ISA Transactions 59 (2015) 303–313

Contents lists available at ScienceDirect

ISA Transactions

journal homepage: www.elsevier.com/locate/isatrans

Dependable control systems with Internet of Things ☆

Tri Tran [a,*], Q.P. Ha [b]

[a] Cambridge CARES, Nanyang Technological University, 62 Nanyang Ave., 639798, Singapore
[b] Faculty of Engineering and Information Technology, University of Technology, 15 Broadway, Ultimo, NSW 2007, Sydney, Australia

ARTICLE INFO

Article history:
Received 28 January 2015
Received in revised form
5 August 2015
Accepted 13 August 2015
Available online 30 August 2015

Keywords:
Dependable control system
Internet of Things
Incrementally dissipative system
Self-recovery constraint

ABSTRACT

This paper presents an Internet of Things (IoT)-enabled dependable control system (DepCS) for continuous processes. In a DepCS, an actuator and a transmitter form a regulatory control loop. Each processor inside such actuator and transmitter is designed as a computational platform implementing the feedback control algorithm. The connections between actuators and transmitters via IoT create a reliable backbone for a DepCS. The centralized input–output marshaling system is not required in DepCSs. A state feedback control synthesis method for DepCS applying the self-recovery constraint is presented in the second part of the paper.

1. Introduction

Manufacturing and processing plants are predominantly operated by specialized computers nowadays. These specialized computers and their variants make up the market of industrial computerized-control systems. Distributed Control Systems (DCSs) together with Programmable Logic Controllers (PLCs) are mainly accounted for the market. The control algorithms in discrete logic with Boolean variables are usually installed in a PLC system, while those for discrete-time regulatory control loops with continuous variables are often installed in a DCS. Both DCS and PLC systems are emerging as multiple purpose platforms lately for both discrete and continuous variables thanks to the improved performance of modern computers. This paper targets the industrial computerized-control systems for regulatory control loops in the processing plants that may have hundreds or even thousands of such loops. For conciseness, the term *feedback control system*, or simply *control system*, is used to address a regulatory control loop in this paper, excluding cases of industrial terms such as DCS, or *computerized-control system*. A DCS is usually a distributed computer system that is implemented with different control algorithms, hence, not the control system as we use here.

1.1. Industrial computerized-control system

DCS, which can also be named as Process Control System (PCS), is a high-integrity and fault-tolerant distributed computer system with fast real-time performances and standardized peripheral interfaces. The dependability specification for a DCS is quantitative and usually much higher than that of an office computer system. The dependability is achieved by redundant components with duty-standby architecture and online switching-over capability. This is a universal approach for satisfying the dependability requirement of a fault-tolerant computer system [1].

Dependability here implies the reliability and availability of a system in operation. The quantitative reliability of a system is measured by its probability of being available and functioning without errors. The quantitative reliability of a system is specified by its *Integrity Level* (IL). The integrity level of 99.9% indicates that the system could possibly (and probably) be malfunctioning or unavailable due to failures in 8.76 h per year $(0.001 \times 365 \times 24 = 8.76)$ while operating. The quantitative reliability is crucial for many industries, and IL is an important metric in designing dependable systems. The integrity level of a component is usually calculated from the mean time between failure (MTBF) and the mean time to repair (MTTR) data using formulas from industry standards such as IEC 61508 [2]. It is noting that IL is fundamentally different to the Safety Integrity Level (SIL) defined for functional safety systems.

From the engineering perspective, a typical DCS layout can be drawn as in Fig. 1 with three tiers of networking, starting from field transmitters and devices, to operator work stations in the central control room for plant operation and management, and up

☆This paper was recommended for publication by Dr. A.B. Rad.

* Corresponding author.
E-mail addresses: tctran@ieee.org (T. Tran), quang.ha@uts.edu.au (Q.P. Ha).

http://dx.doi.org/10.1016/j.isatra.2015.08.008
0019-0578/

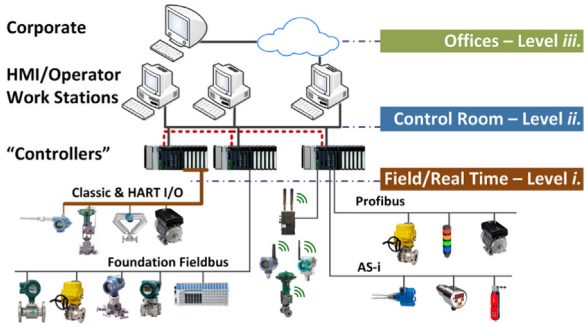

Fig. 1. A typical three level DCS structure.

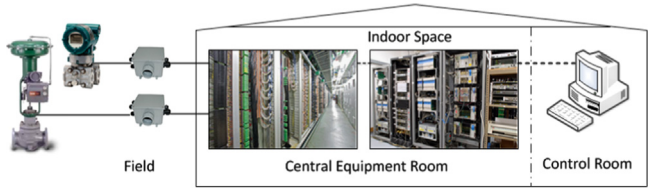

Fig. 2. Loop connections in a DCS with junction boxes, marshaling and system cabinets, as well as central control room.

to business planning and corporate layer. This type of DCS layout can be found in well established standards, such as those from the American Petroleum Institute (API) or the International Society of Automation (ISA). A DCS is designed to accommodate several feedback controllers in its fault-tolerant distributed computer system. The DCS processors, labeled as "Controllers" in Fig. 1, and their input/output (I/O) cards or modules are usually installed in a centralized equipment room next to the control room where the operators interface with, and run the plant, via computer screens.

The graphic system and their database which link with these "Controllers", are usually called Human Machine Interface (HMI) system. The HMI system is also often installed inside the central control room. A block diagram showing field junction boxes, marshaling cabinets, I/O modules and processors are provided in Fig. 2. The I/O and marshalling subsystems as well as equipment room are not required when the proposed collapsing architecture are deployed, as explained in the next section.

1.2. Internet of Things as another step in the advances of process automation

An envisage path for the application of Internet of Things (IoT) in the process automation from our own perspective is illustrated in Fig. 3. The circles with the character "c" next to the smart sensor, transmitter and actuator in this figure represent the additional computational capability of the IoT enabled smart devices, on top of their existing "smart" functionalities currently available.

With this vision, the needs for a centralized processing capability of the traditional DCSs, as displayed in Fig. 1, will vanish in the new system. The loop connections represented in Fig. 2 are now simplified without using the I/O and marshalling subsystems as shown in Fig. 4. The term *IoT in industry* is used to distinguish it with the IoT for public accesses in the corporate environments. It is widely perceived as Industrial Internet of Things (IIoT). The IIoT enabled connectivity also displaces the 'fieldbus' subsystems, as illustrated in Fig. 3. From the device networking perspective, IoT can be considered as another step of fieldbuses. The 'fieldbus' has been implemented with different proprietary protocols developed by different Vendors. Therefore, it has been evaluated as fragmented, and is limited in data capacity and communication speed.

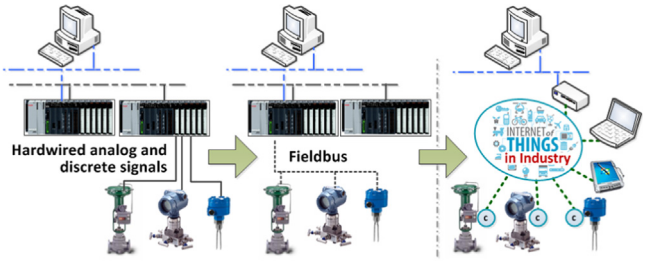

Fig. 3. Industrial IoT as another step in the advances of process automation.

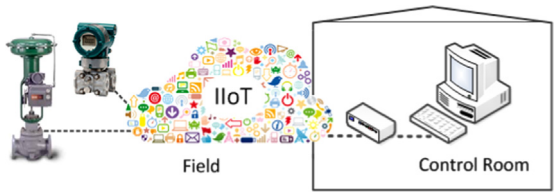

Fig. 4. Loop connections in a DSC system enabled by IIoT.

The hardware infrastructure cost for such proprietary fieldbus is also relatively high.

