 = \Delta\omega^1 = \dot{y}^1,\ x_2^2 = \Delta\omega^2 = \dot{y}^2,\ x_2^3 = \Delta\omega^3 = \dot{y}^3,\ \cdots,\ x_2^n = \Delta\omega^n = \dot{y}^n$$

Moreover, it holds

$$\dot{x}_2^i = -\frac{D_i}{2J_i}x_2^i + \frac{\omega_0}{2J_i}P_{mi} - \\ -\frac{\omega_0}{2J_i}[G_{ii}{x_3^i}^2 + x_3^i\sum_{j=1,j\neq i}^{n}[x_3^j G_{ij} sin(x_1^i - x_1^j - \alpha_{ij})] \quad (4)$$

or using the flat outputs notation

$$\ddot{y}^i = -\frac{D_i}{2J_i}\dot{y}^i + \frac{\omega_0}{2J_i}P_{mi} - \\ -\frac{\omega_0}{2J_i}[G_{ii}{x_3^i}^2 + x_3^i\sum_{j=1,j\neq i}^{n}[x_3^j G_{ij} sin(y^i - y^j - \alpha_{ij})]$$

6. Case studies on robotic and electric power systems

6.2 The model of distributed power generators

The **external mechanical torque** P_{mi} is considered to be a piecewise constant variable

From Eq. (4) and for one $i = 1, 2, \cdots, n$ has a system of n equations which can be solved with respect to the variables $x_3^i, i = 1, 2, \cdots, n$

Actually, all variables x_3^i, can be expressed as **differential functions of the flat outputs**

$$y^i, \; i = 1, 2, \cdots, n$$

and thus one has $x_3^i = f_{x_3}(y^1, y^2, \cdots, y^n)$

Moreover, from

$$\dot{E}_{q_i} = -\frac{1}{T_{d_i}} E'_{q_i} + \frac{1}{T_{d_{oi}}} \frac{x_{d_i} - x'_{d_i}}{x_{d_{\Sigma_i}}} V_{s_i} cos(\Delta\delta_i) + \frac{1}{T_{d_{oi}}} E_{f_i}$$

one can demonstrate that the control inputs $u_i = E_{f_i}$ can be expressed as **differential functions of the flat outputs** $y^i, \; i = 1, 2, \cdots, n$

Consequently, all state variables and the control inputs of the distributed power system can be expressed as differential functions of the flat outputs, and **the system is a differentially flat one.**

6. Case studies on robotic and electric power systems

6.2 The model of distributed power generators

Next, the **external mechanical torque** P_{mi} is considered to be time-varying

The effect of this torque is viewed as a **disturbance** to each power generator

In such a case for a model of *n=2* interconnected generators one obtains the **input-output linearized dynamics**

$$\dot{z}_3^i = a^i(x) + b_1{}^i g_1 u_1 + b_2{}^i g_2 u_2 + \tilde{d}^i \qquad \text{where} \quad z_3^i = \ddot{\delta}^i = \dot{\omega}^i$$

and

$$a^i = (\frac{D_i}{2J_i})^2 x_2^i + \frac{D_i \omega_0}{(2J_i)^2}[G_{ii} {x_3^i}^2 + x_3^i \sum_{j=1, j\neq i}^n x_3^j G_{ij} sin(x_1^i - x_1^j - \alpha_{ij})] -$$
$$-\frac{\omega_0}{2J_i}[G_{ii} x_3^i + \sum_{j=1, j\neq i}^n x_3^j G_{ij} sin(x_1^i - x_1^j - \alpha_{ij})(-\frac{1}{T'_{d_i}} x_3^i + (\frac{1}{T_{doi}} \frac{x_{d_i} - x'_{d_i}}{x'_{d\Sigma_i}} V_{s_i} cos(x_1^i))] -$$
$$-\frac{\omega_0}{2J_i} x_3^i \sum_{j=1, j\neq i}^n G_{ij} sin(x_1^i - x_1^j - \alpha_{ij})(-\frac{1}{T'_{d_i}} x_3^i + (\frac{1}{T_{doi}} \frac{x_{d_i} - x'_{d_i}}{x'_{d\Sigma_i}} V_{s_i} cos(x_1^i)) -$$
$$-\frac{\omega_0}{2J_i} x_3^i \sum_{j=1, j\neq i}^n x_3^j G_{ij} cos(x_1^i - x_1^j - \alpha_{ij}) x_2^i \frac{\omega_0}{2J_i} x_3^i \sum_{j=1, j\neq i}^n x_3^j G_{ij} cos(x_1^i - x_1^j - \alpha_{ij}) x_2^j$$

