A note on diagnosis and performance degradation detection in automatic control systems towards functional safety and cyber security

Steven X. Ding

Institute for Automatic Control and Complex Systems (AKS)
University of Duisburg-Essen, Bismarckstr. 81 BB, 47057 Duisburg, Germany

Abstract: This note addresses diagnosis and performance degradation detection issues from an integrated viewpoint of functionality maintenance and cyber security of automatic control systems. It calls for more research attention on three aspects, (i) application of control and detection unified framework to enhancing diagnosis capability of feedback control systems, (ii) projection-based fault detection, and complementary and explainable applications of projection- and machine learning-based techniques, and (iii) system performance degradation detection that is of elemental importance for today's automatic control systems. Some ideas and conceptual schemes are presented and illustrated by means of examples, serving as convincing arguments for research efforts in these aspects. They would contribute to the future development of capable diagnosis systems for functionality safe and cyber secure automatic control systems.

Keywords: Diagnosis in automatic control systems, cyber security in industrial CPS, unified framework of control and detection, projection-based diagnosis, explainable application of ML-methods, performance degradation detection.

1 Introduction

In the era of industry 4.0, automatic control systems as the centrepiece of industrial cyber-physical systems (CPSs) are fully equipped with intelligent sensors, actuators and an excellent information infrastructure. It is a logic consequence of ever increasing demands for system performance and production efficiency that today's automatic control systems are of an extremely high degree of integration, automation and complexity. Maintaining reliable and safe operations of automatic control systems are of elemental importance for optimally managing industrial CPSs over the whole operation life-cycle. As an indispensable maintenance functionality, real-time monitoring and diagnosis are widely integrated in automatic control systems and runs parallel to the embedded control systems.

This provisional PDF is the accepted version. The article should be cited as: Security and Safety doi: 10.1051/sands/2022004

In a traditional automatic control system, monitoring and diagnosis were mainly dedicated to maintaining functionalities of sensors and actuators as the key components embedded in the system (Frank, 1990; Frank and Ding, 1997). As a response to widely networking in modern automatic control systems, monitoring and diagnosis of networked control systems as a whole have received considerable attention as well in recent years (Ding *et al.*, 2013). Over the past three decades, innumerable capable diagnosis schemes have been developed with various specifications, for instance, detecting abrupt component failures (Gao *et al.*, 2015), identifying and predicting functionality loss caused by ageing in system components (Hwang *et al.*, 2010; Wen *et al.*, 2016), and intermittent faults depending on system operation conditions (Zhou *et al.*, 2020). Recently, a new type of malfunctions, the so-called cyber-attacks on automatic control systems, have drawn attention on the urgent need for developing new monitoring and diagnosis strategies (Pasqualetti *et al.*, 2013; Ding *e al.*, 2018; Giraldo *et al.*, 2018; Dibaji *et al.*, 2019). Cyber-attacks can not only considerably affect functionalities of sensors and actuators, but also impair communications among the system components and sub-systems, which may cause immense damages during system operations (Yan *et al.*, 2019; Tan *et al.*, 2020; Zhang *et al.*, 2021; Zhou *et al.*, 2021). In addition, different from technical faults, cyber-attacks are artificially created and could be designed by attackers in such a way they cannot be detected using the existing diagnosis techniques. Such cyber-attacks are called stealthy (Pasqualetti *et al.*, 2013). A further type of cyber-attacks are the so-called eavesdropping attacks. Although such attacks do not cause changes in system dynamics and performance degradation, they enable adversary to gain system knowledge which can be used to design, for instance, stealthy attacks. In a nutshell, management of cyber-attacks raises, besides functionality maintenance, cyber security issues in the framework of monitoring and diagnosis in automatic control systems.

The objective of this note is to address monitoring and diagnosis issues from an integrated viewpoint of functionality maintenance and cyber security of automatic control systems. We would like to call reader's attention to the following three aspects,

- application of control and detection unified framework (Ding, 2020) to enhancing diagnosis capability of feedback control systems,
- alternative technique of detecting faults in dynamic systems towards complementary and explainable applications of model- and machine learning (ML) based methods to diagnosis, and
- system performance degradation detection issues,

which are, to our best knowledge, not the current research mainstream in the relevant thematic fields. We will report ideas and research efforts, present conceptual schemes, and illustrate, also by means of examples, why research efforts in these three aspects could contribute to the development of capable monitoring and diagnosis methods towards enhancing functionality safety and cyber security of automatic control systems.

This note is motivated by our observations and research experiences in the field of fault diagnosis in technical systems and its industrial applications over past years. Reviewing publications on fault diagnosis in automatic control systems gives a clear picture of research efforts. That is, they were mainly devoted to the development of fault diagnosis functionality as a separate system running in parallel to the control system. With the increasing

complexity of control systems under consideration, from single-loop feedback control systems to networked control systems and recently CPSs, the set of investigated diagnosis issues has been continuously extended, and correspondingly capable but often complicated diagnosis methods have been developed, without paying attention on technical specifications and configurations of controllers embedded in the control system. For instance, successful solutions of detecting the so-called covert, zero dynamics and replay cyber-attacks are achieved by extending the well-established observer-based detection scheme with a moving target or an auxiliary system (Weerakkody and Sinopoli, 2015; Griffioen *et al.*, 2021; Schellenberger and Zhang, 2017) or injecting watermark signals (Mo *et al.*, 2015; Porter *et al.*, 2021; G.Ferrari and H.Teixeira, 2021). On the other hand, the unified control and detection framework (Ding, 2020) not only highlights the common information basis of control and detection, but also gives a functionalisation of a control system, which enables an integrated configuration of control and detection functionality with enhanced diagnosis capacity. Our recent work demonstrates successful applications of the unified framework to uniform detection of covert, zero dynamics and replay cyber-attacks without adding additional systems or signals (Ding *et al.*, 2021).

Thanks to the close relations of observers and controllers, observer-based diagnosis is the most popular technique applied for fault detection in automatic control systems (Frank, 1990; Frank and Ding, 1997). Observing the recent development in the thematic field of monitoring and diagnosis in industrial systems and processes, it can be clearly identified that ML-based methods form the mainstream of research. A detailed survey of publications on ML-based diagnosis in automatic control systems reveals obvious deficits in making use of system knowledge, which is no doubt available, since most of plants, partially or as a whole, are engineering systems. In fact, most of ML-based diagnosis methods are, in their core, based on the principle of reconstructing process variables or simply modelling of system fault-free operations. Thanks to the learning capacity of ML algorithms, in particular neural networks (NNs), and on the assumption of availability of rich data, ML-methods are potential technical solutions. Nevertheless, such diagnosis solutions could be far from optimal with respect to diagnosis performance, also due to the reason that often diagnostic specifications are not or could not be integrated in the existing ML algorithms. In comparison, model-based diagnosis methods, especially the observer-based ones, are fully based on the dynamic model of the system under consideration, and pursue optimal diagnosis performance. To approach this objective, advanced methods of control theory serve as major investigation tools. On the other hand, these methods, comparing with ML-based ones, are less capable of dealing with huge number of data and, above all, lack the learning ability. From these observations, a reasonable question arises: is it possible to efficiently integrate the model- and ML-based diagnosis methods to significantly enhance diagnosis performance? Our recent work on the so-called projection-based fault detection strategy is motivated by this question (Ding *et al.*, 2022). The first results showcase that complementary applications of model- and ML-based methods result in enhanced detection performance. The proposed projection-based fault detection method not only provides us with an alternative and more capable model-based solution than the observer-based ones, but also leads to explainable applications of ML-based methods.

It can be well observed that the major attention of the existing diagnosis methods has been dedicated to faults in hardware components of automatic control systems like sensors and actuators. We call those corresponding diagnosis methods component oriented diagnosis

(COD). In the recent decade, considerable efforts have been made in automation industry to increase the component reliability and, more recently, to enhance the intelligent degree of those key system components. Smart sensors and actuators are nowadays state of the art. And the new generation of smart system components are of the ability of self-diagnosis and self-repair. In an industrial CPS, COD is an issue to be addressed at the process level and locally. At the system level, due to the extremely high degree of automation and complexity, the system performance is often susceptible to variations of operation and environmental conditions. Moreover, it could considerably suffer not only from faults in sub-systems, but also from e.g. mismatching of coupled and networked control loops and controller parameters, interferences in system information infrastructure and cyber-attacks as well. This calls for research endeavour to develop new strategies of monitoring and detecting performance degradations, called performance oriented diagnosis (POD) (Ding and Li, 2021).

The remainder of this note consists of three main sections, respectively dedicated to the three topics, (i) the unified control and detection framework towards enhancing diagnosis capability of feedback control systems, (ii) projection-based detection of faults in dynamic systems and complementary, explainable applications of model- and ML-based methods, and (iii) study on POD issues. We would like to emphasise that the main intention of this note is to report ideas, research efforts, and conceptual schemes for the development of capable monitoring and diagnosis methods towards enhancing functionality safety and cyber security of automatic control systems. So far, no comparison study or survey of relevant publications are included. Concerning related issues, only representative works will be cited if needed. In order to have easy understandable descriptions, we avoid rigorous control theoretical and mathematical formulations, when there is no misleading interpretation or confusion.

2 Unified control and detection framework towards enhancing diagnosis capability of feedback control systems

As the methodological basis of our subsequent discussion, we first introduce the unified framework of control and detection. On this basis, we present functionalisation of a control system and its applications to enhancing diagnosis capability of feedback control systems.

Throughout this note, standard notations known in linear algebra and advanced control theory are adopted. In addition, $\mathcal{RH}_\infty$ is used to denote the set of all stable systems. In the context of cyber-attacks, when signal ξ is attacked, it is denoted by ξ^a, and the corresponding (injected) attack signal by a_ξ, i.e. $\xi^a = \xi + a_\xi$.

2.1 System representations and controller parameterisation

2.1.1 System factorisations, observer-based residual generation, and signal subspaces

In automatic control engineering, transfer functions are a standard model form for system input-output dynamics, which is written as

$$y(z) = G(z)u(z), y(z) \in \mathcal{C}^m, u(z) \in \mathcal{C}^p \tag{1}$$

with u and y as the plant input and output vectors, respectively. It is assumed that $G(z)$ is a proper real-rational matrix and its minimal state space realisation is given by the following discrete-time linear time invariant (LTI) system

$$x(k+1) = Ax(k) + Bu(k), x(0) = x_0, \tag{2}$$
$$y(k) = Cx(k) + Du(k), \tag{3}$$

where $x \in \mathcal{R}^n$ is the state vector and x_0 is the initial condition of the system. Matrices A, B, C, D are appropriately dimensioned real constant matrices. By means of the well-established coprime factorisation, $G(z)$ can be further factorised as

$$G(z) = \hat{M}^{-1}(z)\hat{N}(z) = N(z)M^{-1}(z) \tag{4}$$

with $\left(\hat{M}(z), \hat{N}(z)\right)$ and $(M(z), N(z))$ as left and right coprime pairs (LCP and RCP), which lead to alternative systems representations,

$$r_y(z) := K_G(z) \begin{bmatrix} u(z) \\ y(z) \end{bmatrix}, K_G(z) = \begin{bmatrix} -\hat{N}(z) & \hat{M}(z) \end{bmatrix}, r_y(z) = 0, \tag{5}$$

$$\begin{bmatrix} u(z) \\ y(z) \end{bmatrix} = I_G(z)\, v(z), I_G(z) = \begin{bmatrix} M(z) \\ N(z) \end{bmatrix} \tag{6}$$

for some signal $v(z)$. Their state space realisations are given, respectively, by

$$\hat{x}(k+1) = (A - LC)\hat{x}(k) + (B - LD)u(k) + Ly(k), \tag{7}$$
$$\hat{y}(k) = C\hat{x}(k) + Du(k), r_y(k) = y(k) - \hat{y}(k), \tag{8}$$
$$x(k+1) = (A + BF)x(k) + Bv(k), \tag{9}$$
$$\begin{bmatrix} u(k) \\ y(k) \end{bmatrix} = \begin{bmatrix} F \\ C + DF \end{bmatrix} x(k) + \begin{bmatrix} I \\ D \end{bmatrix} v(k). \tag{10}$$

System (7) is a state observer and builds, together with (8) (equivalently with (5)), an observer-based residual generator with residual vector r_y as its output. If $\hat{x}(0) \neq x_0$ or there exist uncertainties in the system, $r_y(k)$ will deviate from zero. In other words, $r_y(k)$ is an indicator for uncertainties in the system. In system (9)-(10), the input vector $u(k) = Fx(k) + v(k)$ can be interpreted as a state feedback controller with v as reference signal. Corresponding to these interpretations, matrices F and L are called state feedback gain and observer gain matrices and so selected such that $A + BF$ and $A - LC$ are Schur matrices. Systems K_G in (5) and I_G in (6) are also called stable kernel and image representations (SKR and SIR) of system (1).

Remark 1 *Hereafter, we may drop out the domain variable z or k when there is no risk of confusion.*

SKR and SIR are two alternative representations of dynamic systems, based on which the following definitions of kernel and image subspaces are introduced (Vinnicombe, 2000).

Definition: *Given the model (1) and the corresponding LCP and RCP* $\left(\hat{M}, \hat{N}\right)$ *and* (M, N), *the subspaces* $\mathcal{K}_G$ *and* $\mathcal{I}_G$ *defined by*

$$\mathcal{K}_G = \left\{ \begin{bmatrix} u \\ y \end{bmatrix} : \begin{bmatrix} -\hat{N} & \hat{M} \end{bmatrix} \begin{bmatrix} u \\ y \end{bmatrix} = 0 \right\}, \tag{11}$$

$$\mathcal{I}_G = \left\{ \begin{bmatrix} u \\ y \end{bmatrix} : \begin{bmatrix} u \\ y \end{bmatrix} = \begin{bmatrix} M \\ N \end{bmatrix} v \right\} \tag{12}$$

are called kernel and image subspace of G, *respectively.*

It is evident that $\mathcal{K}_G$ and $\mathcal{I}_G$ are subspaces in the $(m+p)$-dimensional data space and have the following properties

- $\mathcal{K}_G = \mathcal{I}_G$,
- $\mathcal{I}_G$ is uniquely generated by the p-dimensional signal v, and thus
- vector v can be understood as a latent (hidden) variable.

These properties enable applications of projection-based technique to dealing with fault diagnosis issues and hence build a bridge between the model- and ML-based methods. It promises the development of more efficient and capable methods for fault diagnosis, performance degradation monitoring and detection of cyber-attacks, as will be discussed in the remainder of this note.