From the information flow and control room perspective, one can also view it as a 'DCS telemetry architecture', since the presenting work is centered on the first layer for feedback control in the field. The role of the control room from the HMI and operational point of views remains unchanged except for some remote accesses to the field instrumentations and devices for configuring the regulatory control loops which are, now, in the field. The communication protocols of the IoT infrastructure will manage the data flows between the control room and the fields. As we focus on the feedback control synthesis method, the engineering details of IoT systems are not under the scope of this paper.

A recent survey of IoT in industry can be found in [3]. IoT is among the chosen technologies in the on-going 'Smart City' projects around the globe. An organization called IIoT Consortium has been founded to promote and test rival technologies for the industrial applications.

1.3. Industrial standards

The industrial standards for the application software development and configuration of a computerized-control system are well applicable to the presenting system since the overall hardware system architecture is simplified with the IoT infrastructure. And the hierarchical and heterogenous software architectures for different time scales and criticality levels of the applications are still in need. The relevant standards and guidelines include, but are not limited to, ISA 88 for batch processes [4], ISA 95 and IEC 62264 for developing the interfaces between enterprise and control systems and their integration - the manufacturing execution systems [5], IEC 61499 for distributed software architecture [6], and the guidelines for security implementations in the industrial systems such as NIST 800-82 [7].

1.4. Research in IoT security

The IIoT in process automation should be less vulnerable to unexpected accesses and malicious attacks, as well as be able to guarantee the quality of services and other communication performances for real-time applications. The measures for IoT security have been developing in the computer science and information technology (IT) field; see, e.g. [8–10] and references therein. Some recent surveys on the protocols, applications and market of IoT can

be found in [11–15]. The IoT security is still in its infancy and is a current research topic in the computer science field, as well as in the industrial research labs. The landscape of this field is expected to grow tremendously in the next five to ten years. We will, therefore, leave it outside the scope of this paper. An important message from this field is that the users should employ a holistic approach to IoT security, but not only focus on hardware and/or software solutions, while the designers should provide products and solutions that create an underlying end-to-end trust system for the IoT enabled platforms.

1.5. Motivation and contribution

This work has been motivated by the current development of IoT and IIoT worldwide in general, and their important role in the future industrial automation systems in particular. With the hope of making accountable contributions to the progress of this new field, we have proposed effective approaches to the hardware structure and control design method. We have also been motivated by the potential to reduce the constructing, installing and running costs, that the DepCSs can bring to future projects in the industry.

The contributions of this paper are four-fold and can be summarized as follows: firstly, a new architecture for the DCS which collapses the unnecessary system hierarchies from the legacy designs is presented. Not only the operational functionalities are emphasized, but the reliability and availability of the new architecture are also addressed. Secondly, a new structure for a reliable control system is introduced with four anonymous (or autonomous) controllers working in harmony communicating via the IoT. The new structure of the DepCS eliminates the inefficient summation of the outputs in the classical approach, which had been designed for analogue and hard-wired signals processed by a limited capacity of the computational platform in the past. Thirdly, with the fully autonomous structure of the DepCS, the plug-and-play design for the future control systems becomes feasible. By the same token, the costly input/output and marshalling systems are not required anymore, which will make the future system much more cost effective to manufacture, install, implement and operate. Finally, a novel control synthesis method employing the self-recovery constraint for the incrementally dissipative systems has been developed and presented with exhaustive numerical simulations. This approach is effective in real time, as it is an event-based triggered approach that only re-computes the feedback gain during the duty-standby transitions. And only static gains are used in the remaining time intervals.

This paper is organized as follows. The structure of a DepCS and the new DSC system architecture is present in Section 2. In Section 3, we present a state feedback control synthesis method for DepCS applying dissipativity constraint. Section 4 is reserved for the stability condition, and the computation procedure for the state feedback gain. Numerical simulation with small-signal models of the hybrid wind-diesel power system and of the automatic generation control of a power system is provided in Section 5. Section 6 concludes this paper.

2. Dependable control systems

The current approach for ensuring the continuous operation of feedback control systems implemented on duty-standby computer systems is to employ the technology of reliable control systems from the control literature; see, e.g. [16] and references therein. A block diagram of such reliable control systems is depicted in Fig. 5. This technology may become difficult for applying to the IIoT enabled architecture to accommodate the clustered topology,

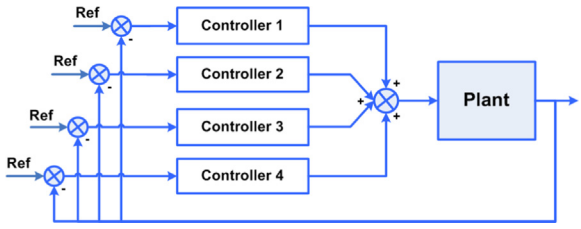

Fig. 5. A classical reliable control system (RCS).

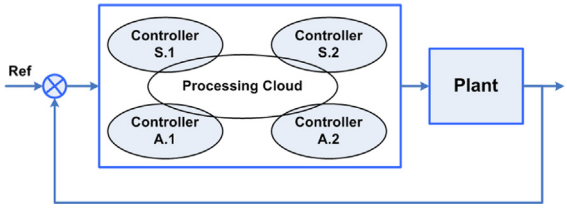

Fig. 6. A Dependable Control System (DepCS).

peer-to-peer communication and cloud-based applications, which have been evaluated as adequate for use with IoT.

The architecture of a dependable control system is presented in Fig. 6. In a processing plant, there may be several regulatory control loops, thus several DepCSs. They form a Dependable Self-recovery Control (DSC) system, as has been designated in [17,18]. Four processors are assumed to involve in this DepCS. Reasonings for having four processors resource to the achievable integrity level and available products. According to industrial data in the computerized-control system field, it is usually expected that the regulatory control loops should achieve IL-2 of the range 99.99–99.999% as a minimum in their design specifications. IL-2 will eventually lead to a dual-redundant architecture, as a minimum, if COTS (commercial-off-the-shelf) components are used in the design [1,2]. The architecture of four duty-standby controllers communicated via the sensor and actuator network is the skeleton of this work to achieve IL 2, or higher, with wired-line or wireless COTS components. By targeting general purpose components in the architecture, the result will not be limited within a proprietary application but outreaches all standardized products currently available.

2.1. Operational description

Each processor in a DepCS will run the control algorithm independently. They are denoted as controllers S.1/2 and A.1/2 as labelled in Fig. 6. Only one of these controllers is active, as a duty controller, at any one time. Therefore, the location of the duty processor varies from time to time. The active program that manipulates the control variables will relocate among these four controllers. We thus call it a processing “cloud” or “fog”, as it is location independent and has a small number of platforms (which has a different meaning to the enterprise cloud). The processors can be integrated into the currently used smart transmitters/sensors or actuators. In general, the number of installed processors will depend on the requirements of dependability, operability and cost effectiveness from a particular application. While the status information is consecutively exchanged between the processors, among those, one acts as a duty processor, the others are in standby, the controller inside each processor will operate in an anonymous manner. On top of the status information, which is managed by the underlying operating system, only one scalar variable is required to be exchanged between the redundantly backed-up controllers. A standby controller can be activated into the duty role relying on this received scalar variable from the duty

controller and the trigger signal from the operating system, as represented by the active links between the controllers in Fig. 7.

2.2. Selection of controller in duty mode

The selection of a controller in the duty mode, which is scheduled by the processor operating system, will be counted upon its hardware healthy state, the communication between processors, diagnostic information, or user decision. With the clustered oriented structure shown in Fig. 6 and small number of participants in the "cloud", it can also be called a "fog" based system.

2.3. Back-up management for fault-tolerant operation

The key for a successful implementation of dependable systems rests with the amount of data to be transferred between the duty and standby components [1]. Fewer amounts of exchanged data will demand less inter-communication over the processor "cloud" or "fog". The presented method requires only one real number to be exchanged between the peer controllers, while the reliable control system of Fig. 5 needs four real numbers. And for multi-variable systems, it is even worse as it requires the value of four vectors to be summed up for the main control vector. More importantly, it is not necessary to exchange this variable at every updating time steps in a DepCS. By virtue of this one-variable approach, which also accepts intermittent data losses, the success of the employed method is assured for both wired-line and wireless networking systems. The IIoT based DSC system is not only energy efficient, but also achieves a higher dependability owing to the simplicity of exchanging only one variable in both single- and several-variable constrained systems. With these advantages, the new DSC system will be able to have the controllers implemented in a fully decentralized architecture. From this reasoning, we have come up with a new system structure shown in Figs. 9 and 10, to be discussed in the following subsections.

