ROYAL INSTITUTE
OF TECHNOLOGY

Wireless Sensor Network Scheduling and Event-based Control for Industrial Processes

TAKUYA IWAKI

Licentiate Thesis
Stockholm, Sweden 2018

KTH Royal Institute of Technology
School of Electrical Engineering and Computer Science
Department of Automatic Control
SE-100 44 Stockholm
SWEDEN

TRITA-EECS-AVL-2018:82
ISBN 978-91-7729-996-7

Akademisk avhandling som med tillstånd av Kungliga Tekniska högskolan framlägges till offentlig granskning för avläggande av teknologie licentiatesexamen i elektro- och systemteknik fridag den 23 november 2018 klockan 10.00 i sal F2, Lindstedtsvägen 26, KTH Campus, Stockholm.

© Takuya Iwaki, November 2018.

Tryck: Universitetsservice US AB

Abstract

Control over wireless sensor and actuator networks is of growing interest in process industry since it enables flexible design, deployment, operation, and maintenance. An important problem in industrial wireless control is how to limit the amount of information that needs to be exchanged over the network. In this thesis, network scheduling and remote control co-design is considered to address this problem.

In the first part, we propose a design of an optimal network schedule for state estimation over a multi-hop wireless sensor network. We formulate an optimization problem, minimizing a linear combination of the averaged estimation error and transmission energy. A periodic network schedule is obtained, which specifies when and through which routes each sensor in the network should transmit its measurement, so that an optimal remote estimate under sensor energy consideration is achieved. We also propose some suboptimal schedules to reduce the computational load. The effectiveness of the suboptimal schedules is evaluated in numerical examples.

In the second part, we propose a co-design framework for sensor scheduling, routing, and control over a multi-hop wireless sensor and actuator network. For a decoupled plant and LQG control performance, we formulate an optimization problem and show that the optimal schedule, routing, and control can be obtained locally for each control loop. In this part, we also introduce algorithms to reconfigure the schedules and routes when a link in the network is disconnected. The results are illustrated in a numerical example.

In the third part, we consider event-based feedforward control from a wireless disturbance sensor. We derive stability conditions when the closed-loop system is subject to actuator saturation. Feedforward control with anti-windup compensation is introduced to reduce the effect of actuator saturation. The effectiveness of the approach is illustrated in some numerical examples.

Sammanfattning

Reglerings över nätverk genom trådlösa mätgivare och ställdon är av växande intresse för processindustrin eftersom det möjliggör flexibel design, implementering, drift och underhåll. Ett viktigt problem vid trådlös reglering är hur man minskar mängden information som behörer utbytas över nätverket. I denna avhandling betraktas schemaläggning och fjärrstyrning för trådlša reglersytem.

I den första delen föreslår vi ett optimalt nätverksschema för tillståndsuppskattning över ett trådlöst multi-hop sensorsnätverk. Vi formulerar ett optimeringsproblem där en linjärkombination av det genomsnittliga uppskattningsfelet och överföringsenergin minimeras. Ett periodiskt nätverksschema erhålls, vilket anger när och genom vilka rutter varje mätgivare i nätverket bör överföra sin mätning för att uppnå den optimala tillståndsuppskattningen. För att minska beräkningsbelastningen så föreslår vi några suboptimala scheman och deras effektivitet utvärderas i numeriska exempel.

I den andra delen föreslår vi ett gemensamt ramverk för mätgivarplanering, dirigering, och reglering över ett trådlöst multi-hop nätverk av mätgivare och ställdon. Vi formulerar ett LQG-optimeringsproblem för ett frikopplat reglersystem och visar att det optimala schemat, rutten och regleringen kan erhållas lokalt för varje reglerslinga. Vi introducerar även algoritmer för att omkonfigurera scheman och rutter när en länk i nätverket går ner. Resultaten illustreras i ett numeriskt exempel.

I den tredje delen betraktar vi händelsestyrd framkoppling från en trådlös mätgivare som mäteren störning. Vi härleder stabilitetskriterier när systemet har ställdon som kan bli mättade. Framkoppling med anti-windup introduceras för att minska effekten av mättade ställdon. Effektiviteten av tillvägagångssättet illustreras i numeriska exempel.

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my supervisor Professor Karl Henrik Johansson for providing me the opportunity to work on such interesting research topics. His continuous inspiration and insightful feedback have made this thesis possible. I would also thank to my co-supervisor Professor Henrik Sandberg for giving me detailed support in my research.

I am grateful to my collaborators Junfeng Wu and Yuchi Wu, for the fruitful discussions with them and their careful attention to our work. I am also thankful to Professor Anders Ahlén, André Teixeira, Alf Isaksson, Johan Åkerberg, Markus Eriksson, Steffi Knorn, and Thomas Lindh for enjoyable discussion in the VINNOVA PiiA project. Heartfelt thanks to Professor Emilia Fridman for her valuable advice to my research. Special thanks to Mohammadhossein Mamduhi, Xiaoqiang Ren, and Kazumune Hashimoto for proof reading of this thesis, and Alexander Johansson for translating the abstract into Swedish.

I would also express my appreciation to all my current and former colleagues at the Automatic Control department for creating a friendly environment and an active working atmosphere, and for their continuous support for everything that I needed. Especially, I thank José Mairton Barros da Silva Júnior, Dirk van Dooren, Yulong Gao, Xinlei Yi, Pian Yu, Mohammadhossein Mamduhi, Ehsan Nekouei, Xiaoqiang Ren, Jieqiang Wei, Han Zhang, Kazumune Hashimoto, Touraj Soleymani, and Takashi Tanaka for interesting discussions and supportive comments. I also want to thank the administrators in our department Silvia Cardenas Svensson, Hanna Holmqvist, Anneli Ström, and Felicia Gustafsson for their assistance and support.

Last, but not least, I would like to thank my wife Kyoko Iwaki for her unconditional love and constant support and my daughter Sakura Iwaki for always being lively with full of smiles.

Takuya Iwaki
Stockholm, November 2018

Contents

Notations

$\mathbb{R}$	Set of real numbers
$\mathbb{R}^n$	Set of real n–dimensional vectors
$\mathbb{R}^{n\times m}$	Set of real $(n \times m)$–dimensional matrices
$\mathbb{N}$	Set of nonnegative integers
$\mathbb{S}^n_{++}$	Set of n-by-n positive definite matrices
$\mathbb{S}^n_{+}$	Set of n-by-n positive semi-definite matrices
$x[i]$	The i-th element of vector x
$X[i]$	The i-th row vector of matrix X
I_n	n-by-n identity matrix
$X^\top$	Transpose of real matrix X
$\mathbf{1}_n$	n-by-1 vector of all ones
$\lambda_{\max}(A)$	the largest eigenvalue of A; A has real eigenvalues
$X > Y$	$X - Y$ is positive definite
$X \geq Y$	$X - Y$ is positive semi-definite
$\mathrm{tr}(A)$	Trace of A
$\emptyset$	Empty set
$\lvert\mathcal{S}\rvert$	Cardinality of set $\mathcal{S}$
$2^{\mathcal{S}}$	Power set of $\mathcal{S}$
$\mathrm{int}(\mathcal{S})$	Interior of set $\mathcal{S}$
$\partial\mathcal{S}$	Boundary of set $\mathcal{S}$
$\begin{bmatrix} \boldsymbol{A} & \star \\ \boldsymbol{B} & \boldsymbol{C} \end{bmatrix}$	Symmetric matrix of the form $\begin{bmatrix} \boldsymbol{A} & \boldsymbol{B}^\top \\ \boldsymbol{B} & \boldsymbol{C} \end{bmatrix}$

Chapter 1

Introduction

Recent advances in wireless communication, sensing, and computation have dramatically changed our life and society. Industrial process control systems have also become highly digitalized. Now, attempts are made to integrate wireless communication into industrial process control systems. The usage of wireless communication in these systems enables more effective design, deployment, operation, and maintenance. As these systems are often safety-critical, a systematic approach with guaranteed performance needs to be developed.

In spite of many recent advances in wireless process control, some challenges arise due to the fact that wireless communication introduces non-negligible delays and transmission failures. These imperfections result in economic losses due to control performance degradation, or may even threaten human safety. In addition to this, wireless devices are often locally powered, so they need to be in continuous operation without battery shortages.

In this thesis, our objective is to provide a co-design framework for designing control and scheduling under trade-off between control performance and communication cost.

The rest of this chapter is organized as follows. In Section 1.1, we motivate the thesis. We outline the problem in Section 1.2. Section 1.3 presents the structure of the thesis and summarizes the contributions.

1.1 Motivation

Industrial process control systems are used in processing plants, such as oil and gas refining, chemical processing, pulp and paper manufacturing, and power generation plants, to ensure proper functionality. Real-time regulatory control of these plants is performed based on the architecture shown in Figure 1.1. At the top layer, operator consoles are located. Plant operators check the plant status and change setpoints, controller parameters, etc., through the operator interface. Control computations are carried out in the controllers at the middle layer and communicate with field sensors and actuators at the bottom layer. Sensors are installed in the field of the plant to monitor physical processes. Sensor information is transmitted to the controllers to compute the control signals and then sent to actuators to adjust the state of physical processes. Sensor and actuator signals are also provided to the operators. The detailed structure of the system is introduced in Chapter 2.

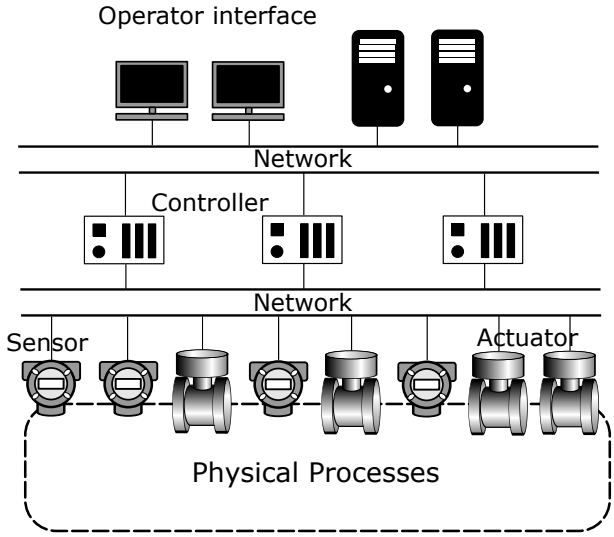

Figure 1.1: A typical architecture of a part of industrial process control systems. It consists of three layers. The top layer is used for operator monitoring and data collection. The controllers are located at the middle layer. The field sensors and actuators are distributed over the bottom layer.

Figure 1.2: Paperboard machine in the Iggesund paper mill. High quality paperboard is manufactured in the machine after starch is mixed to coat the paperboard.

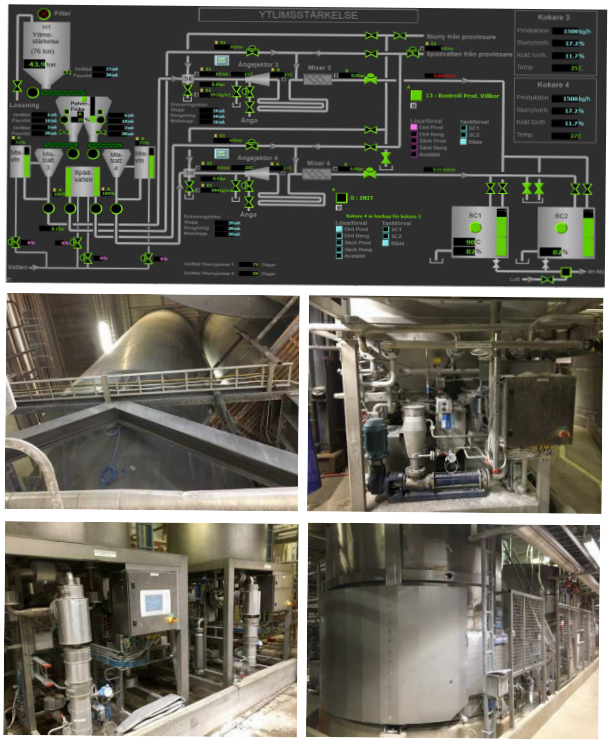

Figure 1.3: Operation panel of the starch cooker process and some equipment. Top: Operation panel of the starch cooker process. Middle left: Starch buffer. Middle right: Mixing tank for starch powder. Bottom left: Starch boiler. Bottom right: Storage tank for boiled starch.

Figure 1.2 shows the Iggesund paper mill, which produce paperboard, a thick paper-based material for high quality packing boxes and graphic printing papers. Before the paperboard is produced in the mill, it is mixed with starch. The starch is produced in a starch cooker process. Figure 1.3 shows an operation panel of the starch cooker process and some equipment in it. In this process, the dry starch powder is stored in a buffer (middle left) at first, which is then transported to a mixing tank (middle right) to mix with water. The mixture is then boiled by steam at the starch boiler (bottom left) to adjust the concentration of the starch. The starch with desired concentration is stored in a storage tank (bottom right). A detailed process flow and control block diagram is shown in Figure 1.4. This process is controlled by seven control loops. At first, water from the mix water tank, the level of which is controlled by the level control loop, is mixed with the starch powder from the mix funnel. Then the obtained starch-water mixture, which is controlled by the coarse flow control loop, is sent to the steam ejector. The concentration of the mixture is monitored to adjust the amount of starch powder to be provided. Next, the starch-water

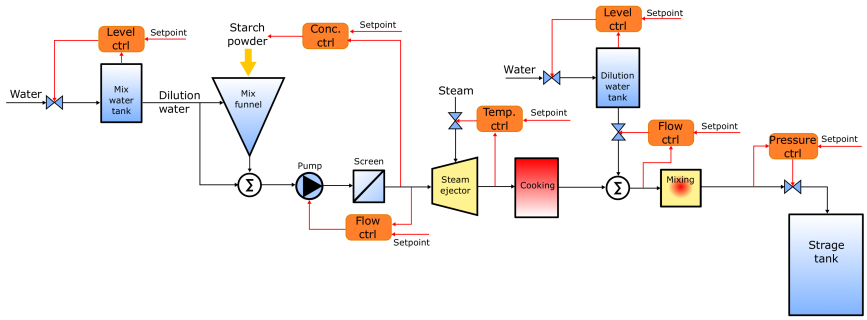

Figure 1.4: The process flow diagram of the starch cooker unit. The starch powder is mixed with water at first. Then the mixture is heated up. The heated starch is then stored at the storage tank after fine adjustment of the concentration.

mixture is heated up by steam at the steam ejector. Here, the temperature is controlled by the ejector temperature control loop. The mixture concentration is further adjusted by adding fine water from the dilution water tank, the level of which is regulated by the level control loop. The flow rate of the fine water is controlled by flow control. Finally, after the pressure of the obtained starch is regulated by the pressure control loop, it is stored in the storage tank.

The starch cooker process is controlled by multiple decoupled control loops to achieve the objective of the unit of the plant. These loops could potentially be controlled by wireless sensors and actuators in the future. Using wireless communication does not require point-to-point hard-wired cabling [1]. This means that there is no cost of the cabling and its maintenance. Furthermore, control over wireless communication allows operators to use mobile devices, which leads to more efficient operation of the plant. In addition to this, wireless sensors can be placed where the sensors with hard-wired cables cannot be installed such as on the rotary machines. Thus, the control system can be easily improved by deploying new sensors or actuators. For example, the concentration of the starch in the starch cooker process is possibly disturbed by the change of steam flow rate into the steam ejector or the change of starch concentration after the screen. The effective way to compensate the disturbances is to deploy a wireless steam flow sensor in order to adjust the fine water flow rate by feedforward control before the disturbances affects the concentration of the starch. Wireless feedforward control is discussed in Chapter 5.

Two major wireless communication frameworks have been proposed for the process industry: WirelessHART [2] by ABB, Emerson, e.g., and ISA-100 [3] by Honeywell, Yokogawa, e.g., both of which hardware and protocols are specified by the standard of the low-rate wireless personal area network, IEEE 802.15.4 [4]. Although WirelessHART and ISA-100 are available for process control systems, only monitoring applications have been practically used so far. Feedback control applications are more critical than monitoring and therefore requires more reliable communication. Since the wireless communication

may incur packet dropouts or delays and usually have no inexhaustible or reliable energy sources, the control over wireless communication in process plants has not been commonly realized yet. More effective scheduling and routing for wireless networks are proposed in Chapter 3–4.

In summary, recent wireless technologies enable the industrial process control systems to communicate through wireless networks. However, some drawbacks such as packet dropouts, delays, and battery limitation have to be considered when we introduce the wireless communication to the critical infrastructure systems. Although some industrial standards have been proposed so far, it is not yet commonly applied to industrial process control.

1.2 Problem Formulation

In this thesis, we address two problems, relevant for the realization of wireless control in process industry. The first one is the problem of co-design of sensor scheduling, routing, estimation, and control over multi-hop wireless networks. The second one is the problem of event-based control for some specific control architectures. The common idea is to consider some constraints relevant for industrial process control systems, such as the ones in the Iggesund paper mill described in Section 1.1. For example, Problem 1, we described below, focuses on network scheduling of multi-hop wireless networks. It is important to consider multi-hop wireless networks since they are used in WirelessHART and ISA-100, but the scheduling problems in previous work mainly focus on single-hop networks. In Problem 2, we focus on event-based sampling introduced in feedforward control. How the event-based sampling of the feedforward control affects the control performance is still unclear even though feedforward control is widely used in process control systems. In what follows, we introduce these problems in more details.

Problem 1 – Sensor network schedule for wireless multi-hop networks

In the next generation process control systems, the controllers, sensors, and actuators communicate with each other over a wireless sensor and actuator network [5–7]. WirelessHART and ISA-100 are the major standards designed for the process control systems with wireless communication. The network structure of these standards is shown in Figure 1.5. The wireless sensors and actuators are connected with their neighbor sensors and actuators which have a function of intermediate network nodes. Thus, the network is configured as a mesh-structured multi-hop network. Each sensor transmits its measurement data to the remote estimator and the controller at the control station. The control signal computed by the controller is sent back to the network and the corresponding actuator. Since sensors in the network are used as intermediate nodes for other sensors and actuators in different control loops, the availability of the intermediate sensors affects the control performance of the other loops. Thus, it is important to consider the scheduling and the routing of the network as a whole.

As inspired by the starch cooker process, sensor–actuator pairs control each decoupled linear process. Let us then consider the linear dynamical systems as the description of the

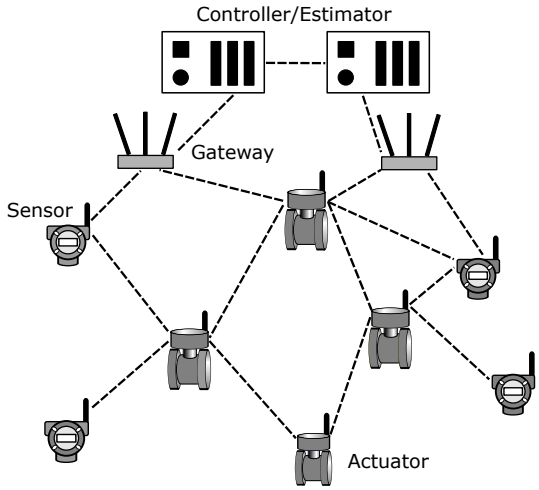

Figure 1.5: Illustration of a wireless network control system for industrial process plants considered in WirelessHART and ISA-100. Distributed sensors and actuators are connected to their neighbors, and configure a mesh-structured multi-hop network.

processes,

$$x_{k+1}^{(i)} = A_i x_k^{(i)} + B_i u_k^{(i)} + w_k^{(i)}$$
$$y_k^{(i)} = C_i x_k^{(i)} + v_k^{(i)}$$

where $x_k^{(i)}$ is the state of process i, $u_k^{(i)}$ is control signal to actuator i, and $y_k^{(i)}$ is the measurement of sensor i, all at time k. We consider that there exist process noise $w_k^{(i)}$ and measurement noise $v_k^{(i)}$, which assumed to be zero-mean i.i.d. Gaussian random processes. The system describes the behavior of process states such as flow rate, pressure, temperature, and level. A block diagram of a typical industrial wireless process control systems is depicted in Figure 1.6. Each sensor transmits its state estimate $\hat{x}_{s,k}^{(i)}$ to the remote estimator or the controller. Then the controllers compute their control signals and send to the corresponding actuators. The data is transmitted over a multi-hop wireless network. The network is described as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a set of network nodes, which contains the sensors and actuators. Links between network nodes are represented by the edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. Then to convey the data from sensor i to controller i, the nodes on the path from the sensor to the controller necessarily consume energy.

The problem is now how to find a sensor network scheduling and routing strategy, i.e., at which time instance and along which path sensors should transmit their data towards the controller. For this purpose, we formulate the abstract optimization problem:

minimize	Control/Estimation cost
subject to	Process dynamics,

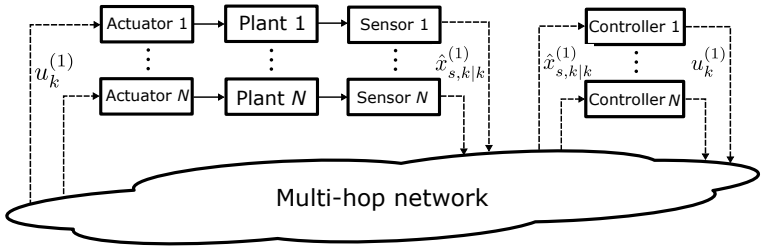

Figure 1.6: A block diagram of wireless process control system. A sensor and an actuator in each system communicate with the corresponding controller through a multi-hop wireless network.

Energy constraints,

Data flow constraints.

Based on this optimization problem, we discuss two specific subproblems. The first one, which is discussed in Chapter 3, is the optimal estimation problem. In this problem, we focus on optimizing the estimation performance at the remote control station. The network is configured only by sensors without actuators. In this problem, we consider data aggregation as some sensors share their paths to the remote estimator. Then we propose a co-design framework for sensor scheduling and routing with optimal remote estimation. The estimation quality is defined by the trace of the estimation error covariance.

In Chapter 4, we address an optimal control problem. In this problem, we consider that the controllers are co-located at the corresponding actuators. The development of smart sensors and smart actuators enables local estimation and control, which leads to a distributed architecture for industrial control. We propose a co-design framework for sensor scheduling and routing while minimizing a linear-quadratic-Gaussian (LQG) control cost. Furthermore, we also propose algorithms to detect link outage and to switch to another route in this case.

Problem 2 – Event-based feedforward control with wireless disturbance sensors

The second approach we address in this thesis is event-based control for some specific control architectures. Especially, we focus on event-based feedforward control and event-based ratio control when combined with PID control. PID control is still the first choice of industrial process control. In fact, as seen in Figure 1.4, the starch cooker process is controlled by seven PID control loops. To see the advantages of event-based PID control, define the first-order process

$$\dot{x}(t) = ax(t) + bu(t)$$
$$y(t) = cx(t)$$

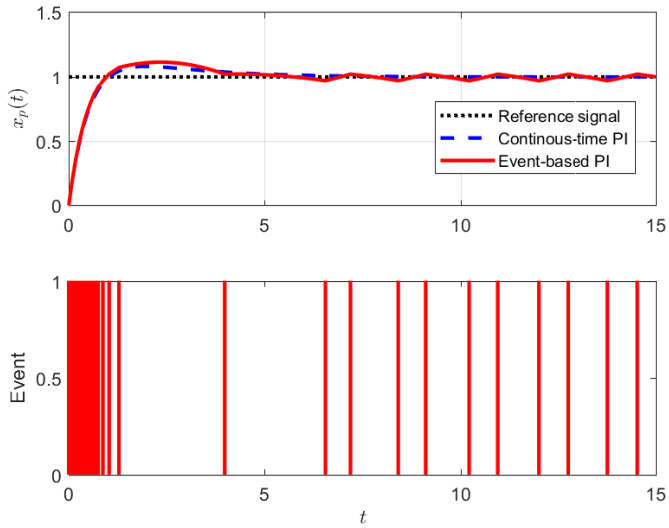

Figure 1.7: Top: Step response of the first-order process with the event-based PI control and continuous-time PI control. Bottom: Time instances of event-generation of the event-based PI control. The event-based control with only 31 samplings does not degrade the response compared to the continuous one.

where $x(t) \in \mathbb{R}$, $u(t) \in \mathbb{R}$, $y(t) \in \mathbb{R}$ denote the state, control signal, and measurement, respectively. For this system, consider an event-based PI controller

$$\begin{aligned} \dot{x}_c(t) &= y(t_k) - r(t) \\ u(t) &= K_i x_c(t) + K_p y(t_k) \end{aligned}$$

where $x_c(t) \in \mathbb{R}$ is controller state, t_k is the time of k-th measurement transmission, and K_i, K_p are PI controller gains. Figure 1.7 shows the step response with $r(t) = 1$, $\forall t > 0$ of the system with $a = -0.1$, $b = 1$ and the parameter $K_i = -1$ and $K_p = -2.3$. In the upper figure, the red solid line indicates the response of the event-based PI control with event generation rule; measurement data is transmitted when

$$|y(t) - y(t_k)| \geq 0.1,$$

and the blue dashed line indicates that of the continuous-time PI control. We can see that event-based PI control achieves a similar step response compared to the continuous-time PI control even though the number of samples is only 31 as shown in the bottom of the figure. Event-based PID control is hence efficient in reducing the transmission rate without performance degradation. Motivated by this fact, we focus on the enhanced control loops which are used in process control systems. In Chapter 5, we deal with event-based

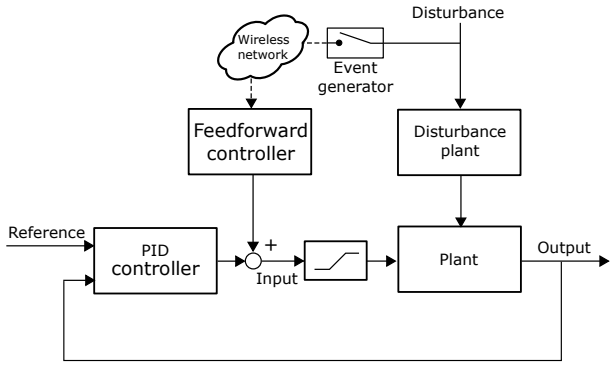

Figure 1.8: A block diagram of event-based feedforward control. The disturbance is measured directly by a sensor. The measured value of the disturbance is sent to a feedforward controller when the event is generated.

feedforward control combined with a PI controller. We consider actuator saturation which occurs in almost all systems due to physical or safety reasons. Especially, in process automation system, the actuators such as valves and pumps are strictly constrained by the possible valve openings or the maximum voltage for the pumps. Thus, we consider the system depicted in Figure 1.8. Event-based sampling is now used in the disturbance sensor. The research problems to be answered in Chapter 5 are how the event-based sampling for the feedforward control affects the stability of the system subject to actuator saturation, and how it reduces the effect of disturbances.

1.3 Thesis Outline and Contributions

The thesis is organized as follows.

Chapter 1: Introduction

This chapter so far motivates the thesis and states the problems with the starch cooker process in the Iggesund paper mill. The discussion is covered by the following contribution.

- A. Ahlén, J. Åkerberg, M. Eriksson, A. J. Isaksson, T. Iwaki, K. H. Johansson, S. Knorn, T. Lindh, and H. Sandberg, "Towards wireless control in industrial process automation," *To be submitted to IEEE Control Systems Magazine*.

Chapter 2: Background

In Chapter 2, we first provide a brief overview of the structure of industrial process control systems. Then we summarize some work on networked control using common wireless protocols. We also outline available theoretical results of networked control systems.

Chapter 3: Sensor network scheduling for remote estimation

In Chapter 3, we propose a design framework of network scheduling for remote estimation over a multi-hop wireless sensor network. We formulate an optimization problem, minimizing estimation error under sensor energy constraints. Some necessary conditions are derived, which transform the optimization problem into a finite-state Markov Decision Process (MDP) problem. It is shown that there exists a periodic optimal network schedule. We also propose design frameworks for suboptimal schedules which can reduce the computational load. Numerical examples are provided to validate the theoretical results and illustrate the effectiveness of the suboptimal schedules. The covered material is based on the following contributions.

- T. Iwaki, Y. Wu, J. Wu, H. Sandberg, and K. H. Johansson, "Wireless sensor network scheduling for remote estimation under energy constraints," *in Proc. of IEEE Conf. on Decision and Control*, 2017, pp. 3362–3367.
- T. Iwaki, Y. Wu, J. Wu, H. Sandberg, and K. H. Johansson, "Sensor and multi-hop network scheduling for optimal remote estimation," *in Preparation*.

Chapter 4: Co-design of scheduling, routing, and LQG controller

Chapter 4 covers a problem of LQG control for multiple decoupled linear systems over a shared multi-hop wireless sensor and actuator network. Under the assumption that controllers are co-located with the corresponding actuators, we propose a co-design framework of control, routing, and scheduling of the system and the network, which can be obtained by solving optimization problems locally for each control loop. Routing reconfiguration algorithms when a link in the network is disconnected are also discussed in this chapter. The results are illustrated in a numerical example. The covered material is based on the following contribution.

- T. Iwaki, and K. H. Johansson, "LQG control and scheduling co-design for wireless sensor and actuator networks," *in Proc. of IEEE Workshop on Signal Processing Advances in Wireless Communications*, 2018, pp. 146–150.

Chapter 5: Event-based feedforward control subject to actuator saturation

In Chapter 5, event-based feedforward control and ratio control for output feedback control loops is studied. We derive the stability conditions subject to actuator saturation in the form of Linear Matrix Inequalities (LMIs). A compensation technique for actuator saturation by using anti-windup compensation is also discussed in this chapter. The effectiveness of the approach is illustrated in some numerical examples. The covered material is based on the following contributions.

- T. Iwaki, J. Wu, and K. H. Johansson, "Event-triggered feedforward control subject to actuator saturation for disturbance compensation," *in Proc. of European Control Conf.*, 2018.

Chapter 6: Conclusions and future research

In Chapter 6, we present a summary of the results, and discuss directions for future research.

Contributions not covered in the thesis

The following publication by the author is not covered in the thesis, but contains related material:

- Y. Wu, T. Iwaki, J. Wu, K. H. Johansson, and L. Shi, "Sensor selection and routing design for state estimation over Wireless Sensor Networks," *in Proc. of Chinese Control Conf.*, 2017, pp. 8008–8013.