It follows from the definition of coprime factorisation that there exist two RCP and LCP $\left(\hat{X}, \hat{Y}\right)$ and (X, Y) so that the so-called Bezout identity holds (Zhou, 1998; Vinnicombe, 2000),

$$\begin{bmatrix} X(z) & Y(z) \\ -\hat{N}(z) & \hat{M}(z) \end{bmatrix} \begin{bmatrix} M(z) & -\hat{Y}(z) \\ N(z) & \hat{X}(z) \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}. \tag{13}$$

It is of considerable interest to note their special state space realisations as controllers, i.e. an observer-based state feedback controller and its input-output dynamics (Ding, 2020),

$$\begin{aligned} \hat{x}(k+1) &= (A + BF)\,\hat{x}(k) + Lr_y(k), \\ \begin{bmatrix} u(k) \\ y(k) \end{bmatrix} &= \begin{bmatrix} F \\ C + DF \end{bmatrix} \hat{x}(k) + \begin{bmatrix} 0 \\ I \end{bmatrix} r_y(k) \\ &\iff \begin{bmatrix} u(z) \\ y(z) \end{bmatrix} = \begin{bmatrix} -\hat{Y}(z) \\ \hat{X}(z) \end{bmatrix} r_y(z), \end{aligned}$$

as well as an observer-based state feedback controller and a closed-loop "residual generator"

$$\begin{aligned} \hat{x}(k+1) &= (A - LC)\,\hat{x}(k) + (B - LD)\,u(k) + Ly(k), \\ v(k) &= u(k) - F\hat{x}(k) \\ \iff v(z) &= X(z)u(z) + Y(z)y(z). \end{aligned}$$

2.1.2 Parameterisation of stabilising controllers and basics of the unified control and detection framework

It is a well-known result that, given plant model (1), all stabilising controllers are parameterised by

$$K(z) = \hat{V}^{-1}(z)\hat{U}(z) = U(z)V^{-1}(z), \tag{14}$$

$$\hat{V}(z) = X(z) - Q(z)\hat{N}(z), \hat{U}(z) = -Y(z) - Q(z)\hat{M}(z), \tag{15}$$

$$V(z) = \hat{X}(z) - N(z)Q(z), U(z) = -\hat{Y}(z) - M(z)Q(z) \tag{16}$$

with the parameter system $Q(z) \in \mathcal{RH}_\infty$, where the RCPs and LCPs (M, N), $\left(\hat{X}, \hat{Y}\right)$ and $\left(\hat{M}, \hat{N}\right), (X, Y)$ are given before and satisfy Bezout identity (13). The parameterisation expression (14)-(15) is called Youla parameterisation (Zhou, 1998). It follows from (5)-(6) and Bezout identity (Ding *et al.*, 2010; Ding, 2020) that any (stabilising) output feedback controller,

$$u(z) = K(z)y(z) + v(z), \tag{17}$$

with $v(z)$ being the reference signal can be equivalently written as

$$u(z) = F\hat{x}(z) - Q(z)r_y(z) + \bar{v}(z), \bar{v}(z) = \hat{V}(z)v(z), \tag{18}$$

where $\hat{x}$ is the state estimate delivered by the observer (7). In other words, any output feedback controller is an observer-based controller and driven by the residual signal r_y. In (Ding, 2020), a further parameterisation form of all stabilising controllers,

$$u(z) = K_0(z)y(z) + Q_0(z)r_y(z) + \bar{v}(z), Q_0(z) \in \mathcal{RH}_\infty, \tag{19}$$

is introduced, where K_0 is an output stabilising controller, and Q_0 denotes the parameterisation system. Consequently, also those widely used industrial controllers like PI controllers can be written in the form of (19), as far as they stabilise the control loops.

2.2 Mapping from the signal space to residual space

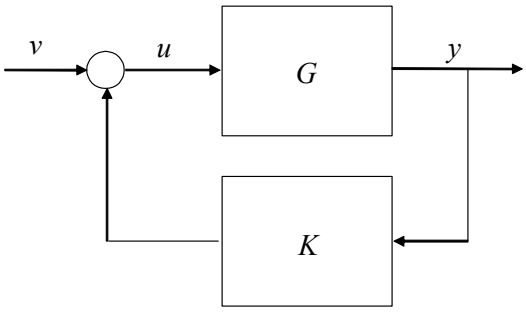

Figure 1: Feedback control loop under consideration

Consider the feedback control loop sketched in Figure 1 with the plant model (1) and con-

troller (17). It turns out

$$\begin{bmatrix} \bar{v}(z) \\ r_y(z) \end{bmatrix} = \begin{bmatrix} \hat{V}(z) & -\hat{U}(z) \\ -\hat{N}(z) & \hat{M}(z) \end{bmatrix} \begin{bmatrix} u(z) \\ y(z) \end{bmatrix} \Longleftrightarrow \tag{20}$$

$$\begin{bmatrix} u(z) \\ y(z) \end{bmatrix} = \begin{bmatrix} M(z) \\ N(z) \end{bmatrix} \bar{v}(z) + \begin{bmatrix} U(z) \\ V(z) \end{bmatrix} r_y(z). \tag{21}$$

From (21), it is obvious that the system signal pair (u, y) consists of two terms: the first one reflects the feed-forward control and the second one the response to the feedback control driven by the residual signal. Denoting uncertainties related to the controller by r_u, which may, for instance, be caused by attacks on actuators like injection of unknown signal, we have

$$\begin{bmatrix} r_u(z) \\ r_y(z) \end{bmatrix} = \begin{bmatrix} \hat{V}(z) & -\hat{U}(z) \\ -\hat{N}(z) & \hat{M}(z) \end{bmatrix} \begin{bmatrix} u(z) \\ y(z) \end{bmatrix} - \begin{bmatrix} \bar{v}(z) \\ 0 \end{bmatrix}. \tag{22}$$

Relation (22) gives a one-to-one mapping between the signal pairs (u, y) and (r_u, r_y) (for given $\bar{v}$). While (u, y) are the system measurement variables and represent the system dynamics, (r_u, r_y) build an information (residual) space and act as indicators for uncertainties in the system, including not only disturbances and parameter variations, but also faults and cyber-attacks when available. Hence, (22) can serve as a residual generator for detecting faults, performance degradation and cyber-attacks. Recall that feedback control is in its core residual-driven. That implies, feedback of residuals is sufficient for the control purpose. In this context, system (22) can be interpreted as an encoder that delivers the residuals (r_u, r_y) as code. It is noteworthy that, on the one hand, an identification of the system dynamics by means of the code (r_u, r_y) is generally impossible, and on the other hand, the cyber-attacks can be identified using the residual pair (r_u, r_y) under certain conditions (Ding *et al.*, 2021).

2.3 Functionalisation of all stabilising feedback controllers

In the light of the observer-based realisation of stabilising controllers given in (18), a feedback controller can be divided into several functional modules (Ding, 2020):

- an observer and an observer-based residual generator, as given in (7)-(8), which serve as an information provider for the controller and diagnostic system, and deliver a state estimation, $\hat{x}$, as well as the primary residual, $r_y = y - \hat{y}$,
- the control law
$$u(z) = F\hat{x}(z) - Q(z)r_y(z) + \hat{V}(z)v(z),$$
including a feedback controller, $F\hat{x} - Qr_y$, and a feed-forward controller, $\hat{V}v$, and in addition,
- for the detection purpose, a detector $R(z)r_y(z)$ with $R(z)$ as a stable post-filter.

This modular structure provides us with a clear parameterisation of the functional modules: the state observer is parameterised by the observer gain L, the feedback controller by F, Q, the feed-forward controller by $\hat{V}$, and the detector by R. Although all five parameters are available for the design and online optimisation objectives, they have evidently different functionalities, as summarised below (Ding, 2020):

- state feedback and observer gains determine the stability and eigen-dynamics of the closed-loop,
- $R, \hat{V}$ have no influence on the system stability, and R serves for the optimisation of the detectability, while $\hat{V}$ for the tracking behavior, and
- Q is used to enhance the system robustness and control performance. The design and update of Q will have influence on the system dynamics and stability, when parameter uncertainties or degradations are present in the system.

It is evident that the above five parameters have to be, due to their different functionalities, treated with different priorities. Recall that system stability and eigen-dynamics are the fundamental requirement on an automatic control system. This requires that the system stability should be guaranteed, also in case of cyber-attacks. Differently, $Q, R, \hat{V}$ are used to optimise control or detection performance. In case that a temporary system performance degradation is tolerable, the real-time demand and the priority for an online optimisation of $Q, R, \hat{V}$ are relatively lower.

When an automatic control system is integrated into a CPS, the cyber security becomes a critical issue. In this context, the unified framework and the functionalisation of controllers offer a useful design tool towards a cyber security-conscious system configuration. To delineate potential applications, consider the controller in its original form and in the observer-based realisation form, respectively,

controller (17) $u(z) = K(z)y(z) + v(z)$ vs. controller (18) $u(z) = F\hat{x}(z) - Q(z)r_y(z) + \bar{v}(z)$,

and suppose that the plant is networked with a control station (referred to Figure 2 as an example). It is clear that for the implementation of the controller in its original form, i.e. (17), the system data (u, y) should be real-time transmitted over the network. Moreover, for any optimisation or degradation recovering effort, controller $K(z)$ should be updated which may yield unexpected dynamic behaviour. Differently, for the implementation of observer-based controller (18), an observer and an observer-based residual generator can be implemented on the plant side. This offers several benefits:

- transformation of the residual r_y from the plant (local) side to the control station and $\bar{v}(z) - Q(z)r_y(z)$ from the control station to the plant, which prevent adversary to gain system knowledge by means of eavesdropping attacks (Ding *et al.*, 2021),
- when performance optimisation or degradation recovery is need, real-time tuning $Q(z)$ is an effective way, as reported in (Li *et al.*, 2019), which can run in the control station,
- updating feedback gain and observer gain matrices, F and L, which will be performed only in very critical operation situations (and thus occasionally) and in the control station. Their transmission to the plant should be well encrypted (Schulze *et al.*, 2021).

As reported in our recent work (Ding *et al.*, 2021), the modules of the observer-based controller (18) together with the Bezout identity (13) can serve as encoders and decoders distributed at the plant and control station sides. It is noteworthy that the observer-based controller form (18) can be viewed as "control sharing", which is similar to the secret sharing scheme well-known in cryptography (Schulze *et al.*, 2021). This additional function enables efficient

detection of cyber-attacks and enhances the cyber security of automatic control systems, which are, for instance, implemented in form of cloud-based control (Schulze *et al.*, 2021).

In the following example, we introduce a conceptual configuration of an encrypted control system based on the above controller functionalisation.

Example 1 *Consider a networked automatic control system schematically sketched in Figure 2. The plant is modelled by (1), equipped with a (local) feedback controller,*

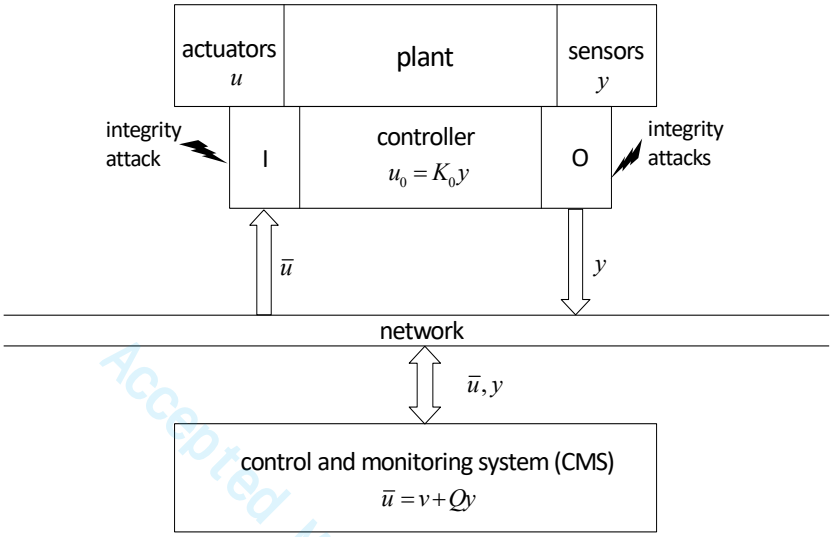

Figure 2: The original configuration of the automatic control system under consideration

$$u_0(z) = K_0(z)y(z),$$

and networked with a control and monitoring system (CMS). It receives signal $\bar{u}$ from CMS,

$$\bar{u}(z) = v + Q(z)y(z), \tag{23}$$

where v is the reference signal and $Q(z)y(z)$ represents a correction of the control signal, for instance, to recover control performance degradation (Ding, 2020). A natural procedure to realise the control law (23) is, as shown in Figure 2, as follows, (i) the plant sends the measurement data y to CMS, and (ii) CMS computes $\bar{u}$ and sends it to the plant. Suppose that integrity cyber-attacks could be executed on the system I/O interface via the network. Now, we introduce a conceptual reconfiguration of the systems on the both network sides, on the basis of the unified control and detection framework, aiming at

- *a reliable detection of integrity cyber-attacks, and*
- *preventing attackers to gain system knowledge by means of system identification using the transmitted data $(\bar{u}, y)$.*

Moreover, it is required that the local controller K_0 should not be changed. For our purpose, consider the control signal

$$u(z) = u_0(z) + \bar{u}(z) = K_0(z)y(z) + Q(z)y(z) + v(z).$$

Following the functionalisation of control systems, u_0 and u can be equivalently written into

$$u_0(z) = F_0\hat{x}(z) - Q_0(z)r_y(z),$$
$$u(z) = F_0\hat{x}(z) - Q_1(z)r_y(z) + \hat{V}(z)v(z)$$

for some $Q_0(z), Q_1(z) \in \mathcal{RH}_\infty$. It turns out

$$\bar{u}(z) = (Q_0(z) - Q_1(z))\, r_y(z) + \hat{V}(z)v(z). \tag{24}$$

Run the following residual generation algorithm on the plant side,

$$\begin{bmatrix} r_u(z) \\ r_y(z) \end{bmatrix} = \begin{bmatrix} X(z) & Y(z) \\ -\hat{N}(z) & \hat{M}(z) \end{bmatrix} \begin{bmatrix} u^a(z) \\ y(z) \end{bmatrix} - \begin{bmatrix} \bar{u}^a(z) - Q_0(z)r_y(z) \\ 0 \end{bmatrix}, \tag{25}$$

where

$$u^a(z) = u(z) + a_u(z) = u_0(z) + \bar{u}^a(z), \bar{u}^a(z) = \bar{u}(z) + a_u(z)$$

with a_u denoting integrity cyber-attacks on the actuators. It yields, recalling (22),

$$r_u(z) = (X(z) - I)\, a_u(z).$$

Thus, the attack a_u can be detected. In attack-free case, r_y is sent to CMS, otherwise, alarm is triggered. On the CMS side, a detection algorithm is applied to check if the residual signal received from the plant side is corrupted by attack signal a_y, i.e.

$$r_y^a(z) = r_y(z) + a_y(z).$$

In case of no attack, $\bar{u}$ computed using algorithm (24) is sent to the plant side. Figure 3 shows the above described control system schematically.

