UNIVERSITAT POLITÈCNICA DE CATALUNYA

DIAGNOSIS AND FAULT-TOLERANT CONTROL USING SET-BASED METHODS

Vicenç Puig

Advanced Control Systems (SAC)

Research Group

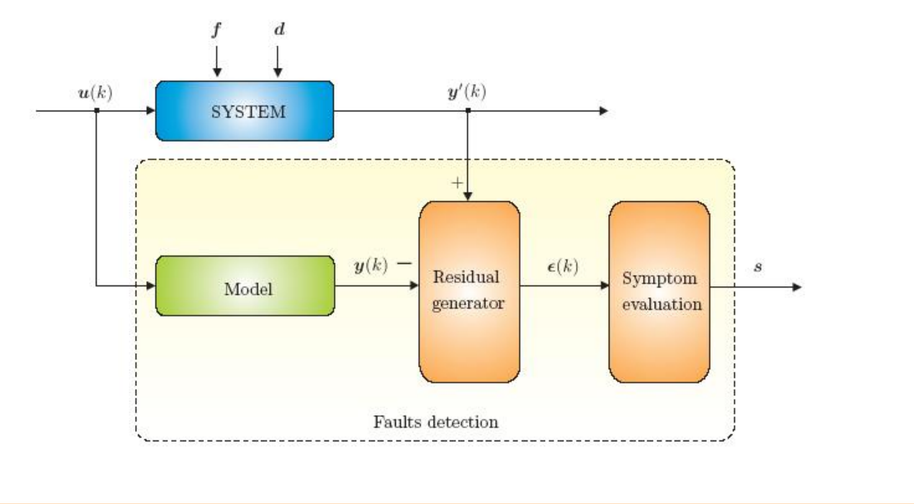
Réunion GT S3 - February 4th, 2016

Index

1. Introduction
2. Interval Models for Fault Detection
3. Fault Detection using the Interval Observer Approach
4. Fault Detection using the Set-membership Approach
5. Identification for Robust Fault Detection
6. Fault-tolerance Evaluation
7. Real Applications
8. Conclusions
9. Further Research

Model-based Fault Detection

- Model-based fault detection methods rely on the concept of **analytical redundancy**.



- However, modeling errors and disturbances in complex engineering systems are inevitable, and hence there is a need to develop robust fault detection algorithms.

Robustness in Model-based Fault Detection

- The ***robustness*** of a fault detection system means that it must be only sensitive to faults, even in the presence of model-reality differences.

DETECTION PHASE

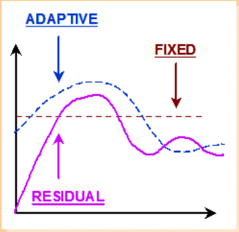
MEASURES → RESIDUAL GENERATION → DECISION MAKING → FAULT INDICATION

ACTIVE ROBUSTNESS

PASSIVE ROBUSTNESS

ROBUST OBSERVERS

ROBUST PARITY EQUATIONS



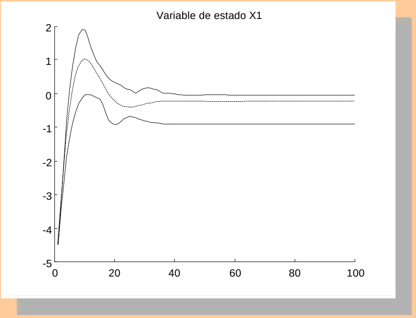
ADAPTIVE THRESHOLD

FUZZY LOGIC

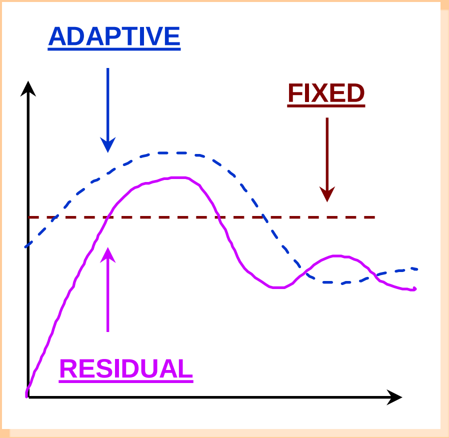
Passive Robust Decision-Making using Interval Models

RESIDUAL → DECISION MAKING → FAULT INDICATION

$$y_k \notin \hat{\Upsilon}_k = \left[\underline{\hat{y}}_k , \overline{\hat{y}}_k \right]$$



THRESHOLD



ADAPTIVE

STRUCTURED UNCERTAINTY → **INTERVAL MODELS AND METHODS**

(ENVELOPES)

UNSTRUCTURED UNCERTAINTY

Index

1. Introduction
2. Interval Models for Fault Detection
3. Fault Detection using the Interval Observer Approach
4. Fault Detection using the Set-membership Approach
5. Identification for Robust Fault Detection
6. Fault-tolerance Evaluation
7. Real Applications
8. Conclusions
9. Further Research

Interval Model for FDI (1)

- Consider that the system to be monitored can be described by a general nonlinear model in discrete-time

$$\begin{aligned} x(k+1) &= f(x(k), u(k), \theta) \\ y(k) &= g(x(k), u(k), \theta) \end{aligned}$$

- The parameters θ2R^m are assumed to be unknown but belong to known intervals

$$\theta_i \in [\underline{\theta}_i, \overline{\theta}_i], \qquad i = 1 \ldots m$$

- An additional equation defining the allowed variance of parameters can be introduced for this purpose:

$$\theta(k+1) = \theta(k) + w(k)$$

where |*w(k)*|·λ.

Interval Model for FDI (2)

- Measurement noise can be taken into account by assuming that the measurements are known to belong to intervals [*y(k)*], often created by adding an noise term σ to the actual measurement *y(k)*, that is,

$$[y(k)] = [y(k) - \sigma, y(k) + \sigma]$$

- In case uncertain parameters appear linearly with respect to inputs/outputs, the system model will be expressed in regressor form

$$y(k) = \varphi^T(k)\theta(k) + e(k)$$

- This corresponds to a MA parity equation.

Fault Detection using Direct Image Test

- Considering the uncertainty in parameters $\boldsymbol{\theta} \in \boldsymbol{\Theta}$, the ***direct image test*** is

$$y(k) \in \left[\underline{\hat{y}}(k), \overline{\hat{y}}(k)\right]$$

Then, no fault is indicated. In other case, a fault is indicated.

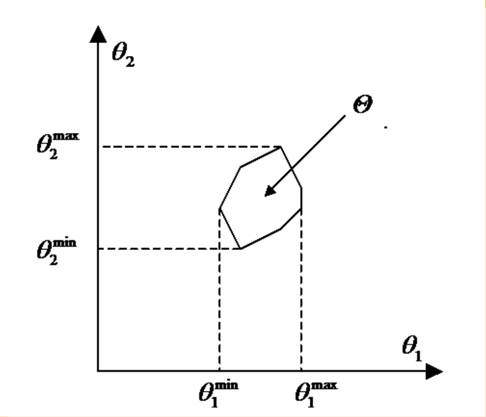
- The interval for the estimated output can be determined by

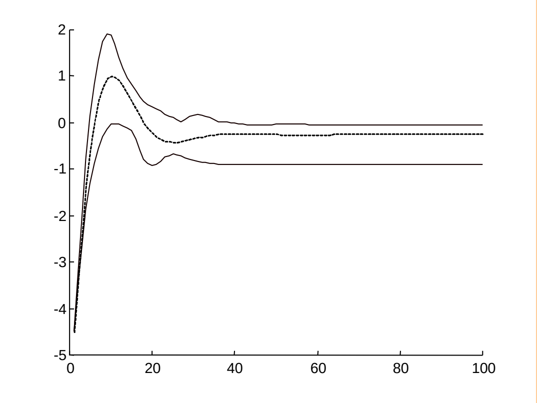
$$\boldsymbol{\varphi}^T(k)\underline{\boldsymbol{\theta}}(k) + \underline{\sigma} \le y(k) \le \boldsymbol{\varphi}^T(k)\overline{\boldsymbol{\theta}}(k) + \overline{\sigma}$$

where:

$$\underline{\boldsymbol{\theta}}(k) = \arg\min_{\boldsymbol{\theta} \in \mathcal{V}} \varphi^T \boldsymbol{\theta}$$

$$\overline{\boldsymbol{\theta}}(k) = \arg\max_{\boldsymbol{\theta} \in \mathcal{V}} \varphi^T \boldsymbol{\theta}$$



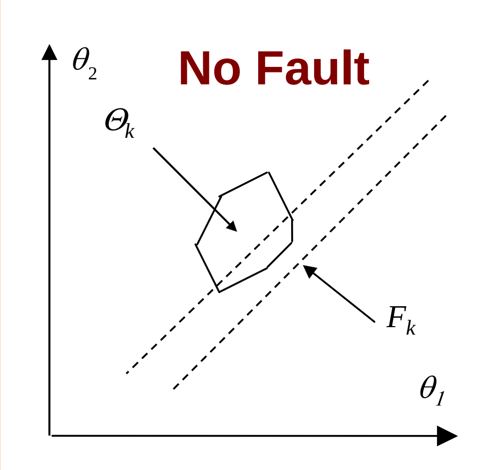


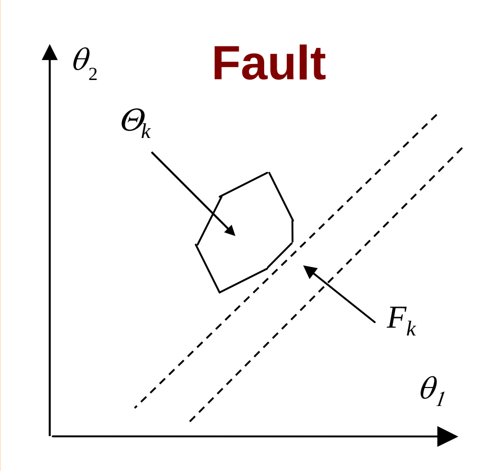
Fault Detection Algorithm using Inverse Test

$$\exists \boldsymbol{\theta} \in \boldsymbol{\Theta} \mid y(k) - \overline{\sigma} \leq \boldsymbol{\varphi}^{T}(k)\boldsymbol{\theta} \leq y(k) - \underline{\sigma}$$

$$F_k = \left\{ \theta \in \mathbb{R}^n : -\sigma \leq y(k) - \varphi(k)^T \theta \leq \sigma \right\}$$

$$F_k \cap \Theta_k \overset{?}{=} \varnothing$$



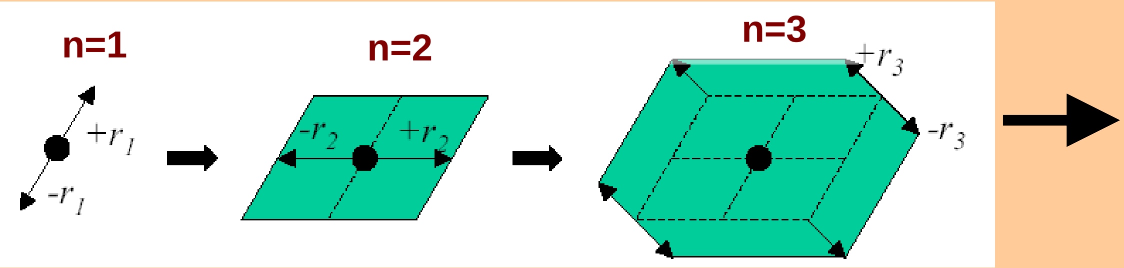


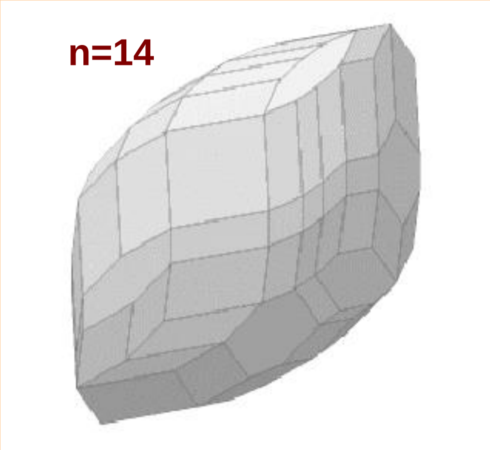
Zonotopes (1)

- A zonotope can be thought of as a ***Minkowski sum*** of a finite set of line segments:

$$\mathcal{X} = \mathbf{p} \oplus \mathbf{R}\mathbf{B}^m = \left\{ \mathbf{p} + \mathbf{R}\mathbf{z} : \mathbf{z} \in \mathbf{B}^m \right\}$$

- A zonotope can also be seen as the linear image of a *m-hypercube* in a *n-space*



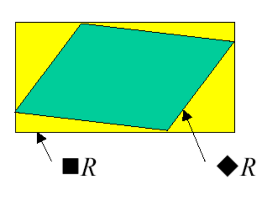


Zonotopes (2)

Zonotope Arithmetic

- Sum of two zonotopes: $\mathcal{X} = \mathbf{p} \oplus \mathbf{R}\mathbf{B}^m = (\mathbf{p}_1 + \mathbf{p}_2) \oplus [\mathbf{R}_1 \quad \mathbf{R}_2]\mathbf{B}^m$
- Image of a zonotope by a linear application ***L*:** $\mathcal{X} = (\mathbf{L}\mathbf{p}) \oplus (\mathbf{L}\mathbf{R})\mathbf{B}^m$
- Smallest interval box containing a zonotope ("interval hull"):

$$\square\mathcal{X} = \left\{ \mathbf{x} : |x_i - p_i| \le \|\mathbf{R}_i\|_1 \right\}$$



- Inverse image of a zonotope by a linear application
- Intersection of two zonotopes

Index

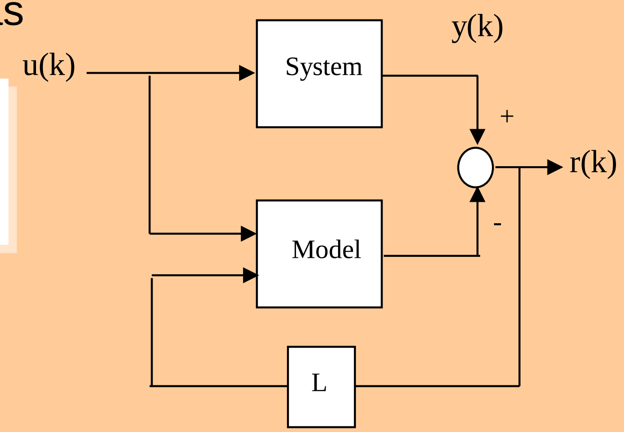
1. Introduction
2. Interval Models for Fault Detection
3. **Fault Detection using the Interval Observer Approach**
4. Fault Detection using the Set-membership Approach
5. Identification for Robust Fault Detection
6. Fault-tolerance Evaluation
7. Real Applications
8. Conclusions
9. Further Research

Interval Observer (1)

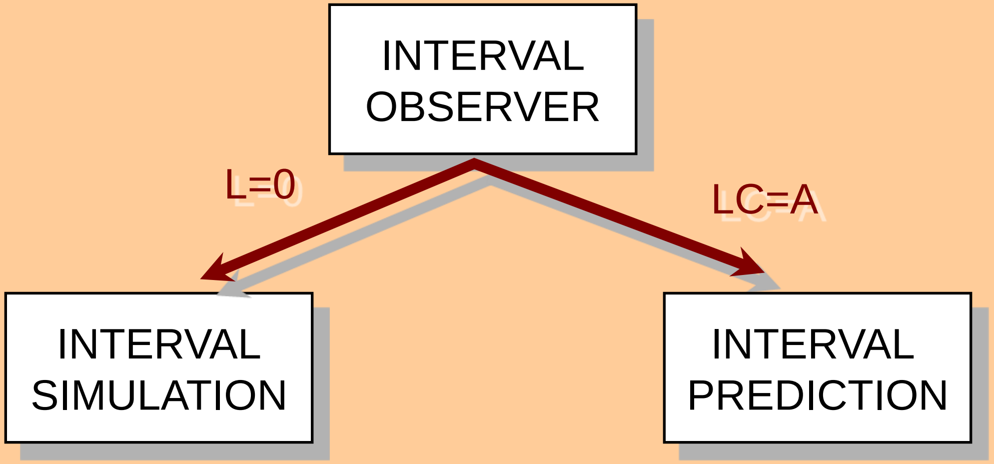
- Let the model for the state estimator of the monitored system described by a ***interval Luenberger observer*** formulated as

$$\hat{\mathbf{x}}_{k+1} = \mathbf{A}(\boldsymbol{\theta})\hat{\mathbf{x}}_k + \mathbf{B}(\boldsymbol{\theta})\mathbf{u}_k + \mathbf{w}_k + \mathbf{L}(\mathbf{y}_k - \hat{\mathbf{y}}_k)$$

$$\hat{\mathbf{y}}_k = \mathbf{C}\mathbf{x}_k + \mathbf{v}_k$$



- This approach is in a half-way between ***simulation*** and ***prediction*** approaches.