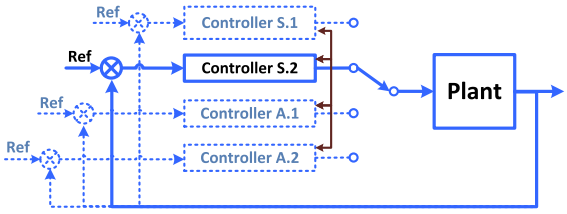

Fig. 7. Active and inactive connections in DepCS.

2.4. IIoT enabled dependable self-recovery control system

2.4.1. Existing distributed control system

A typical DCS architecture in industry extracted from [19] is shown in Fig. 8, wherein the real-time control layer consists of several regulatory control loops. The dependability of the lowest level in this DCS architecture consisting of sensors, actuators and controllers, is paramount by virtue of real-time performances. The currently used DCS is usually a fault-tolerant computer network and system. The main processors are implemented with several controllers connecting with sensors and actuators, and physically installed in a central control room, or in a few satellite control rooms interconnected with proprietary networks.

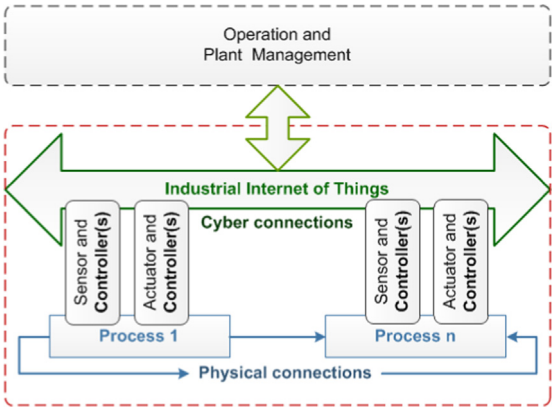

Fig. 9. A DepCS and IIoT based architecture. The controllers are integrated into smart sensors and actuators.

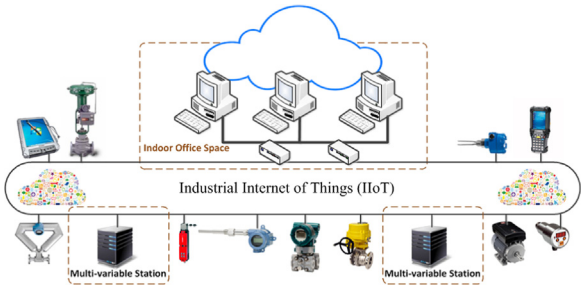

Fig. 10. An IIoT enabling distributed computerized-control system.

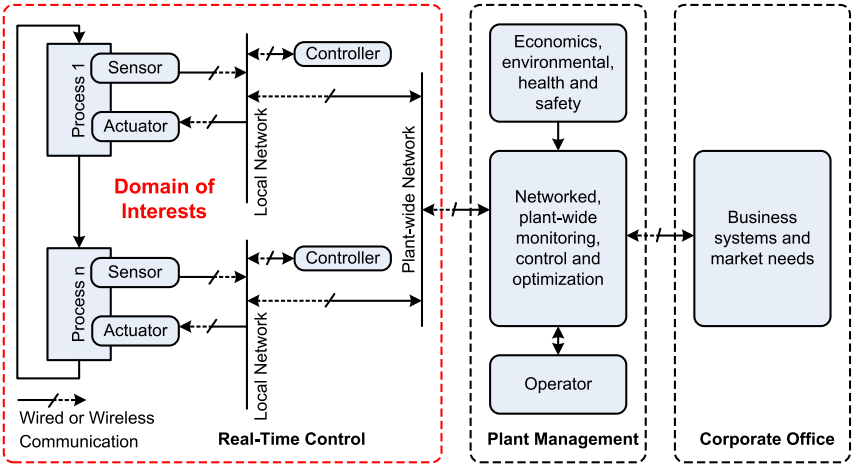

Fig. 8. A typical DCS architecture, extracted from [19].

2.4.2. Dependable self-recovery control system

The centralized control architecture will vanish when a Dependable Self-Recovery Control (DSC) system is installed. The new DSC system will support the Plug-and-Play (PnP) functionalities, which will lower the installation, testing and operational costs. It will facilitate in the controller installation in the field and on site, far away from the central control room, and in a fully decentralized architecture of IIoT based networks.

With a DSC system, which uses IIoT enhanced actuators and transmitters and a novel DepCS structure, the centralized processing and input–output (I/O) marshaling subsystems are not required. The marshaling and I/O system cabinets will thus be eliminated in this new IIoT enabled DSC system.

The architecture of a fault-tolerant DSC system is depicted in Figs. 9 and 10. The new architecture is much simpler than that of the traditional DCS, whereas the "Controllers" shown in Fig. 1 are not installed. In Fig. 9, the controllers are integrated into the smart sensors and actuators in a DSC system. And these smart devices are interconnected via the IIoT.

The relevant control literature for this type of IIoT based DSC system is in the networked control system (NCS) strand. Research in NCSs has been intensive during the past decade and is quite mature in its own right, see, e.g. [20]. However, the currently developed control methods for NCSs have not addressed thoroughly the dependability mandate for high-integrity applications. The study in [17,18] and this paper proposed and partially gave a solution to this problem. A control algorithm for DepCSs of the new DSC system is presented in the next section.

3. A state feedback synthesis scheme for dependable control systems

A novel constrained-state feedback control design method for duty-standby controllers of a dependable control system is presented in this section. As an alternative to the control summation in reliable control systems, only one controller is active at any one time in a dependable control system. The automated managing of duty-standby controllers is challenging, especially in wireless transmitter and actuator networks, owing to the scarcity of both information and processing resources. The solution in this section is effective and feasible, as taking into account both state- and control-incremental constraints, and simply involving a static state-feedback with pre-computed and re-computed strategy. The state feedback gains are synthesized to fulfill the strict requirement on the two incremental constraints, and simultaneously maintain the control performance such as settling time. As a result of that, the duty-standby controllers will be able to operate independently, while assuring the closed-loop system stability with a newly introduced self-recovery constraint. For a dependable control system, the employed self-recovery constraint is a quadratic constraint with respect to the control and state increments. By satisfying such incremental constraints, the self-recovery constraint will facilitate the independent operation of the controllers while having to exchange only one scalar variable between the redundant controllers. It is an effective method from the information and communication perspective. The self-recovery constraint packs two information, the control and state increments, into one, before transmitting to its peers. The self-recovery constraint at the receiver side will then unpack the information and use the result for the local control algorithm.

The incremental passivity approaches have been theoretically presented elsewhere [21,22], which imply the usages of signal increments Δu_k and Δx_k in the supply rate $\xi(\Delta u_k, \Delta x_k)$. The equilibrium-independent passivity has also been introduced in [23]. Recent developments for the theoretical foundation of differential dissipativity and incremental stability can be found in [24,25]. Nevertheless, practical control applications have not been developed from these theoretical studies. The incremental dissipativity is employed in this work for the control design problem, which also incorporates the control- and state-incremental constraints.

3.1. Notation

Capital and lower case alphabet letters denote matrices and column vectors, respectively. $\mathrm{diag}[A_i]_1^h$ stands for the block-diagonal matrix with diagonal entries A_i, $i = 1, 2, \ldots, h$. $\| u_i \|$ is the ℓ_2-norm (Euclidean) of vector u_i. In symmetric block matrices, we use $*$ as an ellipsis for terms that are induced by symmetry.

