Fault Isolation and Estimation in Networks of Linear Process Systems

Wijaya Kurniawan [1,*,†], Katalin M. Hangos [1,2,†] and Lőrinc Márton [3,†]

1 Department of Electrical Engineering and Information Systems, University of Pannonia, 8200 Veszprem, Hungary; hangos@sztaki.hu
2 Systems and Control Laboratory, Institute for Computer Science and Control, 1111 Budapest, Hungary
3 Department of Electrical Engineering, Sapientia Hungarian University of Transylvania, 547367 Corunca, Romania; martonl@ms.sapientia.ro
* Correspondence: wijaya.kurniawan@mik.uni-pannon.hu
† These authors contributed equally to this work.

Abstract: Fault detection and isolation is a ubiquitous task in current complex systems even in the linear networked case when the complexity is mainly caused by the complex network structure. A simple yet practically important special case of networked linear process systems is considered in this paper with only a single conserved extensive quantity but with a network structure containing loops. These loops make fault detection and isolation challenging to perform because the effect of fault is propagated back to where it first occurred. As a dynamic model of network elements, a two input single output (2ISO) LTI state-space model is proposed for fault detection and isolation where the fault enters as an additive linear term into the equations. No simultaneously occurring faults are considered. A steady state analysis and superposition principle are used to analyse the effect of faults in a subsystem that propagates to the sensors' measurements at different positions. This analysis is the basis of our fault detection and isolation procedure that provides the position of the faulty element in a given loop of the network. A disturbance observer is also proposed to estimate the magnitude of the fault inspired by a proportional-integral (PI) observer. The proposed fault isolation and fault estimation methods have been verified and validated by using two simulation case studies in the MATLAB/Simulink environment.

1. Introduction

Nowadays, we live in a world that is surrounded by networks, e.g., computer networks, transportation networks, social networks, electrical networks, etc. Although they could comprise simple elements, their general large number and interconnections make them an important subclass of complex systems. As a dynamic system, a network presents many theoretical challenges during control or diagnosis method design. There have been many excellent surveys about this in systems and control literature (see, e.g., [1–7]).

Fault detection and isolation (FDI) is a subfield of control engineering which mainly deals with process monitoring to ensure the safety of production processes. The basic FDI notions are used similarly as presented in [8]. The *fault* is considered an unexpected variation of some process or environmental variable that could yield unacceptable changes in the process behaviour. During system model construction, *additive faults* are assumed and the faults are defined as fictive inputs. By *fault detection*, we mean a decision on the presence of any fault or the absence of all faults. Meanwhile, *fault isolation* is related to the localization of a fault. During *weak fault isolation*, the faults are assumed to happen one at a time. By *fault estimation*, we mean the method that can determine the steady state value of the additive faults based on the available measurements.

Although it is crucial to ensure safe operation in networks of process systems, the number of works related to FDI in networked systems is still limited [9].

There exist many different techniques that can be applied to solve an FDI problem. The important main categories comprise model based and data-driven methods [10–12]. Commonly, model-based methods use dynamic models derived from first principles that rely heavily on the knowledge of basic physical and/or chemical processes. Meanwhile, the current popular data-driven methods rely on data from system operations that are used to train machine learning (ML) or artificial intelligence (AI) data structures to detect and identify faults. However, even though data-driven methods could be easier to implement compared to model-based approaches, their reliability strictly depends on the availability and quality of the data [13].

The FDI problem in industrial networks, such as process networks, presents special challenges. For the sake of energy and/or material efficiency, such networks usually contain loops. Therefore, fault diagnosis is harder to perform as the effect of the fault is propagated back to the subsystem where it first occurred [14–18]. Recent research has approached the fault isolation problem in this loop using an improved deep neural network [19].

Another difficulty in the complex networked industrial systems related to FDI is the sparsity of sensors. Because of financial reasons, it is rare to install a multitude of sensors along the connections of the interconnected subsystems in the network. Commonly, they are placed at the end of a connection or at some of the subsystem's outputs. This motivates the development of an FDI method that groups network elements for fault identification [20,21].

As mentioned before, the interconnections that define the network topology make a networked dynamic system complex even if one considers simple elements (subsystems) in the network. Therefore, a practically important subclass of process systems is considered in this work where the elements are linear process systems connected by linear (possibly dynamic) connections.

Despite being a special case, it is a very important subclass because it can describe the dynamic behaviour of processes that serve our daily basic needs, e.g., domestic heating/cooling systems. Even some cellular processes belong to this subclass [22]. The fault that occurred in one of the subsystems is considered to be generated by an external source that can be treated as an extra input in a subsystem. Physically, this kind of fault represents some leakage phenomenon common inside networked linear process systems [23].

A model-based approach is proposed in this research to handle the fault diagnosis problem in a network of linear process systems, which may contain cycles/loops. The proposed approach does not need high computational costs. The burden of high computational power is common for fault diagnosis in a network [24].

The research questions that we are going to tackle are:

1. In a process network with loops, how can a fault be identified in a subsystem or group of subsystems regardless of the fault effect propagation through the network loops and branches?
2. Which sensor measurements are necessary to isolate a fault in a subsystem of a process network (or group of subsystems), and how can measurement noise be handled during fault diagnosis?
3. If fault isolation can be conducted, how can the magnitude of the fault be estimated?

In concordance with the addressed research questions, the main contributions of our proposed approach can be summarized as follows:

1. A fault diagnosis-oriented modelling approach is developed for such process networks in which the transport mechanisms can be described by stable and positive linear dynamic systems. The model includes the sources of the fault and indicates the sensor measurements. Based on this model, an analysis of how the fault input affects the steady states of the subsystems in the network is presented.
2. A model-based fault isolation approach is proposed for networks of linear process systems. The main benefit of this approach compared to previously reported ones is

that, under reasonable assumptions, it can isolate the faults in a subsystem or group of subsystems regardless of the loops and ramifications in the network. The proposed algorithm also indicates which sensor measurements are necessary to perform the fault isolation tasks. Moreover, it is also applicable in the presence of measurement noise.

3. Finally, to determine the magnitude of the localized fault, the design of a disturbance observer-based asymptotic fault estimator is proposed.

2. Networked Linear Process Systems

Linear process systems are systems where there are only linear source terms, such as chemical reactions and phase changes (evaporation, condensation etc.), present besides the usual transport terms that include convection, diffusion and transfer through phase boundaries [22]. Although this is a special case, linear process systems are of great practical importance, e.g., heat exchanger networks and domestic heating/cooling systems.

Most linear process systems have a complex structure. They comprise linear subsystems connected by a linear network that can provide static or dynamic connections between the subsystems. Such composite linear process systems are called networked linear process systems. This section describes the dynamic modelling of networked linear process systems where we consider that there are $j = 1 \ldots N$ subsystems.

2.1. Linear Process Subsystems

For fault isolation and estimation, one usually applies simplified dynamic models where the effect of the considered fault(s) is also described. These dynamic models are constructed from first engineering principles (see [22]) where the state equations originate from dynamic balance equations for the conserved extensive quantities (such as energy, overall mass, component masses) in a balanced volume. However, they will be transformed to their intensive variable forms using algebraic constitutive equations. Therefore, the state variables in linear process subsystems are considered to be intensive variables, e.g., temperature, concentration, etc.

Intensive variables usually spatially vary within a balanced volume so that the conservation balance equations related to them are in the form of partial differential equations (PDEs). To obtain simple dynamic models suitable for fault isolation and estimation, the lumped form of process models is used to approximate PDEs to bring them into a set of ordinary differential equations (ODEs) [22].

It is important to note that the dynamic linear interconnections in networked process systems can be realized by PDEs to model linear process subsystems where the lumping produces a finite-dimensional model representing the distributed delay phenomenon in process systems [25].

2.1.1. Modelling Assumptions

The following modelling assumptions are used in this paper:

A1 Only the transport of a single conserved extensive quantity (such as component mass, or energy) is considered in the process systems. Thus, we have either energy-transport or mass-transport systems. Heat exchanger networks and domestic heating/cooling systems belong to the linear energy-transport class. More details can be found in [26,27].

A2 Only linear convection and transfer is considered without any linear source.

A3 Constant overall mass and constant physicochemical parameters (such as density, specific heat, heat transfer coefficient, and convective flow rate) are assumed.

A4 One inlet and one outlet flow are considered where the inputs of the systems are the intensive variable (temperature or concentration) at the inlet and that of the balance volume with which transfer is assumed. Meanwhile, the output is the intensive variable (temperature or concentration) at the outlet.

2.1.2. Model Equations

Based on the above general assumptions and considering spatially homogeneous lumping (i.e., the same parameters in every lump), the dynamic model equations of a linear process subsystem becomes a two input–single output (2ISO) LTI state-space model which has the form:

$$S^{(j)} := \begin{cases} \dot{x}^{(j)} = A^{(j)} x^{(j)} + B^{(j)} u_X^{(j)} \\ y^{(j)} = C x^{(j)} \end{cases}$$

$$A^{(j)} = \begin{bmatrix} -(v^{(j)} + k_E^{(j)}) & 0 & 0 & \dots & \dots & 0 \\ v^{(j)} & -(v^{(j)} + k_E^{(j)}) & 0 & \dots & \dots & 0 \\ 0 & v^{(j)} & -(v^{(j)} + k_E^{(j)}) & \dots & \dots & 0 \\ 0 & \dots & & \dots & \dots & 0 \\ 0 & \dots & & \dots & \dots & v^{(j)} & -(v^{(j)} + k_E^{(j)}) \end{bmatrix} \tag{1}$$

$$B^{(j)} = \begin{bmatrix} v^{(j)} & 0 & \dots & 0 \\ k_E^{(j)} & k_E^{(j)} & \dots & k_E^{(j)} \end{bmatrix}^T , \quad C = \begin{bmatrix} 0 & 0 & 0 & \dots & 1 \end{bmatrix}$$

where $S^{(j)}$ is the general model of the jth subsystem with $x^{(j)}$ being the state variable, $y^{(j)}$ is the output variable, and $u_X^{(j)} = [u^{(j)} \ u_E^{(j)}]^T$ is the input variable which consists of $u^{(j)}$ as the intensive variable of the jth subsystem at the inlet and $u_E^{(j)}$ as the intensive variable of the external balance volume (environment). Meanwhile, $v^{(j)} > 0$ is the mass flow rate and $k_E^{(j)} > 0$ is the transfer coefficient.

By looking at Equation (1), we can derive some important points:

1. As long as $k_E^{(j)}$ is positive, the model will always be stable because the eigenvalues are already stable shown by the negative sign of the diagonal entries of the $A^{(j)}$ matrix (see Gershgorin circle theorem for details [28]).
2. The $A^{(j)} \in \mathbb{R}^{n \times n}$ matrix is a Metzler matrix by which all of its off-diagonal elements are non-negative. Thus, the matrix represents the time delayed differential equations and positive linear dynamical systems [29]. This is understandable because between the input and output states of the intensive variables in the subsystem, a propagation inside the transport element will occur contributing to the increase in delayed processes. This also implies the stability of the model because this matrix is sign stable and Hurwitz [30].
3. It is jointly controllable and observable because it is derived from conservation balance equations (it is a compartmental system) [31,32].

2.2. Interconnections and Topology of the Network

In the following, we consider process networks consisting of linear process subsystems described in Section 2.1 that are connected by static interconnections. In the case of dynamic interconnections, a special linear subsystem is used to represent the dynamics (i.e., the distributed delay) of the connection.

2.2.1. Physically Meaningful Connections

Considering the modelling assumptions and the model equations of the applied linear process subsystem models, we can make physically meaningful connections between two subsystems (the jth and ℓth, for example) by connecting part of the output flow of one (the jth) to the inlet flow of the other (the ℓth). This implies that the characterizing intensive variable $y^{(j)}$ will determine the intensive variable $u^{(\ell)}$ and the inlet of the ℓth subsystem.

