Fault Tolerant Control

Márton, Lőrinc

SAPIENTIA - Hungarian University of Transylvania

Overview

1. Fault Tolerant Control - Basic Notions
2. Control Design in State Space
3. Introduction to System Reconfiguration
4. Fault Hiding - Virtual Sensors and Virtual Actuators
5. Networked Fault-Tolerant Control of Large-Scale Control Systems

Fault Tolerant Control - Basic Notions

Fault Tolerant Control

- Fault-tolerant control deals with the control of systems subject to faults.
- It explicitly takes into consideration the effects of faults on the behavior of the controlled system during control design.
- The main goal of fault-tolerant control is to prevent a fault (an unintended change in the behavior of the controlled system) from becoming a failure (the inability of the system to perform its mission).
- It relies on the results of fault diagnosis.
- A systematic, unified theoretical basis for fault tolerant control is not yet available. Moreover, the theory of the different fault tolerant control design appoaches are not always clearly connected to the theory of fault diagnosis. It is because of the large number of possible fault-induced effects.

The standard control problem

A control problem is defined by the triple: $< O,\ \Sigma,\ C >$

- O - *Control Objectives*: What the controlled system is expected to achieve when the control is active.
- Σ - *Constraints on the controlled system*: Functional relations that the behavior of the controlled system must satisfy over time. They are generally expressed by ordinary differential equations and algebraic equations or inequalities.
- C - *Set of admissible control laws*:
  - Open loop control - mapping from the time domain to the control space.
  - Closed loop control - mapping from the system output space $\times$ reference signal space to the control space.

The control problem in the case of faulty systems

Faulty systems:

- The constraint equations depend on a set of parameters $\Sigma = \Sigma(\theta)$.
- In the case of the fault- free system $\theta = \theta_n$ (nominal parameters)
- In the case of the faulty system $\theta = \theta_f$ (faulty parameters)
- The constraint equations could also change in the presence of faults: $\Sigma \rightarrow \Sigma_f$.

Available knowledge:

- The fault diagnosis is able to provide an estimate of the fault impact $\widehat{\Sigma}_f$
- The fault diagnosis is able to provide constraint set $\mathcal{S}_f$ which contains the faulty systems constraints $\Sigma_f \in \mathcal{S}_f$.
- The diagnosis detects and isolates the fault but it cannot provide any estimate of the fault impact.

The control problem in the case of faulty systems

- Admissible control laws: The occurrence of the faults may also change the set of admissible control laws. The new set of control laws are denoted by C_f.
- As a result of the fault the the control problem is transformed from $< O,\ \Sigma,\ C >$ into $< O,\ \Sigma_f,\ C_f >$.
- Suppose that no solution was found for the problem $< O,\ \Sigma_f,\ C_f >$. In this case the objective set has to be weakened. Let the new set of objectives be O_f. In this case the control problem is $< O_f,\ \Sigma_f,\ C_f >$. Obtaining the new set of objectives is a decision problem in which human operators are generally involved.

Passive vs. active fault tolerant control

- *Passive fault tolerant control*: The nominal control algorithm C is designed such that the system is able to achieve its given objectives in fault-free as well as in faulty cases, without any change in the control algorithm. In this case there is a common solution for the problems $< O,\ \Sigma,\ C >$ and $< O,\ \Sigma_f,\ C_f >$ for each f which is solvable using $C_f = C$.
- *Active fault tolerant control*: The control law is changed when the fault occurs, so the ability of the system to achieve O is preserved, using a control law that is adapted to each faulty situation. In this case $< O,\ \Sigma_f,\ C_f >$ has different solutions for different faults.

Passive fault tolerant control

- The fault is treated as disturbance or modeling uncertainty.

- Robust control techniques are generally applied.

Active fault tolerant control

Fault Accommodation:

- Fault Accommodation: solve the control problem $< O, \widehat{\Sigma}_f, C_f >$
- An estimate of the faulty system or system class, which contains the faulty system, is necessary.
- It involves control design at the moment when the fault is estimated.
- Robust control techniques are generally applied.

Active fault tolerant control

System reconfiguration:

- It is assumed that only isolation information is available from the fault diagnosis procedure.
- In this case the faulty system model is partially unknown.
- A fault problem can only be set if a faulty components are switched off and try to achieve the objective by using the healthy part of the controlled system.
- Let the faulty system constraints be: $\Sigma_f = \Sigma' \bigcup \Sigma'_f$. Σ'_f is unknown.
- System reconfiguration: solve the control problem $< O, \Sigma', C_f >$
- The Input-Output relations between the controller and system are changed.
- In many cases reconfiguration can only be solved if there is redundancy in the control system (e.g. multiple sensors with the same purpose)

Control Design in State Space

Controllable Systems

- Let the state space realization of an LTI process system:

$$\dot{x} = Ax + Bu,\ x(t_0) = x_0$$
$$y = Cx + Du$$

- A state x_0 is *controllable* at time t_0 if there exists an input $u(t)$ that transfers the state (x) from x_0 to the origin $x(t_f) = 0$ in a finite time t_f. The system is called controllable at time t_0 if every state x_0 in the state-space is controllable.
- The LTI system is controllable if the controllability matrix

$$M_c = [B\ AB\ \ldots\ A^{n-1}B]$$

satisfies $rank(M_c) = n$, where $n = dim(x)$.

Controlability

- Let the state dynamics of an LTI process system:

$$\dot{x} = Ax + Bu,\ x(t_0) = 0$$

- In the complex domain

$$sx(s) = Ax(s) + Bu(s)$$

$$x(s) = (sI - A)^{-1}Bu(s) = \begin{bmatrix} \zeta_1(s) \\ \dots \\ \zeta_n(s) \end{bmatrix} \begin{bmatrix} u_1(s) \\ \dots \\ u_m(s) \end{bmatrix}$$

- The system is *uncontrollable* if any row of $[(sI - A)^{-1}B]$ is 0.
- The system is *uncontrollable* if there exists a linear dependence between the columns of $[(sI - A)^{-1}B]$ (e.g. $\zeta_1^T = \alpha\zeta_2^T$). This condition implies that the responses to u of some states are linearly dependent, they cannot be *independently* manipulated by the input u.

State feedback

- Let the state dynamics of a controllable process system:

$$\dot{x} = Ax + Bu,\ x(t_0) = x_0$$

- Assume that x is measurable. Formulate the control input as

$$u = -Kx$$

- The state dynamics of the *controlled system*

$$\dot{x} = (A - BK)x,\ x(t_0) = x_0$$

- The state feedback control design problem: find K such that the controlled system satisfies prescribed objectives. E.g. find K such that $(A - BK)$ has prescribed eigenvalues.

Linear quadratic (LQ) control

- Let the state dynamics of the process system:

$$\dot{x} = Ax + Bu,\ x(0) = x_0$$

- Let the functional, called cost,

$$J = \frac{1}{2}\int_0^\infty \left(x^T Q x + u^T R u\right) dt$$

  where Q, R are symmetric and positive definite.
- Objective: transfer the system state from $x(0) = x_0$ to $x(\infty) = 0$ while solving the optimization problem

$$\begin{aligned} &min_u J \\ &\text{such that } \dot{x} = Ax + Bu,\ x(t_0) = x_0 \end{aligned}$$

Linear quadratic (LQ) control

- Introduce the notation

$$L(x,u) = \frac{1}{2}\left(x(t)^T Q x(t) + u(t)^T R u(t)\right)$$

- Augment the functional with a costate

$$J_\lambda = \int_0^\infty \left(L(x,u) + \lambda(t)(Ax(t) + Bu(t) - \dot{x})\right) dt$$

  The costate $\lambda(t) \in R^n$ can be any vector, since it multiplies $Ax(t) + Bu(t) - \dot{x} = 0$.

- Along the optimal trajectories (in the minimum) the *variations* in J (and J_λ) should vanish.
- The variation of J_λ:

$$\delta J_\lambda = \int_0^\infty \left(\frac{\partial L(x,u)}{\partial x}\delta x + \frac{\partial L(x,u)}{\partial u}\delta u + \lambda(t)^T(A\delta x(t) + B\delta u(t) - \delta\dot{x})\right) dt$$

Let's calculate the terms of δJ_λ

$$\delta J_\lambda = \int_0^\infty \left(\frac{\partial L(x,u)}{\partial x}\delta x + \frac{\partial L(x,u)}{\partial u}\delta u + \lambda(t)^T(A\delta x(t) + B\delta u(t) - \delta\dot{x}) \right) dt$$

- It directly yields that:

$$\begin{aligned} \frac{\partial L(x,u)}{\partial x} &= \frac{\partial \frac{1}{2}\left(x(t)^T Q x(t) + u(t)^T R u(t)\right)}{\partial x} = x^T Q \\ \frac{\partial L(x,u)}{\partial u} &= u^T R \end{aligned}$$

- By using integration by parts:

$$-\int_0^\infty \lambda(t)^T \delta\dot{x} dt = \lambda(0)^T \delta\dot{x}(0) - \lambda(\infty)^T \delta\dot{x}(\infty) + \int_0^\infty \dot{\lambda}(t)^T \delta x dt$$

- $\delta x(0) = 0$ since we cannot vary the constant initial condition of the state x_0 by changing something later in time.
- $\lambda(t)$ is chosen such that $\lambda(\infty) = 0$.

- The variation of J_λ yields in the form:

$$\delta J_\lambda = \int_0^\infty \Big((\lambda^T A + x^T Q - \dot{\lambda}^T)\delta x + (\lambda^T B + u^T R)\delta u \Big)\, dt$$

- To vanish the variations of J_λ to following equations should hold:

$$\begin{aligned} & \lambda^T B + u^T R = 0 \\ & \dot{\lambda}^T = \lambda^T A + x^T Q,\ \lambda(\infty) = 0 \\ \textit{such that} \quad & \dot{x} = Ax + Bu,\ x(0) = x_0 \end{aligned}$$

- The state x propagates forward in time, while the costate λ propagates backward, from ∞ to 0.