3.2. System model and quadratic constraint

Consider a system $\mathcal{S}$ having a discrete-time state-space model of the form:

$$\mathcal{S}: x(k+1) = Ax(k) + Bu(k), \tag{1}$$

where $x(k) \in \mathbb{R}^n$ and $u(k) \in \mathbb{R}^m$ are the state and control vector, respectively. (A, B) is controllable. The following control and state constraints are considered herein:

$$\mathbb{U} := \{u: \| u \|^2 \le \eta,\ \eta > 0\}, \tag{2}$$

$$\mathbb{X} := \{x: \| x \|^2 \le \rho,\ \rho > 0\}. \tag{3}$$

The state increment $\Delta x(k) := x(k+1) - x(k)$, and its constraint is considered for dependable self-recovery control systems. Specifically,

$$\| \Delta x(k) \|^2 \le \Delta\rho, \tag{4}$$

for given $\rho > 0$. Similarly, the control incremental constraint of the form

$$\|\Delta u(k)\|^2 \le \Delta\eta \tag{5}$$

is also inclusive in the problem formulation. Firstly, define a quadratic supply rate for $\mathcal{S}$, as follows:

$$\xi(\Delta u(k), \Delta x(k)) := \begin{bmatrix} \Delta u^T(k) & \Delta x^T(k) \end{bmatrix} \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta x(k) \end{bmatrix} \tag{6}$$

where Q, R, S are multiplier matrices with symmetric Q and R. For conciseness, $\xi(\Delta u(k), \Delta x(k))$ is denoted as $\xi_{\Delta(k)}$. The self-recovery constraint is then defined in the following.

Definition 1. The input and state increment pair $(\Delta u_k, \Delta x_k)$ of $\mathcal{S}$ is said to satisfy the self-recovery constraint if there are $k_0 \in \mathbb{Z}^+$ and $0 \le \gamma < 1$ such that

$$0 \le \xi_{\Delta(k)} \le \gamma \xi_{\Delta(k-1)} \quad \forall\ k \ge k_0. \tag{7}$$

The incremental dissipativity of $\mathcal{S}$, defined next, plays a more important role in the convergence of $\Delta x(k)$ in this development.

Definition 2. $\mathcal{S}$ is said to be (Q, S, R) – incrementally dissipative with respect to the supply rate $\xi_{\Delta k}$, if there exists a non-negative storage function $V(\Delta x) := \Delta x^T P_i \Delta x$, $P_i > 0$, such that for all $\Delta x(k)$ and all $k \in \mathbb{Z}^+$, the following dissipation inequality is satisfied irrespectively of the initial value of the state increment $\Delta x(0)$:

$$V(\Delta x(k+1)) - \tau V(\Delta x(k)) \le \xi_{\Delta(k)}, \quad 0 < \tau < 1. \tag{8}$$

The closed-loop system stability is achieved in this development via the convergence of $\Delta x \to 0$. We then apply the result in [24] to infer the convergence of $x \to x_e$ to its equilibrium.

For systems having control and state constraints, it is necessary to make some assumptions on the invariance of $\mathbb{X}$, in order for the control problem to be feasible.

Definition 3. A set $\Omega \subset \mathbb{R}^n$ is called a constrained control invariant set with respect to $\mathbb{U}$ of the discrete-time system Σ, if for each $x_{(k)} \in \Omega$, $\exists\, u_{(k)} \in \mathbb{U}$, such that $x_{(k+1)} \in \Omega$ for all $k \geq 0$.

Assumption 1. $\mathbb{X} \subset \mathbb{R}^n$ is a constrained control invariant set with respect to $\mathbb{U}$ for the discrete-time system Σ (1).

The problem description is now delineated below.

3.3. Problem description

We are concerned with

- The design problem of the constrained-state feedback control law of the form $u(k) = Kx(k)$ for $\mathcal{S}$, such that the closed-loop system (1) is stable, subject to the satisfactions of all four constraints on u, x, Δu, Δx, (2)–(5).
- Maintaining the asymptotic property of $\xi_{\Delta(k)}$ among the member controllers of a DepCS. The real-time value of $\xi_{\Delta(k)}$ is transferred between the duty and standby controllers, such that (7) is fulfilled by all active controllers. This will be done by the underlying duty-standby role management mechanism.
- Re-computing the state-feedback gain at every transition event, for the newly assigned duty controller, to assure
  - The state convergence of $\Delta x \to 0$,
  - The incremental constraint satisfaction, and
  - The control performance of a DepCS, such as the settling time and closed-loop stability, is achieved in real time.

The re-computation is not a persistent online task, as not occurring at every time step, but only at the duty-standby switching-over incidences. The current state vector $x(k)$ is known to the local controller.

Among the above three tasks, the second one will be managed by the operating system of the computer platform running the control algorithm. It is, therefore, an assumption herein. This paper accomplishes the remaining two tasks of pre-computing and re-computing the feedback gains.

The asymptotic property of $\xi_{\Delta(k)}$ among the member controllers of a DepCS is crucial in this approach, as it ensures that the control performance of a DepCS is maintained throughout the standby-duty switching over incidences.

3.4. Switching-over activity and information

If k_s is the time instant, at which the duty controller is faulty, then $\xi_{\Delta(k_s)}$ is the last known value of $\xi(.)$ to all peer controllers. Here, assume that the switching-over activity will take place in δ time steps, $\delta \geq 1$. During the transition time, the last known value of the control vector $u(k_s)$ will be applied to manipulate the plant. This can be done by having a local buffer at the smart actuator, or simply using a mechanical latch to keep the actuator at the last position. Once the operating system signaled the completion of the transition, at the time instant $k_s + \delta$, the value of $u(k_s)$ will be retrieved to the newly assigned duty controller, by having $u(k_s) = u(k_s + \delta)$.

The development for state-feedback gain pre-computation and re-computation (after every duty standby switching over incidence) is delineated in the next section.

4. Stability condition and constrained-state feedback

The sufficient stability condition is stated in the below theorem as a basis for the gain computations.

Theorem 1. *Let $0 < \tau < 1$, $x_{(0)} \in \mathbb{X}$ and $\xi_{\Delta(0)} > 0$. Consider a system Σ (1). Suppose that Assumption 1 holds, and the closed-loop system Σ and $u = Kx$ is incrementally dissipative with the dissipation inequality (8) and fulfills the asymptotic quadratic constraint (7). Then, it is locally asymptotically stable.*

Proof. By applying the asymptotic property of $\xi_{\Delta(k)}$ in (7) to the dissipation inequality (8), we obtain for every $k \geq 0$,

$$V\big(\Delta x(k+1)\big) \leq \tau V\big(\Delta x(k)\big) + \gamma|\xi_{\Delta(k-1)}|, \quad 0 < \gamma < 1.$$

Thus, by iteration

$$V\big(\Delta x_k\big) \leq \tau^{k-1} V\big(\Delta x_1\big) + \gamma|\xi_{\Delta(0)}|(1 + \tau + \cdots + \tau^{k-2}),$$

$$= \tau^{k-1} V\big(\Delta x_1\big) + \gamma|\xi_{\Delta(0)}|\frac{1-\tau^{k-1}}{1-\tau}.$$

It is to prove herein that for each $\beta > 0$ there is a finite $k(\beta) > 0$ such that

$$V(\Delta x(k)) \leq \beta \quad \forall\, k \geq k(\beta).$$

Indeed, for each $\beta \geq 0$, there exist two time instants $\bar{k}$ and $\tilde{k}$ such that

$$|\xi_{\Delta(k-1)}| \leq \beta\frac{1-\tau}{\gamma} \;\; \forall\, k > \bar{k} \quad \text{and} \quad \tau^{k-\tilde{k}} V\big(\Delta x(\tilde{k})\big) \leq \frac{\beta}{2} \;\; \forall\, k > \tilde{k}$$

due to (7) and $0 < \tau < 1$. Since

$$|\xi_{\iota(k-1)}| \leq \beta\frac{1-\tau}{\gamma} \Rightarrow \gamma|\xi_{\iota(\bar{k}-1)}|\frac{1-\tau^{k-\bar{k}}}{1-\tau} \leq \frac{\beta}{2},$$

there exists $\check{k} = \max(\bar{k}, \tilde{k})$ such that for each $\beta \geq 0$,

$$V\big(\Delta x(k)\big) < \frac{\beta}{2} + \frac{\beta}{2} = \beta \;\; \forall\, k \geq \check{k}.$$

With $P>0$, $\|\Delta x(k)\| \to 0$ as $k \to +\infty$. Applying the result in [24], $x \to x_e$ whenever $\Delta x \to 0$, and $x(k) \in \mathbb{X}$ by Assumption 1, we conclude that the closed-loop system Σ is also locally asymptotically stable.□

There are two feedback gain computations in this approach, the pre-computation in the design phase and the re-computation during the transitions from the standby to duty controller.