2.2.2. Equations Describing the Connections

In a realistic network, it is common to have a branching phenomenon in the interconnection between subsystems. Here, the Kirchoff law applies to the flows of extensive

variables entering and exiting this interconnection. Considering the ℓth junction, the sum of either the overall mass, component mass, or energy flows entering the junction ($v^{(\ell)}$) is equal to the sum of flows exiting ($v^{(\ell)}_{OUT}$), as shown in the following equation for the case of overall mass flows:

$$\sum_{k \in IN^{(\ell)}} v^{(k)} = \sum_{l \in OUT^{(\ell)}} v^{(l)} = V^{(\ell)} \tag{2}$$

where k runs over the branches by which the flow enters the interconnection. Meanwhile, l runs over those where the flows exits the interconnection. $IN^{(\ell)}$ is the input set of the interconnection junction point, $OUT^{(\ell)}$ is the output set of the interconnection junction point, and v is measured in $[\frac{kg}{s}]$.

Similar conservation equations apply for the energy flows $\dot{Q}$ in energy-transport systems or the component mass m_X in mass-transport systems. However, to obtain the relations between the intensive variables (temperature T or concentration c_X, respectively) among the subsystems in this networked process systems, we substitute the algebraic equations representing the relationship of the intensive–extensive variables into the mass conservation results. These algebraic relationships are in the following general form:

$$Q = M c_P T \ , \ \ m_X = M c_X \tag{3}$$

where M is the overall mass and c_P is the specific heat in the balance volume.

Then, we can substitute relations in (3) into the Kirchhoff law in Equation (2) while taking into account that the value of the intensive variable for all the outflows is the same. This way, one obtains the linear algebraic equations for the intensive variables T or c_X at the ℓth junction in the following general form:

$$\sum_{k \in IN^{(\ell)}} \frac{v^{(k)}}{V^{(\ell)}} T^{(k)} = T^{(l)} \ \ \text{or} \ \ \sum_{k \in IN^{(\ell)}} \frac{v^{(k)}}{V^{(\ell)}} c_X^{(k)} = c_X^{(l)}$$

$$\sum_{k \in IN^{(\ell)}} \frac{v^{(k)}}{V^{(\ell)}} y^{(k)} = u^{(l)} \ , \ \ \forall \, l \in OUT^{(\ell)} \tag{4}$$

2.2.3. Network Topology

An underlying graph ($\mathcal{G}$) can be associated with the process network. The edges of the graph represent the dynamic subsystems of the network. Meanwhile, the vertices represent the interconnections among the subsystems which have no dynamics.

Example 1. *Consider a network composed of N subsystems. Figure 1 gives an example for this network of transport elements where $N = 6$ and $v^{(j)}$ is the mass flow rate of the jth subsystem.*

Based on Equation (2), we can derive some relations between $v^{(j)}$ in Figure 1 as follows:

$$\begin{aligned} &1. \ \ v^{(A)} = v^{(B)} + v^{(C)} \\ &2. \ \ v^{(B)} + v^{(C)} = v^{(D)} + v^{(E)} = V^{(1)} \\ &3. \ \ v^{(D)} + v^{(E)} = v^{(F)} \\ &4. \ \ v^{(F)} = v^{(A)} \end{aligned} \tag{5}$$

If we want to obtain some relations in terms of temperature (either $u^{(j)}$ or $y^{(j)}$) among the subsystems from the same Figure 1, we can substitute Equation (3) into each relation in Equation (5). Thus, we obtain:

1. $\dot{Q}^{(A)} = \dot{Q}^{(B)} + \dot{Q}^{(C)}$, so that $y^{(A)} = u^{(B)} = u^{(C)}$

2. $\dot{Q}^{(B)} + \dot{Q}^{(C)} = \dot{Q}^{(D)} + \dot{Q}^{(E)}$, so that $\dfrac{v^{(B)}}{V^{(1)}} y^{(B)} + \dfrac{v^{(C)}}{V^{(1)}} y^{(C)} = u^{(D)} = u^{(E)}$

3. $\dot{Q}^{(D)} + \dot{Q}^{(E)} = \dot{Q}^{(F)}$, so that $\dfrac{v^{(D)}}{v^{(F)}} y^{(D)} + \dfrac{v^{(E)}}{v^{(F)}} y^{(E)} = u^{(F)}$

4. $\dot{Q}^{(F)} = \dot{Q}^{(A)}$, so that $y^{(F)} = u^{(A)}$

$$(6)$$

Note that in the considered process network model, all the edges are part of at least one loop in the graph.
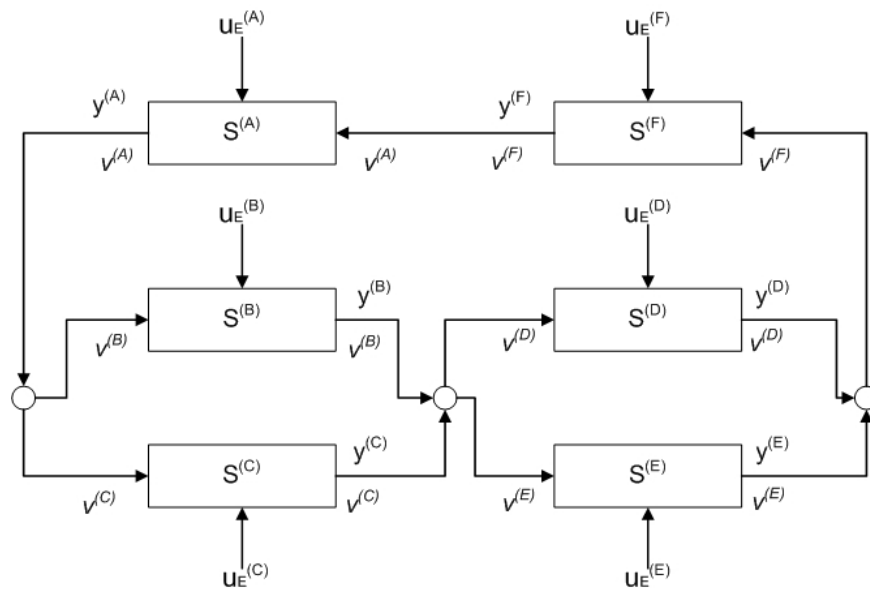

Figure 1. Example of a process network.

3. Faults in the Network

3.1. Fault Modelling

In this work, the considered fault is a constant input signal which additively modifies the external intensive variable input signal $u_E^{(j)}$. The faulty external variable input signal $u_{Ef}^{(j)}$ has the form:

$$u_{Ef}^{(j)} = u_E^{(j)} + f^{(j)} \tag{7}$$

where $f^{(j)}$ is the fault signal in the jth subsystem.

Such fault modelling can describe several fault events: the unforeseen appearance of an unknown external source, or change in the heat transfer coefficient $k_E^{(j)}$.

If there is a change $f_k^{(j)}$ in the heat transfer coefficient, i.e., $k_{Ef}^{(j)} = k_E^{(j)} + f_k^{(j)}$, then the second input of the model (1) can be rephrased as:

$$(k_E^{(j)} + f_k^{(j)}) u_E^{(j)} = k_E^{(j)} (u_E^{(j)} + f^{(j)}),$$
$$\text{where } f^{(j)} = \frac{f_k^{(j)} u_E^{(j)}}{k_E^{(j)}}. \tag{8}$$

During the fault diagnosis algorithm design, we assumed that:

A5 The probability of multiple fault events happening at the same time in the network is negligible (weak fault isolation), i.e., there are no simultaneous faults during the fault isolation and estimation processes.

3.2. The Fault's Effect in Different Measurement Positions in the Network

First, to investigate the fault's effect on the subsystems of the network, the state space model in Equation (1) is converted into input–output realization as follows:

$$
\begin{aligned}
y^{(j)}(s) &= S_1^{(j)}(s)u^{(j)}(s) + S_2^{(j)}(s)u_E^{(j)}(s) \\
S_1^{(j)}(s) &= \left(\frac{v^{(j)}}{s + v^{(j)} + k_E^{(j)}} \right)^n \\
S_2^{(j)}(s) &= \sum_{h=1}^{n} \left[\frac{(v^{(j)})^{(h-1)}}{(s + v^{(j)} + k_E^{(j)})^h} \right] k_E^{(j)}
\end{aligned}
\tag{9}
$$

where $S_1^{(j)}(s)$ is the transfer function in Laplace domain from $u^{(j)}(s)$ to $y^{(j)}(s)$ and $S_2^{(j)}(s)$ is the transfer function in Laplace domain from $u_E^{(j)}(s)$ to $y^{(j)}(s)$. Zero initial states are assumed.

Figure 2 shows the proposed realization for the fault effect analysis. For the sake of convenience, the notation $S_i^{(j)}$ will be used instead of $S_i^{(j)}(s)$ from here on.
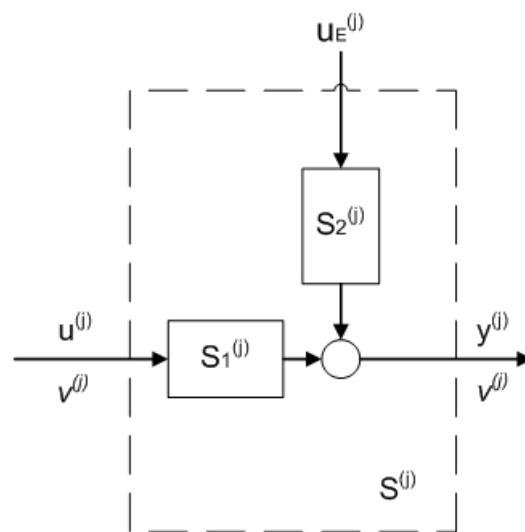

Figure 2. Input–output representation of a subsystem.

Now, consider a loop in the process network as shown in Figure 3. There, $S^{(j)}$ represents the block diagram of the jth subsystem which contains $S_1^{(j)}$ and $S_2^{(j)}$ as shown in Figure 2. Meanwhile, $i^{(j)}$, where $j = 1 \dots m$, represents the inflows of the subsystems that are not part of the loop (possible joining connections), and $o^{(j)}$, where $j = 1 \dots m$, represents the outflows from the loop in splitting connections.

It is considered that the sensors are placed at the outputs $y^{(l)}$ and $y^{(m)}$ where $m \geq 2$ and $0 < l < m$.

The fault $f^{(k)}$ is represented by a constant input in the kth subsystem, where $0 < k < l$, which enters the subsystem from the same channel as the external source $u_E^{(k)}$.
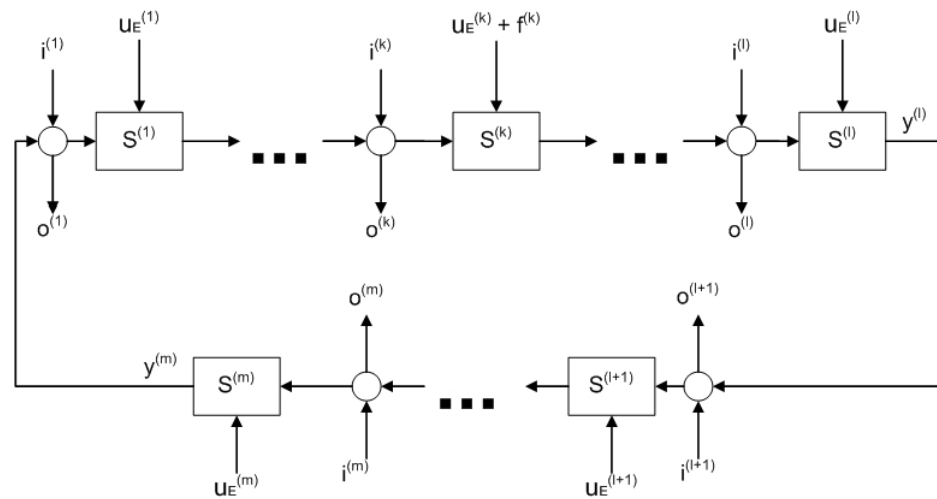
Figure 3. Diagram of a loop/cycle with fault in the process network.