We would like to summarise the main results of this example as follows:

- *the proposed control system is capable for a reliable attack detection thanks to the use of the residual pair (r_u, r_y),*
- *system (24) and residual generator (25) serve simultaneously as encoders, and*
- *the control system operates stable also in the case of an interrupted communication between the plant and CMS.*

It should be moreover mentioned that the control system located at the plant side runs only based on the controller parameters, $K_0(z)$ as well as $F_0, Q_0(z)$, without knowledge about $Q_1(z)$ that is set by CMS for enhancing the control performance.

With the following remarks we would like to conclude this section.

- The control and detection unified framework forms a methodical basis for the development of advanced diagnosis methods aiming at maintaining system functionality and enhancing cyber security of automatic control systems. It deals with the implementation of control, detection and monitoring algorithms. In this context, the information infrastructure for the configuration of automatic control systems plays an essential role. For instance, the networked system in Figure 3 could be alternatively configured using cloud-based system structure, in which the CMS is realised by means of cloud computing.

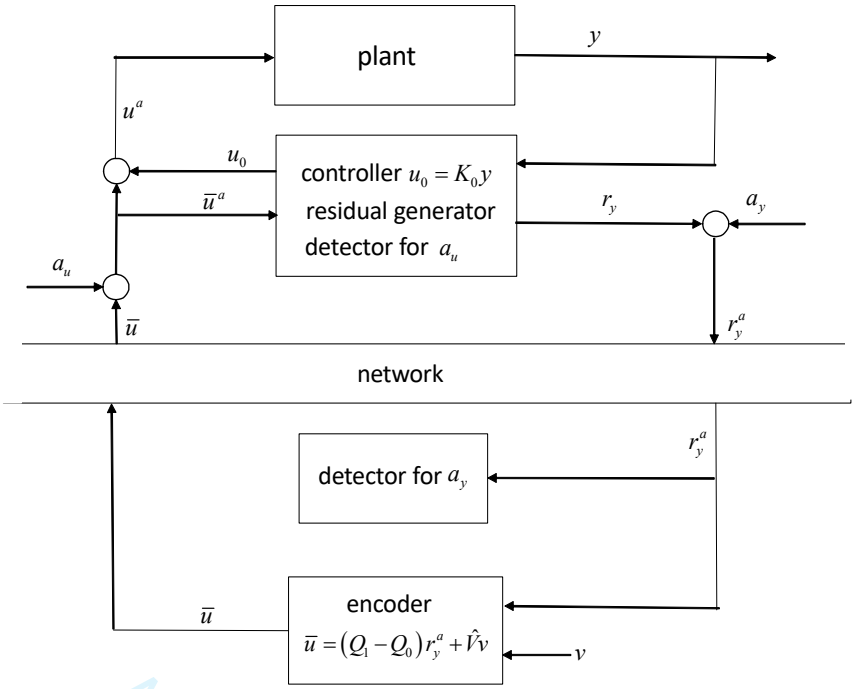

Figure 3: Reconfiguration of the automatic control system under consideration

- Although only LTI systems are addressed in this note, an extension of the unified control and detection framework to linear time-varying (LTV) systems is straightforward using the well-established system coprime factorisation methods and Youla parameterisation of LTV control systems (Feintuch, 1998). Concerning nonlinear control systems, corresponding results have been reported in (Han *et al.*, 2019; Ding, 2020; Han *et al.*, 2021).

- In our example, the application of the unified framework to the detection of cyber-attacks is schematically and shortly illustrated. The reader is referred to (Ding *et al.*, 2021) for a more systematic and detailed description of this application. In a nutshell, this work results in detection of those stealthy cyber-attacks, which cannot be detected using the existing observer-based detection methods (Ding, 2008). These include the so-called covert, zero dynamics and replay cyber-attacks (Pasqualetti *et al.*, 2013; Ding *et al.*, 2018; Giraldo *et al.*, 2018; Dibaji *et al.*, 2019).

3 Projection-based diagnosis methods and their ML-aided explainable realisation

In this section, we introduce a new framework for fault diagnosis in dynamic control systems. The theoretical foundation of this framework is the alternative system representations SIR, SKR and the associated image and kernel subspaces, as well as orthogonal projection technique. Although this framework has been developed in the model-based fashion (Ding *et al.*, 2022), the associated concepts, algorithms and diagnosis approaches can be realised in the data-driven form and using ML-based methods.

In this section, the following notations are adopted. $\mathcal{L}_2 = \mathcal{L}_2(-\infty, 0] \oplus \mathcal{L}_2[0, \infty)$ is the

time domain space of all square summable Lebesgue signals (signals with bounded energy) (Francis, 1987). For transfer matrix $G(z), G^*(z) = G^T(z^{-1})$. $\mathcal{P}_\mathcal{K}$ is an orthogonal projection operator onto subspace $\mathcal{K}$, whose norm is denoted by $\|\mathcal{P}_\mathcal{K}\|$. $\mathcal{P}^*_\mathcal{K}$ is the adjoint of $\mathcal{P}_\mathcal{K}$. $\mathcal{K}^\perp$ represents the orthogonal complement of $\mathcal{K}$.

3.1 A general framework of projection-based diagnosis methods

3.1.1 Basic idea

The basic idea of (orthogonal) projection-based fault detection can be schematically explained by Figure 4. Given a system subspace as the nominal system model, which can be

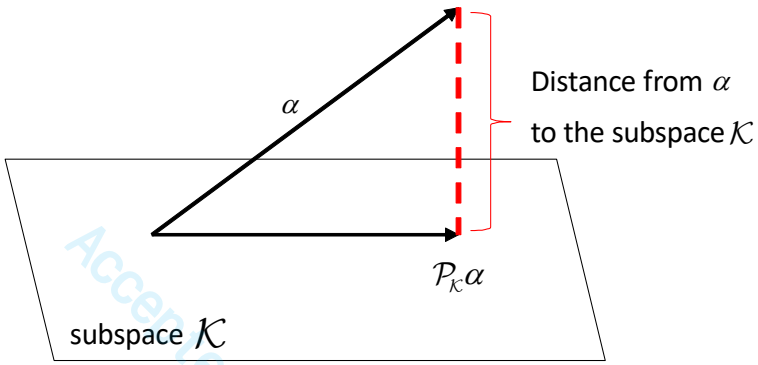

Figure 4: Schemetic description of projection-based classification ($\mathcal{P}_\mathcal{K}\alpha$ denotes the projection of α onto $\mathcal{K}$)

presented in the model-based form (in terms of SIR or SKR) or data-driven or by means of an NN, by (orthogonally) projecting the measurement vector $\begin{bmatrix} u \\ y \end{bmatrix}$ onto the system subspace, the distance between the measurement vector and its projection indicates if the measurement vector belongs to the nominal system operations or it is faulty. To this end, the following mathematical concepts and work are necessary:

- definition and computation of orthogonal projection operator,
- computation of $dist\left(\begin{bmatrix} u \\ y \end{bmatrix}, \mathcal{K}\right)$,
- online realisation algorithms towards constructing a fault detection system, and
- determination of threshold for decision making.

3.1.2 Orthogonal projection: mathematical preliminaries

An orthogonal projection on a subspace $\mathcal{V}$, denoted by $\mathcal{P}_\mathcal{V}$, in Hilbert space endowed with the inner product,

$$\langle x, y\rangle = \sqrt{\sum_{k=-\infty}^{\infty} x^T(k)y(k)}, x, y \in \mathcal{L}_2, \tag{26}$$

is a linear operator satisfying (Kato, 1995)

$$x, y \in \mathcal{V}, \mathcal{P}_{\mathcal{V}}^2 = \mathcal{P}_{\mathcal{V}}, \langle \mathcal{P}_{\mathcal{V}} x, y \rangle = \langle x, \mathcal{P}_{\mathcal{V}} y \rangle . \tag{27}$$

The following well-known properties and definitions of an orthogonal projection are of importance for our subsequent study (Kato, 1995):

- given $y \in \mathcal{L}_2, \forall x \in \mathcal{V} \in \mathcal{L}_2$,

$$\langle y - x, y - x \rangle = \|y - x\|_2 \geq \|y - \mathcal{P}_{\mathcal{V}} y\|_2 , \tag{28}$$

- given a closed subspace $\mathcal{V} \in \mathcal{L}_2$ and a vector $y \in \mathcal{L}_2$, the distance between y and $\mathcal{V}, dist\,(y, \mathcal{V})$, is defined as

$$dist\,(y, \mathcal{V}) = \inf_{x \in \mathcal{V}} \|y - x\|_2 ,$$

  which, following (28), can be computed as

$$dist\,(y, \mathcal{V}) = (\mathcal{I} - \mathcal{P}_{\mathcal{V}})\, y = \mathcal{P}_{\mathcal{V}^\perp} y.$$

  Here, $\mathcal{I}$ is the unit operator.

In order to measure the distance between two (closed) subspaces in Hilbert space, the concept of gap metric is established (Kato, 1995). Given two closed subspaces $\mathcal{V}, \mathcal{U} \in \mathcal{L}_2$, the gap metric between them is defined by

$$\delta\,(\mathcal{V}, \mathcal{U}) = \max \left\{ \vec{\delta}\,(\mathcal{V}, \mathcal{U}), \vec{\delta}\,(\mathcal{U}, \mathcal{V}) \right\}, \tag{29}$$

$$\vec{\delta}\,(\mathcal{V}, \mathcal{U}) = \sup_{\substack{x \in \mathcal{V} \\ \|x\|_2 = 1}} dist\,(x, \mathcal{U}) = \|(\mathcal{I} - \mathcal{P}_{\mathcal{U}})\, \mathcal{P}_{\mathcal{V}}\| = \sup_{x \in \mathcal{V}} \inf_{y \in \mathcal{U}} \frac{\|x - y\|_2}{\|x\|_2}. \tag{30}$$

Here, $\vec{\delta}\,(\mathcal{V}, \mathcal{U})$ is called directed gap. The following properties are well-known (Kato, 1995) and useful for our subsequent investigation:

$$0 \leq \delta\,(\mathcal{V}, \mathcal{U}) \leq 1,$$
$$\text{for } \delta\,(\mathcal{V}, \mathcal{U}) < 1, \vec{\delta}\,(\mathcal{V}, \mathcal{U}) = \vec{\delta}\,(\mathcal{U}, \mathcal{V}) = \delta\,(\mathcal{V}, \mathcal{U}) .$$

3.1.3 Orthogonal projection onto image subspace and its system realisations

In our subsequent study on projection-based fault diagnosis framework, the so-called normalised SKR and SIR play an important role, which are denoted by K_N and I_N and defined by

$$K_N(z) K_N^*(z) = \hat{N}_0(z) \hat{N}_0^*(z) + \hat{M}_0(z) \hat{M}_0^*(z) = I,$$
$$I_N^*(z) I_N(z) = M_0^*(z) M_0(z) + N_0^*(z) N_0(z) = I,$$

where $\left(\hat{M}_0, \hat{N}_0\right)$ and (M_0, N_0) are LCP and RCP with special settings of the observer and state feedback gain matrices using the known algorithms e.g. given in (Hoffmann, 1996). It is a known result that the orthogonal projection onto the image subspace $\mathcal{I}_G$ is given by

$$p_{\mathcal{I}_G} = \mathcal{P}_{\mathcal{I}_G} \begin{bmatrix} u \\ y \end{bmatrix} = I_N I_N^* \begin{bmatrix} u \\ y \end{bmatrix}. \tag{31}$$

Correspondingly, the difference between $\begin{bmatrix} u \\ y \end{bmatrix}$ and $p_{\mathcal{I}_G}$ is subject to

$$r_{\mathcal{I}_G} := \begin{bmatrix} u \\ y \end{bmatrix} - p_{\mathcal{I}_G} = (\mathcal{I} - \mathcal{P}_{\mathcal{I}_G}) \begin{bmatrix} u \\ y \end{bmatrix} = (I - I_N I_N^*) \begin{bmatrix} u \\ y \end{bmatrix}, \tag{32}$$

and called projection-based residual. Due to the relation,

$$I_N I_N^* + K_N^* K_N = I,$$

projection-based residual generation (32) can be equivalently written as

$$r_{\mathcal{I}_G} = (I - I_N I_N^*) \begin{bmatrix} u \\ y \end{bmatrix} = K_N^* K_N \begin{bmatrix} u \\ y \end{bmatrix}. \tag{33}$$

The l_2-norm of $r_{\mathcal{I}_G}$,

$$\|r_{\mathcal{I}_G}\|_2 = dist\left(\begin{bmatrix} u \\ y \end{bmatrix}, \mathcal{I}_G \right) \tag{34}$$

is the distance from $\begin{bmatrix} u \\ y \end{bmatrix}$ to $\mathcal{I}_G$. Moreover, the fact that K_N is a normalised SKR leads to the following implementation form of the residual vector,

$$\|r_{\mathcal{I}_G}\|_2 = \left\| K_N^* K_N \begin{bmatrix} u \\ y \end{bmatrix} \right\|_2 = \left\| K_N \begin{bmatrix} u \\ y \end{bmatrix} \right\|_2 = \|r_y\|_2 . \tag{35}$$

That means, for the detection purpose with the residual evaluation function $\|r_{\mathcal{I}_G}\|_2$, the needed online computation is the observer-based residual generator (7)-(8) or equivalently the SKR (5) with the observer gain setting for a normalised SKR.