Contribution by the Author

The order of authors reflects their contribution to each paper except the one covering Chapter 1. The first author has the most important contribution. In all the listed publications, all the authors were actively involved in formulating the problems, developing the solutions, evaluating the results, and writing the paper.

Chapter 2

Background

In this chapter, we introduce some background material of the thesis. There is a lot of research on wireless process control both from an industrial and an academic perspective. Some industrial communication protocols are discussed [6–8]. Netwroked control theory mainly focuses on how to deal with drawbacks induced by using wireless communication [9–11]. The chapter covers the following two topics. In Section 2.1, we first overview the architecture of industrial process control systems and some Medium Access Control (MAC) protocols used in the industrial process control. In Section 2.2, we overview theoretical work in control over wireless networks.

2.1 Wireless Industrial Process Control Systems

Recent wireless communication technology brings industrial process control systems into a new stage. In this section, we overview the development of the industrial process control systems and see how the system configuration is changed in the next generation wireless control systems to clarify the challenges considered in this thesis.

2.1.1 A brief history of industrial process control systems

Industrial process control systems are introduced in processing plants such as oil and gas refining, chemical processing, pulp and paper manufacturing, and power generation plants. The aims of the systems are to control the plants properly to obtain a desired amount of products with acceptable quality. Furthermore, maintaining the plants in a safe state is another important role of the systems. Processing plants were first constructed at the end of the 19th century. At that time, the plants were manually operated, i.e., plant operators directly opened or closed control valves at the field of the plant or operated by local regulators such as centrifugal governors. After that, control of process plants can be performed remotely. PID control is implemented by mechanical PID regulators, and the signal between field devices and PID controllers were communicated through pneumatic pressure.

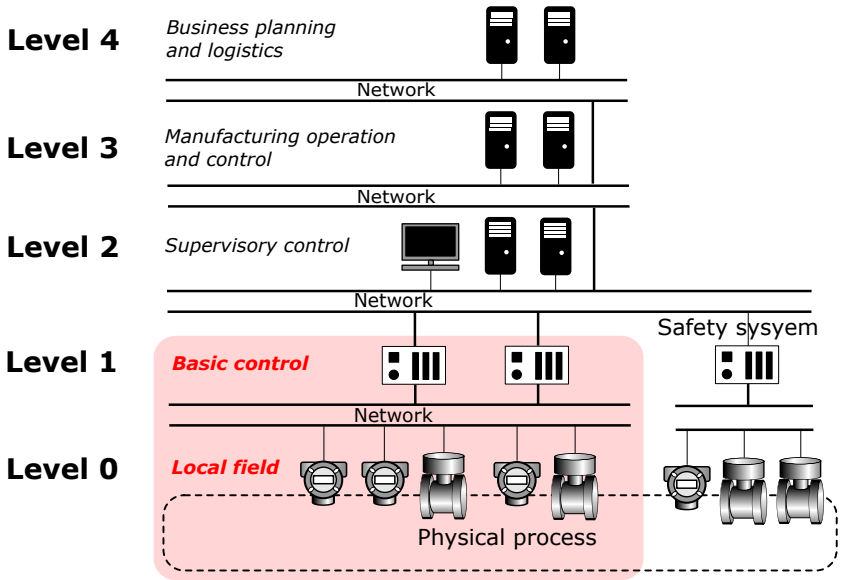

Figure 2.1: The detailed architecture of industrial process control systems [12]. The overall system has a hierarchical structure which is divided into five levels. We focus on control and communication between Level 0 and Level 1 in this thesis (highlighted in red).

Electrical signal replaced pneumatic pressure when digital computers were introduced to industrial process control systems in the 1970s [1]. Controllers communicate with field devices through an analog electric current between 4 mA and 20 mA. Now, the signal between controllers and field devices are digitalized. However, even though many digital communication technologies such as Foundation Fieldbus [13] and Profibus [14] are available, 4-20 mA analog signals are still used in many industrial process control systems.

A typical configuration of an industrial process control system is illustrated in Figure 2.1 [12]. The system consists of five levels. In Level 0, i.e., field level, sensors and actuators are distributed. In Level 1, basic regulatory controllers are located. Those controllers are referred to as Distributed Control System (DCS), which are specialized for continuous monitoring and operation for processing plants. Safety systems which are used to detect abnormal plant conditions and perform an emergency shutdown are also placed in this level. However, the safety systems and corresponding sensors and actuators must be completely independent of the DCS due to safety reasons. Safety systems are out of the scope of this thesis, but we consider the part of the systems highlighted in red in Figure 2.1. In Level 2, operator interfaces are located. The operators can check the plant status, change the setpoint and parameters of the controllers through the interfaces. In this level, some other applications such as Asset Management Systems are located to monitor the conditions of the filed devices. In Level 3, advanced controllers such as Model Predictive Control are implemented. Some servers for business use are located at Level 4.

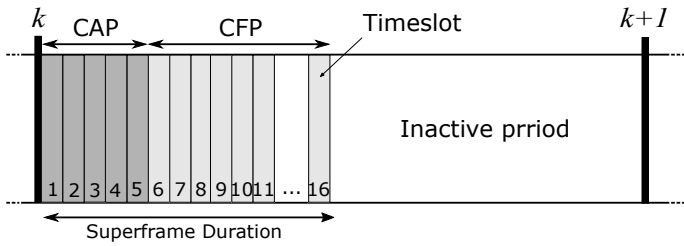

Figure 2.2: Superframe structure of IEEE 802.15.4 MAC protocol. Beacon interval between k and $k+1$ consists of three periods: CAP, CFP and inactive period.

2.1.2 Emergence of wireless process control systems

Wireless devices considered in this thesis are located in Level 0 and Level 1. They are connected with each other and configure a multi-hop sensor and actuator network as shown in Figure 1.5 in Chapter 1. Two well-established standards of wireless communication are WirelessHART and ISA-100, both of which are fully compliant with IEEE 802.15.4. The nodes in the network are periodically synchronized by a beacon message from a network manager. An interval between two beacons is called superframe which consists of three parts as shown in Figure 2.2. The first part is Contention Access Period (CAP), where all network nodes are allowed to access the channel. Hence, collisions may occur. To handle the possible collisions, Carrier Sense Multiple Access with collision avoidance (CSMA/CA) are introduced. The second part is Contention Free Period (CFP). In the CFP, timeslots are allocated to devices which want to transmit data. This scheme is called Time Division Multiple Access (TDMA). At the end of all the transmissions, the network becomes inactive to save the batteries of the network nodes. In Chapter 3, we consider a multi-hop wireless network with a simple MAC protocol. In this protocol, only the CFP is considered, and wireless scheduling framework is proposed.

Although some protocols have been developed and adopted for monitoring, they are not yet widely accepted for control [8, 15]. MAC protocols for industrial process control systems are studied [16–18]. In [19, 20], some key aspects of WirelessHART, ISA-100, and other protocols are compared. The authors of [21] propose a hybrid MAC protocol which switches between two modes: a Contention Access MAC (CA-MAC) mode and a Contention Free MAC (CF-MAC) mode. The modes transit from the CA-MAC to the CF-MAC when a large disturbance occurs in the process. In [22], aperiodic sampling and a MAC protocol with a scheduling algorithm is jointly designed for wireless control systems. An experimental evaluation with water tanks is also conducted in the paper. Furthermore, the authors of [23] develop a network model which captures the WirelessHART protocol and stability conditions are derived.

A future architecture of industrial process control systems is discussed in [5], which is based on the ExxonMobil automation vision [12]. A simplified version of this architecture is illustrated in Figure 2.3. In this vision, the hierarchical structure disappears. The DCS is replaced by Distributed Control Nodes (DCNs). Unlike the DCS, the DCNs are assigned a

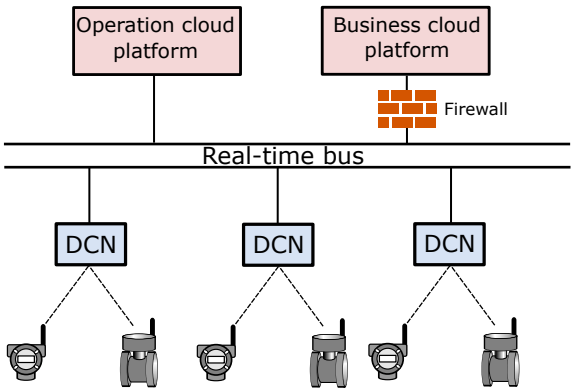

Figure 2.3: Future process control system architechture proposed in [12]

single control loop, which makes the system more robust against controller failures. As the DCNs only require capabilities of simple control computation and I/O processing, they can be located closer to the field. Being conscious of this architecture, we address a co-design framework of sensor scheduling, routing, and control when the controllers are co-located with their corresponding actuators in Chapter 4.

2.2 Networked Control Systems

Industrial process control systems over wireless networks can be modeled as networked control systems (NCSs), which are one of the most active research fields in the control community in recent years. We now overview relevant work that discuss control over wireless networks. In the wireless NCSs, communication between sensors, controllers, and actuators are carried out over wireless communication channels. In general, this leads to communication losses or delays due to limited capacity. Thus, one of the main research directions in NCSs are analysis and synthesis of dynamical control systems under these constraints [24, 25]. The models of NCSs can be categorized into two classes: i) stochastic systems where disturbances are modeled in a stochastic manner, and ii) deterministic systems where uncertain disturbances are modeled in a deterministic manner.

2.2.1 Stochastic NCSs

Stochastic NCSs have been widely studied [10,24,25]. In [26,27], LQG control with packet dropouts is considered. The authors of [10] include a stochastic packet dropout model into the LQG control problems and show that the acknowledgment of successful transmission from the receiver plays a fundamental role in the LQG optimal control problems. Then the problems of the NCSs with stochastic packet dropouts are extended with some varying network settings. In [28], a network with multiple sensors is considered. Furthermore, the

communication through intermediate nodes is studied in [29]. In [30, 31], control through an unreliable channel between a controller and an actuator is studied.

LQG control with network induced delays and access constraints is investigated in [32, 33] where only a subset of sensors can access the controller. Network capacity is explicitly considered as an interface problem between control theory and information theory in [11, 34]. In this context, the authors of [35, 36] propose the NCSs to minimize the data-rate while the privacy constraints are taken into consideration in [37]. The idea of non-stochastic information is considered in [38, 39]. Scheduling of data transmission of stochastic NCSs has also attracted attention in order to reduce the communication amount. In [40, 41], the joint optimization problem is presented where the problem can be separated into an optimal estimation, an optimal regulator, and an optimal scheduling problem by extending the result of [10] when the acknowledgment is available. Scheduling among multiple control loops with a shared communication network is proposed in some papers [42,43]. The authors of [42] propose a prioritizing framework under limited channel slots. The authors of [43,44] develop the scheduling framework under a MAC-like protocol for an estimation problem [44] and an optimal control problem [43].

There are a lot of research considering sensor scheduling problems for state estimation. In [45], a communication control scheme for Kalman filter is developed to improve the trade-off between estimation performance and communication cost. A stochastic sensor selection algorithm is proposed in [46] when a plant is monitored by multiple sensors but only one of them can access the estimator at every time instance. Optimal estimation with a multiple time-step cost is introduced in [47]. The minimum means square error (MMSE) estimation schedule can be obtained in some special cases. In [48], the MMSE schedule between two sensors is obtained, which is extended to more sensors in [49, 50]. These works deal with a single-hop network, i.e., every sensor can directly communicate with an estimator. Chapter 3, however, considers a scheduling problem for remote estimation over a multi-hop network. While these works provide offline schedules, the authors of [51] offer a deterministic online MMSE schedule by using feedback from the estimator. Stochastic online schedules are also studied in [52], and the MMSE stochastic schedule is proposed in [53]. The result of [53] is extended to the multiple sensors case in [54]. With a similar setup, a scheduling framework where the transmissions are invoked by estimation error covariance is proposed in [55]. In [56, 57], packet dropouts are also considered for covariance-based state estimation. These works are then extended to LQG control problem [58]. An obtained sensor schedule in Chapter 4 is based on the result in [58]. We consider LQG control over a multi-hop sensor and actuator network, while a single-hop network is considered in [58].

Further, the explicit energy consumption model is introduced into the covariance-based estimation [59]. In [60], TDMA-like time-triggered schedule and CSMA-like event-triggered schedule with random or dynamic scheduler are analyzed.

Multi-hop networks are explicitly considered in some studies. In [61, 62], the authors consider how to manage the NCSs when the network environment is changed. In [62], they propose the way to reconfigure the network under time-varying channel states. The authors of [63] discuss the controller placement in a multi-hop wireless sensor and actuator network.

Optimal energy allocation of the sensors of NCSs is studied in [64–66]. Here, the energy consumption is dealt with as a control variable, which determines the probability of packet loss. Energy allocation for state estimation is discussed in [64, 66–68] and for optimal control in [65]. In the line of these studies, NCSs with energy harvesting sensors are considered in [69, 70]. Energy harvesting sensors can obtain the power from their environments such as solar power, thermal energy, and vibrations. Thus, the energy harvesting sensors are expected to be a solution for NCSs with non-external energy sources for the sensors. Another variation appears recently, for example, [71] studies the event-based control with a state-dependent Riccati equation. Lately, learning techniques are also applied to NCSs, see [72, 73].

2.2.2 Deterministic NCSs

Also, deterministic NCSs are well studied [9, 24, 74]. The studies appeared at first in the line of sampling problems with digital controllers [75] since the communication between controllers and field devises are performed intermittently as discrete events while the physical plants evolve continuously. Thus, the whole system is modeled as hybrid systems [76] or sampled-data systems [77]. In [78], the stability conditions for the NCSs under communication constraints and the authors propose the try-once-discard (TOD) protocol as a way to decide the communication schedule. For the TOD protocol, network-induced delays are also included in [79]. Furthermore, round-robin (RR) protocol is introduced in [80, 81] where a sampled-data system approach is taken in [80] while a hybrid system approach is used in [81]. The authors of [74] provide a general description of NCSs with network-induced phenomena such as delays, dropouts, and variable sampling, and discuss the trade-off between sampling interval and performance of NCSs under such various communication constraints.

In the context of the NCSs, event-based control has received a lot of attention from many researchers as a measure to reduce the communication load in networks. Event-based control was first studied in [82, 83] where the advantages of event-based PID control are validated through simulations. Various event-based problem setups are discussed in the survey [84]. For example, the event-based control is used in multi-agent systems in [85], and is introduced in adaptive control in [86].

Event-based control for state feedback and output feedback systems is studied in [87–89]. Some variations appear recently. For example, time-delayed systems are considered in [90, 91]. Event-based output feedback control and PID control are also well studied [92, 93]. The stability conditions for event-based PI control of first-order systems are derived in [94] and for general output feedback with actuator saturation [93]. Experimental validation is performed in [93, 95]. Implementation to a real industrial plant is presented in [96–98].

Chapter 3

Sensor Network Scheduling for Remote Estimation

This chapter studies a design problem of how a group of wireless sensors are selected and scheduled to transmit data efficiently over a multi-hop network subject to energy considerations, when the sensors are observing multiple independent discrete-time linear systems. Each time instant, some sensors are selected to transmit their measurements towards a remote estimator. We formulate an optimization problem, minimizing a linear combination of the averaged estimation error and the averaged transmission energy consumption to obtain suitable network scheduling and estimation algorithms. Necessary conditions for optimality are derived. It is shown that the optimal solution can be obtained by transforming the optimization problem into a finite-state MDP problem. We show that there exists a periodic network schedule and how it can be computed by numerically solving the MDP problem. Some effective algorithms to obtain suboptimal schedules are proposed to reduce the computational complexity of the original optimization problem. Numerical studies illustrate the effectiveness of the proposed algorithms.

3.1 Introduction

3.1.1 Literature review

In this chapter, we address multi-hop sensor network scheduling for remote estimation. Wireless sensor networks provide advantages through enhanced and massive sensing, flexible deployment and operation, and more efficient maintenance. However, since wireless sensors have usually no inexhaustible or reliable energy sources, energy limitation of wireless sensors affect system performance. In this context, energy-aware protocols, real-time algorithms as well as empirical studies for optimizing the performance of wireless sensor networks have been discussed in [99–101]. In addition to the energy limitation, data packets in the network may be lost due to interference or network congestion, which leads to poor estimation and control performance of the overall networked systems.

To tackle these problems, sensor scheduling approaches for remote estimation have

been investigated by several research groups in the context of NCSs. A scheduling scheme for the Kalman filter is proposed in [45]. A stochastic sensor scheduling among multiple sensors is proposed in [46]. The authors of [102] derive conditions of a cost function to be submodular so that estimation performance can be guaranteed. In addition, schedules designed by greedy algorithms are studied in [103, 104]. Optimal estimation with a fnite-time horizon cost is introduced in [47] where the authors obtain a suboptimal schedule by formulating a relaxed convex problem. It is then extended to an infinite horizon case in [105, 106]. In [49], collision-free TDMA-like sensor scheduling scheme is proposed, where sensors are connected directly to a gateway, and the gateway transmits one sensor measurement from a single sensor at each time instance. The work [107] considers a similar set-up, but discusses when the sensor data transmission of each sensor need different time durations due to bandwidth capacity. Remote estimation with variance-based triggering is proposed in [55], where sensors directly communicate with the remote estimator through a common bus. Periodic transmission schedules are then obtained.

All research above, however, considers single-hop networks, i.e., each sensor directly communicates with a gateway or an estimator, even though multi-hop networks are widely considered in industrial wireless communication protocols such as WirelessHART [2], ISA-100 [3], and Zigbee [108]. The authors of [23] study the stability of control systems over WirelessHART, but the system includes only a single control loop in the network.

3.1.2 Contribution

In this chapter, we consider the problem of how to select and schedule a set of sensors to transmit their measurement efficiently over a multi-hop network subject to energy constraints, when the sensors observe independent discrete-time linear systems. We propose a design framework of sensor network schedule, i.e., which links are to be activated to transmit the sensor measurement for the optimal remote estimation under sensor energy constraints. In the proposed framework, some sensors are scheduled to transmit to the estimator at each time step. By not allowing all sensors to send data every time, the energy consumption of the sensors can be reduced. Different from [49] and related work, the measurements are not directly sent to the estimator but through some intermediate nodes and a gateway.

For the medium access and communication, we consider a periodic superframe structure common to many existing wireless sensor network protocols [22]. A superframe repeated every sampling interval is divided into timeslots. We assume only one point-to-point link is activated at a time. Then, by activating links in a certain order, the measurement data of selected sensor nodes can be efficiently conveyed to the estimator. The link activation is jointly determined with the sensor selection, by considering data aggregation techniques [109, 110], and constrained by the energy consumption of the sensor nodes.

The contributions of this work are as follows: i) for offline optimal sensor network scheduling for remote estimation under sensor energy constraints, we find necessary conditions for optimality, ii) by exploiting the necessary conditions, we show that there exists a periodic optimal sensor network schedule, iii) efficient algorithms to construct

suboptimal schedules are proposed.

3.1.3 Organization

The remainder of the chapter is organized as follows. Section 3.2 describes preliminaries. System description including process, communication, energy consumption models together with remote estimator is given in Section 3.3. The problem is formulated in Section 3.4. Section 3.5 presents the main result. Suboptimal schedules are obtained in Section 3.6. Numerical examples are provided in Section 3.7. Section 3.8 presents the summary.

3.2 Preliminaries

A directed graph is an ordered pair $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of ordered pairs of nodes. Denoted edges $(j, i) \in \mathcal{E}$, if there is a link from node j to node i. For an edge $e = (j, i) \in \mathcal{E}$, denote the node that (j, i) departs from as $v_{\text{out}}(e)$ and the one that e flows into as $v_{\text{in}}(e)$, i.e., $v_{\text{out}}(e) = j$ and $v_{\text{in}}(e) = i$. Let $\mathcal{N}_i^{\text{in}}$ and $\mathcal{N}_i^{\text{out}}$ denote the in- and out-neighbors of node i, respectively, i.e.,

$$\mathcal{N}_i^{\text{in}} = \{j \in \mathcal{V} \mid (j, i) \in \mathcal{E}\},$$
$$\mathcal{N}_i^{\text{out}} = \{j \in \mathcal{V} \mid (i, j) \in \mathcal{E}\}.$$

In a directed graph $\mathcal{G}$, a directed path from node i_1 to node i_l is a sequence of nodes $(i_1, \ldots, i_l)$ such that $(i_j, i_{j+1}) \in \mathcal{E}$ for $j = 1, \ldots, l-1$. A tree with a root $0 \in \mathcal{V}$ is a directed subgraph of $\mathcal{G}$ such that every node i, where $i \in \mathcal{V}/\{0\}$, has exactly one directed path from itself to node 0. A spanning tree is an tree that contains all the nodes of $\mathcal{G}$. If a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a tree, we define a partial order "$\succeq$" over $\mathcal{E}$ in the following way. For any $e, \tilde{e} \in \mathcal{E}$, we say $e \succeq \tilde{e}$ if there exists a direct path from $v_{\text{out}}(e)$ to $v_{\text{in}}(\tilde{e})$. It is straightforward to see that "$\succeq$" is reflexive, antisymmetric, and transitive, i.e., the following properties for $e, \tilde{e}, \bar{e} \in \mathcal{E}$ hold:

(i). Reflexivity: $e \succeq e$,

(ii). Antisymmetry: if $e \succeq \tilde{e}$ and $\tilde{e} \succeq e$, then $e = \tilde{e}$,

(iii). Transitivity: if $e \succeq \tilde{e}$ and $\tilde{e} \succeq \bar{e}$, then $e \succeq \bar{e}$.

This justifies that "$\succeq$" defines a partial order over $\mathcal{E}$.

3.3 System Description

A set of sensors, denoted $\mathcal{V}_s = \{1, 2, \ldots, N\}$, are distributed in an area, monitoring N decoupled discrete-time linear time-invariant (LTI) processes. The sensors are interconnected via a wireless network and they pass measurements through the network to the remote estimator via a gateway (Figure 3.1). We denote the gateway as node 0 and denote

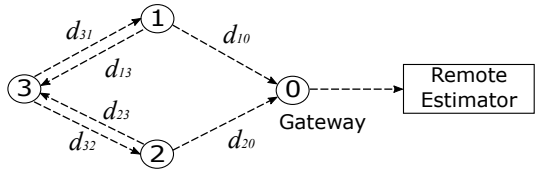

Figure 3.1: System architecture of a multi-hop wireless sensor network with 3 sensors. In the network, each sensor transmit its data to a remote estimator through intermediate sensor nodes and a gateway, where d_{10}, d_{20}, $d_{13} = d_{31}$, and $d_{23} = d_{32}$ are the distances between network nodes.

the whole node set including the gateway as $\mathcal{V} = \mathcal{V}_s \cup \{0\}$. The estimator generates state estimates based on the received information. We will elaborate on the main components of the system in the following subsections.

3.3.1 Process model

We consider N discrete-time LTI processes:

$$x_{k+1}^{(i)} = A_i x_k^{(i)} + w_k^{(i)}, \quad k = 0, 1, \ldots, \quad i \in \mathcal{V}_s \tag{3.1}$$

where $x_k^{(i)} \in \mathbb{R}^n$ is the state of process i at time k, $w_k^{(i)} \in \mathbb{R}^n$ is zero-mean independent and identically distributed (i.i.d.) random disturbance, which is assumed to be selected from a probability distribution with the density function (pdf) $\mu_{w^{(i)}}$ with $\mathbb{E}[w_k^{(i)}(w_k^{(i)})^\top] = W_i$ ($W_i > 0$). The initial state $x_0^{(i)}$, independent of $w_k^{(i)}, k \in \mathbb{N}$, is selected from a probability distribution with pdf $\mu_{x_0^{(i)}}$, with mean $\mathbb{E}[x_0^{(i)}]$ and covariance $\Sigma_0^{(i)}$. Without loss of generality, we assume $\mathbb{E}[x_0^{(i)}] = 0$, as nonzero-mean cases can be translated into zero-mean by the coordinate change $\tilde{x}_k^{(i)} = x_k^{(i)} - \mathbb{E}[x_0^{(i)}]$. We assume that the state $x_k^{(i)}$ can be observed directly by sensor i. The system parameters are all known to the sensors as well as the remote estimator. We assume the plants are unstable, i.e., $|\lambda_{\max}(A_i)| > 1, \ \forall i \in \mathcal{V}_s$.

3.3.2 Communication model

The sensors communicate to the estimator through intermediate sensors and a gateway which define the underlying communication network. The communication network is described by an underlying directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{E}$ is the set of all communication links.

We assume all sensors are perfectly time-synchronized and that the time horizon is partitioned into strips of identical sampling time intervals. Each time interval is divided into two phases: a sensing phase and a communication phase, where the former one is a time period for sensor i to obtain the precess state $x_k^{(i)}$ and the later one is a time period for message delivery (Figure 3.2). The communication phase between time k and $k+1$, which

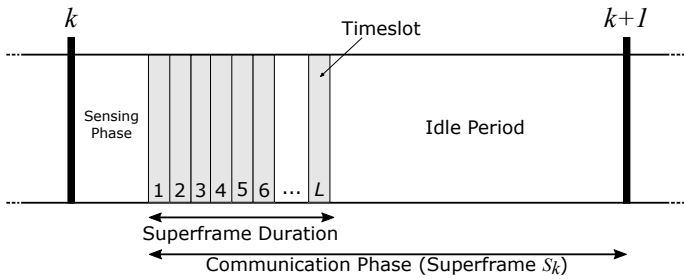

Figure 3.2: A superframe and timeslots

we also call superframe s_k, is divided into L timeslots k_l, where $l \in \mathcal{L} = \{1, 2, \ldots, L\}$. Superframe structures are practically used in industrial wireless communication protocols [2, 108], built upon the IEEE 802.15.4 MAC layer [4].

Let us denote $\mathcal{I}_{k_l}^{(i)}$ as the measurement data set that sensor i has before the transmission at timeslot k_l. Furthermore, let $\tilde{\mathcal{I}}_{k_l}(e_{k_l})$ be the measurement data set transmitted over link e_{k_l}. Obviously, $\tilde{\mathcal{I}}_{k_l}(e_{k_l}) \subseteq \mathcal{I}_{k_l}^{(i)}$ with $i = v_{\text{out}}(e_{k_l})$. The transmission during a single superframe over the network is performed in the following way:

1. At each timeslot, only one link $e \in \mathcal{E}$ can be activated, to avoid interference across the network, and the order of link activation $\{e_{k_1}, \ldots, e_{k_L}\}$ is pre-scheduled. If $e_{k_l} = (i, j)$, then $\tilde{\mathcal{I}}_{k_l}(e_{k_l})$ is transmitted without failure to sensor j through unicast-based communication protocol. Then $\mathcal{I}_{k_l}^{(i)}$ can be recursively written as

$$\mathcal{I}_{k_l}^{(i)} = \begin{cases} \mathcal{I}_{k-1_L}^{(i)} \cup \{x_k^{(i)}\}, & \text{if } l = 1, \\ \mathcal{I}_{k_{l-1}}^{(i)} \cup \tilde{\mathcal{I}}_{k_{l-1}}(e_{k_{l-1}}), & \text{if } l \geq 2 \text{ and } i = v_{\text{in}}(e_{k_{l-1}}), \\ \mathcal{I}_{k_{l-1}}^{(i)}, & \text{if } l \geq 2 \text{ and } i \neq v_{\text{in}}(e_{k_{l-1}}), \end{cases}$$

   with $\mathcal{I}_{0_1}^{(i)} = \{x_0^{(i)}\}$.

2. After L timeslots, the gateway transmits all the measurement $\mathcal{I}_k \triangleq \mathcal{I}_{k_L}^{(0)}$ to the estimator.

We assume that the maximum number of timeslots L is sufficiently large for accommodating all communication links in $\mathcal{G}$. We also make the realistic assumption that communication is much faster than sampling of the processes considered. Hence, the time delays due to communication within a superframe can be ignored. The problem we are interested in is to find a link allocation and the data transmission at each timeslot, i.e., to find a sequence $(e_{k_l}, \tilde{\mathcal{I}}_{k_l})_{l=1}^{L}$ for all superframes s_k, $k = 1, 2, \ldots$, which is efficient in terms of estimation performance and communication usage.

3.3.3 Energy consumption

The sensors consume a certain amount of energy when they receive data from and transmit data to the other sensors. Here we introduce an energy consumption model often used in wireless communication protocols [109–111]. The energy consumption for receiving a packet, which contains p-bits information is,

$$E_r(p) = E_{\text{elec}} p, \tag{3.2}$$

where the energy coefficient E_{elec} is determined by the electronics, coding etc. The energy consumption for sending p-bits information is

$$E_s(p, d) = E_{\text{elec}} p + E_{\text{amp}} d^2 p, \tag{3.3}$$

where E_{amp} is the energy coefficient for the amplifier and d is the distance to the receiving sensor or gateway. When transmitting multiple measurements, each sensor can join them in a single packet in order to reduce the transmission overhead. This technology is called packet aggregation. Assume that a single measurement from any sensor has c bits. Then the bits of information after aggregation is

$$p(q) = c \cdot [1 + (q - 1)(1 - r)], \tag{3.4}$$

where $q \in \mathbb{N}$ is the number of measurements and $r \in [0, 1]$ is the data aggregation rate [112]. If $r = 1$ the data is aggregated perfectly and the bits after aggregation is independent of the number of measurements. If $r = 0$, no packet aggregation is used to reduce the number of bits. Notice that it is difficult to aggregate collected data from different sensors perfectly, but some part of the data such as header can be removed when aggregating.

Let $q_{k_l} \triangleq |\tilde{\mathcal{I}}_{k_l}(e_{k_l})|$ be the number of measurements transmitted to the downstream sensor over link e_{k_l} and d_{k_l} be the distance of link e_{k_l} at timeslot k_l. Then the total energy consumption for sensor i in superframe s_k is

$$E_k^{(i)} = \sum_{l \in \mathcal{T}_{r,k}^{(i)}} E_r(p(q_{k_l})) + \sum_{l \in \mathcal{T}_{s,k}^{(i)}} E_s(p(q_{k_l}), d_{k_l})), \tag{3.5}$$

where $\mathcal{T}_{r,k}^{(i)} = \{l \in \mathcal{L} : v_{\text{in}}(e_{k_l}) = i\}$ and $\mathcal{T}_{s,k}^{(i)} = \{l \in \mathcal{L} : v_{\text{out}}(e_{k_l}) = i\}$.