4.1. Pre-computing the feedback gain

The technique of linear matrix inequality (LMI) is employed in this work. The control law has the classical state feedback form of $u = Kx$. The following LMIs are derived from (7) and (8), by substituting the model of the $\mathcal{S}$ and $u = Kx$ into the corresponding inequalities, rearranging them and applying the Schur complement [26]:

$$\begin{bmatrix} \check{P}_1 & K^TB^T + A^T \\ * & \sigma\check{P}_1 + Y_1 \end{bmatrix} \geqslant 0, \quad \check{P}_1 > 0, \tag{9}$$

$$\begin{bmatrix} \check{M}_1 & K^TB^T + A^T \\ * & \gamma M_1 \end{bmatrix} \geqslant 0, \quad M_1 > 0, \tag{10}$$

where $P_1 := (K^TB^T + A^T)P(A + BK)$, $M_1 := (K^TB^T + A^T)M(A + BK)$, $M := Q + SK + K^TS^T + K^TRK$, $M_1 = P_1Y_1P_1$, $\check{M}_1 := M_1^{-1}$, $\check{P}_1 := P_1^{-1}$.

Similarly, the constraint (2)–(5) are satisfied when the following LMIs are fulfilled:

$$\begin{bmatrix} I & K^TB^T + A^T \\ * & I \end{bmatrix} \geqslant 0, \quad \begin{bmatrix} I & K^TB^T + A^T - I \\ * & \frac{\Delta\rho}{\rho} I \end{bmatrix} \geqslant 0, \tag{11}$$

$$\begin{bmatrix} I & K^T \\ * & \frac{\eta}{\rho} I \end{bmatrix} \geqslant 0, \quad \begin{bmatrix} I & K^T \\ * & \frac{\Delta\eta}{\Delta\rho} I \end{bmatrix} \geqslant 0. \tag{12}$$

The computation for K is then as follows: firstly, the matrices $\check{P}_1$, Y_1 and K are found from the solution to (9), (11) and (12) by the optimization of

$$\max_{\check{P}_1, Y_1, K} x_0^T \check{P}_1 x_0 \quad \text{s. t. (9), (11), (12).} \tag{13}$$

The objective function of $x^T(0)\check{P}_1 x(0)$ is employed herein to guarantee the performance of $\sum_k^\infty x^T P x$ as usually determined in the control literature. Eq. (10) cannot be included as not an LMI. The matrices P_1 and M_1 are then obtained from $\check{P}_1$, Y_1 and K accordingly.

Secondly, assuming $K^T = P_1^{-1}X^T$ and $M_1 = Z_1P_1$, X and Z_1 are then calculated from the resultant P_1 and Y_1. Subsequently, P_1 is re-computed off-line by solving the equivalent LMIs of (9), (11) and (12). The equivalent LMIs, which have been derived from (9), (11) and (12) using Schur complement, are provided as follows:

$$\begin{bmatrix} \check{P}_1 & \check{P}_1X^TB^T + A^T \\ * & \sigma\check{P}_1 + \check{P}_1Z_1 \end{bmatrix} \geqslant 0, \quad \begin{bmatrix} \check{P}_1Z_1^{-1} & \check{P}_1X^TB^T + A^T \\ * & \gamma\check{P}_1Z_1 \end{bmatrix} \geqslant 0, \tag{14}$$

$$\begin{bmatrix} I & \check{P}_1X^TB^T + A^T \\ * & I \end{bmatrix} \geqslant 0, \tag{15}$$

$$\begin{bmatrix} I & \check{P}_1X^TB^T + A^T - I \\ * & \frac{\Delta\rho}{\rho} I \end{bmatrix} \geqslant 0, \tag{16}$$

$$\begin{bmatrix} I & \check{P}_1X^T \\ * & \frac{\eta}{\rho} I \end{bmatrix} \geqslant 0, \quad \begin{bmatrix} I & \check{P}_1X^T \\ * & \frac{\Delta\eta}{\Delta\rho} I \end{bmatrix} \geqslant 0. \tag{17}$$

And the LMI optimization of

$$\max_{\check{P}_1 > 0} x_0^T \check{P}_1 x_0 \quad \text{s. t. (14), (15), (16), (17),} \tag{18}$$

will be solved for $\check{P}_1$. The feedback gain $K = X\check{P}_1^{-1}$ is obtained accordingly. This feedback gain will be used in the control law until the occurrence of a duty-standby transition.

4.2. Re-computing the transition feedback gain

The control needs to keep track with the evolution of the self-recovery constraint to ensure the closed-loop system stability, and simultaneously maintain the control performance of settling time. When the duty-standby transitions occur, the feedback gain K is necessarily re-computed at the new duty controller based on the received value of $\xi_{\Delta(k_s)}$ from the previous duty controller. We assumed that the transition is accomplished at the time step $k_s + \delta$. The newly computed feedback gain K is then applied from the time step $k_s + \delta$ onward, until the occurrence of the next duty-standby transition.

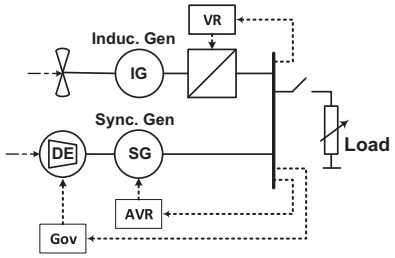

Fig. 11. A typical isolated wind-diesel power system.

Based on the inequality of the self-recovery constraint (7), the current state $x(k)$, the last known value of $\xi_{\Delta(k_s)}$ and the retrieved value of $u(k-1) = u_{(k_s)}$, the following LMI is derived for re-computing the feedback gain K:

$$x_{(k)}^T(A + BK - I)^TM(A + BK - I)x_{(k)} - \gamma\xi_{\Delta(k_s)} \leqslant 0, \tag{19}$$

which is equivalent to the LMI below

$$\begin{bmatrix} M_1^{-1} & x_{(k)} \\ * & \gamma\xi_{\Delta(k_s)} \end{bmatrix} \geqslant 0, \quad M_1 > 0. \tag{20}$$

The above inequality ensures that the constraint on the state increment will be satisfied between two time steps k_s and k.

Similarly, the control incremental constraint (5) also needs to be satisfied, by the fulfillment of the following LMI:

$$\begin{bmatrix} I & X\check{P}_1x_{(k)} - u_{(k_s)} \\ * & \Delta\eta \end{bmatrix} \geqslant 0. \tag{21}$$

Using the known values of $x(k)$, $\xi_{\Delta(k_s)}$, $u_{(k-1)} = u(k_s)$, and the resultant X and Z_1 from the pre-computation design phase, while assuming that $K^T = P_1^{-1}X^T$ and $M_1 = Z_1P_1$, we re-compute K online using the LMI optimization in the following:

$$\max_{\check{P}_1} \quad x_{(k)}^T\check{P}_1x_{(k)}$$
$$\text{s. t.} \quad \begin{bmatrix} \check{P}_1Z_1^{-1} & x_{(k)} \\ * & \gamma\xi_{\Delta(k_s)} \end{bmatrix} \geqslant 0,$$
$$(14),\ (15),\ (16),\ (17),\ \text{and}\ (21). \tag{22}$$

The feedback gain $K = X\check{P}_1^{-1}$ is obtained accordingly. This gain K is applied to the control law until the next occurrence of a duty-standby transition. The feedback gain computations are summarized as follows:

Procedure 1. State feedback gain computation for DepCS.

1. Pre-computation:
   (a) Solve the optimization (13) for $\check{P}_1$, Y_1 and K.
   (b) Obtain P_1, M_1, Z_1, X.
   (c) Solve the optimization (13) for $\check{P}_1$.
   (d) Obtain $K = X\check{P}_1$.
2. Re-computation: Assume that $\xi_{\Delta(k)}$ is transferred between the peer controllers at every step k. At a duty-standby transition step $k_s + \delta$ triggered by the operating system, the newly assigned duty controller will:-
   (a) Solves the optimization (22) for $\check{P}_1$, using X, Z_1 from the pre-computation, and the known $\xi_{\Delta(k_s)}$, $x_{(k)}$ and $u_{(k-1)} = u_{(k_s)}$.
   (b) Obtain $K = X\check{P}_1$.