Next, on the assumption that the system dynamics with uncertainty is described by

$$G = NM^{-1} = (N_0 + \Delta_N)(M_0 + \Delta_M)^{-1} = \hat{M}^{-1}\hat{N} = \left(\hat{M}_0 + \Delta_{\hat{M}}\right)^{-1} \left(\hat{N}_0 + \Delta_{\hat{N}}\right), \tag{36}$$

$$I_G = \begin{bmatrix} M \\ N \end{bmatrix} = \begin{bmatrix} M_0 + \Delta_M \\ N_0 + \Delta_N \end{bmatrix} = I_N + \Delta_I,$$

$$K_G = \begin{bmatrix} -\hat{N} & \hat{M} \end{bmatrix} = \begin{bmatrix} -\hat{N}_0 - \Delta_{\hat{N}} & \hat{M}_0 + \Delta_{\hat{M}} \end{bmatrix} = K_N + \Delta_K$$

$$\sup \|\Delta_I\|_\infty = \delta_{\Delta_I} < 1, \sup \|\Delta_K\|_\infty = \delta_{\Delta_K} < 1,$$

the threshold is to be determined. Considering that the idea of setting threshold is to avoid false alarms caused by model uncertainty during fault-free operations, a basic requirement on the threshold is that

$$\forall \begin{bmatrix} u \\ y \end{bmatrix} \in \mathcal{I}_G, J_{th}(u, y) = \sup_{\|\Delta_I\|_\infty \le \delta_{\Delta_I}} \|r_{\mathcal{I}_G}\|_2 , \tag{37}$$

$$\mathcal{I}_G = \left\{ \begin{bmatrix} u \\ y \end{bmatrix} : \begin{bmatrix} u \\ y \end{bmatrix} = \begin{bmatrix} M \\ N \end{bmatrix} v \right\},$$

which is obviously different from $\mathcal{I}_{G_0}$,

$$\mathcal{I}_{G_0} = \left\{ \begin{bmatrix} u \\ y \end{bmatrix} : \begin{bmatrix} u \\ y \end{bmatrix} = \begin{bmatrix} M_0 \\ N_0 \end{bmatrix} v \right\}.$$

In (Ding *et al.*, 2022), it is proved that the threshold setting problem (37) is equivalent to

$$J_{th}(u,y) = \sup_{\|\Delta_I\|_\infty \le \delta_{\Delta_I}} \|r_{\mathcal{I}_G}\|_2 = \sup_{\begin{bmatrix} u \\ y \end{bmatrix} \in \mathcal{I}_G} \delta(\mathcal{I}_G, \mathcal{I}_{G_0}) \left\| \begin{bmatrix} u \\ y \end{bmatrix} \right\|_2$$

with $\delta(\mathcal{I}_G, \mathcal{I}_{G_0})$ denoting the gap metric between $\mathcal{I}_{G_0}$ and $\mathcal{I}_G$. It leads to

$$J_{th}(u,y) = \delta_{\Delta_K} \left(\|r_y\|_2^2 + \left\| \mathcal{P}_{\mathcal{I}_G} \begin{bmatrix} u \\ y \end{bmatrix} \right\|_2^2 \right)^{1/2} \Longleftrightarrow$$

$$J_{th}(u,y) = \frac{\delta_{\Delta_K}}{\sqrt{1-\delta_{\Delta_K}^2}} \left\| \mathcal{P}_{\mathcal{I}_G} \begin{bmatrix} u \\ y \end{bmatrix} \right\|_2 = \frac{\delta_{\Delta_K}}{\sqrt{1-\delta_{\Delta_K}^2}} \left(\left\| \begin{bmatrix} u \\ y \end{bmatrix} \right\|_2^2 - \|r_y\|_2^2 \right)^{1/2}. \tag{38}$$

Comparing with the well-established threshold setting for observer-based fault detection schemes (Li and Ding, 2020*a*), threshold (38) is of significant advantage that it is considerably robust against uncertainties and sensitive to the faulty operations. In fact, this point becomes more apparent, when the threshold and the residual are normalised as follows,

$$J_{th,N}(u,y) = \frac{J_{th}(u,y)}{\left\| \begin{bmatrix} u \\ y \end{bmatrix} \right\|_2} = \frac{\delta_{\Delta_K}}{\sqrt{1-\delta_{\Delta_K}^2}} \left(1 - \|r_{y,N}\|_2^2\right)^{1/2},$$

$$r_{y,N} = \frac{1}{\left\| \begin{bmatrix} u \\ y \end{bmatrix} \right\|_2} r_y.$$

It can be seen that the threshold $J_{th,N}(u,y)$ reaches its maximal value during the fault-free operations, and becomes smaller as the system is in faulty operations. In this way, the robustness and fault detectability are remarkably enhanced.

Example 2 *In this example, we introduce a data-driven realisation of the projection-based detection scheme. Departing from the system model (2)-(3), the system dynamics can be written as*

$$y_s(k) = \Gamma_s x(k-s) + H_{u,s} u_s(k) \tag{39}$$

$$\Gamma_s = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^s \end{bmatrix} \in \mathcal{R}^{(s+1)m \times n}, H_{u,s} = \begin{bmatrix} D & 0 & & \\ CB & \ddots & \ddots & \\ \vdots & \ddots & \ddots & 0 \\ CA^{s-1}B & \cdots & CB & D \end{bmatrix} \in \mathcal{R}^{(s+1)m \times (s+1)p},$$

where $y_s(k), u_s(k)$ *are signal vectors of the data format*

$$\varsigma_s(k) = \begin{bmatrix} \varsigma(k-s) \\ \vdots \\ \varsigma(k) \end{bmatrix},$$

and s is an integer giving the length of the time interval $[k-s,k]$ of interest. To simplify our study, assume that the system is stable, and $x(k-s)$ is neglectable. By defining the orthogonal projection

$$\mathcal{P}_{\mathcal{I}} = \begin{bmatrix} I \\ H_{u,s} \end{bmatrix} \left(\begin{bmatrix} I & H_{u,s}^T \end{bmatrix} \begin{bmatrix} I \\ H_{u,s} \end{bmatrix} \right)^{-1} \begin{bmatrix} I & H_{u,s}^T \end{bmatrix},$$

a projection-based residual vector is constructed as follows

$$r_{\mathcal{I}}(k) = \begin{bmatrix} u_s(k) \\ y_s(k) \end{bmatrix} - \mathcal{P}_{\mathcal{I}} \begin{bmatrix} u_s(k) \\ y_s(k) \end{bmatrix}.$$

Note that

$$r_s(k) = \Pi^{1/2} \left(y_s(k) - H_{u,s} u_s(k) \right)$$

builds a residual vector and can be interpreted as a data-driven realisation of an observer-based residual generator. Moreover, it holds

$$\begin{aligned} & I - \begin{bmatrix} I \\ H_{u,s} \end{bmatrix} \left(\begin{bmatrix} I & H_{u,s}^T \end{bmatrix} \begin{bmatrix} I \\ H_{u,s} \end{bmatrix} \right)^{-1} \begin{bmatrix} I & H_{u,s}^T \end{bmatrix} \\ = & \begin{bmatrix} -H_{u,s}^T \\ I \end{bmatrix} \left(\begin{bmatrix} H_{u,s} & I \end{bmatrix} \begin{bmatrix} H_{u,s}^T \\ I \end{bmatrix} \right)^{-1} \begin{bmatrix} -H_{u,s} & I \end{bmatrix}, \\ \Pi^{1/2} \begin{bmatrix} H_{u,s} & -I \end{bmatrix} \begin{bmatrix} H_{u,s}^T \\ -I \end{bmatrix} \Pi^{1/2} = I, & \Pi = \left(\begin{bmatrix} H_{u,s} & I \end{bmatrix} \begin{bmatrix} H_{u,s}^T \\ I \end{bmatrix} \right)^{-1}. \end{aligned}$$

It turns out

$$\left\| r_{\mathcal{I}}(k) \right\| = \left\| \Pi^{1/2} \begin{bmatrix} H_{u,s} & -I \end{bmatrix} \begin{bmatrix} u_s(k) \\ y_s(k) \end{bmatrix} \right\| = \left\| r_s(k) \right\|.$$

Suppose that $\Delta_{H_{u,s}}$ represents the uncertainty in the system,

$$y_s(k) = \left(H_{u,s} + \Delta_{H_{u,s}} \right) u_s(k), \left\| \Pi^{1/2} \Delta_{H_{u,s}} \right\|_2 = \sigma_{\max} \left(\Pi^{1/2} \Delta_{H_{u,s}} \right) \leq \delta_{\Delta_K} < 1.$$

Define the residual evaluation function,

$$J(u_s, y_s) = \left\| r_{\mathcal{I}}(k) \right\| = \left\| r_s(k) \right\|.$$

It follows from (38) that the threshold is set equal to

$$J_{th}(u_s, y_s) = \frac{\delta_{\Delta_K}}{\sqrt{1 - \delta 2_{\Delta_K}}} \left(\left\| \begin{matrix} u_s \\ y_s \end{matrix} \right\|^2 - \left\| r_s \right\|^2 \right)^{1/2}.$$

Remark 2 *At the end of this subsection, we would like to give an interpretation of the orthogonal projection $P_{\mathcal{I}_G}$ in the context of reconstructing the system variables (u, y) and its relation to the latent variable v. It is apparent that*

$$\begin{bmatrix} \hat{u} \\ \hat{y} \end{bmatrix} := \mathcal{P}_{\mathcal{I}_G} \begin{bmatrix} u \\ y \end{bmatrix} = I_N I_N^* \begin{bmatrix} u \\ y \end{bmatrix}$$

is an estimation of (u, y) *for the nominal operations. Note that* I_N^* *is the conjugate of* I_N. *Let the state space representation of* I_N^* *be denote by*

$$\xi(k-1) = \bar{A}\xi(k) + \bar{B}\begin{bmatrix} u(k) \\ y(k) \end{bmatrix}, \xi \in \mathcal{R}^n, \begin{bmatrix} u(k) \\ y(k) \end{bmatrix} \in \mathcal{L}_2,$$
$$\varsigma(k) = \bar{C}\xi(k) + \bar{D}\begin{bmatrix} u(k) \\ y(k) \end{bmatrix} \in \mathcal{R}^p.$$

It is known that the above system is dual to I_N *and its output can be interpreted as a reconstruction of the input variable of* I_N, *i.e.* v *(Ding, 2020). In other words, the reconstruction of* (u, y) *is achieved by an estimation of the latent variable* v. *This interpretation is helpful to extend the projection-based detection method to nonlinear control systems. To this end, the so-called Hamiltonian extension of nonlinear systems and its application to the construction of normalised (nonlinear) image representations build useful tool (der Schaft, 2000; Ding, 2020). Moreover, aided by this interpretation, we will introduce, in the next subsection, explainable ML-based fault diagnosis methods.*

3.2 Complementary and explainable application of model-based and ML-based methods

In this sub-section, we would like to discuss about a complementary and explainable application of model-based and machine learning methods to enhancing the capability of fault diagnosis systems. To this end, we will demonstrate the realisation of the projection-based fault diagnosis schemes using the so-called auto-encoder method, a well-established ML-technique.

3.2.1 Auto-encoder technique: preliminaries

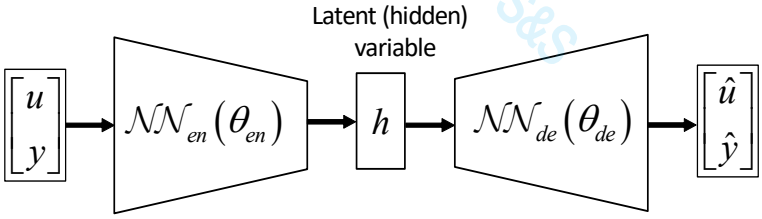

Figure 5: Basic configuration of an auto-encoder

As sketched in Figure 5, the essential function of an auto-encoder (AE) is to reconstruct (estimate) the system variables under consideration using NNs and learning mechanisms. In Figure 5, $\mathcal{NN}_{en}$ and $\mathcal{NN}_{de}$ represent two neural networks serving as encoder and decoder, respectively, and their parameters, θ_{en} and θ_{de}, are, roughly speaking, learnt using sufficient measurement data, (u, y), by minimising the loss function

$$\begin{aligned}\mathcal{L}(\theta_{en}, \theta_{de}) &= \left\| \begin{bmatrix} u \\ y \end{bmatrix} - \begin{bmatrix} \hat{u} \\ \hat{y} \end{bmatrix} \right\| = \left\| \begin{bmatrix} u \\ y \end{bmatrix} - \mathcal{NN}_{de}(\theta_{de}, h) \right\| \\ &= \left\| \begin{bmatrix} u \\ y \end{bmatrix} - \mathcal{NN}_{de}\left(\theta_{de}, \mathcal{NN}_{en}\left(\theta_{en}, \begin{bmatrix} u \\ y \end{bmatrix}\right)\right) \right\|\end{aligned}$$

with respect to θ_{en} and θ_{de}. The basic idea of applying an AE to fault detection can be schematically described as follows. Under assumption that the AE is well trained using fault-free operation data, the minimum value of $\mathcal{L}(\theta_{en}, \theta_{de})$ can be adopted as the threshold,

$$J_{th} = \min_{\theta_{en}, \theta_{de}} \mathcal{L}(\theta_{en}, \theta_{de}).$$

Running the trained AE online to generate projection-based residual r and computing the evaluation function J,

$$\begin{aligned} r &:= \begin{bmatrix} u \\ y \end{bmatrix} - \mathcal{NN}_{de}\left(\theta_{de}, \mathcal{NN}_{en}\left(\theta_{en}, \begin{bmatrix} u \\ y \end{bmatrix}\right)\right), \\ J &:= \|r\| = \left\| \begin{bmatrix} u \\ y \end{bmatrix} - \mathcal{NN}_{de}\left(\theta_{de}, \mathcal{NN}_{en}\left(\theta_{en}, \begin{bmatrix} u \\ y \end{bmatrix}\right)\right)\right\|, \end{aligned}$$

fault detection is then achieved by the detection logic

$$\begin{cases} J \leq J_{th}: & \text{fault-free} \\ J > J_{th}: & \text{faulty} \end{cases}.$$

It is well-known that the hidden variable h in an AE plays a central role as the information carrier of the system under consideration and, more importantly, in the context of the so-called information bottleneck (Bengio *et al.*, 2013; Geiger, 2021). Unfortunately, this aspect has been merely taken into account in most of AE applications to fault diagnosis issues. Typically, the hidden variable is viewed as features, as it is (generated) and as the output of the optimisation (training) process, without any explainable interpretation with regard to the system and the fault diagnosis problem under consideration. This motivates the work presented in the next sub-section.

3.2.2 AE-aided realisation of projection-based fault detection and estimation

The basic idea of applying AE technique to realise a projection-based fault detection consists in training the NNs to follow the major properties of an orthogonal projection onto the system image subspace. In the sequel, we briefly describe the conceptual realisation of the idea by means of two examples. For our purpose, recurrent neural networks are used for the realisation of dynamic systems, denoted by $\mathcal{RNN}_{en}$ and $\mathcal{RNN}_{de}$ for encoder and decoder.