3.3.4 Remote estimation

The optimal remote state estimate for process i, denoted $\hat{x}_k^{(i)}$, is computed as

$$\hat{x}_k^{(i)} = \begin{cases} x_k^{(i)}, & \text{if } x_k^{(i)} \in \mathcal{I}_k; \\ A_i \hat{x}_{k-1}^{(i)}, & \text{otherwise,} \end{cases} \tag{3.6}$$

with the initial state $\hat{x}_0^{(i)} = 0$ if $x_0^{(i)} \notin \mathcal{I}_0$. The error covariance of $x_k^{(i)}$ is denoted

$$P_k^{(i)} = \mathbb{E}[(x_k^{(i)} - \hat{x}_k^{(i)})(x_k^{(i)} - \hat{x}_k^{(i)})^\top | \mathcal{I}_k]. \tag{3.7}$$

It can be recursively updated as

$$P_k^{(i)} = \begin{cases} 0, & \text{if } x_k^{(i)} \in \mathcal{I}_k; \\ h_i(P_{k-1}^{(i)}), & \text{otherwise,} \end{cases} \tag{3.8}$$

where $h_i : \mathbb{S}_n^+ \to \mathbb{S}_n^+$ is an operator defined as $h_i(X) = A_i X A_i^\top + W_i$ with the initial error covariance $P_0^{(i)} = \Sigma_0^{(i)}$ if $x_0^{(i)} \notin \mathcal{I}_0$.

3.4 Problem Formulation

The aim of this chapter is to find multi-hop network schedules under sensor energy constraints which optimize estimation performance. In this chapter, we do not solve the hard-constrained problem directly, but a Lagrangian problem. Define a network schedule in superframe s_k as $\theta_k = (e_{k_l}, \tilde{\mathcal{I}}_{k_l})_{l=1}^L$ and $\Theta \triangleq (\theta_1, \ldots, \theta_i, \ldots)$. The problem of interest is to find an optimal network schedule that minimizes the trace of the error covariance subject to an averaged sensor energy constraint.

Problem 3.1.

$$\min_{\Theta} \quad \limsup_{T\to\infty} \frac{1}{T} \sum_{k=0}^{T-1} \sum_{i=1}^{N} \operatorname{tr}(P_k^{(i)}(\Theta)), \tag{3.9a}$$

$$\text{s.t.} \quad \limsup_{T\to\infty} \frac{1}{T} \sum_{k=0}^{T-1} E_k^{(i)}(\Theta) \le \alpha_i, \ i \in \mathcal{V}_s, \tag{3.9b}$$

where $\alpha_i > 0$ is the averaged sensor energy constraint for node i.

For Problem 3.1 to be well-posed, schedules Θ which lead to uniform boundedness of $P_k^{(i)}(\Theta)$ for all processes, i.e., $\sup_{k\in\mathbb{N}} \operatorname{tr}(P_k^{(i)}(\Theta)) < \infty$ for all $i \in \mathcal{V}_s$ have to exist. Now we make the following assumption.

Assumption 3.1. *The graph $\mathcal{G}$ contains a spanning tree with root being node 0.*

Lemma 3.2. *If graph $\mathcal{G}$ has a spanning tree with a unique root node 0, then there exists at least one schedule Θ such that $\sup_{k\in\mathbb{N}} \operatorname{tr}(P_k^{(i)}(\Theta)) < \infty$ for all $i \in \mathcal{V}_s$.*

Proof. Let $\mathcal{G}_{\text{st}} = (\mathcal{V}, \mathcal{E}_{\text{st}})$ be a spanning tree contained in $\mathcal{G}$ with node 0 being the unique root. Then each node $i \in \mathcal{V}_s$ has a unique directed path going from node i to 0. Next we shall complete the proof by constructing a simple schedule Θ to ensure the remote estimator is stable.

Denote the unique path from node j to 0 by $(j, i_1, \ldots, i_{m_j}, 0)$. Consider $\theta_k = (e_{k_l}, \tilde{\mathcal{I}}_{k_l})_{l=1}^L$ with

$$e_{k_l} = \begin{cases} (j, i_1), & \text{if } l = 1, \\ (i_{m-1}, i_m), & \text{if } l = 2, \ldots, m_j, \\ (i_{m_j}, 0), & \text{if } l = m_j + 1, \end{cases}$$

and $\tilde{\mathcal{I}}_{k_l}(e_{k_l}) = \{x_k^{(j)}\}$ for all $l \in \mathcal{L}$. Then we repeat $\theta_1, \ldots, \theta_N$ every N period, i.e., $\theta_{j+tN} = \theta_j$ for $t \in \mathbb{N}$ and $j \in \{1, \ldots, N\}$. By constructing Θ in this way, it yields

$$\lim_{T\to\infty} \frac{1}{T} \sum_{k=0}^{T-1} \mathrm{tr}(P_k^{(i)}(\Theta)) = \frac{1}{N} \sum_{l=0}^{N-1} \mathrm{tr}(h_i^l(0)) < \infty,$$

where the equality follows from (3.8) and the schedule period N. The inequality follows from the definition of $h_i(\cdot)$. This completes the proof. □

Remark 3.3. *By Lemma 3.2, as long as $\mathcal{G}$ contains a spanning tree, Problem 3.1 is always well-posed.*

To solve Problem 3.1, we use a Lagrangian technique similar to [64, 67], to derive the unconstrained problem:

Problem 3.2.

$$\min_{\Theta} \quad \limsup_{T\to\infty} \frac{1}{T} \sum_{k=0}^{T-1} \sum_{i=1}^{N} \mathrm{tr}(P_k^{(i)}(\Theta)) + \beta_i E_k^{(i)}(\Theta), \tag{3.10}$$

where $\beta_i > 0$ is the Lagrange multiplier. Problem 3.2 corresponds to jointly optimize a weighted average of estimation error and sensor energy consumption. Minimization of (3.10) with given values of β_i corresponds to an optimal schedule of Problem 3.1 with respect to given energy constraints α_i. We focus on Problem 3.2 in the rest of the chapter.

3.5 Optimal Sensor Network Schedule

In this section, we discuss how to solve Problem 3.2. In the first part, we discuss necessary conditions for optimality, to reduce candidate schedules to a certain class. In the second part, we transform the problem to a Markov Decision Process (MDP), and then we show that there exists an optimal periodic schedule. Finally, in the third part, we provide a procedure to search for an optimal schedule.

3.5.1 Necessary conditions for optimality

Let us define the joint graph for a superframe s_k under a sensor network schedule Θ in the following way. We denote by $\mathcal{E}_k$ the sequence of communication links in s_k sampled from the underlying graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, i.e., $\mathcal{E}_k \triangleq \{e_{k_1}, \ldots, e_{k_L}\} \subseteq \mathcal{E}$. Then we call $\mathcal{G}(s_k) = \{\mathcal{V}, \mathcal{E}_k\}$ the joint graph of s_k under network schedule Θ.

Let us denote the set of communication links and the joint graph under an optimal network schedule Θ^* as $\mathcal{E}_k^*$ and $\mathcal{G}^*(s_k)$, $k = 0, 1, \ldots$, respectively. Then the following property holds for $\mathcal{E}_k^*$ and $\mathcal{G}^*(s_k)$.

Lemma 3.4. *Assume that Problem 3.2 has an optimal solution. Then, for all $k \in \mathbb{N}$, $\mathcal{G}^*(s_k)$ is a tree with node 0 being the unique root.*

Proof. The proof has two steps:

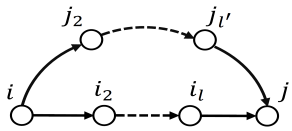

Figure 3.3: Illustration of two paths linking node i and j.

1. We first show that $\mathcal{G}^*(s_k)$ is a disjoint union of trees.

2. We show that $\mathcal{G}^*(s_k)$ has a unique root which is node 0.

From energy saving point of view, for each θ_k^* in Θ^*, $\mathcal{G}^*(s_k)$ is a disjoint union of trees, i.e., any two nodes are connected by at most one path. If not, without loss of generality, assume that there are two paths, denoted as $e = (i, i_1, \ldots, i_l, j)$ and $e' = (i, j_1, \ldots, j_{l'}, j)$, going from nodes i to j (Figure 3.3), and the lengths of e and e' are $l(e)$ and $l(e')$ respectively with $l(e) \leq l(e')$. Node i has q ($q \geq 1$) measurements to be transmitted to node j, among which q_j number of measurements are transmitted through node j_2 and q_i number of measurements through node i_2. Evidently, $q_i + q_j = q$. The number of bits node i transmits to node i_2 is

$$p(q_i) = c(1 - r)q_i + cr$$

and the number of bits node i transmits to node j_2 is

$$p(q_j) = c(1 - r)q_j + cr.$$

We consider another scheduling decision $\tilde{\theta}_k$, where θ_k^* and $\tilde{\theta}_k$ are the same except for that the q measurements of node i are all transmitted to node j through node i_2. We compare θ_k^* with $\tilde{\theta}_k$. In this case, the number of bits node i transmits to the downstream node i_2 is

$$p(q) = c(1 - r)q + cr.$$

Since $p(q_i) + p(q_j) > p(q)$ and $l(e) \leq l(e')$, to transmit the p measurements of node i to node j, θ_k^* consumes more energy compared to $\tilde{\theta}_k$. This contradicts the optimality of Θ^*, showing that $\mathcal{G}^*(s_k)$ is a disjoint union of trees.

Next we will show that $\mathcal{G}^*(s_k)$ has a unique root which is node 0. Notice that in (3.6), $\hat{x}_k^{(i)}$ is updated based on the latest information of process i that node 0 receives. Therefore, removing the trees not having node 0 as the root does not affect the calculation of $\hat{x}_k^{(i)}$, while energy for receiving and sending packets is saved. The proof is now complete. □

In virtue of Lemma 3.4, we have the following lemma.

Lemma 3.5. *Assume that Problem 3.2 has an optimal solution and denote* $\Theta^* \triangleq (\theta_1^*, \ldots, \theta_k^*, \ldots)$ *and* $\theta_k^* = (\tilde{\mathcal{I}}_{k_l}^*, e_{k_l}^*)_{l=1}^L$. *Then*

(*i*). $\tilde{\mathcal{I}}_{k_l}^*(e_{k_l}^*) = \{x_k^{(i)} : i = v_{\text{out}}(e_{k_j}^*), e_{k_j}^* \succeq e_{k_l}^*\}$ *for* $e_{k_l}^* \in \mathcal{E}_k^*$;

(*ii*). $k_i \le k_j$ *if* $e^*_{k_i} \ge e^*_{k_j}$ *for* $e^*_{k_i}, e^*_{k_j} \in \mathcal{E}^*_k$.

Proof. By Lemma 3.4, $\mathcal{G}^*(s_k)$ is a tree with node 0 being the unique root. Also notice that $\hat{x}^{(i)}_k$ in (3.6) is updated only based on the latest information about process i received by node 0. By letting each sensor i in $\mathcal{G}^*(s_k)$ send $x^{(i)}_k$ following upstream-node-first order, all measurements sampled and sent within s_k can reach node 0 free of delays. Otherwise, if either (*i*) or (*ii*) is violated, some measurements received by node 0 will be delayed. In this sense, any Θ violating either (*i*) or (*ii*) can never by optimal. □

By Lemmas 3.4 and 3.5, we only need to consider the cases that $\mathcal{G}(s_k)$ are trees and the measurements of selected sensors reach the estimator within one superframe. Define $\mathcal{S}_k \subseteq \mathcal{V}_s$ as the set of sensors selected to transmits their measurements at time k, and $z^{(i)}_k(e) \in \{0, 1\}$ as the index function if measurement of sensor i goes through link $e \in \mathcal{E}$ at time k. To satisfy that $\mathcal{G}(s_k)$ is a tree, $z^{(i)}_k(e)$ is constrained such that for every k:

$$\sum_{l\in\mathcal{N}^{\text{out}}_j} z^{(i)}_k((j,l)) - \sum_{l\in\mathcal{N}^{\text{in}}_j} z^{(i)}_k((l,j)) = 1, \text{ if } j = i,\ i \in \mathcal{S}_k, \tag{3.11a}$$

$$\sum_{l\in\mathcal{N}^{\text{out}}_j} z^{(i)}_k((j,l)) - \sum_{l\in\mathcal{N}^{\text{in}}_j} z^{(i)}_k((l,j)) = -1, \text{ if } j = 0,\ i \in \mathcal{S}_k, \tag{3.11b}$$

$$\sum_{l\in\mathcal{N}^{\text{out}}_j} z^{(i)}_k((j,l)) - \sum_{l\in\mathcal{N}^{\text{in}}_j} z^{(i)}_k((l,j)) = 0, \text{ if } i \ne j, 0,\ \ i \in \mathcal{S}_k, \tag{3.11c}$$

$$\sum_{l\in\mathcal{N}^{\text{out}}_j} z^{(i)}_k((j,l)) - \sum_{l\in\mathcal{N}^{\text{in}}_j} z^{(i)}_k((l,j)) = 0, \text{ if } i \notin \mathcal{S}_k. \tag{3.11d}$$

Furthermore, let

$$z^{(i)}_k = [z^{(i)}_k(e_1), \ldots, z^{(i)}_k(e_{|\mathcal{E}|})]^\top \in \{0, 1\}^{|\mathcal{E}|}$$

be the vectors of the index functions for every sensor and link aligned in an appropriate order. Then the constraints (3.11) can be written as

$$F z^{(i)}_k = b_i(\mathcal{S}_k), \quad i \in \mathcal{V}, \tag{3.12}$$

where F is the node-arc matrix of the underlying graph $\mathcal{G}$ and $b_i(\mathcal{S}_k) \in \mathbb{R}^{N+1}$ is the vector with elements take 1 if $i \in \mathcal{S}_k$ and the corresponding node is equal to i, take -1 if $i \in \mathcal{S}_k$ and the node is 0, and otherwise 0. Let also

$$z_k = [z^{(1)\top}_k, \ldots, z^{(N)\top}_k]^\top \in \{0, 1\}^{|\mathcal{E}|\cdot N}$$

be the augmented vector of $z^{(i)}_k$ for all $i \in \mathcal{V}_s$. Then in the following we search the optimal solution of the problem with variables $\Xi \triangleq (\xi_1, \ldots, \xi_k, \ldots)$ instead of Θ where $\xi_k = (\mathcal{S}_k, z_k)$ and with cost function

$$W(\Xi, T) = \sum_{k=0}^{T-1} \sum_{i=1}^{N} \Big[\text{tr}(P^{(i)}_k(\{\mathcal{S}_k\}^{T-1}_{k=0})) + \beta_i E^{(i)}_k(z_k)\Big]. \tag{3.13}$$

Problem 3.3.

$$\min_{\Xi} \quad \limsup_{T\to\infty} \frac{1}{T} W(\Xi, T) \tag{3.14a}$$

$$s.t. \quad F z_k^{(i)} = b_i(\mathcal{S}_k), \quad i \in \mathcal{V}, \quad k = 0, 1, \ldots \tag{3.14b}$$

Remark 3.6. *The first term of* (3.13) *does not depend on how sensor measurements are routed to the remote estimator. The second term of* (3.13) *depends only on the current path choice z_k. Note that constraints for z_k are conveniently represented by* (3.12)*.*

Remark 3.7. *Problem 3.3 is well-posed following a similar remark as for Problem 3.1. This can be seen by constructing a constant schedule, i.e., every sensor transmits its data at every time instant through a certain path. Then the obtained averaged cost in Problem 3.3 is bounded.*

Then we have the following lemma.

Lemma 3.8. *Define*

$$\tilde{E}(\mathcal{S}_k) \triangleq \min_{z_k} \sum_{e\in\mathcal{E}} \pi(e) \quad \text{s.t.} \quad F z_k^{(i)} = b_i(\mathcal{S}_k), \quad i \in \mathcal{V}, \tag{3.15}$$

where

$$\pi(e) \triangleq c\eta(e)\Big[(1-r)\sum_{i=1}^{N} z_k^{(i)}(e) + r \cdot \max_{i\in\mathcal{V}_s} z_k^{(i)}(e)\Big] \tag{3.16}$$

with

$$\eta(e) \triangleq \begin{cases} \beta_{v_{out}(e)}(E_{elec} + E_{amp} d^2(e)) + \beta_{v_{in}(e)} E_{elec}, & \text{if } v_{in}(e) \in \mathcal{V}_s; \\ \beta_{v_{out}(e)}(E_{elec} + E_{amp} d^2(e)), & \text{if } v_{in}(e) \in \{0\}. \end{cases} \tag{3.17}$$

Then the optimal solution to Problem 3.3 is obtained by $\mathcal{S} \triangleq (\mathcal{S}_0, \mathcal{S}_1, \ldots, \mathcal{S}_k, \ldots)$ which minimizes

$$J(\mathcal{S}) \triangleq \limsup_{T\to\infty} \frac{1}{T} \sum_{k=0}^{T-1} \Bigg[\sum_{i=1}^{N} \Big[\mathrm{tr}(P_k^{(i)}(\{\mathcal{S}_k\}_{k=0}^{T-1}))\Big] + \tilde{E}(\mathcal{S}_k)\Bigg]. \tag{3.18}$$

Proof. The cost function (3.13) can be written as

$$W(\Xi, T) = \sum_{k=0}^{T-1} \sum_{i=1}^{N} \mathrm{tr}(P_k^{(i)}(\{\mathcal{S}_k\}_{k=0}^{T-1})) + \sum_{k=0}^{T-1} \sum_{i=1}^{N} \beta_i E_k^{(i)}(z_k)$$

and the constraints (3.12) are determined by the current time $\mathcal{S}_k$. Thus, the second term in the above cost function can be minimized at each time instance. Let $p_k(e)$ be the amount of data transmitted over link $e \in \mathcal{E}$ at time instance k. By (3.2)–(3.4), if $q \geq 1$ we have

$$p_k(e) = c\Big[1 + \Big(\sum_{i=1}^{N} z_k^{(i)}(e) - 1\Big)(1-r)\Big]$$

$$= c\Big[(1-r)\sum_{l=1}^{N} z_k^{(i)}(e) + r\Big].$$

Considering the case $q = 0$, we obtain

$$p_k(e) = c\Big[(1-r)\sum_{l=1}^{N} z_k^{(i)}(e) + r\max_{i\in\mathcal{V}_s} z_k^{(i)}(e)\Big].$$

Thus, the energy cost of using link e can be described as

$$\begin{aligned}&\beta_{v_{\text{out}}(e)}E_s(p_k(e), d_e) + \beta_{v_{\text{in}}(e)}E_r(p_k(e))\\ &\quad= \Big[\beta_{v_{\text{out}}(e)}(E_{\text{elec}} + E_{\text{amp}}d_e^2) + \beta_{v_{\text{in}}(e)}E_{\text{elec}}\Big]p_k(e)\end{aligned}$$

for the link e such that $v_{\text{in}}(e) \in \mathcal{V}_s$, and as

$$\beta_{v_{\text{out}}(e)}E_s(p_k(e), d_e) = \beta_{v_{\text{out}}(e)}(E_{\text{elec}} + E_{\text{amp}}d_e^2)p_k(e)$$

for the link e such that $v_{\text{in}}(e) = 0$. Since the term $\sum_{i=1}^{N}\beta_i E_k^{(i)}(z_k)$ is a weighted sum of the energy consumption for all sensors, we have

$$\begin{aligned}\sum_{i=1}^{N}\beta_i E_k^{(i)}(z_k) &= \sum_{e:v_{\text{in}}(e)\in\mathcal{V}_s}\Big[\beta_{v_{\text{out}}(e)}(E_{\text{elec}} + E_{\text{amp}}d_e^2) + \beta_{v_{\text{in}}(e)}E_{\text{elec}}\Big]p_k(e)\\ &\quad+ \sum_{e:v_{\text{in}}(e)=0}\beta_{v_{\text{out}}(e)}(E_{\text{elec}} + E_{\text{amp}}d_e^2)p_k(e)\\ &= \sum_{e\in\mathcal{E}}\eta(e)p_k(e)\\ &= \sum_{e\in\mathcal{E}}\pi(e).\end{aligned}$$

Therefore, Problem 3.3 is transformed to the minimization problem of (3.18), which completes the proof. □

Remark 3.9. *By Lemma 3.8, the optimal schedule can be obtained by searching $\mathcal{S}$ instead of Ξ.*

Remark 3.10. *The energy consumption $\tilde{E}(\mathcal{S}_k)$ in* (3.15) *can be obtained by solving a binary integer programming problem.*

Inspired by [49], we introduce off-duty durations of the sensors over schedule $\mathcal{S}^*$ and show their boundedness. Let $\tilde{\tau}_m^{(i)}(\mathcal{S})$ be a time instance of sensor i's transmission m to the estimator. In addition, denote off-duty duration after transmission m for sensor i as

$$\nu_m^{(i)}(\mathcal{S}) = \tilde{\tau}_{m+1}^{(i)}(\mathcal{S}) - \tilde{\tau}_m^{(i)}(\mathcal{S}).$$

Considering time horizon $[0, T]$, let

$$\Gamma_i(\mathcal{S}, T) = \{\tilde{\tau}_1^{(i)}(\mathcal{S}), \ldots, \tilde{\tau}_{\sigma_i(T)}^{(i)}(\mathcal{S})\}$$

be the set of time instances of all transmissions for sensor i, where $\sigma_i(T)$ is the number of transmissions in $[0, T]$. Let denote the set of time instances with off-duty duration longer than δ_i as

$$\Delta_i(T) \triangleq \{\tilde{\tau}_m^{(i)}(\mathcal{S}^*) \in \Gamma_i(\mathcal{S}^*, T) : \nu_m^{(i)}(\mathcal{S}^*) > \delta_i\},$$

where

$$\delta_i \triangleq \min_{\rho}\{\, \rho \in \mathbb{N} : \mathrm{tr}(h_i^{\rho}(0)) > \tilde{E}(\{i\})\}.$$

Then we have the following theorem.

Theorem 3.11. *Assume there exists an optimal schedule $\mathcal{S}^*$. Then we have*

$$\lim_{T\to\infty} \frac{|\Delta_i(T)|}{T} = 0, \quad \forall i \in \mathcal{V}_s.$$

Proof. The proof is given by contradiction. Consider an optimal schedule $\mathcal{S}^*$ and assume that $\lim_{T\to\infty} |\Delta_i(T)|/T > 0$. Let $\tilde{\tau}_L^{(i)}, \tilde{\tau}_{L+1}^{(i)} \in \Delta_i(T)$ with $\nu_L^{(i)} = \tilde{\tau}_{L+1}^{(i)} - \tilde{\tau}_L^{(i)} > \delta_i$. By constructing a schedule $\mathcal{S}$ which is the same as $\mathcal{S}^*$ expect that $\tilde{\tau}_L^{(i)} + \delta_i \in \Gamma_i(\mathcal{S}, T)$. Then if $\gamma \triangleq \mathrm{tr}(h^{\delta_i}(0)) - \tilde{E}(\{i\})$, we have

$$\begin{aligned}
&W(\mathcal{S}^*, T) - W(\mathcal{S}, T) \\
&= \sum_{l=1}^{\nu_L^{(i)}} \left[\mathrm{tr}(h_i^{l+\delta_i-1}(0) - h^{l-1}(0))\right] + \tilde{E}(\mathcal{S}_{\tilde{\tau}_L^{(i)}+\delta_i}) - \tilde{E}(\mathcal{S}_{\tilde{\tau}_L^{(i)}+\delta_i} \cup \{i\}) \\
&> \mathrm{tr}(h_i^{\delta_i}(0)) + \tilde{E}(\mathcal{S}_{\tilde{\tau}_L^{(i)}+\delta_i}) - \tilde{E}(\mathcal{S}_{\tilde{\tau}_L^{(i)}+\delta_i} \cup \{i\}) \\
&= \gamma + \tilde{E}(\{i\}) + \tilde{E}(\mathcal{S}_{\tilde{\tau}_L^{(i)}+\delta_i}) - \tilde{E}(\mathcal{S}_{\tilde{\tau}_L^{(i)}+\delta_i} \cup \{i\}) \\
&\geq \gamma > 0.
\end{aligned}$$

This holds since $\tilde{E}(\mathcal{S} \cup \{i\}) \leq \tilde{E}(\mathcal{S}) + \tilde{E}(\{i\})$ for $\mathcal{S} \in 2^{\mathcal{V}_s}$. By repeating above procedure for all elements in $\Delta(T)$, we can construct a schedule $\mathcal{S}'$ which satisfies

$$W(\mathcal{S}^*, T) - W(\mathcal{S}', T) \geq \gamma |\Delta_i(T)|.$$

As $T \to \infty$,

$$\begin{aligned}
J(\mathcal{S}^*) - J(\mathcal{S}') &= \limsup \frac{W(\mathcal{S}^*, T) - W(\mathcal{S}', T)}{T} \\
&\geq \limsup \frac{\gamma |\Delta_i(T)|}{T} > 0,
\end{aligned}$$

which contradicts that $\mathcal{S}^*$ is optimal. This completes the proof. □

3.5.2 MDP formulation

Let us now assume that there exists an optimal schedule solving Problem 3.3. In this section, we formulate a finite-state deterministic MDP to derive an optimal schedule solving Problem 3.3. The cost in Problem 3.3 is determined by the error covariances which evolve along the schedule $\mathcal{S}$ from the initial state $\Sigma_0^{(i)}$. Since Theorem 3.11 guarantees that the optimal schedule has finite off-duty durations for all sensors, the error covariances of any sensor will be reset to 0 regardless of their initial states. Thus, we use time durations between current time and the last update time of each sensor as the state of the MDP.

Define the MDP $\mathcal{M} \triangleq (\mathcal{Q}, \mathcal{A}_\tau, f(\cdot,\cdot), C(\cdot,\cdot))$, where

- the state space $\mathcal{Q} = \{\tau \in \mathbb{N}^N : \tau[i] = 1, \ldots, \delta_i,\ i \in \mathcal{V}_s\}$ represents the time duration between current time and the last instance when sensor i transmits the measurement; i.e., the MDP state at time k is given by an N-dimensional integer vector with elements $\tau_k[i] = k - \max_t\{t \le k : i \in \mathcal{S}_t\} + 1$.

- the action space $\mathcal{A}_\tau = 2^{\mathcal{V}_s} \backslash \bar{\mathcal{A}}_\tau$ where $\bar{\mathcal{A}}_\tau = \{\mathcal{S} \in 2^{\mathcal{V}_s} : i \in \mathcal{S},\ \tau[i] = \delta_i\}$ represents the set of sensors to transmit.

- the deterministic transition function from state τ with action $a \in \mathcal{A}_\tau$ to τ' is defined as
$$f(\tau, a) = \tau'$$
where $\tau'[i] = 1$ if $i \in a$ or $\tau'[i] = \delta_i$, otherwise $\tau'[i] = \tau[i] + 1$.

$$C(\tau, a) \triangleq \sum_{i=1}^{N} \mathrm{tr}(h_i^{\tau[i]-1}(0)) + \tilde{E}(a).$$

With this set-up, we formulate the MDP problem to find a policy $\mu \triangleq \{\mu_k\}_{k=1}^{\infty}$, $\mu_k : \mathcal{Q} \to \mathcal{A}$, which minimizes the averaged cost,

$$g_\mu(\tau_0) = \lim_{T\to\infty} \frac{1}{T} \sum_{k=0}^{T-1} C(\tau, a), \tag{3.19}$$

where τ_0 is the initial state.

Lemma 3.12. *For the finite-state MDP $\mathcal{M}$, there exists a stationary optimal policy μ^* with constant averaged cost $g^* \triangleq g_{\mu^*}$ satisfying the Bellman equation*

$$g^*(\tau) + H(\tau) = \min_{a\in\mathcal{A}} \Big\{C(\tau, a) + H(f(\tau, a))\Big\} \tag{3.20}$$

where $H(\cdot)$ is the relative value function [113].

Proof. It follows from Theorems 9.1.4 and 9.1.7 in [113]. □

Remark 3.13. *Theorem 3.11 indicates that the error covariance of sensor i will reset to* 0 *in* (3.8) *within a finite time for all $i \in \mathcal{V}_s$. Once it is reset, the error covariance of sensor i takes only $0, h_i(0), \ldots, h_i^{\delta_i - 1}(0)$ as well as the immediate cost of the MDP. Since optimal cost g^* is constant on the initial states τ_0 and the sum of the costs until the first reset can be ignored as T goes infinity, we have $g^* = J(\mathcal{S}^*)$ for any initial error covariance $\Sigma_0^{(i)}$ for $i \in \mathcal{V}_s$.*

Remark 3.14. *Function $H(\cdot)$ can be interpreted as a relative cost for each state $\tau \in \mathcal{Q}$ in an optimal solution [114].*

Now we are ready to state our main result which shows that the optimal sensor schedule is periodic.

Theorem 3.15. *There exists an optimal periodic policy μ^* that minimizes* (3.19)*.*

Proof. By Lemma 3.12, there exists an optimal stationary policy μ^*. Since the MDP $\mathcal{M}$ is deterministic, we can fix arbitrary one action as an optimal one at any state in $\mathcal{Q}$. Furthermore, since $\mathcal{Q}$ is finite, there exists a recurrent state over μ^*. Thus, if the system reaches the recurrent state again, the state transition will repeat. Hence the results follows. □

Remark 3.16. *The optimal periodic policy μ^* corresponds to an optimal periodic schedule $\mathcal{S}_k^* = \mathcal{S}_{k+d}^*$ for some positive integer d. This follows from how the MDP $\mathcal{M}$ was derived.*

3.5.3 Construction of optimal schedule

In the previous discussion, we showed that the optimal schedule is periodic and can be obtained by solving a finite-state MDP. Thus, some well-known algorithms can be applied. We present a two-step algorithm based on relative value iteration [113].

1. Calculate an optimal path for each sensor selection by Algorithm 3.1.
2. Calculate an optimal policy by Algorithm 3.2.