Interval Observer (2)

- Let us denote the following sequences from the first time instant to time *k*:

$$\tilde{\boldsymbol{u}}_k = (\boldsymbol{u}_j)_0^{k-1} = (\boldsymbol{u}_0, \boldsymbol{u}_1, \cdots, \boldsymbol{u}_{k-1})$$
$$\tilde{\boldsymbol{y}}_k = (\boldsymbol{y}_j)_0^{k-1} = (\boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_k)$$
$$\tilde{\boldsymbol{w}}_k = (\boldsymbol{w}_j)_0^{k-1} = (\boldsymbol{w}_0, \boldsymbol{w}_1, \cdots, \boldsymbol{w}_{k-1})$$
$$\tilde{\boldsymbol{v}}_k = (\boldsymbol{v}_j)_0^{k-1} = (\boldsymbol{v}_0, \boldsymbol{v}_1, \cdots, \boldsymbol{v}_{k-1})$$
$$\tilde{\boldsymbol{\theta}}_k = (\boldsymbol{\theta}_j)_0^{k-1} = (\boldsymbol{\theta}_0, \boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_{k-1})$$

- The set of estimated states at time *k* using the ***interval observer approach*** is expressed by

$$\hat{\mathcal{X}}_k = \left\{ \begin{array}{l} \hat{\boldsymbol{x}}_k \ \textit{such that} \\ (\hat{\boldsymbol{x}}_{k+1} = \boldsymbol{A}(\boldsymbol{\theta}_k)\hat{\boldsymbol{x}}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{w}_k + \boldsymbol{L}(\boldsymbol{y}(k) - \hat{\boldsymbol{y}}(k)))_{j=1}^{k} \\ (\hat{\boldsymbol{y}}_k = \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{v}_k)_{j=0}^{k} \\ (\boldsymbol{w}_k \in \mathcal{W}, \boldsymbol{v}_k \in \mathcal{V}, \boldsymbol{\theta}_k \in \Theta)_{j=0}^{k}, \boldsymbol{x}_0 \in \mathcal{X} \end{array} \right\}$$

Implementation of Interval Observers

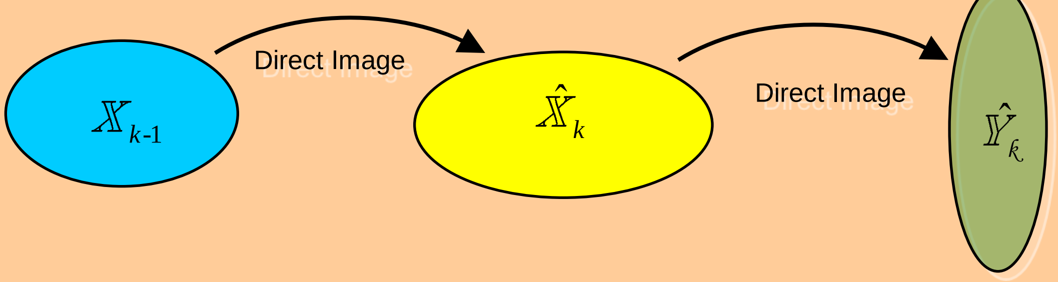
- The previous uncertain state set at time k can be computed approximately by admitting the rupture of the existing relations between variables of consecutive time instants:

Algorithm 1: Worst-case State Observer using Set Computations

---

$\hat{\mathcal{X}}_k \Leftarrow \mathcal{X}_0$
for $k = 1$ to N **do**
    Compute $\hat{\mathcal{X}}_k$
    Compute $\hat{\mathcal{Y}}_k$
end for

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{x}_k + \boldsymbol{B}(\boldsymbol{\theta})\boldsymbol{u}_k + \boldsymbol{w}_k + \boldsymbol{L}(\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k) \qquad \boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{v}_k$$



Problems of Interval Observers

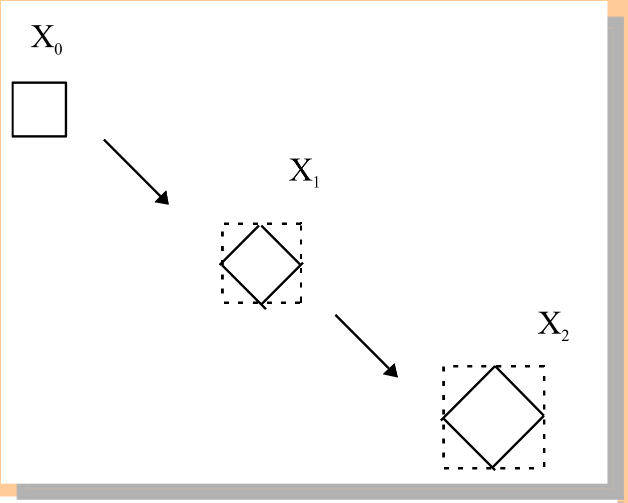
- When approximating the region of system states using sets several problems should be considered:
  - The ***wrapping effect***
  - The preservation of the ***parameter time-invariance***
  - The ***under/over estimation*** of the region
- These problems produce the ***propagation of the uncertainty***, deriving in the production of inconsistent, and even, unstable simulations/observations.

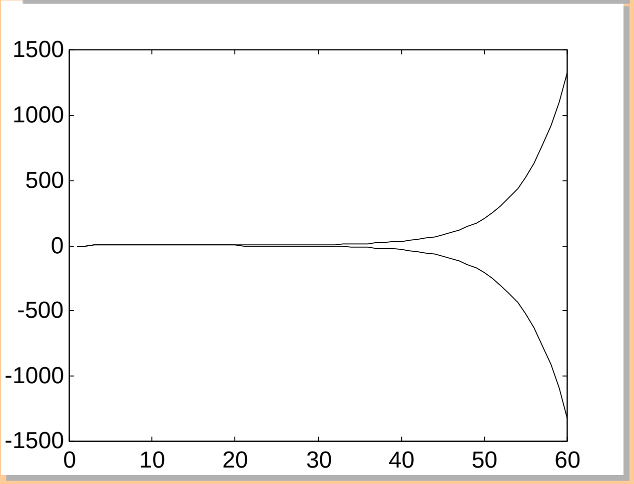
Wrapping Effect

- The ***problem of wrapping*** is related to the use of a crude approximation of the real region of state variables.

- At every stage of the simulation/observation, the ***true region*** of uncertain states ***is wrapped into a superset*** feasible to construct and to represent on a computer.

- Because of the overestimation of the a wrapped set is proportional to its radius, a ***spurious growth of the enclosures*** can result if the composition of wrapping and mapping is iterated.





Designing the Observer Gain to Avoid the Wrapping Effect

- Given a ***non-isotonic interval system***, an interval observer could be designed to fulfil the condition of isotonicity if all the elements of the observer matrix A_0 satisfy: $a_{ii}^{o} \geq 0$.

- In case of an isotonic observer is designed through appropriate selection of the observer gain, the wrapping effect is not present.

- Consequently, a simple iterative scheme based on a region propagation will work, providing the same results than a trajectory propagation algorithm.

- Moreover, a set-based (***time-varying***) interval observation and a trajectory based (***time-invariant***) interval observation will provide the same interval observation

Fault Detection using Interval Observers (1)

- ***Fault detection test:***

  Given the sequences of measured inputs $\tilde{\boldsymbol{u}}_k$ and outputs $\tilde{\boldsymbol{y}}_k$ of the actual system, a ***fault*** is said to have occurred at time *k* if

$$y_k \notin \hat{\Upsilon}_k = \left[\underline{\hat{y}}_k, \overline{\hat{y}}_k\right]$$

or alternatively,

$$0 \notin \left[\underline{r}_k, \overline{r}_k\right] = y(k) - \left[\underline{\hat{y}}_k, \overline{\hat{y}}_k\right]$$

- In case noise in measurements is considered $y_k \in \Upsilon_k = \left[\underline{y}_k, \overline{y}_k\right]$, a fault is detected at time *k* if

$$\Upsilon_k \cap \hat{\Upsilon}_k = \varnothing$$

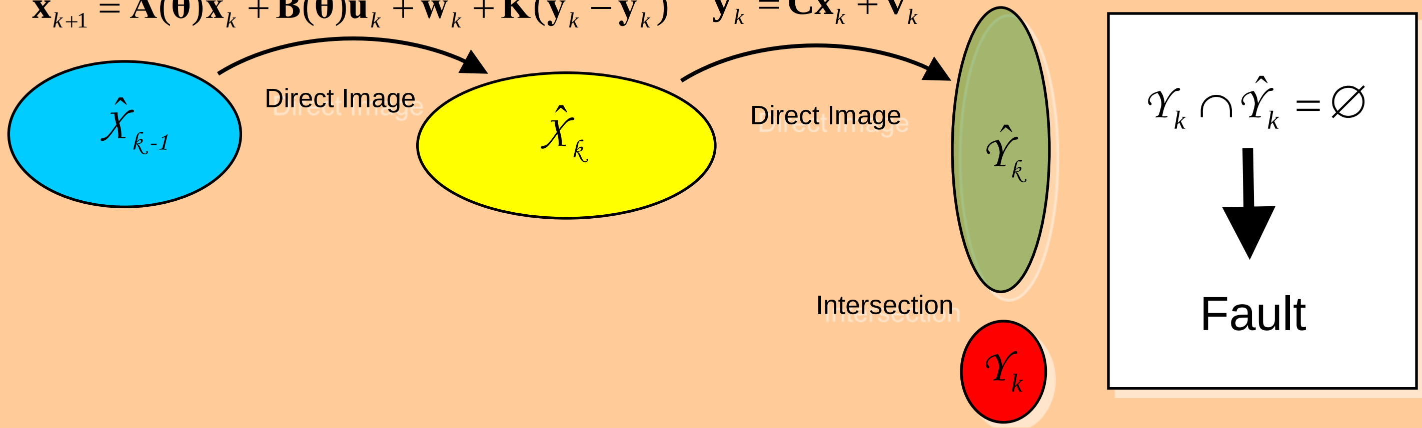
- ***Fault detection*** consists in detecting a fault using the previous test given a sequence of measured inputs $\tilde{\boldsymbol{u}}_k$ and ouputs $\tilde{\boldsymbol{y}}_k$.

Fault Detection using Interval Observers (2)

Algorithm 2: Fault Detection using Worst-case Observer

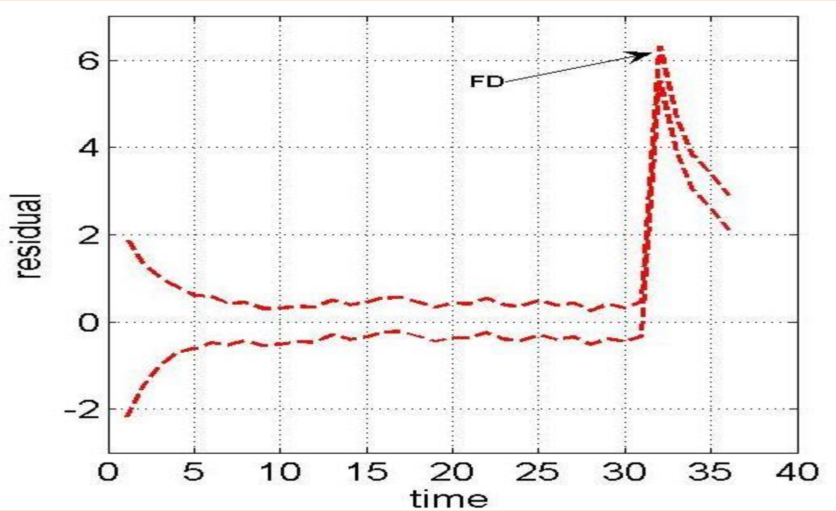
$\hat{\mathcal{X}}_k \Leftarrow \mathcal{X}_0$
for $k = 1$ to N **do**
  Compute $\hat{\mathcal{X}}_k$
  Compute $\hat{\mathcal{Y}}_k$
  if $\hat{\mathcal{Y}}_k \cap \mathcal{Y}_k = \emptyset$ **then**
    Exit (Fault detected)
  end if
end for

$$\mathbf{x}_{k+1} = \mathbf{A}(\boldsymbol{\theta})\mathbf{x}_k + \mathbf{B}(\boldsymbol{\theta})\mathbf{u}_k + \mathbf{w}_k + \mathbf{K}(\mathbf{y}_k - \hat{\mathbf{y}}_k) \qquad \mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{v}_k$$