Linear quadratic (LQ) control

- Since the system is linear we could try to find the costate in the form:

$$\lambda = Px$$

- It yields:

$$\begin{aligned}
&(Px)^T B + u^T R = 0 \\
&(\dot{P}x)^T = (Px)^T A + x^T Q,\ x(\infty) = 0 \\
&\dot{x} = Ax + Bu,\ x(0) = x_0
\end{aligned}$$

- We obtain:

$$\begin{aligned}
&u = -R^{-1} B^T P x \\
&PAx + A^T Px - PBR^{-1} B^T Px + Qx + \dot{P} = 0
\end{aligned}$$

- The Riccati equation above has to hold $\forall x$. The steady state solution ($\dot{P} = 0$) is called the *matrix Riccati equation*:

$$PA + A^T P - PBR^{-1} B^T P + Q = 0$$

Linear quadratic (LQ) control

Summary of the control design:

- Let state- and input matrices of the state space model be A and B.
- The design parameters $Q > 0$, $R > 0$ are given.
- Solve the matrix Riccati equation:

$$PA + A^T P - PBR^{-1}B^T P + Q = 0$$

- Compute the feedback gain:

$$K = R^{-1}B^T P$$

- Implement the control:

$$u = -Kx$$

LQ Control Design Example

- Let a Single Input - Single State system: $\dot{x} = ax + bu$. The system is controllable for $b \neq 0$.
- The quadratic cost functional has the form: $J = \int_0^\infty (qx^2 + ru^2)dt$, $q, r > 0$.
- The Riccati equation:

$$pa + ap - pb\frac{1}{r}bp + q = 0$$

$$-\frac{b}{r}p^2 + 2ap + q = 0$$

- The positive solution of the Riccati equation: $p = \frac{a+\sqrt{a^2+b^2\frac{q}{r}}}{\frac{b^2}{r}}$
- The feedback gain: $k = \frac{1}{r}bp = \frac{a+\sqrt{a^2+b^2\frac{q}{r}}}{b}$.
- The closed loop system with state feedback ($u = -kx$):

$$\dot{x} = -\sqrt{a^2 + b^2\frac{q}{r}}\, x$$

- The pole of the closed loop system: $\lambda = -\sqrt{a^2 + b^2\frac{q}{r}}$

Observer design problem revisited

- Let an observable LTI system:

$$\dot{x} = Ax + Bu,\ x(0) = x_0$$
$$y = Cx + Du$$

- State observer of an LTI system:

$$\dot{\widehat{x}} = A\widehat{x} + Bu + G(y - C\,\widehat{x} - Du)$$

- Observation error ($e = x - \widehat{x}$) dynamics:

$$\dot{e} = (A - GC)e$$

- The solution of the LQ problem can be applied to design a proper G. The design has to be performed for A^T and C^T.

Linear quadratic (LQ) observer

Summary of the observer design:

- Let state- and output matrices of the state space model be A and C.
- The design parameters $Q > 0$, $R > 0$ are given.
- Solve the matrix Riccati equation:

$$PA^T + AP - PC^T R^{-1} CP + Q = 0$$

- Compute the feedback gain:

$$G = PC^T R^{-1}$$

- Implement the observer:

$$\dot{\hat{x}} = A\hat{x} + Bu + G(y - C\,\hat{x} - Du)$$

Control with output feedback

- Let an observable and controllable LTI system:

$$\dot{x} = Ax + Bu,\ x(0) = x_0$$
$$y = Cx$$

- Consider the dynamic controller

$$\dot{\widehat{x}} = A\widehat{x} + Bu + G(y - C\,\widehat{x} - Du)$$
$$u = -K\,\widehat{x}$$

- The closed loop dynamics ($e = x - \widehat{x}$):

$$\begin{pmatrix} \dot{x} \\ \dot{e} \end{pmatrix} = \begin{bmatrix} A - BK & BK \\ 0 & A - GC \end{bmatrix} \begin{pmatrix} x \\ e \end{pmatrix}$$

- *Separation principle*: Since this state matrix is a triangular hypermatrix, the observer and feedback gains (G, K) can be independently designed.

Introduction to System Reconfiguration

Control System Model for Reconfiguration

- Let the model of the fault-free process system:

$$\dot{x} = Ax + Bu,\ x(0) = x_0$$
$$y = Cx$$

- Consider the dynamic controller in a general form which is able to satsfy the prescribed control objectives for the fault free case:

$$y_C = y$$
$$\dot{x}_C = A_C x_C + B_C(w - y_C),\ x_C(0) = x_{C0}$$
$$u_C = C_C x_C + D_C(w - y_C)$$
$$u = u_C$$

  Here w is the reference input (setpoint).

- Example: proportional control

$$u = K_P(w - y)$$
$$\text{i.e. } A_C = 0,\ B_C = 0,\ C_C = 0,\ D_C = K_P = diag(k_{Pi)}) > 0.$$

Faulty system

- Let the faulty system model:

$$\dot{x}_f = A_f x_f + B_f u,\ x_f(0) = x_{f0}$$
$$y_f = C_f x_f$$

- Actuator fault:

$$A_f = A,\ B_f \neq B,\ C_f = C$$

- Sensor fault:

$$A_f = A,\ B_f = B,\ C_f \neq C$$

- Internal fault:

$$A_f \neq A,\ B_f = B,\ C_f = C$$

Reconfiguration block

- It is placed between the faulty plan and the nominal controller.
- It generally can be modelled as:

$$\dot{x}_R = A_R x_R + B_{Ru} u_C + B_{Ry} y_f, \; x_R(0) = x_{R0}$$
$$y_C = C_{Ry} x_R + D_{uy} u_C + D_{yy} y_f$$
$$u_f = C_{Ru} x_R + D_{uu} u_C + D_{uy} y_f$$

- Special case: Input-Output separated, static reconfiguration block:

$$y_c = D_{yy} y_f$$
$$u_f = D_{uu} u_C$$

Reconfiguration goals

- *Stabilization goal*: restore the stability of the control loop. The reconfigured control system is stable iff $\forall\ \varepsilon > 0\ \exists\ \delta > 0$ such that if $\|w\|_\infty < \delta$ then $\|x_f\|_\infty, \|u_f\|_\infty < \varepsilon$.

- *Weak reconfiguration goal*: restore the steady state of the control loop. The reconfigured control loop satisfies the weak reconfiguration goal iff $lim_{t\to\infty}(y - y_f) = 0\ \forall\ w$ and x_{f0}.

- *Strong reconfiguration goal*: restore the dynamic behavior of the control loop. The reconfigured control loop satisfies the strong reconfiguration goal iff $y(t) = y_f(t)\ \forall\ w$ and x_{f0}.

Reconfiguration goals

- *Direct reconfiguration goal*: restore the states of the plant. The reconfigured control loop satisfies the direct reconfiguration goal iff $x(t) = x_f(t)\ \forall\ w$ and x_{f0}.

- *Fault hiding goal*: hide the fault from the controller view. The reconfigured control loop satisfies the fault hiding goal iff $y_C(t) = y(t)$ $\forall\ w$ and x_{f0}

Static Reconfiguration - Actuator Fault

- Recall the model of the fault-free process system:

$$\dot{x} = Ax + Bu,\ x(0) = x_0$$
$$y = Cx$$

- Let the model of the faulty system model with actuator fault:

$$\dot{x}_f = Ax_f + B_f u_f,\ x_f(0) = x_{f0}$$
$$y_f = Cx_f$$

- Direct reconfiguration goal: $x(t) = x_f(t)$ or $\dot{x}(t) = \dot{x}_f(t)$
- To solve the direct reconfiguration goal it is necessary that $B_f u_f = Bu$ $\forall\ u$.

Static Reconfiguration - Actuator Fault

- The linear solution

$$u_f = D_{uu} u$$

- Hence we have to solve the matrix equation

$$B_f D_{uu} = B$$

- The equation is solvable if $rank(B_f) = rank(B_f\ B)$.

Static Reconfiguration - Sensor Fault

- Recall the model of the fault-free process system:

$$\dot{x} = Ax + Bu,\ x(0) = x_0$$
$$y = Cx$$
$$y_C = y$$

- Let the model of the faulty system model with sensor fault:

$$\dot{x} = Ax + Bu,\ x_f(0) = x_{f0}$$
$$y_f = C_f x$$

- Fault hiding goal: $y_C(t) = y(t)$ in the presence of fault.
- The linear solution: $y_C = D_{yy} y_f$
- The relation $D_{yy} C_f x = Cx$ should hold, $\forall\ x$.
- The equation $D_{yy} C_f = C$ is solvable if

$$rank(C_f) = rank \begin{pmatrix} C_f \\ C \end{pmatrix}$$

.

Pseudo-Inverse Method

- The pseudo inverse is a generalization of the inverse matrix.
- Definition: A^+ is the pseudo-inverse of A if

$$AA^+A = A$$
$$A^+AA^+ = A^+$$
$$(AA^+)^T = AA^+$$
$$(A^+A)^T = A^+A$$

  A^+ always exists and it is unique.
- Theorem: $A^+ = lim_{\delta \to 0}(A^TA + \delta^2 I)^{-1}A^T = lim_{\delta \to 0}A^T(AA^T + \delta^2 I)^{-1}$
- If A has full column rank, then $A^+ = (A^TA)^{-1}A^T$ (A is right invertible).
- If A has full row rank, then $A^+ = A^T(AA^T)^{-1}$ (A is left invertible).

Pseudo-Inverse Method

- Let the matrix linear equation $AX = B$ which is solvable if $rank(A) = rank(A\ B)$.
- *Theorem*: $rank(A) = rank(A\ B)$ iff $AA^+B = B$.
- *Theorem*: If $AX = B$ is solvable then the solution is $X = A^+B + (I - A^+A)Y$, where Y is arbitrary.

- Let the matrix linear equation $XA = B$ which is solvable if $rank(A) = rank\begin{pmatrix} A \\ B \end{pmatrix}$
- *Theorem*: $rank(A) = rank\begin{pmatrix} A \\ B \end{pmatrix}$ iff $BA^+A = B$.
- *Theorem*: If $XA = B$ is solvable then the solution is $X = BA^+ + Y(I - AA^+)$, where Y is arbitrary.

Pseudo-Inverse Method

- The pseudo inverse can be computed using Singular Value Decomposition (SVD) even in rank deficient case.
- Singular Value Decomposition: Each matrix can be written in the form $A = U^T \Sigma V$, where $U^T U = I$, i.e. U is unitary matrix, $V^T V = I$ and $\Sigma = diag(\sigma_1\ \sigma_2\ \ldots \sigma_r)$, $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r \geq 0$.
- Dimensions: $A \in \mathbb{R}^{m \times n}$, $U \in \mathbb{R}^{m \times r}$, $\Sigma \in \mathbb{R}^{r \times r}$, $V \in \mathbb{R}^{r \times n}$, $r = min(m, n)$.
- Pseudo inverse computation in the general case: $A^+ = V \Sigma^+ U^T$, where $\Sigma^+ = diag(\sigma_1^+\ \sigma_2^+\ \ldots \sigma_r^+)$, where $\sigma_i^+ = 1/\sigma_i$ if $\sigma_i \neq 0$ and $\sigma_i^+ = 0$ if $\sigma_i = 0$.
- The number of non-zero singular values is equal to the rank of A.