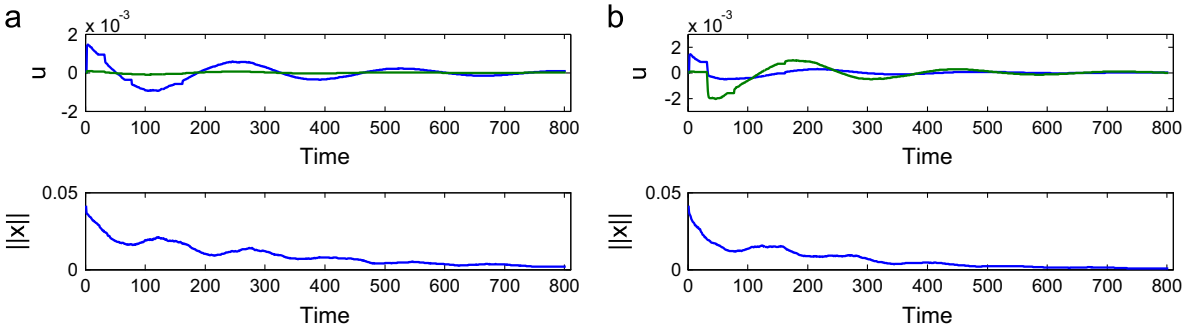

Fig. 12. DSC system with self-recovery constraint. (a) Without gain re-computations. (b) With re-computed gains during transitions.

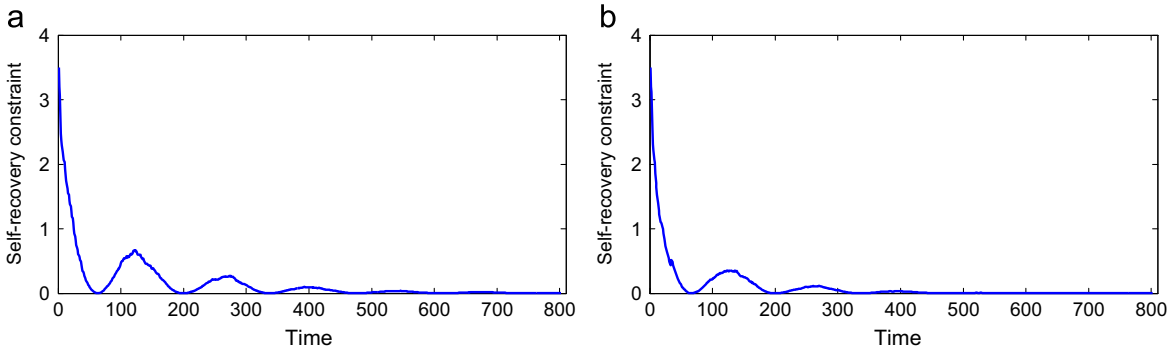

Fig. 13. Self-recovery constraint evolution. (a) Without gain re-computations. (b) Less and lower peaks with re-computed gains.

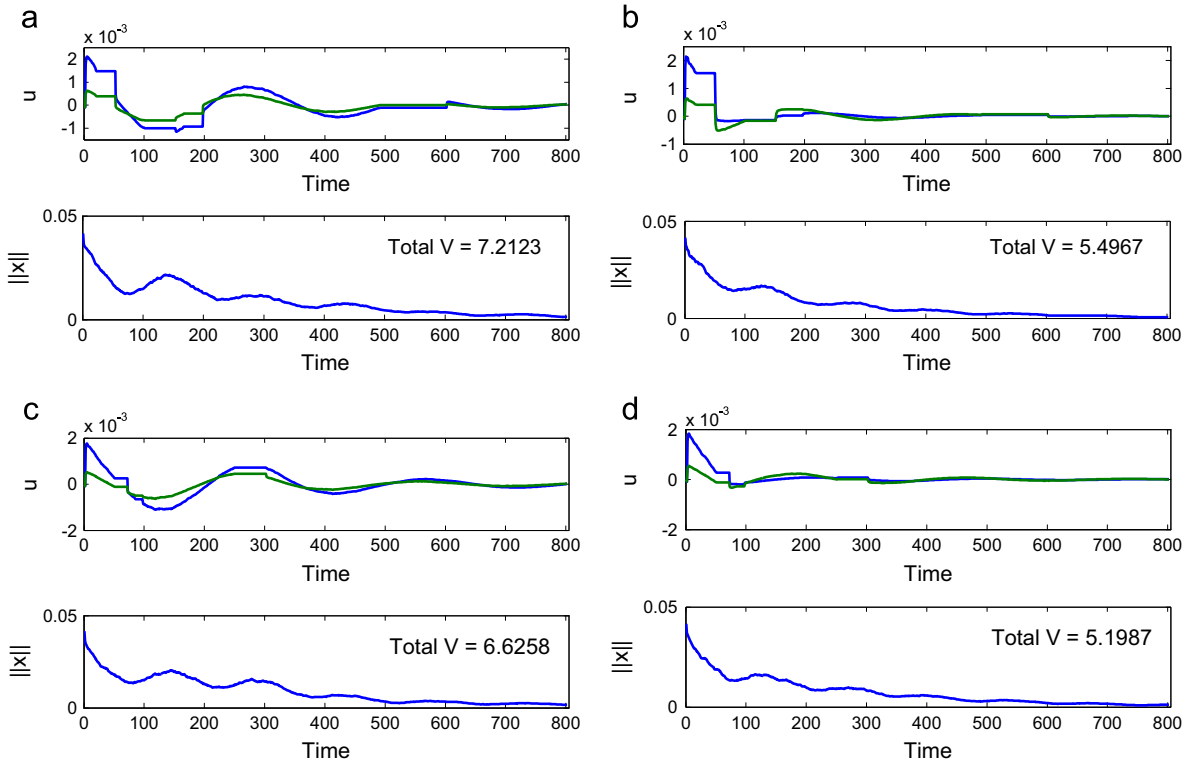

Fig. 14. DSC system – different transition time instants and intervals. (a) Without gain re-computations – Case 2. (b) With gain re-computations – Case 2 with 23.8% improvement. (c) Without gain re-computations – Case 3. (d) With gain re-computations – Case 3 with 21.5% improvement.

The development is generally applicable to single- and several-variable state space systems. An introduction to the deterministic intermittent data losses is given in the next subsection.

4.3. Intermittent data loss process

When an Internet protocol (IP) based connection is used for measurement data and for transmitting control signals to actuators, the data may be intermittently lost due to data package dropouts or some other reasons. The data loss process is modeled here. The state vector x of $\mathcal{S}$ becomes $\check{x}$ beyond the network interface ports. When all the local state vectors $x_i(k)$ at time instant k are transmitted successfully, we have $\check{x}(k) = x(k)$, otherwise $\check{x}(k) = x(k-1)$. This is an actual scenario, when the communication stack and buffer of the IP protocol are not modified.

It is assumed here that the sampling and updating time instants (thus data lost instances) are synchronised to all variables by hardware and firmware configurations. By representing the consecutive updating instants of $\check{x}(k)$ with a sequence of integer numbers $\mathcal{I} := \{j_1, \ldots, j_q, \ldots, j_p, \ldots\}$, $\mathcal{I} \subset \mathbb{Z}^+$, the time interval between j_q and j_{q+1} is treated as one transmission period. If the communication data are perfect at time k, we have $j_p = k$. The upper bound of the successful transmission periods is denoted as $\boldsymbol{\mu}$ (or MATI – maximum allowable transmission interval).

$$\boldsymbol{\mu} := \max_{j_q \in \mathcal{I}} \big(\tau(q)\big), \quad \tau(q) := j_q - j_{q-1}. \tag{23}$$

The self-recovery constraint (7) now becomes

$$0 \le \check{\xi}_{\Delta(\kappa)} \le \gamma \check{\xi}_{\Delta(\kappa-1)} \quad \forall\, k \ge k_0. \tag{24}$$

where $\check{x} = x$ if $\kappa = j_q$, otherwise $\check{x} = \hat{x}$, and $\check{\xi}_{\Delta} = \xi_{\Delta}|_{x=\check{x}}$. This means, $\check{\xi}_{\Delta(k)} \le \gamma \check{\xi}_{\Delta(k-\tau)}$ in the worst case scenarios.