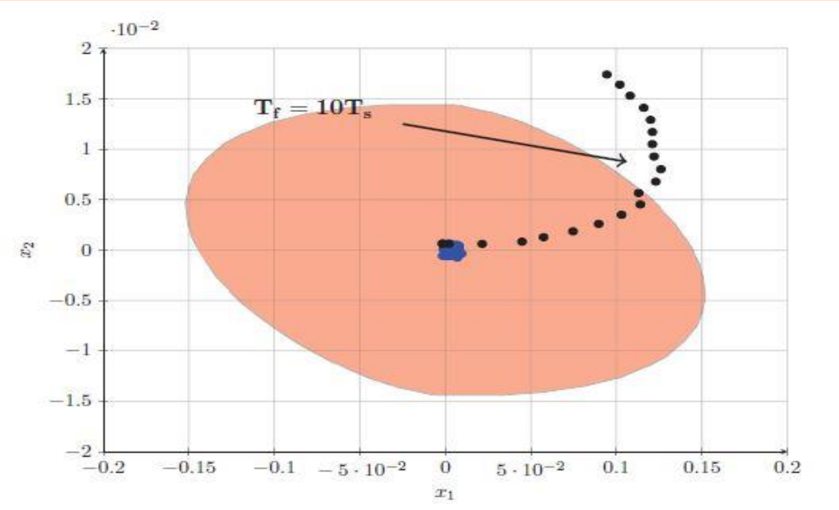


Invariant Sets and Interval Obsevers

Interval observer-based FD principle



Invariant set-based FD principle



Advantages and Disadvantages

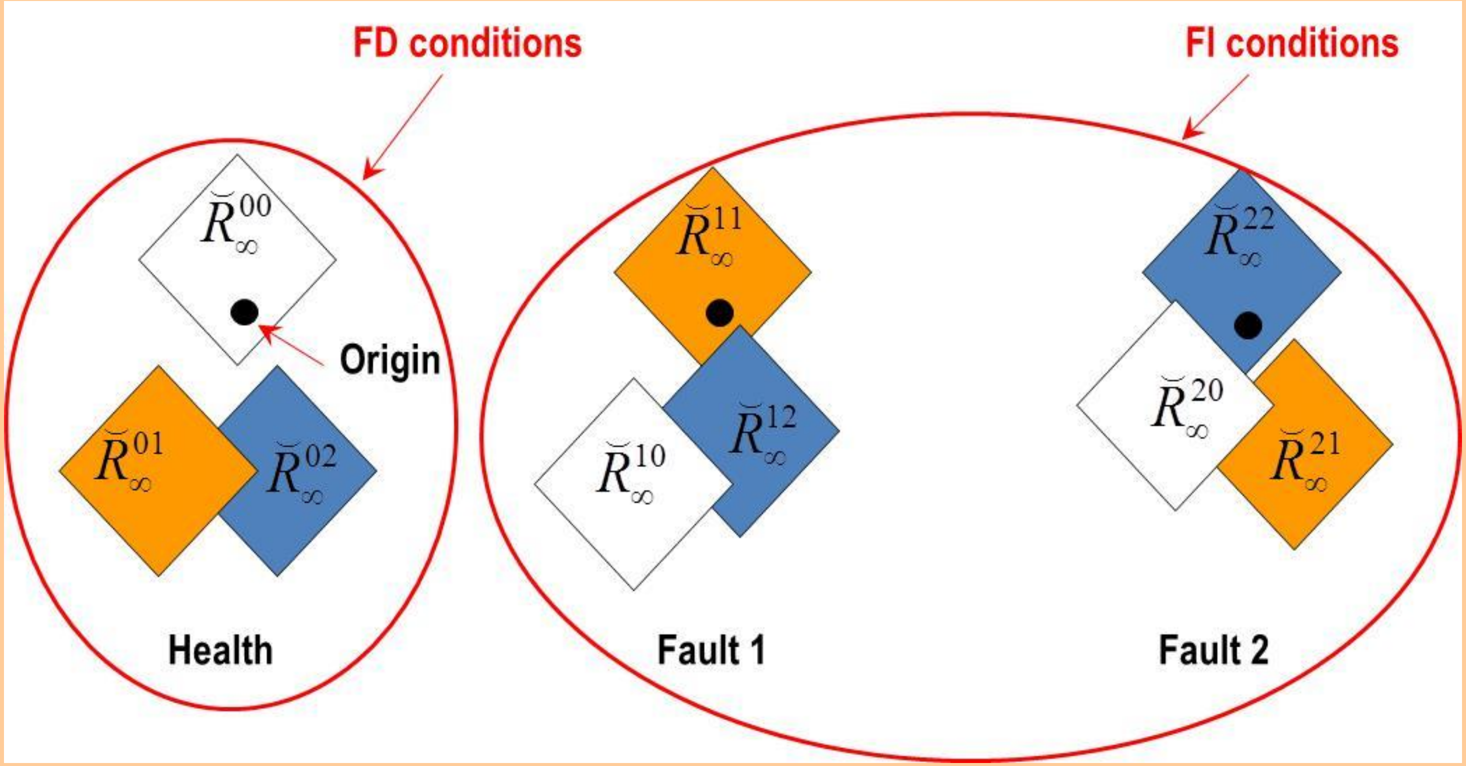
Invariant Sets
- Behaviors at steady state
- Lower fault sensitivity (construct sets off-line)
- Lower complexity

Interval Observers
- System behaviors at transient and steady state
- Higher fault sensitivity (estimate sets on-line)
- Higher complexity

Theoretical FDI Conditions

Theoretical FDI conditions :

$$0 \in \breve{R}_{\infty}^{ii} \text{ and } 0 \notin \breve{R}_{\infty}^{ij} \text{ for all } j \neq i$$



Index

1. Introduction
2. Interval Models for Fault Detection
3. Fault Detection using the Interval Observer Approach
4. Fault Detection using the Set-membership Approach
5. Identification for Robust Fault Detection
6. Fault-tolerance Evaluation
7. Real Applications
8. Conclusions
9. Further Research

Set-membership (or Consistency)-based Estimation Principle

- Let us denote the following sequences from the first time instant to time *k*:

$$\tilde{u}_k = (u_j)_0^{k-1} = (u_0, u_1, \cdots, u_{k-1})$$
$$\tilde{y}_k = (y_j)_0^{k-1} = (y_0, y_1, \cdots, y_k)$$
$$\tilde{w}_k = (w_j)_0^{k-1} = (w_0, w_1, \cdots, w_{k-1})$$
$$\tilde{v}_k = (v_j)_0^{k-1} = (v_0, v_1, \cdots, v_{k-1})$$
$$\tilde{\theta}_k = (\theta_j)_0^{k-1} = (\theta_0, \theta_1, \cdots, \theta_{k-1})$$

- The set of estimated states at time *k* using the ***set-membership approach*** is expressed by

$$\mathcal{X}_k = \left\{ \begin{array}{l} x_k \,\middle|\, \exists \tilde{w}, \tilde{v}, \tilde{\theta}, x_o \text{ such that} \\ (x_{k+1} = A(\theta_k)x_k + B(\theta_k)u_k + w_k)_{j=1}^{k} \\ (y_k = Cx_k + v_k)_{j=0}^{k} \end{array} \right\}$$

Implementation of Set-membership Estimators (1)

- The previous uncertain state set at time *k* can be computed approximately by admitting the rupture of the existing relations between variables of consecutive time instants.

- Two sets are introduced:

  - The ***set of predicted states*** at time *k* is given by

$$\mathbb{X}_k^p = \mathbf{x}_k : \overline{\mathbf{A}}(\theta_{k-1})\mathbf{x}_{k-1} + \overline{\mathbf{B}}(\theta_{k-1})\mathbf{u}_{k-1} + \overline{\mathbf{E}}\mathbf{y}_k + \overline{\mathbf{w}}_{k-1} | \\ \mathbf{x}_{k-1} \in \mathbb{X}_{k-1}, \theta_k \in \Theta, \overline{\mathbf{w}}_{k-1} \in \overline{\mathbb{W}}_{k-1} \Big\}$$

  - The ***set of consistent states*** at time *k* with measurement is defined as

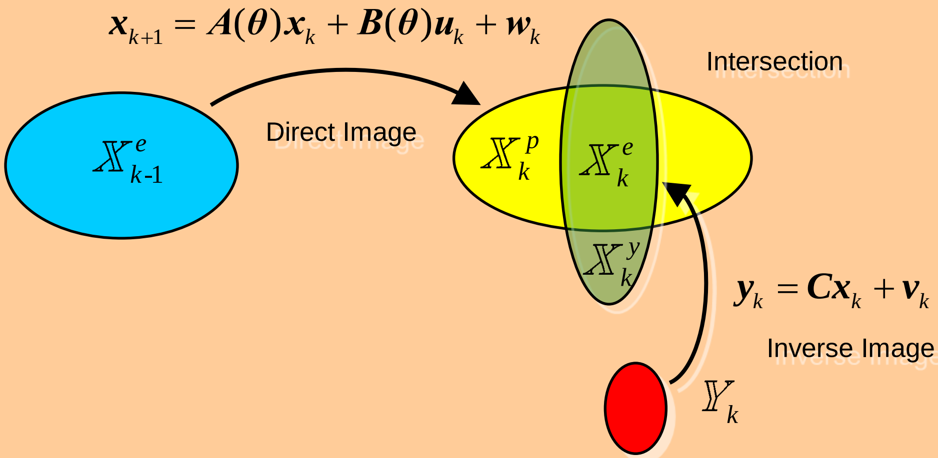
$$\mathbb{X}_k^{y_k} = \Big\{ \mathbf{x}_k \,:\, \mathbf{y}_k = \overline{\mathbf{C}}\mathbf{x}_k + \overline{\mathbf{v}}_k,\ \theta_k \in \Theta, \overline{\mathbf{v}}_k \in \overline{\mathbb{V}}_k \Big\}$$

Implementation of Set-membership Estimators (2)

- This allows to write the following algorithm:

Algorithm 1: Set-membership State Estimation using Set Computations

$\mathcal{X}_k^e \Leftarrow \mathcal{X}_0$
for $k = 1$ to N **do**
  Compute $\mathcal{X}_k^p$
  Compute $\mathcal{X}_k^c$
  Compute $\mathcal{X}_k^e = \mathcal{X}_k^p \cap \mathcal{X}_k^c$
end for



Fault Detection using Set-membership Estimation (1)

- ***Fault detection test:***

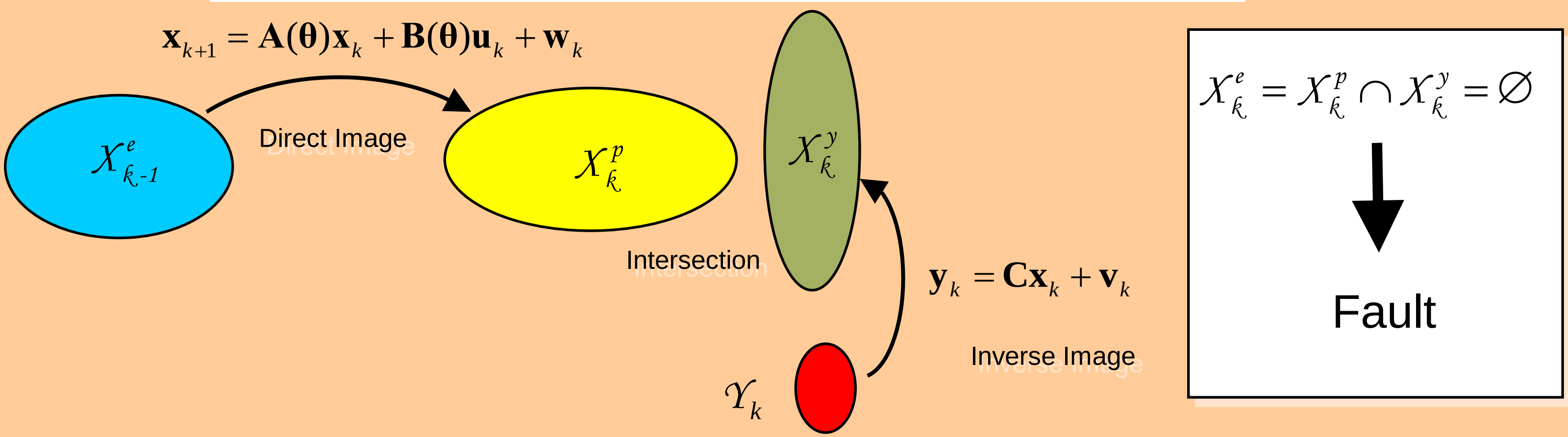
  Given the sequences of measured inputs $\tilde{\boldsymbol{u}}_k$ and outputs $\tilde{\boldsymbol{y}}_k$ of the actual system, a **fault** is said to have occurred at time *k* if there does not exist a set of sequences ($\tilde{\boldsymbol{w}}_k, \tilde{\mathbf{v}}_k, \tilde{\boldsymbol{\theta}}_k$) which satisfy the nominal system description with initial condition, noise, disturbances and parameters belonging to ($\mathcal{X}_o, \mathcal{V}, \mathcal{W}, \Theta$), respectively.

- ***Fault detection*** consists in detecting a fault given a sequence of measured inputs $\tilde{\boldsymbol{u}}_k$ and outputs $\tilde{\boldsymbol{y}}_k$.

Fault Detection using Set-membership Estimation (2)

Algorithm 2: Fault Detection using Set-membership Estimation

$\mathcal{X}_k^e \Leftarrow \mathcal{X}_0$
for $k = 1$ to N **do**
    Compute $\mathcal{X}_k^p$
    Compute $\mathcal{X}_k^c$
    Compute $\mathcal{X}_k^e = \mathcal{X}_k^p \cap \mathcal{X}_k^c$
    if $\mathcal{X}_k^e = \emptyset$ **then**
        Exit (Fault detected)
    end if
end for



Index

Identification for Robust Fault Detection

- One of the key points in model based fault detection is how detection models are estimated.

- In case of set-membership methods, the set for uncertain parameters should be estimated.

- The set for uncertain parameters depend on the way how the uncertain model will be used for fault detection.

- At least two possible types of models can be derived:

  - interval model
  - set-membership or consistency based model

Identification for the Direct Test (1)

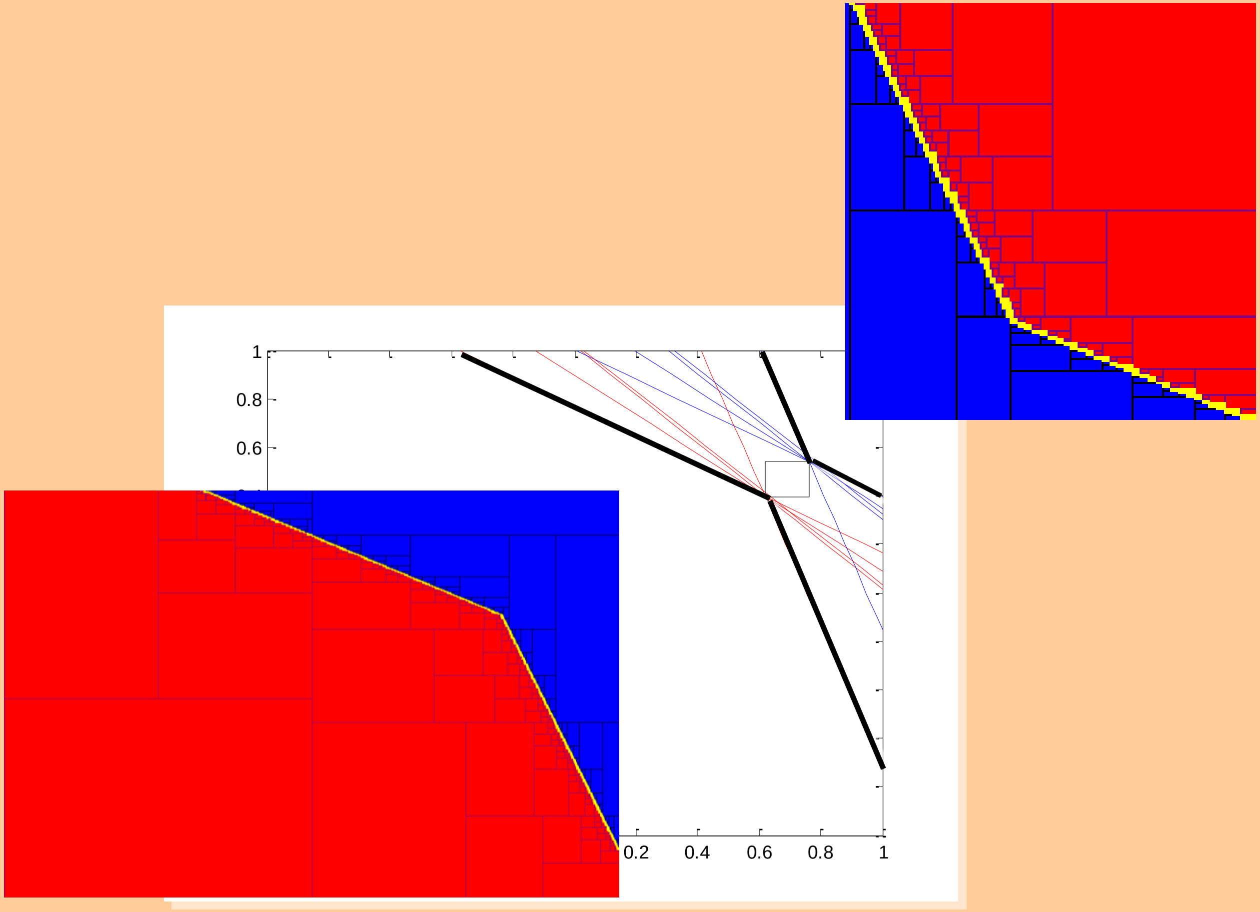
Given a set of measurements $y(k)$ taken in a given interval $k \in [0, N]$, considering that noise is bounded such that $y_m(k) \in Y_m(k)$, then a set of model parameters that produces an envelope that cover all measurements ("**worst-case approach**"):

$$\Theta = \left\{ \theta \in \Theta \mid \forall y(k) \in Y(k), \forall k \in [0, N] \quad (\underline{y}(k,\theta) \leq y(k)) \wedge (y(k) \leq \overline{y}(k,\theta)) \right\}$$

where at each time tinstant k, model temporal envelope is computed according to:

$$\begin{aligned} &\underline{y}(t_k) = min\, y(t_k, \theta) \\ &sujeto\, a: \\ &\theta \in \Theta \end{aligned} \qquad \begin{aligned} &\overline{y}(t_k) = max(t_k, \theta) \\ &sujeto\, a: \\ &\theta \in \Theta \end{aligned}$$