Example 3 ***AE-aided realisation of projection-based fault detection.*** *Let $\mathcal{P}_{AE}$ defined by*

$$\begin{aligned} \begin{bmatrix} \hat{u} \\ \hat{y} \end{bmatrix} &:= \mathcal{P}_{AE} \begin{bmatrix} u \\ y \end{bmatrix}, \\ \mathcal{P}_{AE} \begin{bmatrix} u \\ y \end{bmatrix} &:= \mathcal{RNN}_{de}(\theta_{de}, h) = \mathcal{RNN}_{de}\left(\theta_{de}, \mathcal{RNN}_{en}\left(\theta_{en}, \begin{bmatrix} u \\ y \end{bmatrix}\right)\right) \end{aligned}$$

be an AE. Suppose that M batches of system data are available for the training purpose, and each of them includes N system data,

$$\mathcal{B}^{(i)} := \left\{ \begin{bmatrix} u^{(i)}(k_j) \\ y^{(i)}(k_j) \end{bmatrix}, j = 1, \cdots, N \right\}, i = 1, \cdots, M.$$

Given vectors $\alpha(k_j), \beta(k_j) \in \mathcal{R}^\kappa, j = 1, \cdots, N$, *let*

$$\|\alpha\|_2^2 = \sum_{j=1}^{N} \alpha^T(k_j)\,\alpha(k_j), \langle \alpha, \beta \rangle = \sum_{j=1}^{N} \alpha^T(k_j)\,\beta(k_j).$$

For the training purpose, a cost function consisting of three or four terms is defined,

$$\mathcal{L}(\theta_{en}, \theta_{de}) = \sum_{i=1}^{4} w_i \mathcal{L}_i(\theta_{en}, \theta_{de}), w_i > 0, \text{ weighting factors.}$$

Except the basic term,

$$\begin{aligned}\mathcal{L}_1(\theta_{en}, \theta_{de}) &= \frac{1}{M}\sum_{i=1}^{M}\left\|\begin{bmatrix} u^{(i)} \\ y^{(i)} \end{bmatrix} - \begin{bmatrix} \hat{u}^{(i)} \\ \hat{y}^{(i)} \end{bmatrix}\right\|_2^2 \\ &= \frac{1}{M}\sum_{i=1}^{M}\left\|\begin{bmatrix} u^{(i)} \\ y^{(i)} \end{bmatrix} - \mathcal{NN}_{de}\left(\theta_{de}, \mathcal{NN}_{en}\left(\theta_{en}, \begin{bmatrix} u^{(i)} \\ y^{(i)} \end{bmatrix}\right)\right)\right\|_2^2\end{aligned}$$

the following regularised terms are added:

- *realisation of idempotence operator* $\mathcal{P}_{AE}$ *(refer to (27))*

$$\mathcal{P}_{AE}\mathcal{P}_{AE} = \mathcal{P}_{AE} \Longrightarrow \tag{40}$$

$$\frac{1}{M}\sum_{i=1}^{M}\left\|\begin{bmatrix} \hat{u}^{(i)} \\ \hat{y}^{(i)} \end{bmatrix} - \mathcal{RNN}_{de}\left(\theta_{de}, \mathcal{RNN}_{en}\left(\theta_{en}, \begin{bmatrix} \hat{u}^{(i)} \\ \hat{y}^{(i)} \end{bmatrix}\right)\right)\right\|_2^2; \tag{41}$$

- *realisation of self-adjoint operator* $\mathcal{P}_{AE}$

$$\begin{aligned}&\left\langle \mathcal{P}_{AE}\begin{bmatrix} u^{(i)} \\ y^{(i)} \end{bmatrix}, \begin{bmatrix} u^{(j)} \\ y^{(j)} \end{bmatrix}\right\rangle = \left\langle \begin{bmatrix} u^{(i)} \\ y^{(i)} \end{bmatrix}, \mathcal{P}_{AE}\begin{bmatrix} u^{(j)} \\ y^{(j)} \end{bmatrix}\right\rangle \Longrightarrow \\ &\frac{1}{M^2}\sum_{i=1}^{M}\sum_{j=1}^{M}\left(\left\langle \begin{bmatrix} \hat{u}^{(i)} \\ \hat{y}^{(i)} \end{bmatrix}, \begin{bmatrix} u^{(j)} \\ y^{(j)} \end{bmatrix}\right\rangle - \left\langle \begin{bmatrix} u^{(i)} \\ y^{(i)} \end{bmatrix}, \begin{bmatrix} \hat{u}^{(j)} \\ \hat{y}^{(j)} \end{bmatrix}\right\rangle\right)^2;\end{aligned}$$

- *(optional) realisation of the normalised SIR*

$$\begin{aligned}&\left\|\begin{bmatrix} u \\ y \end{bmatrix}\right\|_2 = \left\|\begin{bmatrix} M_0 \\ N_0 \end{bmatrix} v\right\|_2 = \|v\|_2 \Longrightarrow \\ &\frac{1}{M}\sum_{i=1}^{M}\left(\left\|\begin{bmatrix} \hat{u}^{(i)} \\ \hat{y}^{(i)} \end{bmatrix}\right\|_2 - \|h^{(i)}\|_2\right)^2 \\ &= \frac{1}{M}\sum_{i=1}^{M}\left(\left\|\begin{bmatrix} \hat{u}^{(i)} \\ \hat{y}^{(i)} \end{bmatrix}\right\|_2 - \left\|\mathcal{RNN}_{en}\left(\theta_{en}, \begin{bmatrix} u^{(i)} \\ y^{(i)} \end{bmatrix}\right)\right\|_2\right)^2.\end{aligned}$$

It follows from the projection-based fault detection method that the (online) residual evaluation function and the threshold are defined by

$$J_N(u,y) = \frac{\left\| \begin{bmatrix} u \\ y \end{bmatrix} - \mathcal{RNN}_{de}\left(\theta_{de}, \mathcal{RNN}_{en}\left(\theta_{en}, \begin{bmatrix} u \\ y \end{bmatrix}\right)\right)\right\|_2^2}{\left\| \begin{bmatrix} u \\ y \end{bmatrix}\right\|_2^2},$$

$$J_{th,N}(u,y) = \frac{\delta}{1-\delta}\left(1 - \frac{\left\|\mathcal{RNN}_{de}\left(\theta_{de}, \mathcal{RNN}_{en}\left(\theta_{en}, \begin{bmatrix} u \\ y \end{bmatrix}\right)\right)\right\|_2^2}{\left\| \begin{bmatrix} u \\ y \end{bmatrix}\right\|_2^2}\right),$$

where δ denotes the value

$$\delta = \min_{\theta_{en},\theta_{de}} \frac{1}{M}\sum_{i=1}^{M}\left\| \begin{bmatrix} u^{(i)} \\ y^{(i)} \end{bmatrix} - \begin{bmatrix} \hat{u}^{(i)} \\ \hat{y}^{(i)} \end{bmatrix}\right\|_2^2$$

achieved by training.

This example clearly demonstrates that

- the objective of the construction and, in particular, the training of the AE is the realisation of the projection-based optimal fault detection;
- the hidden variable h can be interpreted as the so-called reference signal v in the context of SIR and image subspace, and this information is fully integrated in the training process. Considering that during fault-free operations the system variables (u, y) are uniquely determined by v and thus can be fully recovered using v without any redundancy, such an AE is optimal in the context of information bottleneck (Bengio *et al.*, 2013; Geiger, 2021);
- the trained AE is embedded in the residual evaluation and threshold computation as well, which, in most of the AE-based fault detection schemes, has not been incorporated.

As next example, we present a conceptual scheme of optimal fault estimation in dynamic systems. To this end, the fault estimation problem is firstly formulated in a general form: considering system dynamics described by

$$y = \mathcal{G}f, y \in \mathcal{L}_2^m, f \in \mathcal{L}_2^p, p > m, \tag{42}$$

find an estimator

$$\hat{f} = \mathcal{E}_f y, \tag{43}$$

where operator $\mathcal{G}$ represents the system dynamics, operator $\mathcal{E}_f$ a dynamic estimator, y is an m-dimensional measurement vector, and f denotes a p-dimensional unknown input vector that is called fault vector, but could also be cyber-attack signals or disturbances. It is well-known that the solution of (42) is not unique. We are interested in solving the

above estimation problem in the data-driven fashion, that is, instead of the system model $\mathcal{G}$, sufficient data, $\left(y^{(i)}\left(k_j\right), f^{(i)}\left(k_j\right)\right), j = 1, \cdots, N, i = 1, \cdots, M$, are available and used for the estimation purpose. Moreover, the estimate should be the so-called least squares (LS) estimation $\hat{f}_{LS}$, i.e.

$$\forall \hat{f} \text{ satisfying } y = \mathcal{G}\hat{f}, \left\|\hat{f}_{LS}\right\|_2 \leq \left\|\hat{f}\right\|_2,$$

with a specified confidence.

In the sequel, we first briefly introduce the model-based LS-solution, which serves as the basis for our AE-based algorithm. Let

$$\mathcal{G} = \mathcal{G}_{co} \circ \mathcal{G}_{ci}$$

be a co-inner-outer factorisation of $\mathcal{G}$ (Ding, 2020). Here $\mathcal{G}_{co}, \mathcal{G}_{ci}$ are co-outer and co-inner operators, respectively, satisfying

$$\mathcal{G}_{ci} \circ \mathcal{G}_{ci}^* = \mathcal{I}, \mathcal{Q} = \mathcal{G}_{co}^{-1} \text{ being stable and causal}$$

with $\mathcal{G}_{ci}^*$ as conjugate of $\mathcal{G}_{ci}$. It is well-known that

$$\hat{f}_{LS} = \mathcal{G}_{ci}^* \circ \mathcal{Q}y = \mathcal{G}_{ci}^* \circ \mathcal{G}_{ci} f \tag{44}$$

is the LS estimate of f. Furthermore, the estimation error,

$$\eta = f - \hat{f}_{LS} = \left(\mathcal{I} - \mathcal{G}_{ci}^* \circ \mathcal{G}_{ci}\right) f, \tag{45}$$

is defined as a specified confidence whose distribution and certain norm indicate the estimation performance.

Example 4 ***Optimal fault estimation in dynamic systems.*** *An AE-based realisation of the dynamic estimator (44) is schematically described in this example. As delineated in Figure 6, $\hat{f}_{LS}$ is achieved by means of two recurrent neural networks $\mathcal{RNN}_{\mathcal{Q}}\left(\theta_{\mathcal{Q}}\right)$ and $\mathcal{RNN}_{de}\left(\theta_{de}\right)$, where $\mathcal{RNN}_{de}\left(\theta_{de}\right)$ is the decoder trained in the AE for constructing $\mathcal{G}_{ci}^*$. The AE is trained using the data set $\left(y, f\right), \left(y^{(i)}\left(k_j\right), f^{(i)}\left(k_j\right)\right), j = 1, \cdots, N, i = 1, \cdots, M$, while the confidence η is generated based on the AE. To train the NNs, the total loss function $\mathcal{L}\left(\theta_{\mathcal{Q}}, \theta_{en}, \theta_{de}\right)$ consists of three terms and is set as follows:*

- $\mathcal{L}_1\left(\theta_{\mathcal{Q}}\right)$:

$$\mathcal{L}_1\left(\theta_{\mathcal{Q}}\right) = \frac{1}{M}\sum_{i=1}^{M}\left(\left\|\mathcal{RNN}_{\mathcal{Q}}\left(\theta_{\mathcal{Q}}, y^{(i)}\right)\right\|_2 - \left\|f^{(i)}\right\|_2\right)^2$$

  that minimises

$$\mathcal{Q}y = \mathcal{G}_{co}^{-1} \circ \mathcal{G}_{co} \circ \mathcal{G}_{ci} f \Longrightarrow \|\mathcal{Q}y\|_2 - \|f\|_2;$$

- $\mathcal{L}_2\left(\theta_{\mathcal{Q}}, \theta_{en}, \theta_{de}\right)$:

$$\mathcal{L}_2\left(\theta_{\varsigma}, \theta_{en}, \theta_{de}\right) = \frac{1}{M}\sum_{i=1}^{M}\left\| \begin{array}{c} \mathcal{RNN}_{de}\left(\theta_{de}, \mathcal{RNN}_{en}\left(\theta_{en}, f^{(i)}\right)\right) \\ -\mathcal{RNN}_{de}\left(\theta_{de}, \mathcal{RNN}_{\mathcal{Q}}\left(\theta_{\mathcal{Q}}, y^{(i)}\right)\right) \end{array} \right\|_2^2,$$

  which minimises

$$r = \mathcal{G}_{ci}^* \circ \mathcal{G}_{ci} f - \hat{f}_{LS} = \mathcal{G}_{ci}^* \circ \mathcal{G}_{ci} f - \mathcal{G}_{ci}^* \circ \mathcal{Q}y;$$

- $\mathcal{L}_3\left(\theta_{en}, \theta_{de}\right)$: *realisation of an AE-based orthogonal projection presented in the previous sub-section.*

The specified confidence could be computed using the (sample) distribution or a certain norm of the variable η.

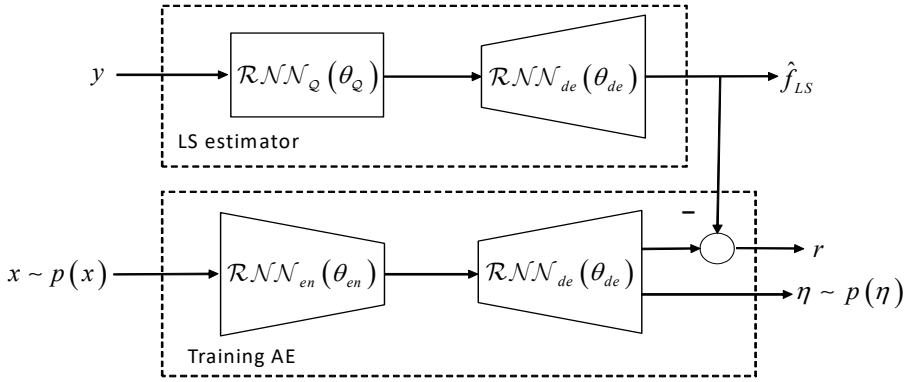

Figure 6: Schematic configuration of the fault estimator

3.2.3 A critical remark

The current enthusiasm for ML and big data technologies is significantly influencing the developments in the diagnosis research and engineering domains. It is a logic consequence that most of the existing ML methods and concepts have been introduced into this thematic field. Reviewing the course of this development, it seems that it is becoming a competition of publishing applications of newly developed ML-methods and algorithms to fault diagnosis. The consequence of this "copy-and-paste" style of research efforts is that very essential engineering requirements on diagnosis in automatic control systems have not been or cannot be fully considered in the use of ML-methods and algorithms. The reason is simple: the construction of most popular learning machines like deep NNs is less explainable, in particular in the context of diagnosis in dynamic systems. This issue becomes even more critical, when such methods are applied for the purpose of functional safety and cyber security. It is remarkable that explainability and interpretability build a very actual research focus in the ML-community (Burkart and Huber, 2021). This research endeavour is helpful for applying ML-based methods to diagnosis in automatic control systems. On the other hand, it should be kept in mind that, although enormously powerful and capable, ML-technology is a tool and its engineering applications should meet technical requirements and be explainable in the engineering context. In this regard, considerable efforts should be made to achieve diagnosis oriented explainable applications of ML-based methods. Our discussion and the examples in this sub-section have plainly documented that complementary and explainable application of model- and ML-based methods is a convincing way to develop advanced diagnosis methods towards enhancing functional safety and cyber security.