For fault effect analysis, the final value theorem (FVT) of the Laplace transform is applied:

$$\lim_{t \to \infty} f(t) = \lim_{s \to 0} sF(s) \tag{10}$$

where $F(s)$ is the Laplace transform of $f(t)$.

To calculate the steady state value of the lth subsystem's output in the fault-free case $f^{(k)} = 0$ (see Figure 2), the FVT is applied to the previous transfer functions along with the superposition principle. By assuming that the inputs are step functions with zero initial conditions, it yields:

$$
\begin{aligned}
y_{ss}^{(l)} = \sum_{j=1}^{l} & \left[\frac{\|S_2^{(j)}\|_0 \prod_{h=j+1}^{l} \|S_1^{(h)}\|_0}{1 - \prod_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) u_E^{(j)} \\
+ \sum_{j=l+1}^{m} & \left[\frac{\|S_2^{(j)}\|_0 \prod_{h=j+1}^{m} \|S_1^{(h)}\|_0 \prod_{h=1}^{l} \|S_1^{(h)}\|_0}{1 - \prod_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) u_E^{(j)} \\
+ \sum_{j=1}^{l} & \left[\frac{\prod_{h=j}^{m} \|S_1^{(h)}\|_0}{1 - \prod_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) i_{ss}^{(j)} + \sum_{j=l+1}^{m} \left[\frac{\prod_{h=j}^{m} \|S_1^{(h)}\|_0 \prod_{h=1}^{l} \|S_1^{(h)}\|_0}{1 - \prod_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) i_{ss}^{(j)}
\end{aligned}
\tag{11}
$$

where $V^{(j)}$ is the sum of the mass flow rate passing through the jth joining/splitting connection before the jth subsystem input, $y_{ss}^{(l)}$ is the steady state value of $y^{(l)}$ when there is no fault, $i_{ss}^{(j)}$ is the steady state value of $i^{(j)}$, and $\| \cdot \|_0$ is the steady state gain of the related transfer function. $u_E^{(j)}$ is assumed to be constant.

Since the addressed subsystem class is positive (see Section 2.1), the steady state gains are also always positive.

Note that the terms of $\left(1 - \prod \|S_1^{(j)}\|_0 \right)$ appear because of the loop. Meanwhile, the terms of $\left(\frac{v^{(j)}}{V^{(j)}} \right)$ come from the mass/energy conservation balance (see Section 2.2).

When a step-like fault arises in the kth subsystem ($f^{(k)} \neq 0$), we obtain:

$$
\begin{aligned}
y_{fss}^{(l)} =\ & \left[\frac{\|S_2^{(k)}\|_0 \prod\limits_{j=k+1}^{l} \|S_1^{(j)}\|_0}{1 - \prod\limits_{j=1}^{m} \|S_1^{(j)}\|_0} \right] \left(\frac{v^{(k)}}{V^{(k)}} \right) f^{(k)} \\
&+ \sum_{j=1}^{l} \left[\frac{\|S_2^{(j)}\|_0 \prod\limits_{h=j+1}^{l} \|S_1^{(h)}\|_0}{1 - \prod\limits_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) u_E^{(j)} \\
&+ \sum_{j=l+1}^{m} \left[\frac{\|S_2^{(j)}\|_0 \prod\limits_{h=j+1}^{m} \|S_1^{(h)}\|_0 \prod\limits_{h=1}^{l} \|S_1^{(h)}\|_0}{1 - \prod\limits_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) u_E^{(j)} \\
&+ \sum_{j=1}^{l} \left[\frac{\prod\limits_{h=j}^{m} \|S_1^{(h)}\|_0}{1 - \prod\limits_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) i_{ss}^{(j)} + \sum_{j=l+1}^{m} \left[\frac{\prod\limits_{h=j}^{m} \|S_1^{(h)}\|_0 \prod\limits_{h=1}^{l} \|S_1^{(h)}\|_0}{1 - \prod\limits_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) i_{ss}^{(j)}
\end{aligned}
\tag{12}
$$

where $y_{fss}^{(j)}$ is the steady state value of $y^{(j)}$ in the presence of a fault.

Now, by subtracting Equation (11) from Equation (12), we obtain the deviation of the faulty output related to the fault-free case:

$$
y_{fss}^{(l)} - y_{ss}^{(l)} = \left[\frac{\|S_2^{(k)}\|_0 \prod\limits_{j=k+1}^{l} \|S_1^{(j)}\|_0}{1 - \prod\limits_{j=1}^{m} \|S_1^{(j)}\|_0} \right] \left(\frac{v^{(k)}}{V^{(k)}} \right) f^{(k)}
\tag{13}
$$

The difference between the faulty and fault-free outputs of the mth subsystem can be computed similarly:

$$
y_{fss}^{(m)} - y_{ss}^{(m)} = \left[\frac{\|S_2^{(k)}\|_0 \prod\limits_{j=k+1}^{m} \|S_1^{(j)}\|_0}{1 - \prod\limits_{j=1}^{m} \|S_1^{(j)}\|_0} \right] \left(\frac{v^{(k)}}{V^{(k)}} \right) f^{(k)}
\tag{14}
$$

Equations (13) and (14) show that the fault influences all subsystems in the loop. However, the fault effect on the outputs of the subsystems is calculable.

Example 2. *Figure 4 presents two connected subsystems which form a loop. The input–output models of the subsystems are $S^{(P)}$ and $S^{(Q)}$, respectively, where each contains $S_1^{(P)}$, $S_2^{(P)}$, $S_1^{(Q)}$, and $S_2^{(Q)}$ as shown in Figure 2.*

No joining and splitting connections are assumed ($i^{(j)} = o^{(j)} = 0$), so $v^{(P)} = v^{(Q)} = V$ which leads to $\frac{v^{(j)}}{V^{(j)}} = 1$ where $j = P, Q$. Moreover, $y^{(P)} = u^{(Q)}$ and $y^{(Q)} = u^{(P)}$.

From Equations (13) and (14), the step-like fault $f^{(P)} \neq 0$ generates deviations in the subsystems as follows:

$$
\begin{aligned}
y_{fss}^{(P)} - y_{ss}^{(P)} &= \left(\frac{\|S_2^{(P)}\|_0}{1 - \|S_1^{(P)}\|_0 \|S_1^{(Q)}\|_0} \right) f^{(P)} \\
y_{fss}^{(Q)} - y_{ss}^{(Q)} &= \left(\frac{\|S_2^{(P)}\|_0 \|S_1^{(Q)}\|_0}{1 - \|S_1^{(P)}\|_0 \|S_1^{(Q)}\|_0} \right) f^{(P)}
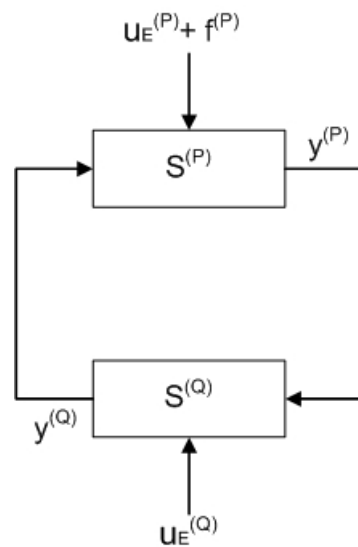\end{aligned}
\tag{15}
$$

$$U_E^{(P)} + f^{(P)}$$

Figure 4. Diagram of two subsystems in a loop with a fault.

4. Fault Diagnosis

4.1. Problem Formulation

Given a process network as defined in Section 2.2, consider that a fault event arises in one of the subsystem in the network. As a consequence of the branches and loops in the underlying graph ($\mathcal{G}$) of the process network, the effect of this fault could induce deviations in the outputs of all the subsystems. Moreover, due to the loops, the fault could propagate back to the input of the faulty subsystem as well. The loops and network branches make the fault effect propagation barely traceable.

The presence of a fault in the network can simply be detected by comparing the measured outputs of the subsystems in the network with the predicted outputs based on a reliable model of the corresponding subsystem. However, due to fault effect propagation, the localization of the fault source in the network is a difficult task.

Recall that the subsystems represent the edges in the underlying graph $\mathcal{G}$ of the process network. We consider a directed path of subsystems $\left\{ S^{(1)} \ldots S^{(l)} \right\}$ where $l > 0$ such that this path may be part of at least one simple loop of $\mathcal{G}$.

Let a fault event happen in a subsystem in the network. Formulate the following fault diagnosis problems:

- Consider a path of l subsystems that can be part of a loop consisting of $m > l$ subsystems (see Figure 3). We must determine whether the fault occurred in the addressed path. Furthermore, we must determine which measurements are necessary to perform the isolation problem (sensors placement) in this path.
- If the fault has been isolated in one subsystem, an estimation algorithm must be designed that outputs the magnitude of the fault.

Note that the path can be part of more than one loop in the graph of the process network, e.g., in Figure 1, the path $\left\{ S^{(A)}, S^{(F)} \right\}$ is in more than one simple loop. For an algorithm that finds all the simple loops in a graph, see, e.g., [33]. In this case, such a loop should be chosen that is more representative for the fault isolation process, e.g., from the perspective of the sensor placement.

The formulated diagnosis problem can specify whether the fault happened in a group of subsystems (or one subsystem for $l = 1$), or in some other part of the network. However, the diagnosis process can be repeated for different groups (or subsystems). Thus, by exhaustive search, the fault in the network can be localized.

The fault isolation algorithm is designed by applying the fault effect analysis as presented in Section 3.2. For fault estimator design, we take the PI observer approach.

4.2. Fault Isolation

To derive a fault isolation algorithm, we consider that two sensors are placed in a loop at two different locations (see Figure 3). They measure the output of the lth and mth subsystems.

The isolation logic is based on Equations (13) and (14). By subtracting these equations, we obtain:

$$(y_{fss}^{(l)} - y_{ss}^{(l)}) - (y_{fss}^{(m)} - y_{ss}^{(m)}) = \left[\left(\frac{\|S_2^{(k)}\|_0 \prod\limits_{j=k+1}^{l} \|S_1^{(j)}\|_0}{1 - \prod\limits_{j=1}^{m} \|S_1^{(j)}\|_0} \right) \left(1 - \prod\limits_{j=l+1}^{m} \|S_1^{(j)})\|_0 \right) \right] \left(\frac{v^{(k)}}{V^{(k)}} \right) f^{(k)} \quad (16)$$

We make the following assumptions on the steady state gains of the subsystems:

A6 Either every $\|S_1^{(j)}\|_0 \in (0,1)$ or every $\|S_1^{(j)}\|_0 > 1$;

A7 $\|S_2^{(j)}\|_0 > 0 \ \forall j$.

In the view of Equation (16), if these assumptions hold, then

$$\left(\frac{\|S_2^{(k)}\|_0 \prod\limits_{j=k+1}^{l} \|S_1^{(j)}\|_0}{1 - \prod\limits_{j=1}^{m} \|S_1^{(j)}\|_0} \right) \left(1 - \prod\limits_{j=l+1}^{m} \|S_1^{(j)})\|_0 \right) > 0 \quad (17)$$

Note that A6 and A7 hold true for the subsystem's model introduced in Section 2.1; if we apply the FVT to Equation (9), we obtain:

$$\|S_1^{(j)}\|_0 = \lim_{s \to 0} S_1^{(j)}(s) = \left(\frac{v^{(j)}}{v^{(j)} + k_E^{(j)}} \right)^n \in (0,1)$$

$$\|S_2^{(j)}\|_0 = \lim_{s \to 0} S_2^{(j)}(s) = \sum_{h=1}^{n} \left[\frac{(v^{(j)})^{(h-1)}}{(v^{(j)} + k_E^{(j)})^h} \right] k_E^{(j)} > 0 \quad (18)$$

Furthermore, by assumptions A6 and A7, Equation (16) leads to:

$$(y_{fss}^{(l)} - y_{ss}^{(l)}) > (y_{fss}^{(m)} - y_{ss}^{(m)}) \text{ for } f^{(k)} > 0 \text{ and}$$
$$(y_{fss}^{(l)} - y_{ss}^{(l)}) < (y_{fss}^{(m)} - y_{ss}^{(m)}) \text{ for } f^{(k)} < 0 \quad (19)$$
$$\therefore \|y_{fss}^{(l)} - y_{ss}^{(l)}\| - \|y_{fss}^{(m)} - y_{ss}^{(m)}\| > 0 \text{ for } f^{(k)} \neq 0$$

where $\| \cdot \|$ is the absolute value of the related function.