Identification for the Direct Test (2)



Identification for the Inverse Test (1)

Given a set of measurements $y(k)$ taken in a given interval $k \in [0, N]$, considering that noise is bounded such that $y_m(t) \in Y_m(t)$, then a set of model parameters that are consistent with model and measurements would be estimated such that ("**consistency approach**"):

$$\Theta = \left\{ \theta \in \Theta \mid \exists y(k) \in Y(k), \forall k \in [0, N] \quad \underline{y}(k) \leq y(k, \theta) \leq \overline{y}(k) \right\}$$

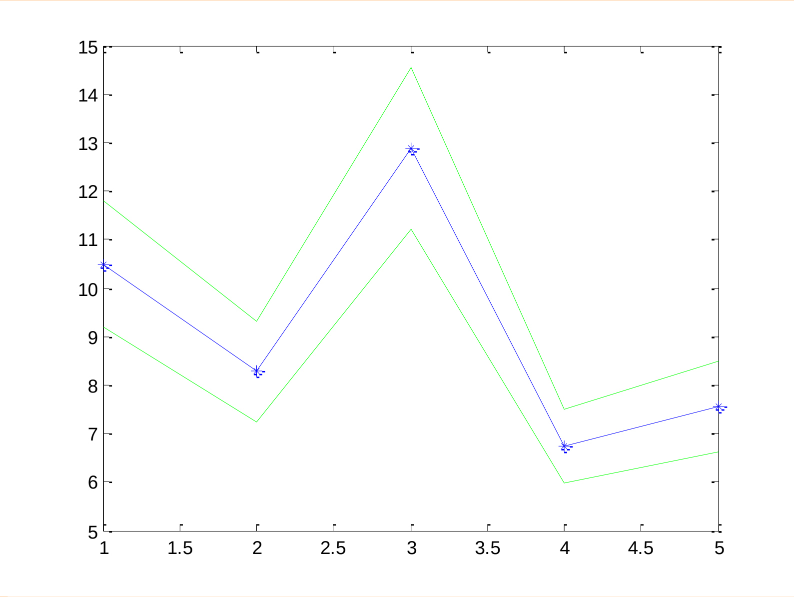
This set can be computed at each sample time instant k :

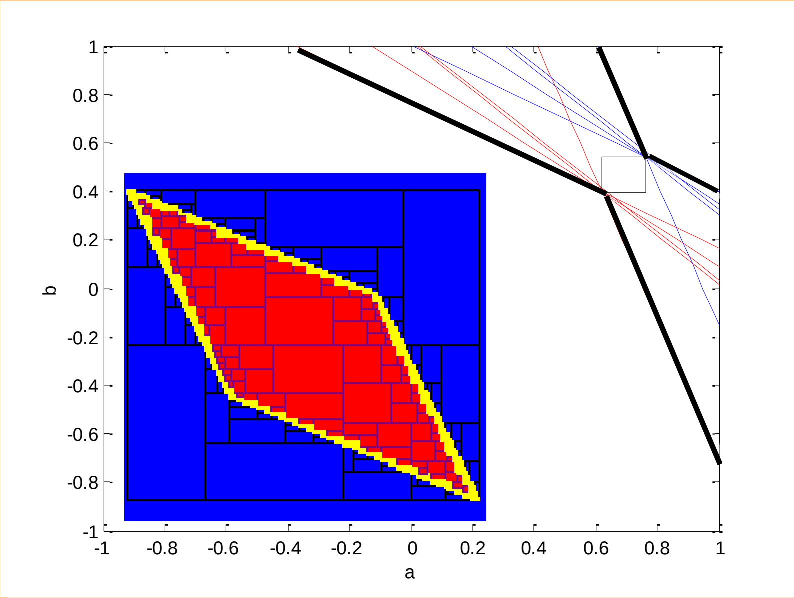
$$\Theta = \left\{ \theta \in \Theta \mid \exists y(k) \in Y(k) \quad \underline{y}(k) \leq y(k, \theta) \leq \overline{y}(k) \right\}$$

such that:

$$\Theta = \bigcap_{k=1}^{N} \Theta_k$$

Identification for the Inverse Test (2)





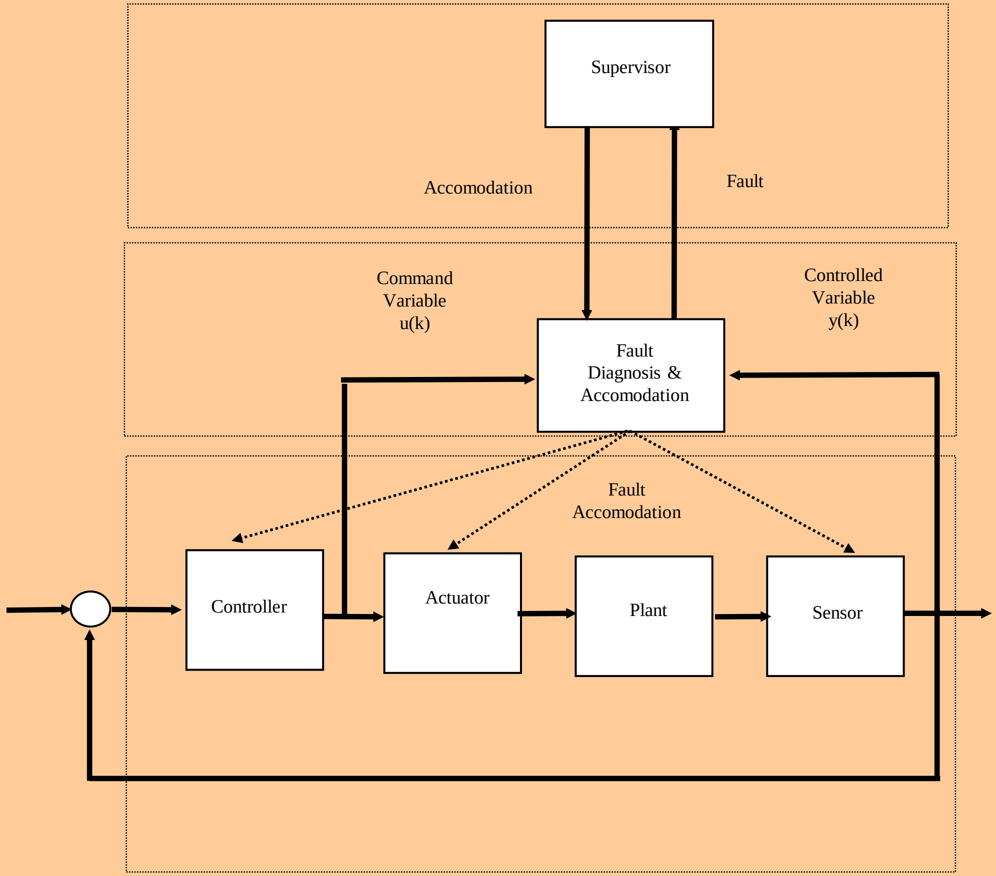
Index

Fault Tolerant Control

Level 3
Supervision

Level 2
Fault Diagnosis
& Accomodation

Level 1
Control



Fault tolerant MPC problem

- The solution of a control problem consists on finding a control law in a given set of ***control laws*** $\mathcal{U}$ such that the controlled system achieves the ***control objectives*** $\mathcal{O}$ while its behavior satisfies a set of ***constraints*** $\mathcal{C}$.
- The solution of the problem is completely defined by the triple: $\langle \mathcal{U}, \mathcal{O}, \mathcal{C} \rangle$
- In the case of a linear constrained predictive control law:

$$\mathcal{O} : \min_{\tilde{u}} J(\tilde{x}, \tilde{u})$$

subject to:

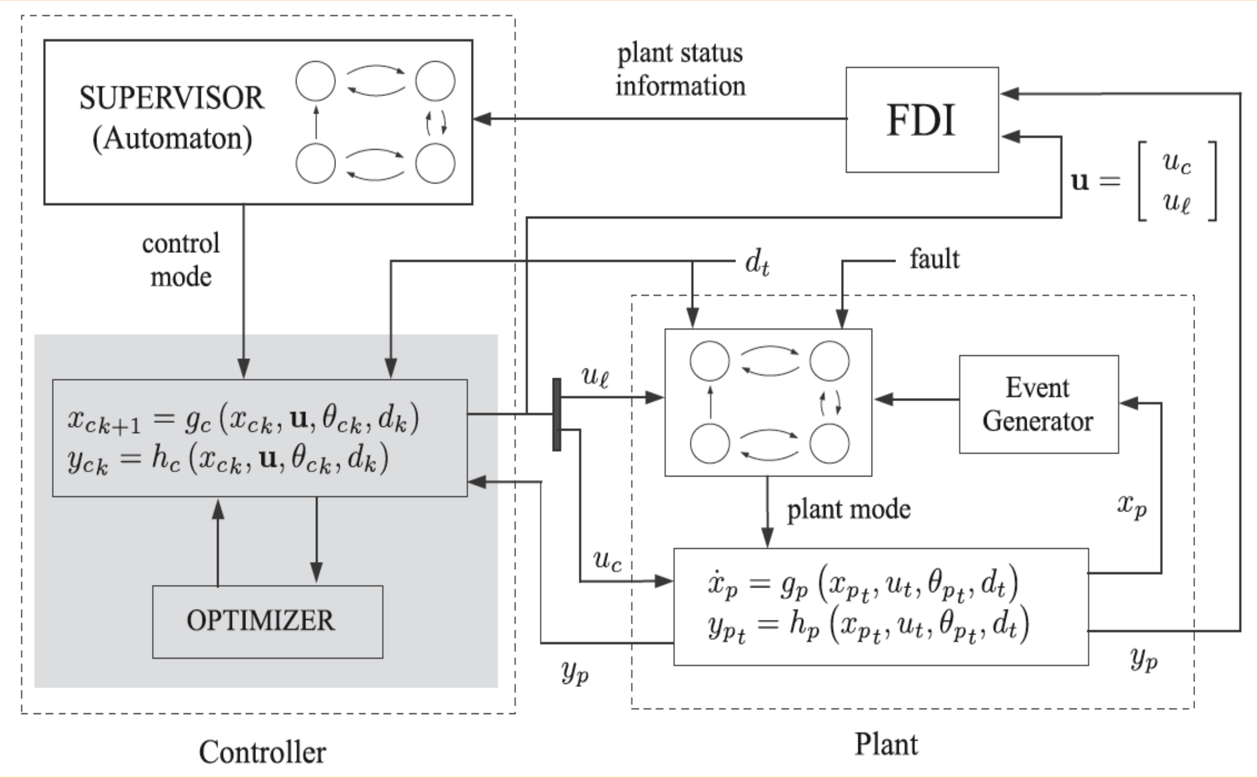
$$\mathcal{C} : \begin{cases} x_{k+1} = f(x_k, u_k) \\ u_k \in \mathcal{U} \quad k = 1, \cdots, N-1 \\ x_k \in \mathcal{X} \quad k = 0, \cdots, N \end{cases}$$

where:

$$\mathcal{U} = \left\{ u_k \in \mathbb{R}^m \middle| u_{min} \le u_k \le u_{min} \right\}$$
$$\mathcal{X} = \left\{ x_k \in \mathbb{R}^n \middle| x_{min} \le x_k \le x_{min} \right\}$$

$$\tilde{u}_k = (u_j)_0^{k-1} = (u_0, u_1, \cdots, u_{k-1})$$
$$\tilde{x}_k = (x_j)_0^{k-1} = (x_0, x_1, \cdots, x_k)$$

Hybrid MPC Fault-tolerant Control



Preliminary Definitions

- *Definition 1.* The ***feasible solution set*** is given by

$$\Omega = \left\{ \tilde{x}, \tilde{u} \middle| \left(x_{k+1} = f(x_k, u_k) \right)_0^{N-1} \right\}$$

and gives the input and state sets compatible with system constraints which originate the set of predictive states.

- *Definition 2.* The ***feasible control objectives set*** is given by

$$J_\Omega = \left\{ J(\tilde{x}, \tilde{u}) \middle| (\tilde{x}, \tilde{u}) \in \Omega \right\}$$

and corresponds to the set of all values of *J* obtained from feasible solutions.

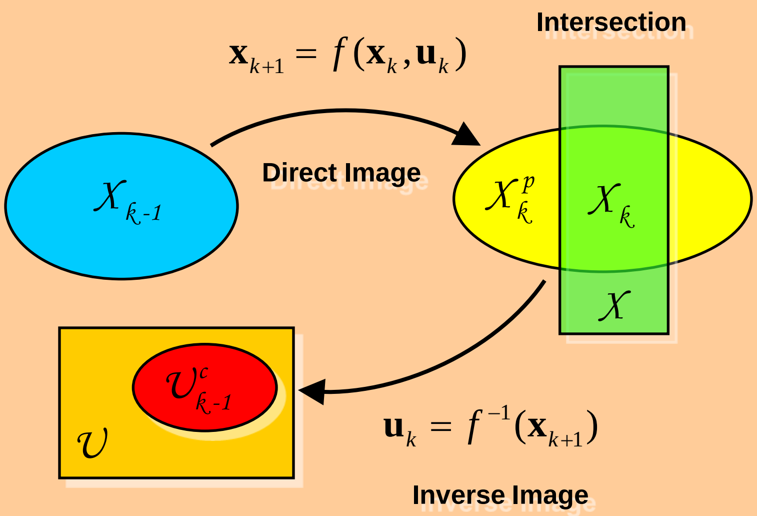
- *Definition 3.* The ***admissible solution set*** is given by $\mathcal{A} = \left\{ (\tilde{x}, \tilde{u}) \in \Omega_f \middle| J(\tilde{x}, \tilde{u}) \in \mathcal{J}_{\mathcal{A}} \right\}$ where Ω_f corresponds to the feasible solution set of a actuator fault configuration and $\mathcal{J}_{\mathcal{A}}$ defined as the admissible control objective set.

Admissibility Evaluation using Set Computations (1)

- The admissibility evaluation using a set computation approach starts obtaining the ***feasible solution set*** Ω given a set of initial states $\mathcal{X}_0$, the system dynamic and the system operating constraints over *N*.