State Feedback Redesign

- Recall the model of the fault-free process system with state feedback:

$$\dot{x} = Ax + Bu,\ x(0) = x_0$$
$$u = -Kx$$

- Let the model of the controlled faulty system:

$$\dot{x}_f = A_f x_f + B_f u,\ x_f(0) = x_{f0}$$
$$u_f = -K_f x_f$$

- Direct closed-loop reconfiguration problem: design K_f such that $x_f(t) = x(t)$.
- Solve the equation

$$A_f - B_f K_f = A - BK$$
$$\text{i.e. } B_f K_f = \underbrace{A - BK - A_f}_{B_{of}}$$

- The equation is solvable if $rank(B_f) = rank(B_f\ B_{of})$.

Pseudo-Inverse Method

- *Pseudo-inverse method*: If the problem dose not have a precise solution, design K_f such to maintain as much similarity as possible between the reconfigured control loop and the fault free control loop.
- It can be formulated as an optimization problem:

$$min_{K_f} \|(A - BK) - (A_f - B_f K_f)\|_F$$

  Here $\|\cdot\|_F$ is the Frombenius norm, i.e. $\|A\|_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij}^2}$.
- The solution of the optimization problem also leads back to $K_f = B_f^+ B_{of}$,
  where B_f^+ is the pseudo-inverse of B_f.

Pseudo-Inverse Method

- The approximate pseudo-inverse method does not necessarily lead to a stable solution.
- *Example*: Let $A = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$, $B = \begin{bmatrix} 1 \\ 5 \end{bmatrix}$, $C = I$.
- With the feedback gain $K = [-1\ 0]$ the eigenvalues of the closed loop system $A - BK$ are $\lambda_1 = -1$, $\lambda_2 = -2$ (stable control loop).
- Let $A_f = A$, $B_f = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$, $C_f = C$.
- With the feedback gain $K_f = B_f^+(A - A_f - BK) = [-2\ 0]$ the eigenvalues of the closed loops system $A_f - B_f K_f$ are $\lambda_1 = -1$, $\lambda_2 = 1$ (unstable reconfigured control loop).

Pseudo-Inverse Method

- It is well known that high feedback gains lead to instability in the control loop.
- Modified pseudo-inverse method: Solve the optimization problem:

$$\begin{aligned} &min_{K_f} \|(A - BK) - (A_f - B_f K_f)\|_F \\ &subject\ to\ |K_f(i,j)| \leq \delta\ \forall i,j \end{aligned}$$

- *Theorem*: For single input systems ($u \in R$) the solution of the optimization problem above is:

$$K_f(i,j) = \begin{cases} \widehat{K}_f(i,j)\ if\ |K_f(i,j)| \leq \delta \\ sign(\widehat{K}_f(i,j))\delta\ otherwise \end{cases}$$

where $\widehat{K}_f = B_f^+(A - BK - A_f)$.

Fault Hiding - Virtual Sensors and Virtual Actuators

Reconfiguration Using a Virtual Sensor

- The concept of the *virtual sensor*: when a sensor is at fault, an observer is used to calculate a replacement value.
- Let the model of the fault-free system:

$$\dot{x} = Ax + Bu,\ x(0) = x_0$$
$$y = Cx$$

- Let the model of the faulty system with sensor fault ($u_f = u_C$):

$$\dot{x} = Ax + Bu_C,\ x(0) = x_0$$
$$y_f = C_f x$$

- Since y_f cannot be used with the existing controller, a reconfiguration block is to be found that generates a suitable signal y_C from y_f and u_f.
- *Fault hiding* goal: It is required that the output of the reconfigured plant be identical to the output of the nominal plant.

Virtual Sensor Design

- Design a state observer for the faulty system:

$$\dot{x}_R = Ax_R + Bu_C + G_R(y_f - C_f x_R)$$

Solvability condition: (A, C_f) observable.

- The generated output for the controller:

$$y_C = Cx_R$$

- Analysis: Let $e = x_R - x_f$. The dynamics of the faulty system together with the reconfiguration block:

$$\begin{pmatrix} \dot{x}_f \\ \dot{e} \end{pmatrix} = \begin{bmatrix} A & 0 \\ 0 & A - G_R C_f \end{bmatrix} \begin{pmatrix} x_f \\ e \end{pmatrix} + \begin{pmatrix} B \\ 0 \end{pmatrix} u_C$$
$$y_C = C(x_f + e)$$

Note that the dynamics of e is not coupled to the dynamics of the faulty system.

Detailed analysis of Virtual Sensor

- Consider that the faulty system is also affected by the disturbance:

$$\dot{x}_f = Ax_f + Bu_f + B_d d$$

- Consider a dynamic linear controller:

$$\begin{aligned} y_C &= C(x_f + e) \\ \dot{x}_C &= A_C x_C + B_C(w - y_C),\ x_C(0) = x_{C0} \\ u &= C_C x_C + D_C(w - y_C) \end{aligned}$$

  Here w is the reference input.

- The dynamics of the reconfigured control loop:

$$\begin{pmatrix} \dot{x}_f \\ \dot{x}_C \\ \dot{e} \end{pmatrix} = \begin{bmatrix} A - BD_C C & BC_C & 0 \\ B_C C & A_C & B_C C \\ 0 & 0 & A - G_R C_f \end{bmatrix} \begin{pmatrix} x_f \\ x_C \\ e \end{pmatrix} + \begin{pmatrix} B_d \\ 0 \\ B_d \end{pmatrix} d + \begin{pmatrix} BD_C \\ B_C \\ 0 \end{pmatrix} w$$

- The estimation error e is affected by affected by the disturbance d!

Reconfiguration Using a Virtual Actuator

- The idea of a virtual actuator is to use the input signal meant for the nominal process and to transform it into a signal useful for the remaining actuators of the faulty plant.
- Let the model of the fault-free system:

$$\dot{x} = Ax + Bu,\ x(0) = x_0$$
$$y = Cx$$

- Let the model of the faulty system model with sensor fault ($u_f = u_C$):

$$\dot{x}_f = Ax_f + B_f u_f,\ x(0) = x_0$$
$$y_f = Cx_f$$

- B_f modifies the control input. A novel reconfiguration block is to be found that generates a suitable control signal u_f based on the $y_C = y$ and u_C.
- *Fault hiding* goal: It is required that the reconfigured process (faulty process + the reconfiguration clock) has the same input/output behavior as the nominal plant, and therefore the nominal controller is not affected by the fault.

Virtual Actuator Design

- Implement the *model* of the fault free system in the reconfiguration block and let the nominal controller to compute u_C based of the output of this model:

$$\dot{x} = Ax + Bu_C,\ x(0) = x_0$$
$$y = Cx$$

- The control is computed such to ensure that $x - x_f$ converges to zero. It has the form:

$$u_f = K_R(x - x_f)$$

- Problem: The states of the faulty system (x_f) are not measurable.

Virtual Actuator Design

- Let $x_R = x - x_f$.
- The dynamics of x_R:

$$\dot{x} = Ax + Bu_C$$
$$\dot{x}_f = Ax_F + Bu_f$$
$$\text{i.e.}$$
$$\dot{x}_R = Ax_R + Bu_C - B_f u_f$$

- The plant output to be presented for the controller

$$y_C = Cx = C(x_f + x_R)$$
$$\text{i.e.}$$
$$y_C = y_f + Cx_R$$

- The control for the faulty system:

$$u_f = K_R x_R$$

Virtual Actuator Design

- The dynamics of the virtual actuator:

$$\dot{x}_R = (A - B_f K_R)x_R + Bu_C$$

- The state matrix $(A - B_f K_R)$ has to be stable, i.e. the *solvability condition* of the virtual actuator design is: (A, B_f) has to be controllable.
- Analysis: The dynamics of the the reconfiguration block:

$$\begin{pmatrix} \dot{x} \\ \dot{x}_R \end{pmatrix} = \begin{bmatrix} A & 0 \\ 0 & A - B_f K_R \end{bmatrix} \begin{pmatrix} x \\ x_R \end{pmatrix} + \begin{pmatrix} B \\ B \end{pmatrix} u_C$$
$$y_C = C x_R$$

  Note that the dynamics of x_R is uncoupled form the dynamics of the system modell (which is controlled).
- Problem: even in the case of stable dynamics x_R tends to zero only if u_C tends to zero.
  It is satisfied in the case of stabilizing controllers that have the form $u_C = -Kx$, but generally $u_C \neq 0$.

Detailed analysis of Virtual Actuator

- Consider that the faulty system is also affected by the disturbance:

$$\dot{x}_f = Ax_f + B_f u_f + B_d d$$

- Recall $x = x_f + x_R$. Consider a dynamic linear controller:

$$\begin{aligned} y_C &= Cx \\ \dot{x}_C &= A_C x_C + B_C(w - y_C),\ x_C(0) = x_{C0} \\ u &= C_C x_C + D_C(w - y_C) \end{aligned}$$

  Here w is the reference input.

- The dynamics of the reconfigured control loop:

$$\begin{pmatrix} \dot{x}_R \\ \dot{x} \\ \dot{x}_C \end{pmatrix} = \begin{bmatrix} A - B_f K_R & -BD_C C & 0 \\ 0 & A - BD_C C & BC_C \\ 0 & -B_C C & A_C \end{bmatrix} \begin{pmatrix} x_R \\ x \\ x_C \end{pmatrix} + \begin{pmatrix} 0 \\ B_d \\ 0 \end{pmatrix} d + \begin{pmatrix} BD_C \\ BD_C \\ B_C \end{pmatrix} w$$

- The state deviation x_R is NOT affected directly by the disturbance d! However, it is affected by the reference signal w!