Algorithm 3.1 Computation of the optimal path and energy cost

1: **INPUT:** $\eta(e)$, r
2: **OUTPUT:** $\tilde{E}(a)$
3: **for** $a \in \mathcal{A}$ **do**
4: Compute $\tilde{E}(a)$
5: **end for**

Remark 3.17. *In general, it is not guaranteed that Algorithm 3.2 stops within finite iterations of k since $\mathcal{M}$ is a periodic MDP. However, we can apply the aperiodic transformation [113] to $\mathcal{M}$ in order to guarantee the convergence of the relative value iteration. In this way, it is possible to always obtain the optimal periodic schedule.*

Algorithm 3.2 Computation of the optimal schedule

1: **INPUT:** $\mathcal{M}$, $\tilde{E}(a)$, v^0, $\epsilon > 0$, $\bar{\tau}$
2: **OUTPUT:** $\mu^*(\tau)$
3: $u^0 = v^0 - v^0(\bar{\tau}) \cdot \mathbf{1}_{|\mathcal{Q}|}$
4: **for** $\tau \in \mathcal{Q}$ **do**
5: Compute

$$v^k(\tau) = \min_{a \in \mathcal{A}} \left\{ C(\tau, a) + v^{k-1}(f(\tau, a)) \right\} \tag{3.21}$$

6: **end for**
7: $u^k = v^k - v^k(\bar{\tau}) \cdot \mathbf{1}_{|\mathcal{Q}|}$
8: **if** $\max(v^{k+1}(\tau) - v^k(\tau)) - \min(v^{k+1}(\tau) - v^k(\tau)) \leq \epsilon$ **then**
9: Go to Step 13
10: **else**
11: $k \leftarrow k + 1$ and return to Step 4
12: **end if**
13: For each $\tau \in \mathcal{Q}$, set

$$\mu^*(\tau) = \arg\min_{a \in \mathcal{A}} \left\{ C(\tau, a) + v^{k-1}(f(\tau, a)) \right\}$$

It is difficult to find an optimal solution in general because of the computational complexity. The number of states of $\mathcal{M}$ depends on δ_i. In addition, since we allow to pick any sensor at every time instance, the size of action space is 2^N at every iteration in Algorithm 3.2. This fact motivates us to provide some alternative ways to construct suboptimal schedules in the next section.

3.6 Construction of Suboptimal Schedule

Finding an optimal solution is in general difficult due to its computational complexity. In this section, we introduce two algorithms to construct suboptimal schedules in an efficient way.

3.6.1 Reduced MDP algorithm

The first algorithm solves the MDP $\mathcal{M}$ by restricting the size of the state and action spaces. To do this, we introduce some sets of sensors and assume that the sensors in the same set are always scheduled to transmit together. The reduced MDP (R-MDP), $\tilde{\mathcal{M}} = (\tilde{\mathcal{Q}}, \mathcal{A}_\tau, f(\cdot,\cdot), \tilde{C}(\cdot,\cdot))$ is obtained as follows:

- Split $\mathcal{V}_s$ into M disjoint subsets $\tilde{\mathcal{V}} \triangleq \{\mathcal{V}_1, \ldots, \mathcal{V}_M\}$.
- Define the bounds $\delta_i = \min\{\delta_i : i \in \mathcal{V}_i\}$, $i = 1, \ldots, M$.

- Define the state space as

$$\tilde{Q} = \{\tau \in \mathbb{N}^M : \tau[i] = 1, \ldots, \delta_i, i = 1, \ldots, M\}$$

  and the action space $\mathcal{A}_\tau = 2^{\tilde{\mathcal{V}}} \backslash \bar{\mathcal{A}}_\tau$.

- Define the immediate cost

$$\tilde{C}(\tau, a) = \sum_{i=1}^{N} \mathrm{tr}(h_i^{\tau[j(i)]-1}(0)) + \tilde{E}(a),$$

  where $j(i)$ indicates the subset in $\tilde{\mathcal{V}}$ to which sensor i belongs.

We then obtain the schedule by calling Algorithm 3.1 and 3.2 for $\tilde{\mathcal{M}}$.

3.6.2 Fixed period algorithm

The idea of the second algorithm is to fix the transmission period of each sensor by solving the optimization problem without data aggregation. Then the whole schedule is obtained by combining each sensor schedule. The procedure is given by the fixed period algorithm (FPA) in Algorithm 3.3.

Algorithm 3.3 Fixed Period Algorithm

1: **INPUT:** $\eta(e)$
2: **OUTPUT:** $\{\mathcal{S}_{\mathrm{FPA},k}\}_{k=0}^{D}$
3: **for** $i \in \mathcal{V}_s$ **do**
4: Compute $\tilde{E}_i \triangleq \tilde{E}(\{i\})$
5: Set $\mathcal{M}_i = (\mathcal{Q}_i, \mathcal{A}_i, f(\cdot,\cdot), C(\cdot,\cdot))$ with $\mathcal{Q}_i = \{\tau_i \in \mathbb{N} : \tau_i = 1, \ldots, \delta_i\}$ and $\mathcal{A}_i = \{\emptyset, i\}$
6: Solve $\mathcal{M}_i$ and compute a period D_i
7: Set

$$\mathcal{S}_{\mathrm{FPA},k}^{(i)} = \begin{cases} i, & \text{if } k \equiv 0 \mod D_i \\ \emptyset, & \text{if } k \not\equiv 0 \mod D_i \end{cases}$$

8: **end for**
9: Compute D which is the least common multiple of D_i, $i = 1, \ldots, N$
10: **for** $k = 0, 1, \ldots, D$ **do**
11: Set $\mathcal{S}_{\mathrm{FPA},k} = \bigcup_{i \in \mathcal{V}_s} \mathcal{S}_{\mathrm{FPA},k}^{(i)}$
12: Compute $\tilde{E}(\mathcal{S}_{\mathrm{FPA},k})$
13: **end for**

For this algorithm, we have the following result.

Proposition 3.18. *The schedule obtained by Algorithm 3.3, $\Xi_{FPA} \triangleq \{(\mathcal{S}_{FPA,k}, z_{FPA,k})\}_{k=0}^{\infty}$ is optimal if $r = 0$, and the optimal value is upper bounded by the non-aggregated optimal value, i.e., $J(\Xi_{FPA}) \leq J(\Xi^*|_{r=0})$.*

Proof. Suppose that $r = 0$. Then we have $\pi(e) = \sum_{i=1}^{N} \eta(e) z_k^{(i)}(e)$ in (3.16), which means that the energy consumption $\tilde{E}(\mathcal{S}_k)$ is a linear combination of $z_k^{(i)}(e)$. That is, we have

$$\tilde{E}(\mathcal{S}_{\mathrm{FPA},k}) = \sum_{i=1}^{N} \sigma_i(\mathcal{S}_{\mathrm{FPA},k}) \tilde{E}_i,$$

where $\sigma_i(\mathcal{S}_k) = 1$ if $i \in \mathcal{S}_k$, otherwise 0. The cost function (3.18) is then

$$\begin{aligned}
&\lim_{T\to\infty} \frac{1}{T} \sum_{k=0}^{T-1} \left[\sum_{i=1}^{N} \left[\mathrm{tr}(P_k^{(i)}(\{\mathcal{S}_{\mathrm{FPA},k}\}_{k=0}^{T-1})) \right] + \tilde{E}(\mathcal{S}_{\mathrm{FPA},k}) \right] \\
&= \lim_{T\to\infty} \frac{1}{T} \sum_{k=0}^{T-1} \left[\sum_{i=1}^{N} \left[\mathrm{tr}(P_k^{(i)}(\{\mathcal{S}_{\mathrm{FPA},k}\}_{k=0}^{T-1})) + \sigma_i(\mathcal{S}_{\mathrm{FPA},k}) \tilde{E}_i \right] \right] \\
&= \sum_{i=1}^{N} \left[\lim_{T\to\infty} \frac{1}{T} \sum_{k=0}^{T-1} \mathrm{tr}(P_k^{(i)}(\{\mathcal{S}_{\mathrm{FPA},k}^{(i)}\}_{k=0}^{T-1})) + \sigma_i(\mathcal{S}_{\mathrm{FPA},k}) \tilde{E}_i \right].
\end{aligned}$$

Thus, minimization of

$$\lim_{T\to\infty} \frac{1}{T} \sum_{k=0}^{T-1} \mathrm{tr}(P_k^{(i)}(\{\mathcal{S}_{\mathrm{FPA},k}^{(i)}\}_{k=0}^{T-1})) + \sigma_i(\mathcal{S}_{\mathrm{FPA},k}) \tilde{E}_i \Big]$$

for each $i = 1, \ldots, N$ yields the minimum cost of (3.18), i.e., the obtained schedule with $\mathcal{M}_i$ is optimal when $r = 0$.

Since data aggregation maintains or reduces the energy cost, we have $J(\Xi_{\mathrm{FPA}}) \leq J(\Xi^*|_{r=0})$. □

3.7 Numerical Example

In this section, we provide two numerical examples to illustrate our results in this chapter. In the first example, we evaluate the performance of the suboptimal algorithms by comparing the optimal solution with a small network described in Figure 3.1. In the second example, we provide a larger network, so that we see that the two suboptimal algorithms can obtain the schedules even if the size of the original MDP becomes large.

3.7.1 Small network

To see the performances of the proposed algorithms, we consider a small network shown in Figure 3.1. To be specific, the system parameters of the three plants are

$$A_1 = \begin{bmatrix} 1.0 & 1.2 \\ 0 & 1.4 \end{bmatrix}, \; A_2 = \begin{bmatrix} 1.0 & 0.8 \\ 0 & 1.1 \end{bmatrix}, \; A_3 = \begin{bmatrix} 3.5 & 1.0 \\ 0 & 2.1 \end{bmatrix}.$$

with $W_i = I_2$, for $i = 1, 2, 3$. For communication parameters, we assume that $E_{\mathrm{elec}} = E_{\mathrm{amp}} = 1$, $c = 4$, $\beta_i = 1$ for $i = 1, 2, 3$, $d_{10} = d_{20} = d_{13} = d_{23} = 1$, and $r = 0.5$. The action set consists of every possible subset of sensors selected to transmit accompanied by all possible routes among them as shown in Table 3.1.

Action index	Sensor selection	Path
0	∅	–
1	1	$1 \to 0$
2	1	$1 \to 3 \to 2 \to 0$
3	2	$2 \to 0$
4	2	$2 \to 3 \to 1 \to 0$
5	3	$3 \to 1 \to 0$
6	3	$3 \to 2 \to 0$
7	1,2	$1 \to 0, 2 \to 0$
8	1,2	$1 \to 3 \to 2 \to 0$
9	1,2	$2 \to 3 \to 1 \to 0$
10	2,3	$3 \to 2 \to 0$
11	2,3	$3 \to 1 \to 0, 2 \to 0$
12	2,3	$2 \to 3 \to 1 \to 0$
13	3,1	$3 \to 1 \to 0$
14	3,1	$3 \to 2 \to 0, 1 \to 0$
15	3,1	$1 \to 3 \to 2 \to 0$
16	1,2,3	$3 \to 1 \to 0, 2 \to 0$
17	1,2,3	$3 \to 2 \to 0, 1 \to 0$
18	1,2,3	$1 \to 3 \to 2 \to 0$
19	1,2,3	$2 \to 3 \to 1 \to 0$

Table 3.1: All possible sensor selection and their routes to the gateway for the small network shown in Figure 3.1.

Optimal schedule

First, we derive the optimal schedule. Algorithm 3.1 gives the optimal paths and their energy costs as shown in Table 3.2. By the value of $\tilde{E}(\{i\})$, $i = 1, 2, 3$, we obtain the bounds of the MDP state space as $\delta_1 = 5$, $\delta_2 = 7$, $\delta_3 = 3$ with Theorem 3.11. Then we can find the optimal schedule by Algorithm 3.2, which result is shown in Figure 3.4. The period of the optimal schedule is 6 in which actions 0, 1, 10, and 13 are taken. The periodic optimal schedule with the corresponding optimal paths is illustrated in Figure 3.5.

R-MDP schedule

Next, we formulate R-MDP by setting $\mathcal{V}_1 = \{1\}$ and $\mathcal{V}_2 = \{2, 3\}$. Then we have $\delta_1 = 5$ and $\delta_2 = 3$. The obtained schedule is shown in Figure 3.6. It has also period 6 with actions 0,

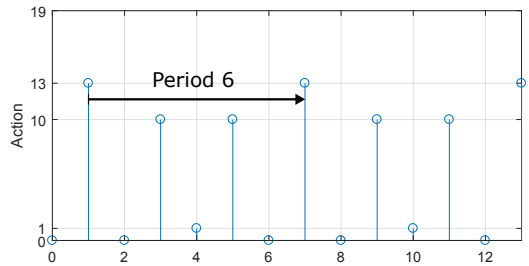

Figure 3.4: The optimal schedule obtained by Algorithm 3.2

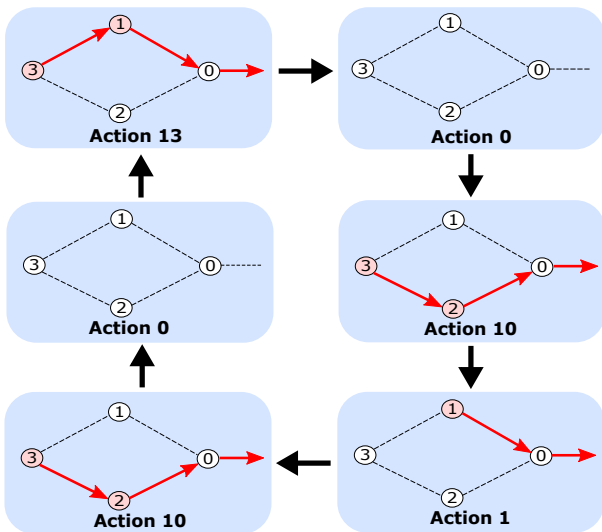

Figure 3.5: The period 6 optimal schedule for the small network.

Sensor selection	Optimal path	Energy cost
$\emptyset$	–	0
1	$1 \to 0$	8
2	$2 \to 0$	8
3	$3 \to 1 \to 0$	20
1,2	$1 \to 0, 2 \to 0$	16
2,3	$3 \to 2 \to 0$	24
1,3	$3 \to 1 \to 0$	24
1,2,3	$3 \to 1 \to 0, 2 \to 0$	32

Table 3.2: The optimal paths for each sensor selection

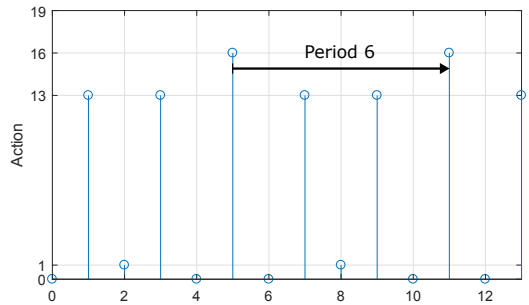

Figure 3.6: The R-MDP schedule for the small network obtained by Algorithm 3.2.

1, 13, and 16. We can see that sensors 2 and 3 are always selected together.

FPA schedule

We derive FPA schedule by Algorithm 3.3. Now we have $\tilde{E}_1 = \tilde{E}_2 = 8$, and $\tilde{E}_3 = 20$, with which we formulate MDP $\mathcal{M}_i$ for $i = 1, 2, 3$. Then we obtain the fixed activation period for each sensor: $D_1 = 2$, $D_2 = 4$, and $D_3 = 3$ which yields the period 12 schedule as shown in Figure 3.7.

Performance evaluation

The averaged cost and the sizes of the MDPs for each schedule are summarized in Table 3.3. We can see that the FPA schedule obtains similar performance compared to the optimal one. We also show the averaged cost of these two schedules with respect to the

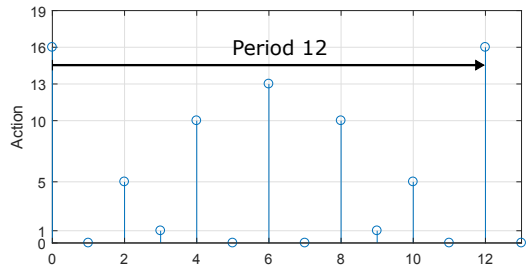

Figure 3.7: The FPA schedule for the small network obtained by Algorithm 3.3.

	Averaged cost	$\lvert\mathcal{Q}\rvert$	$\lvert\mathcal{A}\rvert$
Optimal	16.0768	105	8
R-MDP	17.0234	15	4
FPA	16.2702	$\leq$7	$\leq$2

Table 3.3: Averaged costs and the sizes of MDP

data aggregation rate r in Figure 3.8. It confirms Proposition 3.18, i.e., the averaged cost of the FPA schedule is optimal when $r = 0$ and this averaged cost is the upper bound of the FPA averaged cost for any r. The difference of the averaged costs of the two schedules becomes larger when r increases, since the optimal schedule enjoys much benefit of the data aggregation. In Figure 3.7, the FPA schedule takes action 5 twice in the period, i.e., sensor 3 is selected alone. However, this is not effective in terms of the energy cost since the data cannot be aggregated even though it passes through sensor 1. In the optimal schedule in Figure 3.4, sensor 3 is always selected together with sensor 1 (action 13) or sensor 2 (action 10). The R-MDP schedule obtains larger costs for any r. However, the cost is close to that of the FPA schedule if $r = 1$, since the R-MDP still tries to take advantage of the benefit of the data aggregation.

3.7.2 Larger network

To see the scalability of the proposed suboptimal scheduling algorithms, we consider the network shown in Figure 3.9. The network consists of 9 sensors distributed over a square field and a gateway at the origin. The sensors can communicate with the other sensors when the distances are shorter than $d_{\max} = 4$. The linear systems are given by

$$A_1 = \begin{bmatrix} 1.3 & 0.5 \\ 0.2 & 0.9 \end{bmatrix}, \; A_2 = \begin{bmatrix} 1.2 & 0 \\ 0 & 1.1 \end{bmatrix}, \; A_3 = \begin{bmatrix} 1.5 & 0.4 \\ 0.2 & 2 \end{bmatrix},$$

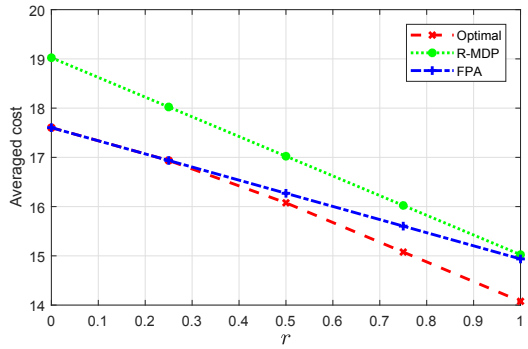

Figure 3.8: The averaged cost of the optimal, the R-MDP, and the FPA schedules with respect to r.

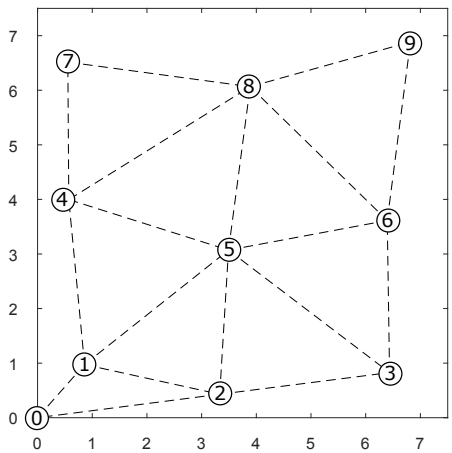

Figure 3.9: A large sensor network with $N = 9$.

	Averaged cost	Period	$\lvert\mathcal{Q}\rvert$	$\lvert\mathcal{A}\rvert$	CPU-time
R-MDP1	160.8498	20	1470	16	76.0193
R-MDP2	158.0748	12	3080	16	58.0781
FPA	161.5086	1260	$\leq$15	$\leq$2	0.0312

Table 3.4: Averaged costs, periods, sizes of MDP, and computation time

$$A_4 = \begin{bmatrix} 1.4 & 0.2 \\ 0.5 & 1.0 \end{bmatrix}, A_5 = \begin{bmatrix} 1.3 & 0.5 \\ 0.2 & 0.9 \end{bmatrix}, A_6 = \begin{bmatrix} 1.2 & 0 \\ 0 & 1.5 \end{bmatrix},$$
$$A_7 = \begin{bmatrix} 1 & 0.2 \\ 0.8 & 1.3 \end{bmatrix}, A_8 = \begin{bmatrix} 1.1 & 1.2 \\ 0 & 1 \end{bmatrix}, A_9 = \begin{bmatrix} 1.5 & 0.6 \\ 0.3 & 1.9 \end{bmatrix},$$

with $W_i = I_2$, for $i = 1, \ldots, 9$, $E_{\text{elec}} = E_{\text{amp}} = 1$, $c = 4, \beta_i = 1$ for $i = 1, \ldots, 9$, and $r = 0.5$. The bounds of the MDP states are obtained as $\delta_1 = 7$, $\delta_2 = 15$, $\delta_3 = 6$, $\delta_4 = 7$ $\delta_5 = 9$, $\delta_6 = 9$, $\delta_7 = 8$, $\delta_8 = 11$, and $\delta_9 = 5$. This means that the original MDP problem is no longer feasible to solve as the size of its state space is now of the order of $\prod_{i=1}^{N} \delta_i \sim 10^8$. The averaged cost and the sizes of MDPs for the three cases are summarized in Table 3.4. R-MDP1 uses R-MDP algorithm with grouping sensors based on their location in order to take advantage of the data aggregation. That is, we include sensors placed in the near distance into the same set. In the simulation, we take $\mathcal{V}_1 = \{1, 2\}, \mathcal{V}_2 = \{3, 5\}, \mathcal{V}_3 = \{4, 7\}$, and $\mathcal{V}_4 = \{6, 8, 9\}$. For R-MDP2, we make sensor sets based on the bound δ_i to avoid too many or too few transmissions with respect to the divergence speed of each error covariance. That is, sensors with close bounds are included in the same sets. In the simulation, we use $\mathcal{V}_1 = \{1, 4\}, \mathcal{V}_2 = \{2, 8\}, \mathcal{V}_3 = \{3, 9\}$, and $\mathcal{V}_4 = \{5, 6, 7\}$.

FPA schedule, on the other hand, can be obtained by repeating small MDPs $\mathcal{M}_i$, $i = 1, \ldots, 9$. The obtained FPA schedule generated by $D_1 = 4$, $D_2 = 9$, $D_3 = 4$, $D = 5$ $D_5 = 6$, $D_6 = 6$, $D_7 = 5$, $D_8 = 7$, and $D_9 = 4$ is with period 1260.

Figure 3.10 shows the averaged costs of the three schedules with respect to r. As in Proposition 3.18, the FPA schedule is the optimal when $r = 0$. Thus, it has a near optimal performance if r is small. The performance becomes worse compared to the R-MDPs when r is large. Both of approaches for the R-MDP schedules reduce cost when r is large. Furthermore, R-MDP2 has a comparatively better performance regardless of the value of r. It follows from how sensors are grouped influence the performance.

3.8 Summary

In this chapter, we proposed a co-design framework of multi-hop network scheduling for remote estimation. We formulated an optimization problem minimizing an infinite-time averaged estimation error covariance subject to sensor energy constraints. When the net-

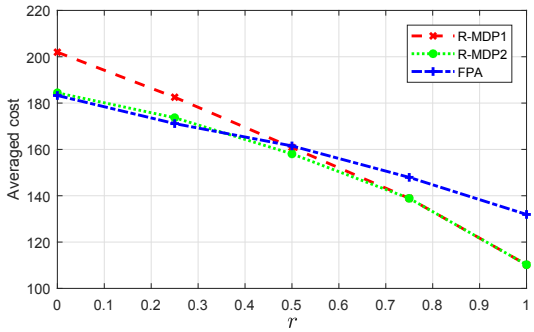

Figure 3.10: The averaged cost of the two reduced MDP and the FPA schedules with respect to r for the larger network.

work is sufficiently small, a periodic optimal solution can be found by exploiting necessary conditions for optimality. To reduce the computational complexity, we also proposed two alternative algorithms to obtain suboptimal schedules. It was also demonstrated how the proposed algorithms are effective in numerical examples of a small and a slightly larger networks.

Chapter 4

Co-design of Scheduling, Routing and LQG Controller

This chapter addresses a co-design framework of controllers, scheduling and routing of a wireless multi-hop sensor and actuator network. The network consists of sensors co-located with estimators and actuators with controllers. Sensors are observing multiple decoupled linear systems and transmit their estimated data to actuators in which controllers are co-located. We formulate an optimization problem, minimizing a linear combination of the averaged LQG control performance and the averaged transmission energy consumption. Optimal solutions are derived and their performance is illustrated in a numerical example. Algorithms to reconfigure routing between sensors and actuators in case of link disconnection are also provided.

4.1 Introduction

4.1.1 Literature review

In wireless process control systems, communication networks consist of wireless sensors, actuators, and controllers which are connected with each other. This chapter then focuses on control over a multi-hop wireless network. Especially, we consider that plants are controlled by LQG controllers. The problems of the networked LQG control are earlier investigated in [10, 27]. As stated in [10], an availability of acknowledgment from the controller plays a fundamental role. Event-based strategies have been taken to reduce communication and energy usage for state estimation and control over wireless networks [41, 115]. Therein, it is shown that controller gain obtained by standard methods and event-triggering law based on estimation error between the sensor and the controller gives an optimal event-based controller. The optimal estimation and control with multiple sensors are considered in [55, 57, 58]. In [55], the authors propose variance-based triggering for an optimal estimation problem which yields a periodic transmission schedule if the transmissions are always successful. For unreliable communication, variance-based triggering is given by the form of error covariance threshold policy [56, 57], where

communications are invoked when the error-covariance exceeds a certain value of the threshold. This framework is extended to LQG control in [58]. Therein, it is shown that the separation principle holds for the design of the event-based scheduling and the LQG controller. Under a related setup, [49, 116] consider sensor scheduling for remote state estimation. In [49], the authors study remote estimation of multiple linear systems where at most one sensor can communicate with the remote estimator at every time instance. In [116], a remote estimation problem over a multi-hop wireless network is discussed. However, despite the fact that a multi-hop wireless network architecture is accepted in some industrial standards such as wirelessHART [2, 19], it is still unclear how these event-triggering and sensor scheduling frameworks affect the performance of closed-loop systems. A co-design problem of control, scheduling, routing over a multi-hop network are proposed [117], which minimizes L_2 gain of error signals of the closed loop system with respect to a step reference. This work uses a mathematical framework defined in [118] to describe multi-hop control networks.

Wireless networks in industrial plants are imposed on a long-term channel variation [119, 120] due to moving objects in the area. It is investigated in an actual industrial environment of a steel rolling mill that the radio channel characteristics may change over a period of several hours [119]. In [62], the channel variation is model by a semi-Markov chain and network topology reconfiguration for state estimation is investigated. Motivated by this long-term channel variation in an industrial environment, we study network topology reconfiguration algorithms when a link in the multi-hop network is disconnected.

4.1.2 Contribution

In this chapter, we study LQG control of multiple discrete-time linear systems with covariance-based triggering over a multi-hop sensor and actuator network (Figure 4.1). Here, sensors and actuators are distributed over a field and can communicate with their neighborhoods. We assume that the sensors and the actuators are smart enough to carry out regular estimation and control. This has been discussed as a future architecture for process automation [5]. In this system, each sensor communicates with the corresponding controller co-located with the actuator. In Figure 4.1, there are four plants (red, yellow, blue, and green) which are controlled by each local control loop consisting of a sensor and an actuator. To derive the LQG control gains and sensor schedules, we formulate an optimization problem which shows that the optimal solution is periodic. This implies that one can automatically determine a sampling time of the system, which otherwise is usually chosen by a heuristic [75]. We also offer algorithms implemented in the sensors and the actuators which can detect a network link disconnection and reroute its path when other paths are available.

4.1.3 Organization

The remainder of the chapter is organized as follows. Section 4.2 describes the system including process and energy consumption models, and formulates the optimization prob-

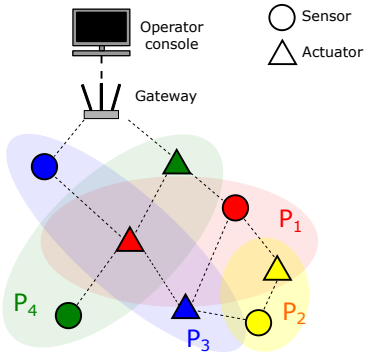

Figure 4.1: Process control over wireless sensor and actuator network. Four pairs of wireless sensors and actuators are controlling independent processes: P_1 (red), P_2 (yellow), P_3 (blue), and P_4 (green). Sensors and actuators are connected with neighborhood nodes and configures a multi-hop wireless sensor and actuator network. Operators can check the status of the plants by the information taken from the network.

lem. The optimal solution is discussed in Section 4.3. Algorithms for route reconfiguration are offered in Section 4.4. A numerical example is provided in Section 4.5. Section 4.6 presents the summary.

4.2 Problem Formulation

4.2.1 System model

A diagram of the system model is shown in Figure 4.2. Consider N linear plants

$$x_{k+1}^{(i)} = A_i x_k^{(i)} + B_i u_k^{(i)} + w_k^{(i)}, \quad i \in \mathcal{N} \tag{4.1}$$

where $x_k^{(i)} \in \mathbb{R}^{n_i}$ is the state vector at time k, $u_k^{(i)} \in \mathbb{R}^{m_i}$ is the input, $w_k^{(i)} \in \mathbb{R}^{n_i}$ is zero-mean i.i.d. Gaussian noise with covariance W_i, and $\mathcal{N} = \{1, \ldots, N\}$ is the plant index set. Each plant is monitored and controlled by a sensor–actuator pair $C_i = \{s_i, a_i\}$. The sensors have measurements

$$y_k^{(i)} = C_i x_k^{(i)} + z_k^{(i)}, \quad i \in \mathcal{N} \tag{4.2}$$

where $y_k^{(i)} \in \mathbb{R}^{p_i}$ is the output, and $z_k^{(i)} \in \mathbb{R}^{p_i}$ is zero-mean i.i.d. Gaussian noise with covariance V_i. The pairs of N sensors and actuators are distributed over a field and connected through an underlying communication network denoted $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \bigcup_{i=1}^{N} C_i$ is the sensor and actuator node set, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of communication links.