System $\mathcal{S}$ is then said to be data-lost robust (Q, S, R) – incrementally dissipative with respect to the supply rate $\xi_{\Delta k}$, if there exists a non-negative storage function $V(\Delta x) := \Delta x^T P_i \Delta x$, $P_i > 0$, such that for all $\Delta x(k)$ and all $k \in \mathbb{Z}^+$, the following dissipation inequality is satisfied irrespectively of the initial value of the state increment $\Delta x(0)$:

$$V(\Delta x(k+\tau)) - \tau V(\Delta x(k)) \le \xi_{\Delta(k)}, \quad 0 < \tau < 1.$$
$$\forall\, \tau = i, \ldots, \boldsymbol{\mu}. \tag{25}$$

The LMIs (9) and (10) will have similar forms, but employ the predictive model of $\mathcal{A} = A^\tau$ and $\mathcal{B} := [A^{\tau-1}B\ A^{\tau-2}B\ \ldots\ AB\ B]$, instead of A and B. The remaining development remains unchanged, thus is not re-produced herein. We leave the detailed development and numerical simulation for the intermittent package drops to future work. Some recent results in networked control systems can be found in, e.g. [27–29].

5. Numerical examples

5.1. Isolated wind-diesel power system

The small-signal linearized model of an isolated wind-diesel power system with local PI controllers, taken from [30], has been been used in this numerical example. This wind-diesel power system consists of a wind generator and a diesel generator connecting to a common bus bar. The wind generator has a wind turbine, an induction generator and the converter/inverter with its own voltage regulator. The diesel generator has a diesel engine with governor and a synchronous generator with AVR (automatic voltage regulator), as sketched out by Fig. 11. The two state space realization matrices A and B of $\mathcal{S}$ (1) are provided below. The model parameters are borrowed from the work in [30]

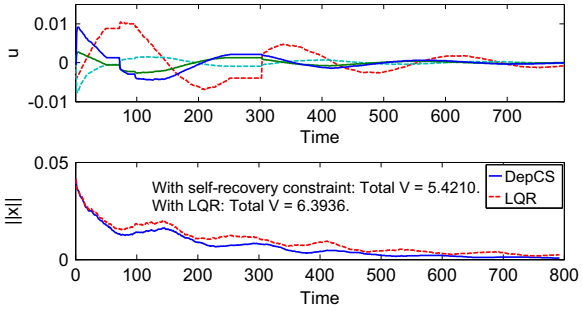

Fig. 15. Self-recovery constraint versus LQR. The performance with LQR depends on the choices of weighting matrices, and is not as good as the one with re-computed gains using the self-recovery constraint. The control moves are larger with LQR.

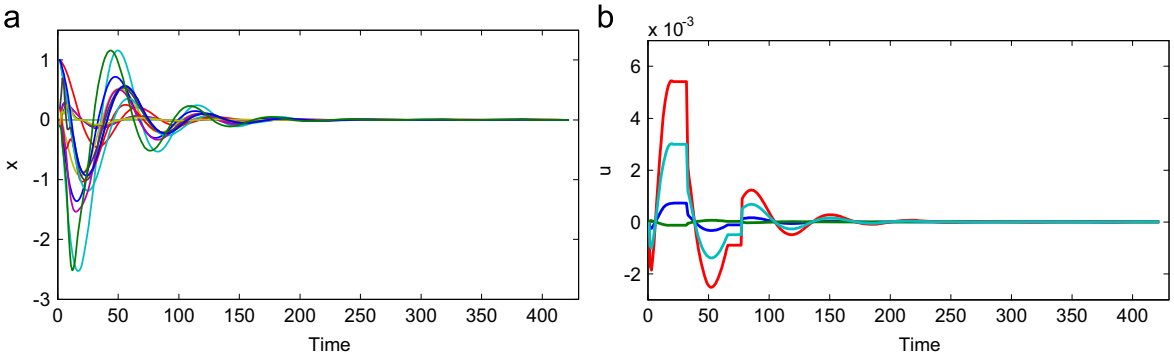

Fig. 16. DSC system in AGC of a power system, with three duty-standby transitions. (a) State trajectories. (b) Control trajectories.

$$A = \begin{bmatrix} -7.4 & 5 & 0 & 0 & 7.47 & 0 & 0 & 0 \\ 0 & -0.333 & 0.333 & 0.333 & 0 & 0 & 0 & 0 \\ -0.02 & 0 & -0.5 & 0 & 0 & 0 & 0 & 0 \\ -1.58 & 0 & 0 & -40.0 & 0 & 0 & 0 & 0 \\ 0.374 & 0 & 0 & 0 & -0.623 & 0.25 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0.14 & 0.084 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0.4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -24.39 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{K_D(T_{D2}-T_{D1})}{T_{D2}(T_{D2}-T_{D3})} & 0 \\ \frac{K_D(T_{D3}-T_{D1})}{T_{D3}(T_{D3}-T_{D2})} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & \frac{K_{P2}}{T_{P2}} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.101 & 0 \\ 7.898 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 24.390 \end{bmatrix} L = \begin{bmatrix} 5 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0.25 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

The supplement load frequency control here is to stabilize the system frequency and diesel generator power in the events of small load changes or wind power variations. The two states of interest are the deviations of system frequency and diesel demand power, which are the first and second elements in the state vector, respectively. The updating time is chosen at $\tau_s = 0.1$. The initial state vector is chosen as $x(0) = 10^{-3} \times [24.5\ 15.7\ -16.3\ -8.7\ 11\ -10\ 11\text{–}14]^T$. The constraints are set with $\eta = 2 \times 10^{-4}$, $\rho = 5 \times 10^{-3}$, $\Delta\eta = 2 \times 10^{-5}$, $\Delta\rho = 5 \times 10^{-4}$. The coefficients $\gamma = 0.999$ and $\tau = 0.9999$ have been selected in this simulation study. The pre-computed feedback gain is as follows:

$$K = 10^{-2} \times \begin{bmatrix} 3.20 & 3.42 & 0.55 & 18.80 & 2.58 & 0.65 & 2.31 & 0.48 \\ 0.22 & 0.38 & 0.37 & 0.22 & 0.43 & 0.41 & 0.41 & 0.61 \end{bmatrix}.$$

In the first simulation, the re-computed gains are not applied. Only the pre-computed gain is used in the control laws. The control and state trends are depicted in Fig. 12a with three transition events occurring at the time steps 20, 65 and 150. During the transition time, the control maintains its previous known value. The transitions are assumed taken place in 10 time steps, i.e. $\delta = 10$. The state and control trends using the above feedback gain show a stabilized system, but the state does not goes to zero after 800 time steps. A fading disturbance signal which is proportional to the state vector has been added to the model as an input disturbance, to show the effectiveness of the self-recovery constraint.

In the second simulation, the gain is re-computed for the three transition events. The corresponding trends are shown in Fig. 12b. The control trend with the re-computed gains indicates a different trajectory compared to that in Fig. 12a, using the same disturbance pattern and magnitudes. As a result, the state vector reaches zero after 600 time steps, which is better than that in Fig. 12a. The three re-computed gains are provided below for information:

$$10^{-2} \times \begin{bmatrix} 3.04 & 3.06 & 3.40 & 77.53 & 3.16 & 3.17 & 3.10 & 2.11 \\ 2.34 & 2.29 & 14.11 & -0.06 & 1.82 & 13.35 & 6.87 & 28.58 \end{bmatrix},$$

$$10^{-2} \times \begin{bmatrix} 2.82 & 2.91 & 3.05 & 77.59 & 3.01 & 2.85 & 2.82 & 1.94 \\ 2.23 & 1.97 & 14.05 & -.06 & 1.57 & 13.11 & 6.89 & 28.18 \end{bmatrix},$$

$$10^{-2} \times \begin{bmatrix} 2.68 & 2.78 & 2.85 & 77.63 & 2.91 & 2.64 & 2.67 & 1.82 \\ 2.25 & 1.83 & 14.28 & -.05 & 1.40 & 13.37 & 7.21 & 27.81 \end{bmatrix}.$$

When there are not any transition events, the state also reaches zero after 600 time steps, which is compatible to those of the DepCS having re-computed feedback gains in Fig. 12b. The settling time in the case of applying the re-computed gains is around 600 time steps, which is also the settling time of the case without having duty-standby transition. The re-computation uses the value of $\xi_{\Delta(k_s)}$ transferred from the previous active duty controller. The trend of self-recovery constraints is depicted in Fig. 13 for the pre-computed and re-computed gains. The latter demonstrates a smoother trajectory and settling time conservation, owing to the gain re-computation using the passing-on value of the self-recovery constraint.