4 Performance degradation monitoring towards functional safety and cyber-security

Control performance monitoring is an application-driven research area and has its applications mainly in process industry (Bauer *et al.*, 2016). Roughly speaking, the essential tasks of control performance monitoring consist of assessment of control loop performance, detection of performance degradation and diagnosis of (component) faults (Ding and Li, 2021). Recently, new research efforts on POD can be observed (Li *et al.*, 2019; Li and Ding, 2020*b*), in which performance of automatic control systems is assessed at the system level and under various aspects like energy consumption, system reliability and safety etc. Moreover, dif-

ferent from the traditional efforts focused on recovering performance degradation caused by component faults (Perez *et al.*, 2003; Zhang and Jiang, 2003; Zhang *et al.*, 2008), advanced methods for control performance degradation monitoring and loop performance recovery have been reported (Li and Ding, 2020*b*; Li *et al.*, 2020; Ding and Li, 2021).

In this section, we address POD issues with a focus on residual-centred modelling and detection of system performance degradation.

4.1 Residual-centred system model

In (Ding, 2020), a so-called observer-based input-output model is introduced, which models the input-output dynamics of any LTI automatic control systems and is expressed, given the system nominal model (1)-(3), by

$$\hat{x}(k+1) = A\hat{x}(k) + Bu(k) + Lr(k), \tag{46}$$

$$y(k) = r(k) + C\hat{x}(k) + Du(k). \tag{47}$$

It is evident that the centrepiece of the above model is a state observer. Different from the state space model (2)-(3) that solely represents the nominal system dynamics, model (46)-(47) gives the system input-output dynamics also for the case that uncertainties exist in the system. As illustrated in (Ding, 2020), the influences of any uncertainties in the system are showcased by the residual vector r, which is available and accessible in the model (46)-(47). Moreover, in the light of the observer-based and residual-driven realisation of any feedback controllers introduced in Section 2,

$$u(z) = K(z)y(z) + v(z) = F\hat{x}(z) - Q(z)r_y(z) + \bar{v}(z), \tag{48}$$

any standard control loop shown in Figure 1 can be equivalently represented by the model (46)-(48), which is called residual-centred system model to underline the role of the residual vector in the model. Figure 7 showcases the equivalence between the standard control loop and its residual-centred model, in which Δ is used to denote system uncertainties schematically.

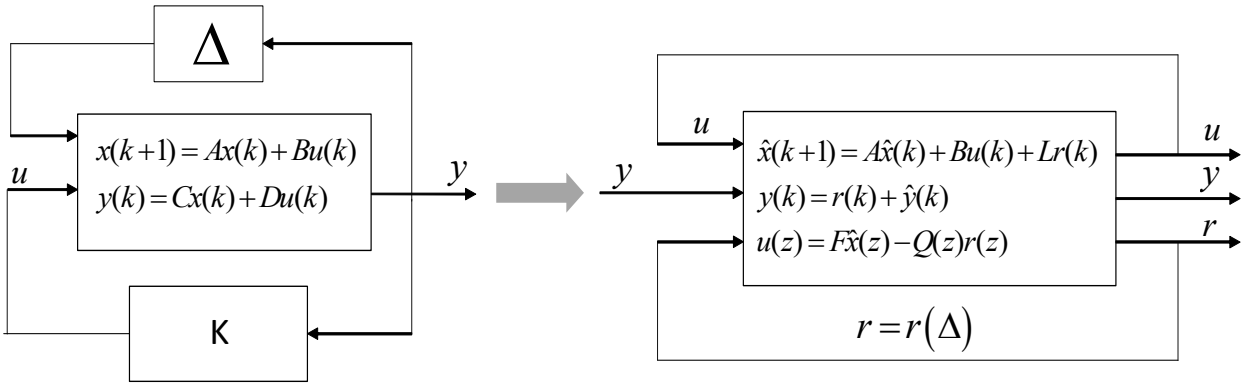

Figure 7: From the standard model to the observer-based I/O-model: a schematic description

The advantages of the residual-centred system model lie on hand:

- all system variables in the model, independent of the existence of any uncertainties, are accessible (for further computations),

- the implementation of the model is numerically reliable and stable, since only stable dynamics are concerned in the model, and
- with the embedded residual vector, the model is equipped with a capable indicator for the existence of uncertainties in the system.

The last function can be further ground using the projection-based method introduced in the previous section. According to (35), the l_2-norm of the residual vector generated by the normalised SKR (and the corresponding observer) is the distance of the measurement data (u, y) to the system image subspace and thus an indicator for the intensity of the uncertainty in the system. Accordingly,

$$\|r_y\|_2 = \left\| K_N \begin{bmatrix} u \\ y \end{bmatrix} \right\|_2 \tag{49}$$

is an indicator for the quality of the residual-centred model as well as system operation performance. It can, for instance, substitute the numerical involved algorithm for online estimation of gap metric and system stability margin adopted in (Li *et al.*, 2019).

Example 5 *In this example, we introduce a conceptual configuration of automatic control systems, which consists of four functional layers and is schematically sketched in Figure 8. "Information layer" is the core of the multi-layer configuration, whose centrepiece is the observer-based input-output model (46)-(47). Except for providing the needed online information for real-time control and diagnosis, various additional functionalities, in particular those safety and cyber security related ones, can be well integrated in this layer, for instance, serving as*

- *a fusing algorithm of sensor data,*
- *soft sensors for estimation of plant key variables,*
- *an encoder for encrypting the plant data as described in Section 2,*
- *an indicator for system uncertainties as given by (49).*

In "Real-time control and diagnosis layer", the standard (feedback) control and diagnosis algorithms described in Subsection 2.3 are performed. "Performance monitoring and optimisation layer" includes advanced performance degradation detection and recovery algorithms, for instance reported in (Li et al.*, 2019; Li and Ding, 2020*b*; Li* et al.*, 2020; Ding and Li, 2021) or described below. In "Learning and adaptation layer", ML-algorithms like the AEs introduced in Subsection 3.2 run aiming at updating the functional layers to match changes in the system.*

4.2 Functionality-oriented performance degradation monitoring

Consider system (1)-(3). Associated with it, the following Lyapunov equation provides us with a basic form of performance models for the system functionality and control,

$$S^T P S - P + Q = 0, P > 0, Q \geq 0, S \text{ is Schur.} \tag{50}$$

Here, matrices $S, Q \in \mathcal{R}^{n \times n}$ are functions of the system matrices (A, B, C) and state feedback gain matrix F, which are given corresponding to the following (representative) system functionalities and controller configuration:

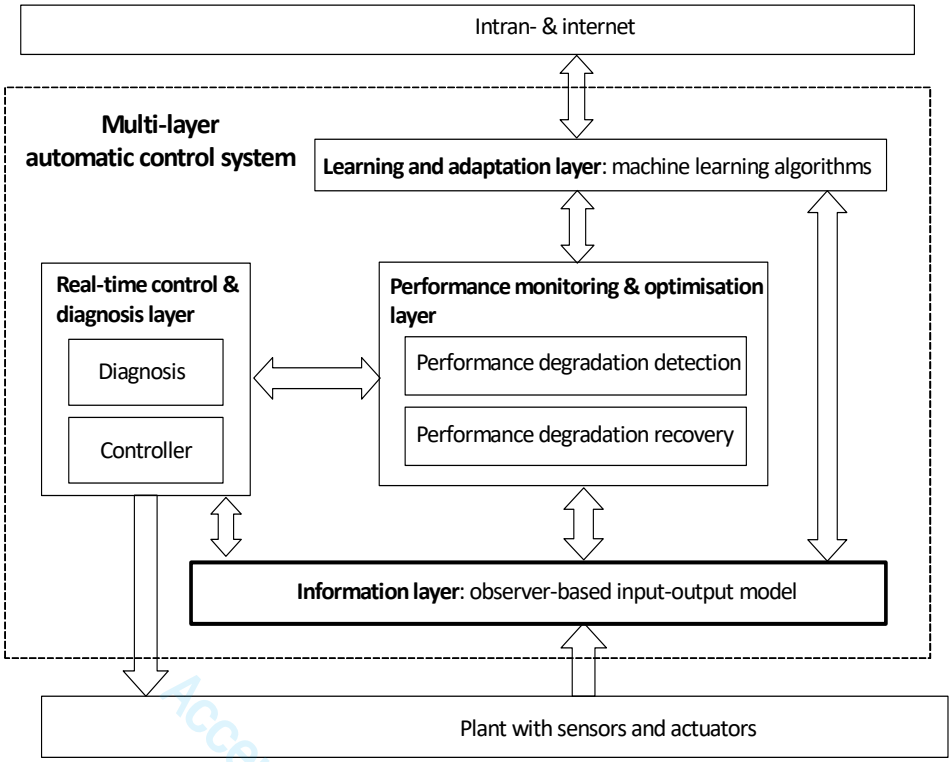

Figure 8: Schematic configuration of a multi-layer automatic control system

- for

$$S = A, Q = BB^T, \tag{51}$$

P as the solution of (50) is the controllability gramian that indicates the capability of the actuators,

- for

$$S = A^T, Q = C^T C, \tag{52}$$

P is the observability gramian indicating the capability of the sensors,

- for either (51) or (52), $\mathcal{H}_2$-norm of transfer function $C\left(zI - A\right)^{-1} B$ as performance can be assessed as follows

$$\left\|C\left(zI - A\right)^{-1} B\right\|_2 = tr^{1/2}\left(CPC^T\right) \text{ or } \left\|C\left(zI - A\right)^{-1} B\right\|_2 = tr^{1/2}\left(B^T PB\right),$$

- for

$$S = A + BF, Q = Q_0 + F^T RF, R > 0, Q_0 \geq 0,$$

performance of an LQ state feedback controller, $u = Fx$,

$$J\left(k\right) = \sum_{i=k}^{\infty} \left(x^T(i) Q_0 x(i) + u^T\left(i\right) Ru\left(i\right)\right) = x^T(k) P x(k),$$

is assessed.

There exist several strategies to monitor the above-described system performance. Assume that the system dynamics is governed by

$$x(k+1) = Sx(k),$$

and $x(k)$ is measurable. Define

$$J(k) = \sum_{i=k}^{\infty} x^T(i)Qx(i) = x^T(k)Px(k).$$

It holds

$$J(k+1) + x^T(k)Qx(k) - J(k) = 0 \tag{53}$$

during degradation-free operations. Hence, introducing a performance residual r_p defined by

$$r_p(k) := x^T(k+1)Px(k+1) + x^T(k)(Q-P)x(k),$$

performance degradation can be detected using standard residual-based detection schemes. This endeavour is unfortunately limited to a theoretical concept and often vain in practical applications due to its minor detection capability and strict constraints on the system dynamics. Aiming at improving the detection performance, (Li *et al.*, 2022) have proposed a sophisticated detection scheme, which is briefly described in the sequel.

By means of a vectorisation of P matrix, re-write the performance model

$$\begin{aligned} J(k) = x^T(k)Qx(k) + J(k+1) \Longrightarrow \\ x^T(k)Px(k) - x^T(k+1)Px(k+1) = x^T(k)Qx(k) \end{aligned} \tag{54}$$

as

$$\left(x^T(k) \otimes x^T(k) - x^T(k+1) \otimes x^T(k+1)\right) D_n hvec(P) = x^T(k)Qx(k). \tag{55}$$

In the above equation, $hvec(P)$ denotes a half-vectorisation of the symmetric matrix $P \in \mathcal{R}^{n\times n}$, represents the $\frac{n(n+1)}{2}$ parameters to be identified (considering $P = P^T$) and satisfies

$$D_n hvec(P) = vec(P), hvec(P) \in \mathcal{R}^{\frac{n(n+1)}{2}}, D_n \in \mathcal{R}^{n^2 \times \frac{n(n+1)}{2}}$$

with D_n being the so-called duplication matrix (Magnus, 1988). Notation $\otimes$ stands for the Kronecker product. Suppose that sufficient number of data, $x(k+i), i = 0, \cdots, N$, are collected, which enables us to write (55) into

$$\Psi hvec(P) = \phi, \tag{56}$$

$$\Psi = \begin{bmatrix} x^T(k) \otimes x^T(k) - x^T(k+1) \otimes x^T(k+1) \\ \vdots \\ x^T(k+N-1) \otimes x^T(k+N-1) - x^T(k+N) \otimes x^T(k+N) \end{bmatrix},$$

$$\phi = \begin{bmatrix} x^T(k)Qx(k) \\ \vdots \\ x^T(k+N-1)Qx(k+N-1) \end{bmatrix}.$$

As a result, on the assumption of sufficient excitation, matrix P can be identified using e.g. a standard LS estimation algorithm. If the difference between the identified and the nominal

goes beyond a decision threshold, performance degradation is declared. Considering that the solution of (50) is a symmetric positive definite (SPD) matrix, the Riemannian metric method (Ding, 2020; Li *et al.*, 2022) can be applied to achieve an efficient degradation detection. In (Ding, 2020), variations of the above algorithm are provided to solve the similar performance degradation problems using system output data $y(k)$ instead of the state variable $x(k)$.

Note that the above presented detection schemes are limited to the case that $u = Fx$. Although extensions have been proposed in (Ding, 2020), a general solution for arbitrary input u remains to be an open issue. In the following example, we present a conceptual solution for performance degradation detection.

Example 6 *For the simplicity, we only consider controllability gramian as functionality performance with system model*

$$x(k+1) = Ax(k) + Bu(k), A \text{ is Schur}$$

and a function

$$J(k) = \sum_{i=k}^{\infty} \left(x^T(i) Q x(i) + \bar{u}^T(i) R \bar{u}(i) \right), Q = BB^T, \bar{u}(i) = \begin{cases} u(k), i = k, \\ 0, i > k. \end{cases}$$

It yields

$$\begin{aligned} J(k) &= x^T(k) Q x(k) + u^T(k) R u(k) + x^T(k+1) P x(k+1) \\ &= \begin{bmatrix} x^T(k) & u^T(k) \end{bmatrix} \begin{bmatrix} A^T P A + Q & A^T P B \\ B^T P A & R + B^T P B \end{bmatrix} \begin{bmatrix} x(k) \\ u(k) \end{bmatrix} \\ &= \begin{bmatrix} x^T(k+1) & 0 \end{bmatrix} \begin{bmatrix} A^T P A + Q & A^T P B \\ B^T P A & R + B^T P B \end{bmatrix} \begin{bmatrix} x(k+1) \\ 0 \end{bmatrix} \\ &\quad + \begin{bmatrix} x^T(k) & u^T(k) \end{bmatrix} \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} x(k) \\ u(k) \end{bmatrix}, \\ &\quad A^T P A - P + Q = 0, P > 0, \end{aligned}$$

which can be further written into

$$\begin{aligned} &\begin{bmatrix} x^T(k) & u^T(k) \end{bmatrix} \Phi \begin{bmatrix} x(k) \\ u(k) \end{bmatrix} - \begin{bmatrix} x^T(k+1) & 0 \end{bmatrix} \Phi \begin{bmatrix} x(k+1) \\ 0 \end{bmatrix} \\ &\quad = \begin{bmatrix} x^T(k) & u^T(k) \end{bmatrix} \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} x(k) \\ u(k) \end{bmatrix}, \qquad (57) \\ &\quad \Phi = \begin{bmatrix} A^T P A + Q & A^T P B \\ B^T P A & R + B^T P B \end{bmatrix}. \qquad (58) \end{aligned}$$

Note that (57) is of the identical form with (54). Consequently, applying the same procedure with (55)-(56), matrix Φ can be identified, which then enables a reliable performance degradation detection. It is noteworthy that Φ contains more information than P, which can be adopted for monitoring other system performance as well. For instance, given $Q = C_s^T C_s$ and R, the value

$$q = tr^{1/2} \left(\hat{\Phi}(2,2) - R \right)$$

with $\hat{\Phi}(2,2) = R + B^T P B$ *denoting the identified sub-block of matrix* Φ, *gives an estimation of* $\mathcal{H}_2$*-norm of transfer function* $C_s (zI - A)^{-1} B$, *i.e.*

$$q = \left\| C_s (zI - A)^{-1} B \right\|_2$$

which could e.g. represent the system dynamics from u *to a certain sensor block modelled by* $C_s x$.