Define the information set available at sensor i at time k as

$$\mathcal{I}_{s,k}^{(i)} = \{y_0^{(i)}, \ldots, y_k^{(i)}, u_0^{(i)}, \ldots, u_{k-1}^{(i)}, v_0^{(i)}, \ldots, v_k^{(i)}\}$$

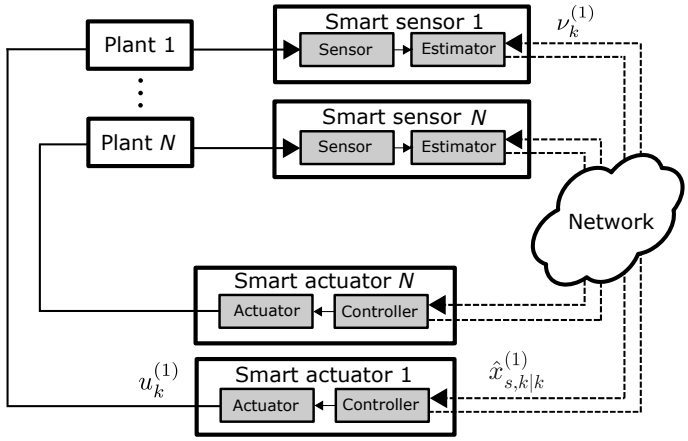

Figure 4.2: A block diagram of the system considered. A communication network is shared by independent N control loops. Smart sensors have a functionality of an estimator and smart actuators have that of a controller. Smart sensor i transmits its estimated value $\hat{x}_{s,k|k}^{(i)}$ to the corresponding smart actuator. The data update request (decision variable) $\nu_k^{(i)}$ is sent back from the smart actuator i to the smart sensor i.

where $\nu_k^{(i)} \in \{0, 1\}$ is decision variable such that $\nu_k^{(i)} = 1$ when the state estimate $\hat{x}_{s,k|k}^{(i)}$ is transmitted to actuator a_i. We assume that the transmission is carried out without failure until a link outage occur. Decisions for transmission are made by each actuator and fed back to the corresponding sensor. This enables the actuator to detect a link outage, which is discussed later. Note that actuators are not required to transmit their decision at every time instance since the transmission is perfect and then the sensors can emulate the controllers.

The state estimate and the corresponding error covariance at sensor i are given by

$$\begin{aligned}
\hat{x}_{s,k|k-1}^{(i)} &\triangleq \mathbb{E}[x_k^{(i)}|\mathcal{I}_{s,k-1}^{(i)}], \\
\hat{x}_{s,k|k}^{(i)} &\triangleq \mathbb{E}[x_k^{(i)}|\mathcal{I}_{s,k}^{(i)}], \\
P_{s,k|k-1}^{(i)} &\triangleq \mathbb{E}[(x_k^{(i)} - \hat{x}_{s,k|k-1}^{(i)})(x_k^{(i)} - \hat{x}_{s,k|k-1}^{(i)})^\top|\mathcal{I}_{s,k-1}^{(i)}], \\
P_{s,k|k}^{(i)} &\triangleq \mathbb{E}[(x_k^{(i)} - \hat{x}_{s,k|k}^{(i)})(x_k^{(i)} - \hat{x}_{s,k|k}^{(i)})^\top|\mathcal{I}_{s,k}^{(i)}].
\end{aligned}$$

In the same way, define the information set at actuator i at time k as

$$\mathcal{I}_{a,k}^{(i)} = \{\nu_0^{(i)}, \ldots, \nu_k^{(i)}, \nu_0^{(i)}\hat{x}_{s,0|0}^{(i)}, \ldots, \nu_k^{(i)}\hat{x}_{s,k|k}^{(i)}, u_0^{(i)}, \ldots, u_{k-1}^{(i)}, \}$$

and the state estimate and the error covariance

$$\begin{aligned}
\hat{x}_{a,k|k-1}^{(i)} &\triangleq \mathbb{E}[x_k^{(i)}|\mathcal{I}_{a,k-1}^{(i)}], \\
\hat{x}_{a,k|k}^{(i)} &\triangleq \mathbb{E}[x_k^{(i)}|\mathcal{I}_{a,k}^{(i)}],
\end{aligned}$$

$$P^{(i)}_{a,k|k-1} \triangleq \mathbb{E}[(x^{(i)}_k - \hat{x}^{(i)}_{a,k|k-1})(x^{(i)}_k - \hat{x}^{(i)}_{a,k|k-1})^\top | \mathcal{I}^{(i)}_{a,k-1}],$$
$$P^{(i)}_{a,k|k} \triangleq \mathbb{E}[(x^{(i)}_k - \hat{x}^{(i)}_{a,k|k})(x^{(i)}_k - \hat{x}^{(i)}_{a,k|k})^\top | \mathcal{I}^{(i)}_{a,k}].$$

4.2.2 Energy consumption

We introduce the energy consumption model used in [109, 111, 121]. For data receiving and sending, a node consumes its energy

$$E_R = E_{\text{elec}} f,$$
$$E_S = E_{\text{elec}} f + E_{\text{amp}} d^2 f,$$

respectively, where f [bit] is an amount of data receiving or sending and d is a distance to a downstream node. Note that the energy consumption for sending depends on the link used. Denote $\theta^{(i)}_k((j,l)) : \mathcal{E} \to \{0,1\}$ as the indicator function whether the data of sensor i is sent through link (j,l) at time k. If link (j,l) is used, then $\theta^{(i)}_k((j,l)) = 1$, otherwise 0. Then the energy consumption of node $j \in \mathcal{V}$ at time k is given by

$$E_{j,k} = \sum_{l:(l,j)\in\mathcal{E}} \left[E_{\text{elec}} \sum_{i\in\mathcal{N}} c_i \theta^{(i)}_k((j,l)) \right] + \sum_{l:(j,l)\in\mathcal{E}} \left[(E_{\text{elec}} + E_{\text{amp}} d^2_{jl}) \sum_{i\in\mathcal{N}} c_i \theta^{(i)}_k((j,l)) \right] \tag{4.3}$$

where c_i [bit] is a constant amount of data transmitted from sensor i to actuator i. It is reasonable to assume that data flow is conserved such that for all $i \in \mathcal{N}$ and $k \geq 0$:

$$\sum_{l:(j,l)\in\mathcal{E}} \theta^{(i)}_k((j,l)) - \sum_{l:(l,j)\in\mathcal{E}} \theta^{(i)}_k((j,l)) = 0, \text{ if } j \neq s_i, a_i, \tag{4.4a}$$
$$\sum_{l:(j,l)\in\mathcal{E}} \theta^{(i)}_k((j,l)) - \sum_{l:(l,j)\in\mathcal{E}} \theta^{(i)}_k((j,l)) = \nu^{(l)}_k, \text{ if } j = s_i, \tag{4.4b}$$
$$\sum_{l:(j,l)\in\mathcal{E}} \theta^{(i)}_k((j,l)) - \sum_{l:(l,j)\in\mathcal{E}} \theta^{(i)}_k((j,l)) = -\nu^{(l)}_k, \text{ if } j = a_i, \tag{4.4c}$$

in order to guarantee that sensor data can reach the corresponding actuator.

4.2.3 Optimization problem

We formulate an optimization problem as LQG control with network node energy consumption to find the optimal feedback control, scheduling, and routing. With a weight factor $\beta_i > 0$ and some vectors $\nu_k = [\nu^{(1)}_k, \ldots, \nu^{(N)}_k]^\top$, $u_k = [u^{(1)\top}_k, \ldots, u^{(N)\top}_k]^\top$, and $\theta_k = [\theta^{(1)\top}_k, \ldots, \theta^{(N)\top}_k]^\top$ with $\theta^{(i)}_k = [\ldots, \theta^{(i)}_k((j,l)), \ldots]^\top \in \{0,1\}^{|\mathcal{E}|}$, the problem is given by:

Problem 4.1.

$$\min_{\{\nu_k, u_k, \theta_k\}_{k=0}^{\infty}} \limsup_{T\to\infty} \frac{1}{T} \sum_{k=0}^{T-1} \left[\sum_{i=1}^{N} (x^{(i)\top}_k Q_i x^{(i)}_k + u^{(i)\top}_k R_i u^{(i)}_k) + \sum_{j\in\mathcal{V}} \beta_j E_{j,k} \right] \tag{4.5a}$$
$$\text{s.t.} \quad (4.4), \quad i \in \mathcal{N}, \quad k \geq 0. \tag{4.5b}$$

Note that there is no controller which can access all the variables $\{\nu_k\}_{k=0}^{\infty}$, $\{u_k\}_{k=0}^{\infty}$, and $\{\theta_k\}_{k=0}^{\infty}$, but we will show in the next section that the optimal solution can be found by distributed optimization at each controller without loss of performance.

4.3 Optimal Controller and Scheduler

In this section, we discuss the optimality of Problem 4.1. By equation (4.3), the last term of (4.5a) can be rewritten as

$$\begin{aligned}
\sum_{j\in\mathcal{V}}\beta_j E_{j,k} &= \sum_{i\in\mathcal{N}}\Bigg[\sum_{l:(l,j)\in\mathcal{E}}\Big[E_{\text{elec}}\sum_{i\in\mathcal{N}}c_i\theta_k^{(i)}((j,l))\Big]+\sum_{l:(j,l)\in\mathcal{E}}\Big[(E_{\text{elec}}+E_{\text{amp}}d_{jl}^2)\sum_{i\in\mathcal{N}}c_i\theta_k^{(i)}((j,l))\Big]\Bigg] \\
&= \sum_{i\in\mathcal{N}}\Bigg[\sum_{(j,l)\in\mathcal{E}}(\beta_j E_{\text{elec}}+\beta_l E_{\text{elec}}+\beta_j E_{\text{amp}}d_{jl}^2)c_i\theta_k^{(i)}((j,l))\Bigg] \\
&\triangleq \sum_{i\in\mathcal{N}}\Big[\sum_{(j,l)\in\mathcal{E}}\alpha_{jl}c_i\theta_k^{(i)}((j,l))\Big] \qquad (4.6)\\
&\triangleq \sum_{i\in\mathcal{N}}E_k^{(i)} \qquad (4.7)
\end{aligned}$$

where $E_k^{(i)}$ is a weighted total energy consumption for loop i transmission. Now, we have the following lemma.

Lemma 4.1. *The optimal solution to Problem 4.1 is obtained by solving the distributed optimization problem:*

$$\min_{\{\nu_k^{(i)},u_k^{(i)},\theta_k^{(i)}\}_{k=0}^{\infty}}\ \limsup_{T\to\infty}\frac{1}{T}\sum_{k=0}^{T-1}\Big[x_k^{(i)\top}Q_i x_k^{(i)}+u_k^{(i)\top}R_i u_k^{(i)}+\nu_k^{(i)}\tilde{E}_i\Big] \qquad (4.8)$$

where $\tilde{E}_i$ is the minimum-cost path for loop i when $\nu_k^{(i)}=1$, i.e., $\tilde{E}_i$ is the optimal value of the problem:

$$\tilde{E}_i \triangleq \min_{\theta_k^{(i)}}\ \pi^\top\theta_k^{(i)} \quad \text{s.t. (4.4)} \qquad (4.9)$$

where $\pi=[\ldots,\pi_{jl},\ldots]^\top\in\mathbb{R}^{|\mathcal{E}|}$ is given by $\pi_{jl}=\alpha_{jl}c_i$.

Proof. Using (4.7), the objective function (4.5a) is equivalent to the sum of the function

$$\limsup_{T\to\infty}\frac{1}{T}\sum_{k=0}^{T-1}\Big[x_k^{(i)\top}Q_i x_k^{(i)}+u_k^{(i)\top}R_i u_k^{(i)}+E_k^{(i)}\Big]$$

up to $i=1,\ldots,N$. Since $E_k^{(i)}$ is only a function of $\theta_k^{(i)}$, and $x_k^{(i)}$ and $u_k^{(i)}$ are not affected by $\theta_k^{(i)}$, we can take any $\theta_k^{(i)}$ provided that (4.4) is satisfied. Thus, the optimal value of $E_k^{(i)}$ when $\nu_k^{(i)}=1$ can be obtained by solving problem (4.9). □

By Lemma 4.1, distributed optimization can achieve the optimality of Problem 4.1.

Remark 4.2. *Problem* (4.9) *is the shortest path problem which can be solved by polynomial-time algorithms [122]. The transmission paths are pre-calculated before starting the operation.*

Remark 4.3. *Problem* (4.8) *is a special case in [58] where the energy consumption is determined by* (4.9) *and where there is no packet drop.*

To see the optimal solution of the distributed optimization problem (4.8), we state the following theorem.

Theorem 4.4. *There exists a stationary solution to* (4.8)*, and the solution* $\{u_k^{(i)*}\}_{k=0}^{\infty}$ *is given by*

$$u_k^{(i)*} = -(B_i^\top S_i B_i + R_i)^{-1} B_i^\top S_i A_i \hat{x}_{a,k|k}^{(i)} \triangleq L_i^{(i)} \hat{x}_{a,k|k}^{(i)} \tag{4.10}$$

with

$$\hat{x}_{a,k|k}^{(i)} = \begin{cases} A_i \hat{x}_{a,k-1|k-1}^{(i)} + B_i u_{k-1}^{(i)}, & \text{if} \quad \nu_k^{(i)} = 0 \\ \hat{x}_{s,k|k}^{(i)} & \text{if} \quad \nu_k^{(i)} = 1, \end{cases} \tag{4.11}$$

$$P_{a,k|k}^{(i)} = \begin{cases} A_i P_{a,k-1|k-1}^{(i)} A_i^\top + W_i, & \text{if} \quad \nu_k^{(i)} = 0 \\ \bar{P}_i, & \text{if} \quad \nu_k^{(i)} = 1, \end{cases} \tag{4.12}$$

where $S_i \in \mathbb{S}_{++}^n$ *is a solution of the Riccati equation*

$$S_i = A_i^\top S_i A_i + Q_i - A_i^\top S_i B_i (B_i^\top S_i B_i + R_i)^{-1} B_i^\top S_i A_i$$

and $\bar{P}_i \in \mathbb{S}_{++}^n$ *is a solution of the Riccati equation for the standard Kalman filter at sensor i. In addition, the stationary solution* $\{\nu_k^{(i)*}\}_{k=0}^{\infty}$ *is given by a threshold policy*

$$\nu_k^{(i)*} = \begin{cases} 0, & \text{if} \quad P_{a,k-1|k-1}^{(i)} < P_i^* \\ 1, & \text{otherwise}, \end{cases} \tag{4.13}$$

where $P_i^* \in \mathbb{S}_{++}^n$ *is the threshold matrix.*

Proof. Follows from the proof of Theorem 3 in [58]. This is a special case when $\gamma_k = 1$ in [58]. □

Remark 4.5. *The controller* (4.10) *is a certainty equivalence controller and is optimal thanks to side-information available according to the control architecture in Figure 4.2 [11].*

Remark 4.6. *The schedule of* $\nu_k^{(i)*}$ *converges to the periodic solution when* A_i *is unstable. This follows from the fact that there exists* $t_i \in \mathbb{N}$ *such that* $f_i^{t_i}(\bar{P}_i) = P_i^*$ *where* $f_i(X) \triangleq A_i X A_i^\top + W_i$ *which is calculated numerically. See [57]. As in [57, 58], the optimal cost of problem* (4.8) *is given by*

$$tr(S_i Q_i) + \frac{1}{t_i + 1}\Big[tr\big((A_i^\top S_i A_i + W_i - S_i) \sum_{j=0}^{t_i} f_i^j(\bar{P}_i)\big) + \tilde{E}_i\Big].$$

	Sensor	Actuator
Loop 1	(5.7521, 1.5978)	(0.4302, 2.6899)
Loop 2	(2.3478, 4.5316)	(8.4912, 8.3172)
Loop 3	(8.2119, 1.1540)	(6.4775, 5.5092)

Table 4.1: Sensor and actuator location

4.4 Link Disconnection and Route Reconfiguration

A controller recognizes a link disconnection when it fails to receive the new data from the sensor despite that $\nu_k^{(i)*} = 1$. In this case, the path is reconfigured by searching a new one. Let $\mathcal{P}_i = \{p_1^{(i)}, \ldots, p_j^{(i)}, \ldots, p_{M_i}^{(i)}\}$ be a set of possible paths from s_i to a_i where $p_j^{(i)} = ((s_i, \cdot), \ldots, (\cdot, a_i))$ is the j-th minimum-cost path. Furthermore, let $\mathcal{M}_i = \{P_1^{(i)*}, \ldots, P_{M_i}^{(i)*}\}$ be the set of threshold matrices induced by each path. The sets $\mathcal{P}_i$ and $\mathcal{M}_i$ are assumed to be pre-set in s_i and a_i. Algorithms 4.4 and 4.5 provide the implementation of the proposed controller, scheduling, and routing reconfiguration. In the algorithms, if the controller detects a link disconnection, it changes its path to the second best one. If no other paths are available, the control loop goes to fail safe mode.

Algorithm 4.4 Iterative algorithm for smart sensor i

Calculate $\hat{x}_{s,k|k}^{(i)}$
if $\nu_k^{(i)} = 1$ **then**
  Send $\hat{x}_{s,k|k}^{(i)}$ along path $p_j^{(i)}$
end if
if New p_{j+l} received **then**
  Calculate $P_{a,k|k}^{(i)} = f^l(P_{a,k-1|k-1}^{(i)})$
  Set a new path $p_{j+l}^{(i)}$ and a threshold $P_{j+l}^{(i)*}$
else
  Calculate $P_{a,k|k}^{(i)}$ by (4.12)
end if
Calculate $\nu_{k+1}^{(i)}$ by (4.13)
$k \leftarrow k + 1$

4.5 Numerical Example

To illustrate our results, we consider a small network with $N = 3$ where sensors and actuators are distributed over a square field shown in Table 4.1 and Figure 4.3. The system

Algorithm 4.5 Iterative algorithm for smart actuator i

if $\nu_k^{(i)} = 1$ **and** $\hat{x}_{s,k|k}^{(i)}$ not received **then**
  Calculate $\hat{x}_{a,k|k}^{(i)} = A_i \hat{x}_{a,k-1|k-1}^{(i)} + B_i u_{k-1}^{(i)}$
  Calculate $P_{a,k|k}^{(i)} = f(P_{a,k-1|k-1}^{(i)})$
  if $j = M_i$ **then**
    Go to fail safe mode
  else
    Set a new path $p_{j+1}^{(i)}$ a threshold $P_{j+1}^{(i)*}$
    Send new path $p_{j+1}^{(i)}$ along a backward route of $p_{j+1}^{(i)}$
  end if
else
  Calculate $\hat{x}_{a,k|k}^{(i)}$ by (4.11)
  Calculate $P_{a,k|k}^{(i)}$ by (4.12)
end if
Calculate $u_k^{(i)}$ by (4.10)
Calculate $\nu_{k+1}^{(i)}$ by (4.13)
$k \leftarrow k + 1$

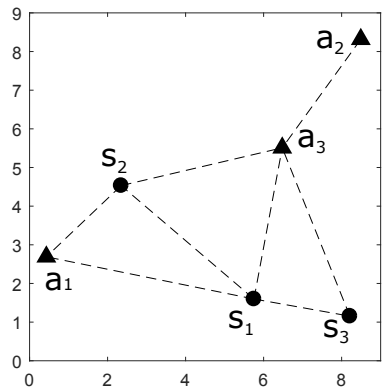

Figure 4.3: Network with three sensor and actuator pairs over a square field.

	Path	Energy cost $\tilde{E}_i$
Loop 1	$s_1 \to s_2 \to a_1$	1.000×10^4
Loop 2	$s_2 \to a_3 \to a_2$	1.086×10^4
Loop 3	$s_3 \to a_3$	0.767×10^4

Table 4.2: Optimal path and its energy cost of each control loop

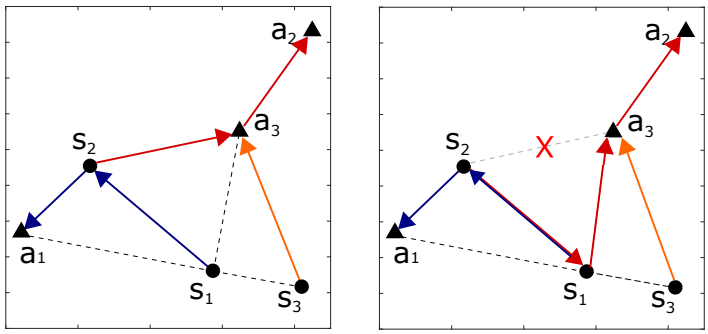

Figure 4.4: The minimum-cost paths for each loop before disconnection (left) and after disconnection (right). The path of loop 1: blue, loop 2: red, and loop 3: orange. The link between s_2 and a_3 is disconnected in the right network which results in the re-routing of loop 2.

parameters of the three plants are given by

$$A_1 = \begin{bmatrix} 1.3 & 0.5 \\ 0.2 & 0.9 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad C_1 = \begin{bmatrix} 1 & 1 \end{bmatrix},$$
$$A_2 = \begin{bmatrix} 1.2 & 0 \\ 0 & 1.4 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad C_2 = \begin{bmatrix} 1 & 1 \end{bmatrix},$$
$$A_3 = \begin{bmatrix} 1.3 & 1.2 \\ 0 & 1 \end{bmatrix}, \quad B_3 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad C_3 = \begin{bmatrix} 1 & 1 \end{bmatrix},$$

with $W_i = 0.01 I_2$, $V_i = 1$, $Q_i = I_2$, and $R_i = 1$ for all $i = 1, 2, 3$. For communication parameters, we assume that $c_i = 32$ and $\beta_i = 10$ for $i = 1, 2, 3$, and $E_{\text{elec}} = E_{\text{amp}} = 1$. Under the given network parameters, we can derive the minimum-cost path for each control loop as in Table 4.2 and Figure 4.4 (left). The optimal schedules are shown in Figure 4.5. We see that the solutions are periodic as stated in Remark 4.6. Sensor s_1 transmits its new estimate

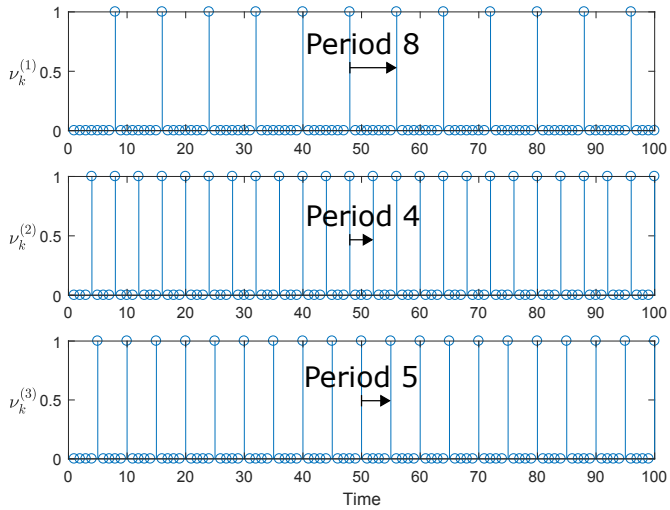

Figure 4.5: Optimal schedules of three loops. Loop 1 has a period 8 schedule, loop 2 has a period 4, and loop 3 has a period 5.

	Proposed method	**Every-time Communication**
Averaged cost	0.861×10^4	3.022×10^4

Table 4.3: Comparison of averaged cost between the proposed method and the every-time communication case

every eighth time instance, sensor s_2 every fourth time instance, and sensor s_3 every fifth time instance, respectively. The difference of the periods among the loops comes from the relation of the eigenvalues of A_i and the energy costs for transmission. The optimal averaged cost of the proposed method is shown in Table 4.3 compared with the case that all the sensors communicate with the actuators at every time instance, i.e.,

$$\sum_{i \in \mathcal{N}} \left[\mathrm{tr}(S_i Q_i + (A_i^\top S_i A_i + W_i - S_i)\bar{P}_i) + \tilde{E}_i \right].$$

We find that the proposed method obtains much lower cost than the every-time transmission case.

We also simulate the case that the link between s_2 and a_3 is disconnected at time $k = 300$ which leads to reroute the path between s_2 and a_2 to the second best path $s_2 \to s_1 \to a_3 \to a_1$ (Figure 4.4 (right)). The optimal schedule of loop 2 obtained by Algorithms 4.4

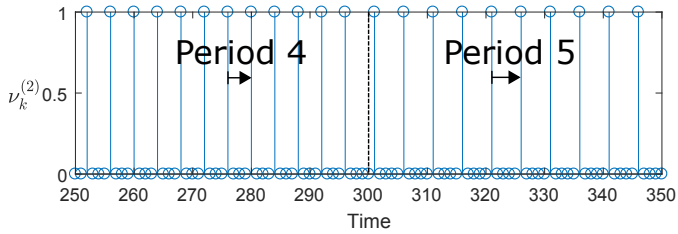

Figure 4.6: Optimal schedule of loop 2 around rerouting at $k = 300$. Disconnection leads to a longer period schedule with period 5.

and 4.5 is indicated in Figure 4.6. Since it leads to more energy consumption, the period of loop 2 becomes five which is longer than the period before the disconnection. The averaged energy consumptions of sensors and actuators from $k = 0$ to $k = T$, i.e.,

$$E_j^{\mathrm{ave}}(T) \triangleq \frac{1}{T+1} \sum_{k=0}^{T} E_{j,k}, \quad j \in \mathcal{V}$$

are shown in Figure 4.7. We found that the averaged energy consumption of s_1 increases after $k = 300$, since it is used as new intermediate node for loop 2. The energy consumption of s_2 and a_3 decreases since the period of loop 2 activation becomes longer.

4.6 Summary

In this chapter, we investigated a co-design framework of LQG control, sensor scheduling, and routing over a multi-hop sensor and actuator network by formulating an optimization problem which minimizes an infinite time averaged LQG control performance and energy consumption. We also proposed algorithms for sensors and actuators to let them configure a new path when a link is disconnected. A numerical example was provided to see how the scheduling and the routing are designed with the LQG controller. The route reconfiguration algorithms were also illustrated in the numerical example.

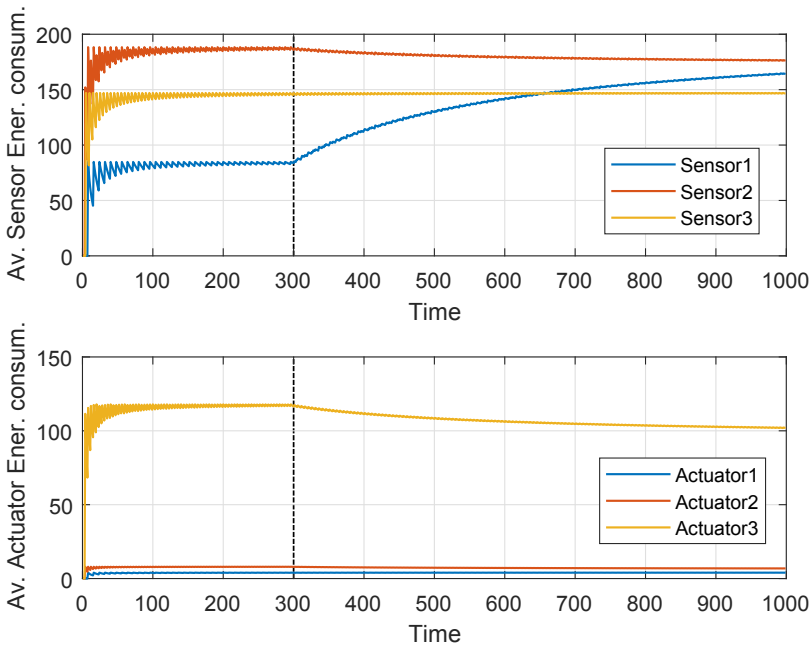

Figure 4.7: Averaged energy consumption for sensor and actuator communication. Averaged energy consumption of Sensor 1 increases after the disconnection at $k = 300$ while those of Sensor 2 and Actuator 3 decrease.

Chapter 5

Event-based Feedforward Control subject to Actuator Saturation

In this chapter, we address event-based feedforward and ratio control when combined with a feedback control loop. Stability conditions for these systems are derived when the systems are subject to actuator saturation. Numerical examples illustrate that event-based feedforward control significantly reduces communication between the disturbance sensor and the controller without performance degradation.

5.1 Introduction

5.1.1 Feedforward and ratio control

In process control systems, there are two common control architecture: feedforward control and ratio control. Compensating external disturbances is one of the most important roles of feedback control systems. The controller takes corrective action when the controlled variable deviates from its setpoint, for instance, due to an external disturbance. However, feedback disturbance compensation is unsuccessful when the time constant of the closed-loop system is too large. To improve the performance in this situation, feedforward control [123, 124] is widely used when the disturbance can be measured directly. By measuring disturbances, the control system can take corrective action before the disturbance affects the controlled variables. Thus, adding feedforward control is a promising way to improve control performance against external disturbances. A block diagram for feedforward control is illustrated in Figure 5.1. In Figure 5.1, the disturbance comes into the plant we are interested in through a disturbance plant. The feedforward controller modifies the control signal from the PID controller to take quick action against the disturbance.

The objective of ratio control is to maintain the ratio of two process outputs at the desired value [123]. In Figure 5.2, output 1 comes into a ratio controller. Then the ratio controller sets a targeting ratio between output 1 and output 2 and multiply the value of output 1 with the ratio. PID controller 2 receives it as a reference signal. As a result, the ratio between output 1 and 2 are kept the desired value.

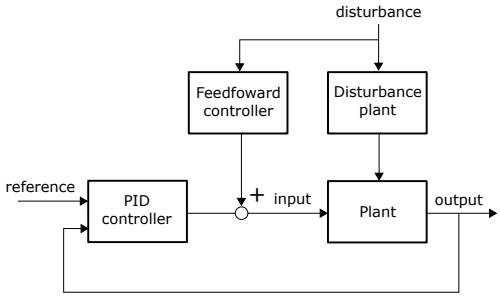

Figure 5.1: A block diagram of standard PID control with feedforward control. The control input is adjusted by the feedforward controller which receives the measurement of the external disturbance.