and

$$b_1^i = -\frac{\omega_0}{2J_i}[2G_{ii} x_3^i + \sum_{j=1, j\neq i}^n x_3^j G_{ij} sin(x_1^i - x_1^j - \alpha_{ij})] \frac{1}{T_{doi}}$$
$$b_2^i = -\frac{\omega_0}{2J_i} G_{i2} sin(x_1^i - x_1^2 - \alpha_{i2}) \frac{1}{T_{do2}}$$

while

$$\tilde{d}^i = -\frac{D_i \omega_0}{2{J_i}^2} P_m^i + \frac{\omega_0}{2J_i} \dot{P}_m^i$$

6. Case studies on robotic and electric power systems

6.2 The model of distributed power generators

For the **two interconnected generators** (*i=1,2*) one has the linearized dynamics

$$\begin{aligned}\dot{z}_1^i &= z_2^i \\ \dot{z}_2^i &= z_3^i \\ \dot{z}_3^i &= a^i(x) + {b_1}^i g_1 u_1 + {b_2}^i g_2 u_2 + \tilde{d}^i\end{aligned}$$

It is used that

$$\begin{aligned}\dot{z}_3^1 &= a^1(x) + {b_1}^1 g_1 u_1 + {b_2}^1 g_2 u_2 + \tilde{d}^1 \\ \dot{z}_3^2 &= a^2(x) + {b_1}^2 g_1 u_1 + {b_2}^2 g_2 u_2 + \tilde{d}^2\end{aligned}$$

or in matrix form

$$\dot{z}_3 = f_a(x) + Mu + \tilde{d}$$

where

$$z_3 = [z_3^1, z_3^2]^T,\ u = [u_1, u_2]^T \text{ and } \tilde{d} = [\tilde{d}_1, \tilde{d}_2]^T$$

and

$$f_a(x) = \begin{pmatrix} a^1(x) \\ a^2(x) \end{pmatrix},\quad M = \begin{pmatrix} {b_1}^1 g_1 & {b_2}^1 g_2 \\ {b_1}^2 g_1 & {b_2}^2 g_2 \end{pmatrix}$$

Setting, $v = f_a(x) + Mu + \tilde{d}$ one obtains

$$\begin{pmatrix} \dot{z}_1^i \\ \dot{z}_2^i \\ \dot{z}_3^i \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} z_1^i \\ z_2^i \\ z_3^i \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} (v^i + \tilde{d}^i)$$

6. Case studies on robotic and electric power systems

6.2 The model of distributed power generators

For **the model of the 2-area distributed power generation system** it holds that

$$x_{1,1}^{(3)} = f_1(x,t) + g_1(x,t)u + d_1$$
$$x_{1,2}^{(3)} = f_2(x,t) + g_2(x,t)u + d_2$$

By denoting

$$x_1 = x_{1,1},\ x_2 = \dot{x}_{1,1},\ x_3 = \ddot{x}_{1,1}$$

$$x_2 = x_{2,1},\ x_5 = \dot{x}_{2,1},\ x_6 = \ddot{x}_{2,1}$$

the **Brunovsky (canonical form) of the distributed power system** is obtained

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

where

$$v_1 = f_1(x) + g_{11}(x)u_1 + g_{12}(x)u_2$$
$$v_2 = f_2(x) + g_{21}(x)u_1 + g_{22}(x)u_2$$

- For the 2-area distributed power system differential flatness properties hold and one can apply the control scheme analyzed in Sections 3 and 4.