Remark 3 *Even though only LTI systems are considered in the schemes introduced above, the ideas can be well adopted to address performance degradation monitoring of nonlinear control systems. Below, we schematically outline the conceptual steps of approaching solutions. Let the system performance under monitoring be*

$$J(k) = \sum_{i=k}^{\infty} q(x(k), u(k)).$$

Analogue to (53), it holds

$$J(k+1) + q(x(k), u(k)) - J(k) = 0. \tag{59}$$

On the assumption that $J(k)$ *as solution of (59) could be approximated by*

$$J(k) = \sum_{i=i}^{N} w_i \phi_i (x(k), u(k)), \tag{60}$$

where $\{\phi_i (x(k), u(k)), i = 1, \cdots, N\}$ *is the set of some basic functions and* $w_i, i = 1, \cdots, N$, *are weights (Parr* et al.*, 2008), difference equation (59) is re-written into*

$$\sum_{i=i}^{N} w_i \left(\phi_i (x(k), u(k)) - \phi_i (x(k+1), u(k+1)) \right) = q(x(k), u(k)). \tag{61}$$

Equation (61) is similar to (54) and can serve as a performance model. During online operations, the system performance can be assessed by an online identification of weights $w_i, i = 1, \cdots, N$, *and computation of* $J(k)$ *according to (60). It is noteworthy that the performance value function* $J(k)$ *can be generally approximated using NNs (Al-Tamimi* et al.*, 2008).*

At the end of this subsection, we would like to call the reader's attention to the fact that application of the aforementioned schemes requires knowledge of the system state vector $x(k)$, which is, unfortunately, not available in most of real practical applications. It is an open and challenging issue to realise those performance degradation monitoring schemes using system data (u, y) instead of the state vector x. In (Ding, 2020), this issue has been investigated.

4.3 Performance degradation monitoring in the probabilistic setting

Considering that the performance degradation schemes presented in the previous sub-section are based on the assumption of ideal system models without uncertainty, adaptations are

needed before they are efficiently applied in practice. Although their extensions to systems with normally distributed process and measurement noises have been addressed in (Ding, 2020), efficient handling of model uncertainties remains to be an open issue. Recently, (Xue *et al.*, 2020; Shang *et al.*, 2021; Wan *et al.*, 2021) have proposed to apply the so-called distributionally robust optimisation (DRO) technique (Rahimian and Mehrotra, 2019; Lin *et al.*, 2022) to enhancing the robustness of fault detection systems against model uncertainties. In particular, it is advantageous that DRO technique enables handlings and solutions in a probabilistic setting. In this subsection, we briefly introduce the ideas of applying DRO technique to performance degradation detection by means of two examples.

In the sequel, notation Ξ is adopted for support, $\mathbb{P}$ is used for probability. $\mathbb{P}_\xi$ and $\mathbb{E}_{\mathbb{P}_\xi}$ represent probability distribution of ξ and expectation taken with respect to ξ following $\mathbb{P}_\xi$.

Example 7 *In this example, we delineate a data-driven realisation of performance indicator (49) in the probabilistic setting. Departing from the system model*

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + \omega(k), \\ y(k) &= Cx(k) + Du(k) + \upsilon(k) \end{aligned}$$

with $\omega(k), \upsilon(k)$ being the process and measurement noise vectors, the system dynamics are written as

$$y_s(k) = \Gamma_s x\left(k-s\right) + H_{u,s}u_s(k) + H_{\omega,s}\omega_s(k) + \upsilon_s(k), \tag{62}$$

where $y_s(k), u_s(k), \Gamma_s, H_{u,s}$ are given in Example 2, and $\omega_s(k), \upsilon_s(k), H_{\omega,s}$ are as follows

$$H_{\omega,s} = \begin{bmatrix} 0 & 0 & & \\ C & \ddots & \ddots & \\ \vdots & \ddots & \ddots & 0 \\ CA^{s-1} & \cdots & C & 0 \end{bmatrix} \in \mathcal{R}^{(s+1)m\times(s+1)n},$$

$$\omega_s(k) = \begin{bmatrix} \omega(k-s) \\ \vdots \\ \omega(k) \end{bmatrix}, \upsilon_s(k) = \begin{bmatrix} \upsilon(k-s) \\ \vdots \\ \upsilon(k) \end{bmatrix}.$$

To simplify our study, assume that the system is stable, $x\left(k-s\right)$ is a random vector and $\phi_s(k)$ is a wide sense stationary (w.s.s) stochastic process. We then further write (62) into

$$\begin{aligned} y_s(k) &= H_{u,s}u_s(k) + \phi_s(k), \\ \phi_s(k) &= \Gamma_s x\left(k-s\right) + H_{\omega,s}\omega_s(k) + \upsilon_s(k). \end{aligned} \tag{63}$$

Using the results presented in Example 2, the projection-based residual vector and the corresponding evaluation function are equivalently realised as follows

$$\begin{aligned} r_{\mathcal{I}}(k) &= \begin{bmatrix} u_s(k) \\ y_s(k) \end{bmatrix} - \mathcal{P}_{\mathcal{I}} \begin{bmatrix} u_s(k) \\ y_s(k) \end{bmatrix}, \\ r_s(k) &= \Pi^{1/2}\left(y_s(k) - H_{u,s}u_s(k)\right), \|r_{\mathcal{I}}(k)\| = \|r_s(k)\|. \end{aligned}$$

Note that $r_s(k)$ can be written as

$$r_s(k) = \Pi^{1/2}\left(\Delta H_{u,s}u_s(k) + \phi_s(k)\right) =: \Pi^{1/2}\bar{\phi}_s,$$

where $\Delta H_{u,s}$ represents uncertainty in the system, which leads to

$$\|r_{\mathcal{I}}(k)\|^2 = \|r_s(k)\|^2 = \bar{\phi}_s^T \Pi \bar{\phi}_s.$$

Suppose that the distribution of unknown random vector $\bar{\phi}_s$ belongs to the moment-based ambiguity set (Lin et al.*, 2022)*

$$\mathcal{D}(\gamma_1, \gamma_2) = \left\{ \mathbb{P}_{\bar{\phi}_s} \left| \begin{array}{c} \mathbb{P}\left(\bar{\phi}_s \in \Xi\right) = 1 \\ \left(\mathbb{E}_{\mathbb{P}_{\bar{\phi}_s}}\left\{\bar{\phi}_s\right\} - \mu_0\right)^T \Sigma_0^{-1} \left(\mathbb{E}_{\mathbb{P}_{\bar{\phi}_s}}\left\{\bar{\phi}_s\right\} - \mu_0\right) \leq \gamma_1 \\ \mathbb{E}_{\mathbb{P}_{\bar{\phi}_s}}\left\{\left(\mathbb{E}_{\mathbb{P}_{\bar{\phi}_s}}\left\{\bar{\phi}_s\right\} - \mu_0\right)\left(\mathbb{E}_{\mathbb{P}_{\bar{\phi}_s}}\left\{\bar{\phi}_s\right\} - \mu_0\right)^T\right\} \leq \gamma_2 \Sigma_0 \end{array} \right. \right\},$$

where vector μ_0, matrix Σ_0, and constants $\gamma_1 \geq 0, \gamma_2 \geq 1$ are estimated using the sufficient number of collected data and thus assumed to be known. It is obvious that threshold setting

$$J_{th} = \sup_{\bar{\phi}_s} \bar{\phi}_s^T \Pi \bar{\phi}_s$$

would result in considerably conservative performance degradation detection. More reasonable setting can be achieved in the probabilistic setting as follows,

$$\forall \mathbb{P}_{\bar{\phi}_s} \in \mathcal{D}(\gamma_1, \gamma_2), \mathbb{P}\left(\bar{\phi}_s^T \Pi \bar{\phi}_s > J_{th}\right) \leq \alpha,$$

where α is a tolerable upper bound of false alarm rate. In this context, the probabilistic performance degradation problem is formulated as: given $\alpha \in (0,1)$, solve

$$\min \beta =: J_{th} \tag{64}$$

$$\sup_{\mathbb{P}_{\bar{\phi}_s} \in \mathcal{D}(\gamma_1, \gamma_2)} \mathbb{P}\left(\bar{\phi}_s^T \Pi \bar{\phi}_s > \beta\right) \leq \alpha \tag{65}$$

for the threshold J_{th}. The DRO problem (64)-(65) can be solved using well-established DRO technique, see e.g. (Shang et al.*, 2021; Lin* et al.*, 2022).*

Example 8 *Consider observer-based input-output model (46)-(47). Suppose that $u(k) = F\hat{x}(k)$, and the residual vector is a w.s.s. stochastic process over the time interval $[k-s, k)$, and its (unknown) distribution belongs to the moment-based ambiguity set*

$$\mathcal{D}(\gamma_1, \gamma_2) = \left\{ \mathbb{P}_{r_{s-1}} \left| \begin{array}{c} \mathbb{P}\left(r_{s-1} \in \Xi\right) = 1 \\ \mathbb{E}^T_{\mathbb{P}_{r_{s-1}}}\left\{r_{s-1}\right\} \Sigma_0^{-1} \mathbb{E}_{\mathbb{P}_{r_{s-1}}}\left\{r_{s-1}\right\} \leq \gamma_1 \\ \mathbb{E}_{\mathbb{P}_{r_{s-1}}}\left\{\mathbb{E}_{\mathbb{P}_{r_{s-1}}}\left\{r_{s-1}\right\} \mathbb{E}^T_{\mathbb{P}_{r_{s-1}}}\left\{r_{s-1}\right\}\right\} \leq \gamma_2 \Sigma_0 \end{array} \right. \right\},$$

$$r_{s-1} = \begin{bmatrix} r(k-s) \\ \vdots \\ r(k-1) \end{bmatrix},$$

where s is a sufficiently large integer so that $A_F^s \approx 0$. We would like to call the reader's attention to the random vector r_{s-1}. As described in Sub-section 4.1, it represents uncertainties in the system, including noises and model uncertainty. Define cost function for control performance assessment as

$$J(k) = \mathbb{E} \sum_{i=k}^{\infty} \hat{x}^T(i) Q \hat{x}(i) = \hat{x}^T(k) P \hat{x}(k), \tag{66}$$

$$A_F^T P A_F - P + Q = 0, P > 0, \tag{67}$$

It follows from (46) that

$$\begin{aligned} \hat{x}(k) &= A_F^s \hat{x}(k-s) + r_x(k) \approx r_x(k), \\ r_x(k) &= \Theta r_{s-1}, \Theta = \left[\begin{array}{cccc} A_F^{s-1}L & \cdots & A_F L & L \end{array} \right]. \end{aligned} \tag{68}$$

Assume that Θ is of full row-rank. The moment-based ambiguity set of r_x is given by

$$\begin{aligned} \mathcal{D}_{r_x}(\gamma_3, \gamma_4) &= \left\{ \mathbb{P}_{r_x} \left| \begin{array}{c} \mathbb{P}(r_x \in \Xi) = 1 \\ \mathbb{E}_{\mathbb{P}_{r_x}}^T \{r_x\} \bar{\Sigma}_0^{-1} \mathbb{E}_{\mathbb{P}_{r_x}} \{r_x\} \leq \gamma_3 \\ \mathbb{E}_{\mathbb{P}_{r_x}} \left\{ \mathbb{E}_{\mathbb{P}} \{r_x\} \mathbb{E}_{\mathbb{P}_{r_x}} \{r_x\}^T \right\} \leq \gamma_4 \bar{\Sigma}_0 \end{array} \right. \right\}, \\ \bar{\Sigma}_0 &= \Theta \Sigma_0 \Theta^T, \end{aligned}$$

where γ_3, γ_4 and $\bar{\Sigma}_0$ are known. The probabilistic performance degradation detection problem is then formulated as: given $\alpha \in (0,1)$, solve

$$\min \beta =: J_{th} \tag{69}$$

$$\sup_{\mathbb{P}_{r_x} \in \mathcal{D}_{r_x}(\gamma_3, \gamma_4)} \mathbb{P}\left(r_x^T P r_x > \beta\right) \leq \alpha \tag{70}$$

for the threshold J_{th}.

The above two examples showcase that DRO technique can serve as a powerful tool to deal with performance degradation detection issues efficiently. It is noteworthy that various ambiguity sets are investigated in the DRO framework (Lin *et al.*, 2022), which enables us to handle different types of model uncertainties and study performance degradation detection issues both in model-based and data-driven fashions. A further aspect is to address safety issues in a probabilistic setting (Yang, 2018). For instance, let

$$\mathcal{S}_x = \{x \,|\, g_i(x) \leq 0, i = 1, \cdots, \kappa\}$$

denote the set of the system state variables that are in the safe region defined by the safety requirements $g_i(x) \leq 0, i = 1, \cdots, \kappa$. Then, the probability,

$$\mathbb{P}(x \in \mathcal{S}_x) > \beta >> 0, \tag{71}$$

can be, as a constraint, embedded in a probabilistic performance degradation detection and recovery problem.

5 Conclusion

In this note, we have discussed about diagnosis and performance degradation detection issues from an integrated viewpoint of functionality maintenance and cyber security of automatic control systems. Three aspects have been addressed:

- application of control and detection unified framework to enhancing diagnosis capability of feedback control systems, in which the functionalisation of the control system plays an essential role. It is showcased that rational utilisation of the residual signal as an information provider and cyber security oriented configuration of functional units of the control system promises enhanced capacity of detecting technical faults and cyber attacks, and preventing attackers to gain system knowledge by means of system identification using the transmitted data;

- projection-based technique of detecting faults in dynamic systems, which is based on an orthogonal projection of the system data onto the system image and kernel subspaces. This technique is more capable than the well-established observer-based schemes in dealing with detecting faults in dynamic systems. And more importantly, it enables explainable applications of ML-based technique like AE methods to diagnosis. It is illustrated that complementary application of model- and ML-based methods is the future of the diagnosis technique for industrial automatic control systems;

- system performance degradation detection, which is of elemental importance for industrial CPSs and, unfortunately, has received less attention in the research domain. The residual-centred model form for dynamic systems is a useful system tool to deal with performance degradation detection issues. Moreover, some performance degradation monitoring schemes are introduced, whose core, roughly speaking, is modelling of system performance and online identification of the associated model parameters. It is demonstrated that by means of DRO technique, performance degradation detection can be handled in a probabilistic setting, which enables an efficient and more reliable degradation detection.