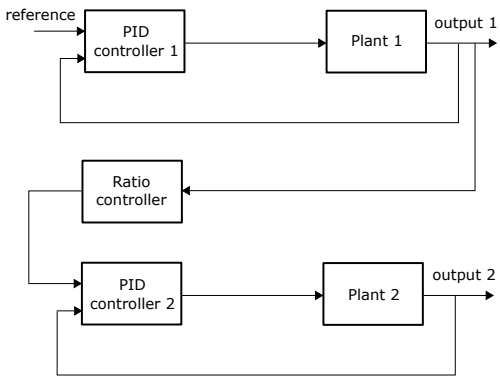

Figure 5.2: A block diagram of ratio control. The control input is adjusted by the ratio controller, which receives the measurement of a different plant as a reference signal.

5.1.2 Literature review

In this chapter, we focus on the problem of event-based control applying feedforward control and ratio control. Event-based control has received much attention from the 1990s [82, 83, 87, 88]. The main motivation for event-based feedback control is to reduce communication among the system components while keeping the same level of performance. Event-based PID control is discussed by many research groups [82, 92, 94, 125–128]. As presented in [82], event-based PID control can significantly reduce the communication effort with only slight or no degradation of control performance. This fact motivates the process industry to use event-based PID control [98]. In [96, 97], the event-based PID control is evaluated in a real industrial plant.

Some practical problems when introducing event-based PI control are discussed

in [93, 94, 127]. In [94], it is shown that event-based sampling may result in the sticking effect or stationary large oscillations. To overcome these problems, [94] proposes PIDPLUS [129–131]. The asymptotic stability conditions are derived with a relative threshold policy in [127]. Furthermore, [93, 128, 132, 133] focus on actuator saturation for event-based control. Actuator saturation is often observed in practical application due to physical or safety constraints. Even for linear plants, the closed-loop with actuator saturation may become unstable and stability can be guaranteed only locally [134]. Thus, it is important to consider actuator saturation for event-based control. In fact, the stability region is influenced by the use of event-based control [93]. In [93], it is shown that an anti-windup technique can significantly improve the performance of control systems with event-based sampling. The authors of [133] propose to introduce an event-based anti-windup scheme. In [128], the authors consider zero-order hold between a controller and an actuator and derive asymptotic stability conditions subject to actuator saturation.

PID controller design problems are considered in [126, 127]. In [126], the authors introduce a linear quadratic (LQ) cost and propose a design synthesis which minimizes that cost. The event-based sampling strategy is also proposed based on the value of the LQ cost. PI control synthesis with a relative threshold strategy is proposed in [127].

Sensor scheduling among multiple PI control systems is investigaed in [125]. However, the event-based sampling of enhanced PID control using additional sensors such as cascade control, PID with feedforward control, or decouplering control [123, 124] are not yet studied, even though these control architectures are widely introduced in industrial process control systems. Thus, it is important to consider these event-based architectures for wireless process control systems.

5.1.3 Contribution

The contributions of this work are the followings: i) we formulate event-based feedforward control and ratio control, and derive stability conditions by using LMIs when the control input is subject to actuator saturation, ii) we derive stability conditions of event-based feedforward control with anti-windup compensation, and iii) we provide a numerical example which shows that the event-based feedforward control significantly reduces the communication without performance degradation compared to conventional continuous-time feedforward control. Furthermore, we illustrate that the anti-windup compensation is effective.

5.1.4 Organization

The remainder of this chapter is organized as follows. In Section 5.2, we introduce a plant model and an actuator saturation model interested in this chapter. Then we introduce a controller model and derive its closed loop formulation in Section 5.3. Stability conditions for some cases are studied in Section 5.4. Anti-windup compensation is introduced and its stability conditions are derived in Section 5.5. In Section 5.6, some numerical examples are provided. Section 5.7 presents the summary of this chapter.

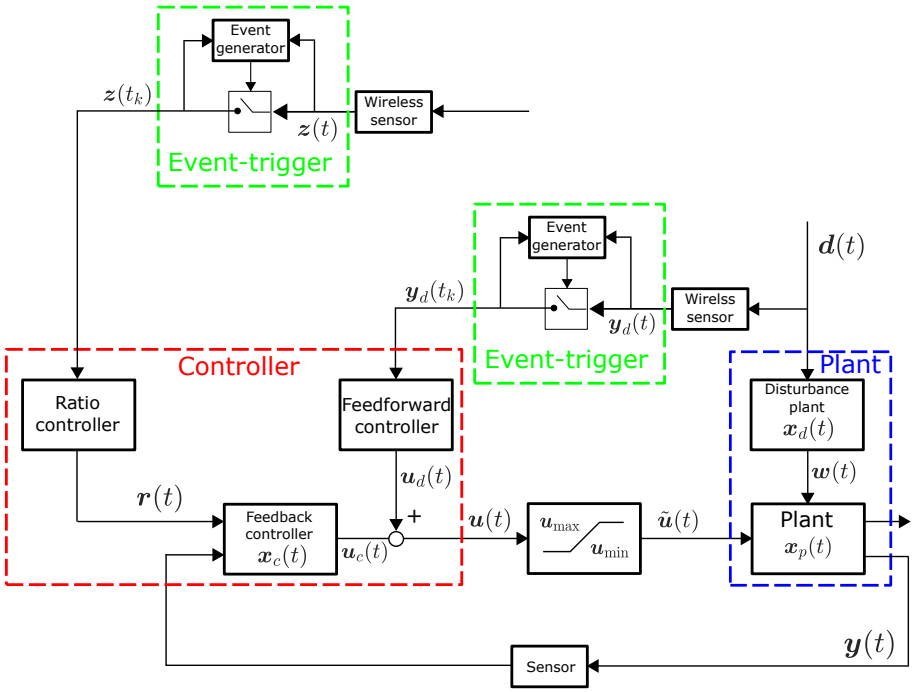

Figure 5.3: A block diagram of the control system with event-based feedforward control and ratio control. In this system, two event-triggers are implemented. The event-trigger for feedforward control generates an event when the value of a disturbance changes a lot, while the event-trigger for ratio control generates an event when the output of the upstream plant changes a lot.

5.2 System Model

In this chapter, we consider event-based feedforward control compensating an external disturbance and event-based ratio control as depicted in Figure 5.3. The plant is given by a continuous-time linear system

$$\dot{\boldsymbol{x}}_p(t) = \tilde{\boldsymbol{A}}\boldsymbol{x}_p(t) + \tilde{\boldsymbol{B}}\tilde{\boldsymbol{u}}(t) + \tilde{\boldsymbol{B}}_w\boldsymbol{w}(t), \quad \boldsymbol{x}_p(0) = \boldsymbol{x}_{p0} \tag{5.1}$$

$$\boldsymbol{y}(t) = \tilde{\boldsymbol{C}}\boldsymbol{x}_p(t) \tag{5.2}$$

where $\boldsymbol{x}_p \in \mathbb{R}^{n_p}$ denotes the state, $\tilde{\boldsymbol{u}} \in \mathbb{R}^m$ control, $\boldsymbol{w} \in \mathbb{R}^p$ disturbance, and $\boldsymbol{y} \in \mathbb{R}^q$ the measurement output. The disturbance affects the plant state through the linear disturbance system

$$\dot{\boldsymbol{x}}_d(t) = \tilde{\boldsymbol{A}}_d\boldsymbol{x}_d(t) + \tilde{\boldsymbol{B}}_d\boldsymbol{d}(t), \quad \boldsymbol{x}_d(0) = \boldsymbol{x}_{d0} \tag{5.3}$$

$$\boldsymbol{w}(t) = \tilde{\boldsymbol{C}}_w\boldsymbol{x}_d(t) \tag{5.4}$$

where $\boldsymbol{x}_d \in \mathbb{R}^{n_d}$ denotes the disturbance state, $\boldsymbol{d} \in \mathbb{R}^r$ the original disturbance, which is assumed to be continuous in t and bounded according to

$$\boldsymbol{d} \in \mathcal{V}_D = \left\{ \boldsymbol{d} \in \mathbb{R}^r : \boldsymbol{d}^\top \boldsymbol{Q}_D \boldsymbol{d} \leq \epsilon_D^{-1} \right\} \tag{5.5}$$

with $\boldsymbol{Q}_D \in \mathbb{S}^r_{++}$ and $\epsilon_D > 0$. The matrices $\tilde{\boldsymbol{A}}$, $\tilde{\boldsymbol{B}}$, $\tilde{\boldsymbol{B}}_w$, $\tilde{\boldsymbol{C}}$, $\tilde{\boldsymbol{A}}_d$, $\tilde{\boldsymbol{B}}_d$, and $\tilde{\boldsymbol{C}}_w$ are real matrices of appropriate dimensions. In the following, for the plant (5.1)–(5.2), we assume that $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}})$ is controllable and $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{C}})$ is observable. To ensure the boundedness of $\boldsymbol{w}(t)$, we also need to assume that the disturbance system (5.3)–(5.4) is stable, i.e., $\tilde{\boldsymbol{A}}_d$ is Hurwitz.

The plant input $\tilde{\boldsymbol{u}}(t)$ is given by $\tilde{\boldsymbol{u}}(t) = \mathbf{sat}(\boldsymbol{u}(t))$, where $\mathbf{sat}(\cdot)$ denotes the saturation function

$$\mathbf{sat}(\boldsymbol{u})[i] = \begin{cases} \boldsymbol{u}_{\max}[i], & \text{if } \boldsymbol{u}[i] > \boldsymbol{u}_{\max}[i]; \\ \boldsymbol{u}[i], & \text{if } -\boldsymbol{u}_{\min}[i] \leq \boldsymbol{u}[i] \leq \boldsymbol{u}_{\max}[i]; \\ -\boldsymbol{u}_{\min}[i], & \text{if } \boldsymbol{u}[i] < -\boldsymbol{u}_{\min}[i], \end{cases} \tag{5.6}$$

with $i \in \{1, \ldots, m\}$, where $\boldsymbol{u}_{\max}$ and $\boldsymbol{u}_{\min}$ are the upper and lower bound vectors of the input $\tilde{\boldsymbol{u}}$ with $\boldsymbol{u}_{\max}[i] \geq 0$ and $\boldsymbol{u}_{\min}[i] \geq 0$ for all i, respectively. For simplicity, we assume symmetric constraints $\boldsymbol{u}_0 = \boldsymbol{u}_{\max} = \boldsymbol{u}_{\min}$.

5.3 Feedback Control with Event-based Feedforward and Ratio Control

The goal of this chapter is to investigate event-based feedforward control and ratio control when applied to a feedback control system already under operation. We assume that the feedback control is established with continuous-time information exchanged through wired communication.

5.3.1 Feedback controller

In the following we consider a general linear dynamic output feedback controller given by

$$\dot{\boldsymbol{x}}_c(t) = \tilde{\boldsymbol{A}}_c \boldsymbol{x}_c(t) + \tilde{\boldsymbol{B}}_c \boldsymbol{y}(t) + \tilde{\boldsymbol{B}}_{cR} \boldsymbol{r}(t), \quad \boldsymbol{x}_c(0) = \boldsymbol{x}_{c0} \tag{5.7}$$

$$\boldsymbol{u}_c(t) = \tilde{\boldsymbol{C}}_c \boldsymbol{x}_c(t) + \tilde{\boldsymbol{D}}_c \boldsymbol{y}(t) + \tilde{\boldsymbol{D}}_{cR} \boldsymbol{r}(t) \tag{5.8}$$

where $\boldsymbol{x}_c \in \mathbb{R}^{n_c}$ denotes the state, $\boldsymbol{u}_c \in \mathbb{R}^m$ feedback control, and $\boldsymbol{r} \in \mathbb{R}^s$ reference signal. The matrices $\tilde{\boldsymbol{A}}_c$, $\tilde{\boldsymbol{B}}_c$, $\tilde{\boldsymbol{B}}_{cR}$, $\tilde{\boldsymbol{C}}_c$, $\tilde{\boldsymbol{D}}_c$, and $\tilde{\boldsymbol{D}}_{cR}$ are real matrices of appropriate dimensions.

5.3.2 Feedforward controller

We assume that the disturbance considered can be observed by an event-based wireless sensor which is described as

$$\boldsymbol{y}_d(t) = \tilde{\boldsymbol{C}}_d \boldsymbol{d}(t) \tag{5.9}$$

where $\boldsymbol{y}_d \in \mathbb{R}^{m_d}$ is the original disturbance measurement output and $\tilde{\boldsymbol{C}}_d$ is a real matrix with appropriate dimension. Based on the measurement $\boldsymbol{y}_d(t)$, a wireless sensor invokes a

new communication event. Let t_k with $k \in \mathbb{N}$ be the time of transmission k. Then the new event occurs whenever the disturbance error $\boldsymbol{e}(t)$ given by

$$\boldsymbol{e}_d(t) = \boldsymbol{y}_d(t) - \boldsymbol{y}_d(t_k), \quad \forall t \in [t_k, t_{k+1}) \tag{5.10}$$

reaches the boundary of the set

$$\mathcal{W}_D = \left\{ \boldsymbol{e}_d \in \mathbb{R}^{m_d} : \boldsymbol{e}_d^\top \boldsymbol{R}_D \boldsymbol{e}_d \leq \delta_d^{-1} \right\} \tag{5.11}$$

with $\boldsymbol{R} \in \mathbb{S}_{++}^{m_d}$ and $\delta > 0$, that is, when $\boldsymbol{e}_d(t) \in \partial\mathcal{W}_D$.

The feedforward controller calculates the output $\boldsymbol{u}_d(t) \in \mathbb{R}^m$ based on the disturbance information from the wireless sensor. We consider static feedforward control described as

$$\boldsymbol{u}_d(t) = \tilde{\boldsymbol{D}}_{cD} \boldsymbol{y}_d(t_k), \quad \forall t \in [t_k, t_{k+1}), \tag{5.12}$$

which compensates the control vector by

$$\boldsymbol{u}(t) = \boldsymbol{u}_c(t) + \boldsymbol{u}_d(t). \tag{5.13}$$

5.3.3 Ratio controller

Let us denote the output signal from the other plant as $\boldsymbol{z}(t) \in \mathbb{R}^{m_z}$. We assume that $\boldsymbol{z}(t)$ is continuous in t and bounded according to

$$z \in \mathcal{V}_Z = \left\{ \boldsymbol{d} \in \mathbb{R}^{m_z} : \boldsymbol{z}^\top \boldsymbol{Q}_Z \boldsymbol{d} \leq \epsilon_Z^{-1} \right\} \tag{5.14}$$

with $\boldsymbol{Q}_Z \in \mathbb{S}_{++}^{m_z}$ and $\epsilon_Z > 0$. Based on the measurement $\boldsymbol{z}(t)$, a wireless sensor invokes a new communication event. Then the new event occurs whenever the error $\boldsymbol{e}_z(t)$ given by

$$\boldsymbol{e}_z(t) = \boldsymbol{z}(t) - \boldsymbol{z}(t_k), \quad \forall t \in [t_k, t_{k+1}) \tag{5.15}$$

reaches the boundary of the set

$$\mathcal{W}_Z = \left\{ \boldsymbol{e}_z \in \mathbb{R}^{m_r} : \boldsymbol{e}_z^\top \boldsymbol{R}_Z \boldsymbol{e}_z \leq \delta_z^{-1} \right\} \tag{5.16}$$

with $\boldsymbol{R}_Z \in \mathbb{S}_{++}^{m_r}$ and $\delta_z > 0$, i.e., when $\boldsymbol{e}_z(t) \in \partial\mathcal{W}_Z$. Then the ratio controller calculates the reference signal as

$$\boldsymbol{r}(t) = \tilde{\boldsymbol{D}}_{cZ} \boldsymbol{z}(t_k), \quad \forall t \in [t_k, t_{k+1}). \tag{5.17}$$

5.3.4 Closed-loop system

From (5.1)–(5.4), (5.7)–(5.8), (5.10)–(5.12), (5.13)–(5.15), and (5.17), and by introducing the augmented state vector

$$\boldsymbol{x}(t) = \begin{bmatrix} \boldsymbol{x}_p(t) \\ \boldsymbol{x}_c(t) \\ \boldsymbol{x}_d(t) \end{bmatrix} \in \mathbb{R}^n$$

where $n = n_p + n_c + n_d$, we obtain the augmented state-space model

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\tilde{\boldsymbol{u}}(t) + \boldsymbol{B}_D\boldsymbol{d}(t) + \boldsymbol{B}_Z\boldsymbol{z}(t) + \boldsymbol{B}_{E_z}\boldsymbol{e}_z(t) \tag{5.18}$$

$$\boldsymbol{u}(t) = \boldsymbol{K}\boldsymbol{x}(t) + \boldsymbol{K}_D\boldsymbol{d}(t) + \boldsymbol{K}_{E_d}\boldsymbol{e}_d(t) + \boldsymbol{K}_Z\boldsymbol{z}(t) + \boldsymbol{K}_{E_z}\boldsymbol{e}_z(t) \tag{5.19}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) \tag{5.20}$$

with

$$\boldsymbol{A} = \begin{bmatrix} \tilde{\boldsymbol{A}} & \boldsymbol{O} & \tilde{\boldsymbol{B}}_w\tilde{\boldsymbol{C}}_w \\ \tilde{\boldsymbol{B}}_c\tilde{\boldsymbol{C}} & \tilde{\boldsymbol{A}}_c & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{O} & \tilde{\boldsymbol{A}}_d \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} \tilde{\boldsymbol{B}} \\ \boldsymbol{O} \\ \boldsymbol{O} \end{bmatrix},$$

$$\boldsymbol{B}_D = \begin{bmatrix} \boldsymbol{O} \\ \boldsymbol{O} \\ \tilde{\boldsymbol{B}}_d \end{bmatrix}, \quad \boldsymbol{B}_Z = \begin{bmatrix} \boldsymbol{O} \\ \tilde{\boldsymbol{B}}_{cR}\tilde{\boldsymbol{D}}_{cZ} \\ \boldsymbol{O} \end{bmatrix}, \quad \boldsymbol{B}_{E_z} = \begin{bmatrix} \boldsymbol{O} \\ -\tilde{\boldsymbol{B}}_{cR}\tilde{\boldsymbol{D}}_{cZ} \\ \boldsymbol{O} \end{bmatrix},$$

$$\boldsymbol{K} = \begin{bmatrix} \tilde{\boldsymbol{D}}_c\tilde{\boldsymbol{C}} & \tilde{\boldsymbol{C}}_c & \boldsymbol{O} \end{bmatrix}, \quad \boldsymbol{K}_D = \tilde{\boldsymbol{D}}_{cD}\tilde{\boldsymbol{C}}_d, \quad \boldsymbol{K}_{E_d} = -\tilde{\boldsymbol{D}}_{cD},$$

$$\boldsymbol{K}_Z = \tilde{\boldsymbol{D}}_{cR}\tilde{\boldsymbol{D}}_{cZ}, \quad \boldsymbol{K}_{E_z} = -\tilde{\boldsymbol{D}}_{cR}\tilde{\boldsymbol{D}}_{cZ}, \quad \boldsymbol{C} = \begin{bmatrix} \tilde{\boldsymbol{C}} & \boldsymbol{O} & \boldsymbol{O} \end{bmatrix}.$$

To characterize the stability of the closed-loop system (5.18)–(5.20), we first introduce the deadzone nonlinearity [134], which is defined by

$$\phi(\boldsymbol{u}) = \mathbf{sat}(\boldsymbol{u}) - \boldsymbol{u}. \tag{5.21}$$

The deadzone nonlinearity allows us to use a modified sector condition as follows.

Lemma 5.1. *[134] If $\boldsymbol{v} \in \mathbb{R}^m$ and $\bar{\boldsymbol{z}} \in \mathbb{R}^m$ are elements of the set*

$$\mathcal{S} = \left\{ \boldsymbol{v}, \bar{\boldsymbol{z}} \in \mathbb{R}^m : |\boldsymbol{v}[i] - \bar{\boldsymbol{z}}[i]| \leq \boldsymbol{u}_0[i],\ \forall i \in \{1, \ldots, m\} \right\}$$

then the nonlinearity $\phi(\boldsymbol{v})$ satisfies the inequality

$$\phi(\boldsymbol{v})^\top \boldsymbol{T}(\phi(\boldsymbol{v}) + \bar{\boldsymbol{z}}) \leq 0$$

for any diagonal matrix $\boldsymbol{T} \in \mathbb{S}_{++}$.

By using (5.21), we can rewrite the closed-loop system as

$$\begin{aligned} \dot{\boldsymbol{x}}(t) =& \bar{\boldsymbol{A}}\boldsymbol{x}(t) + \bar{\boldsymbol{B}}\phi(\boldsymbol{K}\boldsymbol{x}(t) + \boldsymbol{K}_D\boldsymbol{d}(t) + \boldsymbol{K}_{E_d}\boldsymbol{e}_d(t) + \boldsymbol{K}_Z\boldsymbol{z}(t) + \boldsymbol{K}_{E_z}\boldsymbol{e}_z(t)) \\ &+ (\boldsymbol{B}_D + \boldsymbol{B}\boldsymbol{K}_D)\boldsymbol{d}(t) + \boldsymbol{B}\boldsymbol{K}_{E_d}\boldsymbol{e}_d(t) \\ &+ (\boldsymbol{B}_Z + \boldsymbol{B}\boldsymbol{K}_Z)\boldsymbol{z}(t) + (\boldsymbol{B}_{E_z} + \boldsymbol{B}\boldsymbol{K}_{E_z})\boldsymbol{e}_z(t) \end{aligned} \tag{5.22}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t). \tag{5.23}$$

where $\bar{\boldsymbol{A}} = \boldsymbol{A} + \boldsymbol{B}\boldsymbol{K}$ and $\bar{\boldsymbol{B}} = \boldsymbol{B}$. Note that the feedback controller (5.7)–(5.8) can stabilize the plant (5.1)–(5.2) at least for a sufficiently small region around the equilibrium point of the system (5.22). Thus, the matrix $\bar{\boldsymbol{A}}$ is Hurwitz. The closed-loop system is illustrated in Figure 5.4.

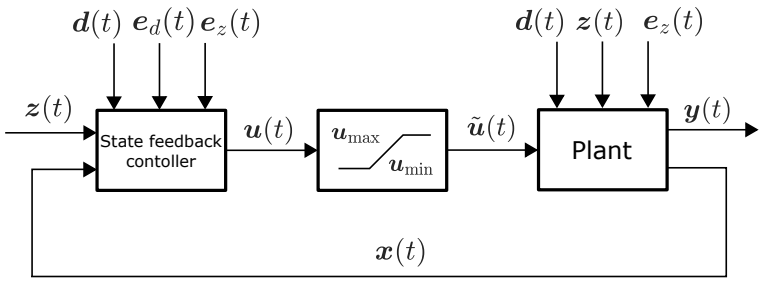

Figure 5.4: A block diagram of the closed-loop system with event-based feedforward control and ratio control. The errors due to event-based samplings $\boldsymbol{e}_d$ and $\boldsymbol{e}_z$ are treated as additional external disturbances.

5.4 Stability Analysis under Actuator Saturation

For practical application, it is important to consider input constraints since almost all systems have physical or safety constraints. Under actuator saturation, the stability is guaranteed only locally. Hence, our stability analysis focuses on estimating the stability region. In this section, we derive stability conditions for some cases as a form of LMIs.

5.4.1 Continuous-time feedforward control and ratio control

First, we derive stability conditions of the system with continuous-time feedforward control and ratio control, which is simple extension of the discussion in [134]. These results are used later to evaluate the effect of the event-based control.

Stability of continuous-time feedforward control

To derive the stability conditions of feedforward control subject to actuator saturation, for simplicity, we assume that $\boldsymbol{r}(t) \equiv 0$. With continuous-time feedforward control, the closed-loop system (5.22)–(5.23) can be rewritten as

$$\dot{\boldsymbol{x}}(t) = \bar{\boldsymbol{A}}\boldsymbol{x}(t) + \bar{\boldsymbol{B}}\phi(\boldsymbol{K}\boldsymbol{x}(t) + \boldsymbol{K}_D\boldsymbol{d}(t)) + (\boldsymbol{B}_D + \boldsymbol{B}\boldsymbol{K}_D)\boldsymbol{d}(t) \tag{5.24}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t). \tag{5.25}$$

Then we have the following proposition.

Proposition 5.2. *If there exist a symmetric matrix $\boldsymbol{W} \in \mathbb{S}^n_{++}$, a diagonal matrix $\boldsymbol{S} \in \mathbb{S}^m_{++}$, a matrix $\boldsymbol{Z} \in \mathbb{R}^{m\times n}$, three positive scalars τ_1, τ_2 and η satisfying*

$$\begin{bmatrix} \boldsymbol{W}\bar{\boldsymbol{A}}^\top + \bar{\boldsymbol{A}}\boldsymbol{W} + \tau_1\boldsymbol{W} & \star & \star \\ \boldsymbol{S}\bar{\boldsymbol{B}}^\top - \boldsymbol{Z} - \boldsymbol{K}\boldsymbol{W} & -2\boldsymbol{S} & \star \\ (\boldsymbol{B}_D + \boldsymbol{B}\boldsymbol{K}_D)^\top & -\boldsymbol{K}_D^\top & -\tau_2\boldsymbol{Q}_D \end{bmatrix} < 0 \tag{5.26}$$

$$\begin{bmatrix} \boldsymbol{W} & \boldsymbol{Z}[i]^\top \\ \boldsymbol{Z}[i] & \eta \boldsymbol{u}_0[i]^2 \end{bmatrix} \geq 0, \quad i = 1, \ldots, m \tag{5.27}$$

$$-\tau_1 \epsilon_D + \tau_2 \eta < 0 \tag{5.28}$$

then for any $\boldsymbol{d} \in \mathcal{V}_D$ *and* $\boldsymbol{x}(0) \in \mathcal{E}(\boldsymbol{P}, \eta)$ *with* $\boldsymbol{P} = \boldsymbol{W}^{-1}$*, the state* $\boldsymbol{x}(t)$ *of closed-loop system* (5.24)–(5.25) *does not leave the ellipsoid* $\mathcal{E}(\boldsymbol{P}, \eta)$ *for all* $t \geq 0$*.*

Proof. It follows from Remark 3.1 and Proposition 3.6 in [134]. □

Remark 5.3. *It is obvious that the stability conditions of the system without feedforward control can be derived by substituting* $\boldsymbol{K}_D = \boldsymbol{O}$ *which corresponds Proposition 3.6 in [134].*

Stability of continuous-time ratio control

In the same way, we derive the stability conditions of continuous-time ratio control subject to actuator saturation. Here, we assume that $\boldsymbol{d}(t) \equiv 0$. With continuous-time ratio control, the closed-loop system (5.22)–(5.23) can be rewritten as

$$\dot{\boldsymbol{x}}(t) = \bar{\boldsymbol{A}}\boldsymbol{x}(t) + \bar{\boldsymbol{B}}\phi(\boldsymbol{K}\boldsymbol{x}(t) + \boldsymbol{K}_Z \boldsymbol{z}(t)) + (\boldsymbol{B}_Z + \boldsymbol{B}\boldsymbol{K}_Z)\boldsymbol{z}(t) \tag{5.29}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t), \tag{5.30}$$

which is equivalent to (5.24)–(5.25) by replacing $\boldsymbol{K}_Z$ to $\boldsymbol{K}_D$ and $\boldsymbol{B}_Z$ to $\boldsymbol{B}_D$. Thus we have the following proposition.

Proposition 5.4. *If there exist a symmetric matrix* $\boldsymbol{W} \in \mathbb{S}^n_{++}$*, a diagonal matrix* $\boldsymbol{S} \in \mathbb{S}^m_{++}$*, a matrix* $\boldsymbol{Z} \in \mathbb{R}^{m \times n}$*, three positive scalars* τ_1*,* τ_2 *and* η *satisfying*

$$\begin{bmatrix} \boldsymbol{W}\bar{\boldsymbol{A}}^\top + \bar{\boldsymbol{A}}\boldsymbol{W} + \tau_1 \boldsymbol{W} & \star & \star \\ \boldsymbol{S}\bar{\boldsymbol{B}}^\top - \boldsymbol{Z} - \boldsymbol{K}\boldsymbol{W} & -2\boldsymbol{S} & \star \\ (\boldsymbol{B}_Z + \boldsymbol{B}\boldsymbol{K}_Z)^\top & -\boldsymbol{K}_Z^\top & -\tau_2 \boldsymbol{Q}_Z \end{bmatrix} < 0 \tag{5.31}$$

$$\begin{bmatrix} \boldsymbol{W} & \boldsymbol{Z}[i]^\top \\ \boldsymbol{Z}[i] & \eta \boldsymbol{u}_0[i]^2 \end{bmatrix} \geq 0, \quad i = 1, \ldots, m \tag{5.32}$$

$$-\tau_1 \epsilon_Z + \tau_2 \eta < 0 \tag{5.33}$$

then for any $\boldsymbol{d} \in \mathcal{V}_D$ *and* $\boldsymbol{x}(0) \in \mathcal{E}(\boldsymbol{P}, \eta)$ *with* $\boldsymbol{P} = \boldsymbol{W}^{-1}$*, the state* $\boldsymbol{x}(t)$ *of closed-loop system* (5.24)–(5.25) *does not leave the ellipsoid* $\mathcal{E}(\boldsymbol{P}, \eta)$ *for all* $t \geq 0$*.*

5.4.2 Event-based feedforward control and ratio control

Next, we derive stability conditions for event-based feedforward control and ratio control.

Stability of event-based feedforward control

For the feedforward control we have the following theorem.