Setpoint tracking

- The original form of the virtual actuator cannot assure setpoint tracking, as $lim_{t\to\infty} x_R \neq 0$ if $w \neq 0$.
- Problem (weak reconfiguration): ensure that $\bar{x}_R = lim_{t\to\infty} x_R = 0$ for w constant.
- Solution: Extend the reconfigured control signal with a *feed-forward term*:

$$u_f = K_R x_R + F_R u_C$$

- F_R has to be designed such that u_f solves the proposed weak reconfiguration problem.

Setpoint tracking design

- The state dynamics of the reconfiguration block:

$$\dot{x}_R = (A - B_f K_R)x_R + Bu_C - B_f F_R u_C$$

- The steady state equation

$$0 = (A - B_f K_R)\bar{x}_R + B\bar{u}_C - B_f F_R \bar{u}_C$$
$$\text{i.e. } \bar{x}_R = (A - B_f K_R)^{-1}(B_f F_R - B)\bar{u}_C$$

- To ensure that $\bar{x}_R = 0 \ \forall \bar{u}_C$, F_R has to be computed such that

$$(A - B_f K_R)^{-1}(B_f F_R - B) = 0$$
$$\text{i.e. } \underbrace{(A - B_f K_R)^{-1} B_f}_{M_R} F_R = \underbrace{(A - B_f K_R)^{-1} B}_{N_R}$$

- The problem is solvable if $A - B_f K_R$ is invertible, which is true since it is Huwritz, and $rank(M_R) = rank(M_R \ N_R)$. The solution is:

$$F_R = M_R^+ N_R$$

Disturbance compensation in virtual sensors

- The principle of direct compensation can be applied to deal with disturbances in the virtual sensors.
- Recall the form of the virtual sensor ($e = x_R - x_f$) :

$$\dot{x}_R = Ax_R + Bu_C + G_R(y_f - C_f x_R)$$
$$y_C = Cx_R$$
$$\text{i.e.}$$
$$\dot{e} = (A - G_R C_f)e$$
$$y_C = y_f + Ce$$

- Consider that the faulty system is affected by disturbance:

$$\dot{x}_f = Ax_f + Bu_f + B_d d$$

- As it was discussed, the disturbance directly influences the dynamics of the virtual estimator:

$$\dot{e} = (A - G_R C_f)e + B_d d$$

Disturbance compensation in virtual sensors

- Extend the output of the disturbance affected virtual sensor with an extra term as $(e = x_R - x_f)$

$$\dot{e} = (A - G_R C_f)e + B_d d$$
$$y_C = Cx_R + F_R(y_f - C_f x_R)$$

- Problem: ensure that $\bar{e}_y = lim_{t\to\infty}(y_C - Cx_f) = 0$.
- Computations again... :

$$0 = (A - G_R C_f)\bar{e} + B_d \bar{d}$$
$$\bar{e}_y = C\bar{e} - F_R C_f \bar{e}$$
$$\bar{e}_y = (C - F_R C_f)(A - G_R C_f)^{-1} B_d \bar{d}$$

Disturbance compensation in virtual sensors

- To ensure that $\bar{e}_y = 0 \ \forall \bar{d}$, F_R has to be computed such that

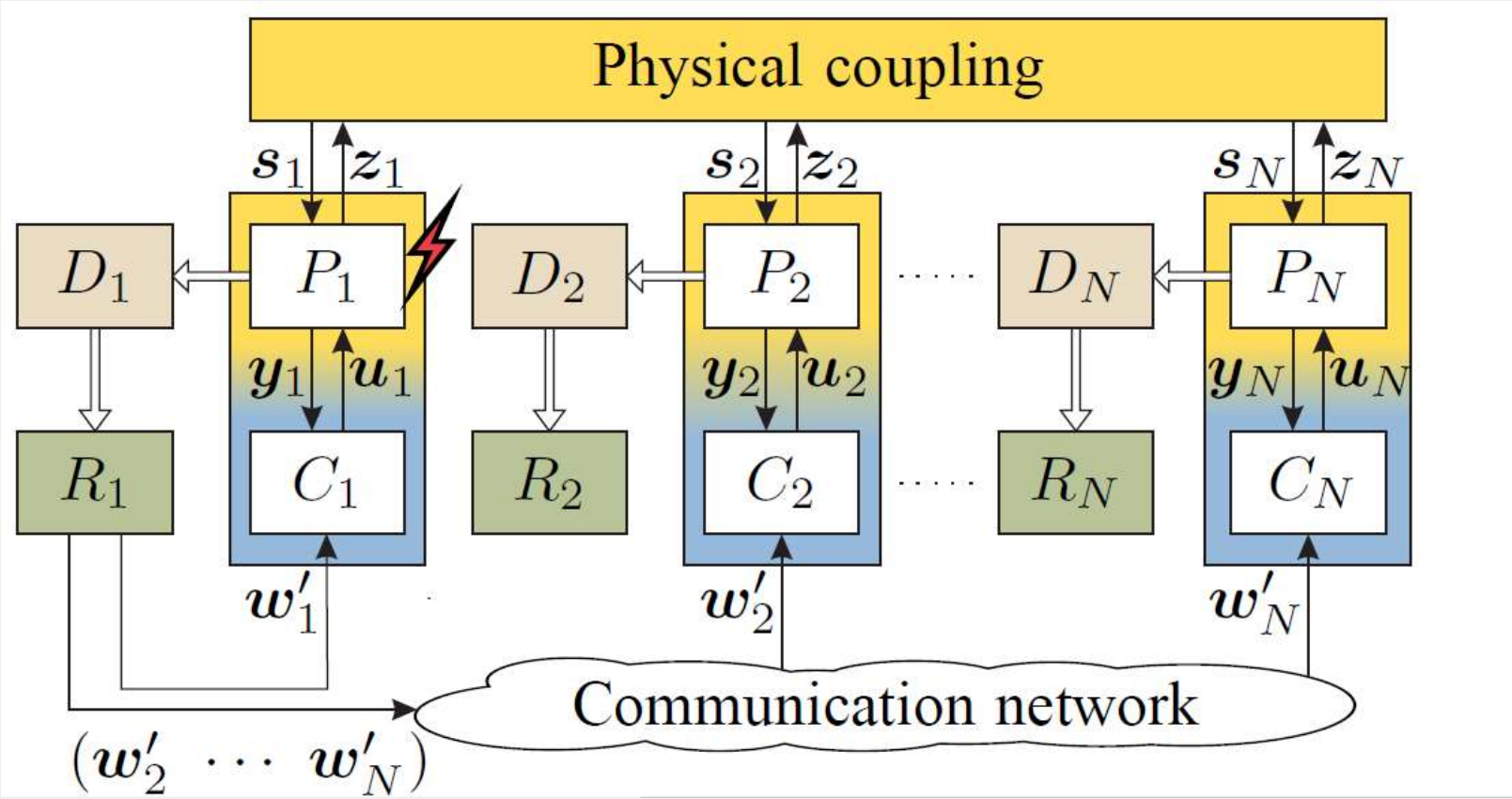
$$(C - F_R C_f)(A - G_R C_f)^{-1} = 0$$
$$\text{i.e. } F_R \underbrace{C_f(A - G_R C_f)^{-1}}_{M_R} = \underbrace{C(A - G_R C_f)^{-1}}_{N_R}$$

- The problem is solvable if $A - G_R C_f$ is invertible, which is true since it is Huwritz, and $rank(M_R) = rank \begin{pmatrix} M_R \\ N_R \end{pmatrix}$. The solution is:

$$F_R = N_R M_R^+$$

Networked Fault-Tolerant Control of Large-Scale Control Systems

Control Using Networks



Model of Large-Scale Control System

The interaction-oriented model of a large-scale linear control system consisting of N stable subsystems ($\Sigma_k,\ k = 1 \dots N$)

$$\Sigma_k : \begin{cases} \dot{\mathbf{x}}_k = A_k \mathbf{x}_k + B_k \mathbf{w}_k + E_k \sum_{i=1}^{N} \mathbf{s}_i, \\ \mathbf{y}_k = C_k \mathbf{x}_k, \\ \mathbf{z}_k = C_{kz} \mathbf{x}_k \end{cases}$$

The coupling between the subsystems is defined by the interconnection matrix $\mathbf{s} = \mathbf{L}\mathbf{z}$.

$$\mathbf{L} = \begin{bmatrix} L_{11} & \dots & L_{1N} \\ \dots & \dots & \dots \\ L_{N1} & \dots & L_{NN} \end{bmatrix}.$$

Generally it is considered that $L_{ii} = O$.

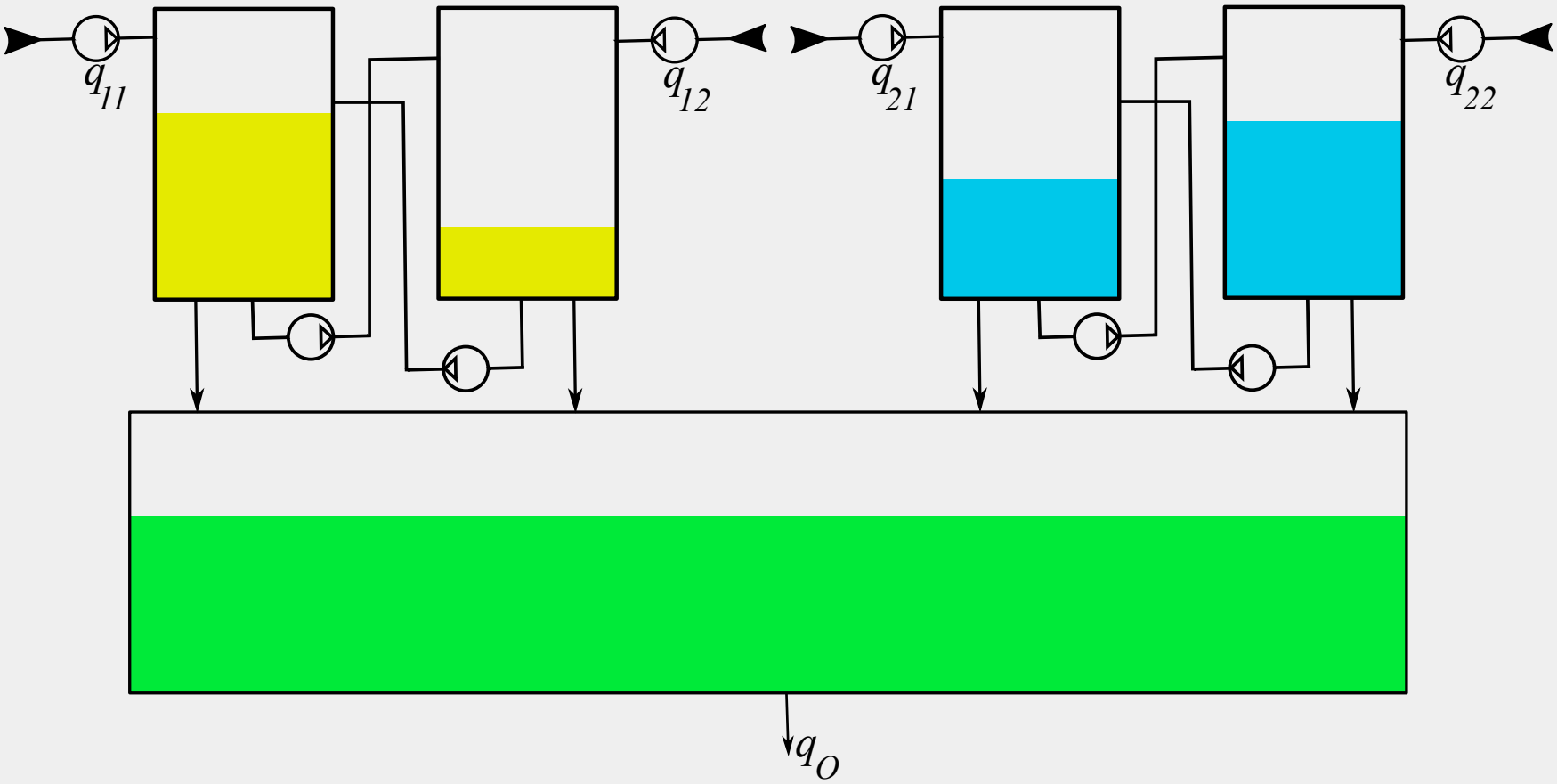
Control Goal for the Fault-Free Large-Scale Control System