Algorithm 1 Computation of Ω

$\mathcal{X}_k \Leftarrow \mathcal{X}_0$
$\Omega_0 \Leftarrow \mathcal{X}_0$
for $k = 1$ to N **do**
  $\mathcal{U}_{k-1} \Leftarrow \mathcal{U}$
  Compute $\mathcal{X}_k^p$ from $\mathcal{X}_{k-1}$ and $\mathcal{U}_{k-1}$
  Compute $\mathcal{X}_k^c = \mathcal{X} \cap \mathcal{X}_k^p$
  Compute $\mathcal{U}_{k-1}^c$ from $\mathcal{X}_k^c$
  $\Omega_k = \mathcal{X}_k^c \times \mathcal{U}_{k-1}^c$
  $\mathcal{X}_k \Leftarrow \mathcal{X}_k^c$
end for
$\Omega = \bigcup_{k=0}^{N} \Omega_k$

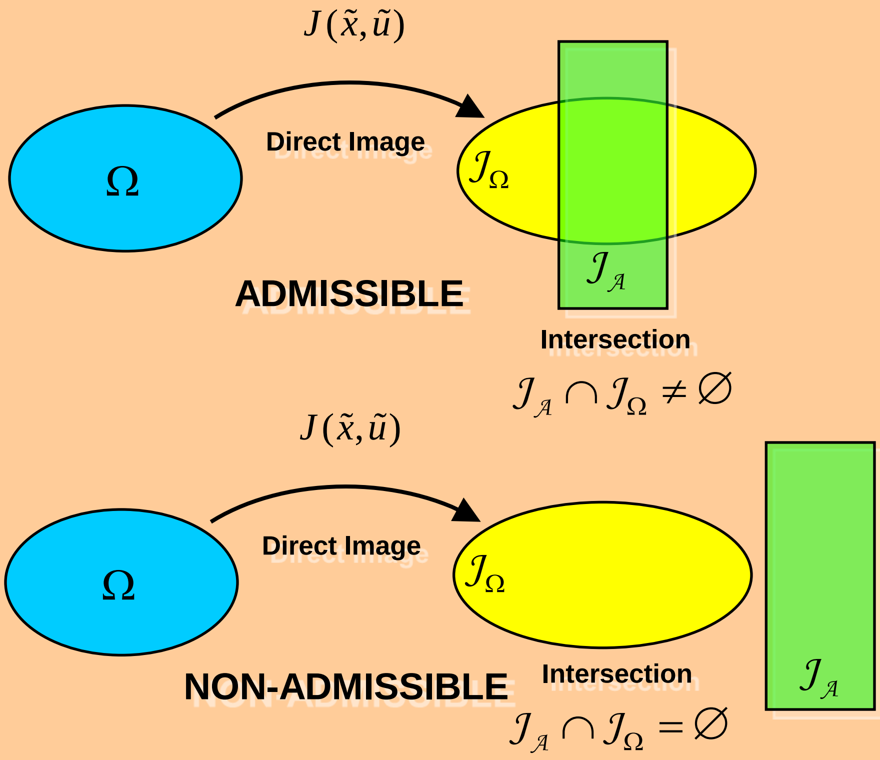


Admissibility Evaluation using Set Computations (2)

- At the same time that the ***feasible solution set*** is computed Ω , the ***feasible control objectives set*** $\mathcal{J}_\Omega$ at time *k*=*N* can be obtained using the following algorithm:

Algorithm 2 Computation of $\mathcal{J}_\Omega$ using Ω_k

$\mathcal{X}_k \Leftarrow \mathcal{X}_0$
$\Omega_0 \Leftarrow \mathcal{X}_0$
for $k = 1$ to N **do**
  Compute Ω_k *(See Algorithm 1)*
  Compute $\mathcal{J}_{\Omega_k}$ using $\Omega_k = \mathcal{X}_k^c \times \mathcal{U}_{k-1}^c$
end for
$\mathcal{J}_\Omega = \bigcup_{k=0}^{N} \mathcal{J}_{\Omega_k}$



Admissibility Evaluation using Constraints Satisfactions (1)

- ***Constraints satisfaction problem***:

  "*A **constraints satisfaction problem** (CSP) on sets can be formulated as a 3-tuple H = (V,D,C) where:*

  - V = { v_1 , ⋯ ,v_n } *is a finite set of variables,*
  - D = {D_1 , ⋯ ,D_n } *is the set of their domains represented by closed sets*
  - C ={c_1 , ⋯ ,c_n } *is a finite set of constraints relating variables of V "*

- A point solution of H is a n-tuple (v_1 ,⋯ ,v_n) 2 D such that all constraints C are satisfied.
- The set of all point solutions of H is denoted by S(H). This set is called the global solution set.
- The variable v_i 2 V_i is consistent in H if and only if:

$$\forall v_i \in \mathcal{V}_i \ \exists \ (\tilde{v}_1 \in \mathcal{D}_1, \cdots, \tilde{v}_n \in \mathcal{D}_n) \ |(\tilde{v}_1, \cdots, \tilde{v}_n) \in \mathcal{S}(\mathcal{H})$$

with *i=1...n*

Admissibility Evaluation using Constraints Satisfaction (2)

- The admissibility evaluation requires the computation of the admissible solution set: $\Omega = \left\{ \tilde{x},\tilde{u} \middle| \left(x_{k+1} = f(x_k, u_k) \right)_0^{N-1} \right\}$
- Its definition suggests a way of implementation since its mathematical description can be viewed as a constraints satisfaction problem:

Algorithm 1: Admissibility Evaluation using Constraints Satisfaction

At each time instant k over N, the feasible solution set is determined by solving the CSP $\mathcal{H} = (\mathcal{V}, \mathcal{D}, \mathcal{C})$ associated with the constraints $\mathcal{C}$ of the CNMPC problem,where

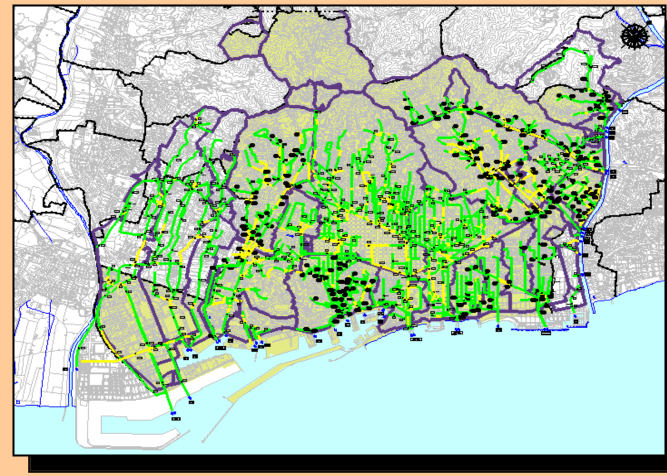
$$\begin{aligned}
\mathcal{V} &= \{\overbrace{x_1, x_2, \cdots, x_N}^{\tilde{x}}, \overbrace{u_1, u_2, \cdots, u_{N-1}}^{\tilde{u}}, J\} \\
\mathcal{D} &= \{\mathcal{X}_1, \mathcal{X}_2, \cdots, \mathcal{X}_N, \mathcal{U}_1, \mathcal{U}_2, \cdots, \mathcal{U}_{N-1}, \mathcal{J}_\mathcal{A}\} \\
\mathcal{C} &= \Big\{ \left(x_{k+1} = f(x_k, u_k) \right)_0^{N-1}, \\
&\qquad J(\tilde{x}, \tilde{u}) = \phi(x_N) + \sum_{i=0}^{N-1} \Phi(x_i, u_i) \Big\}
\end{aligned}$$

Index

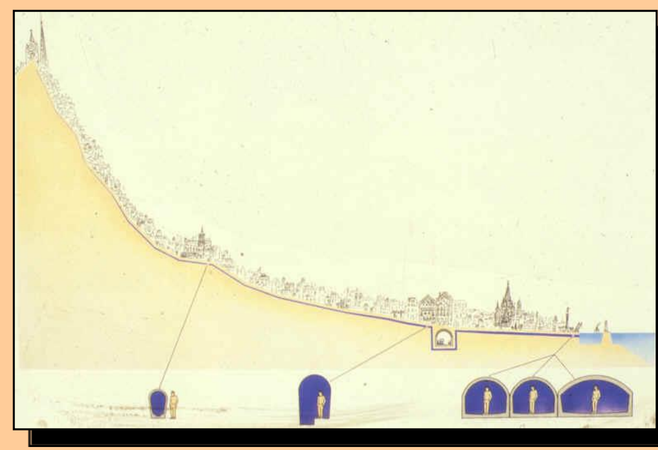
1. Introduction
2. Interval Models for Fault Detection
3. Fault Detection using the Interval Observer Approach
4. Fault Detection using the Set-membership Approach
5. Identification for Robust Fault Detection
6. Fault-tolerance Evaluation
7. Real Applications
8. Conclusions
9. Further Research

The Barcelona Sewer Network

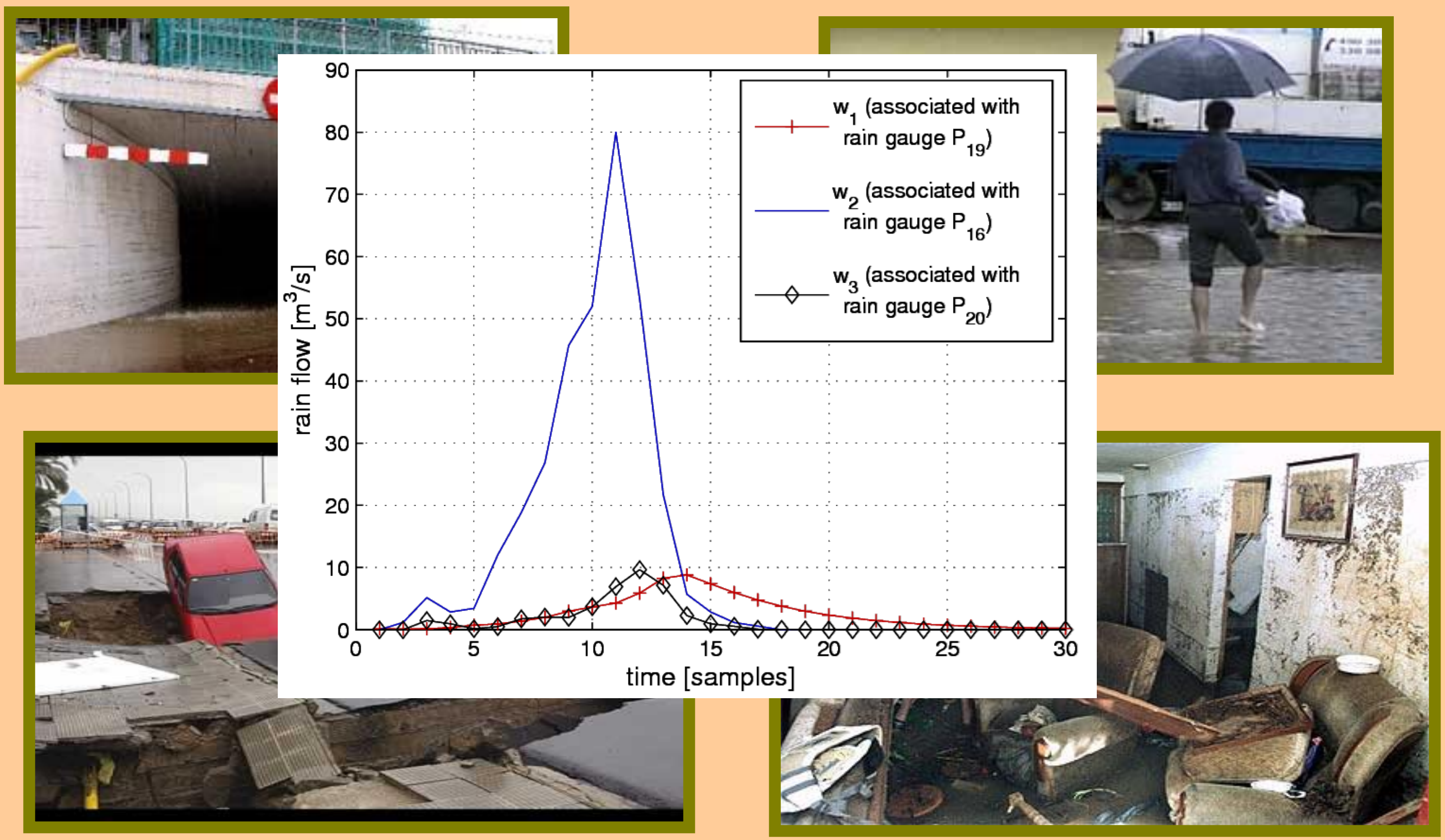
- Data
  - Typology combined
  - Length 1.650 km
  - Storage capacity 2.634.124 m^3
  - Visitable portion 55,12%
  - Mean transversal section 1,8 m^2
  - 31 catchment area 12.326 ha



- Particularities
  - Topographic profile: steep slope, gentle at rivers and sea
  - Urban ground: 90% impervious
  - Meteorology: yearly precipitation: 600mm, intensity: up to 150 mm/h in 15 minutes



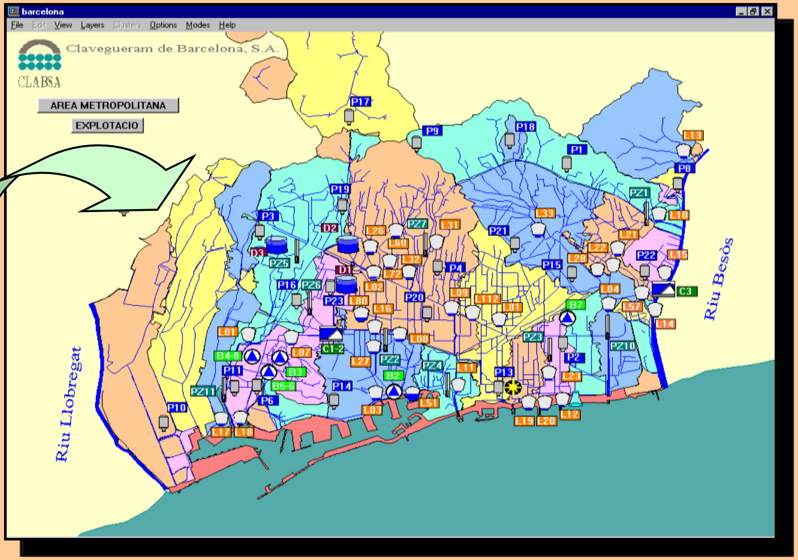
Barcelona and its Rain



Solution (1): Detention Tanks

Solution (2): Barcelona's RTC System

ELEMENTS	NUMBER
Rain gauges	22
Water level sensors	119
Pumping Stations	11
Gates	23
Detention Tanks	10



MPC Multicriteria optimization

$$J = \sum_{k=0}^{N-1} (\alpha J_{flood}{}^{k} + \beta J_{CSO}{}^{k} + \gamma J_{WWTP}{}^{k})$$