Theorem 5.5. *If there exist a symmetric matrix $\boldsymbol{W} \in \mathbb{S}^n_{++}$, a diagonal matrix $\boldsymbol{S} \in \mathbb{S}^m_{++}$, a matrix $\boldsymbol{Z} \in \mathbb{R}^{m\times n}$, four positive scalars τ_1, τ_2, τ_3 and η satisfying*

$$\begin{bmatrix} \boldsymbol{W}\bar{\boldsymbol{A}}^\top + \bar{\boldsymbol{A}}\boldsymbol{W} + \tau_1\boldsymbol{W} & \star & \star & \star \\ \boldsymbol{S}\bar{\boldsymbol{B}}^\top - \boldsymbol{Z} - \boldsymbol{K}\boldsymbol{W} & -2\boldsymbol{S} & \star & \star \\ (\boldsymbol{B}_D + \boldsymbol{B}\boldsymbol{K}_D)^\top & -\boldsymbol{K}_D^\top & -\tau_2\boldsymbol{Q}_D & \star \\ \boldsymbol{K}_{E_d}^\top\boldsymbol{B}^\top & -\boldsymbol{K}_{E_d}^\top & \boldsymbol{O} & -\tau_3\boldsymbol{R}_D \end{bmatrix} < 0 \tag{5.34}$$

$$\begin{bmatrix} \boldsymbol{W} & \boldsymbol{Z}[i]^\top \\ \boldsymbol{Z}[i] & \eta\boldsymbol{u}_0[i]^2 \end{bmatrix} \geq 0, \quad i = 1, \ldots, m \tag{5.35}$$

$$-\tau_1\delta\epsilon_D + \tau_2\delta\eta + \tau_3\epsilon_D\eta < 0 \tag{5.36}$$

then for any $\boldsymbol{d} \in \mathcal{V}_D$, $\boldsymbol{e}_d \in \mathcal{W}_D$ and $\boldsymbol{x}(0) \in \mathcal{E}(\boldsymbol{P}, \eta)$ with $\boldsymbol{P} = \boldsymbol{W}^{-1}$, the state $\boldsymbol{x}(t)$ of closed-loop system (5.22)–(5.23) *does not leave the ellipsoid $\mathcal{E}(\boldsymbol{P}, \eta)$ for all $t \geq 0$.*

Proof. By setting $\boldsymbol{v} = \boldsymbol{u} = \boldsymbol{K}\boldsymbol{x} + \boldsymbol{K}_D\boldsymbol{d} + \boldsymbol{K}_{E_d}\boldsymbol{e}_d$ and $\bar{\boldsymbol{z}} = \boldsymbol{u} + \boldsymbol{G}\boldsymbol{x} = \boldsymbol{K}\boldsymbol{x} + \boldsymbol{K}_D\boldsymbol{d} + \boldsymbol{K}_{E_d}\boldsymbol{e}_d + \boldsymbol{G}\boldsymbol{x}$, Lemma 5.1 guarantees that

$$\phi^\top(\boldsymbol{u})\boldsymbol{T}(\phi(\boldsymbol{u}) + \boldsymbol{u} + \boldsymbol{G}\boldsymbol{x}) \leq 0 \tag{5.37}$$

for any $\boldsymbol{x}$ belonging to the set

$$\mathcal{S}_{\boldsymbol{G}} = \{\boldsymbol{x} \in \mathbb{R}^n : |\boldsymbol{G}[i]\boldsymbol{x}| \leq \boldsymbol{u}_0[i], \forall i\}.$$

Consider Lyapunov function candidate

$$V(\boldsymbol{x}) = \boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{x}$$

with $\boldsymbol{P} = \boldsymbol{P}^\top > 0$, which defines the ellipsoid $\mathcal{E}(\boldsymbol{P}, \eta)$. This ellipsoid is included in the set $\mathcal{S}_{\boldsymbol{G}}$ if the condition (5.35) is satisfied. This can be shown by left-multiplying the vector $[\eta\boldsymbol{u}_0[i](\boldsymbol{W}^{-1}\boldsymbol{x})^\top \ \pm 1]$ and right-multiplying $[\eta\boldsymbol{u}_0[i](\boldsymbol{W}^{-1}\boldsymbol{x})^\top \ \pm 1]^\top$ by the matrix in the condition (5.35).

Next, we will show that $\dot{V}(\boldsymbol{x}) < 0$ for any $\boldsymbol{x} \in \mathrm{int}(\mathcal{E}(\boldsymbol{P}, \eta))$ and $\boldsymbol{d} \in \mathcal{V}_D$, $\boldsymbol{e}_d \in \mathcal{W}_D$ so that any trajectories of $\boldsymbol{x}(t)$ never leave the ellipsoid $\mathcal{E}(\boldsymbol{P}, \eta)$. By applying the S-procedure, we have the condition

$$\dot{V}(\boldsymbol{x}) + \tau_1(\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{x} - \eta^{-1}) + \tau_2(\epsilon_D^{-1} - \boldsymbol{d}^\top\boldsymbol{Q}_D\boldsymbol{d}) + \tau_3(\delta^{-1} - \boldsymbol{e}_d^\top\boldsymbol{R}_D\boldsymbol{e}_d) < 0,$$

which can be split further into two conditions:

$$\begin{aligned} \dot{V}(\boldsymbol{x}) + \tau_1\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{x} - \tau_2\boldsymbol{d}^\top\boldsymbol{Q}_D\boldsymbol{d} - \tau_3\boldsymbol{e}_d^\top\boldsymbol{R}_D\boldsymbol{e}_d < 0 \\ -\tau_1\eta^{-1} + \tau_2\epsilon_D^{-1} + \tau_3\delta^{-1} < 0. \end{aligned}$$

The condition (5.36) directly results in the second inequality above. By the inequality (5.37), we have

$$\begin{aligned}&\dot{V}(\boldsymbol{x}) + \tau_1 \boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x} - \tau_2 \boldsymbol{d}^\top \boldsymbol{Q}_D \boldsymbol{d} - \tau_3 \boldsymbol{e}_d^\top \boldsymbol{R}_D \boldsymbol{e}_d \\ &\quad \leq \dot{V}(\boldsymbol{x}) + \tau_1 \boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x} - \tau_2 \boldsymbol{d}^\top \boldsymbol{Q}_D \boldsymbol{d} - \tau_3 \boldsymbol{e}_d^\top \boldsymbol{R}_D \boldsymbol{e}_d - 2\boldsymbol{\phi}^\top \boldsymbol{T}(\boldsymbol{\phi} + \boldsymbol{u} + \boldsymbol{G}\boldsymbol{x}).\end{aligned}$$

By using the system representation (5.18) and transformation $\boldsymbol{W} = \boldsymbol{P}^{-1}$, $\boldsymbol{S} = \boldsymbol{T}^{-1}$ and $\boldsymbol{Z} = \boldsymbol{G}\boldsymbol{W}$, the condition (5.34) guarantees that the right term of the above inequality is negative, which can be shown by left-multiplying $[(\boldsymbol{W}^{-1}\boldsymbol{x})^\top\ (\boldsymbol{S}^{-1}\boldsymbol{\phi})^\top\ \boldsymbol{d}^\top\ \boldsymbol{e}_d^\top]$ and right-multiplying $[(\boldsymbol{W}^{-1}\boldsymbol{x})^\top\ (\boldsymbol{S}^{-1}\boldsymbol{\phi})^\top\ \boldsymbol{d}^\top\ \boldsymbol{e}_d^\top]^\top$ by the matrix in the condition (5.34). This completes the proof. □

Stability of event-based ratio control

Similarly, we have the stability conditions for the event-based ratio control.

Corollary 5.6. *If there exist a symmetric matrix $\boldsymbol{W} \in \mathbb{S}^n_{++}$, a diagonal matrix $\boldsymbol{S} \in \mathbb{S}^m_{++}$, a matrix $\boldsymbol{Z} \in \mathbb{R}^{m\times n}$, four positive scalars τ_1, τ_2, τ_3 and η satisfying*

$$\begin{bmatrix} \boldsymbol{W}\bar{\boldsymbol{A}}^\top + \bar{\boldsymbol{A}}\boldsymbol{W} + \tau_1 \boldsymbol{W} & \star & \star & \star \\ \boldsymbol{S}\bar{\boldsymbol{B}}^\top - \boldsymbol{Z} - \boldsymbol{K}\boldsymbol{W} & -2\boldsymbol{S} & \star & \star \\ (\boldsymbol{B}_Z + \boldsymbol{B}\boldsymbol{K}_Z)^\top & -\boldsymbol{K}_Z^\top & -\tau_2 \boldsymbol{Q}_Z & \star \\ (\boldsymbol{B}_{E_z} + \boldsymbol{B}\boldsymbol{K}_{E_z})^\top & -\boldsymbol{K}_{E_z}^\top & \boldsymbol{O} & -\tau_3 \boldsymbol{R}_Z \end{bmatrix} < 0 \tag{5.38}$$

$$\begin{bmatrix} \boldsymbol{W} & \boldsymbol{Z}[i]^\top \\ \boldsymbol{Z}[i] & \eta \boldsymbol{u}_0[i]^2 \end{bmatrix} \geq 0, \quad i = 1, \ldots, m \tag{5.39}$$

$$-\tau_1 \delta \epsilon_Z + \tau_2 \delta \eta + \tau_3 \epsilon_Z \eta < 0 \tag{5.40}$$

then for any $\boldsymbol{z} \in \mathcal{V}_Z$, $\boldsymbol{e}_z \in \mathcal{W}_Z$ and $\boldsymbol{x}(0) \in \mathcal{E}(\boldsymbol{P}, \eta)$ with $\boldsymbol{P} = \boldsymbol{W}^{-1}$, the state $\boldsymbol{x}(t)$ of closed-loop system (5.22)–(5.23) *does not leave the ellipsoid $\mathcal{E}(\boldsymbol{P}, \eta)$ for all $t \geq 0$.*

5.5 Anti-windup Compensation

It is known that anti-windup is effective to compensate performance degradation due to actuator saturation [93, 134]. In this section, we derive the stability conditions for event-based feedforward control with anti-windup compensation. The idea of anti-windup compensation is to feed back the difference between control input and actual actuator output, i.e., $\boldsymbol{\phi}(\boldsymbol{u})$, to the controller as shown in Figure 5.5. For simplicity, we discuss that the event-based feedforward contol with anti-windup compensation, and the reference signal is assumed to be $\boldsymbol{r}(t) \equiv 0$. We consider that the anti-windup feedback gain is static $\boldsymbol{K}_{AW}$, then the controller state is given by

$$\dot{\boldsymbol{x}}_c(t) = \tilde{\boldsymbol{A}}_c \boldsymbol{x}_c(t) + \tilde{\boldsymbol{B}}_c \boldsymbol{y}(t) + \boldsymbol{K}_{AW} \boldsymbol{\phi}(\boldsymbol{u}), \quad \boldsymbol{x}_c(0) = \boldsymbol{x}_{c0},$$

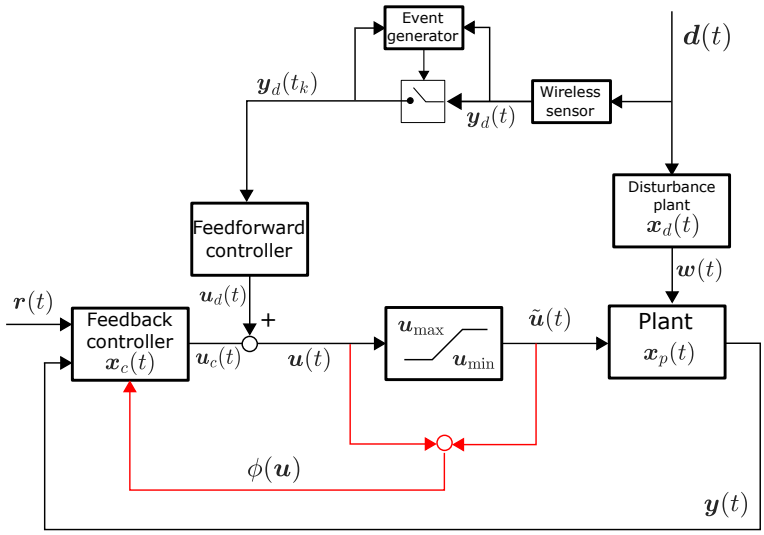

Figure 5.5: A block diagram of the feedforward control with anti-windup compensation. The gap between control input $\boldsymbol{u}(t)$ and actual actuator output $\tilde{\boldsymbol{u}}(t)$ fed back to the feedback controller.

and therefore, the closed loop system becomes

$$\dot{\boldsymbol{x}}(t) = \bar{\boldsymbol{A}}\boldsymbol{x}(t) + \boldsymbol{B}_{AW}\phi(\boldsymbol{K}\boldsymbol{x}(t) + \boldsymbol{K}_D\boldsymbol{d}(t) + \boldsymbol{K}_E\boldsymbol{e}(t)) + (\boldsymbol{B}_D + \boldsymbol{B}\boldsymbol{K}_D)\boldsymbol{d}(t) + \boldsymbol{B}\boldsymbol{K}_E\boldsymbol{e}(t) \tag{5.41}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t). \tag{5.42}$$

with

$$\boldsymbol{B}_{AW} = \begin{bmatrix} \tilde{\boldsymbol{B}} \\ \boldsymbol{K}_{AW} \\ \boldsymbol{O} \end{bmatrix}.$$

Now, we have the following stability conditions which is obtained by replacing $\bar{\boldsymbol{B}}$ by $\boldsymbol{B}_{AW}$ in Theorem 5.5.

Corollary 5.7. *If there exist a symmetric matrix $\boldsymbol{W} \in \mathbb{S}^n_{++}$, a diagonal matrix $\boldsymbol{S} \in \mathbb{S}^m_{++}$, a matrix $\boldsymbol{Z} \in \mathbb{R}^{m\times n}$, four positive scalars τ_1, τ_2, τ_3 and η satisfying*

$$\begin{bmatrix} \boldsymbol{W}\bar{\boldsymbol{A}}^\top + \bar{\boldsymbol{A}}\boldsymbol{W} + \tau_1\boldsymbol{W} & \star & \star & \star \\ \boldsymbol{S}\boldsymbol{B}_{AW}^\top - \boldsymbol{Z} - \boldsymbol{K}\boldsymbol{W} & -2\boldsymbol{S} & \star & \star \\ (\boldsymbol{B}_D + \boldsymbol{B}\boldsymbol{K}_D)^\top & -\boldsymbol{K}_D^\top & -\tau_2\boldsymbol{Q}_D & \star \\ \boldsymbol{K}_E^\top\boldsymbol{B}^\top & -\boldsymbol{K}_E^\top & \boldsymbol{O} & -\tau_3\boldsymbol{R} \end{bmatrix} < 0 \tag{5.43}$$

$$\begin{bmatrix} \boldsymbol{W} & \boldsymbol{Z}[i]^\top \\ \boldsymbol{Z}[i] & \eta \boldsymbol{u}_0[i]^2 \end{bmatrix} \geq 0, \quad i = 1, \ldots, m \tag{5.44}$$

$$-\tau_1 \delta \epsilon_D + \tau_2 \delta \eta + \tau_3 \epsilon_D \eta < 0 \tag{5.45}$$

then for any $\boldsymbol{d} \in \mathcal{V}_D$, $\boldsymbol{e} \in \mathcal{W}$ *and* $\boldsymbol{x}(0) \in \mathcal{E}(\boldsymbol{P}, \eta)$ *with* $\boldsymbol{P} = \boldsymbol{W}^{-1}$, *the state* $\boldsymbol{x}(t)$ *of closed-loop system* (5.41)–(5.42) *does not leave the ellipsoid* $\mathcal{E}(\boldsymbol{P}, \eta)$ *for all* $t \geq 0$.

5.6 Numerical Example

In this section, we provide numerical examples of scalar PI control to see the effect of event-based feedforward control. In the examples, we compare the performance among the event-based feedforward control, continuous-time feedforward control, and without feedforward control.

5.6.1 Plant and controller

Consider the following scalar unstable system

$$\begin{aligned} \dot{x}_p(t) &= 0.1 x_p(t) + \tilde{u}(t) + 2 x_d(t), \quad x(0) = 0 \\ y(t) &= x_p(t) \end{aligned}$$

and the disturbance system

$$\begin{aligned} \dot{x}_d(t) &= -3 x_d(t) + 2 d(t), \quad x_d(0) = 0 \\ w(t) &= x_d(t) \\ y_d(t) &= d(t_k), \ t \in [t_k, t_{k+1}) \end{aligned}$$

with PI control including feedforward controller

$$\begin{aligned} \dot{x}_c(t) &= -y(t) \\ u(t) &= x_c(t) - 1.2 y(t) + k_f y_d(t) \end{aligned}$$

where k_f is the scalar feedfoward gain. The input is affected by the actuator saturation

$$\tilde{u}(t) = \mathrm{sat}(u(t)) = \begin{cases} 2, & \text{if } u(t) > 2; \\ u(t), & \text{if } -2 \leq u(t) \leq 2; \\ -2, & \text{if } u(t) < -2. \end{cases}$$

An event is generated whenever $e^2(t) = \delta^{-1}$, i.e., $\boldsymbol{R} = 1$. We define $\bar{e} \triangleq \delta^{-1/2}$ for simplicity.

5.6.2 Computation of stability region

In the simulation, we evaluate the region of stability $\mathcal{E}(\boldsymbol{P}, \eta)$ for some cases. To estimate the region, we formulate the following optimization problem with different constraints

corresponding to the three cases in Sections 5.4 and 5.5:

$$\begin{aligned} \min \quad & \mathrm{trace}(-\boldsymbol{W}) \\ \text{s.t.} \quad & (5.26)\text{–}(5.28), \text{ or} \\ & (5.34)\text{–}(5.36), \text{ or} \\ & (5.43)\text{–}(5.45). \end{aligned}$$

With this objective function, the optimization problem is a semi-definite program under given $\tau_i,\ i = 1, 2, 3$, which is effectively solved by YALMIP toolbox [135]. Note that the outcome of the optimization problem depends on the values of τ_i. Thus, a search on a grid defined by τ_i is needed in order to obtain the maximum stability region [134].

Figure 5.6 shows the stability regions of x_p and x_c derived based on Theorem 5.2 and Theorem 5.5 for the three cases: (i) event-based feedforward control (ET-FF: red) with $\bar{e} = 0.1$ and $k_f = -0.75$, (ii) continuous-time feedforward control (CT-FF: green) with $k_f = -0.75$, and (iii) no feedforward control (no-FF: blue) with $k_f = 0$.

We find that the continuous-time feedforward control obtains the largest stability region and PI control without feedfoward control does the smallest. The event-based feedforward control has smaller stability region than continuous-time one. The difference of the stability regions with the continuous-time feedforward control stems from the disturbance measurement error $\boldsymbol{e}(t)$. However, even if the information of disturbance is thinned out by event-generator, the event-based feedforward control still has larger stability region compared with the case without feedforward control. In addition, comparing two event-triggering conditions with $\bar{e} = 0.1$ and $\bar{e} = 0.3$, the case with $\bar{e} = 0.1$ has larger stability region. This is due to smaller disturbance error $e(t)$ than the case with $\bar{e} = 0.3$. We also compare the two cases: event-based feedforward control with and without anti-windup compensation. The result with $k_{AW} = -1$ is shown in Figure 5.7. We find that anti-windup compensation has much influence on the size of the stability region for event-based feedforward control.

5.6.3 Behaviors of the control loop

We also show the behaviors of each control loop with a given disturbance. Here, we assume that a disturbance appears when $t = 1$ [s] with

$$d(t) = 1 - \mathrm{e}^{-0.5(t+1)}. \tag{5.46}$$

The results of the three cases: (i) PI control with event-based feedforward control (ET-FF: red) with $\bar{e} = 0.1$ and $k_f = -0.75$, (ii) PI control with continuous-time feedforward control (CT-FF, green) with $k_f = -0.75$, and (iii) PI control without feedforward control (no-FF, blue), are shown in Figures 5.6 and Figure 5.8. From Figure 5.8, we find that the event-based feedfoward control achieves almost the same performance against the disturbance as the continuous-time feedforward control with only 9 samples of the disturbance being communicated. This implies that the event-based feedforward control significantly reduces the communication with basically no performance degradation compared with

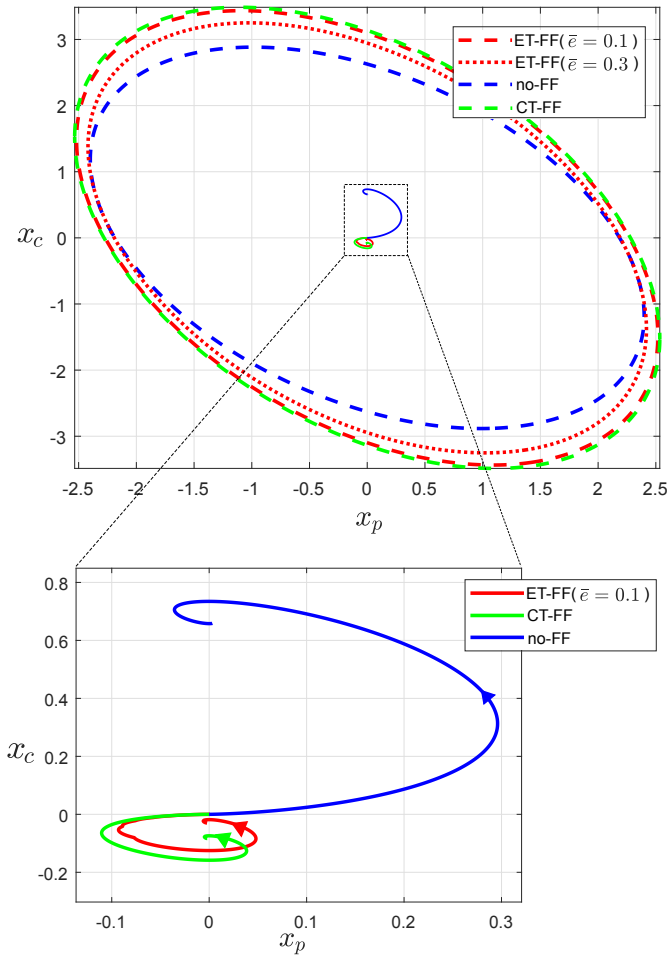

Figure 5.6: Stability regions (dashed line, dotted lines) and the trajectories (solid lines) with the disturbance (5.46) of three cases: (i) event-based feedforward control (red, ET-FF), (ii) continuous-time feedforward control (green, CT-FF), and (iii) no feedforward control (blue, no-FF)

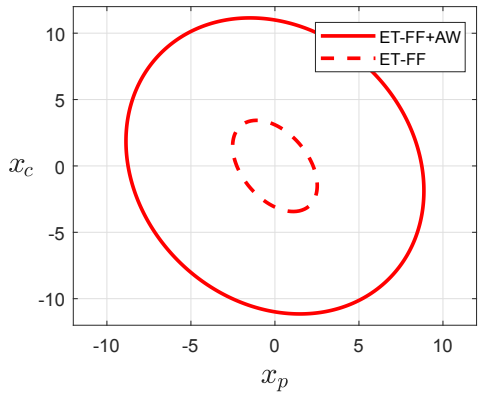

Figure 5.7: Stability regions of event-based feedforward control: (i) with anti-windup compensation (solid line, ET-FF+AW), (ii) without anti-windup compensation (dashed line, ET-FF)

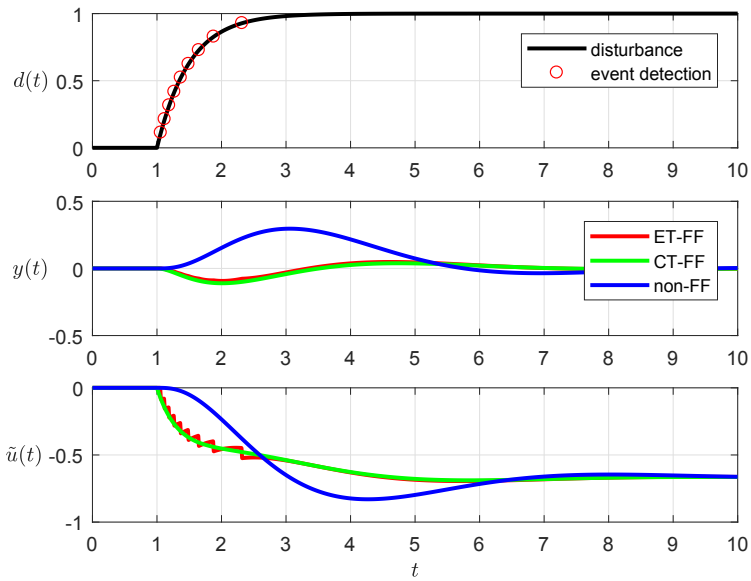

Figure 5.8: Responses against the disturbance (5.46). Top: Disturbance and triggering times. Middle: Outputs of three cases; (i) with event-based feedforward control (red), (ii) with continuous-time feedworward control (green), and (iii) no feedforward control (blue). Bottom: Inputs of the same three cases.

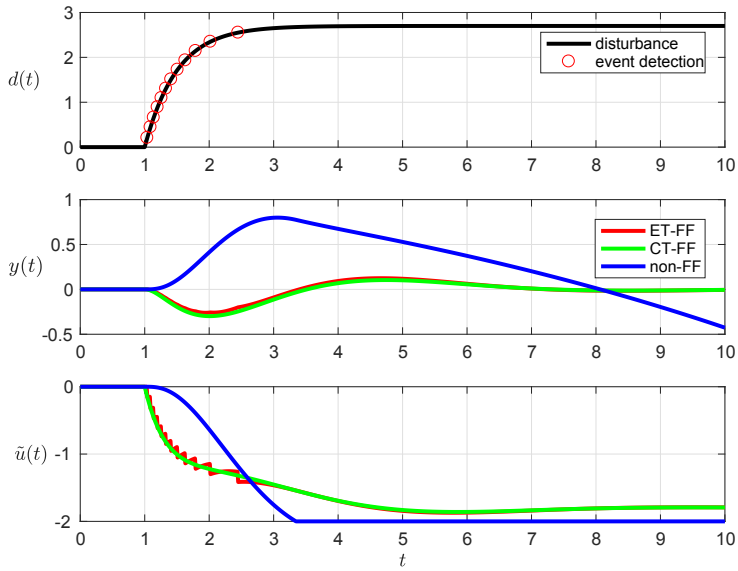

Figure 5.9: Responses against the disturbance (5.47). Top: Disturbance and triggering times. Middle: Outputs of three cases; (i) with event-based feedforward control (red), (ii) with continuous-time feedworward control (green), and (iii) no feedforward control (blue). Bottom: Inputs of the same three cases.

the continuous-time feedforward control. In Figure 5.6, the trajectories of the three cases converge to different equilibrium points. This difference comes from the feedforward gain k_f, and leads to the performance improvement.

The response against a severer disturbance

$$d(t) = 2.8(1 - \mathrm{e}^{-0.5(t+1)}) \tag{5.47}$$

with a threshold $\bar{e} = 0.2$ is illustrated in Figure 5.9. With this disturbance, it is shown that the non-feedforward control case (blue) is not stable. The output diverges to infinity because of actuator saturation. However, the outputs with the event-based feedforward control (red) and continuous-time feedforward control (green) are still stable. The effect of the disturbance can be eliminated around after 7 seconds. It can be noted that the event-based feedforward control with only 12 samplings has no performance degradation compared to the continuous-time one.

The anti-windup compensation is implemented in the system where there is a temporal

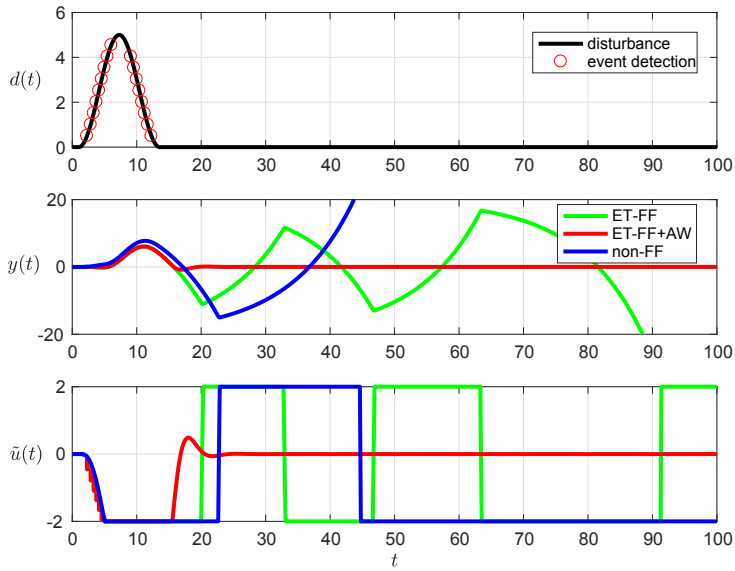

Figure 5.10: Responses against the disturbance (5.48). Top: Disturbance and triggering times. Middle: Outputs of three cases; (i) with event-based feedforward control (red), (ii) with event-based feedworward control and anti-windup compensation(green), and (iii) no feedforward control (blue). Bottom: Inputs of the same three cases.

disturbance

$$d(t) = \begin{cases} 0, & 0 \le t < 1,\ t > 4\pi + 1, \\ 2.5\left(1 + \sin\left(\frac{t-1}{2} - \frac{\pi}{2}\right)\right), & 1 \le t \le 4\pi + 1. \end{cases} \tag{5.48}$$

The responses of the three cases are illustrated in Figure 5.10. The output of the non-feedforward control case (blue) diverges due to actuator saturation. The input sticks on the lower bound around after 45 seconds. The output of the event-based feedforward control without anti-windup compensation (green) also diverges. The input goes back and forth between the upper and the lower bounds. However, it can be seen that the anti-windup compensation mitigates the effect of actuator saturation. The output of feedforward control with the anti-windup compensation (red) comes back to the origin even though the input is saturated for around 10 seconds after the disturbance appears. The anti-windup compensation acts to strip off the input signal from the lower bound, then the input signal comes back to the controllable band around 15 seconds.

5.7 Summary

In this chapter, we investigated event-based feedforward control and ratio control under actuator saturation. As a main result, LMI conditions were derived to determine the stability regions of the control loop with the event-based feedforward control and ratio control. We also considered anti-windup compensation for event-based feedforward control and derived the stability conditions. Numerical examples showed that event-based feedforward control was able to significantly reduce the communication with no performance degradation. In addition, we found that anti-windup compensation was effective for event-based feedforward control to reduce the effect of actuator saturation.

Chapter 6

Conclusions and Future Research

In this chapter, we summarize the main results presented in Chapters 3–5 and discuss possible directions for future research.

6.1 Conclusions

In this thesis, we focused on two problems: network scheduling of a multi-hop network for remote estimation (Chapter 3) and for optimal control (Chapter 4), and event-based feedforward control (Chapter 5). These problems were proposed to reduce energy consumption of wireless sensors and actuators in process plants.