The outputs of the subsystems are coupled through a weight matrix $\mathbf{C}$ and produce the performance output ($\mathbf{y}^{\Sigma}$) of the large-scale system:

$$\mathbf{y}^{\Sigma} = \sum_{k=1}^{N} C_k^{\Sigma} \mathbf{y}_k,$$

or equivalently $\mathbf{y}^{\Sigma} = \mathbf{C}\mathbf{y}$ where $\mathbf{C} = [C_1^{\Sigma} \ \dots \ C_N^{\Sigma}]$.
The control goal for the fault-free control System: $\mathbf{y}_{\infty}^{\Sigma} = \mathbf{y}_{d\infty}^{\Sigma}$ (in steady state).
Assume that the fault-free system satisfies the control goal.

Illustrative Example



The control goal: $c_1(q_{11} + q_{12}) + c_2(q_{21} + q_{22}) = c_d q_O$ and/or $q_{11} + q_{11} + q_{21} + q_{21} = K_h$.

Input-Output Model of Large-Scale Control System

In s-domain the model of a subsystem has the form:

$$\Sigma_k : \begin{cases} \mathbf{y}_k(s) = G_k(s)\mathbf{w}_k(s) + G_{ks}(s)\mathbf{s}_k(s), \\ \mathbf{z}_k(s) = G_{kz}(s)\mathbf{w}_k(s) + G_{kzs}(s)\mathbf{s}_k(s) \end{cases}$$

where $\mathbf{s}_k(s) = \sum_{i=1}^{N} \mathbf{s}_i(s) = \sum_{i=1}^{N} L_{ki}\mathbf{z}_i(s)$,
$G_k(s) = C_k(sI - A_k)^{-1}B_k$, $G_{ks}(s) = C_k(sI - A_k)^{-1}E_k$,
$G_{kz}(s) = C_{kz}(sI - A_k)^{-1}B_k$, $G_{kzs}(s) = C_{kz}(sI - A_k)^{-1}E_k$.
Assumptions:

- $G_k(s)$, $G_{ks}(s)$, $G_{kz}(s)$, $G_{kzs}(s)$ are finite gain stable $\forall k$
- $G_k(0)$, $G_{ks}(0)$, $G_{kz}(0)$, $G_{kzs}(0)$ are known $\forall k$.
- $\mathrm{rank}(G_k(0)) = \dim(\mathbf{y}_k) \leq \dim(\mathbf{w}_k)$ $\forall k$.
- $\exists$ C_{kyz} such that $C_{kz} = C_{kzy}C_k$ $\forall k$

Unstructured Control System Model

The output of the unstructured model of the large-scale system in s-domain has the form

$$\mathbf{y}(s) = \mathbf{G}(s)\mathbf{w}(s) + \boldsymbol{\sigma}(s),$$
$$\text{where } \boldsymbol{\sigma}(s) = \mathbf{G}_s(s)\mathbf{s}(s)$$

and $\mathbf{G}(s) = \mathrm{diag}(G_k(s))$, $\mathbf{G}_s(s) = \mathrm{diag}(G_{ks}(s))$ and $\boldsymbol{\sigma}(s)$ is the bias induced by the phyisical couplings.
In general $\boldsymbol{\sigma}(s)$ is considered unknown but its steady state value can be computed in fault-free case as

$$\boldsymbol{\sigma}_\infty = \mathbf{y}_\infty - \mathbf{G}(0)\mathbf{w}.$$

Fault Modeling

A fault in the kth subsystem is modeled as a change in the parameter matrices B_k or C_k of the subsystem's model. In the input-output model of the subsystem the fault is described using multiplicative uncertainties:

$$\Sigma_{kf}: \begin{cases} \mathbf{y}_{kf}(s) = G_f(s)\mathbf{w}_k(s) + G_{fs}(s)\mathbf{s}_{kf}(s), \\ \mathbf{z}_{kf}(s) = G_{fz}(s)\mathbf{w}_k(s) + G_{fzs}(s)\mathbf{s}_{kf}(s) \end{cases}$$

where $G_f(s) = G_k(s)(I + \Delta G_f(s))$, $G_{fs}(s) = G_{ks}(s)(I + \Delta G_{fs}(s))$, $G_{fz}(s) = G_{kz}(s)(I + \Delta G_{fz}(s))$, $G_{fzs}(s) = G_{kzs}(s)(I + \Delta G_{fzs}(s))$. The faulty interconnection input $\mathbf{s}_{kf}(s)$ in considered in the form:

$$\mathbf{s}_{kf}(s) = \mathbf{s}_k(s) + \delta\mathbf{s}_k(s).$$

Model for Fault-Tolerant Control

It is assumed that the fault does not destabilize the large-scale system. To facilitate the fault estimation, the first equation in the faulty large-scale system model is rewritten in the form:

$$\mathbf{y}_{kf}(s) = G_k(s)\mathbf{w}_k(s) + G_{ks}(s)\mathbf{s}_k(s) + \Delta_{kf}(s)\mathbf{f}_k(s)$$
$$\text{where } \Delta_{kf}(s)\mathbf{f}_k(s) = G_k(s)\Delta G_k(s)\mathbf{w}_k(s) + G_{ks}(s)(I + \Delta G_{fs}(s))\delta\mathbf{s}_k(s)$$
$$+ G_{ks}(s)\Delta G_{fs}(s)\mathbf{s}_{kf}(s).$$

Here $\Delta_{kf}(s)$ is a stable unknown diagonal transfer matrix with the property $\Delta_{kf}(0) = I$, $\mathbf{f}_k(s)$ is a bounded, unknown fault input vector with step signal-like elements.

Faulty Large-Scale System

The entire fault-free (healthy) part of the large-scale control system is considered to be described by the model:

$$\Sigma_h : \begin{cases} \mathbf{y}_h(s) = \mathbf{G}_h(s)\mathbf{w}_h(s) + \mathbf{G}_{hs}(s)\mathbf{s}_h(s), \\ \mathbf{z}_h(s) = \mathbf{G}_{hz}(s)\mathbf{w}_h(s) + \mathbf{G}_{hzs}(s)\mathbf{s}_h(s). \end{cases}$$

The interconnection input $\mathbf{s}_h(s)$ also contains a fault induced drift $\delta\mathbf{s}(s)$ term.
The interconnection between the faulty and healthy system can be obtained from the corresponding parts of the matrix $\mathbf{L}$:

$$\begin{pmatrix} \mathbf{s}_f(s) \\ \mathbf{s}_h(s) \end{pmatrix} = \begin{bmatrix} O & \mathbf{L}_{fh} \\ \mathbf{L}_{hf} & O \end{bmatrix} \begin{pmatrix} \mathbf{z}_f(s) \\ \mathbf{z}_h(s) \end{pmatrix}$$

Fault-Tolerant Control Goal

In the presence of faults the performance output of the system is $\mathbf{y}_f^{\Sigma}$. The reconfiguration problem can be formulated as: find the command inputs ($\mathbf{w}_h$) of the non-faulty subsystems such that in steady state

$$\mathbf{y}_{\infty}^{\Sigma} = \mathbf{y}_{f\infty}^{\Sigma}.$$

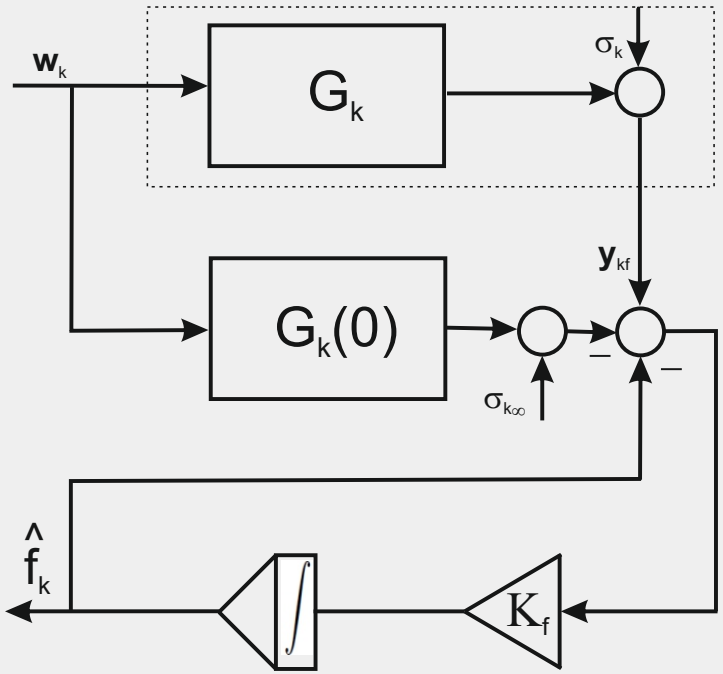
It is considered that the faulty subsystems cannot be reconfigured locally.