Thus, Equation (19) shows that $\|y_{fss}^{(l)} - y_{ss}^{(l)}\| - \|y_{fss}^{(m)} - y_{ss}^{(m)}\| > 0$ when a fault occurs in a subsystem between $S^{(1)}$ and $S^{(l)}$.

With the same assumptions, we can perform the same derivation to obtain $\|y_{fss}^{(l)} - y_{ss}^{(l)}\| - \|y_{fss}^{(m)} - y_{ss}^{(m)}\| < 0$ when a fault occurs in a subsystem between $S^{(l+1)}$ and $S^{(m)}$.

For implementation of the fault isolation algorithm, the fault-free steady state value $y_{ss}^{(l)}$ and $y_{ss}^{(m)}$ have to be known prior, or they have to be computed. In view of the relation in (11), to compute $y_{ss}^{(l)}$ and $y_{ss}^{(m)}$, the steady state value of the inputs $i_{ss}^{(j)}$ and $u_E^{(j)}$ have to be measured.

To conclude, the fault isolation can be performed according to Algorithm 1 as follows:

Algorithm 1 Fault isolation algorithm.

- Measure $y_{fss}^{(l)}$ and $y_{fss}^{(m)}$ in steady state.
- Compute $y_{ss}^{(l)}$ and $y_{ss}^{(m)}$.
- Isolate the fault:
    - If $\|y_{fss}^{(l)} - y_{ss}^{(l)}\| = \|y_{fss}^{(m)} - y_{ss}^{(m)}\| = 0$, then no fault event occurred.
    - If $\|y_{fss}^{(l)} - y_{ss}^{(l)}\| - \|y_{fss}^{(m)} - y_{ss}^{(m)}\| > 0$, then the fault occurred before l and after m.
    - If $\|y_{fss}^{(l)} - y_{ss}^{(l)}\| - \|y_{fss}^{(m)} - y_{ss}^{(m)}\| < 0$, then the fault occurred before m and after l.

4.3. Fault Isolation in the Presence of Measurement Noise

In a realistic environment, it has to be considered that the measurements on the subsystems are affected by signal noise. In model (1), the noise inputs that influence the model's inputs and outputs are introduced as:

$$\begin{cases} \dot{x}^{(j)} = A^{(j)} x^{(j)} + B^{(j)} \left(u_X^{(j)} + w_{uX}^{(j)} \right) \\ y^{(j)} = C x^{(j)} + w_y^{(j)} \end{cases} \tag{20}$$

where $w_{uX}^{(j)} = [w_u^{(j)} \; w_E^{(j)}]^T$.

The following assumptions are considered for signal noise:

A8　$\|w_u^{(j)}(t)\|_\infty \le w_{uM}^{(j)}$, $\|w_E^{(j)}(t)\|_\infty \le w_{EM}^{(j)}$, $\|w_y^{(j)}(t)\|_\infty \le w_{yM}^{(j)}$

　　where $\|\cdot\|_\infty$ is the infinity norm of the related function.

A9　$w_u^{(j)}(t)$, $w_E^{(j)}(t)$, $w_y^{(j)}(t)$ are locally integrable.

A10　The signal noise is unbiased with known variances:

　　$w_E^{(j)}(t) \sim (0, R_u^{(j)})$, $w_y^{(j)}(t) \sim (0, Q_y^{(j)})$.

A11　$w_u^{(j)}(t)$, $w_E^{(j)}(t)$, $w_y^{(j)}(t)$ are mutually uncorrelated with both each other and system states.

The input–output model in (9) in the presence of newly considered signal noise takes the form:

$$y_n^{(j)}(s) = S_1^{(j)}(s) \left[u^{(j)}(s) + w_u^{(j)}(s) \right] + S_2^{(j)}(s) \left[u_E^{(j)}(s) + w_E^{(j)}(s) \right] + w_y^{(j)}(s)$$

$$y_n^{(j)}(s) = y^{(j)}(s) + \left[S_1^{(j)}(s) w_u^{(j)}(s) + S_2^{(j)}(s) w_E^{(j)}(s) + w_y^{(j)}(s) \right] \tag{21}$$

$$y_n^{(j)}(s) = y^{(j)}(s) + w^{(j)}(s).$$

where $y_n^{(j)}$ is the subsystem's output in the presence of noise and $w^{(j)}$ is the noise's effect where

$$\|w^{(j)}(t)\|_\infty \le w_M^{(j)}. \tag{22}$$

As the noise from each subsystem propagates inside the loop, the measurement at both points l and m in the network are affected. Thus, to compensate for noise in the fault isolation process, a suitable threshold needs to be specified.

By applying the FVT and superposition principle, the noise's effect on the measurements at sensors l and m can be computed, obtaining:

$$y_{fnss}^{(l)} = y_{fss}^{(l)} + \sum_{j=1}^{l} \left[\frac{\prod_{h=j+1}^{l} \|S_1^{(h)}\|_0}{1 - \prod_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) w^{(j)} + \sum_{j=l+1}^{m} \left[\frac{\prod_{h=j+1}^{m} \|S_1^{(h)}\|_0 \prod_{h=1}^{l} \|S_1^{(h)}\|_0}{1 - \prod_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) w^{(j)}$$

$$y_{fnss}^{(m)} = y_{fss}^{(m)} + \sum_{j=1}^{m-1} \left[\frac{\prod_{h=j+1}^{m} \|S_1^{(h)}\|_0}{1 - \prod_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) w^{(j)} + \left[\frac{\prod_{h=1}^{m} \|S_1^{(h)}\|_0}{1 - \prod_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(m)}}{V^{(m)}} \right) w^{(m)} \tag{23}$$

where $y_{fnss}^{(j)}$ is the steady state value of $y^{(j)}$ in the presence of a fault and noise, $w^{(l)}$ is the noise's effect on the measurement at sensor l, and $w^{(m)}$ is the noise's effect on the measurement at sensor m.

With the addition of the noise's effect, Equation (16) becomes:

$$
(y_{fnss}^{(l)} - y_{ss}^{(l)}) - (y_{fnss}^{(m)} - y_{ss}^{(m)}) = \left[\left(\frac{\|S_2^{(k)}\|_0 \prod\limits_{j=k+1}^{l} \|S_1^{(j)}\|_0}{1 - \prod\limits_{j=1}^{m} \|S_1^{(j)}\|_0} \right) \left(1 - \prod\limits_{j=l+1}^{m} \|S_1^{(j)})\|_0 \right) \right] \left(\frac{v^{(k)}}{V^{(k)}} \right) f^{(k)}
$$

$$
+ \sum_{j=1}^{l} \underbrace{\left[\frac{\prod\limits_{h=j+1}^{l} \|S_1^{(h)}\|_0 \left(1 - \prod\limits_{h=l+1}^{m} \|S_1^{(h)}\|_0 \right)}{1 - \prod\limits_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) w^{(j)}}_{S_{wl}^{(j)}}
$$

$$
+ \sum_{j=l+1}^{m} \underbrace{\left[\frac{\prod\limits_{h=j+1}^{m} \|S_1^{(h)}\|_0 \left(\prod\limits_{h=1}^{l} \|S_1^{(h)}\|_0 - 1 \right)}{1 - \prod\limits_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) w^{(j)}}_{S_{wm}^{(j)}} \tag{24}
$$

Now, for measurements at points l and m, the threshold value $th^{(lm)}$ is defined as:

$$
th^{(lm)} = \sum_{j=1}^{l} \|S_{wl}^{(j)}\|_\infty w_M^{(j)} + \sum_{j=l+1}^{m} \|S_{wm}^{(j)}\|_\infty w_M^{(j)} \tag{25}
$$

Then, by considering the measurement noise, Algorithm 1 is updated into Algorithm 2 as follows:

Algorithm 2 Fault isolation algorithm in the presence of measurement noise.

- Measure $y_{fnss}^{(l)}$ and $y_{fnss}^{(m)}$ in steady state.
- Compute $y_{ss}^{(l)}$, $y_{ss}^{(m)}$, and $th^{(lm)}$.
- Isolate the fault:
  - If $\|y_{fnss}^{(l)} - y_{ss}^{(l)}\| - \|y_{fnss}^{(m)} - y_{ss}^{(m)}\| \in (-th^{(lm)}, th^{(lm)})$, then no fault event occurred or the fault is negligible in comparison to the threshold.
  - If $\|y_{fnss}^{(l)} - y_{ss}^{(l)}\| - \|y_{fnss}^{(m)} - y_{ss}^{(m)}\| > th^{(lm)}$, then the fault occurred before l and after m.
  - If $\|y_{fnss}^{(l)} - y_{ss}^{(l)}\| - \|y_{fnss}^{(m)} - y_{ss}^{(m)}\| < -th^{(lm)}$, then the fault occurred before m and after l.

Example 3. *This example is an extension of Example 2.*

Consider that in the loop shown in Figure 4, the subsystem's sensors are affected by measurement noises $w^{(P)}$ and $w^{(Q)}$, where $\|w^{(P)}(t)\|_\infty < w_M^{(P)}$ and $\|w^{(Q)}(t)\|_\infty < w_M^{(Q)}$.

Then, using Equations (24) and (25), the threshold value can be computed as follows:

$$
th^{(PQ)} = \left\| \frac{1 - \|S_1^{(Q)}\|_0}{1 - \|S_1^{(P)}\|_0 \|S_1^{(Q)}\|_0} \right\|_\infty w_M^{(P)} + \left\| \frac{\|S_1^{(P)}\|_0 - 1}{1 - \|S_1^{(P)}\|_0 \|S_1^{(Q)}\|_0} \right\|_\infty w_M^{(Q)} \tag{26}
$$

4.4. Fault Estimation

After the fault has been isolated, under certain sensor placement assumptions, the fault estimation can be performed. A linear disturbance–observer approach is proposed to determine the magnitude of the fault given by Equation (7).

During the estimation process, consider $k = l = 1$ and the fault has been isolated in subsystem $k = 1$.

The state space model given by Equation (1) with fault (7) can be rewritten as follows:

$$
\begin{aligned}
\dot{x}^{(k)} &= A^{(k)}x^{(k)} + B^{(k)}u_X^{(k)} + E^{(k)}f^{(k)} \\
y^{(k)} &= Cx^{(k)} \\
E^{(k)} &= \begin{bmatrix} k_E^{(k)} & k_E^{(k)} & \dots & k_E^{(k)} & \dots & k_E^{(k)} & k_E^{(k)} \end{bmatrix}^T
\end{aligned}
\tag{27}
$$

where $E^{(k)}$ is a fault distribution column matrix and $f^{(k)}$ is assumed to be constant.