- Reduction of the risk of floods

$$J_{flood} = \sum_{j} \max(o, q_j - \lim_{q_j})$$

q_j flow through sewer *j*

- Environment protection

$$J_{cso} = \sum_{l} CSO_l$$

CSO_l^k combined sewer overflow volume at site *l*

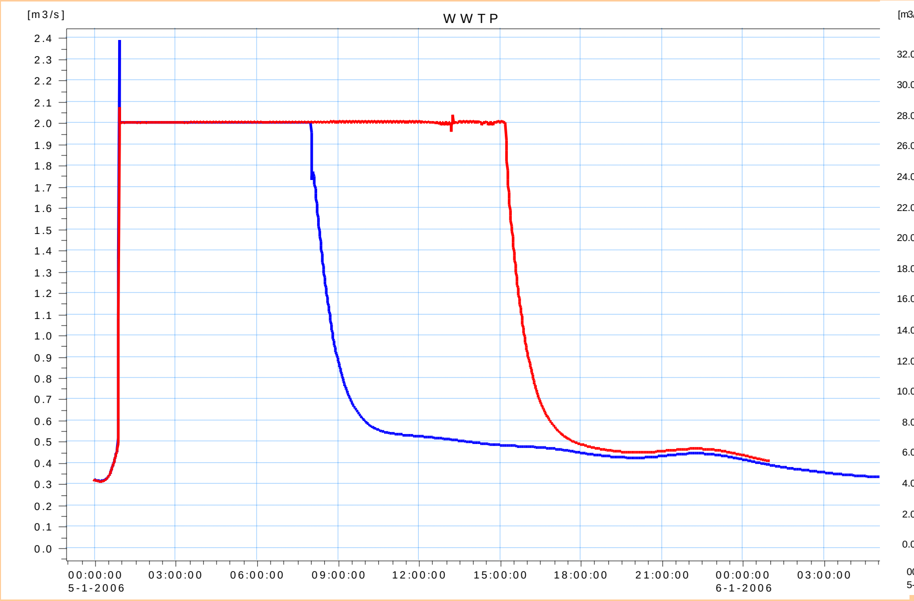
- Optimization of the WWTP

$$J_{WWTP} = \sum_{i} (WWTP_i - WWTP^*{}_i)$$

$WWTP_i$ waste water treatment plant flow *i*

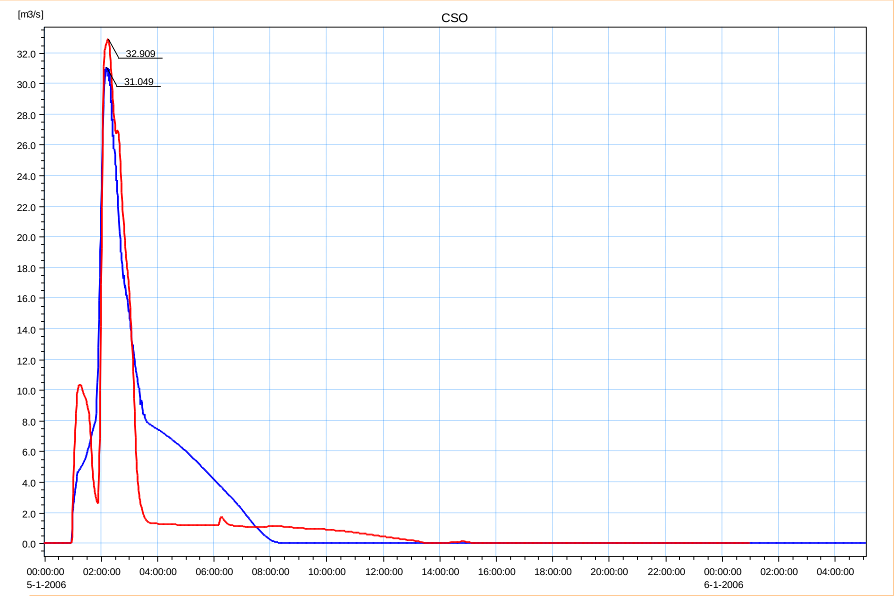
Global Control vs Local control

WWTP Volume



50 % improvement

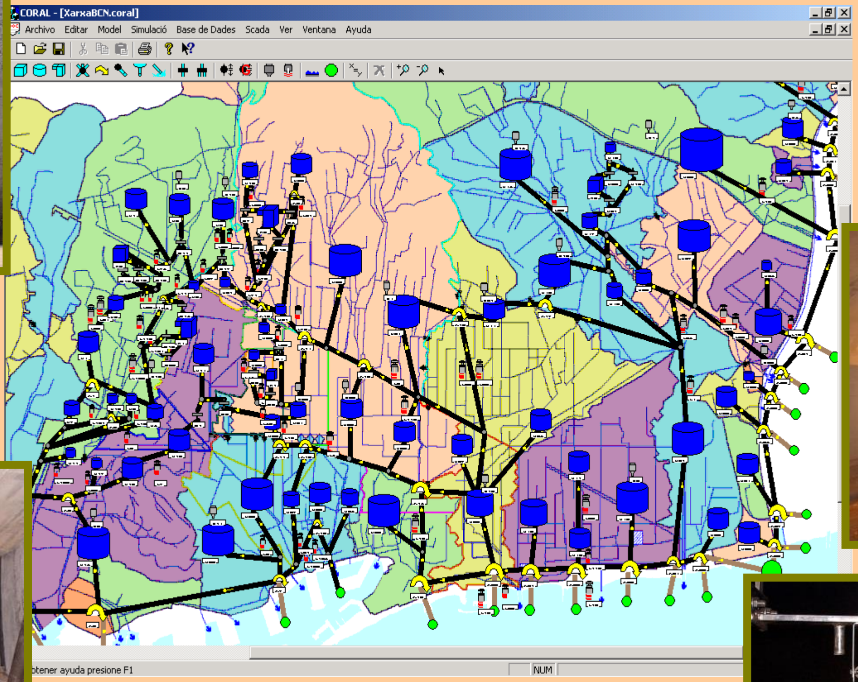
CSO Volume

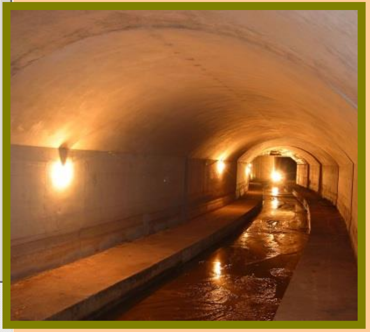


-18 % reduction

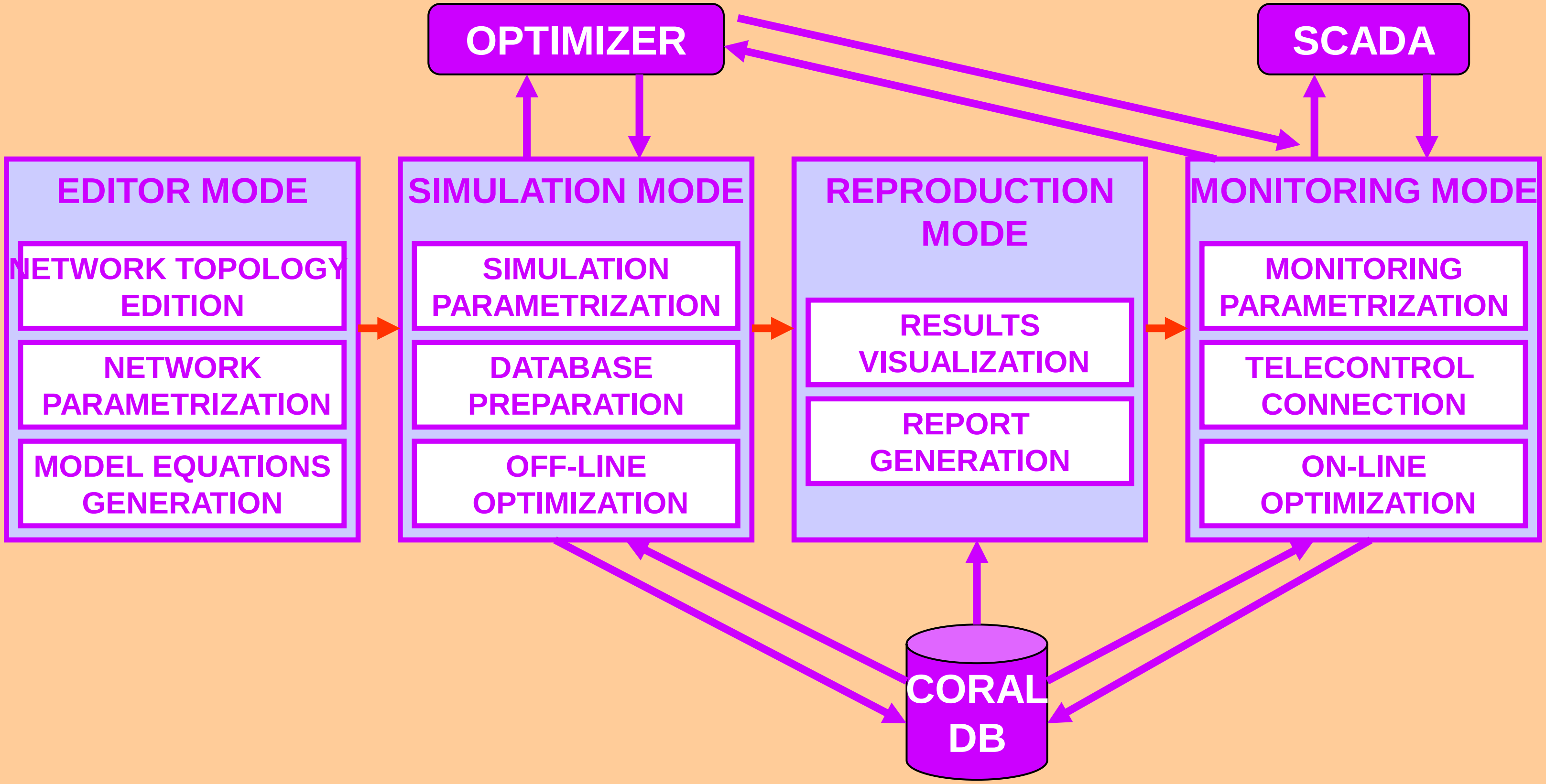
Blue: Local Control
Red: Global Control

CORAL: MPC tool for Sewer Networks





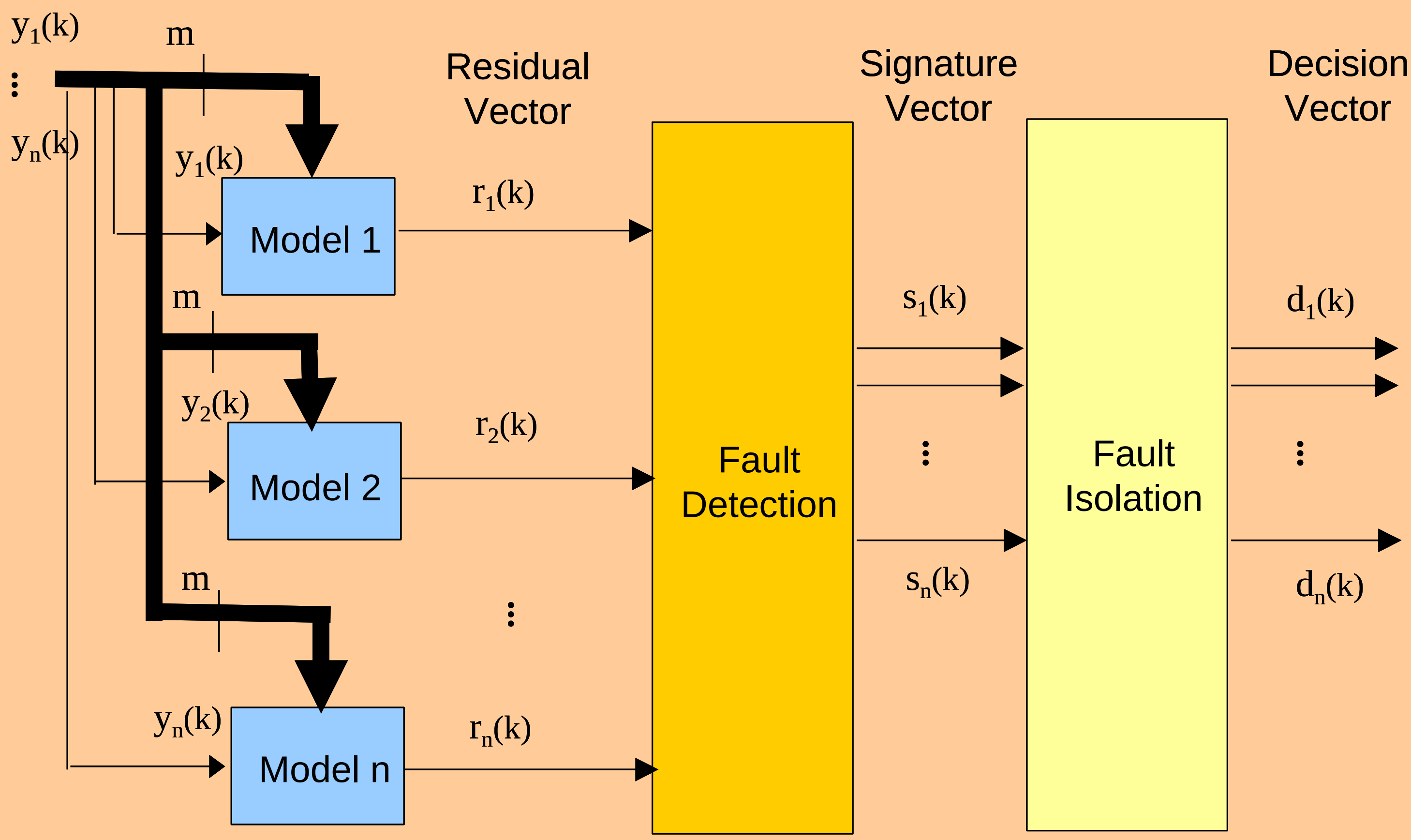
CORAL Architecture



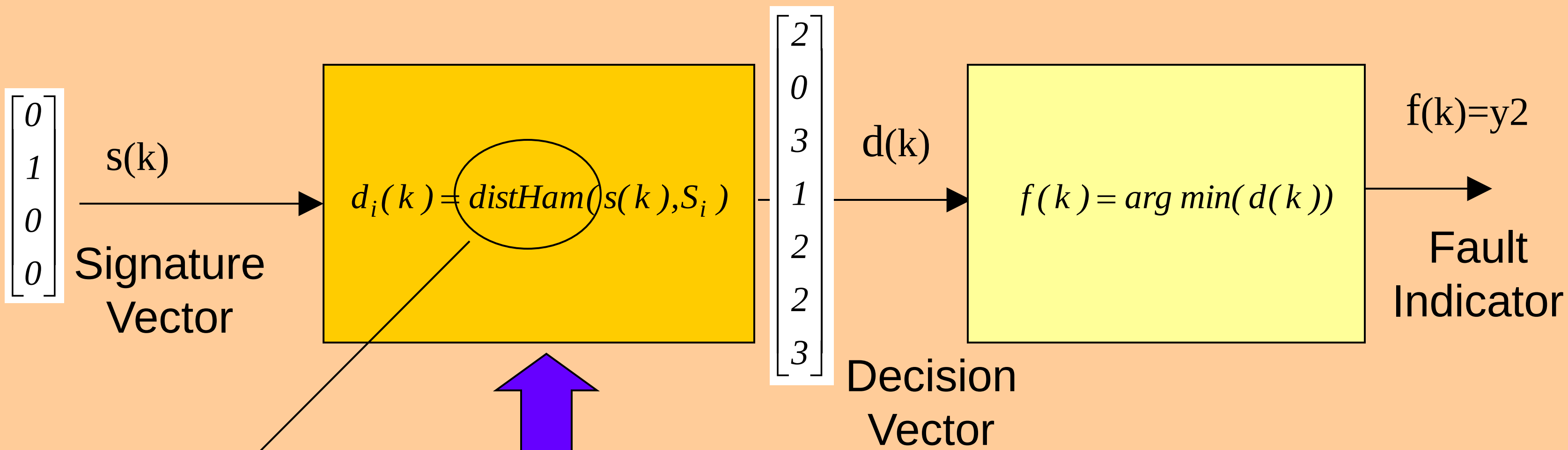
Introduction to FDI in Sewer Networks

- In this presentation, the FDI problem of rain gauges and limnimeters of Barcelona's urban sewer system is addressed.
- Rain gauges and limnimeters are used for the real-time global control of the whole Barcelona network.
- Often these instruments are out of order in rain scenarios when the control system must be fully operative.
- In order to detect and isolate faulty instruments and to reconstruct faulty measurements from data fusion, a fault diagnosis system is necessary.