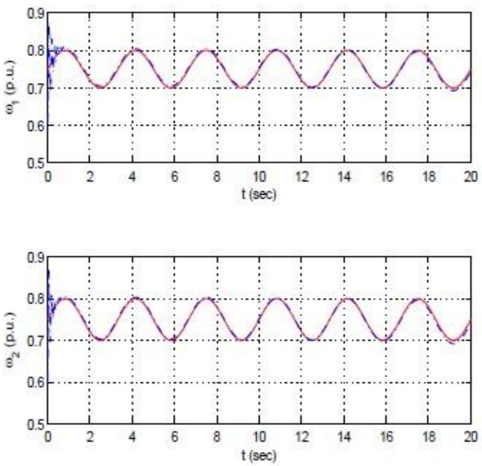
7. Simulation tests

The dynamic model of the distributed power generators was taken to be completely unknown, while the state vector could be partially measured

setpoint 1

ω_1

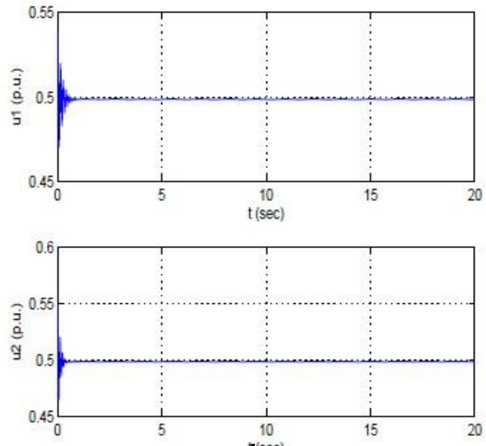
ω_2



time

u_1

u_2

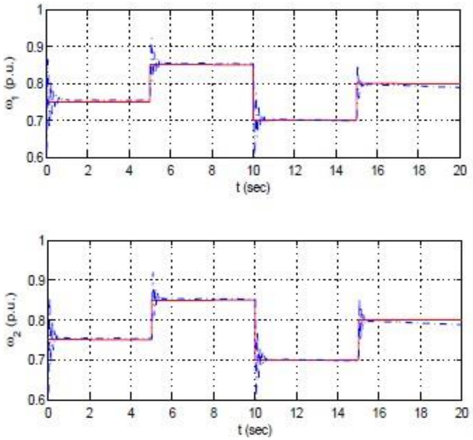


time

setpoint 2

ω_1

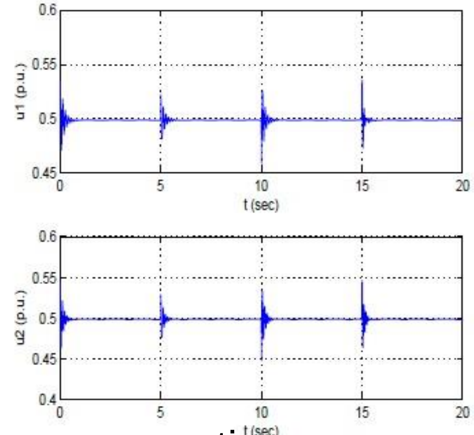
ω_2



time

u_1

u_2

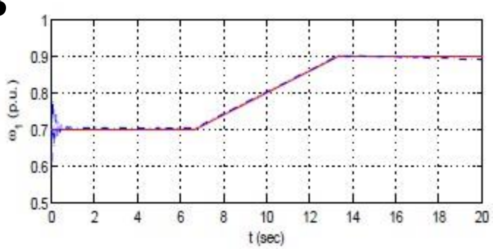


time

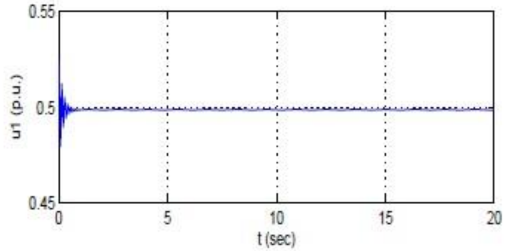
7. Simulation tests

setpoint 3

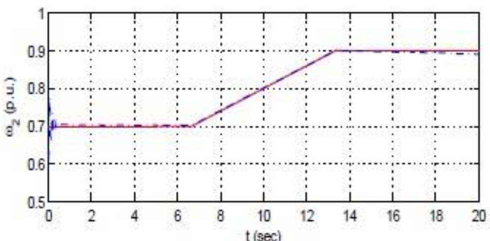
ω_1



u_1

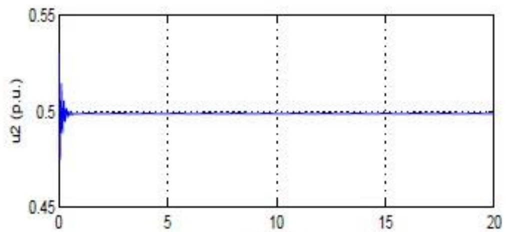


ω_2



time

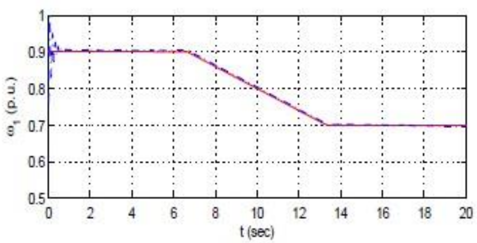
u_2



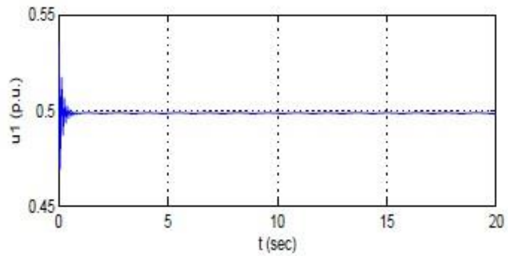
time

setpoint 4

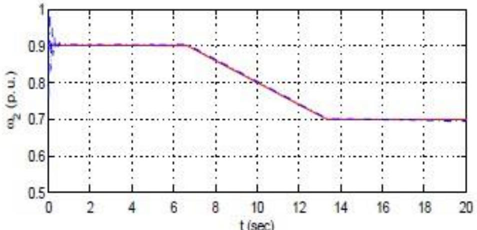
ω_1



u_1

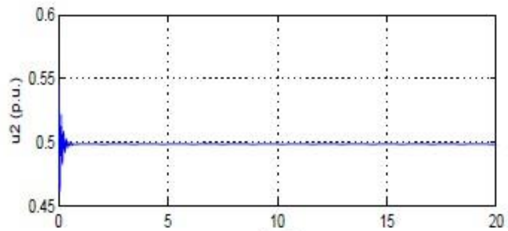


ω_2



time

u_2

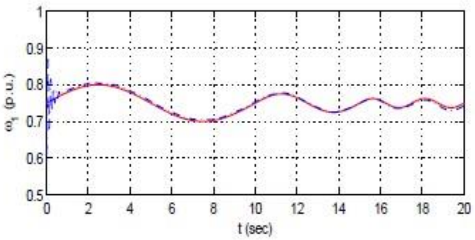


time

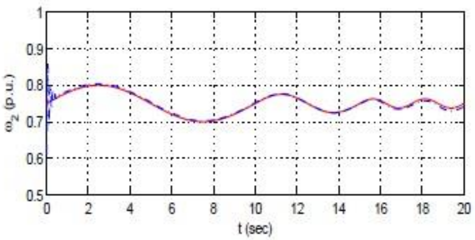
7. Simulation tests

setpoint 5

ω_1

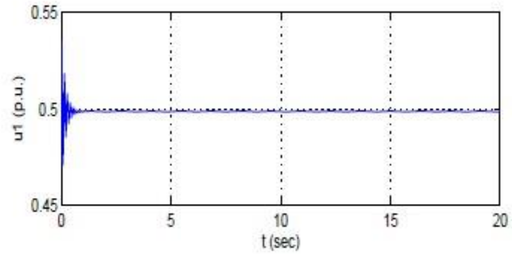


ω_2

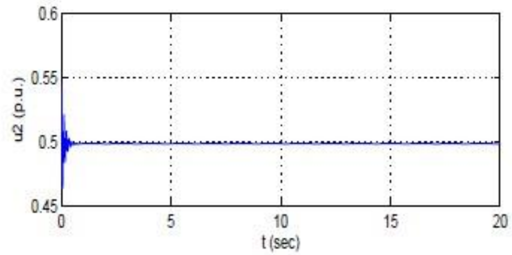


time

u_1



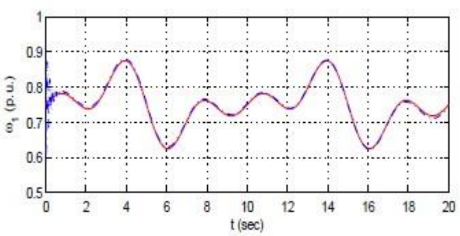
u_2



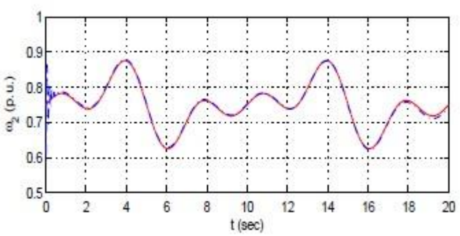