This new model in Equation (27) can be further transformed into an extended state space model by defining an extended state vector containing the fault as $z^{(k)} = [x^{(k)}\ f^{(k)}]^T$:

$$
\begin{aligned}
\dot{z}^{(k)} &= \begin{bmatrix} A^{(k)} & E^{(k)} \\ 0 & 0 \end{bmatrix}\begin{bmatrix} x^{(k)} \\ f^{(k)} \end{bmatrix} + \begin{bmatrix} B^{(k)} \\ 0 \end{bmatrix}u_X^{(k)} = A_z^{(k)}z^{(k)} + B_z^{(k)}u_X^{(k)} \\
y^{(k)} &= \begin{bmatrix} C & 0 \end{bmatrix}\begin{bmatrix} x^{(k)} \\ f^{(k)} \end{bmatrix} = C_z z^{(k)}
\end{aligned}
\tag{28}
$$

As stated by [34], if the steady state value of the fault is not zero, a proportional observer cannot correctly estimate the states of the plant because there will always be a steady state error between the actual and estimated states. However, using a proportional-integral (PI) observer, the steady state error can be reduced.

The state space model of a PI observer is:

$$
\begin{aligned}
\dot{\hat{x}}^{(k)} &= A^{(k)}\hat{x}^{(k)} + B^{(k)}u_X^{(k)} + L_P^{(k)}(y^{(k)} - \hat{y}^{(k)}) + E^{(k)}\hat{f}^{(k)} \\
\dot{\hat{f}}^{(k)} &= L_I^{(k)}(y^{(k)} - \hat{y}^{(k)}) \\
\hat{y}^{(k)} &= C\hat{x}^{(k)}
\end{aligned}
\tag{29}
$$

where $\hat{x}^{(k)}$ is the estimated state vector, $\hat{f}^{(k)}$ is the estimated fault magnitude, $L_P^{(k)}$ is the observer's proportional gain, and $L_I^{(k)}$ is the observer's integral gain.

Hence, a PI observer can be designed for the extended state space model in Equation (28) to not only estimate the states but also the fault. By applying Equation (29) into Equation (28) with the assumption that $(A_z^{(k)}, C_z)$ is an observable pair, we obtain:

$$
\begin{aligned}
\dot{\hat{z}}^{(k)} &= \left(\begin{bmatrix} A^{(k)} & E^{(k)} \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} L_P^{(k)} \\ L_I^{(k)} \end{bmatrix}\begin{bmatrix} C & 0 \end{bmatrix} \right)\hat{z}^{(k)} + \begin{bmatrix} L_P^{(k)} \\ L_I^{(k)} \end{bmatrix}y^{(k)} + \begin{bmatrix} B^{(k)} \\ 0 \end{bmatrix}u_X^{(k)} \\
&= (A_z^{(k)} - L_z^{(k)}C_z)\hat{z}^{(k)} + L_z^{(k)}y^{(k)} + B_z^{(k)}u_X^{(k)} \\
\hat{y}^{(k)} &= C_z\hat{z}^{(k)}
\end{aligned}
\tag{30}
$$

Thus, if $L_P^{(k)}$ and $L_I^{(k)}$ are chosen such that $(A_z^{(k)} - L_z^{(k)}C_z)$ is Hurwitz, then $\lim_{t\to\infty}(z^{(k)} - \hat{z}^{(k)}) = 0$.

Meanwhile, in the presence of noise, the widely known linear quadratic estimator (LQE) procedure can be used to compute the observer's gain $L_z^{(k)}$. For details, see Appendix A.

5. Case Studies

To verify and validate the proposed fault isolation and fault estimation methods, two simulation case studies were examined in the MATLAB/Simulink environment. The first case study is based on Example 2. Meanwhile, for the second case study, the investigated network comprises six subsystems as presented in Example 1.

5.1. Case Study 1

In this case study, two identical subsystems $S^{(P)}$ and $S^{(Q)}$ are connected as shown in Figure 4. The model of the subsystems is given by Equation (20) where each subsystem is considered to have five state variables ($n = 5$). The applied parameters and external inputs are shown in Table 1.

Table 1. Case study 1—parameters and external inputs.

j	P	Q
$v^{(j)}$	10	10
$k_E^{(j)}$	3	3
$u_E^{(j)}$	400	400
$R_u^{(j)}$	20	20
$Q_y^{(j)}$	20	20

It is considered that the measurements $y^{(P)}$ and $y^{(Q)}$ are influenced by noise as presented in (20). To compensate for the noise that influences the system, a suitable threshold $th^{(PQ)}$ is computed by applying Equation (25). From the proposed fault isolation logic (see Algorithm 2), as the loop is composed of two subsystems $S^{(P)}$ and $S^{(P)}$, the simulation results should show that $\|y_{fnss}^{(P)} - y_{ss}^{(P)}\| - \|y_{fnss}^{(Q)} - y_{ss}^{(Q)}\|$ is above $th^{(PQ)}$ when there is a fault in subsystem $S^{(P)}$, or $\|y_{fnss}^{(P)} - y_{ss}^{(P)}\| - \|y_{fnss}^{(Q)} - y_{ss}^{(Q)}\|$ is below $-th^{(PQ)}$ when there is a fault in subsystem $S^{(Q)}$.

In the first scenario, no fault is injected into the subsystems. The simulation results in Figure 5 show $\|y_{fnss}^{(P)} - y_{ss}^{(P)}\| - \|y_{fnss}^{(Q)} - y_{ss}^{(Q)}\| \in (-th^{(PQ)}, th^{(PQ)})$, i.e., no fault is injected.
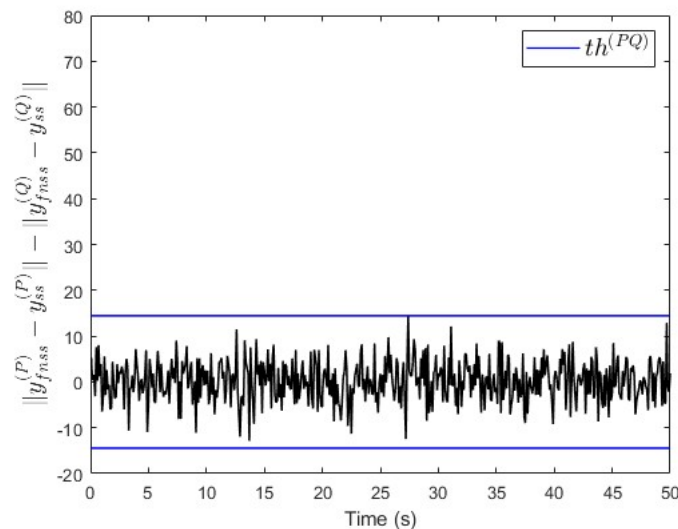
Figure 5. Case study 1—fault free case.

In the second scenario, a constant fault signal $f^{(P)} = 100$ is injected into subsystem $S^{(P)}$ at $t = 20$ s. Figure 6 shows that the value of $\|y_{fnss}^{(P)} - y_{ss}^{(P)}\| - \|y_{fnss}^{(Q)} - y_{ss}^{(Q)}\|$ is above the threshold after the fault event occurred. This indicates that a fault is occurring in subsystem $S^{(P)}$.

To verify the proposed fault estimation approach, an observer is designed specifically for subsystem $S^{(P)}$ based on Equation (30). By using the `lqe` MATLAB function (see Appendix A), the observer's gain is computed such that the measurement noise variances are taken into consideration. Figure 7 shows the fault and its estimated value. It is seen that the fault's magnitude is correctly estimated. Meanwhile, Figure 8 shows that the observer successfully estimates the states of subsystem $S^{(P)}$.
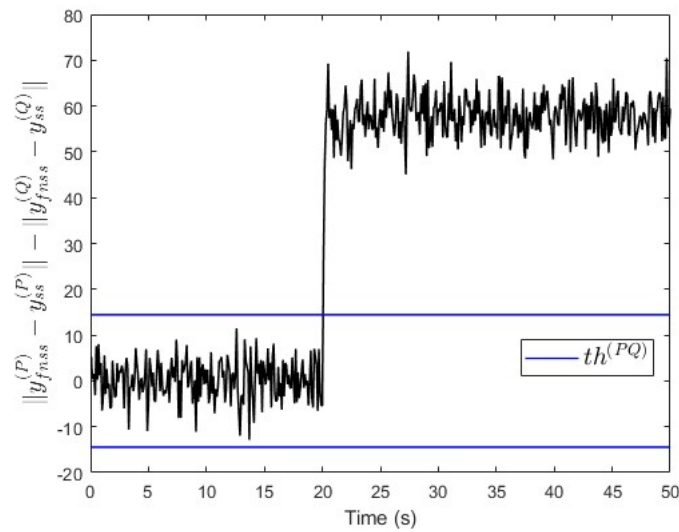
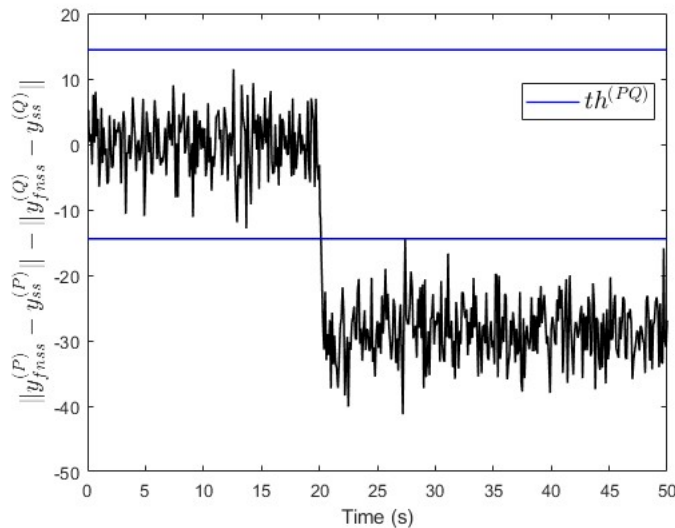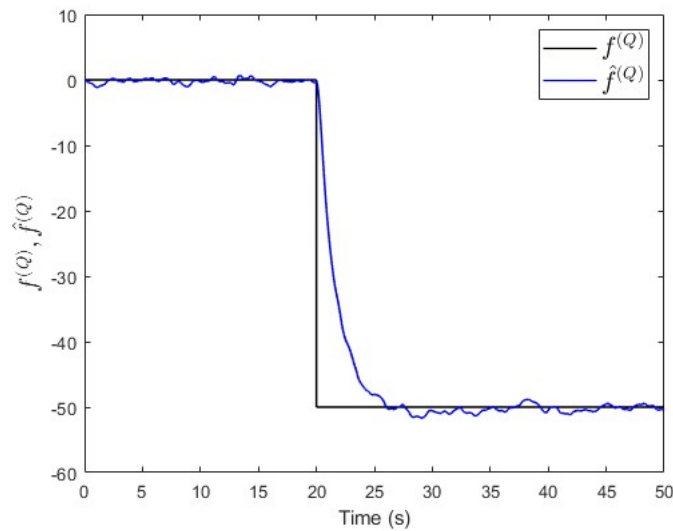**Figure 6.** Case study 1—fault isolation in subsystem $S^{(P)}$.
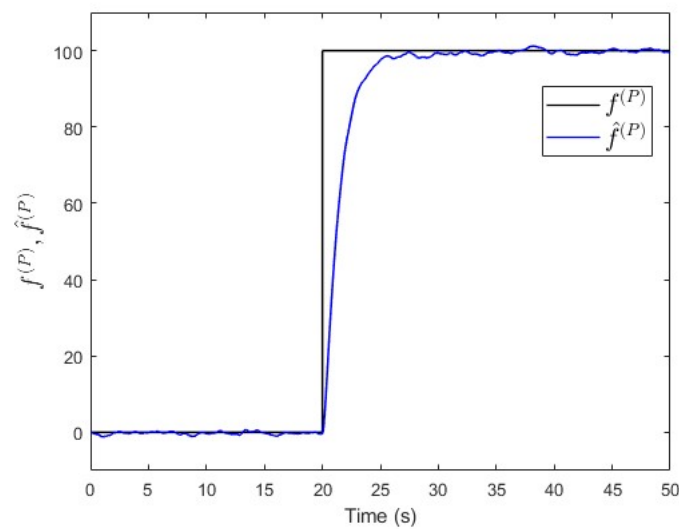
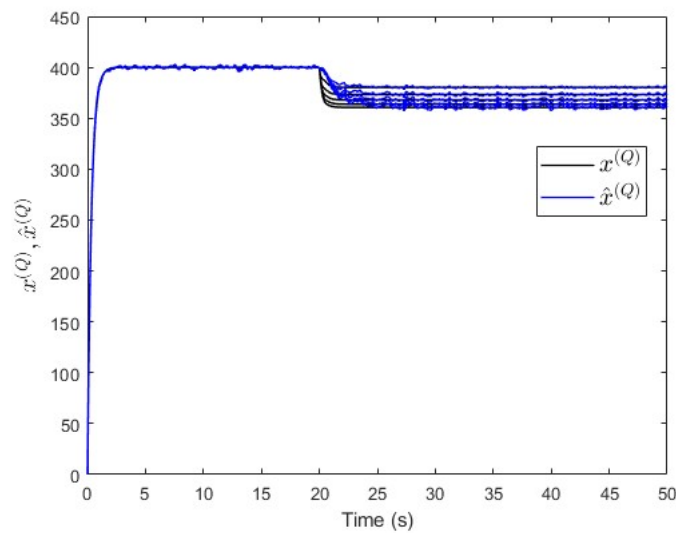**Figure 7.** Case study 1—fault estimation in subsystem $S^{(P)}$.
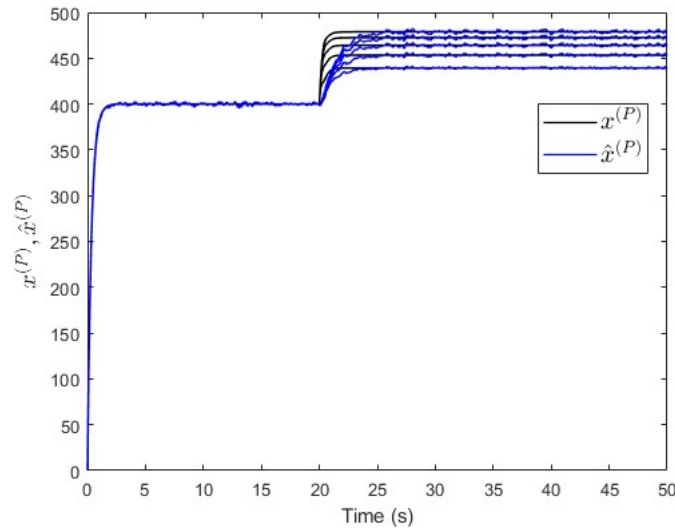
Figure 8. Case study 1—states estimation in subsystem $S^{(P)}$.