In Chapter 3, we discussed network scheduling of a multi-hop wireless sensor network for remote estimation. We formulated an optimization problem which minimizes an infinite time averaged estimation error covariance subject to sensor energy constraints. Under some assumptions, we showed that activated links in an optimal network schedule formed a tree graph with a root node at the gateway. By using this fact and by transforming the optimization problem to a finite-state MDP, a periodic optimal solution could be found. To reduce the computational load when searching for an optimal schedule, we proposed two alternative algorithms. In the numerical examples of a small and a slightly larger network, we saw that the proposed suboptimal algorithms were effective.

In Chapter 4, we proposed a co-design framework of sensor scheduling, routing, and LQG control over a multi-hop wireless sensor and actuator network, when the controllers were co-located with the corresponding actuators. In this framework, we showed that an optimal schedules can be obtained by distributed covariance-based threshold policies and the optimal routes can be derived by solving the minimum-cost path problem between each sensor–actuator pair. In addition, we showed that a standard LQG controller can be used as the optimal controller. We also provided algorithms to be implemented at sensors and actuators to let them switch routes and schedules locally when a link in the network is disconnected. The theoretical results and the proposed algorithms were illustrated in a numerical example.

In Chapter 5, we discussed event-based feedforward control and event-based ratio control. For both, we derived stability conditions in terms of LMIs when the system is subject

to actuator saturation. We also discussed anti-windup compensation for the event-based feedforward control and derived stability conditions. A numerical example illustrated that the event-based feedforward control with few samplings over the disturbance transient was effective for disturbance rejection. Furthermore, it was demonstrated that anti-windup compensation could dramatically improve the performance of the closed-loop system.

6.2 Future Research Directions

There are several interesting research directions based on the work presented in this thesis. We summarize some possible extensions in this section.

Our frameworks in Chapter 3 and 4 consider a relatively small network so that we can use as many timeslots as we need in a single superframe. This made the problems simpler since scheduled sensor data can be reached to a remote estimator or the corresponding controller before the next sampling time. However, the number of timeslots in a single superframe is limited in existing protocols such as WirelessHART and ISA-100. We are planning to consider larger networks in which sensor data may not be able to reach an estimator or a controller in a single superframe. In this case, an optimal network schedule in Chapter 3 does not necessarily form a tree graph. In Chapter 4, we cannot then obtain optimal routes and schedules independently for each control loop. This could lead to increased computational complexity.

The wireless environment in process plants may change significantly over a period of hours [120]. Thus, optimal schedules and routes also may change. Time-varying network environment is modeled by a semi-Markov chain in [62]. In Chapter 4, we proposed algorithms to switch schedule and route when a link is unavailable. The algorithms can be improved by introducing a semi-Markov chain to make our framework more tolerant against network environment change.

As an extension of the work in Chapter 5, it is interesting to cover some other control architectures used in industrial process control systems. For example, an important direction is to consider event-based cascade control, split-range control, override control, and decoupling control [123]. In addition, joint design synthesis for the feedforward controller, anti-windup compensation, and event-generating condition, needs to be studied.

Bibliography

[1] T. Samad, P. McLaughlin, and J. Lu, "System architecture for process automation: Review and trends," *J. of Process Control*, vol. 17, no. 3, pp. 191–201, 2007.

[2] D. Chen, M. Nixon, and A. Mok, *WirelessHART: Real-Time Mesh Network for Industrial Automation*. Springer, 2010.

[3] International Society of Automation, "Wireless systems for industrial automation: Process control and related applications, ISA-100.11a-2009," 2009.

[4] IEEE 802.15.4 Standard: Wireless Medium Access Control (MAC) and Pysical Layer (PHY) Specification for Low-Rate Wireless Personal Area Networks (WPANs). [Online]. Available: http://www.ieee802.org/15/pub/TG4.html

[5] A. J. Isaksson, I. Harjunkoski, and G. Sand, "The impact of digitalization on the future of control and operations," *Computers and Chemical Engineering*, pp. 122–129, 2017.

[6] P. Park, S. C. Ergen, C. Fischione, C. Lu, and K. H. Johansson, "Wireless network design for control systems: A survey," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 2, pp. 978–1013, 2018.

[7] J. R. Moyne and D. M. Tilbury, "The emergence of industrial control networks for manufacturing control, diagnostics, and safety data," *Proc. of the IEEE*, vol. 95, no. 1, pp. 29–47, 2007.

[8] C. Lu, A. Saifullah, B. Li, M. Sha, H. Gonzalez, D. Gunatilaka, C. Wu, L. Nie, and Y. Chen, "Real-time wireless sensor-actuator networks for industrial cyber-physical systems," *Proc. of the IEEE*, vol. 104, no. 5, pp. 1013–1024, 2016.

[9] W. Zhang, M. S. Branicky, and S. M. Phillips, "Stability of networked control systems," *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 84–99, 2001.

[10] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. S. Sastry, "Foundations of control and estimation over lossy networks," *Proc. of the IEEE*, vol. 95, no. 1, pp. 163–187, 2007.

[11] G. N. Nair, F. Fagnani, S. Zampieri, and R. J. Evans, "Feedback control under data rate constraints: An overview," *Proc. of the IEEE*, vol. 95, no. 1, pp. 108–137, 2007.

[12] H. Forbes, "ExxonMobil's quest for the future of process automation," *ARC Insights, ARC Advisory Group*, 2016.

[13] I. Verhappen and A. Pereira, *Foundation Fieldbus*. ISA, 2008.

[14] PROFIBUS Standard. [Online]. Available: https://www.profibus.com/

[15] A. Willig, "Recent and emerging topics in wireless industrial communications: A selection," *IEEE Trans. on Industrial Informatics*, vol. 4, no. 2, pp. 102–124, 2008.

[16] A. Willig, K. Matheus, and A. Wolisz, "Wireless technology in industrial networks," *Proceedings of the IEEE*, vol. 93, no. 6, pp. 1130–1151, 2005.

[17] V. C. Gungor and G. P. Hancke, "Industrial wireless sensor networks: Challenges, design principles, and technical approaches." *IEEE Trans. Industrial Electronics*, vol. 56, no. 10, pp. 4258–4265, 2009.

[18] A. A. Kumar S, K. Øvsthus, and L. M. Kristensen, "An industrial perspective on wireless sensor networks—a survey of requirements, protocols, and challenges," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1391–1412, 2014.

[19] S. Petersen and S. Carlsen, "WirelessHART versus ISA100. 11a: The format war hits the factory floor," *IEEE Industrial Electronics Magazine*, vol. 5, no. 4, pp. 23–34, 2011.

[20] Q. Wang and J. Jiang, "Comparative examination on architecture and protocol of industrial wireless sensor network standards," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 2197–2219, 2016.

[21] J. Araújo, Y. Ariba, P. Park, H. Sandberg, and K. H. Johansson, "Control over a hybrid MAC wireless network," in *IEEE Int. Conf. on Smart Grid Communications*, 2010, pp. 197–202.

[22] J. Araújo, M. Mazo, A. Anta, P. Tabuada, and K. H. Johansson, "System architectures, protocols and algorithms for aperiodic wireless control systems," *IEEE Trans. on Industrial Informatics*, vol. 10, no. 1, pp. 175–184, 2014.

[23] A. I. Maass, D. Nešić, R. Postoyan, P. M. Dower, and V. S. Varma, "Emulation-based stabilisation of networked control systems over wirelesshart," in *Proc. of IEEE Conf. on Decision and Control*, 2017, pp. 6628–6633.

[24] J. P. Hespanha, P. Naghshtabrizi, and Y. Xu, "A survey of recent results in networked control systems," *Proc. of the IEEE*, vol. 95, no. 1, pp. 138–162, 2007.

[25] J. Lunze, *Control theory of digitally networked dynamic systems*. Springer, 2014.

[26] V. Gupta, D. Spanos, B. Hassibi, and R. M. Murray, "On LQG control across a stochastic packet-dropping link," in *Proc. of American Control Conf.*, 2005, pp. 360–365.

[27] V. Gupta, B. Hassibi, and R. M. Murray, "Optimal LQG control across packet-dropping links," *Systems & Control Letters*, vol. 56, no. 6, pp. 439–446, 2007.

[28] V. Gupta, N. C. Martins, and J. S. Baras, "Optimal output feedback control using two remote sensors over erasure channels," *IEEE Trans. on Automatic Control*, vol. 54, no. 7, pp. 1463–1476, 2009.

[29] V. Gupta, A. F. Dana, J. P. Hespanha, R. M. Murray, and B. Hassibi, "Data transmission over networks for estimation and control," *IEEE Trans. on Automatic Control*, vol. 54, no. 8, pp. 1807–1819, 2009.

[30] V. Gupta and N. C. Martins, "On stability in the presence of analog erasure channel between the controller and the actuator," *IEEE Trans. on Automatic Control*, vol. 55, no. 1, pp. 175–179, 2010.

[31] B. Demirel, V. Gupta, D. E. Quevedo, and M. Johansson, "On the trade-off between communication and control cost in event-triggered dead-beat control," *IEEE Trans. on Automatic Control*, vol. 62, no. 6, pp. 2973–2980, 2017.

[32] D. Hristu-Varsakelis and L. Zhang, "LQG control of networked control systems with access constraints and delays," *Int. J. of Control*, vol. 81, no. 8, pp. 1266–1280, 2008.

[33] D. Maity, M. H. Mamduhi, S. Hirche, K. H. Johansson, and J. S. Baras, "Optimal LQG control under delay-dependent costly information," *IEEE Control Systems Letters*, vol. 3, no. 1, pp. 102–107, 2019.

[34] S. Tatikonda and S. Mitter, "Control under communication constraints," *IEEE Trans. on Automatic Control*, vol. 49, no. 7, pp. 1056–1068, 2004.

[35] T. Tanaka, P. M. Esfahani, and S. K. Mitter, "LQG control with minimum directed information," in *Proc. of IEEE Conf. on Decision and Control*, 2016, pp. 7359–7364.

[36] ——, "LQG control with minimum directed information: Semidefinite programming approach," *IEEE Trans. on Automatic Control*, vol. 63, no. 1, pp. 37–52, 2018.

[37] T. Tanaka and H. Sandberg, "Sdp-based joint sensor and controller design for information-regularized optimal LQG control," in *Proc. of IEEE Conf. on Decision and Control*, 2015, pp. 4486–4491.

[38] G. N. Nair, "A nonstochastic information theory for communication and state estimation," *IEEE Trans. on Automatic Ccontrol*, vol. 58, no. 6, pp. 1497–1510, 2013.

[39] ——, "Nonstochastic information concepts for estimation and control," in *Proc. of IEEE Conf. on Decision and Control*, 2015, pp. 45–56.

[40] A. Molin and S. Hirche, "On LQG joint optimal scheduling and control under communication constraints," in *Proc. of IEEE Conf. on Decision and Control*, 2009, pp. 5832–5838.

[41] ——, "On the optimality of certainty equivalence for event-triggered control systems," *IEEE Trans. on Automatic Control*, vol. 58, no. 2, pp. 470–474, 2013.

[42] ——, "Price-based adaptive scheduling in multi-loop control systems with resource constraints," *IEEE Trans. on Automatic Control*, vol. 59, no. 12, pp. 3282–3295, 2014.

[43] C. Ramesh, H. Sandberg, and K. H. Johansson, "Design of state-based schedulers for a network of control loops," *IEEE Trans. on Automatic Control*, vol. 58, no. 8, pp. 1962–1975, 2013.

[44] ——, "Multiple access with attention-based tournaments for monitoring over wireless networks," in *Proc. of European Control Conf.*, 2009, pp. 4302–4307.

[45] Y. Xu and J. P. Hespanha, "Estimation under uncontrolled and controlled communications in networked control systems," in *Proc. of IEEE Conf. on Decision and Control and European Control Conf.*, 2005, pp. 842–847.

[46] V. Gupta, T. H. Chung, B. Hassibi, and R. M. Murray, "On a stochastic sensor selection algorithm with applications in sensor scheduling and sensor coverage," *Automatica*, vol. 42, no. 2, pp. 251–260, 2006.

[47] Y. Mo, R. Ambrosino, and B. Sinopoli, "Sensor selection strategies for state estimation in energy constrained wireless sensor networks," *Automatica*, vol. 47, no. 7, pp. 1330–1338, 2011.

[48] L. Shi and H. Zhang, "Scheduling two gauss–markov systems: An optimal solution for remote state estimation under bandwidth constraint," *IEEE Trans. on Signal Processing*, vol. 60, no. 4, pp. 2038–2042, 2012.

[49] D. Han, J. Wu, H. Zhang, and L. Shi, "Optimal sensor scheduling for multiple linear dynamical systems," *Automatica*, vol. 75, pp. 260–270, 2017.

[50] S. Wu, X. Ren, S. Dey, and L. Shi, "Optimal scheduling of multiple sensors with packet length constraint," in *Proc. of IFAC World Congress*, 2017, pp. 14 430–14 435.

[51] J. Wu, Q.-S. Jia, K. H. Johansson, and L. Shi, "Event-based sensor data scheduling: Trade-off between communication rate and estimation quality," *IEEE Trans. on Automatic Control*, vol. 58, no. 4, pp. 1041–1046, 2013.

[52] J. Wu, K. H. Johansson, and L. Shi, "A stochastic online sensor scheduler for remote state estimation with time-out condition," *IEEE Trans. on Automatic Control*, vol. 59, no. 11, pp. 3110–3116, 2014.

[53] D. Han, Y. Mo, J. Wu, S. Weerakkody, B. Sinopoli, and L. Shi, "Stochastic event-triggered sensor schedule for remote state estimation," *IEEE Trans. on Automatic Control*, vol. 60, no. 10, pp. 2661–2675, 2015.

[54] S. Weerakkody, Y. Mo, B. Sinopoli, D. Han, and L. Shi, "Multi-sensor scheduling for state estimation with event-based, stochastic triggers," *IEEE Trans. on Automatic Control*, vol. 61, no. 9, pp. 2695–2701, 2016.

[55] S. Trimpe and R. D'Andrea, "Event-based state estimation with variance-based triggering," *IEEE Trans. on Automatic Control*, vol. 59, no. 12, pp. 3266–3281, 2014.

[56] A. S. Leong, S. Dey, and D. E. Quevedo, "On the optimality of threshold policies in event triggered estimation with packet drops," in *Proc. of European Control Conf.*, 2015, pp. 927–933.

[57] ——, "Sensor scheduling in variance based event triggered estimation with packet drops," *IEEE Trans. on Automatic Control*, vol. 62, no. 4, pp. 1880–1895, 2017.

[58] A. S. Leong, D. E. Quevedo, T. Tanaka, S. Dey, and A. Ahlén, "Event-based transmission scheduling and LQG control over a packet dropping link," in *Proc. of IFAC World Congress*, 2017, pp. 8945–8950.

[59] T. Soleymani, S. Zoppi, M. Vilgelm, S. Hirche, W. Kellerer, and J. Baras, "Covariance-based transmission power control for estimation over wireless sensor networks," in *Proc. of European Control Conf.*, 2018.

[60] M. Xia, V. Gupta, and P. J. Antsaklis, "Networked state estimation over a shared communication medium," *IEEE Trans. on Automatic Control*, vol. 62, no. 4, pp. 1729–1741, 2017.

[61] D. E. Quevedo, A. Ahlen, and K. H. Johansson, "State estimation over sensor networks with correlated wireless fading channels," *IEEE Trans. on Automatic Control*, vol. 58, no. 3, pp. 581–593, 2013.

[62] A. S. Leong, D. E. Quevedo, A. Ahlén, and K. H. Johansson, "On network topology reconfiguration for remote state estimation." *IEEE Trans. Automatic Control*, vol. 61, no. 12, pp. 3842–3856, 2016.

[63] D. E. Quevedo, K. H. Johansson, A. Ahlén, and I. Jurado, "Adaptive controller placement for wireless sensor–actuator networks with erasure channels," *Automatica*, vol. 49, no. 11, pp. 3458–3466, 2013.

[64] A. S. Leong, S. Dey, G. N. Nair, and P. Sharma, "Power allocation for outage minimization in state estimation over fading channels," *IEEE Trans. on Signal Processing*, vol. 59, no. 7, pp. 3382–3397, 2011.

[65] K. Gatsis, A. Ribeiro, and G. J. Pappas, "Optimal power management in wireless control systems," *IEEE Trans. on Automatic Control*, vol. 59, no. 6, pp. 1495–1510, 2014.

[66] X. Ren, J. Wu, K. H. Johansson, G. Shi, and L. Shi, "Infinite horizon optimal transmission power control for remote state estimation over fading channels," *IEEE Trans. on Automatic Control*, vol. 63, no. 1, pp. 85–100, 2018.

[67] A. S. Leong and S. Dey, "Power allocation for error covariance minimization in kalman filtering over packet dropping links," in *Proc. of IEEE Conf. on Decision and Control*, 2012, pp. 3335–3340.

[68] D. E. Quevedo, A. Ahlén, A. S. Leong, and S. Dey, "On kalman filtering over fading wireless channels with controlled transmission powers," *Automatica*, vol. 48, no. 7, pp. 1306–1316, 2012.

[69] A. Nayyar, T. Başar, D. Teneketzis, and V. V. Veeravalli, "Optimal strategies for communication and remote estimation with an energy harvesting sensor," *IEEE Trans. on Automatic Control*, vol. 58, no. 9, pp. 2246–2260, 2013.

[70] M. Nourian, A. S. Leong, and S. Dey, "Optimal energy allocation for kalman filtering over packet dropping links with imperfect acknowledgments and energy harvesting constraints," *IEEE Trans. on Automatic Control*, vol. 59, no. 8, pp. 2128–2143, 2014.

[71] M. Kishida, "Event-triggered control for discrete-time nonlinear systems using state-dependent Riccati equation," in *Proc. of European Control Conf.*, 2018, pp. 1499–1504.

[72] M. Eisen, K. Gatsis, G. J. Pappas, and A. Ribeiro, "Learning in wireless control systems over non-stationary channels," *arXiv preprint arXiv:1803.01078*, 2018.

[73] B. Demirel, A. Ramaswamy, D. E. Quevedo, and H. Karl, "DeepCAS: A deep reinforcement learning algorithm for control-aware scheduling," *arXiv preprint arXiv:1803.02998*, 2018.

[74] W. M. H. Heemels, A. R. Teel, N. Van de Wouw, and D. Nesic, "Networked control systems with communication constraints: Tradeoffs between transmission intervals, delays and performance," *IEEE Trans. on Automatic control*, vol. 55, no. 8, pp. 1781–1796, 2010.

[75] K. J. Åström and B. Wittenmark, *Computer-controlled Systems: Theory and Design*. Prentice-Hall, 2013.

[76] R. Goebel, R. G. Sanfelice, and A. R. Teel, "Hybrid dynamical systems," *IEEE Control Systems Magazine*, vol. 29, no. 2, pp. 28–93, 2009.

[77] E. Fridman, *Introduction to time-delay systems: Analysis and control*. Springer, 2014.

[78] G. C. Walsh, H. Ye, and L. G. Bushnell, "Stability analysis of networked control systems," *IEEE Trans. on Control Systems Technology*, vol. 10, no. 3, pp. 438–446, 2002.

[79] K. Liu, E. Fridman, and L. Hetel, "Network-based control via a novel analysis of hybrid systems with time-varying delays," in *Proc. of IEEE Conf. on Decision and Control*, 2012, pp. 3886–3891.

[80] ——, "Network-based control under round-robin scheduling and quantization," in *Proc. of IFAC Symposium on Robust Control Design*, 2012, pp. 91–96.

[81] M. Donkers, W. Heemels, N. Van de Wouw, and L. Hetel, "Stability analysis of networked control systems using a switched linear systems approach," *IEEE Trans. on Automatic Control*, vol. 56, no. 9, pp. 2101–2115, 2011.

[82] K.-E. Årzén, "A simple event-based PID controller," in *Proc. of IFAC World Congress*, vol. 18, 1999, pp. 423–428.

[83] K. J. Astrom and B. M. Bernhardsson, "Comparison of riemann and lebesgue sampling for first order stochastic systems," in *Proc. of IEEE Conf. on Decision and Control and European Control Conf.*, vol. 2, 2002, pp. 2011–2016.

[84] W. Heemels, K. H. Johansson, and P. Tabuada, "An introduction to event-triggered and self-triggered control," in *Proc. of IEEE Conf. on Decision and Control*, 2012, pp. 3270–3285.

[85] D. V. Dimarogonas, E. Frazzoli, and K. H. Johansson, "Distributed event-triggered control for multi-agent systems," *IEEE Trans. on Automatic Control*, vol. 57, no. 5, pp. 1291–1297, 2012.

[86] D. Ogawa and T. Hayakawa, "Adaptive control for linear uncertain discrete-time systems with event-triggered mechanisms," in *Proc. of IFAC World Congress*, 2017, pp. 10 108–10 113.

[87] P. Tabuada, "Event-triggered real-time scheduling of stabilizing control tasks," *IEEE Trans. on Automatic Control*, vol. 52, no. 9, pp. 1680–1685, 2007.

[88] W. Heemels, J. Sandee, and P. van den Bosch, "Analysis of event-driven controllers for linear systems," *Int. J. of Control*, vol. 81, no. 4, pp. 571–590, 2008.

[89] J. Lunze and D. Lehmann, "A state-feedback approach to event-based control," *Automatica*, vol. 46, no. 1, pp. 211–215, 2010.

[90] W. Wu, S. Reimann, D. Görges, and S. Liu, "Suboptimal event-triggered control for time-delayed linear systems," *IEEE Trans. on Automatic Control*, vol. 60, no. 5, pp. 1386–1391, 2015.

[91] A. Selivanov and E. Fridman, "Sampled-data implementation of derivative-dependent control using artificial delays," *arXiv preprint arXiv:1801.05696*, 2018.

[92] M. Rabi and K. H. Johansson, "Event-triggered strategies for industrial control over wireless networks," in *Proc. of Int. Conf. on Wireless Internet*, 2008, pp. 1–7.

[93] G. A. Kiener, D. Lehmann, and K. H. Johansson, "Actuator saturation and anti-windup compensation in event-triggered control," *Discrete Event Dynamic Systems*, vol. 24, no. 2, pp. 173–197, 2014.

[94] U. Tiberi, J. Araújo, and K. H. Johansson, "On event-based PI control of first-order processes," in *Proc. of IFAC Conf. on Advances in PID Control*, 2012, pp. 448–453.

[95] D. Lehmann and J. Lunze, "Extension and experimental evaluation of an event-based state-feedback approach," *Control Engineering Practice*, vol. 19, no. 2, pp. 101–112, 2011.

[96] T. Norgren, J. Styrud, A. J. Isaksson, J. Åkerberg, and T. Lindh, "Industrial evaluation of process control using non-periodic sampling," in *Proc. of the 17th IEEE Conf. on Emerging Technologies and Factory Automation*, 2012, pp. 1–8.

[97] C.-F. Lindberg and A. J. Isaksson, "Comparison of different sampling schemes for wireless control subject to packet losses," in *Proc. of Int. Conf. on Event-based Control, Communication, and Signal Processing*, 2015, pp. 1–8.

[98] T. Blevins, D. Chen, M. Nixon, and W. Wojsznis, *Wireless Control Foundation: Continuous and Discrete Control for the Process Industry*. International Society of Automation, 2015.

[99] D. Hasenfratz, A. Meier, C. Moser, J.-J. Chen, and L. Thiele, "Analysis, comparison, and optimization of routing protocols for energy harvesting wireless sensor networks," in *Proc. of IEEE Int. Conf. on Sensor Networks, Ubiquitous, and Trustworthy Computing*, 2010, pp. 19–26.

[100] M. Sha, D. Gunatilaka, C. Wu, and C. Lu, "Empirical study and enhancements of industrial wireless sensor-actuator network protocols," *IEEE Internet of Things J.*, vol. 4, no. 3, pp. 696–704, 2017.

[101] O. Chipara, Z. He, G. Xing, Q. Chen, X. Wang, C. Lu, J. Stankovic, and T. Abdelzaher, "Real-time power-aware routing in sensor networks," in *Proc. of IEEE Int. Workshop on Quality of Service*, 2006, pp. 83–92.

[102] S. T. Jawaid and S. L. Smith, "Submodularity and greedy algorithms in sensor scheduling for linear dynamical systems," *Automatica*, vol. 61, pp. 282–288, 2015.

[103] H. Zhang, R. Ayoub, and S. Sundaram, "Sensor selection for optimal filtering of linear dynamical systems: Complexity and approximation," in *Proc. of IEEE Conf. on Decision and Control*, 2015, pp. 5002–5007.

[104] ——, "Sensor selection for kalman filtering of linear dynamical systems: Complexity, limitations and greedy algorithms," *Automatica*, vol. 78, pp. 202–210, 2017.

[105] Y. Mo, E. Garone, and B. Sinopoli, "On infinite-horizon sensor scheduling," *Systems & Control Letters*, vol. 67, pp. 65–70, 2014.

[106] L. Zhao, W. Zhang, J. Hu, A. Abate, and C. J. Tomlin, "On the optimal solutions of the infinite-horizon linear sensor scheduling problem," *IEEE Trans. on Automatic Control*, vol. 59, no. 10, pp. 2825–2830, 2014.

[107] S. Wu, X. Ren, S. Dey, and L. Shi, "Optimal scheduling of multiple sensors over shared channels with packet transmission constraint," *Automatica*, vol. 96, pp. 22–31, 2018.

[108] ZigBee Alliance, "Zigbee specification," 2006. [Online]. Available: https://www.zigbee.org/download/standards-zigbee-specification/

[109] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.

[110] R. Rajagopalan and P. K. Varshney, "Data aggregation techniques in sensor networks: A survey," *IEEE Communications Surveys & Tutorials*, vol. 8, no. 4, pp. 48–63, 2006.

[111] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proc. of IEEE Hawaii Int. Conf. on System Sciences*, 2000, pp. 1–10.

[112] J. Dou, Z. Guo, J. Cao, and G. Zhang, "Data aggregation rate assumption for wireless sensors," in *Proc. of IEEE Inter. Conf. on Radio-Frequency Integration Technology*, 2007, pp. 278–281.

[113] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 2005.

[114] D. P. Bertsekas, *Dynamic Programming and Optimal Control, Volume II*. Athena Scientific, 2005.

[115] A. Molin and S. Hirche, "Structural characterization of optimal event-based controllers for linear stochastic systems," in *Proc. of IEEE Conf. on Decision and Control*, 2010, pp. 3227–3233.

[116] T. Iwaki, Y. Wu, J. Wu, H. Sansberg, and K. H. Johansson, "Wireless sensor network scheduling for remote estimation under energy constraints," in *Proc. of IEEE Conf. on Decision and Control*, 2017, pp. 3362–3367.

[117] F. Smarra, A. D'Innocenzo, and M. D. Di Benedetto, "Optimal co-design of control, scheduling and routing in multi-hop control networks," in *Proc. of IEEE Conf. on Decision and Control*, 2012, pp. 1960–1965.

[118] R. Alur, A. d'Innocenzo, K. H. Johansson, G. J. Pappas, and G. Weiss, "Compositional modeling and analysis of multi-hop control networks," *IEEE Trans. on Automatic control*, vol. 56, no. 10, pp. 2345–2357, 2011.

[119] P. Agrawal, A. Ahlén, T. Olofsson, and M. Gidlund, "Characterization of long term channel variations in industrial wireless sensor networks," in *IEEE Int. Conf. on Communications*, 2014, pp. 1–6.

[120] ——, "Long term channel characterization for energy efficient transmission in industrial environments." *IEEE Trans. Communications*, vol. 62, no. 8, pp. 3004–3014, 2014.

[121] H. Ö. Tan and İ. Körpeoğlu, "Power efficient data gathering and aggregation in wireless sensor networks," *ACM Sigmod Record*, vol. 32, no. 4, pp. 66–71, 2003.

[122] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*. Dover, 1998.

[123] D. E. Seborg, D. A. Mellichamp, T. F. Edgar, and F. J. Doyle III, *Process Dynamics and Control*. John Wiley & Sons, 2010.

[124] K. J. Åström and T. Hägglund, *Advanced PID Control*. International Society of Automation, 2006.

[125] S. Reimann, W. Wu, and S. Liu, "PI control and scheduling design for embedded control systems," in *Proc. of IFAC World Congress*, 2014, pp. 11 111–11 116.

[126] J. G. Silva Jr., W. F. Lages da, and D. Sbarbaro, "Event-triggered PI control design." in *Proc. of IFAC World Congress*, 2014, pp. 6947–6952.

[127] S. Reimann, D. H. Van, S. Al-Areqi, and S. Liu, "Stability analysis and PI control synthesis under event-triggered communication," in *Proc. of European Control Conf.*, 2015, pp. 2174–2179.

[128] L. Moreira, L. Groff, J. G. da Silva, and S. Tarbouriech, "Event-triggered PI control for continuous plants with input saturation," in *Proc. of American Control Conf.*, 2016, pp. 4251–4256.

[129] J. Song, A. K. Mok, D. Chen, M. Nixon, T. Blevins, and W. Wojsznis, "Improving PID control with unreliable communications," in *Proc. of ISA EXPO Technical Conf.*, 2006, pp. 17–19.

[130] O. Kaltiokallio, L. M. Eriksson, and M. Bocca, "On the performance of the PIDPLUS controller in wireless control systems," in *Proc. of Mediterranean Conf. on Control and Automation*, 2010, pp. 707–714.

[131] T. Blevins, M. Nixon, and W. Wojsznis, "PID control using wireless measurements," in *Proc. of American Control Conf.*, 2014, pp. 790–795.

[132] D. Lehmann, G. A. Kiener, and K. H. Johansson, "Event-triggered PI control: Saturating actuators and anti-windup compensation," in *Proc. of IEEE Conf. on Decision and Control*, 2012, pp. 6566–6571.

[133] D. Lehmann and K. H. Johansson, "Event-triggered PI control subject to actuator saturation," in *Proc. of IFAC Conf. on Advances in PID Control*, 2012, pp. 430–435.

[134] S. Tarbouriech, G. Garcia, J. M. G. da Silva Jr, and I. Queinnec, *Stability and Stabilization of Linear Systems with Saturating Actuators*. Springer, 2011.

[135] J. Löfberg, "Yalmip: A toolbox for modeling and optimization in matlab," in *Proc. of IEEE Int. Symp. on Computer Aided Control Systems Design*, 2004, pp. 284–289.