We have reported ideas, presented conceptual schemes, and illustrated by means of examples why research efforts in these three aspects could contribute to the future development of capable monitoring and diagnosis methods towards enhancing functionality safety and cyber security of automatic control systems. We would like to mention that a number of the basic design schemes and algorithms reported in this note have been successfully tested on laboratory systems, including

- application of the control and detection unified framework to cyber-attack detection in three-tank control system (Ding *et al.*, 2021),

- projection-based fault detection in three-tank control system (Ding *et al.*, 2022),

- DRO technique-based fault detection in three-tank control system (Xue *et al.*, 2020; Shang *et al.*, 2021),

- performance degradation monitoring and recovery of vision-based inverted pendulum control system (Xu *et al.*, 2021).

The focus of this note is on diagnosis and performance degradation detection issues. So far, key maintenance technologies like condition monitoring (CM), prognostics and health management (PHM), performance degradation recovery (PDR) or fault-tolerant control (FTC) are not addressed. The interested reader is referred to (Liao and Köttig, 2014; Lei *et al.*, 2018; Si *et al.*, 2020; Ding, 2020; Ding and Li, 2021; Hwang *et al.*, 2010; Yin *et al.*, 2016) and references cited therein. We would like to emphasise the two aspects of fault diagnosis and performance degradation monitoring in automatic control systems. On the one hand, it builds the technical basis and an indispensable part of technologies like CM, PHM, PDR and FTC. Consequently, its development is significantly stamped by progresses in these technologies. On the other hand, as a basic function of today's automatic control systems, fault diagnosis and performance monitoring should match ongoing developments in automatic control systems. CPS, internet of things (IoT) and cloud computing as a service are

the key technologies that will decisively impact the evolution of automatic control systems in the era of industry 4.0. In this context, integrated study on functional safety and cyber security of automatic control systems is of essential importance. Our work reported in this note is a contribution to this study.

Acknowledgement: The author is very grateful to Dr.-Ing. L. Li for the collaborative work on the unified framework of control and detection as well as on the projection-based detection methods, to Dr.-Ing. Z. Chen for the valuable contributions to ML-methods and AE-based realisation of projection methods, and to Dr. D. Zhao for the intensive and valuable discussions on cyber security issues. Also, the author is thankful to the anonymous reviewers for their valuable and constructive comments and suggestions.

References

Al-Tamimi, A., F. L. Lewis and M. Abu-Khalaf (2008). Discrete-time nonlinear hjb solution using approximate dynamic programming: Convergence proof. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* **38**(4), 943–949.

Bauer, M., A. Horch, L. Xie, M. Jelali and N. Thornhill (2016). The current state of control loop performance monitoring, a survey of application in industry. *Journal of Process Control* **38**, 1 – 10.

Bengio, Y., A. Courville and P. Vincent (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **35**(8), 1798–1828.

Burkart, N. and M. F. Huber (2021). A survey on the explainability of supervised machine learning. *Journal of Artificial Intelligence Research* **70**, 245–317.

der Schaft, A. Van (2000). *L2 - Gain and Passivity Techniques in Nonlinear Control*. Springer. London.

Dibaji, S. M., M. Pirani, D. B. Flamholz, A. M. Annaswamy, K. H. Johansson and A. Chakrabortty (2019). A systems and control perspective of CPS security. *Annual Reviews in Control* **47**, 394–411.

Ding, D., Q.-L. Han, Y. Xiang, X. Ge and X.-M. Zhang (2018). A survey on security control and attack detection for industrial cyber-physical systems. *Neurocomputing* **275**, 1674–1683.

Ding, S. X. (2008). *Model-Based Fault Diagnosis Techniques - Design Schemes, Algorithms, and Tools*. Springer-Verlag.

Ding, S. X. (2020). *Advanced Methods for Fault Diagnosis and Fault-tolerant Control*. Springer-Verlag. Berlin.

Ding, S. X. and L. Li (2021). Control performance monitoring and degradation recovery in automatic control systems: A review, some new results, and future perspectives. *Control Engineering Practice* **111**, 104790.

Ding, S. X., G. Yang, P. Zhang, E.L. Ding, T. Jeinsch, N. Weinhold and M. Schulalbers (2010). Feedback control structures, embedded residual signals and feedcak control schemes with an integrated residual access. *IEEE Trans. on Contr. Syst. Tech.* **18**, 352–367.

Ding, S. X., L. Li and T. Liu (2022). An alternative paradigm of fault diagnosis in dynamic systems: orthogonal projection-based methods. *arXiv:2202.08108*.

Ding, S. X., L. Li, D. Zhao, Ch. Louen and T. Liu (2021). Application of the unified control and detection framework to detecting stealthy integrity cyber-attacks on feedback control systems. *arXiv:2103.00210*.

Ding, S. X., P. Zhang, S. Yin and E. L. Ding (2013). An integrated design framework of fault-tolerant wireless networked control systems for industrial automatic control applications. *IEEE Transactions on Industrial Informatics* **9**(1), 462–471.

Feintuch, A. (1998). *Robust Control Theory in Hilbert Space*. Springer-Verlag. New York.

Francis, B. A. (1987). *A Course in H-Infinity Control Theory*. Springer-Verlag. Berlin – New York.

Frank, P. M. (1990). Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy - a survey. *Automatica* **26**, 459–474.

Frank, P. M. and X. Ding (1997). Survey of robust residual generation and evaluation methods in observer-based fault detection systems. *Journal of Process Control* **7(6)**, 403–424.

Gao, Z. W., C. Cecati and S. X. Ding (2015). A survey of fault diagnosis and fault-tolerant techniques, part i: Fault diagnosis with model-based and signal-based approaches. *IEEE Trans. on Industrial Electronics* **62**, 3757–3767.

Geiger, B. C. (2021). On information plane analyses of neural network classifiers–a review. *IEEE Transactions on Neural Networks and Learning Systems* pp. 1–13.

G.Ferrari, R. M. and A. M. H.Teixeira (2021). A switching multiplicative watermarking scheme for detection of stealthy cyber-attacks. *IEEE Transactions on Automatic Control* **66**(6), 2558–2573.

Giraldo, J., D. Urbina, A. Cardenas, J. Valente, M. Faisal, J. Ruths, N. O. Tippenhauer, H. Sandberg and R. Candell (2018). A survey of physics-based attack detection in cyber-physical systems. *ACM Comput. Surv.*

Griffioen, P., S. Weerakkody and B. Sinopoli (2021). A moving target defense for securing cyber-physical systems. *IEEE Transactions on Automatic Control* **66**, 2016–2031.

Han, H., Y. Yang, L. Li and S. X. Ding (2019). Control performance-based fault detection and fault-tolerant control schemes for a class of nonlinear systems. *International Journal of Robust and Nonlinear Control* **30**(4), 1431–1450.

Han, H., Y. Yang, L. Li and S. X. Ding (2021). Performance-based fault detection and fault-tolerant control for nonlinear systems with t-s fuzzy implementation. *IEEE Trans. on Cybernetics* **51**, 801–814.

Hoffmann, J. W. (1996). Normalized coprime factorizations in continuous and discrete time – a joint state-space approach. *IMA Journal of Mathematical Control and Information* **13**(4), 359–384.

Hwang, I., S. Kim, Y. Kim and C.E. Seah (2010). A survey of fault detection, isolation, and reconfiguration methods. *IEEE Trans. Contr. Syst. Tech.* **18**, 636–653.

Kato, T. (1995). *Perturbation Theory for Linear Operators*. Springer-Verlag. Berlin.

Lei, Y., N. Li, L. Guo, N. Li, T. Yan and J. Lin (2018). Machinery health prognostics: A systematic review from data acquisition to rul prediction. *Mechanical Systems and Signal Processing* **104**, 799–834.

Li, L. and S. X. Ding (2020*a*). Gap metric techniques and their application to fault detection performance analysis and fault isolation schemes. *Automatica* **118**, 109029.

Li, L. and S. X. Ding (2020*b*). Performance supervised fault detection schemes for industrial feedback control systems and their data-driven implementation. *IEEE Trans. on Industrial Informatics* **16**(4), 2849–2858.

Li, L., H. Luo, S. X. Ding, Y. Yang and K. Peng (2019). Performance-based fault detection and fault-tolerant control for automatic control systems. *Automatica* **99**, 309–316.

Li, L., S. Li, S. X. Ding, X. Peng and K. Peng (2022). Riemannian metric based performance monitoring and diagnosis for a class of feedback control systems. *Acta Automatica Sinica* p. doi:10.16383/j.aas.c210027.

Li, L., S. X. Ding, H. Luo, K. Peng and Y. Yang (2020). Performance-based fault-tolerant control approaches for industrial processes with multiplicative faults. *IEEE Trans. on Industrial Informatics* **16**(7), 4759–4768.

Liao, L. and F. Köttig (2014). Review of hybrid prognostics approaches for remaining useful life prediction of engineered systems, and an application to battery life prediction. *IEEE Transactions on Reliability* **63**(1), 191–207.

Lin, F., X. Fang and Z. Gao (2022). Distributionally robust optimization: A review on theory and applications. *Numerical Algebra, Control & Optimization* **12**(1), 159–212.

Magnus, J. R. (1988). *Linear Structures*. Oxford University Press.

Mo, Y., S. Weerakkody and B. Sinopoli (2015). Physical authentication of control systems: Designing watermarked control inputs to detect counterfeit sensor outputs. *IEEE Control Systems Magazine* **35**, 93–109.

Parr, R., L. Li, G. Taylor, C. Painter-Wakefield and M. L. Littman (2008). An analysis of linear models, linear value-function approximation, and feature selection for reinforcement learning. In: *Proceedings of the 25th International Conference on Machine Learning*. ICML '08. Association for Computing Machinery. New York, NY, USA. p. 752–759.

Pasqualetti, F., F. Doerfler and F. Bullo (2013). Attack detection and identification in cyber-physical systems. *IEEE Transactions on Automatic Control* **58**(11), 2715–2729.

Perez, T., G.C. Goodwin and M.M. Seron (2003). Performance degradation in feedback control due to constraints. *IEEE Transactions on Automatic Control* **48**(8), 1381–1385.

Porter, M., P. Hespanhol, A. Aswani, M. Johnson-Roberson and R. Vasudevan (2021). Detecting generalized replay attacks via time-varying dynamic watermarking. *IEEE Transactions on Automatic Control* **66**(8), 3502–3517.

Rahimian, H. and S. Mehrotra (2019). Distributionally robust optimization: A review. *arXiv:1908.05659*.

Schellenberger, C. and P. Zhang (2017). Detection of covert attacks on cyber-physical systems by extending the system dynamics with an auxiliary system. In: *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. pp. 1374–1379.

Schulze, D. M., A. B. Alexandru, D. E. Quevedo and G. J. Pappas (2021). Encrypted control for networked systems: An illustrative introduction and current challenges. *IEEE Control Systems Magazine* **41**(3), 58–78.

Shang, Chao, Steven X. Ding and Hao Ye (2021). Distributionally robust fault detection design and assessment for dynamical systems. *Automatica* **125**, 109434.

Si, X., Z. Ren, X. Hu, C. Hu and Q. Shi (2020). A novel degradation modeling and prognostic framework for closed-loop systems with degrading actuator. *IEEE Transactions on Industrial Electronics* **67**(11), 9635–9647.

Tan, S., J. M. Guerrero, P. Xie, R. Han and J. C. Vasquez (2020). Brief survey on attack detection methods for cyber-physical systems. *IEEE Systems Journal* **14**, 5329–5339.

Vinnicombe, G. (2000). *Uncertainty and Feedback: H_∞ Loop-Shaping and the ν Gap Metric*. World Scientific.

Wan, Y., Y. Ma and M. Zhong (2021). Distributionally robust trade-off design of parity relation based fault detection systems. *International Journal of Robust and Nonlinear Control* **31**(18), 9149–9174.

Weerakkody, S. and B. Sinopoli (2015). Detecting integrity attacks on control systems using a moving target approach. In: *2015 54th IEEE Conference on Decision and Control (CDC)*. pp. 5820–5826.

Wen, C.L., F.Y. Lv, Z.J. Bao and M.Q. Liu (2016). A review of data driven-based incipient fault diagnosis. *ACTA Automatica SINICA* **42**, 1285 – 1299.

Xu, Y., S. X. Ding, S. Yin, H. Luo and Z. Zhao (2021). Performance degradation monitoring and recovery of vision-based control systems. *IEEE Transactions on Control Systems Technology* **29**(6), 2712–2719.

Xue, T., M. Zhong, L. Li and S. X. Ding (2020). An optimal data-driven approach to distribution independent fault detection. *IEEE Transactions on Industrial Informatics* **16**(11), 6826–6836.

Yan, W., L. K. Mestha and M. Abbaszadeh (2019). Attack detection for securing cyber physical systems. *IEEE Internet of Things Journal* **6**(5), 8471–8481.

Yang, I. (2018). A dynamic game approach to distributionally robust safety specifications for stochastic systems. *Automatica* **94**, 94–101.

Yin, S., B. Xiao, S. X. Ding and D. Zhou (2016). A review on recent development of spacecraft attitude fault-tolerant control system. *IEEE Trans. on Industrial Electronics* **63**, 3311–3320.

Zhang, D., Q.-G. Wang, G. Feng, Y. Shi and A. V. Vasilakos (2021). A survey on attack detection, estimation and control of industrial cyberâĂŞphysical systems. *ISA Transactions* **116**, 1–16.

Zhang, Y. and J. Jiang (2003). Fault tolerant control system design with explicit consideration of performance degradation. *IEEE Transactions on Aerospace and Electronic Systems* **39**(3), 838–848.

Zhang, Y., J. Jiang and D. Theilliol (2008). Incorporating performance degradation in fault tolerant control system design with multiple actuator failures. *Journal of Control, Automation, and Systems* **6**, 327–338.

Zhou, C., B. Hu, Y. Shi, Y.-C. Tian, X. Li and Y. Zhao (2021). A unified architectural approach for cyberattack-resilient industrial control systems. *Proceedings of the IEEE* **109**, 517–541.

Zhou, D.H., Y. Zhao, Z. Wang, X. He and M. Gao (2020). Review on diagnosis techniques for intermittent faults in dynamic systems. *IEEE Trans. on Indus. Electronics* **67**, 2337 – 2347.

Zhou, K. (1998). *Essential of Robust Control*. Prentice-Hall. Englewood Cliffs, NJ.