time

setpoint 6

ω_1

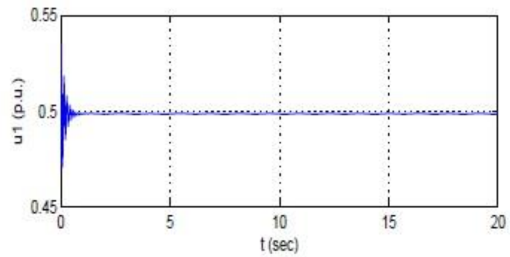


ω_2

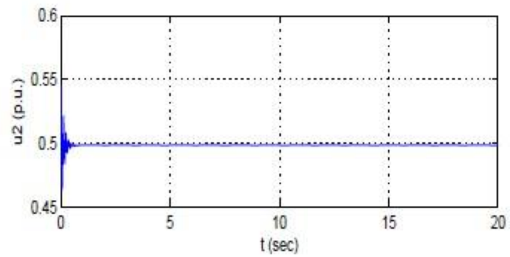


time

u_1



u_2



time

7. Simulation tests

7.2 Optimization-based modelling and control of a distributed power generators

Table I: RMSE of the power generator's state variables

parameter	ω_1	$\dot{\omega}_1$	ω_2	$\dot{\omega}_2$
$RMSE_1$	0.0035	0.0002	0.0034	0.0002
$RMSE_2$	0.0123	0.0545	0.0118	0.0602
$RMSE_3$	0.0035	0.0020	0.0035	0.0020
$RMSE_4$	0.0031	0.0020	0.0026	0.0020
$RMSE_5$	0.0034	0.0003	0.0033	0.0002
$RMSE_6$	0.0035	0.0003	0.0033	0.0002

The **tracking accuracy** of the control method was remarkable despite the fact that

(i) the **dynamic model** of the systems was **completely unknown**,

(ii) **only output feedback** was used in the implementation of the control scheme.

It has been also confirmed that the transient characteristics of the control scheme are quite satisfactory

The proposed **optimization-based modelling and control method** is of generic use and can be applied to a **wide class of nonlinear dynamical** systems of unknown model

8. Conclusions

• A **Lyapunov theory-based method of assured convergence and stability** has been developed. The method is suitable for **modelling** and optimization-based control in a wide class of nonlinear systems

• By exploiting the **differential flatness** properties of the **MIMO nonlinear model of the dynamical systems** this was transformed into the **linear canonical (Brunovsky) form.** For the latter description the design of a feedback controller was possible.

• Moreover, to cope with **unknown nonlinear terms** appearing in the new control inputs of the transformed state-space description of the system, the use of nonlinear regressors (neurofuzzy approximators) has been proposed..

• These estimators were online trained to **identify the unknown dynamics of the system** and the associated learning procedure was determined by the requirement the **first derivative of the control loop's Lyapunov function to be a negative one**.

• The computation of the control input required the **solution of two algebraic Riccati equation**.

• Through **Lyapunov stability analysis** it was proven that the closed loop satisfies the **H-infinity tracking performance criterion**, while also an **asymptotic stability condition** has been formulated.