The Architecture of the FDI System



Fault Isolation Procedure



Hamming Distance: number of different bits between two binary codes

$$distHam = c_1 \oplus c_2$$

	y1	y2	y3	y4	y5	y7	y8
r1	1	0	1	0	0	0	1
r2	1	1	0	1	0	0	0
r3	1	0	1	0	0	1	0
r4	0	0	0	1	1	0	1

$S_i(k)$

Signature Matrix

$$r_1(k) = y_1(k) - a_{13}y_3(k) - a_{18}y_8(k)$$
$$r_2(k) = y_2(k) - a_{21}y_1(k) - a_{24}y_4(k)$$
$$r_3(k) = y_3(k) - a_{31}y_1(k) - a_{37}y_7(k)$$
$$r_4(k) = y_4(k) - a_{45}y_5(k) - a_{48}y_8(k)$$

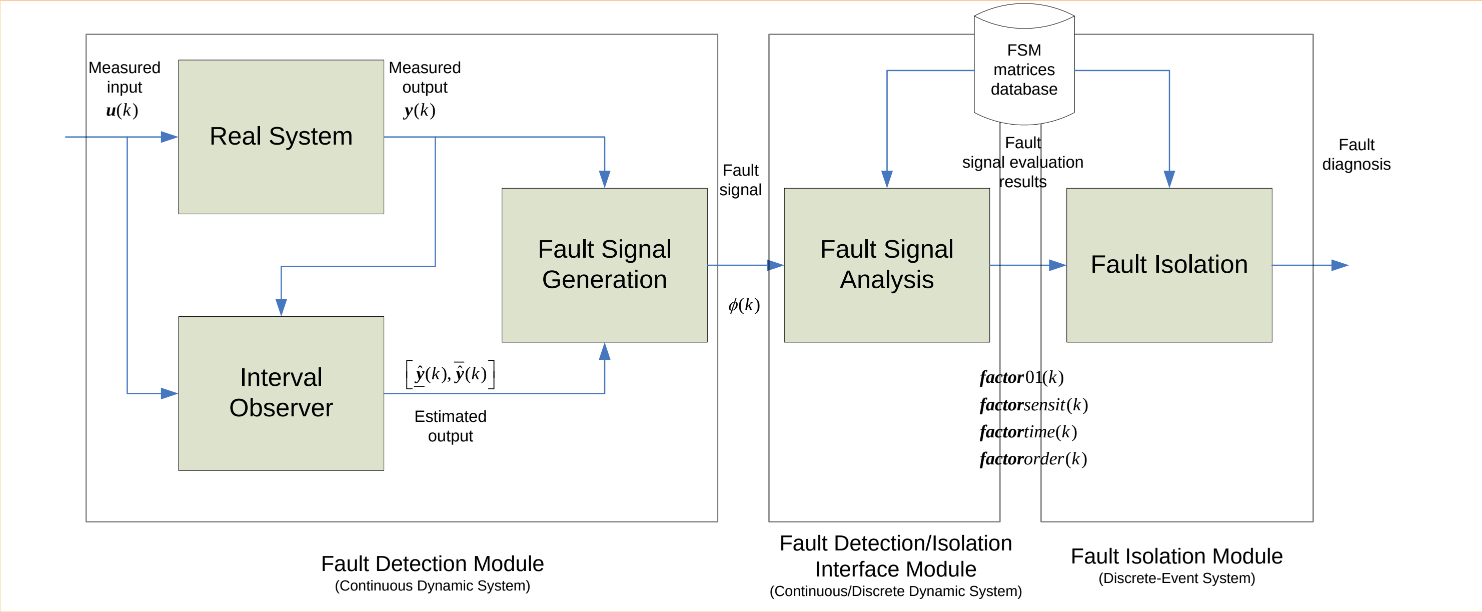
Usual assumptions

- Single fault and exoneration
- If multiple faults, no compensation

Enhanced Fault Isolation Scheme

In particular, such interface can be improved taking into account the following information:

- ***residual value size***: big violation of the threshold or only a small fault signal activation.
- ***residual sensitivity*** with respect to a certain fault.
- ***time pattern*** of fault signal occurrence.
- ***order*** of fault signal occurrence.



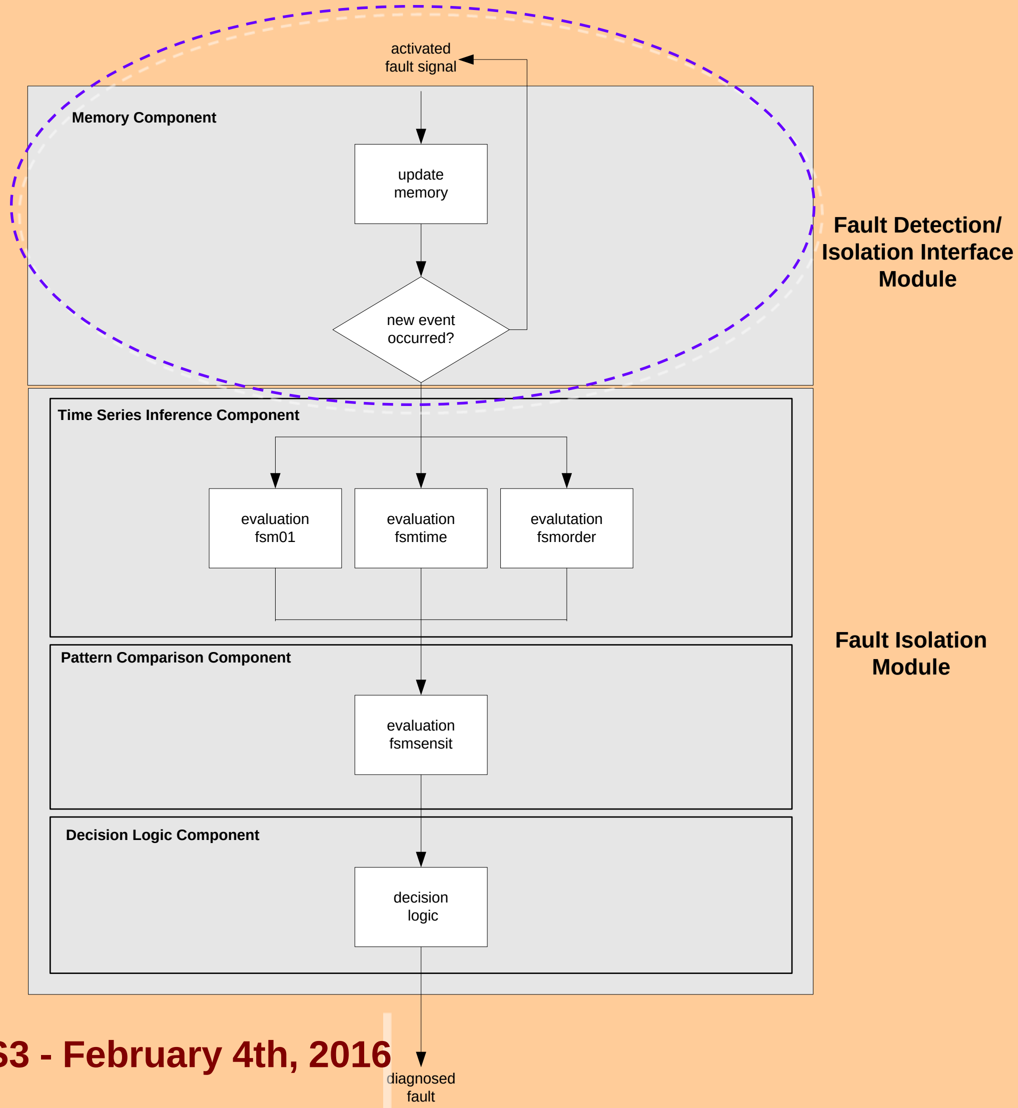
Interface between Fault Detection and Isolation Modules

- The interface is based on a memory implemented as a table in which events in the residual history are stored:

$$\phi_i(k) = \begin{cases} \dfrac{(r_i^o(k)/\overline{r}_i^o(k))^4}{1+(r_i^o(k)/\overline{r}_i^o(k))^4} & if \quad r_i^o(k) \geq 0 \\ -\dfrac{(r_i^o(k)/\underline{r}_i^o(k))^4}{1+(r_i^o(k)/\underline{r}_i^o(k))^4} & if \quad r_i^o(k) < 0 \end{cases}$$

- For each row, the first column stores the occurrence time t_i, the second one stores, the $\phi_{i,max}$, and the third one stores the sign of the residual.

- If the fault detection component detects a new fault signal, it updates the memory by filling out the three fields.



Fault Detection and Isolation Interface: *FSM* Matrices

- It is based on the concept of the theoretical ***fault signature matrix*** (FSM) which was introduced by (Gertler, 1998).
- This matrix stores the theoretical binary influence of a given fault f_j (column of ***FSM***) on a given residual $r_i(k)$ or equivalently, on a given fault signal $\phi_i(k)$ (row of ***FSM***).
- Here, the fault signature matrix concept is generalized since the binary interface is extended taking into account more fault signal properties.

Fault Signal Properties	***FSM*** *Matrix*
Binary	***FSM*** *01*
Sign	***FSM*** *sign*
Fault residual sensitivity	***FSM*** *sensit*
Occurrence order	***FSM*** *order*
Occurrence time instant	***FSM*** *time*

Limnimeter Modelling (1): "Virtual Reservoir Approach"

- Propagation of flows through sewer pipes can be described by numerical solution of the continuity and momentum Saint-Vennant's partial differential equations.

$$\frac{\partial Q}{\partial x} + \frac{\partial A}{\partial t} = 0$$

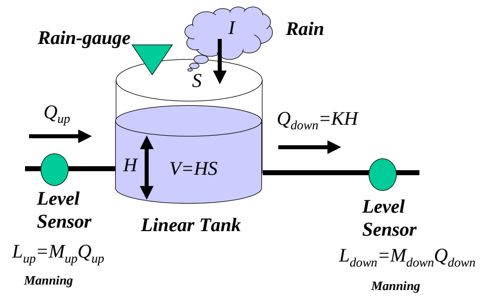
$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x}\left(\frac{Q^2}{A}\right) + gA\frac{\partial y}{\partial x} - gA\left(I_0 - I_f\right) = 0$$

- Saint-Vennant's equations can be used to perform simulation studies but are highly complex to solve in real-time, specially for large scale systems.

Limnimeter Modelling (2): “Virtual Reservoir Approach”

- The sewerage network is modeled through a simplified graph relating the main sewers and set of virtual and real reservoirs.
- A virtual reservoir is an aggregation of a catchment of the sewage network which approximates the hydraulics of rain, runoff and sewage water retention thereof.
- The hydraulics of virtual reservoirs are:

$$\frac{dV(t)}{dt} = Q_{in}(t) + I(t)S - Q_{out}(t)$$

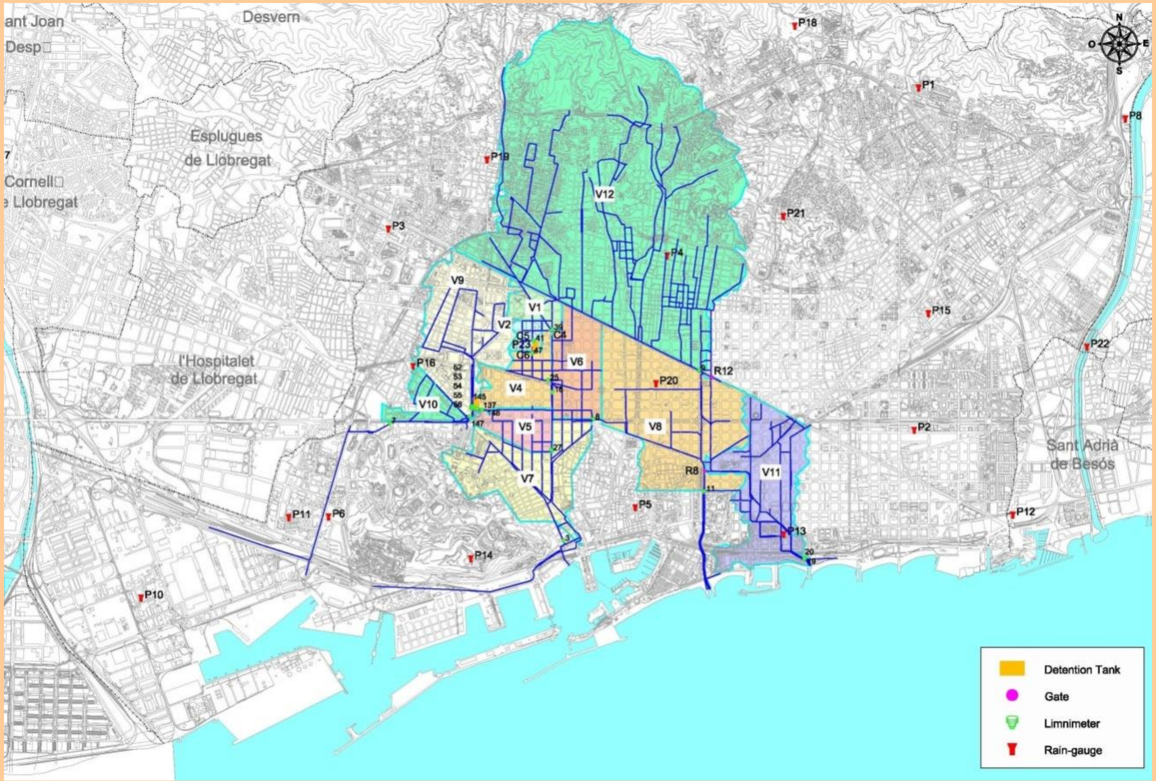


Using Manning’s formula and discretising:

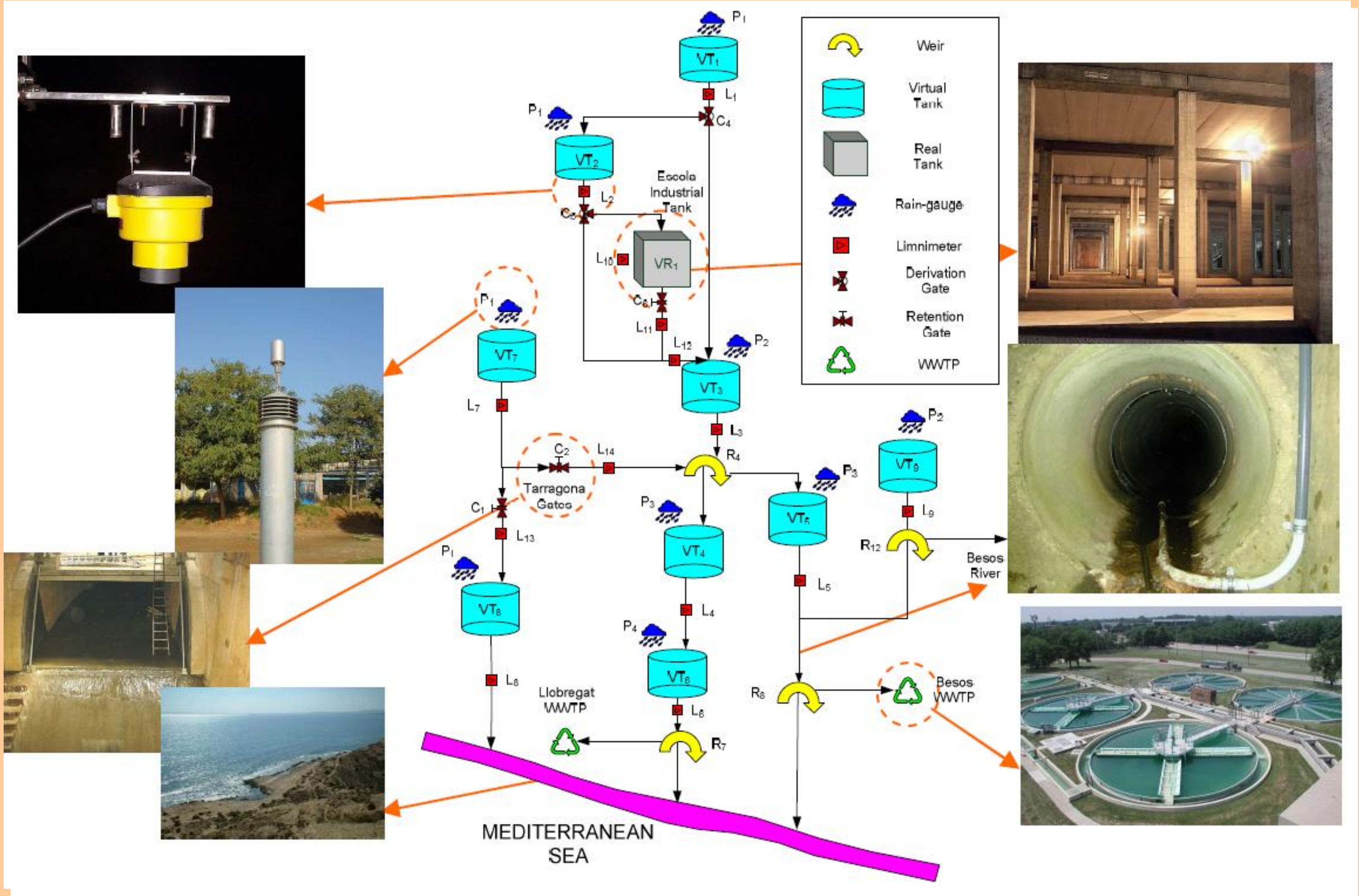
$$Q_{up}(t) = M_{up} L_{up}(t)$$
$$Q_{down}(t) = M_{down} L_{down}(t)$$