In the third scenario, a constant fault signal $f^{(Q)} = -50$ is injected into subsystem $S^{(Q)}$ at $t = 20$ s. Figure 9 shows that the value of $\|y_{fnss}^{(P)} - y_{ss}^{(P)}\| - \|y_{fnss}^{(Q)} - y_{ss}^{(Q)}\|$ is below $-th^{(PQ)}$ after the fault event occurred. This indicates that a fault is occurring in subsystem $S^{(Q)}$, which is correct.

Moreover, an observer is designed specifically for subsystem $S^{(Q)}$ based on Equation (30). The observer's gain is also computed using the same `lqe` MATLAB function such that the noise variances are taken into consideration. Figure 10 shows the fault and its estimated value. Here, the fault's magnitude is also correctly estimated. Meanwhile, Figure 11 shows that the observer successfully estimates the states of subsystem $S^{(Q)}$.

Figure 9. Case study 1—fault isolation in subsystem $S^{(Q)}$.

Figure 10. Case study 1—fault estimation in subsystem $S^{(Q)}$.

Figure 11. Case study 1—states estimation in subsystem $S^{(Q)}$.

5.2. Robustness Analysis (for Case Study 1)

To analyse the robustness of our proposed approach, modified parameter values are considered in the subsystem model (1) only during the simulations. Here, we have two parameters to modify: the mass flow rate and the transfer coefficient. The mass flow rate $v^{(j)}$ can be easily measured. However, the transfer coefficient $k_E^{(j)}$ can only be estimated as its value changes depending on the physical conditions and circumstances. Thus, the robustness analysis is performed by checking the performance of fault isolation and estimation in the presence of transfer coefficient parameter uncertainty.

By using case study 1 for the sake of convenience but without loss of generality, the simulation is carried out by increasing the value of $k_E^{(j)}$ in Table 1 by 25% ($k_E^{(j)} = 3.75$ where $j = P, Q$). After this, a constant fault signal $f^{(Q)} = -50$ is injected into subsystem $S^{(Q)}$ at $t = 20$ s. Figure 12 shows that the value of $\|y_{fnss}^{(P)} - y_{ss}^{(P)}\| - \|y_{fnss}^{(Q)} - y_{ss}^{(Q)}\|$ is below $-th^{(PQ)}$ after the fault event occurred. This indicates that a fault is occurring in subsystem $S^{(Q)}$, which is correct. The real and estimated fault values are shown in Figure 13. There, the fault's magnitude is estimated with an error of less than 10%. This error is acceptable provided that the maximum tolerable value of the uncertainty in $k_E^{(j)}$ is 25%.
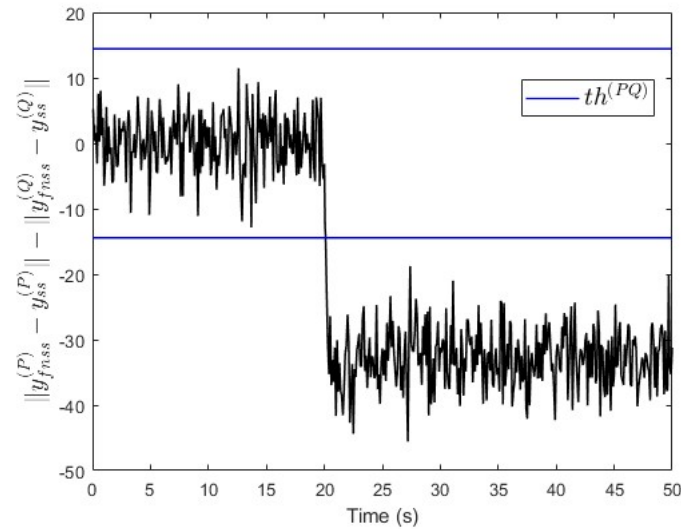
Figure 12. Case study 1—fault isolation in subsystem $S^{(Q)}$ with a parameter change.
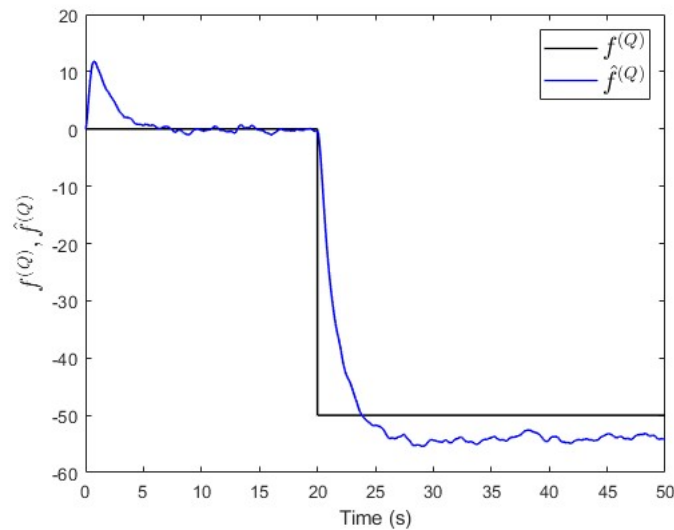

Figure 13. Case study 1—fault estimation in subsystem $S^{(Q)}$ with a parameter change.

5.3. Incipient Fault Analysis (for Case Study 1)

We also analysed the performance of our proposed approach in the case of an incipient fault (slowly developing fault). To do this, by using case study 1 again, a linearly increasing fault signal is assumed with an initial value of 0 and a slope value of 0.1 per unit of time. This fault is injected into subsystem $S^{(P)}$ at $t = 20$ s. Figure 14 shows that the value of $\|y^{(P)}_{fnss} - y^{(P)}_{ss}\| - \|y^{(Q)}_{fnss} - y^{(Q)}_{ss}\|$ is above the threshold $th^{(PQ)}$ some time after the fault event occurred. This indicates that a fault is occurring in subsystem $S^{(P)}$, which is correct. Meanwhile, Figure 15 shows that the observer successfully estimates the magnitude of the incipient fault.
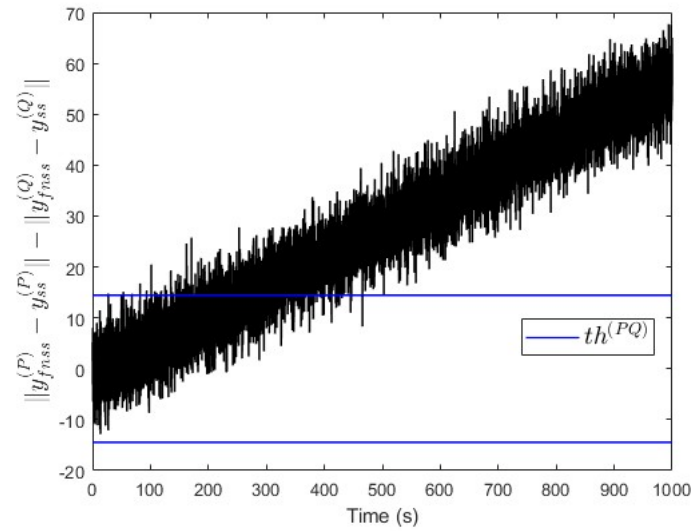
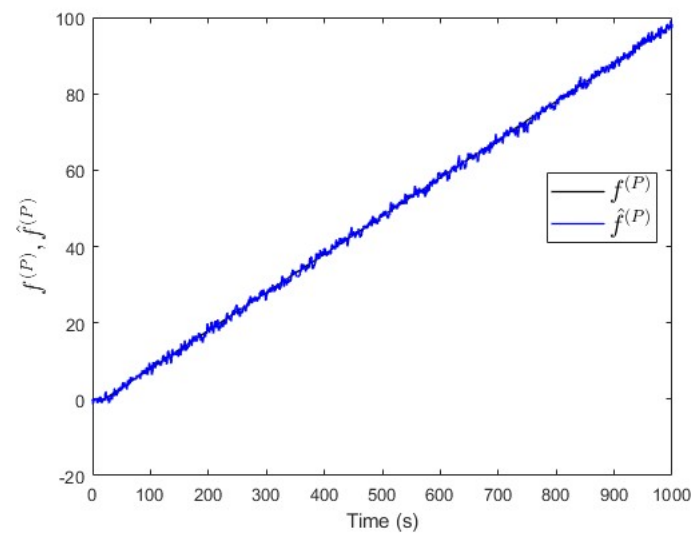**Figure 14.** Case study 1—fault isolation in subsystem $S^{(P)}$ with an incipient fault.

Figure 15. Case study 1—fault estimation in subsystem $S^{(P)}$ with an incipient fault.

5.4. Case Study 2

In this case study, six subsystems $S^{(A)}$, $S^{(B)}$, $S^{(C)}$, $S^{(D)}$, $S^{(E)}$, and $S^{(F)}$ are connected as shown in Figure 1. The model of the subsystems is given by Equation (20) where each subsystem is considered to have five state variables ($n = 5$). The applied parameters and external inputs are shown in Table 2.