Two other cases of different time instants and intervals of transition events are given in Fig. 14. Both cases show the improved performances of the DSC by employing gain re-computations, evaluated by the accumulative sum of $V = x^T P x$, as provided in Fig. 14.

The simulation study has demonstrated that the settling time of DepCS is approximately maintained in the three events of duty-standby switching-over, as a result of applying the self-recovery constraint.

We have also made some simulation comparisons between the presented control design method and those from the well-known LQR method. A typical result is given in Fig. 15, wherein both methods delivered compatible results. The re-computed gains for DepCS also achieve a better performance in comparison to LQR. Further, the self-recovery constraint method does not required a tuning of the weighting matrices Q, R, while the LQR does.

5.2. Automatic generation control

Another example with the centralized Automatic Generation Control (AGC) of a power system [31,32] employing the self-recovery constraint has shown similar results in stabilization and improved performance. Fig. 16 shows a simulation result using the model parameters borrowed from [31]. The performance from using the re-computed gains is improved by 35%.

6. Conclusion

A dependable self-recovery control (DSC) architecture that utilizes Internet of Things (IoT) connectivity and dependable control systems (DepCSs) has been presented in the first part of this paper as an alternative to the widely perceived architecture for the industrial computerized-control systems. The newly presented Industrial Internet of Things (IIoT) enabled system is capable of accommodating a fully decentralized architecture and Plug-and-Play (PnP) operation using wired-line or wireless communication networks, yet meet the quantitative reliability specification. The marshaling and I/O systems, as well as centralized processing system, are not required in the new IoT enabled system. Therefore, the installation, implementation and maintenance costs will be significantly reduced thanks to the IoT connectivity.

In the second part, we have presented a novel state feedback control method for DepCS applying the self-recovery constraint with respect to variable increments. The state feedback gain is re-computed at every duty-standby switching-over incidences. The gain re-computation ensures that the incremental constraint is satisfied and the control performance is not degraded. Numerical simulations for an isolated wind-diesel power system and for the centralized AGC of four areas in a power system have demonstrated the effectiveness of the self-recovery constraint for dependable control systems.

Acknowledgments

This work was supported by the National Research Foundation (NRF) of Singapore under its Campus for Research Excellence And Technological Enterprise (CREATE) programme, and the

Cambridge Centre for Advanced Research in Energy Efficiency in Singapore (Cambridge CARES), C4T project.

References

[1] Schooman ML. Reliability of computer systems and networks: fault tolerance, analysis and design. New York: Wiley-Interscience; 2001.

[2] Gruhn P, Cheddie H. Safety instrumented systems: design, analysis and justification, 2nd ed. International Society of Automation – ISA, Research Triangle Park, NC; 2006.

[3] Xu LD, He W, Li S. Internet of Things in industries: a survey. IEEE Trans Ind Inf 2014;10(4):2233–43.

[4] Vyatkin V. Software engineering in industrial automation: state-of-the-art review. IEEE Trans Ind Inf 2013;9(3):1234–49.

[5] Govindaraju R, Lukman K, Chandra DR. Manufacturing execution system design using ISA-95. Adv Mater Res 2014;980:248–52.

[6] Yan J, Vyatkin V. Distributed software architecture enabling peer-to-peer communicating controllers. IEEE Trans Ind Inf 2013;9(4):2200–9.

[7] Candell R, Stouffer K, Anand D. A cybersecurity testbed for industrial control systems. In: Proceedings of the 2014 process control and safety symposium, ISA; 2014.

[8] Patton M, Gross E, Chinn R, Forbis S, Walker L, Chen H. Uninvited connections – A study of vulnerable devices on the Internet of Things. In: Proceedings of the IEEE joint intelligence and security informatics conference. Hague, the Netherlands; September 2014. p. 232–5.

[9] Lu T, Guo X, Li Y, Peng Y, Zhang X, Xie F, et al. Cyberphysical security for industrial control systems based on wireless sensor networks. International J Distrib Sens Netw 438350 (2014) 1–17 (Article ID).

[10] Granjal J, Monteiro E, Silva JS. Security for the Internet of Things: A survey of existing protocols and open research issues. IEEE Commun Surv Tutor 2015;17 (3):1294–312.

[11] Aijaz A, Aghvami AH. Cognitive machine-to-machine communications for Internet-of-Things: a protocol stack perspective. IEEE Internet Things J 2015;2 (2):103–12.

[12] Perera C, Liu CH, Jayawardena S. The emerging Internet of Things marketplace from an industrial perspective: a survey. IEEE Trans Emerg Top Comput 2015 http://dx.doi.org/10.1109/TETC.2015.2390034 .

[13] Gubbia J, Buyyab R, Marusic S, Palaniswami M. Internet of Things (IoT): a vision, architectural elements, and future directions. Future Gener Comput Syst 2013;29:1645–60.

[14] Miorandi D, Sicari S, Pellegrini FD, Chlamtac I. Internet of Things (IoT): a vision, applications and research challenges. Ad Hoc Netw 2012;10:1497–516.

[15] Atzori L, Iera A, Morabito G. Internet of things: a survey. Comput Netw 2010;54:2787–805.

[16] Befekadu GK, Gupta V, Antsaklis PJ. On reliable stabilization via rectangular dilated LMIs and dissipativity-based certifications. IEEE Trans Autom Control 2013;58(3):792–6.

[17] Tri Tran. Dependable model predictive control for reliable constrained systems. In: Proceedings of the 2nd IEEE international conference on control, automation and information science, ICCAIS'13, Nha Trang, Vietnam; 2013. p. 128–33.

[18] Tri Tran, Ha QP. Dependable control systems with self-recovery constraint. In: Proceedings of the 3rd international conference on automation and information science, ICCAIS'14. Gwangyu, South Korea; 2014. p. 87–92.

[19] Christofides PD, Davis JF, El-Farra NH, Clark D, Harris KR, Gipson JN. Smart plant operation: vision, progress and challenges. AiChE J 2007;53:2734–41.

[20] Gupta V, Dana AF, Hespanha JP, Murray RM, Hassibi B. Data transmission over networks for estimation and control. IEEE Trans Autom Control 2009;54 (8):1807–19.

[21] Sanders SR, Verghese GC. Lyapunov-based control for switched power converters. IEEE Trans Power Electron 1992;7(1):17–24.

[22] Pavlov A, Marconi L. Incremental passivity and output regulation. Syst Control Lett 2008;57:400–9.

[23] Hines GH, Arcak M, Packard AK. Equilibrium-independent passivity: a new definition and numerical certification. Automatica 2011;47(9):1949–56.

[24] Forni F, Sepulchre R. On differentially dissipative dynamical systems. In: Proceedings of IFAC symposium in nonlinear control systems, NOLCOS'13. Toulouse, France; September 2013). arXiv:1305.3456.

[25] Forni F, Sepulchre R. A differential Lyapunov framework for contraction analysis. IEEE Trans Autom Control 2014;59(3):614–28.

[26] Boyd S, ElGhaoui L, Feron E, Balakrishnan V. Linear matrix inequalities in system and control theory. Philadelphia: SIAM; 1994.

[27] Valencia-Palomo G, Rossiter JA. Programmable logic controller implementation of an auto-tuned predictive control based on minimal plant information. ISA Trans 2011;50(1):92–100.

[28] Li J-N, Er M-J, Tan Y-K, Yu H-B, Zeng P. Adaptive sampling rate control for networked systems based on statistical characteristics of packet disordering. ISA Trans 2015.In Press, http://dx.doi.org/10.1016/j.isatra.2015.04.005.

[29] Rahmani B, Markazi AH, Seyfi B. A new method for control of networked systems with an experimental verification. ISA Trans 2015;56(2):299–307.

[30] Bhatti TS, Al-Ademi AAF, Bansal NK. Load frequency control of isolated wind diesel hybrid power systems. Energy Convers Manag 1997;38(9):829–37.

[31] Tran Tri, Ling K-V, Maciejowski JM. Application of quadratically-constrained model predictive control in power systems. In: Proceedings of the IEEE international conference on automatic control on robotics and vision. Singapore; December 2014. p. 193–8.

[32] Wood AJ, Woolenberg BF, Sheblé GB. Power generation operation and control. 3rd ed.. New Jersey: John Wiley & Son; 2013.