Steps of the Fault-Tolerant Control Design

- Estimate the magnitude of the fault in the faulty subsystem.
- Compute a compensator term for each fault-free system in function of the estimated fault by neglecting the physical interconnections.
- Reformulate the input healthy subsystems with an integrator term to suppress the fault propagation through the physical interconnections.

Formulate the fault estimation law as:

$$\widehat{\mathbf{f}}_k(s) = K_f \mathrm{diag}(1/s)\left(\mathbf{y}_{kf}(s) - G_k(0)\mathbf{w}_k(s) - \mathrm{diag}(\boldsymbol{\sigma}_{k\infty})\mathbf{1}(s) - \widehat{\mathbf{f}}_k(s)\right)$$



Lemma

Consider a faulty subsystem modeled as $\mathbf{y}_{kf}(s) = G_k(s)\mathbf{w}_k(s) + G_{ks}(s)\mathbf{s}_k(s) + \Delta_{kf}(s)\mathbf{f}_k(s)$. *The estimation law assures that the estimation error* $\widetilde{f}_k = \mathbf{f}_k - \widehat{\mathbf{f}}_k$ *is always bounded and* $\widehat{\mathbf{f}}_{k\infty} = \mathbf{f}_k$ *if and* K_f *is chosen as* $K_f = \mathrm{diag}(k_i),\ k_i > 0$.

Idea of proof: Apply the assumptions related to the finite gain stability of the fault induced uncertainty terms and the direct computation of $lim_{s\to 0}\mathrm{diag}(s)\widetilde{\mathbf{f}}_k(s)$ which is equal to zero.

Remark

In the case of the fault-free subsystems the estimation algorithm returns the influence of the fault through the physical interconnections on the corresponding subsystem. In these subsystems the result of the estimation can be used to update the $\boldsymbol{\sigma}_{k\infty}$ term in the equations to deal with possible future faults:

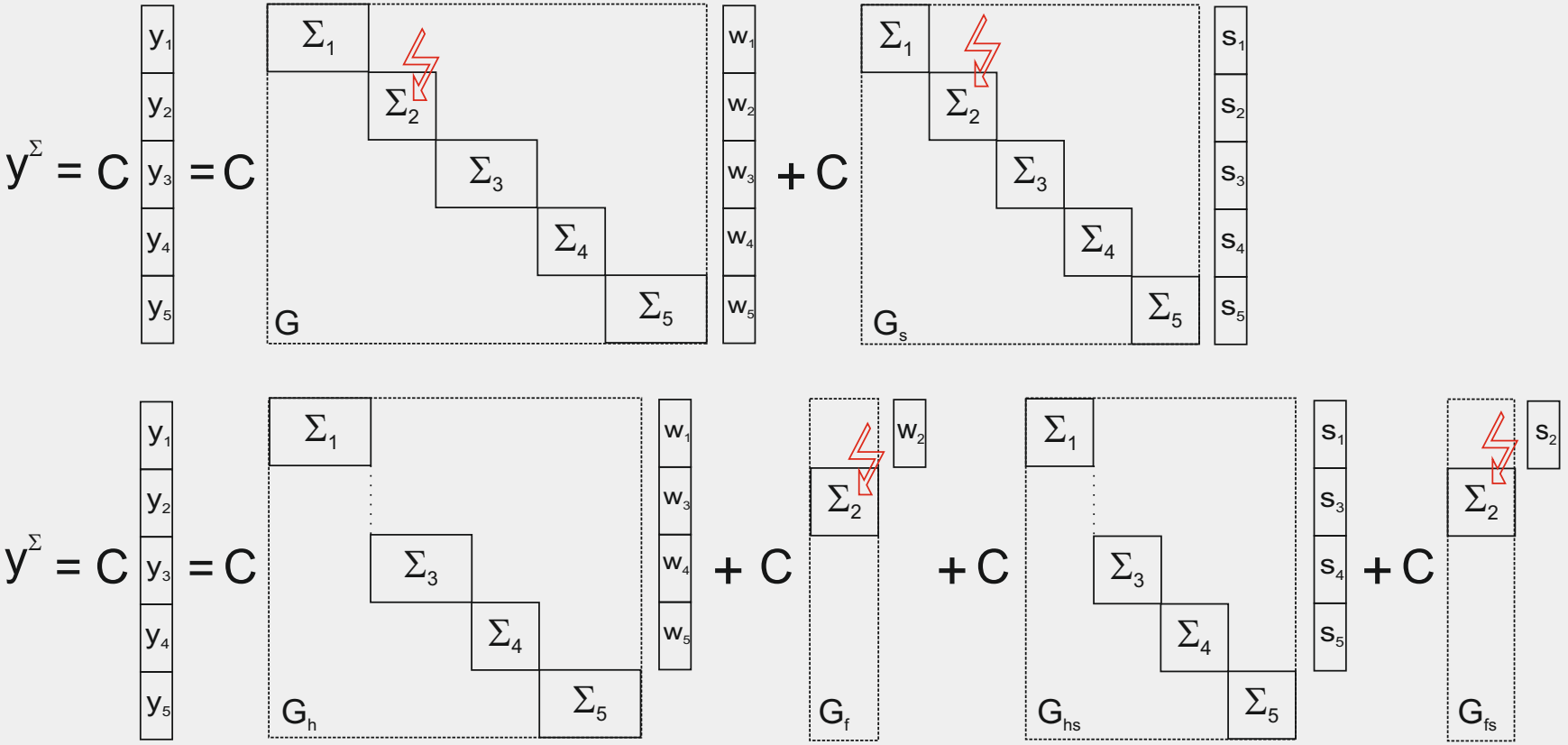
$$\begin{aligned} \delta\widehat{\boldsymbol{\sigma}}_{k\infty} &= \widehat{\mathbf{f}}_{k\infty} \text{ if } \Sigma_k \text{ fault} - \text{free}, \\ \boldsymbol{\sigma}_{k\infty} &=: \boldsymbol{\sigma}_{k\infty} + \delta\widehat{\boldsymbol{\sigma}}_{k\infty}. \end{aligned}$$

Fault Compensation

To compensate the drifts in the performance output induced by the fault, the command inputs of the fault-free systems are extended by adding a fault compensator term to the original input.
In the case of the kth fault-free subsystem the augmented command has the form:

$$\mathbf{w}_{kh}^{C} = \mathbf{w}_{kh} + \delta\mathbf{w}_{k}$$

Split the transfer matrix model of the large-scale system into faulty and fault-free (healthy) parts:



The performance output of the large-scale system in the presence of fault:

$$\mathbf{y}_f^{\Sigma}(s) \quad = \quad \mathbf{y}^{\Sigma}(s) + \mathbf{C}\Big(\mathbf{G}_h(s)\delta\mathbf{w}(s) + \ {}_f(s)\mathbf{f}(s) + \mathbf{G}_{hs}(s)\delta\mathbf{s}(s)\Big).$$

The faulty terms are formulated as e.g. $\mathbf{f}(s) = (\mathbf{0}^T \ldots \mathbf{f}_k(s)^T \ldots \mathbf{0}^T)^T$.
In order to satisfy the fault tolerant control goal ($\mathbf{y}_{\infty}^{\Sigma} = \mathbf{y}_{f\infty}^{\Sigma}$) the following equation has to be solved:

$$-\mathbf{C}\mathbf{G}_h(0)\delta\mathbf{w} = \mathbf{C}\mathbf{f}_{\infty} + \mathbf{C}\mathbf{G}_{hs}(0)\delta\mathbf{s}_{\infty}.$$

The second term on the right hand side represents the effect of the fault propagation through the physical interconnections.

Fault Compensation

First neglect the effect of the fault propagation through the physical interconnections. In this case $\delta\mathbf{w}$ is a solution of a system of linear equations and the fault reconfiguration problem is solvable if

$$\operatorname{rank}[\mathbf{C}\mathbf{G}_h(0)] = \operatorname{rank}[\mathbf{C}\mathbf{G}_h(0)\ \mathbf{C}\mathbf{f}_\infty].$$

If the relation above is satisfied, the fault compensator term is computable as

$$\delta\mathbf{w} = -(\mathbf{C}\mathbf{G}_h(0))^\dagger \mathbf{C}\mathbf{f}_\infty.$$

The compensator signal $\delta\mathbf{w}_k$ for a subsystem is the part of $\delta\mathbf{w}$ corresponding to subsystem k.

Formulate the desired output for each fault-free subsystem as:

$$\begin{aligned} \mathbf{w}_{kh}^{C} &= \mathbf{w}_{kh} + \delta \mathbf{w}_{k} \\ \mathbf{y}_{k}^{D} &= [G_h(0)\mathbf{w}_{h}^{C} + \boldsymbol{\sigma}_{h\infty}]_k \end{aligned}$$

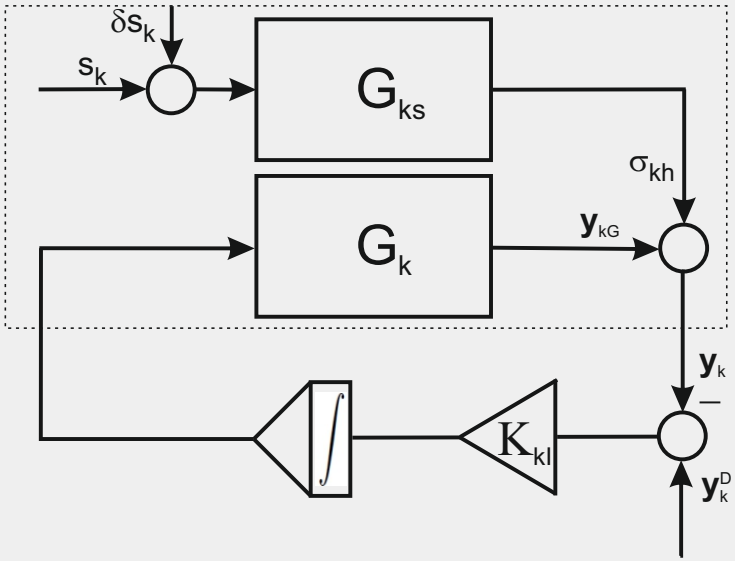
where $[\cdot]_k$ denotes the elements of the vector corresponding to the kth subsystem. $\boldsymbol{\sigma}_{h\infty}$ corresponds to the interconnections before the fault event. Note that $\mathbf{y}_{k}^{D}$ represents the steady state output of the kth fault-free subsystem that can compensate the deviation in the faulty subsystem output, but it assumes that the steady state physical connections correspond to the situation before the fault event.