Table 2. Case study 2—parameters and external inputs.

j	A	B	C	D	E	F
$v^{(j)}$	20	4	16	8	12	20
$k_E^{(j)}$	3	3	3	3	3	3
$R_u^{(j)}$	20	20	20	20	20	20
$Q_y^{(j)}$	20	20	20	20	20	20

In this network, two sensors are placed to measure both $y^{(A)}$ and $y^{(F)}$ that are affected by noise as presented in (20). To compensate for the noise that influences the system, a suitable threshold $th^{(AF)}$ is computed as presented in Equation (25). As both subsystem $S^{(A)}$ and $S^{(F)}$ are inside the loops of either $\left\{S^{(A)}, S^{(B)}, S^{(D)}, S^{(F)}\right\}$ or $\left\{S^{(A)}, S^{(B)}, S^{(E)}, S^{(F)}\right\}$, the simulation's results should show that: $\|y_{fnss}^{(A)} - y_{ss}^{(A)}\| - \|y_{fnss}^{(F)} - y_{ss}^{(F)}\|$ is above the threshold $th^{(AF)}$ when there is a fault in subsystem $S^{(A)}$, or $\|y_{fnss}^{(A)} - y_{ss}^{(A)}\| - \|y_{fnss}^{(F)} - y_{ss}^{(F)}\|$ is below $-th^{(AF)}$ when there is a fault in either subsystem $S^{(B)}$, $S^{(C)}$, $S^{(D)}$, $S^{(E)}$, or $S^{(F)}$ (see Algorithm 2).

In the first scenario, no fault is injected into the subsystems. The simulation results in Figure 16 show that $\|y_{fnss}^{(A)} - y_{ss}^{(A)}\| - \|y_{fnss}^{(F)} - y_{ss}^{(F)}\| \in (-th^{(AF)}, th^{(AF)})$, i.e., no fault is injected.
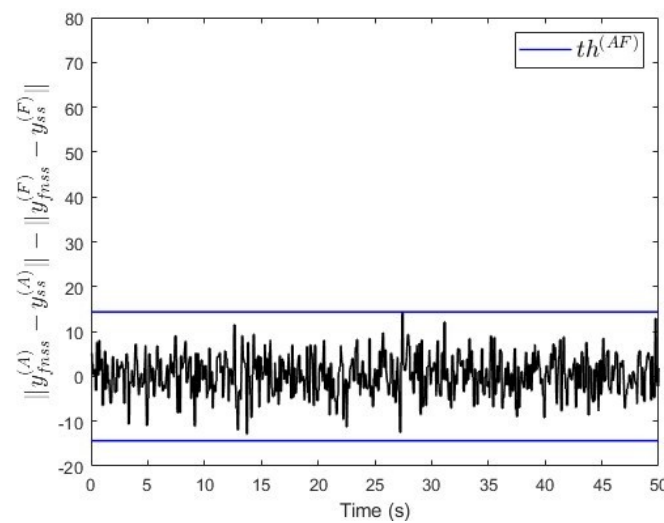

Figure 16. Case study 2—fault free case.

In the second scenario, a constant fault signal $f^{(A)} = 200$ is injected into subsystem $S^{(A)}$ at $t = 20$ s. Figure 17 shows that the value of $\|y_{fnss}^{(A)} - y_{ss}^{(A)}\| - \|y_{fnss}^{(F)} - y_{ss}^{(F)}\|$ is above the threshold after the fault event occurred. This indicates that a fault is occurring in subsystem $S^{(A)}$, which is correct.

To verify the proposed fault estimation approach, an observer is designed for subsystem $S^{(A)}$ based on Equation (30). By using the `lqe` MATLAB function, the observer's gain is computed such that the measurement noise variances are taken into consideration. Figure 18 shows the fault and its estimated value. It is seen that the fault's magnitude is correctly estimated. Meanwhile, Figure 19 shows that the observer successfully estimates the states of subsystem $S^{(A)}$.

In the third scenario, a constant fault signal $f^{(F)} = -150$ is injected into subsystem $S^{(F)}$ at $t = 20$ s. Figure 20 shows that the value of $\|y_{fnss}^{(A)} - y_{ss}^{(A)}\| - \|y_{fnss}^{(F)} - y_{ss}^{(F)}\|$ is below $-th^{(AF)}$ after the fault event occurred. This indicates that a fault is occurring either in subsystem $S^{(B)}$, $S^{(C)}$, $S^{(D)}$, $S^{(E)}$, or $S^{(F)}$, which is correct.

Moreover, an observer is designed specifically for subsystem $S^{(F)}$ based on the same Equation (30). The observer's gain is computed using the same `lqe` MATLAB function such that the noise variances are taken into consideration. Figure 21 shows the fault and its estimated value. Here, the fault's magnitude is also correctly estimated. Meanwhile, Figure 22 shows that the observer successfully estimates the states of subsystem $S^{(F)}$.
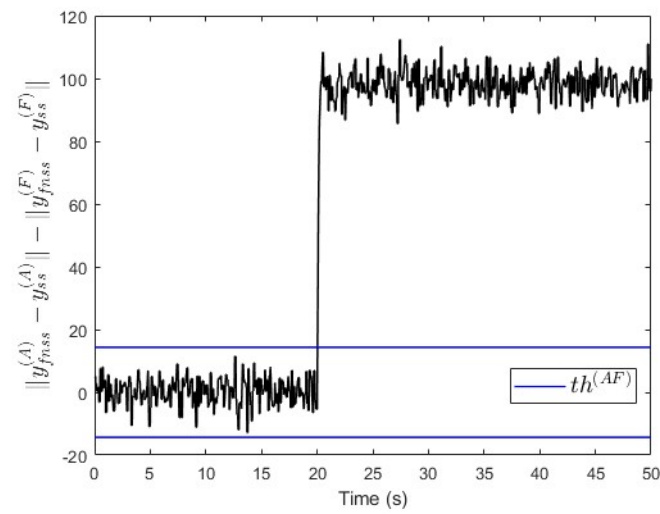
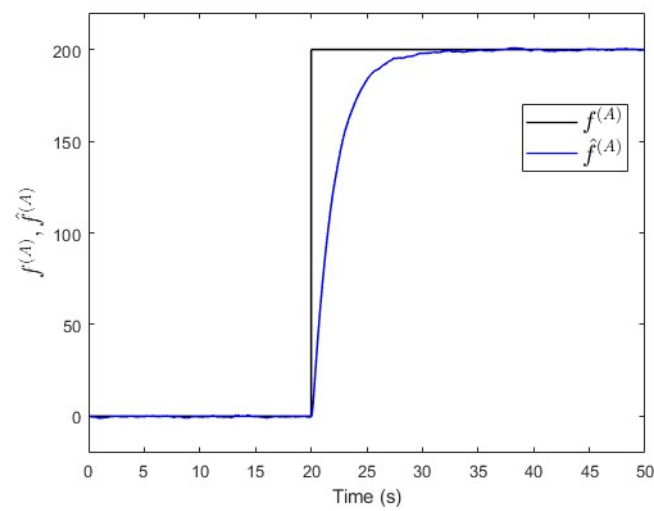**Figure 17.** Case study 2—fault isolation in subsystem $S^{(A)}$.

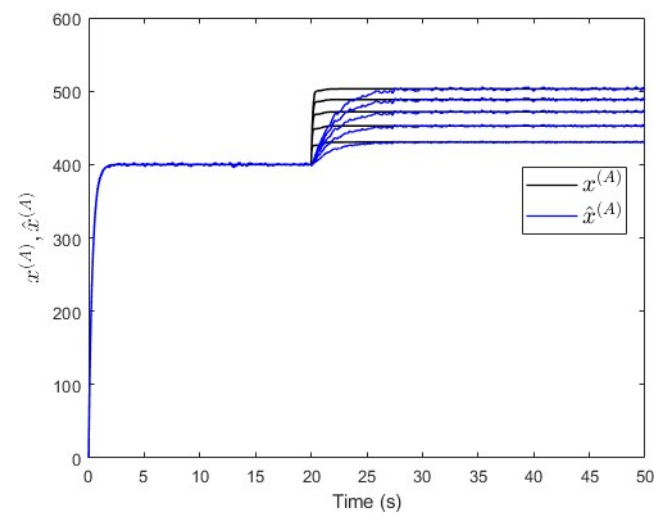**Figure 18.** Case study 2—fault estimation in subsystem $S^{(A)}$.

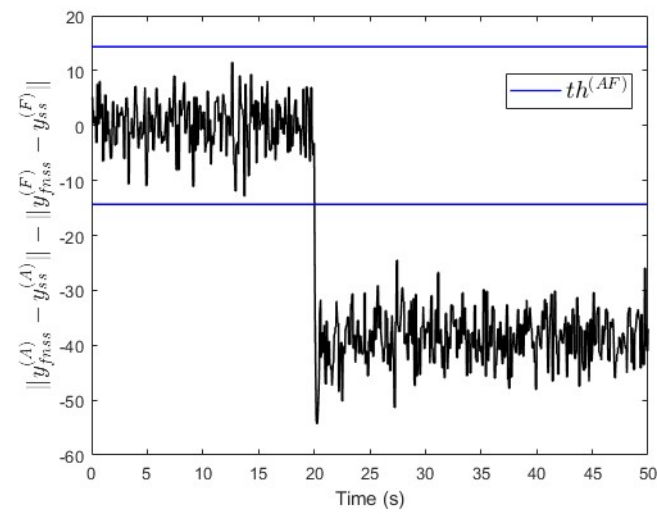**Figure 19.** Case study 2—states estimation in subsystem $S^{(A)}$.

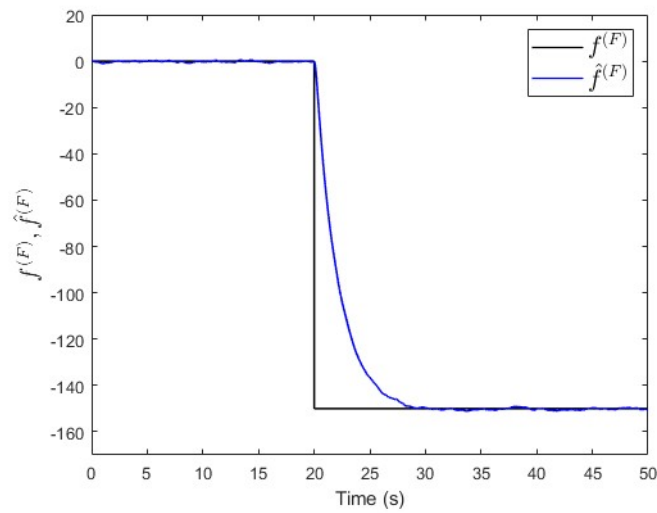**Figure 20.** Case study 2—fault isolation in subsystem $S^{(F)}$.

Figure 21. Case study 2—fault estimation in subsystem $S^{(F)}$.
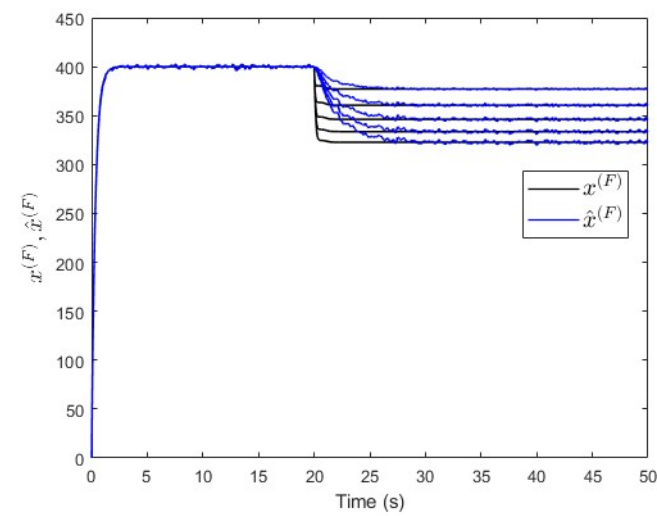

Figure 22. Case study 2—states estimation in subsystem $S^{(F)}$.