$$L_{down}(k+1) = aL_{down}(k) + bL_{up}(k) + cI(k))$$

Application Example (1): Modeling Barcelona Sewer Network using Virtual Tanks



Application Example (2): Modeling Barcelona Sewer Network using Virtual Tanks



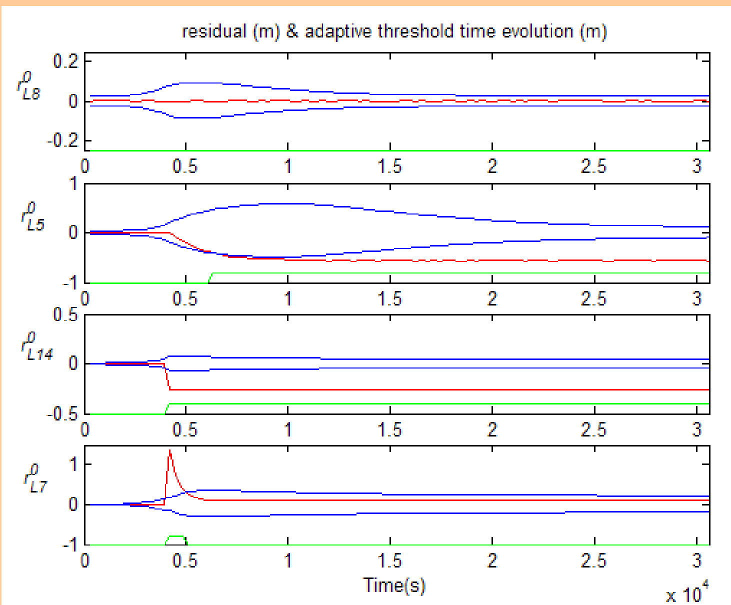
Application Example: Structure of the Limnimeter Models

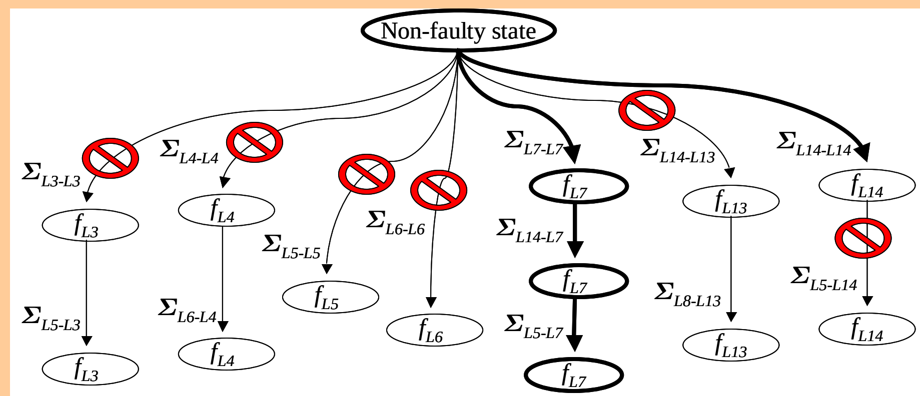
- Applying the limnimeter modelling methodology based on “virtual tanks” to the considered sewer network:
  - 12 limnimeters are modelled allowing to compute 12 residuals.
  - Faults affecting 14 limnimeters can be diagnosed.

	L_1	L_2	L_3	L_4	L_5	L_6	L_7	L_8	L_9	L_{10}	L_{11}	L_{12}	L_{13}	L_{14}	P_1	P_2	P_3	P_4
L_1	X														X			
L_2	X	X													X			
L_3			X									X				X		
L_4				X													X	
L_5			X		X		X							X			X	
L_6				X		X												X
L_7							X								X			
L_8								X					X		X			
L_9									X							X		
L_{10}		X								X	X							
L_{12}											X	X						
L_{14}							X						X	X				

Application Example:
Fault Scenario affecting L_7

- A fault affecting limnimeter L_7 occurs at t_0= *4000s*.





Residual time evolution

Fault Tolerant Control

Application Example (1)

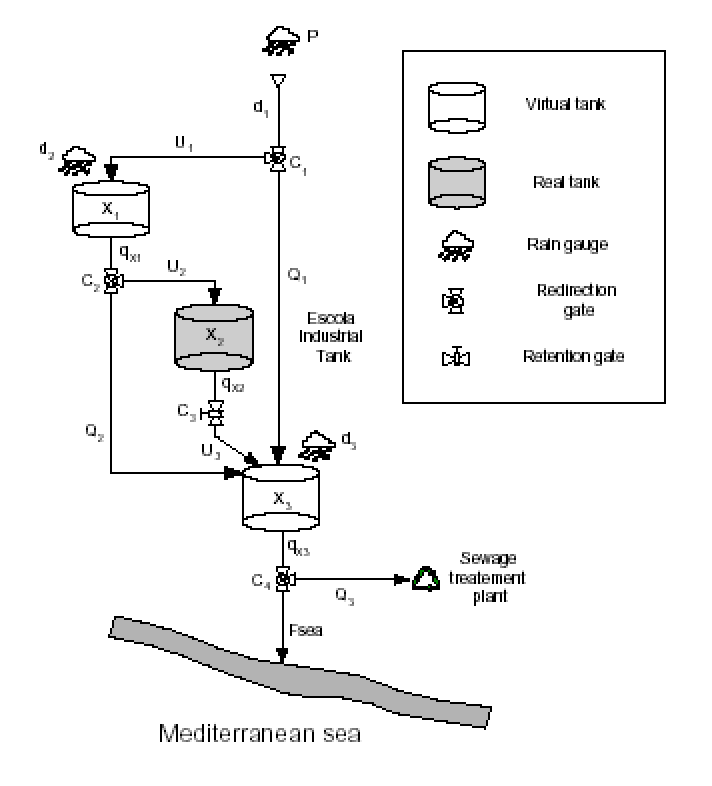
- Consider the system corresponding to a piece of Barcelona sewer network described by the discrete-time state equations

$$x_{k+1} = Ax_k + Bu_k + B_p d_k$$

where:

$$A = \begin{bmatrix} 1-\Delta t\beta_1 & 0 & 0 \\ 0 & 1 & 0 \\ \Delta t\beta_1 & 0 & 1-\Delta t\beta_3 \end{bmatrix}$$

$$B = \Delta t \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ -1 & -1 & 1 \end{bmatrix} \qquad B_p = \Delta t \begin{bmatrix} 0 & \alpha_2 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & \alpha_3 \end{bmatrix}$$



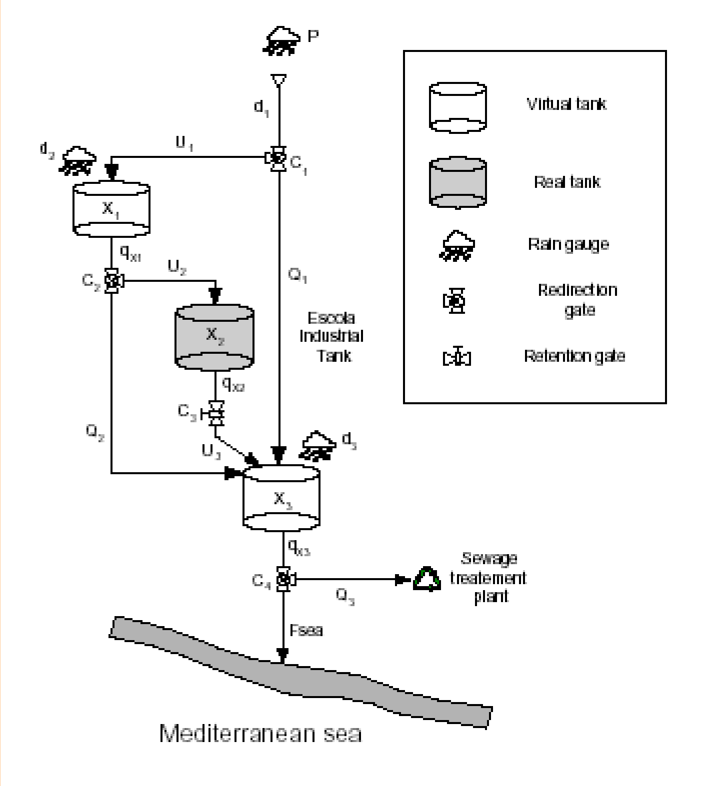
Application Example (2)

- The systems constraints are:
  - ***Bounding constraints***: refers to physical restrictions.

$$x_{1,k} \in [0,\infty] \qquad u_{1,k} \in [0,11]$$
$$x_{2,k} \in [0,35000] \qquad u_{2,k} \in [0,25]$$
$$x_{3,k} \in [0,\infty] \qquad u_{3,k} \in [0,7]$$

  - ***Mass conservation constraints:***

$$d_{1,k} = u_{1,k} + Q_1(k)$$
$$q_{x_1,k} = u_{2,k} + Q_2(k)$$
$$q_{x_2,k} \geq u_{3,k}$$



Reconfiguration Case

- This case considers actuators completely closed or completely open due to the fault, what would change the admissibility of the obtained actuator fault configurations.

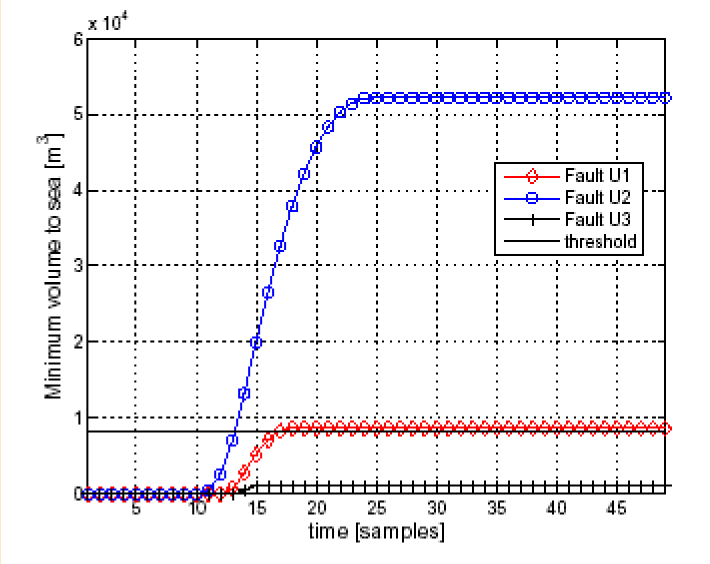


TABLE I

ADMISSIBILITY OF FAULT CONFIGURATIONS FOR POLLUTION - RECONFIGURATION

Fault Location	Min. Volume $[m^3]$	Admissibility Status
No fault	1050	—
Fault in u_1	8800	No Admissible
Fault in u_2	52200	No Admissible
Fault in u_3	1050	Admissible

ADMISSIBILITY CRITERIA: $V_{sea}^{f} \geq 8V_{sea}^{o}$

Accomodation Case

- This case considers that faults produces the reduction of the actuators operating range (for example from 0-100\% to 0-50\%).

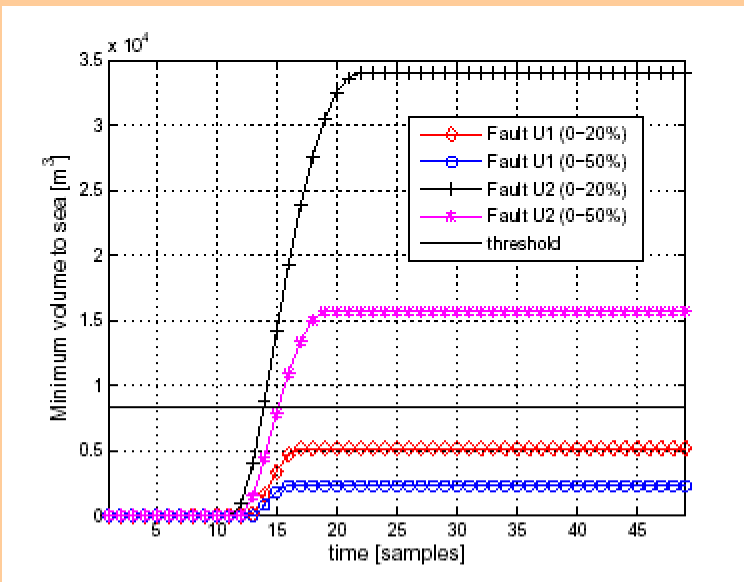


TABLE II

ADMISSIBILITY OF FAULT CONFIGURATIONS - ACCOMMODATION

Fault Location	Operation range	Min. Volume $[m^3]$	Admissibility Status
No fault	—	1050	—
Fault in U_1	0-20%	5200	Admissible
Fault in U_1	0-50%	2300	Admissible
Fault in U_2	0-20%	34000	No Admissible
Fault in U_2	0-50%	15700	No Admissible

ADMISSIBILITY CRITERIA: $V_{sea}^{f} \geq 8V_{sea}^{o}$

Index

1. Introduction
2. Interval Models for Fault Detection
3. Fault Detection using the Interval Observer Approach
4. Fault Detection using the Set-membership Approach
5. Identification for Robust Fault Detection
6. Fault-tolerance Evaluation
7. Real Applications
8. Conclusions
9. Further Research

Conclusions (1)

- This presentation has reviewed the use of set-membership methods in robust fault detection and isolation (FDI) and tolerant control (FTC).
- Alternatively to the statistical methods, set-membership methods use a deterministic unknown-but-bounded description of noise and parametric uncertainty (interval models).
- Using approximating sets to approximate the set of possible behaviours (in parameter or state space), these methods allows to check the consistency between observed and predicted behaviour.
- When an inconsistency is detected a fault can be indicated, otherwise nothing can be stated.

Conclusions (2)

- The same principle has been used to estimate interval models for fault detection and to develop methods for fault tolerance evaluation.

- Finally, same real application of these methods has been used to exemplify the successful uses in FDI/FTC.

Index

Further Research

- As further research, the set-membership approach could be extended to:

- ➢ extension to non-linear systems via the use of LPV models.
- ➢ deal with the fault isolation and estimation tasks exploiting the set arithmetic concepts
- ➢ adaptive thresholding in the the frequency domain
- ➢ better understand the links between the set-membership and interval approach revised in this presentation
- ➢ further extend their application to fault tolerant control as means to specify admissible closed loop behaviours.

Thank you very much
for your attention!!!