Decentralized Integral Control

If $\mathbf{y}_k^D = \mathbf{y}_k$ for each fault-free subsystem, the fault tolerant control goal would be satisfied. Formulate the command input of the fault-free system as:

$$\boldsymbol{\omega}_{kh}(s) = K_{kI}\mathrm{diag}(1/s)(\mathbf{y}_k^D(s) - \mathbf{y}_k(s))$$

where K_{kI} is the integral gain matrix.



Lemma

Let a fault-free subsystem with the control
$\omega_{kh}(s) = K_{kl}\mathrm{diag}(\frac{1}{s})(\mathbf{y}_k^D(s) - \mathbf{y}_k(s))$
in which

$$K_{kl} = G_k(0)^{\dagger} diag(\kappa_i),\ \kappa_i > 0\ \forall i$$

If the reconfigured large-scale system is stable, then $\mathbf{y}_{k\infty} = \mathbf{y}_k^D$ *for a constant* $\mathbf{y}_k^D$.

Sketch of the proof:
By the assumption $\mathrm{rank}(G_k(0)) = \dim(\mathbf{y}_k) \leq \dim(\mathbf{w}_k)$ it yields $G_k(0)G_k(0)^\dagger diag(\kappa_i) = diag(\kappa_i)$.
Denote the output of $G_k(s)$ by $\mathbf{y}_{kG}(s)$, i.e. $\mathbf{y}_k(s) = \mathbf{y}_{kG}(s) + \boldsymbol{\sigma}_{kh}(s)$. By applying the proposed control, its value reads as:

$$\mathbf{y}_{kG}(s) = (sI + G_k(s)K_{kI})^{-1} G_k(s)K_{kI}(\mathbf{y}_k^D(s) - \boldsymbol{\sigma}_{kh}(s)).$$

In steady state
$\lim_{s\to\infty}\big(\mathrm{diag}(s)\mathbf{y}_{kG}(s)\big) = \mathbf{y}_{kG\infty} = \mathbf{y}_k^D - \boldsymbol{\sigma}_{kh\infty}$, i.e. $\mathbf{y}_{k\infty} = \mathbf{y}_k^D$.

Reconfigured Large-Scale System - Faulty Subsystem

During modeling the fault estimation algorithm in the reconfiguration control can be viewed as a new interconnection signal: the estimated fault depends on the output of the faulty subsystem and it serves as an input for the fault-free subsystems.
The model of the augmented faulty subsystem yields in the form:

$$\mathbf{y}_{kf}(s) = \begin{bmatrix} G_k^f(s) & 0 \end{bmatrix} \begin{bmatrix} \mathbf{w}_k(s) \\ \mathbf{y}_0^f(s) \end{bmatrix} + G_{ks}^f(s)\mathbf{s}_{kf}(s),$$

$$\begin{bmatrix} \mathbf{z}_{kf}(s) \\ \widehat{\mathbf{f}}_k(s) \end{bmatrix} = \begin{bmatrix} G_k^f(s) & 0 \\ G_{ke}(s)G_k^f(s) & G_{ke}(s) \end{bmatrix} \begin{bmatrix} \mathbf{w}_{kf}(s) \\ \mathbf{y}_0^f(s) \end{bmatrix} + \begin{bmatrix} G_{kzs}^f(s) \\ G_{ke}(s)G_{ks}^f(s) \end{bmatrix} \mathbf{s}_{kf}(s)$$

where $G_{ke}(s) = (sI + K_{fk})^{-1} K_{fk}$.

Reconfigured Large-Scale System - Fault-Free Part

The model of the reconfigured fault-free part:

$$\Sigma_h^r : \begin{cases} \mathbf{y}_h(s) = \mathbf{G}_h^r(s)\mathbf{w}_h(s) + \mathbf{G}_{hf}^r(s)\widehat{\mathbf{f}}(s) + \mathbf{G}_{hs}^r(s)\mathbf{s}_h(s), \\ \mathbf{z}_h(s) = \mathbf{G}_{hz}^r(s)\mathbf{w}_h(s) + \mathbf{G}_{hzf}^r(s)\widehat{\mathbf{f}}(s) + \mathbf{G}_{hzs}^r(s)\mathbf{s}_h(s) \end{cases}$$

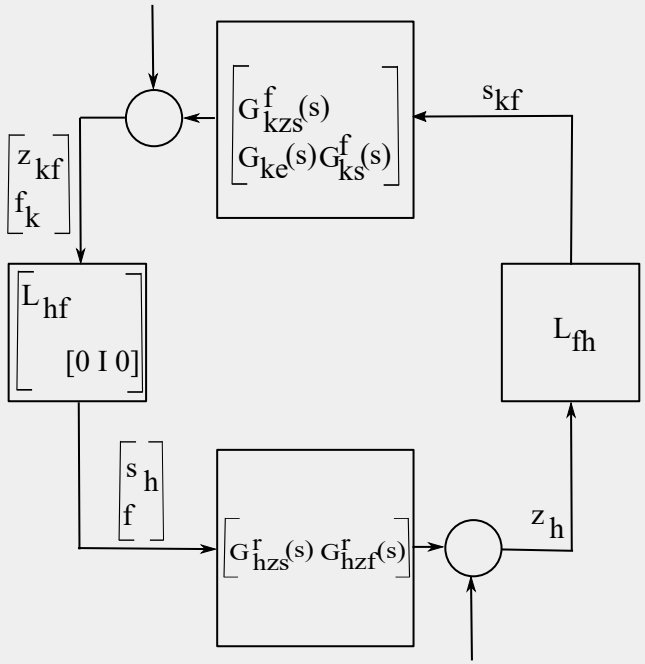
where e.g.

$$\mathbf{G}_{hf}^r(s) \quad = \quad (\mathrm{diag}(s) + \mathbf{G}_h(s)\mathrm{diag}(K_{kl}))^{-1}\mathbf{G}_h(s)\mathrm{diag}(K_{kl})(\mathbf{C}\mathbf{G}_h(0))^{\dagger}\mathbf{C}$$

The interconnection matrix between the faulty subsystem and the fault-free part modifies as:

$$\begin{pmatrix} \mathbf{s}_f(s) \\ \mathbf{s}_h(s) \\ \widehat{\mathbf{f}}(s) \end{pmatrix} = \begin{bmatrix} O & O & \mathbf{L}_{fh} \\ \mathbf{L}_{hf} & O & O \\ 0 & [O \dots I \dots O]^T & O \end{bmatrix} \begin{pmatrix} \mathbf{z}_f(s) \\ \mathbf{z}_h(s) \\ \widehat{\mathbf{f}}_k(s) \end{pmatrix}.$$

Reconfigured Large-Scale System - Stability



The reconfigured large-scale system is stable if all the transfer matrices in the models of the faulty system + fault estimator and reconfigured local controllers are stable and

$$\left\| L_{fh} \left[\mathbf{G}^r_{hzf}(j\omega)\ \mathbf{G}^r_{hzs}(j\omega) \right] \begin{bmatrix} \mathbf{L}_{hf} & O \\ 0 & [O \dots I \dots O]^T \end{bmatrix} \begin{bmatrix} G^f_{kzs}(j\omega) \\ G_{ke}(j\omega) G^f_{ks}(j\omega) \end{bmatrix} \right\| \leq 1$$

Summary of the General Case and Remarks

Fault estimation in the faulty subsystems:

$$\widehat{\mathbf{f}}_k(s) = K_f \mathrm{diag}(1/s)\left(\mathbf{y}_{kf}(s) - G_k(0)\mathbf{w}_k(s) - \mathrm{diag}(\boldsymbol{\sigma}_{k\infty})\mathbf{1}(s) - \widehat{\mathbf{f}}_k(s)\right)$$

Reconfiguration in the fault-free subsystems:

$$\begin{aligned} \mathbf{w}_{kh}^C &= \mathbf{w}_{kh} - [(\mathbf{C}\mathbf{G}_h(0))^\dagger \mathbf{C}\widehat{\mathbf{f}}_k]_k \\ \mathbf{y}_k^D &= [G_h(0)\mathbf{w}_h^C + \boldsymbol{\sigma}_{h\infty}]_k \\ \boldsymbol{\omega}_{kh}(s) &= K_{kl}\mathrm{diag}(1/s)(\mathbf{y}_k^D(s) - \mathbf{y}_k(s)) \end{aligned}$$

Remarks:

- The proposed control approach assumes a continuous communication among the faulty and fault-free subsystems.
- The small gain theorem based stability analysis in general is not practical.

Reconfiguration with Reduced Communication Costs

The case is considered when the fault-free subsystems do not receive the estimated fault value continuously. $\widehat{\mathbf{f}}$ is broadcasted only once in the time instant t_r.