6. Conclusions

The case of networked linear process systems is considered in this paper with only a single conserved extensive quantity but with a network structure containing loops. It is assumed that the probability of multiple faults happening at the same time in the network is negligible.

For fault detection and isolation purposes, the network elements are described by a simple 2ISO LTI state space model where the fault enters as an additive linear term into the second input of the equations. Using the models of the network elements, a general model of the network is constructed which includes static splitting and joining connections. This results in an LTI state space model for the overall system that is suitable for fault detection and isolation.

By analysing the effect of the fault in a subsystem that propagates to the sensors' measurements at different positions, a fault isolation algorithm (Algorithm 1) was proposed. It uses two sensors, installed at the output of the two subsystems placed along a path, to locate the fault (i.e., decide if the fault has occurred in a subsystem on the connecting path or outside it). A steady state analysis and superposition principle were used to build the algorithm. An improved version of the algorithm (Algorithm 2) was also proposed to perform the localization in the presence of measurement noise.

Having completed the fault isolation process, a PI-based disturbance observer was then proposed to estimate the magnitude of the fault.

Two simulation case studies were used in the MATLAB/Simulink environment to verify and validate the proposed fault isolation and fault estimation methods. The first case study comprised a loop with two subsystems only. Meanwhile, the investigated network of the second case study comprised six subsystems. Separate subsections on analysing the parametric robustness and effect of an incipient fault were provided using the first case study for the sake of convenience but without generality loss.

In both case studies, the simulation results show good performance both in fault detection and isolation, and in fault magnitude estimation. In addition, our proposed method also shows good robustness against the transfer coefficient k_E which is difficult to estimate in practice. By using a ramp-like fault signal as an incipient fault, the simulation results show that our proposed fault estimation method can correctly estimate its magnitude.

Further work includes the extension of our method to cases of multiple faults, and verifying and validating the method on real data of, e.g., household heating/cooling systems.

Author Contributions: Conceptualization, W.K., K.M.H. and L.M.; methodology, W.K., K.M.H. and L.M.; software, W.K.; validation, W.K.; formal analysis, W.K., K.M.H. and L.M.; investigation, W.K., K.M.H. and L.M.; writing—original draft preparation, W.K.; writing—review and editing, W.K., K.M.H. and L.M.; supervision, K.M.H. and L.M. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Data will be made available on request.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Linear Quadratic Estimator (LQE)

Consider an extended state space model, as in Equation (28), is influenced by noise inputs as follows:

$$
\begin{aligned}
\dot{z}^{(k)} &= A_z^{(k)} z^{(k)} + B_z^{(k)} u_X^{(k)} + G_z^{(k)} w_{uX}^{(k)} \\
y^{(k)} &= C_z z^{(k)} + w_y^{(k)}
\end{aligned}
\tag{A1}
$$

where $G_z^{(k)}$ is the noise distribution matrix. In this system class, $G_z^{(k)} = B_z^{(k)}$.

Furthermore, assume that $\boldsymbol{w}_{uX}^{(k)}$ and $w_y^{(k)}$ are uncorrelated white noise with variance: $\boldsymbol{w}_{uX}^{(k)}(t) \sim (0, R_u^{(k)})$, $w_y^{(k)}(t) \sim (0, Q_y^{(k)})$.

Now, a state estimation error covariance matrix $P^{(k)}$ is defined that satisfies:

$$P^{(k)} \succ 0$$
$$\dot{P}^{(k)} = A_z^{(k)} P^{(k)} + P^{(k)} (A_z^{(k)})^T + G_z^{(k)} R_u^{(k)} (G_z^{(k)})^T - P^{(k)} C_z^T (Q_y^{(k)})^{-1} C_z P^{(k)} \tag{A2}$$

As expected, the error covariance $P^{(k)}$ diminishes quickly in a steady state value ($\dot{P}^{(k)} = 0$); therefore, the following Riccati equation can be solved:

$$0 = A_z^{(k)} P^{(k)} + P^{(k)} (A_z^{(k)})^T + G_z^{(k)} R_u^{(k)} (G_z^{(k)})^T - P^{(k)} C_z^T (Q_y^{(k)})^{-1} C_z P^{(k)} \tag{A3}$$

Then, the observer gain $L_z^{(k)}$ can be computed as [35,36]:

$$L_z^{(k)} = P^{(k)} C_z^T (Q_y^{(k)})^{-1} \tag{A4}$$

The gain $L_z^{(k)}$ in (A4) can be directly obtained by using the MATLAB function 'lqe'.

References

1. Yang, T.C. Networked control system: A brief survey. *IEE Proc.-Control Theory Appl.* **2006**, *153*, 403–412. [CrossRef]
2. Newman, M.E.; Barabási, A.L.E.; Watts, D.J. *The Structure and Dynamics of Networks*; Princeton University Press: Princeton, NJ, USA, 2006.
3. Abdallah, C.T.; Tanner, H.G. Complex networked control systems: Introduction to the special section. *IEEE Control Syst. Mag.* **2007**, *27*, 30–32.
4. Baillieul, J.; Antsaklis, P.J. Control and communication challenges in networked real-time systems. *Proc. IEEE* **2007**, *95*, 9–28. [CrossRef]
5. Gupta, R.A.; Chow, M.Y. Networked control system: Overview and research trends. *IEEE Trans. Ind. Electron.* **2009**, *57*, 2527–2535. [CrossRef]
6. Zhang, L.; Gao, H.; Kaynak, O. Network-induced constraints in networked control systems—A survey. *IEEE Trans. Ind. Inform.* **2012**, *9*, 403–416. [CrossRef]
7. Zhang, X.M.; Han, Q.L.; Ge, X.; Ding, D.; Ding, L.; Yue, D.; Peng, C. Networked control systems: A survey of trends and techniques. *IEEE/CAA J. Autom. Sin.* **2019**, *7*, 1–17. [CrossRef]
8. Varga, A. *Solving Fault Diagnosis Problems: Linear Synthesis Techniques*; Springer: Gewerbestrasse, Switzerland, 2017.
9. Boem, F.; Ferrari, R.M.; Parisini, T. Distributed fault detection and isolation of continuous-time non-linear systems. *Eur. J. Control* **2011**, *17*, 603–620. [CrossRef]
10. Granderson, J.; Singla, R.; Mayhorn, E.; Ehrlich, P.; Vrabie, D.; Frank, S. Characterization and Survey of Automated Fault Detection and Diagnostic Tools. Report Number LBNL-2001075. 2017. Available online: https://betterbuildingssolutioncenter.energy.gov/sites/default/files/\tools/lbnl-2001075.pdf (accessed on 4 April 2023).
11. Buffa, S.; Fouladfar, M.H.; Franchini, G.; Lozano Gabarre, I.; Andrés Chicote, M. Advanced control and fault detection strategies for district heating and cooling systems—A review. *Appl. Sci.* **2021**, *11*, 455. [CrossRef]
12. Manservigi, L.; Bahlawan, H.; Losi, E.; Morini, M.; Spina, P.R.; Venturini, M. A diagnostic approach for fault detection and identification in district heating networks. *Energy* **2022**, *251*, 123988. [CrossRef]
13. Zhou, S.; O'Neill, Z.; O'Neill, C. A review of leakage detection methods for district heating networks. *Appl. Therm. Eng.* **2018**, *137*, 567–574. [CrossRef]
14. Zhou, J.; Bennett, S. Dynamic system fault diagnosis based on neural network modelling. *IFAC Proc. Vol.* **1997**, *30*, 55–60. [CrossRef]
15. Niemann, H.H.; Stoustrup, J. Robust fault detection in open loop vs. closed loop. In Proceedings of the 36th IEEE Conference on Decision and Control, San Diego, CA, USA, 12 December 1997; Volume 5, pp. 4496–4497.
16. Niemann, H. A setup for active fault diagnosis. *IEEE Trans. Autom. Control* **2006**, *51*, 1572–1578. [CrossRef]
17. Keliris, C.; Polycarpou, M.M.; Parisini, T. A robust nonlinear observer-based approach for distributed fault detection of input–output interconnected systems. *Automatica* **2015**, *53*, 408–415. [CrossRef]
18. Ashari, A.E.; Nikoukhah, R.; Campbell, S.L. Active Robust Fault Detection of Closed-Loop Systems: General Cost Cas. *IFAC Proc. Vol.* **2009**, *42*, 585–590. [CrossRef]
19. Sun, B.; Wang, J.; He, Z.; Zhou, H.; Gu, F. Fault identification for a closed-loop control system based on an improved deep neural network. *Sensors* **2019**, *19*, 2131. [CrossRef]

20. Niemann, H.; Poulsen, N.K. Active fault detection in MIMO systems. In Proceedings of the 2014 American Control Conference, Portland, OR, USA, 4–6 June 2014; pp. 1975–1980.
21. Niemann, H.; Poulsen, N.K. Active fault isolation in MIMO systems. *IFAC Proc. Vol.* **2014**, *47*, 8012–8017. [CrossRef]
22. Cameron, I.T.; Hangos, K. *Process Modelling and Model Analysis*; Academic Press: San Diego, CA, USA, 2001.
23. Li, M.; Deng, W.; Xiahou, K.; Ji, T.; Wu, Q. A data-driven method for fault detection and isolation of the integrated energy-based district heating system. *IEEE Access* **2020**, *8*, 23787–23801. [CrossRef]
24. Shames, I.; Teixeira, A.M.; Sandberg, H.; Johansson, K.H. Distributed fault detection for interconnected second-order systems. *Automatica* **2011**, *47*, 2757–2764. [CrossRef]
25. Lipták, G.; Pituk, M.; Hangos, K.M. Modelling and stability analysis of complex balanced kinetic systems with distributed time delays. *J. Process. Control* **2019**, *84*, 13–23. [CrossRef]
26. Kurniawan, W.; Hangos, K.M.; Marton, L. Parameter fault diagnosis in heat exchange networks with distributed time delay. *IFAC-PapersOnLine* **2022**, *55*, 39–44. [CrossRef]
27. Kurniawan, W.; Hangos, K.M.; Marton, L. Parameter fault estimation in distributed heating/cooling systems. In Proceedings of the 7th International Conference on Sustainable Information Engineering and Technology 2022, Malang, Indonesia, 22–23 November 2022; pp. 111–118.
28. Horn, R.A.; Johnson, C.R. *Matrix Analysis*; Cambridge University Press: Cambridge, UK, 2012.
29. Berman, A.; Plemmons, R.J. *Nonnegative Matrices in the Mathematical Sciences*; SIAM: Philadelphia, PA, USA, 1994.
30. Briat, C. Sign properties of Metzler matrices with applications. *Linear Algebra Its Appl.* **2017**, *515*, 53–86. [CrossRef]
31. Cobelli, C.; Romanin-Jacur, G. Controllability, observability and structural identifiability of multi input and multi output biological compartmental systems. *IEEE Trans. Biomed. Eng.* **1976**, *2*, 93–100. [CrossRef] [PubMed]
32. Benvenuti, L.; Farina, L. Positive and compartmental systems. *IEEE Trans. Autom. Control* **2002**, *47*, 370–373. [CrossRef]
33. Johnson, D.B. Finding all the elementary circuits of a directed graph. *SIAM J. Comput.* **1975**, *4*, 77–84. [CrossRef]
34. Jibril, M.; Tadesse, M.; Alemayehu, E. State and disturbance estimation of a linear systems using proportional integral observer. *Int. Res. J. Mod. Eng. Technol. Sci.* **2020**, *2*, 111–118.
35. Åström, K.J. *Introduction to Stochastic Control Theory*; Dover Publications: Meniola, NY, USA, 2012.
36. Lewis, F.L.; Xie, L.; Popa, D. *Optimal and Robust Estimation: With an Introduction to Stochastic Control Theory*; CRC Press: Boca Raton, FL, USA, 2017.