Theorem

Consider a faulty large-scale system with the introduced assumptions. With the decentralized integral control law the reconfigured large-scale system is stable if the compensation signal $\delta\mathbf{w}$ is formulated in function of a $\widehat{\mathbf{f}}$ and

$$\mathrm{Re}[\lambda_i[G_h(0)\mathrm{diag}(K_{kl})]] > 0 \ \forall i. \tag{1}$$

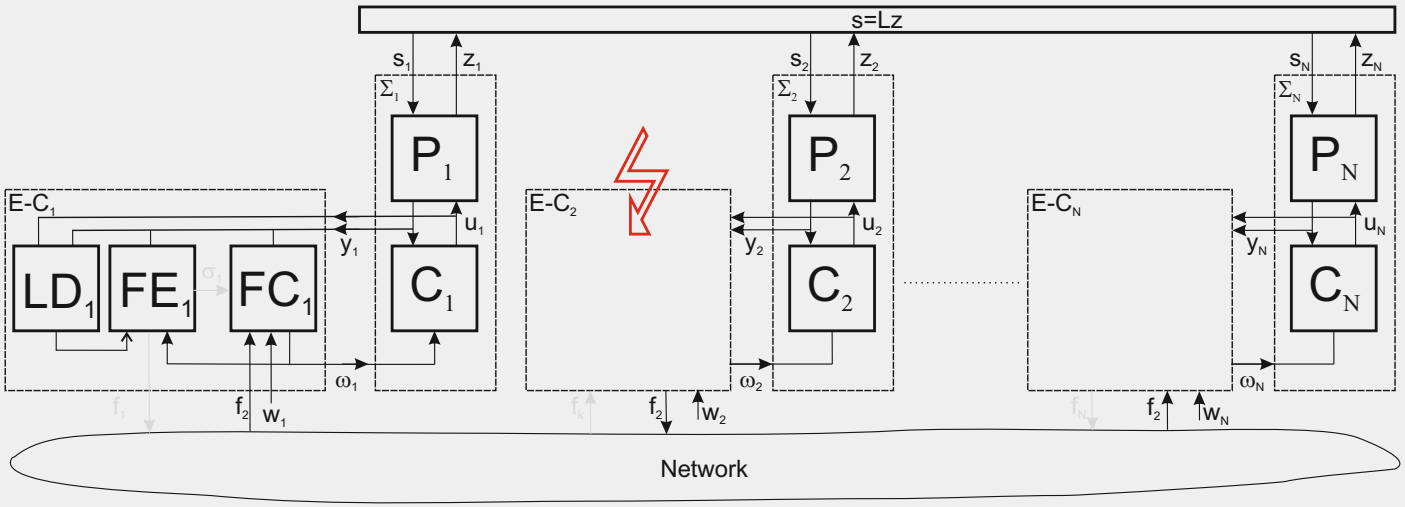
If in the equation $\delta\mathbf{w}$ is formed by using the output of the estimation law in the time instant $t_r < \infty$ $(\widehat{\mathbf{f}}(t_r))$, then the control goal $\mathbf{y}^{\Sigma}_{\infty} = \mathbf{y}^{\Sigma}_{f\infty}$ is satisfied with an accuracy $\|\mathbf{C}\|\|\mathbf{f} - \widehat{\mathbf{f}}(t_r)\|$.

Reconfiguration with Reduced Communication Costs

Sketch of the proof:
In this case there is no continuous communication between the faulty and fault-free subsystems, the interconnection loop does not change due to the control.
The condition $\mathrm{Re}[\lambda_i[G_h(0)\mathrm{diag}(K_{kl})]] > 0 \ \forall i$ is a stability condition for MIMO systems with integral control.
The accuracy yields from the previously introduced lemmas and from direct computation:

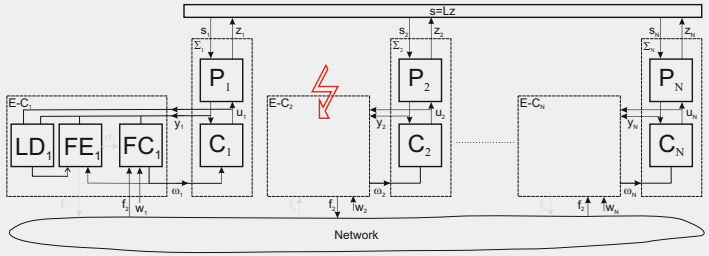
$$\mathbf{y}_f^{\Sigma} = \mathbf{y}^{\Sigma} - \mathbf{C}\mathbf{G}_h(0)(\mathbf{C}\mathbf{G}_h(0))^{\dagger}\mathbf{C}\widehat{\mathbf{f}}(t_r)) - \mathbf{C}\mathbf{f}_{\infty},$$
$$\|\mathbf{y}_f^{\Sigma} - \mathbf{y}^{\Sigma}\| \leq \|\mathbf{C}\|\|\widehat{\mathbf{f}}(t_r) - \mathbf{f}_{\infty}\|.$$

Reconfiguration with Reduced Communication Costs - Implementation



- FE_k - fault estimation block
- FC_k - fault compensation block
- LD_k - local fault detector - Necessary, because generally the subsystems cannot decide whether the fault emerged in the local or the effect of a fault event, which happen in another subsystem, reached the subsystem through the physical interconnections.

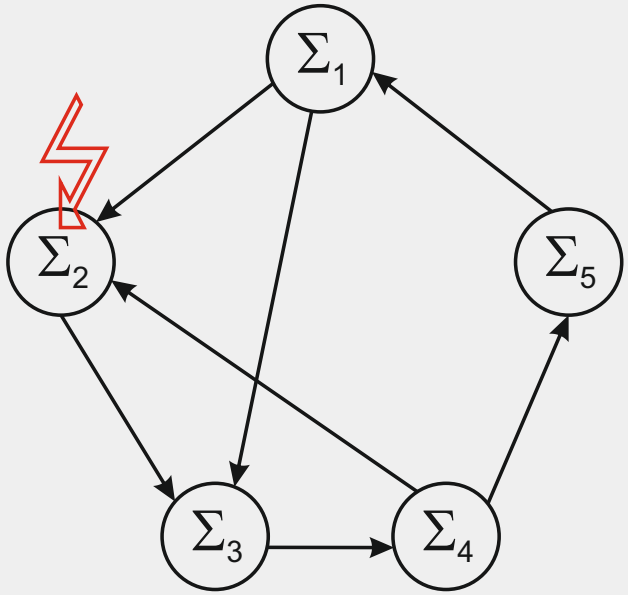
Reconfiguration with Reduced Communication Costs - Implementation



Procedure for fault estimation and reconfiguration:

- LD_k detects the actuator fault; FE_k estimates the effect of the fault on Σ_k.
- FE_k broadcasts through the network to fault-free subsystems the value $\widehat{\mathbf{f}}_k$.
- The fault compensator blocks of the fault-free subsystems (FC_i) compute and apply the new command.
- FE_i of the fault-free systems re-estimate the effect of the fault and the compensation on Σ_i.

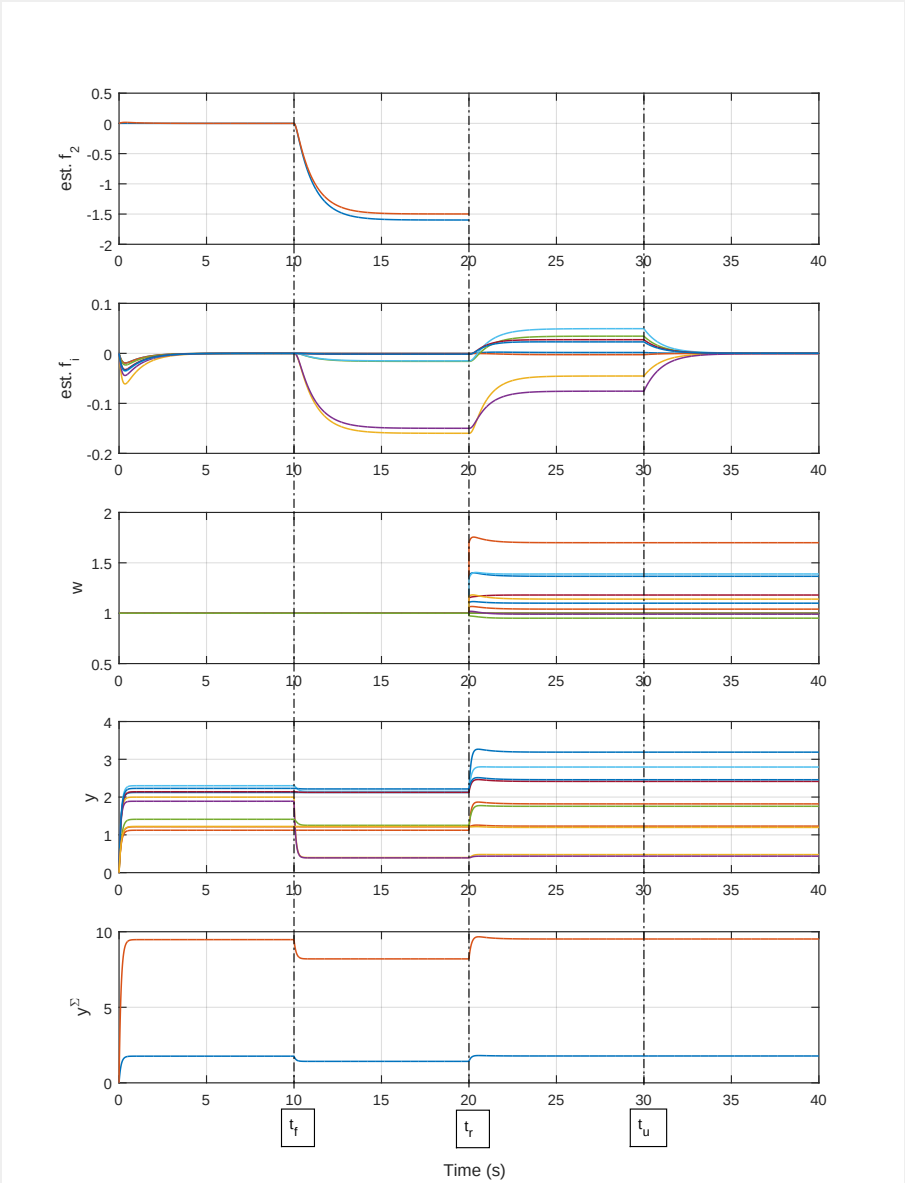
Simulation Measurements



Simulation conditions:

- $\dim(\mathbf{y}_k) = 2$, $\dim(\mathbf{w}_k) = \{2, 3\}$, $\dim(\mathbf{z}_k) = 2$, $k = 1 \ldots 5$.
- A typical element of the transfer matrices: $g_{ij}(s) = g_{ij}(0)/(0.1s + 1)$
- Performance outputs: $y_1^\Sigma = \sum_{k=1}^{5} \sum_{i=1}^{2} y_{ki}/10$;
  $y_2^\Sigma = \sum_{k=1}^{5} \sum_{i=1}^{2} k y_{ki}/10$.

Simulation Measurements



Bibliography

- Thomas Steffen, *Control Reconfiguration of Dynamical Systems*, Springer, 2005.
- Mogens Blanke, Michel Kinnaert, Jan Lunze, Marcel Staroswiecki, *Diagnosis and Fault-Tolerant Control*, Springer, 2016.
- L. Márton, Kai Schenk, Jan Lunze, *Fault Estimation and Networked Reconfiguration in Large-Scale Control Systems*, IFAC World Congress, Toulouse, France, 2017, pp. 10342-10349.